Contents

[.\concentrador\src\br\ufrj\coppetec\concentrador\Blinker.java 6](#_Toc515467585)

[.\concentrador\src\br\ufrj\coppetec\concentrador\Concentrador.java 8](#_Toc515467586)

[.\concentrador\src\br\ufrj\coppetec\concentrador\DocumentSizeFilter.java 11](#_Toc515467587)

[.\concentrador\src\br\ufrj\coppetec\concentrador\Janela.java 13](#_Toc515467588)

[.\concentrador\src\br\ufrj\coppetec\concentrador\LoginController.java 184](#_Toc515467589)

[.\concentrador\src\br\ufrj\coppetec\concentrador\LoginJanela.java 188](#_Toc515467590)

[.\concentrador\src\br\ufrj\coppetec\concentrador\package-info.java 193](#_Toc515467591)

[.\concentrador\src\br\ufrj\coppetec\concentrador\PropertyLoader.java 194](#_Toc515467592)

[.\concentrador\src\br\ufrj\coppetec\concentrador\Splash.java 196](#_Toc515467593)

[.\concentrador\src\br\ufrj\coppetec\concentrador\Util.java 199](#_Toc515467594)

[.\concentrador\src\br\ufrj\coppetec\concentrador\WebServer.java 207](#_Toc515467595)

[.\concentrador\src\br\ufrj\coppetec\concentrador\database\Db.java 210](#_Toc515467596)

[.\concentrador\src\br\ufrj\coppetec\concentrador\database\DBFileImporter.java 213](#_Toc515467597)

[.\concentrador\src\br\ufrj\coppetec\concentrador\database\ImportedDB.java 216](#_Toc515467598)

[.\concentrador\src\br\ufrj\coppetec\concentrador\database\myDB.java 224](#_Toc515467599)

[.\concentrador\src\br\ufrj\coppetec\concentrador\database\PVKey.java 238](#_Toc515467600)

[.\concentrador\src\br\ufrj\coppetec\concentrador\database\PVregister.java 239](#_Toc515467601)

[.\concentrador\src\br\ufrj\coppetec\concentrador\exporter\JSONBuilder.java 240](#_Toc515467602)

[.\concentrador\src\br\ufrj\coppetec\concentrador\exporter\JSONExporter.java 241](#_Toc515467603)

[.\concentrador\src\br\ufrj\coppetec\concentrador\exporter\OdJSONbuilder.java 246](#_Toc515467604)

[.\concentrador\src\br\ufrj\coppetec\concentrador\exporter\PvJSONbuilder.java 250](#_Toc515467605)

[.\concentrador\src\br\ufrj\coppetec\concentrador\exporter\Zipper.java 253](#_Toc515467606)

[.\pnctcheck\src\main\java\br\ufrj\coppetec\pnct\pnctcheck\PNCTCheck.java 254](#_Toc515467607)

[.\pnctcheck\src\test\java\br\ufrj\coppetec\pnct\pnctcheck\AppTest.java 258](#_Toc515467608)

[.\mobile\platforms\ios\www\controllers\pergunta\textunderscore extra.js 259](#_Toc515467609)

[.\mobile\platforms\ios\www\js\index.js 261](#_Toc515467610)

[.\mobile\platforms\ios\www\js\logins.js 288](#_Toc515467611)

[.\mobile\platforms\ios\www\js\myDb.js 330](#_Toc515467612)

[.\mobile\platforms\ios\www\js\produtos.js 338](#_Toc515467613)

[.\mobile\www\controllers\confirmar\textunderscore veiculo.js 364](#_Toc515467614)

[.\mobile\www\controllers\menu.js 366](#_Toc515467615)

[.\mobile\www\controllers\pergunta\textunderscore extra.js 369](#_Toc515467616)

[.\mobile\www\controllers\selecionar\textunderscore tipo.js 371](#_Toc515467617)

[.\mobile\www\controllers\sumario.js 372](#_Toc515467618)

[.\mobile\www\controllers\carga\caracterizacao\textunderscore carga.js 374](#_Toc515467619)

[.\mobile\www\controllers\carga\caracterizacao\textunderscore viagem2\textunderscore carga.js 377](#_Toc515467620)

[.\mobile\www\controllers\carga\caracterizacao\textunderscore viagem\textunderscore carga.js 385](#_Toc515467621)

[.\mobile\www\controllers\carga\identificacao\textunderscore visual\textunderscore carga.js 388](#_Toc515467622)

[.\mobile\www\controllers\onibus\caracterizacao\textunderscore onibus.js 394](#_Toc515467623)

[.\mobile\www\controllers\onibus\caracterizacao\textunderscore viagem\textunderscore onibus.js 396](#_Toc515467624)

[.\mobile\www\controllers\onibus\identificacao\textunderscore visual\textunderscore onibus.js 400](#_Toc515467625)

[.\mobile\www\js\carga\textunderscore perigosa.js 402](#_Toc515467626)

[.\mobile\www\js\combustivel.js 443](#_Toc515467627)

[.\mobile\www\js\controllers.js 444](#_Toc515467628)

[.\mobile\www\js\datas.js 445](#_Toc515467629)

[.\mobile\www\js\eventQueue.js 446](#_Toc515467630)

[.\mobile\www\js\index.js 447](#_Toc515467631)

[.\mobile\www\js\ipadID.js 474](#_Toc515467632)

[.\mobile\www\js\jsonWriter.js 477](#_Toc515467633)

[.\mobile\www\js\logger.js 479](#_Toc515467634)

[.\mobile\www\js\logins.js 481](#_Toc515467635)

[.\mobile\www\js\motivoEscolhaRota.js 523](#_Toc515467636)

[.\mobile\www\js\motivoViagem.js 524](#_Toc515467637)

[.\mobile\www\js\myDb.js 525](#_Toc515467638)

[.\mobile\www\js\myDialogs.js 534](#_Toc515467639)

[.\mobile\www\js\paises.js 537](#_Toc515467640)

[.\mobile\www\js\produtos.js 800](#_Toc515467641)

[.\mobile\www\js\propriedadeVeiculo.js 826](#_Toc515467642)

[.\mobile\www\js\util.js 827](#_Toc515467643)

[.\concentrador\novos\textunderscore od\textunderscore 1.json 845](#_Toc515467644)

[.\concentrador\novos\textunderscore volumetrica.json 846](#_Toc515467645)

[.\concentrador\novos\textunderscore volumetrica\textunderscore 1.json 847](#_Toc515467646)

[.\concentrador\todos\textunderscore od.json 849](#_Toc515467647)

[.\concentrador\todos\textunderscore od\textunderscore 8.json 854](#_Toc515467648)

[.\concentrador\todos\textunderscore od\textunderscore 9.json 860](#_Toc515467649)

[.\concentrador\todos\textunderscore volumetrica.json 868](#_Toc515467650)

[.\concentrador\src\logins\users.json 869](#_Toc515467651)

[.\concentrador\src\logins\users\textunderscore fase2\textunderscore old.json 916](#_Toc515467652)

[.\mobile\.esdoc.json 957](#_Toc515467653)

[.\mobile\package-lock.json 958](#_Toc515467654)

[.\mobile\package.json 1008](#_Toc515467655)

[.\mobile\platforms\platforms.json 1009](#_Toc515467656)

[.\mobile\www\css\bootstrap.min.css 1010](#_Toc515467657)

[.\mobile\www\css\bootstrap\textunderscore buttons.css 1017](#_Toc515467658)

[.\mobile\www\css\bootstrap\textunderscore grid.css 1026](#_Toc515467659)

[.\mobile\www\css\font-awesome.min.css 1038](#_Toc515467660)

[.\mobile\www\css\full\textunderscore bootstrap.min.css 1047](#_Toc515467661)

[.\mobile\www\css\index.css 1072](#_Toc515467662)

[.\mobile\www\css\jquery-confirm.css 1075](#_Toc515467663)

[.\mobile\www\css\login.css 1085](#_Toc515467664)

[.\mobile\www\css\reset.css 1087](#_Toc515467665)

[.\mobile\www\css\style.css 1088](#_Toc515467666)

[.\mobile\platforms\ios\www\views\pergunta\textunderscore extra.html 1100](#_Toc515467667)

[.\mobile\www\index.html 1102](#_Toc515467668)

[.\mobile\www\views\confirmar\textunderscore veiculo.html 1105](#_Toc515467669)

[.\mobile\www\views\menu.html 1107](#_Toc515467670)

[.\mobile\www\views\pergunta\textunderscore extra.html 1109](#_Toc515467671)

[.\mobile\www\views\selecionar\textunderscore tipo.html 1111](#_Toc515467672)

[.\mobile\www\views\sumario.html 1115](#_Toc515467673)

[.\mobile\www\views\carga\caracterizacao\textunderscore carga.html 1116](#_Toc515467674)

[.\mobile\www\views\carga\caracterizacao\textunderscore viagem2\textunderscore carga.html 1118](#_Toc515467675)

[.\mobile\www\views\carga\caracterizacao\textunderscore viagem\textunderscore carga.html 1124](#_Toc515467676)

[.\mobile\www\views\carga\identificacao\textunderscore visual.html 1127](#_Toc515467677)

[.\mobile\www\views\onibus\caracterizacao\textunderscore onibus.html 1132](#_Toc515467678)

[.\mobile\www\views\onibus\caracterizacao\textunderscore viagem\textunderscore onibus.html 1134](#_Toc515467679)

[.\mobile\www\views\onibus\identificacao\textunderscore visual.html 1137](#_Toc515467680)

[.\document.py 1139](#_Toc515467681)

[.\concentrador\src\logins\verifica\textunderscore fase\textunderscore 4.py 1142](#_Toc515467682)

[.\concentrador\build.xml 1143](#_Toc515467683)

[.\concentrador\nbproject\build-impl.xml 1146](#_Toc515467684)

[.\concentrador\nbproject\project.xml 1180](#_Toc515467685)

[.\mobile\config.xml 1181](#_Toc515467686)

[.\mobile\nbproject\build.xml 1183](#_Toc515467687)

[.\mobile\nbproject\project.xml 1190](#_Toc515467688)

[.\mobile\nbproject\private\private.xml 1191](#_Toc515467689)

[.\mobile\nbproject\private\retriever\catalog.xml 1192](#_Toc515467690)

[.\mobile\nbproject\private\retriever\www.w3.org\ns\widgets\index.xml 1193](#_Toc515467691)

[.\mobile\platforms\ios\Contagem Mobile\config.xml 1195](#_Toc515467692)

[.\pnctcheck\pom.xml 1197](#_Toc515467693)

[.\concentrador\nbproject\genfiles.properties 1199](#_Toc515467694)

[.\concentrador\nbproject\project.properties 1200](#_Toc515467695)

[.\concentrador\nbproject\configs\concentrador.properties 1203](#_Toc515467696)

[.\mobile\nbproject\plugins.properties 1204](#_Toc515467697)

[.\mobile\nbproject\project.properties 1205](#_Toc515467698)

[.\mobile\nbproject\configs\android.properties 1206](#_Toc515467699)

[.\mobile\nbproject\configs\android\textunderscore 1.properties 1207](#_Toc515467700)

[.\mobile\nbproject\configs\ios.properties 1208](#_Toc515467701)

[.\mobile\nbproject\configs\ios\textunderscore 1.properties 1209](#_Toc515467702)

[.\mobile\nbproject\private\private.properties 1210](#_Toc515467703)

[.\concentrador\src\logins\controle.txt 1211](#_Toc515467704)

## .\concentrador\src\br\ufrj\coppetec\concentrador\Blinker.java

1:/\*\*  
 2:\*@author ludes - PESC - COPPE - ufrj  
 3:\*@author Eduardo Mangeli  
 4:\*@author Marcelo Areas  
 5:\*@author Fabrício Pereira  
 6:\*@author Geraldo Xexéo  
 7:\*  
 8:\* Faz um JTextField piscar  
 9:\*  
 10:\*  
 11:\*  
 12:\*/  
 13:package br.ufrj.coppetec.concentrador;  
 14:  
 15:import java.awt.Color;  
 16:import java.awt.event.ActionEvent;  
 17:import java.awt.event.ActionListener;  
 18:import javax.swing.JTextField;  
 19:  
 20:/\*\*  
 21: \*  
 22: \*  
 23: \* Faz um JTextField piscar  
 24: \*  
 25: \* @author ludes - PESC - COPPE - ufrj  
 26: \* @author Eduardo Mangeli  
 27: \* @author Marcelo Areas  
 28: \* @author Fabrício Pereira  
 29: \* @author Geraldo Xexéo  
 30: \*  
 31: \*/  
 32:public class Blinker implements ActionListener{  
 33: int counter = 0; ///< conta quantos blinks até 10  
 34: JTextField txtField; ///< Mantém o link para o pai  
 35: Color color; ///< cor atual  
 36:  
 37: /\*\*  
 38: \* Construtor precisa receber um JTextField  
 39: \*  
 40: \* param txtField um JTextField que vai piscar  
 41: \*/  
 42: public Blinker(JTextField txtField){  
 43: super();  
 44: this.txtField = txtField;  
 45: this.color = txtField.getBackground();  
 46: }  
 47:  
 48:  
 49: /\*\*  
 50: \* Executa o piscar trocando o background de normal para laranja  
 51: \*/  
 52: @Override  
 53: public void actionPerformed(ActionEvent e) {  
 54: if (counter % 2 == 0){  
 55: txtField.setBackground(Color.ORANGE);  
 56: } else {  
 57: txtField.setBackground(color);  
 58: }  
 59: counter++;  
 60: if (counter>9) counter = 0; // para não explodir  
 61: }  
 62:  
 63:  
 64:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\Concentrador.java

1:package br.ufrj.coppetec.concentrador;  
 2:  
 3:import java.io.IOException;  
 4:import java.text.ParseException;  
 5:import java.util.Properties;  
 6:  
 7:import javax.swing.JOptionPane;  
 8:  
 9:import org.apache.logging.log4j.LogManager;  
 10:import org.apache.logging.log4j.Logger;  
 11:  
 12:import br.ufrj.coppetec.concentrador.database.myDB;  
 13:import java.awt.Color;  
 14:  
 15:/\*\*  
 16: \* Classe principal do sistema, representa o fluxo de inicialização e mantém as informações de identificação do posto   
 17: \*   
 18: \* @author ludes - PESC - COPPE - ufrj  
 19: \* @author Eduardo Mangeli  
 20: \* @author Marcelo Areas  
 21: \* @author Fabrício Pereira  
 22: \* @author Geraldo Xexéo  
 23: \*/  
 24:  
 25:public class Concentrador {  
 26: private static Logger logger = LogManager.getLogger(Concentrador.class);  
 27:  
 28: // static WebServer wServer;  
 29: static Janela janela; ///< Janela da interface com o usuário  
 30: public static myDB database; ///< Classe de conexão com o banco de dados  
 31: public static String trecho; ///< Trecho do posto de pesquisa  
 32: public static String posto; ///< Identificação do posto de pesquisa  
 33: public static String version = "3.6"; ///< Versão do programa  
 34: public static int dbVersion = 2; ///< Versão do banco de dados  
 35: public static Properties configuration; ///< Configurações do programa  
 36: public static boolean treinamento = false; ///< Identificação do modo de treinamento ou pesquisa  
 37:  
 38:   
 39: /\*\*  
 40: \* Método do fluxo principal de execução do sistema.  
 41: \* Exibe splash screen com a versão atual do sistema, carrega configurações, inicia conexão com o banco de dados,   
 42: \* exibe a janela de login e a janela principal da interface do usuário  
 43: \*   
 44: \* @param args o programa é executado sem parâmetros  
 45: \*/  
 46: public static void main(String[] args) {  
 47: StackTraceElement[] trace = Thread.currentThread().getStackTrace();  
 48:   
 49: logger.info("PNCT Concentrador versão: " + version);  
 50:  
 51: Splash splash = new Splash();  
 52: splash.setLocationRelativeTo(null);  
 53: splash.setVersionLabet(version);  
 54: splash.setVisible(true);  
 55:  
 56: // lendo arquivo properties  
 57: try {  
 58: PropertyLoader propLoader = new PropertyLoader();  
 59: configuration = propLoader.getProperties();  
 60: Util.populateInputLimits();  
 61: } catch (IOException ex) {  
 62: logger.catching(ex);  
 63: System.exit(1);  
 64: }  
 65:  
 66: // criando o banco de dados  
 67: try {  
 68: database = myDB.getInstance();  
 69: database.initDatabaseTables();  
 70:  
 71: } catch (Exception e) {  
 72: logger.error("Erro ao acessar o BD.", e);  
 73: JOptionPane.showMessageDialog(null, "Erro acessando o banco de dados: \n" + e.getMessage());  
 74: e.printStackTrace();  
 75: System.exit(1);  
 76: }  
 77:  
 78: janela = new Janela();  
 79:  
 80:   
 81: LoginJanela loginJanela = new LoginJanela(janela, true);  
 82: loginJanela.setLocationRelativeTo(null);  
 83: loginJanela.setVisible(true);  
 84:  
 85: // wait  
 86: try {  
 87: Thread.sleep(2000);  
 88: } catch (InterruptedException ex) {  
 89: Thread.currentThread().interrupt();  
 90: }  
 91:  
 92: splash.setVisible(false);  
 93:   
 94: janela.setTitle(janela.getTituloJanela());  
 95: janela.setAlertaTreinamento();  
 96:   
 97: try {  
 98: janela.initDatesToShow();  
 99: janela.lblPosto\_dados.setText(posto);  
 100: janela.lblTrecho\_dados.setText(trecho);  
 101: janela.setVisible(true);  
 102: } catch (ParseException e) {  
 103: logger.error("Erro ao carregar datas válidas.", e);  
 104: JOptionPane.showMessageDialog(null, "Erro ao carregar datas válidas." + e.getMessage());  
 105: System.exit(1);  
 106: }  
 107: }  
 108:  
 109: /\*\*  
 110: \* Retorna o posto de pesquisa sob operação do sistema  
 111: \*   
 112: \* @return posto de pesquisa sob operação do sistema  
 113: \*/  
 114: public static int getPostoInt() {  
 115: return Integer.parseInt(Concentrador.posto);  
 116: }  
 117:   
 118:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\DocumentSizeFilter.java

1:/\*  
 2: \* Copyright (c) 1995, 2008, Oracle and/or its affiliates. All rights reserved.  
 3: \*  
 4: \* Redistribution and use in source and binary forms, with or without  
 5: \* modification, are permitted provided that the following conditions  
 6: \* are met:  
 7: \*  
 8: \* - Redistributions of source code must retain the above copyright  
 9: \* notice, this list of conditions and the following disclaimer.  
 10: \*  
 11: \* - Redistributions in binary form must reproduce the above copyright  
 12: \* notice, this list of conditions and the following disclaimer in the  
 13: \* documentation and/or other materials provided with the distribution.  
 14: \*  
 15: \* - Neither the name of Oracle or the names of its  
 16: \* contributors may be used to endorse or promote products derived  
 17: \* from this software without specific prior written permission.  
 18: \*  
 19: \* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS  
 20: \* IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,  
 21: \* THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR  
 22: \* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR  
 23: \* CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,  
 24: \* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,  
 25: \* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR  
 26: \* PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF  
 27: \* LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING  
 28: \* NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS  
 29: \* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.  
 30: \*/  
 31:  
 32:package br.ufrj.coppetec.concentrador;  
 33:  
 34:import java.awt.Toolkit;  
 35:  
 36:import javax.swing.text.AttributeSet;  
 37:import javax.swing.text.BadLocationException;  
 38:import javax.swing.text.DocumentFilter;  
 39:  
 40:public class DocumentSizeFilter extends DocumentFilter {  
 41: int maxCharacters;  
 42: boolean DEBUG = false;  
 43:  
 44: public DocumentSizeFilter(int maxChars) {  
 45: maxCharacters = maxChars;  
 46: }  
 47:  
 48: public void insertString(FilterBypass fb, int offs, String str, AttributeSet a) throws BadLocationException {  
 49: if (DEBUG) {  
 50: System.out.println("in DocumentSizeFilter's insertString method");  
 51: }  
 52:  
 53: // This rejects the entire insertion if it would make  
 54: // the contents too long. Another option would be  
 55: // to truncate the inserted string so the contents  
 56: // would be exactly maxCharacters in length.  
 57: if ((fb.getDocument().getLength() + str.length()) <= maxCharacters)  
 58: super.insertString(fb, offs, str, a);  
 59: else  
 60: Toolkit.getDefaultToolkit().beep();  
 61: }  
 62:  
 63: public void replace(FilterBypass fb, int offs, int length, String str, AttributeSet a) throws BadLocationException {  
 64: if (DEBUG) {  
 65: System.out.println("in DocumentSizeFilter's replace method");  
 66: }  
 67: // This rejects the entire replacement if it would make  
 68: // the contents too long. Another option would be  
 69: // to truncate the replacement string so the contents  
 70: // would be exactly maxCharacters in length.  
 71: if ((fb.getDocument().getLength() + str.length() - length) <= maxCharacters)  
 72: super.replace(fb, offs, length, str, a);  
 73: else  
 74: Toolkit.getDefaultToolkit().beep();  
 75: }  
 76:  
 77:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\Janela.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador;  
 7:  
 8:import static br.ufrj.coppetec.concentrador.Util.TODAS;  
 9:import br.ufrj.coppetec.concentrador.database.DBFileImporter;  
 10:import java.awt.Component;  
 11:import java.awt.Dimension;  
 12:import java.awt.Font;  
 13:import java.io.File;  
 14:import java.lang.reflect.Field;  
 15:import java.text.ParseException;  
 16:import java.util.ArrayList;  
 17:import java.util.Date;  
 18:import java.util.HashMap;  
 19:import java.util.List;  
 20:import java.util.Map;  
 21:import java.util.Vector;  
 22:  
 23:import javax.swing.BorderFactory;  
 24:import javax.swing.DefaultComboBoxModel;  
 25:import javax.swing.ImageIcon;  
 26:import javax.swing.InputVerifier;  
 27:import javax.swing.JComboBox;  
 28:import javax.swing.JComponent;  
 29:import javax.swing.JLabel;  
 30:import javax.swing.JOptionPane;  
 31:import javax.swing.JPanel;  
 32:import javax.swing.JPasswordField;  
 33:import javax.swing.JTable;  
 34:import javax.swing.JTextField;  
 35:import javax.swing.SwingConstants;  
 36:import javax.swing.SwingUtilities;  
 37:import javax.swing.event.DocumentEvent;  
 38:import javax.swing.event.DocumentListener;  
 39:import javax.swing.filechooser.FileFilter;  
 40:import javax.swing.filechooser.FileSystemView;  
 41:import javax.swing.table.DefaultTableCellRenderer;  
 42:import javax.swing.table.DefaultTableModel;  
 43:import javax.swing.text.AttributeSet;  
 44:import javax.swing.text.BadLocationException;  
 45:import javax.swing.text.PlainDocument;  
 46:  
 47:import org.apache.commons.lang3.ArrayUtils;  
 48:import org.apache.logging.log4j.LogManager;  
 49:import org.apache.logging.log4j.Logger;  
 50:import org.json.JSONObject;  
 51:  
 52:import br.ufrj.coppetec.concentrador.database.DBFileImporter;  
 53:import br.ufrj.coppetec.concentrador.database.PVKey;  
 54:import br.ufrj.coppetec.concentrador.database.PVregister;  
 55:import br.ufrj.coppetec.concentrador.database.myDB;  
 56:import br.ufrj.coppetec.concentrador.exporter.JSONExporter;  
 57:import java.awt.Color;  
 58:import javax.swing.Timer;  
 59:  
 60:/\*\*  
 61: \* Tela principal da interface gráfica com o usuário  
 62: \*   
 63: \* @author ludes - PESC - COPPE - ufrj  
 64: \* @author Eduardo Mangeli  
 65: \* @author Marcelo Areas  
 66: \* @author Fabrício Pereira  
 67: \* @author Geraldo Xexéo  
 68: \*/  
 69:@SuppressWarnings("serial")  
 70:public class Janela extends javax.swing.JFrame {  
 71:  
 72: private static Logger logger = LogManager.getLogger(Janela.class);  
 73: private String[] datesToShow; ///< Datas a serem exibidas para a entrada das pesquisas  
 74:  
 75: /\*\*  
 76: \* Cria a tela, constrói estruturas de dados para armazenar as informações das pesquisas,   
 77: \* prepara validadores para os dados informados.  
 78: \*/  
 79: public Janela() {  
 80: // quando estava usando tabelas  
 81: numeros = new DefaultTableCellRenderer();  
 82: numeros.setHorizontalAlignment(SwingConstants.CENTER);  
 83: numeros.setVerticalAlignment(SwingConstants.CENTER);  
 84:  
 85: imagens = new ImagemRenderer();  
 86: imagens.setHorizontalAlignment(SwingConstants.CENTER);  
 87:  
 88: intVerifier = new InputVerifier() {  
 89: public boolean verify(JComponent comp) {  
 90: boolean returnValue;  
 91: JTextField textField = (JTextField) comp;  
 92: int maxLength = 3;  
 93: if (textField.getText().length() > maxLength) {  
 94: JOptionPane.showMessageDialog(Janela.this,  
 95: "A entrada máxima é de " + maxLength + " caracteres numéricos para cada campo.", "Entrada incorreta",  
 96: JOptionPane.ERROR\_MESSAGE);  
 97: returnValue = false;  
 98: } else {  
 99: try {  
 100: Integer.parseInt(textField.getText());  
 101: returnValue = true;  
 102: } catch (NumberFormatException e) {  
 103: JOptionPane.showMessageDialog(Janela.this, "Apenas números inteiros são permitidos como entrada",  
 104: "Entrada incorreta", JOptionPane.ERROR\_MESSAGE);  
 105: returnValue = false;  
 106: }  
 107: }  
 108:  
 109: return returnValue;  
 110: }  
 111: };  
 112:  
 113: initComponents();  
 114: initFieldValues();  
 115: // hide server tab  
 116: tabRelatorio.remove(pnl\_servidor);  
 117: btnApagar.setVisible(false);  
 118: chk\_exportadas\_sumVol.setVisible(false);  
 119: chk\_nao\_exportadas\_sumVol.setVisible(false);  
 120:  
 121: relatorio.setDefaultRenderer(Object.class, new RelatorioODRender());  
 122: relatorio.setDragEnabled(false);  
 123: relatorio.setRowHeight(25);  
 124: relatorio.getTableHeader().setReorderingAllowed(false);  
 125: relatorio.getTableHeader().setResizingAllowed(false);  
 126:  
 127: }  
 128:   
 129: /\*\*  
 130: \* Retorna o texto para a barra de títulos da janela.  
 131: \* O título contém a versão atual do sistema e um aviso se o modo de treinamento estiver ativado.  
 132: \*   
 133: \* @return título da tela  
 134: \*/  
 135: public String getTituloJanela(){  
 136: String titulo = "Concentrador de dados - versão " + Concentrador.version;  
 137: if(Concentrador.treinamento){  
 138: titulo = "TREINAMENTO - " + titulo + " - TREINAMENTO";  
 139: }  
 140: return titulo;  
 141: }  
 142:  
 143: /\*\*  
 144: \* Exibe barra de alerta se o sistema estiver rodando em modo de treinamento.  
 145: \*   
 146: \*/  
 147: public void setAlertaTreinamento(){  
 148: if (!Concentrador.treinamento){  
 149: this.alertaTreinamento.setVisible(false);  
 150: } else {  
 151:// Timer timer = new Timer(1000, new Blinker(this.txtAlertTreinamento));  
 152:// timer.setInitialDelay(0);  
 153:// timer.start();  
 154: this.txtAlertTreinamento.setBackground(Color.red);  
 155: }  
 156: }  
 157:   
 158: /\*\*  
 159: \* Inicializa a estrutura de dados datesToShow, que contém as datas a serem exibidas.  
 160: \*   
 161: \* Carrega as datas da pesquisa a serem exibidas para seleção e entrada de dados. No caso do sistema estar operando no modo   
 162: \* de treinamento, as datas caregadas são fictícias.   
 163: \*   
 164: \*   
 165: \* @throws ParseException   
 166: \*/  
 167: public void initDatesToShow() throws ParseException {  
 168: datesToShow = null;  
 169: if (Concentrador.treinamento) {  
 170: datesToShow = Util.getTrainingDatesStr();  
 171: } else {  
 172: datesToShow = Util.getValidDatesStr();  
 173: }  
 174: String[] vazio = { "" };  
 175: cmbData.setModel(new DefaultComboBoxModel<String>(Util.concatArrays(vazio, datesToShow)));  
 176: cmbData.setSelectedItem(0);  
 177: }  
 178:  
 179: /\*\*  
 180: \* Recupera do banco de dados as datas das pesquisas realizadas.  
 181: \*   
 182: \* O método recupera do banco de dados as datas das pesquisas Volumétricas e Origem/Destino já realizadas. Usa o método  
 183: \* \_\_prepareFillComboDates\_\_ para preencher as \_caixas de seleção\_.  
 184: \*/  
 185: private void fillCmbDatesExp() {  
 186: String[] dataBaseDatesVol = null;  
 187: String[] dataBaseDatesOD = null;  
 188: try {  
 189: myDB database = Concentrador.database;  
 190: dataBaseDatesVol = database.getVolDates(Integer.valueOf(Concentrador.posto));  
 191: dataBaseDatesOD = database.getODDates(Integer.valueOf(Concentrador.posto));  
 192:  
 193: } catch (Exception e) {  
 194: logger.error("Erro ao conectar com o BD.", e);  
 195: JOptionPane.showMessageDialog(Janela.this, "Erro ao conectar com o banco de dados:\n" + e.getMessage(),  
 196: "Erro de conexão com o banco de dados.", JOptionPane.ERROR\_MESSAGE);  
 197: }  
 198:  
 199: try {  
 200: prepareFillComboDates(cmbDateExpVol, dataBaseDatesVol, 1, false);  
 201: prepareFillComboDates(cmbDateExpOD, dataBaseDatesOD, 1, true);  
 202: } catch (ParseException e) {  
 203: logger.error("Erro no preenchimento dos campos de data para exportação.", e);  
 204: JOptionPane.showMessageDialog(Janela.this,  
 205: "Erro no preenchimento dos campos de data para exportação:\n" + e.getMessage(),  
 206: "Erro no preenchimento de campos.", JOptionPane.ERROR\_MESSAGE);  
 207: }  
 208: }  
 209:  
 210: /\*\*  
 211: \* Recupera as datas das pesquisas volumétricas já realizadas.  
 212: \*   
 213: \* Recupera do banco de dados as datas das pesquisas volumétricas já realizadas. Usa o método \_\_prepareFillComboDates\_\_   
 214: \* para preencher a \_caixas de seleção\_ no sumário.  
 215: \*/  
 216: private void fillCmbDatesSumVol() {  
 217: String[] dataBaseDates = null;  
 218: try {  
 219: myDB database = Concentrador.database;  
 220: dataBaseDates = database.getVolDates(Integer.valueOf(Concentrador.posto));  
 221: } catch (Exception e) {  
 222: logger.error("Erro ao conectar com o BD.", e);  
 223: JOptionPane.showMessageDialog(Janela.this, "Erro ao conectar com o banco de dados:\n" + e.getMessage(),  
 224: "Erro de conexão com o banco de dados.", JOptionPane.ERROR\_MESSAGE);  
 225: }  
 226: try {  
 227: prepareFillComboDates(cmbDataSumVol, dataBaseDates, 0, false);  
 228: } catch (ParseException e) {  
 229: logger.error("Erro no preenchimento dos campos de data do sumário da pesquisa volumétrica.", e);  
 230: JOptionPane.showMessageDialog(Janela.this,  
 231: "Erro no preenchimento dos campos de data do sumário da pesquisa volumétrica:\n" + e.getMessage(),  
 232: "Erro no preenchimento de campos.", JOptionPane.ERROR\_MESSAGE);  
 233: }  
 234: }  
 235:   
 236:   
 237: /\*\*  
 238: \* Preenche datas de pesquisa em \_caixas de seleção\_.  
 239: \*   
 240: \* @param cmbDateExp \_Caixa de seleção\_ onde as datas serão exibidas.  
 241: \* @param dataBaseDates Datas a serem exibidas.  
 242: \* @param defaultIndex Índice do item selecionado após o preenchimento.  
 243: \* @param includeDatasDB\_OD Flag para incluir todas as datas, mesmo as fora da estrutura de dados \_\_datesToShow\_\_.  
 244: \* @throws ParseException   
 245: \*/  
 246: protected void prepareFillComboDates(JComboBox<String> cmbDateExp, String[] dataBaseDates, int defaultIndex,  
 247: boolean includeDatasDB\_OD) throws ParseException {  
 248:   
 249: cmbDateExp.removeAllItems();  
 250: cmbDateExp.addItem("--");  
 251:   
 252: if (dataBaseDates != null) {  
 253: String date;  
 254:   
 255: for (String dataBaseDate : dataBaseDates) {  
 256: date = Util.sdfBrazil.format(Util.sdfSQL.parse(dataBaseDate)).trim();  
 257: if (includeDatasDB\_OD || ArrayUtils.contains(datesToShow, date)) {  
 258: cmbDateExp.addItem(date);  
 259: }  
 260: }  
 261: cmbDateExp.addItem(TODAS);  
 262: cmbDateExp.setSelectedIndex(defaultIndex);  
 263: }  
 264: }  
 265:  
 266: /\*\*  
 267: \* Preenche os totais por dia da pesquisa volumétrica na tela do sumário.   
 268: \*   
 269: \* Recupera do banco de dados e preenche os totais por dia nos elementos da tela do sumário.  
 270: \*   
 271: \* @param date Data a ser pesquisada   
 272: \*/  
 273: private void setSumVolData(String date) {  
 274: Map<String, Integer> volData;  
 275: try {  
 276: myDB database = Concentrador.database;  
 277: volData = database.getSumVol(volFieldsNames, date);  
 278: Class<Janela> janelaClass = Janela.class;  
 279: Field sumVol;  
 280: for (String volFieldsName : volFieldsNames) {  
 281: sumVol = janelaClass.getDeclaredField("txtSumVol" + volFieldsName);  
 282: JTextField txtField = (JTextField) sumVol.get(this);  
 283: txtField.setText(volData.get(volFieldsName).toString());  
 284: }  
 285: } catch (Exception e) {  
 286: logger.error("Erro preenchendo o sumário da pesquisa volumétrica.", e);  
 287: JOptionPane.showMessageDialog(Janela.this, "Erro ao conectar com o banco de dados:\n" + e.getMessage(),  
 288: "Erro de conexão com o banco de dados.", JOptionPane.ERROR\_MESSAGE);  
 289: }  
 290: }  
 291:   
 292: /\*\*  
 293: \* Limpa os elementos da tela de sumário da pesquisa volumétrica.  
 294: \*/  
 295: private void clearSumVolData(){  
 296: Class<Janela> janelaClass = Janela.class;  
 297: Field sumVol;  
 298: try{  
 299: for (String volFieldsName : volFieldsNames){  
 300: sumVol = janelaClass.getDeclaredField("txtSumVol" + volFieldsName);  
 301: JTextField txtField = (JTextField) sumVol.get(this);  
 302: txtField.setText("0");  
 303: }  
 304: } catch (Exception e){  
 305: logger.error("Erro ao limpar o sumário da pesquisa volumétrica.", e);  
 306: JOptionPane.showMessageDialog(Janela.this, "Erro ao ao limpar o sumário da pesquisa volumétrica:\n" + e.getMessage(),  
 307: "Erro.", JOptionPane.ERROR\_MESSAGE);  
 308: }  
 309:   
 310: }  
 311:  
 312: /\*\*  
 313: \* Preenche os totais por hora da pesquisa volumétrica na tela do sumário.   
 314: \*   
 315: \* Recupera do banco de dados e preenche os totais por hora nos elementos da tela do sumário.  
 316: \*   
 317: \* @param date Data a ser pesquisada   
 318: \*/  
 319: private void setSumVolTable(String date) {  
 320: Map<String, Map<Integer, Integer>> volData;  
 321:  
 322: try {  
 323: myDB database = Concentrador.database;  
 324: volData = database.getSumVolPerHour(volFieldsNames, date);  
 325: int col;  
 326: for (int i = 0; i < volFieldsNames.length; i++) {  
 327: col = 1;  
 328: for (int j = 0; j < 24; j += 2) {  
 329: tblSumVol.getModel().setValueAt(volData.get(volFieldsNames[i]).get(j), i, col++);  
 330: }  
 331: }  
 332: } catch (Exception e) {  
 333: logger.error("Erro preenchendo o sumário da pesquisa volumétrica.", e);  
 334: JOptionPane.showMessageDialog(Janela.this, "Erro ao conectar com o banco de dados:\n" + e.getMessage(),  
 335: "Erro de conexão com o banco de dados.", JOptionPane.ERROR\_MESSAGE);  
 336: }  
 337: }  
 338:   
 339: /\*\*  
 340: \* Limpa os elementos da tabela do sumário da tela da pesquisa volumétrica.  
 341: \*/  
 342: private void clearSumVolTable(){  
 343: int col;  
 344: for (int i = 0; i < volFieldsNames.length; i++) {  
 345: col = 1;  
 346: for (int j = 0; j < 24; j += 2) {  
 347: tblSumVol.getModel().setValueAt("0", i, col++);  
 348: }  
 349: }  
 350: }  
 351:  
 352: /\*\*  
 353: \* Inicializa estruturas de dados para manipulação dos campos das telas.  
 354: \*/  
 355: private void initFieldValues() {  
 356:  
 357: // leves  
 358: JTextField[] p1Fields = { tl0, tl1, tl2, tl3, tl4, tl5, tl6, tl7 };   
 359: JTextField[] p3Fields = { tl8, tl9, tl10, tl11, tl12, tl13, tl14, tl15 };   
 360: JTextField[] p2Fields = { tl16, tl17, tl18, tl19, tl20, tl21, tl22, tl23 };   
 361: JTextField[] mFields = { tl24, tl25, tl26, tl27, tl28, tl29, tl30, tl31 };   
 362: JTextField[] o1Fields = { tl32, tl33, tl34, tl35, tl36, tl37, tl38, tl39 };   
 363: JTextField[] o2Fields = { tl40, tl41, tl42, tl43, tl44, tl45, tl46, tl47 };   
 364: JTextField[] o3Fields = { tl48, tl49, tl50, tl51, tl52, tl53, tl54, tl55 };   
 365: JTextField[] c1Fields = { tl56, tl57, tl58, tl59, tl60, tl61, tl62, tl63 };   
 366: JTextField[] c2Fields = { tl64, tl65, tl66, tl67, tl68, tl69, tl70, tl71 };   
 367: JTextField[] c3Fields = { tl72, tl73, tl74, tl75, tl76, tl77, tl78, tl79 };   
 368: JTextField[] c4Fields = { tl80, tl81, tl82, tl83, tl84, tl85, tl86, tl87 };   
 369: JTextField[] c5Fields = { tl88, tl89, tl90, tl91, tl92, tl93, tl94, tl95 };   
 370: // pesados  
 371: JTextField[] s1Fields = { tp0, tp1, tp2, tp3, tp4, tp5, tp6, tp7 };   
 372: JTextField[] s2Fields = { tp8, tp9, tp10, tp11, tp12, tp13, tp14, tp15 };   
 373: JTextField[] s3Fields = { tp16, tp17, tp18, tp19, tp20, tp21, tp22, tp23 };   
 374: JTextField[] s4Fields = { tp24, tp25, tp26, tp27, tp28, tp29, tp30, tp31 };   
 375: JTextField[] s5Fields = { tp32, tp33, tp34, tp35, tp36, tp37, tp38, tp39 };   
 376: JTextField[] s6Fields = { tp40, tp41, tp42, tp43, tp44, tp45, tp46, tp47 };   
 377: JTextField[] se1Fields = { tp48, tp49, tp50, tp51, tp52, tp53, tp54, tp55 };   
 378: JTextField[] se2Fields = { tp56, tp57, tp58, tp59, tp60, tp61, tp62, tp63 };   
 379: JTextField[] se3Fields = { tp64, tp65, tp66, tp67, tp68, tp69, tp70, tp71 };   
 380: JTextField[] se4Fields = { tp72, tp73, tp74, tp75, tp76, tp77, tp78, tp79 };   
 381: JTextField[] se5Fields = { tp80, tp81, tp82, tp83, tp84, tp85, tp86, tp87 };   
 382: JTextField[] r1Fields = { tp88, tp89, tp90, tp91, tp92, tp93, tp94, tp95 };   
 383: JTextField[] r2Fields = { tp96, tp97, tp98, tp99, tp100, tp101, tp102, tp103 };   
 384: JTextField[] r3Fields = { tp104, tp105, tp106, tp107, tp108, tp109, tp110, tp111 };  
 385: JTextField[] r4Fields = { tp112, tp113, tp114, tp115, tp116, tp117, tp118, tp119 };  
 386: JTextField[] r5Fields = { tp120, tp121, tp122, tp123, tp124, tp125, tp126, tp127 };  
 387: JTextField[] r6Fields = { tp128, tp129, tp130, tp131, tp132, tp133, tp134, tp135 };  
 388:  
 389: fieldsMap = new HashMap<String, JTextField[]>();  
 390: fieldsMap.put("P1", p1Fields);  
 391: fieldsMap.put("P3", p3Fields);  
 392: fieldsMap.put("P2", p2Fields);  
 393: fieldsMap.put("M", mFields);  
 394: fieldsMap.put("O1", o1Fields);  
 395: fieldsMap.put("O2", o2Fields);  
 396: fieldsMap.put("O3", o3Fields);  
 397: fieldsMap.put("C1", c1Fields);  
 398: fieldsMap.put("C2", c2Fields);  
 399: fieldsMap.put("C3", c3Fields);  
 400: fieldsMap.put("C4", c4Fields);  
 401: fieldsMap.put("C5", c5Fields);  
 402: // pesados  
 403: fieldsMap.put("S1", s1Fields);  
 404: fieldsMap.put("S2", s2Fields);  
 405: fieldsMap.put("S3", s3Fields);  
 406: fieldsMap.put("S4", s4Fields);  
 407: fieldsMap.put("S5", s5Fields);  
 408: fieldsMap.put("S6", s6Fields);  
 409: fieldsMap.put("SE1", se1Fields);  
 410: fieldsMap.put("SE2", se2Fields);  
 411: fieldsMap.put("SE3", se3Fields);  
 412: fieldsMap.put("SE4", se4Fields);  
 413: fieldsMap.put("SE5", se5Fields);  
 414: fieldsMap.put("R1", r1Fields);  
 415: fieldsMap.put("R2", r2Fields);  
 416: fieldsMap.put("R3", r3Fields);  
 417: fieldsMap.put("R4", r4Fields);  
 418: fieldsMap.put("R5", r5Fields);  
 419: fieldsMap.put("R6", r6Fields);  
 420:  
 421: volFieldsNames = Util.concatArrays(volFieldsNamesLeves, volFieldsNamesPesados);  
 422:  
 423: // select text on enter  
 424: for (String s : volFieldsNames) {  
 425: for (JTextField f : fieldsMap.get(s)) {  
 426: f.setText("0");  
 427: f.addFocusListener(new java.awt.event.FocusAdapter() {  
 428: @Override  
 429: public void focusGained(java.awt.event.FocusEvent evt) {  
 430: SwingUtilities.invokeLater(new Runnable() {  
 431: @Override  
 432: public void run() {  
 433: f.selectAll();  
 434: }  
 435: });  
 436: }  
 437: });  
 438: }  
 439: }  
 440:  
 441: // sum values  
 442: HashMap<fieldTypes, String[]> separatedFields = new HashMap<fieldTypes, String[]>();  
 443: separatedFields.put(fieldTypes.LEVES, volFieldsNamesLeves);  
 444: separatedFields.put(fieldTypes.PESADOS, volFieldsNamesPesados);  
 445:  
 446: for (Map.Entry<fieldTypes, String[]> entry : separatedFields.entrySet()) {  
 447: for (String s : entry.getValue()) {  
 448: for (JTextField f : fieldsMap.get(s)) {  
 449: f.getDocument().addDocumentListener(new DocumentListener() {  
 450:  
 451: @Override  
 452: public void insertUpdate(DocumentEvent e) {  
 453: sumFields(entry.getKey());  
 454: }  
 455:  
 456: @Override  
 457: public void removeUpdate(DocumentEvent e) {  
 458: sumFields(entry.getKey());  
 459: }  
 460:  
 461: @Override  
 462: public void changedUpdate(DocumentEvent e) {  
 463: sumFields(entry.getKey());  
 464: }  
 465: });  
 466:  
 467: }  
 468: }  
 469: }  
 470:  
 471: // limitando a antrada nos campos de texto  
 472: txtFree = new HashMap<String, JTextField>();  
 473: txtFree.put("pesquisador\_1", txtPesquisador1);  
 474: txtFree.put("pesquisador\_2", txtPesquisador2);  
 475: txtFree.put("local", txtLocal);  
 476:  
 477: for (Map.Entry<String, JTextField> entry : txtFree.entrySet()) {  
 478: entry.getValue().setDocument(new PlainDocument() {  
 479: @Override  
 480: public void insertString(int offs, String str, AttributeSet a) throws BadLocationException {  
 481: if (getLength() + str.length() <= Integer  
 482: .parseInt(Concentrador.configuration.getProperty(entry.getKey(), "3"))) {  
 483: super.insertString(offs, str, a);  
 484: }  
 485: }  
 486: });  
 487: }  
 488:  
 489: DefaultTableCellRenderer centerRenderer = new DefaultTableCellRenderer();  
 490: centerRenderer.setHorizontalAlignment(SwingConstants.CENTER);  
 491: tblSumVol.setAutoResizeMode(JTable.AUTO\_RESIZE\_OFF);  
 492: tblSumVol.setRowHeight(50);  
 493: tblSumVol.getTableHeader().setReorderingAllowed(false);  
 494: for (int i = 0; i < tblSumVol.getModel().getColumnCount(); i++) {  
 495: tblSumVol.getColumnModel().getColumn(i).setCellRenderer(centerRenderer);  
 496: tblSumVol.getColumnModel().getColumn(i).setMinWidth(100);  
 497:  
 498: }  
 499:  
 500: };  
 501:  
 502: /\*\*  
 503: \* Verifica se todos os dados que identificam um registro de pesquisa volumétrica foram informados.  
 504: \*   
 505: \* Um registro de pesquisa volumétrica é identificado pelo seguinte conjunto de campos: hora, data, sentido e posto.  
 506: \*   
 507: \* @return \_\_true\_\_ se os dados analisados tiverem sido informados   
 508: \*/  
 509: private boolean checkPVKeyDataEnter() {  
 510: return (cmbHora.getSelectedIndex() != -1 && cmbData.getSelectedIndex() != 0  
 511: && (rdo\_SentidoAB.isSelected() || rdo\_SentidoBA.isSelected()));  
 512: }  
 513:  
 514: /\*\*  
 515: \* Constrói um objeto que identifica de forma unívoca um registro de pesquisa volumétrica.  
 516: \*   
 517: \* O objeto construído contém os dados da pesquisa (hora, data, sentido e posto) sob operação da tela no momento.  
 518: \*   
 519: \* @return Objeto \_\_PVKey\_\_ que representa um identificar unívoco de registro de pesquisa volumétrica.  
 520: \* @throws ParseException   
 521: \*/  
 522: private PVKey makePVKey() throws ParseException {  
 523: PVKey pvKey = new PVKey();  
 524: pvKey.data = cmbData.getSelectedItem().toString();  
 525: pvKey.data = Util.sdfSQL.format(Util.sdfBrazil.parse(cmbData.getSelectedItem().toString())).toString();  
 526: pvKey.hora = Integer.parseInt(cmbHora.getSelectedItem().toString());  
 527: pvKey.posto = Integer.parseInt(Concentrador.posto);  
 528: pvKey.sentido = (rdo\_SentidoAB.isSelected()) ? "AB" : "BA";  
 529: return pvKey;  
 530: }  
 531:  
 532: /\*\*  
 533: \* Trata recuperação e modificação de pesquisas volumétricas.  
 534: \*  
 535: \* Verifica se já existe um registro de pesquisa volumétrica no banco de com os mesmos dados para data, hora, sentido e posto.  
 536: \* No caso de existir, pergunta ao usuário se deseja recuperar os dados e, no caso de retorno positivo, preenche as   
 537: \* informações recuperadas na tela.  
 538: \*/  
 539: private void askForDataRetrieve() {  
 540: if (checkPVKeyDataEnter() && ctlGetValuesFromDataBase) {  
 541: btnApagar.setVisible(false);  
 542:  
 543: try {  
 544: PVKey pvKey = makePVKey();  
 545: myDB database = Concentrador.database;  
 546: int alreadyInDataBase = database.verifyPV(pvKey);  
 547: if (alreadyInDataBase != 0) {  
 548: int returnedValue = JOptionPane.showConfirmDialog(Janela.this,  
 549: "Já existe um registro no Banco de Dados com o mesmo posto, sentido, data e hora.\nVocê deseja recuperar os dados já gravados?",  
 550: "Dados já existentes.", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE);  
 551: if (returnedValue == JOptionPane.YES\_OPTION) {  
 552:  
 553: // preenche os dados  
 554: ctlGetValuesFromDataBase = false;  
 555: btnApagar.setVisible(true);  
 556: PVregister pvR = database.getPVRegister(alreadyInDataBase);  
 557: txtLocal.setText(pvR.local);  
 558: txtPesquisador1.setText(pvR.pesquisador1);  
 559: txtPesquisador2.setText(pvR.pesquisador2);  
 560: // data.setDate(Util.sdf.parse(pvR.data));  
 561: cmbData.setSelectedItem(Util.sdfBrazil.format(Util.sdfSQL.parse(pvR.data)));  
 562: if (pvR.sentido.equals("AB")) {  
 563: rdo\_SentidoAB.doClick();  
 564: } else {  
 565: rdo\_SentidoBA.doClick();  
 566: }  
 567:  
 568: if (pvR.pista.equals("S")) {  
 569: rdo\_PistaSimples.doClick();  
 570: } else {  
 571: rdo\_PistaDupla.doClick();  
 572: }  
 573: ctlGetValuesFromDataBase = true;  
 574:  
 575: Class<PVregister> registerClass = PVregister.class;  
 576: Field vehicleType;  
 577: String jsonData = null;  
 578: JSONObject typeData;  
 579: String[] values;  
 580: int i;  
 581: for (String fieldName : fieldsMap.keySet()) {  
 582: vehicleType = registerClass.getDeclaredField(fieldName.toLowerCase());  
 583: vehicleType.setAccessible(true);  
 584: jsonData = (String) vehicleType.get(pvR);  
 585: typeData = new JSONObject(jsonData);  
 586: values = ((String) typeData.get("Valores")).split(",");  
 587: for (i = 0; i < fieldsMap.get(fieldName).length; i++) {  
 588: fieldsMap.get(fieldName)[i].setText(values[i]);  
 589: }  
 590:  
 591: }  
 592: }  
 593: return;// exit and don't get data  
 594: }  
 595: } catch (Exception e) {  
 596: logger.error("Erro ao conectar com o BD.", e);  
 597: JOptionPane.showMessageDialog(Janela.this, "Erro ao conectar com o banco de dados:\n" + e.getMessage(),  
 598: "Erro de conexão com o banco de dados.", JOptionPane.ERROR\_MESSAGE);  
 599: }  
 600:  
 601: }  
 602: }  
 603:  
 604: /\*\*  
 605: \* Preenche os dados do total de veículos informados na pesquisa volumétrica atual.  
 606: \*   
 607: \* Calcula o total de veículos informados na pesquisa volumétrica atual e preenche a informação no respectivo \_campo de texto\_.  
 608: \*   
 609: \* @param type tipo de veículo [LEVES | PESADOS]  
 610: \*/  
 611: private void sumFields(fieldTypes type) {  
 612: String[] fields = null;  
 613: JTextField sumField = null;  
 614: int r = 0;  
 615: switch (type) {  
 616: case LEVES:  
 617: fields = volFieldsNamesLeves;  
 618: sumField = txtTotalLeves;  
 619: break;  
 620: case PESADOS:  
 621: fields = volFieldsNamesPesados;  
 622: sumField = txtTotalPesados;  
 623: break;  
 624: }  
 625:  
 626: for (String s : fields) {  
 627: for (JTextField f : fieldsMap.get(s)) {  
 628: try {  
 629: r += Integer.parseInt(f.getText());  
 630: } catch (Exception e) {  
 631: // nothing  
 632: }  
 633: }  
 634: }  
 635: sumField.setText(String.valueOf(r));  
 636: }  
 637:  
 638: /\*\*  
 639: \* Método utilizado pelo construtor para iniciar os componentes da tela  
 640: \*   
 641: \* WARNING: Não modifique esse método. O seu conteúdo é sempre gerado automaticamente pelo gerador de formulários  
 642: \* da IDE NetBeans.  
 643: \*/  
 644: @SuppressWarnings({ "deprecation" })  
 645: // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents  
 646: private void initComponents() {  
 647: java.awt.GridBagConstraints gridBagConstraints;  
 648:  
 649: grpSentido = new javax.swing.ButtonGroup();  
 650: grpPista = new javax.swing.ButtonGroup();  
 651: dadosFileChooser = new javax.swing.JFileChooser();  
 652: exporterFileChooser = new javax.swing.JFileChooser();  
 653: alertaTreinamento = new javax.swing.JPanel();  
 654: txtAlertTreinamento = new javax.swing.JTextField();  
 655: tabRelatorio = new javax.swing.JTabbedPane();  
 656: pnl\_volumetrica = new javax.swing.JPanel();  
 657: lblPosto = new javax.swing.JLabel();  
 658: lblPosto\_dados = new javax.swing.JLabel();  
 659: lblHora = new javax.swing.JLabel();  
 660: cmbHora = new javax.swing.JComboBox<String>();  
 661: lblLocal = new javax.swing.JLabel();  
 662: txtLocal = new javax.swing.JTextField();  
 663: lblSentido = new javax.swing.JLabel();  
 664: rdo\_SentidoBA = new javax.swing.JRadioButton();  
 665: rdo\_SentidoAB = new javax.swing.JRadioButton();  
 666: lblPista = new javax.swing.JLabel();  
 667: rdo\_PistaSimples = new javax.swing.JRadioButton();  
 668: rdo\_PistaDupla = new javax.swing.JRadioButton();  
 669: lblData = new javax.swing.JLabel();  
 670: jTabbedPane1 = new javax.swing.JTabbedPane();  
 671: tab\_leves = new javax.swing.JPanel();  
 672: lblPesquisador = new javax.swing.JLabel();  
 673: txtPesquisador1 = new javax.swing.JTextField();  
 674: jScrollPane3 = new javax.swing.JScrollPane();  
 675: jPanel119 = new javax.swing.JPanel();  
 676: panelLeves = new javax.swing.JPanel();  
 677: leves\_hora1 = new javax.swing.JTextField();  
 678: lev15\_1 = new javax.swing.JTextField();  
 679: lev30\_1 = new javax.swing.JTextField();  
 680: lev45\_1 = new javax.swing.JTextField();  
 681: lev60\_1 = new javax.swing.JTextField();  
 682: leves\_hora2 = new javax.swing.JTextField();  
 683: lev15\_2 = new javax.swing.JTextField();  
 684: lev30\_2 = new javax.swing.JTextField();  
 685: lev45\_2 = new javax.swing.JTextField();  
 686: lev60\_2 = new javax.swing.JTextField();  
 687: lblP1 = new javax.swing.JLabel();  
 688: jPanel3 = new javax.swing.JPanel();  
 689: tl0 = new javax.swing.JTextField();  
 690: tl1 = new javax.swing.JTextField();  
 691: tl3 = new javax.swing.JTextField();  
 692: tl5 = new javax.swing.JTextField();  
 693: tl7 = new javax.swing.JTextField();  
 694: lblP3 = new javax.swing.JLabel();  
 695: tl9 = new javax.swing.JTextField();  
 696: tl11 = new javax.swing.JTextField();  
 697: tl13 = new javax.swing.JTextField();  
 698: tl15 = new javax.swing.JTextField();  
 699: lblP2 = new javax.swing.JLabel();  
 700: tl17 = new javax.swing.JTextField();  
 701: tl19 = new javax.swing.JTextField();  
 702: tl21 = new javax.swing.JTextField();  
 703: tl23 = new javax.swing.JTextField();  
 704: lblM = new javax.swing.JLabel();  
 705: tl25 = new javax.swing.JTextField();  
 706: tl27 = new javax.swing.JTextField();  
 707: tl29 = new javax.swing.JTextField();  
 708: tl31 = new javax.swing.JTextField();  
 709: lbl2CB = new javax.swing.JLabel();  
 710: tl33 = new javax.swing.JTextField();  
 711: tl35 = new javax.swing.JTextField();  
 712: tl37 = new javax.swing.JTextField();  
 713: tl39 = new javax.swing.JTextField();  
 714: lbl3CB = new javax.swing.JLabel();  
 715: tl41 = new javax.swing.JTextField();  
 716: tl43 = new javax.swing.JTextField();  
 717: tl45 = new javax.swing.JTextField();  
 718: tl47 = new javax.swing.JTextField();  
 719: lbl4CB = new javax.swing.JLabel();  
 720: tl49 = new javax.swing.JTextField();  
 721: tl51 = new javax.swing.JTextField();  
 722: tl53 = new javax.swing.JTextField();  
 723: tl55 = new javax.swing.JTextField();  
 724: lbl2C = new javax.swing.JLabel();  
 725: tl57 = new javax.swing.JTextField();  
 726: tl59 = new javax.swing.JTextField();  
 727: tl61 = new javax.swing.JTextField();  
 728: tl63 = new javax.swing.JTextField();  
 729: lbl3C = new javax.swing.JLabel();  
 730: tl65 = new javax.swing.JTextField();  
 731: tl67 = new javax.swing.JTextField();  
 732: tl69 = new javax.swing.JTextField();  
 733: tl71 = new javax.swing.JTextField();  
 734: lbl4C = new javax.swing.JLabel();  
 735: tl73 = new javax.swing.JTextField();  
 736: tl75 = new javax.swing.JTextField();  
 737: tl77 = new javax.swing.JTextField();  
 738: tl79 = new javax.swing.JTextField();  
 739: lbl4CD = new javax.swing.JLabel();  
 740: tl81 = new javax.swing.JTextField();  
 741: tl83 = new javax.swing.JTextField();  
 742: tl85 = new javax.swing.JTextField();  
 743: tl87 = new javax.swing.JTextField();  
 744: lbl3D = new javax.swing.JLabel();  
 745: tl89 = new javax.swing.JTextField();  
 746: tl91 = new javax.swing.JTextField();  
 747: tl93 = new javax.swing.JTextField();  
 748: tl95 = new javax.swing.JTextField();  
 749: jLabel5 = new javax.swing.JLabel();  
 750: jLabel6 = new javax.swing.JLabel();  
 751: jLabel7 = new javax.swing.JLabel();  
 752: jLabel8 = new javax.swing.JLabel();  
 753: jLabel9 = new javax.swing.JLabel();  
 754: jLabel10 = new javax.swing.JLabel();  
 755: jLabel11 = new javax.swing.JLabel();  
 756: jLabel12 = new javax.swing.JLabel();  
 757: jLabel13 = new javax.swing.JLabel();  
 758: jLabel14 = new javax.swing.JLabel();  
 759: jLabel15 = new javax.swing.JLabel();  
 760: jLabel16 = new javax.swing.JLabel();  
 761: jPanel4 = new javax.swing.JPanel();  
 762: tl8 = new javax.swing.JTextField();  
 763: jPanel5 = new javax.swing.JPanel();  
 764: tl16 = new javax.swing.JTextField();  
 765: jPanel6 = new javax.swing.JPanel();  
 766: tl24 = new javax.swing.JTextField();  
 767: jPanel7 = new javax.swing.JPanel();  
 768: tl32 = new javax.swing.JTextField();  
 769: jPanel8 = new javax.swing.JPanel();  
 770: tl40 = new javax.swing.JTextField();  
 771: jPanel9 = new javax.swing.JPanel();  
 772: tl48 = new javax.swing.JTextField();  
 773: jPanel10 = new javax.swing.JPanel();  
 774: tl56 = new javax.swing.JTextField();  
 775: jPanel11 = new javax.swing.JPanel();  
 776: tl64 = new javax.swing.JTextField();  
 777: jPanel12 = new javax.swing.JPanel();  
 778: tl72 = new javax.swing.JTextField();  
 779: jPanel13 = new javax.swing.JPanel();  
 780: tl80 = new javax.swing.JTextField();  
 781: jPanel14 = new javax.swing.JPanel();  
 782: tl88 = new javax.swing.JTextField();  
 783: jPanel15 = new javax.swing.JPanel();  
 784: tl2 = new javax.swing.JTextField();  
 785: jPanel16 = new javax.swing.JPanel();  
 786: tl10 = new javax.swing.JTextField();  
 787: jPanel17 = new javax.swing.JPanel();  
 788: tl18 = new javax.swing.JTextField();  
 789: jPanel18 = new javax.swing.JPanel();  
 790: tl26 = new javax.swing.JTextField();  
 791: jPanel19 = new javax.swing.JPanel();  
 792: tl34 = new javax.swing.JTextField();  
 793: jPanel20 = new javax.swing.JPanel();  
 794: tl42 = new javax.swing.JTextField();  
 795: jPanel21 = new javax.swing.JPanel();  
 796: tl50 = new javax.swing.JTextField();  
 797: jPanel22 = new javax.swing.JPanel();  
 798: tl58 = new javax.swing.JTextField();  
 799: jPanel23 = new javax.swing.JPanel();  
 800: tl66 = new javax.swing.JTextField();  
 801: jPanel24 = new javax.swing.JPanel();  
 802: tl74 = new javax.swing.JTextField();  
 803: jPanel25 = new javax.swing.JPanel();  
 804: tl82 = new javax.swing.JTextField();  
 805: jPanel26 = new javax.swing.JPanel();  
 806: tl90 = new javax.swing.JTextField();  
 807: jPanel27 = new javax.swing.JPanel();  
 808: tl4 = new javax.swing.JTextField();  
 809: jPanel28 = new javax.swing.JPanel();  
 810: tl12 = new javax.swing.JTextField();  
 811: jPanel29 = new javax.swing.JPanel();  
 812: tl20 = new javax.swing.JTextField();  
 813: jPanel30 = new javax.swing.JPanel();  
 814: tl28 = new javax.swing.JTextField();  
 815: jPanel31 = new javax.swing.JPanel();  
 816: tl36 = new javax.swing.JTextField();  
 817: jPanel32 = new javax.swing.JPanel();  
 818: tl44 = new javax.swing.JTextField();  
 819: jPanel33 = new javax.swing.JPanel();  
 820: tl52 = new javax.swing.JTextField();  
 821: jPanel34 = new javax.swing.JPanel();  
 822: tl60 = new javax.swing.JTextField();  
 823: jPanel35 = new javax.swing.JPanel();  
 824: tl68 = new javax.swing.JTextField();  
 825: jPanel36 = new javax.swing.JPanel();  
 826: tl76 = new javax.swing.JTextField();  
 827: jPanel37 = new javax.swing.JPanel();  
 828: tl84 = new javax.swing.JTextField();  
 829: jPanel38 = new javax.swing.JPanel();  
 830: tl92 = new javax.swing.JTextField();  
 831: jPanel39 = new javax.swing.JPanel();  
 832: tl6 = new javax.swing.JTextField();  
 833: jPanel40 = new javax.swing.JPanel();  
 834: tl14 = new javax.swing.JTextField();  
 835: jPanel41 = new javax.swing.JPanel();  
 836: tl22 = new javax.swing.JTextField();  
 837: jPanel42 = new javax.swing.JPanel();  
 838: tl30 = new javax.swing.JTextField();  
 839: jPanel43 = new javax.swing.JPanel();  
 840: tl38 = new javax.swing.JTextField();  
 841: jPanel44 = new javax.swing.JPanel();  
 842: tl46 = new javax.swing.JTextField();  
 843: jPanel45 = new javax.swing.JPanel();  
 844: tl54 = new javax.swing.JTextField();  
 845: jPanel46 = new javax.swing.JPanel();  
 846: tl62 = new javax.swing.JTextField();  
 847: jPanel47 = new javax.swing.JPanel();  
 848: tl70 = new javax.swing.JTextField();  
 849: jPanel48 = new javax.swing.JPanel();  
 850: tl78 = new javax.swing.JTextField();  
 851: jPanel49 = new javax.swing.JPanel();  
 852: tl86 = new javax.swing.JTextField();  
 853: jPanel50 = new javax.swing.JPanel();  
 854: tl94 = new javax.swing.JTextField();  
 855: txtTotalLeves = new javax.swing.JTextField();  
 856: jLabel36 = new javax.swing.JLabel();  
 857: lblFolha1 = new javax.swing.JLabel();  
 858: jLabel39 = new javax.swing.JLabel();  
 859: tab\_pesados = new javax.swing.JPanel();  
 860: jScrollPane4 = new javax.swing.JScrollPane();  
 861: panelPesados = new javax.swing.JPanel();  
 862: pesados\_hora1 = new javax.swing.JTextField();  
 863: pes15\_1 = new javax.swing.JTextField();  
 864: pes30\_1 = new javax.swing.JTextField();  
 865: pes45\_1 = new javax.swing.JTextField();  
 866: pes60\_1 = new javax.swing.JTextField();  
 867: pesados\_hora2 = new javax.swing.JTextField();  
 868: pes15\_2 = new javax.swing.JTextField();  
 869: pes30\_2 = new javax.swing.JTextField();  
 870: peso45\_2 = new javax.swing.JTextField();  
 871: peso60\_2 = new javax.swing.JTextField();  
 872: lbl2S1 = new javax.swing.JLabel();  
 873: tp1 = new javax.swing.JTextField();  
 874: tp3 = new javax.swing.JTextField();  
 875: tp5 = new javax.swing.JTextField();  
 876: tp7 = new javax.swing.JTextField();  
 877: lbl2S2 = new javax.swing.JLabel();  
 878: tp9 = new javax.swing.JTextField();  
 879: tp11 = new javax.swing.JTextField();  
 880: tp13 = new javax.swing.JTextField();  
 881: tp15 = new javax.swing.JTextField();  
 882: lbl2S3 = new javax.swing.JLabel();  
 883: tp17 = new javax.swing.JTextField();  
 884: tp19 = new javax.swing.JTextField();  
 885: tp21 = new javax.swing.JTextField();  
 886: tp23 = new javax.swing.JTextField();  
 887: lbl3S1 = new javax.swing.JLabel();  
 888: tp25 = new javax.swing.JTextField();  
 889: tp27 = new javax.swing.JTextField();  
 890: tp29 = new javax.swing.JTextField();  
 891: tp31 = new javax.swing.JTextField();  
 892: lbl3S2 = new javax.swing.JLabel();  
 893: tp33 = new javax.swing.JTextField();  
 894: tp35 = new javax.swing.JTextField();  
 895: tp37 = new javax.swing.JTextField();  
 896: tp39 = new javax.swing.JTextField();  
 897: lbl3S3 = new javax.swing.JLabel();  
 898: tp41 = new javax.swing.JTextField();  
 899: tp43 = new javax.swing.JTextField();  
 900: tp45 = new javax.swing.JTextField();  
 901: tp47 = new javax.swing.JTextField();  
 902: lbl3T4 = new javax.swing.JLabel();  
 903: tp49 = new javax.swing.JTextField();  
 904: tp51 = new javax.swing.JTextField();  
 905: tp53 = new javax.swing.JTextField();  
 906: tp55 = new javax.swing.JTextField();  
 907: lbl3T6 = new javax.swing.JLabel();  
 908: tp57 = new javax.swing.JTextField();  
 909: tp59 = new javax.swing.JTextField();  
 910: tp61 = new javax.swing.JTextField();  
 911: tp63 = new javax.swing.JTextField();  
 912: lbl3T6B = new javax.swing.JLabel();  
 913: tp65 = new javax.swing.JTextField();  
 914: tp67 = new javax.swing.JTextField();  
 915: tp69 = new javax.swing.JTextField();  
 916: tp71 = new javax.swing.JTextField();  
 917: lbl3V5 = new javax.swing.JLabel();  
 918: tp73 = new javax.swing.JTextField();  
 919: tp75 = new javax.swing.JTextField();  
 920: tp77 = new javax.swing.JTextField();  
 921: tp79 = new javax.swing.JTextField();  
 922: lbl3M6 = new javax.swing.JLabel();  
 923: tp81 = new javax.swing.JTextField();  
 924: tp83 = new javax.swing.JTextField();  
 925: tp85 = new javax.swing.JTextField();  
 926: tp87 = new javax.swing.JTextField();  
 927: lbl3Q4 = new javax.swing.JLabel();  
 928: tp89 = new javax.swing.JTextField();  
 929: tp91 = new javax.swing.JTextField();  
 930: tp93 = new javax.swing.JTextField();  
 931: tp95 = new javax.swing.JTextField();  
 932: lbl2C2 = new javax.swing.JLabel();  
 933: tp97 = new javax.swing.JTextField();  
 934: tp99 = new javax.swing.JTextField();  
 935: tp101 = new javax.swing.JTextField();  
 936: tp103 = new javax.swing.JTextField();  
 937: lbl2C3 = new javax.swing.JLabel();  
 938: tp105 = new javax.swing.JTextField();  
 939: tp107 = new javax.swing.JTextField();  
 940: tp109 = new javax.swing.JTextField();  
 941: tp111 = new javax.swing.JTextField();  
 942: lbl3C2 = new javax.swing.JLabel();  
 943: tp113 = new javax.swing.JTextField();  
 944: tp115 = new javax.swing.JTextField();  
 945: tp117 = new javax.swing.JTextField();  
 946: tp119 = new javax.swing.JTextField();  
 947: lbl3C3 = new javax.swing.JLabel();  
 948: tp121 = new javax.swing.JTextField();  
 949: tp123 = new javax.swing.JTextField();  
 950: tp125 = new javax.swing.JTextField();  
 951: tp127 = new javax.swing.JTextField();  
 952: lbl3D4 = new javax.swing.JLabel();  
 953: tp129 = new javax.swing.JTextField();  
 954: tp131 = new javax.swing.JTextField();  
 955: tp133 = new javax.swing.JTextField();  
 956: tp135 = new javax.swing.JTextField();  
 957: jLabel17 = new javax.swing.JLabel();  
 958: jLabel18 = new javax.swing.JLabel();  
 959: jLabel19 = new javax.swing.JLabel();  
 960: jLabel20 = new javax.swing.JLabel();  
 961: jLabel21 = new javax.swing.JLabel();  
 962: jLabel22 = new javax.swing.JLabel();  
 963: jLabel23 = new javax.swing.JLabel();  
 964: jLabel24 = new javax.swing.JLabel();  
 965: jLabel25 = new javax.swing.JLabel();  
 966: jLabel26 = new javax.swing.JLabel();  
 967: jLabel27 = new javax.swing.JLabel();  
 968: jLabel28 = new javax.swing.JLabel();  
 969: jLabel29 = new javax.swing.JLabel();  
 970: jLabel30 = new javax.swing.JLabel();  
 971: jLabel31 = new javax.swing.JLabel();  
 972: jLabel32 = new javax.swing.JLabel();  
 973: jLabel33 = new javax.swing.JLabel();  
 974: jPanel51 = new javax.swing.JPanel();  
 975: tp0 = new javax.swing.JTextField();  
 976: jPanel52 = new javax.swing.JPanel();  
 977: tp8 = new javax.swing.JTextField();  
 978: jPanel53 = new javax.swing.JPanel();  
 979: tp16 = new javax.swing.JTextField();  
 980: jPanel54 = new javax.swing.JPanel();  
 981: tp24 = new javax.swing.JTextField();  
 982: jPanel55 = new javax.swing.JPanel();  
 983: tp32 = new javax.swing.JTextField();  
 984: jPanel56 = new javax.swing.JPanel();  
 985: tp40 = new javax.swing.JTextField();  
 986: jPanel57 = new javax.swing.JPanel();  
 987: tp48 = new javax.swing.JTextField();  
 988: jPanel58 = new javax.swing.JPanel();  
 989: tp56 = new javax.swing.JTextField();  
 990: jPanel59 = new javax.swing.JPanel();  
 991: tp64 = new javax.swing.JTextField();  
 992: jPanel60 = new javax.swing.JPanel();  
 993: tp72 = new javax.swing.JTextField();  
 994: jPanel61 = new javax.swing.JPanel();  
 995: tp80 = new javax.swing.JTextField();  
 996: jPanel62 = new javax.swing.JPanel();  
 997: tp88 = new javax.swing.JTextField();  
 998: jPanel63 = new javax.swing.JPanel();  
 999: tp96 = new javax.swing.JTextField();  
1000: jPanel64 = new javax.swing.JPanel();  
1001: tp104 = new javax.swing.JTextField();  
1002: jPanel65 = new javax.swing.JPanel();  
1003: tp112 = new javax.swing.JTextField();  
1004: jPanel66 = new javax.swing.JPanel();  
1005: tp120 = new javax.swing.JTextField();  
1006: jPanel67 = new javax.swing.JPanel();  
1007: tp128 = new javax.swing.JTextField();  
1008: jPanel68 = new javax.swing.JPanel();  
1009: tp2 = new javax.swing.JTextField();  
1010: jPanel69 = new javax.swing.JPanel();  
1011: tp10 = new javax.swing.JTextField();  
1012: jPanel70 = new javax.swing.JPanel();  
1013: tp18 = new javax.swing.JTextField();  
1014: jPanel71 = new javax.swing.JPanel();  
1015: tp26 = new javax.swing.JTextField();  
1016: jPanel72 = new javax.swing.JPanel();  
1017: tp34 = new javax.swing.JTextField();  
1018: jPanel73 = new javax.swing.JPanel();  
1019: tp42 = new javax.swing.JTextField();  
1020: jPanel74 = new javax.swing.JPanel();  
1021: tp50 = new javax.swing.JTextField();  
1022: jPanel75 = new javax.swing.JPanel();  
1023: tp58 = new javax.swing.JTextField();  
1024: jPanel76 = new javax.swing.JPanel();  
1025: tp66 = new javax.swing.JTextField();  
1026: jPanel77 = new javax.swing.JPanel();  
1027: tp74 = new javax.swing.JTextField();  
1028: jPanel78 = new javax.swing.JPanel();  
1029: tp82 = new javax.swing.JTextField();  
1030: jPanel79 = new javax.swing.JPanel();  
1031: tp90 = new javax.swing.JTextField();  
1032: jPanel80 = new javax.swing.JPanel();  
1033: tp98 = new javax.swing.JTextField();  
1034: jPanel81 = new javax.swing.JPanel();  
1035: tp106 = new javax.swing.JTextField();  
1036: jPanel82 = new javax.swing.JPanel();  
1037: tp114 = new javax.swing.JTextField();  
1038: jPanel83 = new javax.swing.JPanel();  
1039: tp122 = new javax.swing.JTextField();  
1040: jPanel84 = new javax.swing.JPanel();  
1041: tp130 = new javax.swing.JTextField();  
1042: jPanel85 = new javax.swing.JPanel();  
1043: tp4 = new javax.swing.JTextField();  
1044: jPanel86 = new javax.swing.JPanel();  
1045: tp12 = new javax.swing.JTextField();  
1046: jPanel87 = new javax.swing.JPanel();  
1047: tp20 = new javax.swing.JTextField();  
1048: jPanel88 = new javax.swing.JPanel();  
1049: tp28 = new javax.swing.JTextField();  
1050: jPanel89 = new javax.swing.JPanel();  
1051: tp36 = new javax.swing.JTextField();  
1052: jPanel90 = new javax.swing.JPanel();  
1053: tp44 = new javax.swing.JTextField();  
1054: jPanel91 = new javax.swing.JPanel();  
1055: tp52 = new javax.swing.JTextField();  
1056: jPanel92 = new javax.swing.JPanel();  
1057: tp60 = new javax.swing.JTextField();  
1058: jPanel93 = new javax.swing.JPanel();  
1059: tp68 = new javax.swing.JTextField();  
1060: jPanel94 = new javax.swing.JPanel();  
1061: tp76 = new javax.swing.JTextField();  
1062: jPanel95 = new javax.swing.JPanel();  
1063: tp84 = new javax.swing.JTextField();  
1064: jPanel96 = new javax.swing.JPanel();  
1065: tp92 = new javax.swing.JTextField();  
1066: jPanel97 = new javax.swing.JPanel();  
1067: tp100 = new javax.swing.JTextField();  
1068: jPanel98 = new javax.swing.JPanel();  
1069: tp108 = new javax.swing.JTextField();  
1070: jPanel99 = new javax.swing.JPanel();  
1071: tp116 = new javax.swing.JTextField();  
1072: jPanel100 = new javax.swing.JPanel();  
1073: tp124 = new javax.swing.JTextField();  
1074: jPanel101 = new javax.swing.JPanel();  
1075: tp132 = new javax.swing.JTextField();  
1076: jPanel102 = new javax.swing.JPanel();  
1077: tp6 = new javax.swing.JTextField();  
1078: jPanel103 = new javax.swing.JPanel();  
1079: tp14 = new javax.swing.JTextField();  
1080: jPanel104 = new javax.swing.JPanel();  
1081: tp22 = new javax.swing.JTextField();  
1082: jPanel105 = new javax.swing.JPanel();  
1083: tp30 = new javax.swing.JTextField();  
1084: jPanel106 = new javax.swing.JPanel();  
1085: tp38 = new javax.swing.JTextField();  
1086: jPanel107 = new javax.swing.JPanel();  
1087: tp46 = new javax.swing.JTextField();  
1088: jPanel108 = new javax.swing.JPanel();  
1089: tp54 = new javax.swing.JTextField();  
1090: jPanel109 = new javax.swing.JPanel();  
1091: tp62 = new javax.swing.JTextField();  
1092: jPanel110 = new javax.swing.JPanel();  
1093: tp70 = new javax.swing.JTextField();  
1094: jPanel111 = new javax.swing.JPanel();  
1095: tp78 = new javax.swing.JTextField();  
1096: jPanel112 = new javax.swing.JPanel();  
1097: tp86 = new javax.swing.JTextField();  
1098: jPanel113 = new javax.swing.JPanel();  
1099: tp94 = new javax.swing.JTextField();  
1100: jPanel114 = new javax.swing.JPanel();  
1101: tp102 = new javax.swing.JTextField();  
1102: jPanel115 = new javax.swing.JPanel();  
1103: tp110 = new javax.swing.JTextField();  
1104: jPanel116 = new javax.swing.JPanel();  
1105: tp118 = new javax.swing.JTextField();  
1106: jPanel117 = new javax.swing.JPanel();  
1107: tp126 = new javax.swing.JTextField();  
1108: jPanel118 = new javax.swing.JPanel();  
1109: tp134 = new javax.swing.JTextField();  
1110: lblPesquisador1 = new javax.swing.JLabel();  
1111: txtPesquisador2 = new javax.swing.JTextField();  
1112: txtTotalPesados = new javax.swing.JTextField();  
1113: jLabel37 = new javax.swing.JLabel();  
1114: lblFolha2 = new javax.swing.JLabel();  
1115: jLabel40 = new javax.swing.JLabel();  
1116: btnSalvarForms = new javax.swing.JButton();  
1117: jLabel34 = new javax.swing.JLabel();  
1118: lblTrecho\_dados = new javax.swing.JLabel();  
1119: btnApagar = new javax.swing.JButton();  
1120: cmbData = new javax.swing.JComboBox();  
1121: pnl\_servidor = new javax.swing.JPanel();  
1122: jLabel1 = new javax.swing.JLabel();  
1123: jScrollPane1 = new javax.swing.JScrollPane();  
1124: txtAreaLog = new javax.swing.JTextArea();  
1125: txtPorta = new javax.swing.JTextField();  
1126: jLabel2 = new javax.swing.JLabel();  
1127: pnl\_envio = new javax.swing.JPanel();  
1128: jPanel2 = new javax.swing.JPanel();  
1129: jLabel3 = new javax.swing.JLabel();  
1130: btnInDados = new javax.swing.JButton();  
1131: jPanel1 = new javax.swing.JPanel();  
1132: btnExpVolDate = new javax.swing.JButton();  
1133: jLabel4 = new javax.swing.JLabel();  
1134: jLabel35 = new javax.swing.JLabel();  
1135: cmbDateExpVol = new javax.swing.JComboBox<String>();  
1136: jLabel71 = new javax.swing.JLabel();  
1137: cmbDateExpOD = new javax.swing.JComboBox<String>();  
1138: jLabel72 = new javax.swing.JLabel();  
1139: btnExpODDate = new javax.swing.JButton();  
1140: pnl\_sumario\_volumetrica = new javax.swing.JPanel();  
1141: jLabel38 = new javax.swing.JLabel();  
1142: cmbDataSumVol = new javax.swing.JComboBox();  
1143: chk\_exportadas\_sumVol = new javax.swing.JCheckBox();  
1144: chk\_nao\_exportadas\_sumVol = new javax.swing.JCheckBox();  
1145: jLabel41 = new javax.swing.JLabel();  
1146: jLabel42 = new javax.swing.JLabel();  
1147: jLabel43 = new javax.swing.JLabel();  
1148: jLabel44 = new javax.swing.JLabel();  
1149: jLabel45 = new javax.swing.JLabel();  
1150: jLabel46 = new javax.swing.JLabel();  
1151: jLabel47 = new javax.swing.JLabel();  
1152: jLabel48 = new javax.swing.JLabel();  
1153: jLabel49 = new javax.swing.JLabel();  
1154: jLabel50 = new javax.swing.JLabel();  
1155: jLabel51 = new javax.swing.JLabel();  
1156: jLabel52 = new javax.swing.JLabel();  
1157: txtSumVolP1 = new javax.swing.JTextField();  
1158: txtSumVolP2 = new javax.swing.JTextField();  
1159: txtSumVolP3 = new javax.swing.JTextField();  
1160: txtSumVolM = new javax.swing.JTextField();  
1161: txtSumVolO1 = new javax.swing.JTextField();  
1162: txtSumVolO2 = new javax.swing.JTextField();  
1163: txtSumVolO3 = new javax.swing.JTextField();  
1164: txtSumVolC1 = new javax.swing.JTextField();  
1165: txtSumVolC2 = new javax.swing.JTextField();  
1166: txtSumVolC3 = new javax.swing.JTextField();  
1167: txtSumVolC4 = new javax.swing.JTextField();  
1168: txtSumVolC5 = new javax.swing.JTextField();  
1169: jLabel53 = new javax.swing.JLabel();  
1170: jLabel54 = new javax.swing.JLabel();  
1171: jLabel55 = new javax.swing.JLabel();  
1172: jLabel56 = new javax.swing.JLabel();  
1173: jLabel57 = new javax.swing.JLabel();  
1174: jLabel58 = new javax.swing.JLabel();  
1175: jLabel59 = new javax.swing.JLabel();  
1176: jLabel60 = new javax.swing.JLabel();  
1177: jLabel61 = new javax.swing.JLabel();  
1178: jLabel62 = new javax.swing.JLabel();  
1179: jLabel63 = new javax.swing.JLabel();  
1180: jLabel64 = new javax.swing.JLabel();  
1181: jLabel65 = new javax.swing.JLabel();  
1182: jLabel66 = new javax.swing.JLabel();  
1183: jLabel67 = new javax.swing.JLabel();  
1184: jLabel68 = new javax.swing.JLabel();  
1185: jLabel69 = new javax.swing.JLabel();  
1186: txtSumVolS1 = new javax.swing.JTextField();  
1187: txtSumVolS2 = new javax.swing.JTextField();  
1188: txtSumVolS3 = new javax.swing.JTextField();  
1189: txtSumVolS4 = new javax.swing.JTextField();  
1190: txtSumVolS5 = new javax.swing.JTextField();  
1191: txtSumVolS6 = new javax.swing.JTextField();  
1192: txtSumVolSE1 = new javax.swing.JTextField();  
1193: txtSumVolSE2 = new javax.swing.JTextField();  
1194: txtSumVolSE3 = new javax.swing.JTextField();  
1195: txtSumVolSE4 = new javax.swing.JTextField();  
1196: txtSumVolSE5 = new javax.swing.JTextField();  
1197: txtSumVolR1 = new javax.swing.JTextField();  
1198: txtSumVolR2 = new javax.swing.JTextField();  
1199: txtSumVolR3 = new javax.swing.JTextField();  
1200: txtSumVolR4 = new javax.swing.JTextField();  
1201: txtSumVolR5 = new javax.swing.JTextField();  
1202: txtSumVolR6 = new javax.swing.JTextField();  
1203: jLabel70 = new javax.swing.JLabel();  
1204: jScrollPane2 = new javax.swing.JScrollPane();  
1205: tblSumVol = new javax.swing.JTable();  
1206: pnl\_relatorio = new javax.swing.JPanel();  
1207: jScrollPane5 = new javax.swing.JScrollPane();  
1208: relatorio = new javax.swing.JTable();  
1209: odStatus = new javax.swing.JLabel();  
1210:  
1211: dadosFileChooser.setDialogTitle("Importar dados iPad");  
1212: dadosFileChooser.setFileFilter(new SQLiteFilter());  
1213:  
1214: exporterFileChooser.setDialogTitle("Exportação de dados");  
1215: exporterFileChooser.setFileFilter(new ZipFilter());  
1216:  
1217: setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);  
1218: setTitle(getTituloJanela());  
1219: setIconImage(new ImageIcon(Janela.this.getClass().getResource("/images/icon.png")).getImage());  
1220: getContentPane().setLayout(new java.awt.GridBagLayout());  
1221:  
1222: alertaTreinamento.setPreferredSize(new java.awt.Dimension(1451, 20));  
1223: alertaTreinamento.setLayout(new javax.swing.BoxLayout(alertaTreinamento, javax.swing.BoxLayout.LINE\_AXIS));  
1224:  
1225: txtAlertTreinamento.setEditable(false);  
1226: txtAlertTreinamento.setBackground(new java.awt.Color(255, 0, 0));  
1227: txtAlertTreinamento.setFont(new java.awt.Font("Lucida Grande", 0, 24)); // NOI18N  
1228: txtAlertTreinamento.setForeground(new java.awt.Color(255, 255, 255));  
1229: txtAlertTreinamento.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1230: txtAlertTreinamento.setText("Modo de Treinamento");  
1231: alertaTreinamento.add(txtAlertTreinamento);  
1232:  
1233: gridBagConstraints = new java.awt.GridBagConstraints();  
1234: gridBagConstraints.gridx = 0;  
1235: gridBagConstraints.gridy = 0;  
1236: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1237: gridBagConstraints.anchor = java.awt.GridBagConstraints.PAGE\_START;  
1238: getContentPane().add(alertaTreinamento, gridBagConstraints);  
1239:  
1240: tabRelatorio.setDoubleBuffered(true);  
1241: tabRelatorio.addChangeListener(new javax.swing.event.ChangeListener() {  
1242: public void stateChanged(javax.swing.event.ChangeEvent evt) {  
1243: tabRelatorioStateChanged(evt);  
1244: }  
1245: });  
1246:  
1247: lblPosto.setText("Posto:");  
1248:  
1249: lblPosto\_dados.setText("-----");  
1250:  
1251: lblHora.setText("Hora Inicial:");  
1252:  
1253: cmbHora.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "0", "2", "4", "6", "8", "10", "12", "14", "16", "18", "20", "22" }));  
1254: cmbHora.setSelectedIndex(-1);  
1255: cmbHora.setToolTipText("");  
1256: cmbHora.addActionListener(new java.awt.event.ActionListener() {  
1257: public void actionPerformed(java.awt.event.ActionEvent evt) {  
1258: cmbHoraActionPerformed(evt);  
1259: }  
1260: });  
1261:  
1262: lblLocal.setText("Local:");  
1263:  
1264: lblSentido.setText("Sentido:");  
1265:  
1266: grpSentido.add(rdo\_SentidoBA);  
1267: rdo\_SentidoBA.setText("B->A");  
1268: rdo\_SentidoBA.addActionListener(new java.awt.event.ActionListener() {  
1269: public void actionPerformed(java.awt.event.ActionEvent evt) {  
1270: rdo\_SentidoBAActionPerformed(evt);  
1271: }  
1272: });  
1273:  
1274: grpSentido.add(rdo\_SentidoAB);  
1275: rdo\_SentidoAB.setText("A->B");  
1276: rdo\_SentidoAB.addActionListener(new java.awt.event.ActionListener() {  
1277: public void actionPerformed(java.awt.event.ActionEvent evt) {  
1278: rdo\_SentidoABActionPerformed(evt);  
1279: }  
1280: });  
1281:  
1282: lblPista.setText("Pista:");  
1283:  
1284: grpPista.add(rdo\_PistaSimples);  
1285: rdo\_PistaSimples.setText("Simples");  
1286:  
1287: grpPista.add(rdo\_PistaDupla);  
1288: rdo\_PistaDupla.setText("Dupla");  
1289:  
1290: lblData.setText("Data:");  
1291:  
1292: jTabbedPane1.setName(""); // NOI18N  
1293:  
1294: lblPesquisador.setText("Pesquisador\_1:");  
1295:  
1296: txtPesquisador1.setNextFocusableComponent(tl0);  
1297:  
1298: jPanel119.setPreferredSize(new java.awt.Dimension(616, 458));  
1299:  
1300: panelLeves.setBackground(new java.awt.Color(255, 255, 255));  
1301: panelLeves.setMinimumSize(new java.awt.Dimension(0, 0));  
1302: panelLeves.setLayout(new java.awt.GridBagLayout());  
1303:  
1304: leves\_hora1.setEditable(false);  
1305: leves\_hora1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1306: gridBagConstraints = new java.awt.GridBagConstraints();  
1307: gridBagConstraints.gridx = 2;  
1308: gridBagConstraints.gridy = 0;  
1309: gridBagConstraints.gridwidth = 4;  
1310: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1311: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1312: panelLeves.add(leves\_hora1, gridBagConstraints);  
1313:  
1314: lev15\_1.setEditable(false);  
1315: lev15\_1.setBackground(new java.awt.Color(204, 204, 204));  
1316: lev15\_1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1317: lev15\_1.setText("15");  
1318: gridBagConstraints = new java.awt.GridBagConstraints();  
1319: gridBagConstraints.gridx = 2;  
1320: gridBagConstraints.gridy = 1;  
1321: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1322: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1323: panelLeves.add(lev15\_1, gridBagConstraints);  
1324:  
1325: lev30\_1.setEditable(false);  
1326: lev30\_1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1327: lev30\_1.setText("30");  
1328: gridBagConstraints = new java.awt.GridBagConstraints();  
1329: gridBagConstraints.gridx = 3;  
1330: gridBagConstraints.gridy = 1;  
1331: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1332: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1333: panelLeves.add(lev30\_1, gridBagConstraints);  
1334:  
1335: lev45\_1.setEditable(false);  
1336: lev45\_1.setBackground(new java.awt.Color(204, 204, 204));  
1337: lev45\_1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1338: lev45\_1.setText("45");  
1339: gridBagConstraints = new java.awt.GridBagConstraints();  
1340: gridBagConstraints.gridx = 4;  
1341: gridBagConstraints.gridy = 1;  
1342: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1343: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1344: gridBagConstraints.weightx = 1.0;  
1345: panelLeves.add(lev45\_1, gridBagConstraints);  
1346:  
1347: lev60\_1.setEditable(false);  
1348: lev60\_1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1349: lev60\_1.setText("60");  
1350: gridBagConstraints = new java.awt.GridBagConstraints();  
1351: gridBagConstraints.gridx = 5;  
1352: gridBagConstraints.gridy = 1;  
1353: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1354: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1355: panelLeves.add(lev60\_1, gridBagConstraints);  
1356:  
1357: leves\_hora2.setEditable(false);  
1358: leves\_hora2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1359: gridBagConstraints = new java.awt.GridBagConstraints();  
1360: gridBagConstraints.gridx = 6;  
1361: gridBagConstraints.gridy = 0;  
1362: gridBagConstraints.gridwidth = 4;  
1363: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1364: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1365: panelLeves.add(leves\_hora2, gridBagConstraints);  
1366:  
1367: lev15\_2.setEditable(false);  
1368: lev15\_2.setBackground(new java.awt.Color(204, 204, 204));  
1369: lev15\_2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1370: lev15\_2.setText("15");  
1371: gridBagConstraints = new java.awt.GridBagConstraints();  
1372: gridBagConstraints.gridx = 6;  
1373: gridBagConstraints.gridy = 1;  
1374: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1375: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1376: gridBagConstraints.weightx = 1.0;  
1377: panelLeves.add(lev15\_2, gridBagConstraints);  
1378:  
1379: lev30\_2.setEditable(false);  
1380: lev30\_2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1381: lev30\_2.setText("30");  
1382: gridBagConstraints = new java.awt.GridBagConstraints();  
1383: gridBagConstraints.gridx = 7;  
1384: gridBagConstraints.gridy = 1;  
1385: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1386: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1387: panelLeves.add(lev30\_2, gridBagConstraints);  
1388:  
1389: lev45\_2.setEditable(false);  
1390: lev45\_2.setBackground(new java.awt.Color(204, 204, 204));  
1391: lev45\_2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1392: lev45\_2.setText("45");  
1393: gridBagConstraints = new java.awt.GridBagConstraints();  
1394: gridBagConstraints.gridx = 8;  
1395: gridBagConstraints.gridy = 1;  
1396: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1397: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1398: gridBagConstraints.weightx = 1.0;  
1399: panelLeves.add(lev45\_2, gridBagConstraints);  
1400:  
1401: lev60\_2.setEditable(false);  
1402: lev60\_2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1403: lev60\_2.setText("60");  
1404: gridBagConstraints = new java.awt.GridBagConstraints();  
1405: gridBagConstraints.gridx = 9;  
1406: gridBagConstraints.gridy = 1;  
1407: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1408: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
1409: panelLeves.add(lev60\_2, gridBagConstraints);  
1410:  
1411: lblP1.setBackground(new java.awt.Color(255, 255, 255));  
1412: lblP1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
1413: lblP1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/p1\_01.png"))); // NOI18N  
1414: lblP1.setToolTipText("");  
1415: lblP1.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
1416: lblP1.setFocusable(false);  
1417: lblP1.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
1418: lblP1.setMinimumSize(new java.awt.Dimension(200, 76));  
1419: lblP1.setName(""); // NOI18N  
1420: lblP1.setOpaque(true);  
1421: gridBagConstraints = new java.awt.GridBagConstraints();  
1422: gridBagConstraints.gridx = 0;  
1423: gridBagConstraints.gridy = 2;  
1424: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1425: panelLeves.add(lblP1, gridBagConstraints);  
1426:  
1427: jPanel3.setToolTipText("");  
1428: jPanel3.setLayout(new java.awt.GridBagLayout());  
1429:  
1430: tl0.setColumns(5);  
1431: tl0.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1432: tl0.setToolTipText("");  
1433: tl0.setInputVerifier(intVerifier);  
1434: tl0.setMinimumSize(new java.awt.Dimension(50, 19));  
1435: tl0.setName(""); // NOI18N  
1436: tl0.setNextFocusableComponent(tl1);  
1437: gridBagConstraints = new java.awt.GridBagConstraints();  
1438: gridBagConstraints.gridx = 0;  
1439: gridBagConstraints.gridy = 0;  
1440: gridBagConstraints.weightx = 1.0;  
1441: jPanel3.add(tl0, gridBagConstraints);  
1442:  
1443: gridBagConstraints = new java.awt.GridBagConstraints();  
1444: gridBagConstraints.gridx = 2;  
1445: gridBagConstraints.gridy = 2;  
1446: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
1447: gridBagConstraints.weightx = 1.0;  
1448: panelLeves.add(jPanel3, gridBagConstraints);  
1449:  
1450: tl1.setColumns(5);  
1451: tl1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1452: tl1.setToolTipText("");  
1453: tl1.setInputVerifier(intVerifier);  
1454: tl1.setMinimumSize(new java.awt.Dimension(50, 19));  
1455: tl1.setName(""); // NOI18N  
1456: tl1.setNextFocusableComponent(tl2);  
1457: gridBagConstraints = new java.awt.GridBagConstraints();  
1458: gridBagConstraints.gridx = 3;  
1459: gridBagConstraints.gridy = 2;  
1460: gridBagConstraints.weightx = 1.0;  
1461: panelLeves.add(tl1, gridBagConstraints);  
1462:  
1463: tl3.setColumns(5);  
1464: tl3.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1465: tl3.setToolTipText("");  
1466: tl3.setInputVerifier(intVerifier);  
1467: tl3.setMinimumSize(new java.awt.Dimension(50, 19));  
1468: tl3.setName(""); // NOI18N  
1469: tl3.setNextFocusableComponent(tl4);  
1470: gridBagConstraints = new java.awt.GridBagConstraints();  
1471: gridBagConstraints.gridx = 5;  
1472: gridBagConstraints.gridy = 2;  
1473: gridBagConstraints.weightx = 1.0;  
1474: panelLeves.add(tl3, gridBagConstraints);  
1475:  
1476: tl5.setColumns(5);  
1477: tl5.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1478: tl5.setToolTipText("");  
1479: tl5.setInputVerifier(intVerifier);  
1480: tl5.setMinimumSize(new java.awt.Dimension(50, 19));  
1481: tl5.setName(""); // NOI18N  
1482: tl5.setNextFocusableComponent(tl6);  
1483: gridBagConstraints = new java.awt.GridBagConstraints();  
1484: gridBagConstraints.gridx = 7;  
1485: gridBagConstraints.gridy = 2;  
1486: gridBagConstraints.weightx = 1.0;  
1487: panelLeves.add(tl5, gridBagConstraints);  
1488:  
1489: tl7.setColumns(5);  
1490: tl7.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1491: tl7.setToolTipText("");  
1492: tl7.setInputVerifier(intVerifier);  
1493: tl7.setMinimumSize(new java.awt.Dimension(50, 19));  
1494: tl7.setName(""); // NOI18N  
1495: tl7.setNextFocusableComponent(tl8);  
1496: gridBagConstraints = new java.awt.GridBagConstraints();  
1497: gridBagConstraints.gridx = 9;  
1498: gridBagConstraints.gridy = 2;  
1499: gridBagConstraints.weightx = 1.0;  
1500: panelLeves.add(tl7, gridBagConstraints);  
1501:  
1502: lblP3.setBackground(new java.awt.Color(255, 255, 255));  
1503: lblP3.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
1504: lblP3.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/p3.png"))); // NOI18N  
1505: lblP3.setToolTipText("");  
1506: lblP3.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
1507: lblP3.setFocusable(false);  
1508: lblP3.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
1509: lblP3.setMinimumSize(new java.awt.Dimension(200, 76));  
1510: lblP3.setOpaque(true);  
1511: gridBagConstraints = new java.awt.GridBagConstraints();  
1512: gridBagConstraints.gridx = 0;  
1513: gridBagConstraints.gridy = 3;  
1514: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1515: panelLeves.add(lblP3, gridBagConstraints);  
1516:  
1517: tl9.setColumns(5);  
1518: tl9.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1519: tl9.setToolTipText("");  
1520: tl9.setInputVerifier(intVerifier);  
1521: tl9.setMinimumSize(new java.awt.Dimension(50, 19));  
1522: tl9.setName(""); // NOI18N  
1523: tl9.setNextFocusableComponent(tl10);  
1524: gridBagConstraints = new java.awt.GridBagConstraints();  
1525: gridBagConstraints.gridx = 3;  
1526: gridBagConstraints.gridy = 3;  
1527: gridBagConstraints.weightx = 1.0;  
1528: panelLeves.add(tl9, gridBagConstraints);  
1529:  
1530: tl11.setColumns(5);  
1531: tl11.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1532: tl11.setToolTipText("");  
1533: tl11.setInputVerifier(intVerifier);  
1534: tl11.setMinimumSize(new java.awt.Dimension(50, 19));  
1535: tl11.setName(""); // NOI18N  
1536: tl11.setNextFocusableComponent(tl12);  
1537: gridBagConstraints = new java.awt.GridBagConstraints();  
1538: gridBagConstraints.gridx = 5;  
1539: gridBagConstraints.gridy = 3;  
1540: gridBagConstraints.weightx = 1.0;  
1541: panelLeves.add(tl11, gridBagConstraints);  
1542:  
1543: tl13.setColumns(5);  
1544: tl13.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1545: tl13.setToolTipText("");  
1546: tl13.setInputVerifier(intVerifier);  
1547: tl13.setMinimumSize(new java.awt.Dimension(50, 19));  
1548: tl13.setName(""); // NOI18N  
1549: tl13.setNextFocusableComponent(tl14);  
1550: gridBagConstraints = new java.awt.GridBagConstraints();  
1551: gridBagConstraints.gridx = 7;  
1552: gridBagConstraints.gridy = 3;  
1553: gridBagConstraints.weightx = 1.0;  
1554: panelLeves.add(tl13, gridBagConstraints);  
1555:  
1556: tl15.setColumns(5);  
1557: tl15.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1558: tl15.setToolTipText("");  
1559: tl15.setInputVerifier(intVerifier);  
1560: tl15.setMinimumSize(new java.awt.Dimension(50, 19));  
1561: tl15.setName(""); // NOI18N  
1562: tl15.setNextFocusableComponent(tl16);  
1563: gridBagConstraints = new java.awt.GridBagConstraints();  
1564: gridBagConstraints.gridx = 9;  
1565: gridBagConstraints.gridy = 3;  
1566: gridBagConstraints.weightx = 1.0;  
1567: panelLeves.add(tl15, gridBagConstraints);  
1568:  
1569: lblP2.setBackground(new java.awt.Color(255, 255, 255));  
1570: lblP2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
1571: lblP2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/p2.png"))); // NOI18N  
1572: lblP2.setToolTipText("");  
1573: lblP2.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
1574: lblP2.setFocusable(false);  
1575: lblP2.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
1576: lblP2.setMinimumSize(new java.awt.Dimension(200, 76));  
1577: lblP2.setOpaque(true);  
1578: gridBagConstraints = new java.awt.GridBagConstraints();  
1579: gridBagConstraints.gridx = 0;  
1580: gridBagConstraints.gridy = 4;  
1581: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1582: panelLeves.add(lblP2, gridBagConstraints);  
1583:  
1584: tl17.setColumns(5);  
1585: tl17.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1586: tl17.setToolTipText("");  
1587: tl17.setInputVerifier(intVerifier);  
1588: tl17.setMinimumSize(new java.awt.Dimension(50, 19));  
1589: tl17.setName(""); // NOI18N  
1590: tl17.setNextFocusableComponent(tl18);  
1591: gridBagConstraints = new java.awt.GridBagConstraints();  
1592: gridBagConstraints.gridx = 3;  
1593: gridBagConstraints.gridy = 4;  
1594: gridBagConstraints.weightx = 1.0;  
1595: panelLeves.add(tl17, gridBagConstraints);  
1596:  
1597: tl19.setColumns(5);  
1598: tl19.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1599: tl19.setToolTipText("");  
1600: tl19.setInputVerifier(intVerifier);  
1601: tl19.setMinimumSize(new java.awt.Dimension(50, 19));  
1602: tl19.setName(""); // NOI18N  
1603: tl19.setNextFocusableComponent(tl20);  
1604: gridBagConstraints = new java.awt.GridBagConstraints();  
1605: gridBagConstraints.gridx = 5;  
1606: gridBagConstraints.gridy = 4;  
1607: gridBagConstraints.weightx = 1.0;  
1608: panelLeves.add(tl19, gridBagConstraints);  
1609:  
1610: tl21.setColumns(5);  
1611: tl21.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1612: tl21.setToolTipText("");  
1613: tl21.setInputVerifier(intVerifier);  
1614: tl21.setMinimumSize(new java.awt.Dimension(50, 19));  
1615: tl21.setName(""); // NOI18N  
1616: tl21.setNextFocusableComponent(tl22);  
1617: gridBagConstraints = new java.awt.GridBagConstraints();  
1618: gridBagConstraints.gridx = 7;  
1619: gridBagConstraints.gridy = 4;  
1620: gridBagConstraints.weightx = 1.0;  
1621: panelLeves.add(tl21, gridBagConstraints);  
1622:  
1623: tl23.setColumns(5);  
1624: tl23.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1625: tl23.setToolTipText("");  
1626: tl23.setInputVerifier(intVerifier);  
1627: tl23.setMinimumSize(new java.awt.Dimension(50, 19));  
1628: tl23.setName(""); // NOI18N  
1629: tl23.setNextFocusableComponent(tl24);  
1630: gridBagConstraints = new java.awt.GridBagConstraints();  
1631: gridBagConstraints.gridx = 9;  
1632: gridBagConstraints.gridy = 4;  
1633: gridBagConstraints.weightx = 1.0;  
1634: panelLeves.add(tl23, gridBagConstraints);  
1635:  
1636: lblM.setBackground(new java.awt.Color(255, 255, 255));  
1637: lblM.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
1638: lblM.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/m.png"))); // NOI18N  
1639: lblM.setToolTipText("");  
1640: lblM.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
1641: lblM.setFocusable(false);  
1642: lblM.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
1643: lblM.setMinimumSize(new java.awt.Dimension(200, 76));  
1644: lblM.setOpaque(true);  
1645: gridBagConstraints = new java.awt.GridBagConstraints();  
1646: gridBagConstraints.gridx = 0;  
1647: gridBagConstraints.gridy = 5;  
1648: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1649: panelLeves.add(lblM, gridBagConstraints);  
1650:  
1651: tl25.setColumns(5);  
1652: tl25.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1653: tl25.setToolTipText("");  
1654: tl25.setInputVerifier(intVerifier);  
1655: tl25.setMinimumSize(new java.awt.Dimension(50, 19));  
1656: tl25.setName(""); // NOI18N  
1657: tl25.setNextFocusableComponent(tl26);  
1658: gridBagConstraints = new java.awt.GridBagConstraints();  
1659: gridBagConstraints.gridx = 3;  
1660: gridBagConstraints.gridy = 5;  
1661: gridBagConstraints.weightx = 1.0;  
1662: panelLeves.add(tl25, gridBagConstraints);  
1663:  
1664: tl27.setColumns(5);  
1665: tl27.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1666: tl27.setToolTipText("");  
1667: tl27.setInputVerifier(intVerifier);  
1668: tl27.setMinimumSize(new java.awt.Dimension(50, 19));  
1669: tl27.setName(""); // NOI18N  
1670: tl27.setNextFocusableComponent(tl28);  
1671: gridBagConstraints = new java.awt.GridBagConstraints();  
1672: gridBagConstraints.gridx = 5;  
1673: gridBagConstraints.gridy = 5;  
1674: gridBagConstraints.weightx = 1.0;  
1675: panelLeves.add(tl27, gridBagConstraints);  
1676:  
1677: tl29.setColumns(5);  
1678: tl29.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1679: tl29.setToolTipText("");  
1680: tl29.setInputVerifier(intVerifier);  
1681: tl29.setMinimumSize(new java.awt.Dimension(50, 19));  
1682: tl29.setName(""); // NOI18N  
1683: tl29.setNextFocusableComponent(tl30);  
1684: gridBagConstraints = new java.awt.GridBagConstraints();  
1685: gridBagConstraints.gridx = 7;  
1686: gridBagConstraints.gridy = 5;  
1687: gridBagConstraints.weightx = 1.0;  
1688: panelLeves.add(tl29, gridBagConstraints);  
1689:  
1690: tl31.setColumns(5);  
1691: tl31.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1692: tl31.setToolTipText("");  
1693: tl31.setInputVerifier(intVerifier);  
1694: tl31.setMinimumSize(new java.awt.Dimension(50, 19));  
1695: tl31.setName(""); // NOI18N  
1696: tl31.setNextFocusableComponent(tl32);  
1697: gridBagConstraints = new java.awt.GridBagConstraints();  
1698: gridBagConstraints.gridx = 9;  
1699: gridBagConstraints.gridy = 5;  
1700: gridBagConstraints.weightx = 1.0;  
1701: panelLeves.add(tl31, gridBagConstraints);  
1702:  
1703: lbl2CB.setBackground(new java.awt.Color(255, 255, 255));  
1704: lbl2CB.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
1705: lbl2CB.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/o1.png"))); // NOI18N  
1706: lbl2CB.setToolTipText("");  
1707: lbl2CB.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
1708: lbl2CB.setFocusable(false);  
1709: lbl2CB.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
1710: lbl2CB.setMinimumSize(new java.awt.Dimension(200, 76));  
1711: lbl2CB.setOpaque(true);  
1712: gridBagConstraints = new java.awt.GridBagConstraints();  
1713: gridBagConstraints.gridx = 0;  
1714: gridBagConstraints.gridy = 6;  
1715: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1716: panelLeves.add(lbl2CB, gridBagConstraints);  
1717:  
1718: tl33.setColumns(5);  
1719: tl33.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1720: tl33.setToolTipText("");  
1721: tl33.setInputVerifier(intVerifier);  
1722: tl33.setMinimumSize(new java.awt.Dimension(50, 19));  
1723: tl33.setName(""); // NOI18N  
1724: tl33.setNextFocusableComponent(tl34);  
1725: gridBagConstraints = new java.awt.GridBagConstraints();  
1726: gridBagConstraints.gridx = 3;  
1727: gridBagConstraints.gridy = 6;  
1728: panelLeves.add(tl33, gridBagConstraints);  
1729:  
1730: tl35.setColumns(5);  
1731: tl35.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1732: tl35.setToolTipText("");  
1733: tl35.setInputVerifier(intVerifier);  
1734: tl35.setMinimumSize(new java.awt.Dimension(50, 19));  
1735: tl35.setName(""); // NOI18N  
1736: tl35.setNextFocusableComponent(tl36);  
1737: gridBagConstraints = new java.awt.GridBagConstraints();  
1738: gridBagConstraints.gridx = 5;  
1739: gridBagConstraints.gridy = 6;  
1740: panelLeves.add(tl35, gridBagConstraints);  
1741:  
1742: tl37.setColumns(5);  
1743: tl37.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1744: tl37.setToolTipText("");  
1745: tl37.setInputVerifier(intVerifier);  
1746: tl37.setMinimumSize(new java.awt.Dimension(50, 19));  
1747: tl37.setName(""); // NOI18N  
1748: tl37.setNextFocusableComponent(tl38);  
1749: gridBagConstraints = new java.awt.GridBagConstraints();  
1750: gridBagConstraints.gridx = 7;  
1751: gridBagConstraints.gridy = 6;  
1752: panelLeves.add(tl37, gridBagConstraints);  
1753:  
1754: tl39.setColumns(5);  
1755: tl39.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1756: tl39.setToolTipText("");  
1757: tl39.setInputVerifier(intVerifier);  
1758: tl39.setMinimumSize(new java.awt.Dimension(50, 19));  
1759: tl39.setName(""); // NOI18N  
1760: tl39.setNextFocusableComponent(tl40);  
1761: gridBagConstraints = new java.awt.GridBagConstraints();  
1762: gridBagConstraints.gridx = 9;  
1763: gridBagConstraints.gridy = 6;  
1764: panelLeves.add(tl39, gridBagConstraints);  
1765:  
1766: lbl3CB.setBackground(new java.awt.Color(255, 255, 255));  
1767: lbl3CB.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
1768: lbl3CB.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/o2.png"))); // NOI18N  
1769: lbl3CB.setToolTipText("");  
1770: lbl3CB.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
1771: lbl3CB.setFocusable(false);  
1772: lbl3CB.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
1773: lbl3CB.setMinimumSize(new java.awt.Dimension(200, 76));  
1774: lbl3CB.setOpaque(true);  
1775: gridBagConstraints = new java.awt.GridBagConstraints();  
1776: gridBagConstraints.gridx = 0;  
1777: gridBagConstraints.gridy = 7;  
1778: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1779: panelLeves.add(lbl3CB, gridBagConstraints);  
1780:  
1781: tl41.setColumns(5);  
1782: tl41.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1783: tl41.setToolTipText("");  
1784: tl41.setInputVerifier(intVerifier);  
1785: tl41.setMinimumSize(new java.awt.Dimension(50, 19));  
1786: tl41.setName(""); // NOI18N  
1787: tl41.setNextFocusableComponent(tl42);  
1788: gridBagConstraints = new java.awt.GridBagConstraints();  
1789: gridBagConstraints.gridx = 3;  
1790: gridBagConstraints.gridy = 7;  
1791: panelLeves.add(tl41, gridBagConstraints);  
1792:  
1793: tl43.setColumns(5);  
1794: tl43.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1795: tl43.setToolTipText("");  
1796: tl43.setInputVerifier(intVerifier);  
1797: tl43.setMinimumSize(new java.awt.Dimension(50, 19));  
1798: tl43.setName(""); // NOI18N  
1799: tl43.setNextFocusableComponent(tl44);  
1800: gridBagConstraints = new java.awt.GridBagConstraints();  
1801: gridBagConstraints.gridx = 5;  
1802: gridBagConstraints.gridy = 7;  
1803: panelLeves.add(tl43, gridBagConstraints);  
1804:  
1805: tl45.setColumns(5);  
1806: tl45.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1807: tl45.setToolTipText("");  
1808: tl45.setInputVerifier(intVerifier);  
1809: tl45.setMinimumSize(new java.awt.Dimension(50, 19));  
1810: tl45.setName(""); // NOI18N  
1811: tl45.setNextFocusableComponent(tl46);  
1812: gridBagConstraints = new java.awt.GridBagConstraints();  
1813: gridBagConstraints.gridx = 7;  
1814: gridBagConstraints.gridy = 7;  
1815: panelLeves.add(tl45, gridBagConstraints);  
1816:  
1817: tl47.setColumns(5);  
1818: tl47.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1819: tl47.setToolTipText("");  
1820: tl47.setInputVerifier(intVerifier);  
1821: tl47.setMinimumSize(new java.awt.Dimension(50, 19));  
1822: tl47.setName(""); // NOI18N  
1823: tl47.setNextFocusableComponent(tl48);  
1824: gridBagConstraints = new java.awt.GridBagConstraints();  
1825: gridBagConstraints.gridx = 9;  
1826: gridBagConstraints.gridy = 7;  
1827: panelLeves.add(tl47, gridBagConstraints);  
1828:  
1829: lbl4CB.setBackground(new java.awt.Color(255, 255, 255));  
1830: lbl4CB.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
1831: lbl4CB.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/o3.png"))); // NOI18N  
1832: lbl4CB.setToolTipText("");  
1833: lbl4CB.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
1834: lbl4CB.setFocusable(false);  
1835: lbl4CB.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
1836: lbl4CB.setMinimumSize(new java.awt.Dimension(200, 76));  
1837: lbl4CB.setOpaque(true);  
1838: gridBagConstraints = new java.awt.GridBagConstraints();  
1839: gridBagConstraints.gridx = 0;  
1840: gridBagConstraints.gridy = 8;  
1841: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1842: panelLeves.add(lbl4CB, gridBagConstraints);  
1843:  
1844: tl49.setColumns(5);  
1845: tl49.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1846: tl49.setToolTipText("");  
1847: tl49.setInputVerifier(intVerifier);  
1848: tl49.setMinimumSize(new java.awt.Dimension(50, 19));  
1849: tl49.setName(""); // NOI18N  
1850: tl49.setNextFocusableComponent(tl50);  
1851: gridBagConstraints = new java.awt.GridBagConstraints();  
1852: gridBagConstraints.gridx = 3;  
1853: gridBagConstraints.gridy = 8;  
1854: panelLeves.add(tl49, gridBagConstraints);  
1855:  
1856: tl51.setColumns(5);  
1857: tl51.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1858: tl51.setToolTipText("");  
1859: tl51.setInputVerifier(intVerifier);  
1860: tl51.setMinimumSize(new java.awt.Dimension(50, 19));  
1861: tl51.setName(""); // NOI18N  
1862: tl51.setNextFocusableComponent(tl52);  
1863: gridBagConstraints = new java.awt.GridBagConstraints();  
1864: gridBagConstraints.gridx = 5;  
1865: gridBagConstraints.gridy = 8;  
1866: panelLeves.add(tl51, gridBagConstraints);  
1867:  
1868: tl53.setColumns(5);  
1869: tl53.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1870: tl53.setToolTipText("");  
1871: tl53.setInputVerifier(intVerifier);  
1872: tl53.setMinimumSize(new java.awt.Dimension(50, 19));  
1873: tl53.setName(""); // NOI18N  
1874: tl53.setNextFocusableComponent(tl54);  
1875: gridBagConstraints = new java.awt.GridBagConstraints();  
1876: gridBagConstraints.gridx = 7;  
1877: gridBagConstraints.gridy = 8;  
1878: panelLeves.add(tl53, gridBagConstraints);  
1879:  
1880: tl55.setColumns(5);  
1881: tl55.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1882: tl55.setToolTipText("");  
1883: tl55.setInputVerifier(intVerifier);  
1884: tl55.setMinimumSize(new java.awt.Dimension(50, 19));  
1885: tl55.setName(""); // NOI18N  
1886: tl55.setNextFocusableComponent(tl56);  
1887: gridBagConstraints = new java.awt.GridBagConstraints();  
1888: gridBagConstraints.gridx = 9;  
1889: gridBagConstraints.gridy = 8;  
1890: panelLeves.add(tl55, gridBagConstraints);  
1891:  
1892: lbl2C.setBackground(new java.awt.Color(255, 255, 255));  
1893: lbl2C.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
1894: lbl2C.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/c1.png"))); // NOI18N  
1895: lbl2C.setToolTipText("");  
1896: lbl2C.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
1897: lbl2C.setFocusable(false);  
1898: lbl2C.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
1899: lbl2C.setMinimumSize(new java.awt.Dimension(200, 76));  
1900: lbl2C.setOpaque(true);  
1901: gridBagConstraints = new java.awt.GridBagConstraints();  
1902: gridBagConstraints.gridx = 0;  
1903: gridBagConstraints.gridy = 9;  
1904: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1905: panelLeves.add(lbl2C, gridBagConstraints);  
1906:  
1907: tl57.setColumns(5);  
1908: tl57.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1909: tl57.setToolTipText("");  
1910: tl57.setInputVerifier(intVerifier);  
1911: tl57.setMinimumSize(new java.awt.Dimension(50, 19));  
1912: tl57.setName(""); // NOI18N  
1913: tl57.setNextFocusableComponent(tl58);  
1914: gridBagConstraints = new java.awt.GridBagConstraints();  
1915: gridBagConstraints.gridx = 3;  
1916: gridBagConstraints.gridy = 9;  
1917: panelLeves.add(tl57, gridBagConstraints);  
1918:  
1919: tl59.setColumns(5);  
1920: tl59.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1921: tl59.setToolTipText("");  
1922: tl59.setInputVerifier(intVerifier);  
1923: tl59.setMinimumSize(new java.awt.Dimension(50, 19));  
1924: tl59.setName(""); // NOI18N  
1925: tl59.setNextFocusableComponent(tl60);  
1926: gridBagConstraints = new java.awt.GridBagConstraints();  
1927: gridBagConstraints.gridx = 5;  
1928: gridBagConstraints.gridy = 9;  
1929: panelLeves.add(tl59, gridBagConstraints);  
1930:  
1931: tl61.setColumns(5);  
1932: tl61.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1933: tl61.setToolTipText("");  
1934: tl61.setInputVerifier(intVerifier);  
1935: tl61.setMinimumSize(new java.awt.Dimension(50, 19));  
1936: tl61.setName(""); // NOI18N  
1937: tl61.setNextFocusableComponent(tl62);  
1938: gridBagConstraints = new java.awt.GridBagConstraints();  
1939: gridBagConstraints.gridx = 7;  
1940: gridBagConstraints.gridy = 9;  
1941: panelLeves.add(tl61, gridBagConstraints);  
1942:  
1943: tl63.setColumns(5);  
1944: tl63.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1945: tl63.setToolTipText("");  
1946: tl63.setInputVerifier(intVerifier);  
1947: tl63.setMinimumSize(new java.awt.Dimension(50, 19));  
1948: tl63.setName(""); // NOI18N  
1949: tl63.setNextFocusableComponent(tl64);  
1950: gridBagConstraints = new java.awt.GridBagConstraints();  
1951: gridBagConstraints.gridx = 9;  
1952: gridBagConstraints.gridy = 9;  
1953: panelLeves.add(tl63, gridBagConstraints);  
1954:  
1955: lbl3C.setBackground(new java.awt.Color(255, 255, 255));  
1956: lbl3C.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
1957: lbl3C.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/c2.png"))); // NOI18N  
1958: lbl3C.setToolTipText("");  
1959: lbl3C.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
1960: lbl3C.setFocusable(false);  
1961: lbl3C.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
1962: lbl3C.setMinimumSize(new java.awt.Dimension(200, 76));  
1963: lbl3C.setOpaque(true);  
1964: gridBagConstraints = new java.awt.GridBagConstraints();  
1965: gridBagConstraints.gridx = 0;  
1966: gridBagConstraints.gridy = 10;  
1967: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
1968: panelLeves.add(lbl3C, gridBagConstraints);  
1969:  
1970: tl65.setColumns(5);  
1971: tl65.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1972: tl65.setToolTipText("");  
1973: tl65.setInputVerifier(intVerifier);  
1974: tl65.setMinimumSize(new java.awt.Dimension(50, 19));  
1975: tl65.setName(""); // NOI18N  
1976: tl65.setNextFocusableComponent(tl66);  
1977: gridBagConstraints = new java.awt.GridBagConstraints();  
1978: gridBagConstraints.gridx = 3;  
1979: gridBagConstraints.gridy = 10;  
1980: panelLeves.add(tl65, gridBagConstraints);  
1981:  
1982: tl67.setColumns(5);  
1983: tl67.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1984: tl67.setToolTipText("");  
1985: tl67.setInputVerifier(intVerifier);  
1986: tl67.setMinimumSize(new java.awt.Dimension(50, 19));  
1987: tl67.setName(""); // NOI18N  
1988: tl67.setNextFocusableComponent(tl68);  
1989: gridBagConstraints = new java.awt.GridBagConstraints();  
1990: gridBagConstraints.gridx = 5;  
1991: gridBagConstraints.gridy = 10;  
1992: panelLeves.add(tl67, gridBagConstraints);  
1993:  
1994: tl69.setColumns(5);  
1995: tl69.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
1996: tl69.setToolTipText("");  
1997: tl69.setInputVerifier(intVerifier);  
1998: tl69.setMinimumSize(new java.awt.Dimension(50, 19));  
1999: tl69.setName(""); // NOI18N  
2000: tl69.setNextFocusableComponent(tl70);  
2001: gridBagConstraints = new java.awt.GridBagConstraints();  
2002: gridBagConstraints.gridx = 7;  
2003: gridBagConstraints.gridy = 10;  
2004: panelLeves.add(tl69, gridBagConstraints);  
2005:  
2006: tl71.setColumns(5);  
2007: tl71.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2008: tl71.setToolTipText("");  
2009: tl71.setInputVerifier(intVerifier);  
2010: tl71.setMinimumSize(new java.awt.Dimension(50, 19));  
2011: tl71.setName(""); // NOI18N  
2012: tl71.setNextFocusableComponent(tl72);  
2013: gridBagConstraints = new java.awt.GridBagConstraints();  
2014: gridBagConstraints.gridx = 9;  
2015: gridBagConstraints.gridy = 10;  
2016: panelLeves.add(tl71, gridBagConstraints);  
2017:  
2018: lbl4C.setBackground(new java.awt.Color(255, 255, 255));  
2019: lbl4C.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
2020: lbl4C.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/c3.png"))); // NOI18N  
2021: lbl4C.setToolTipText("");  
2022: lbl4C.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
2023: lbl4C.setFocusable(false);  
2024: lbl4C.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
2025: lbl4C.setMinimumSize(new java.awt.Dimension(200, 76));  
2026: lbl4C.setOpaque(true);  
2027: gridBagConstraints = new java.awt.GridBagConstraints();  
2028: gridBagConstraints.gridx = 0;  
2029: gridBagConstraints.gridy = 11;  
2030: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
2031: panelLeves.add(lbl4C, gridBagConstraints);  
2032:  
2033: tl73.setColumns(5);  
2034: tl73.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2035: tl73.setToolTipText("");  
2036: tl73.setInputVerifier(intVerifier);  
2037: tl73.setMinimumSize(new java.awt.Dimension(50, 19));  
2038: tl73.setName(""); // NOI18N  
2039: tl73.setNextFocusableComponent(tl74);  
2040: gridBagConstraints = new java.awt.GridBagConstraints();  
2041: gridBagConstraints.gridx = 3;  
2042: gridBagConstraints.gridy = 11;  
2043: panelLeves.add(tl73, gridBagConstraints);  
2044:  
2045: tl75.setColumns(5);  
2046: tl75.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2047: tl75.setToolTipText("");  
2048: tl75.setInputVerifier(intVerifier);  
2049: tl75.setMinimumSize(new java.awt.Dimension(50, 19));  
2050: tl75.setName(""); // NOI18N  
2051: tl75.setNextFocusableComponent(tl76);  
2052: gridBagConstraints = new java.awt.GridBagConstraints();  
2053: gridBagConstraints.gridx = 5;  
2054: gridBagConstraints.gridy = 11;  
2055: panelLeves.add(tl75, gridBagConstraints);  
2056:  
2057: tl77.setColumns(5);  
2058: tl77.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2059: tl77.setToolTipText("");  
2060: tl77.setInputVerifier(intVerifier);  
2061: tl77.setMinimumSize(new java.awt.Dimension(50, 19));  
2062: tl77.setName(""); // NOI18N  
2063: tl77.setNextFocusableComponent(tl78);  
2064: gridBagConstraints = new java.awt.GridBagConstraints();  
2065: gridBagConstraints.gridx = 7;  
2066: gridBagConstraints.gridy = 11;  
2067: panelLeves.add(tl77, gridBagConstraints);  
2068:  
2069: tl79.setColumns(5);  
2070: tl79.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2071: tl79.setToolTipText("");  
2072: tl79.setInputVerifier(intVerifier);  
2073: tl79.setMinimumSize(new java.awt.Dimension(50, 19));  
2074: tl79.setName(""); // NOI18N  
2075: tl79.setNextFocusableComponent(tl80);  
2076: gridBagConstraints = new java.awt.GridBagConstraints();  
2077: gridBagConstraints.gridx = 9;  
2078: gridBagConstraints.gridy = 11;  
2079: panelLeves.add(tl79, gridBagConstraints);  
2080:  
2081: lbl4CD.setBackground(new java.awt.Color(255, 255, 255));  
2082: lbl4CD.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
2083: lbl4CD.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/c4.png"))); // NOI18N  
2084: lbl4CD.setToolTipText("");  
2085: lbl4CD.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
2086: lbl4CD.setFocusable(false);  
2087: lbl4CD.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
2088: lbl4CD.setMinimumSize(new java.awt.Dimension(200, 76));  
2089: lbl4CD.setOpaque(true);  
2090: gridBagConstraints = new java.awt.GridBagConstraints();  
2091: gridBagConstraints.gridx = 0;  
2092: gridBagConstraints.gridy = 12;  
2093: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
2094: panelLeves.add(lbl4CD, gridBagConstraints);  
2095:  
2096: tl81.setColumns(5);  
2097: tl81.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2098: tl81.setToolTipText("");  
2099: tl81.setInputVerifier(intVerifier);  
2100: tl81.setMinimumSize(new java.awt.Dimension(50, 19));  
2101: tl81.setName(""); // NOI18N  
2102: tl81.setNextFocusableComponent(tl82);  
2103: gridBagConstraints = new java.awt.GridBagConstraints();  
2104: gridBagConstraints.gridx = 3;  
2105: gridBagConstraints.gridy = 12;  
2106: panelLeves.add(tl81, gridBagConstraints);  
2107:  
2108: tl83.setColumns(5);  
2109: tl83.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2110: tl83.setToolTipText("");  
2111: tl83.setInputVerifier(intVerifier);  
2112: tl83.setMinimumSize(new java.awt.Dimension(50, 19));  
2113: tl83.setName(""); // NOI18N  
2114: tl83.setNextFocusableComponent(tl84);  
2115: gridBagConstraints = new java.awt.GridBagConstraints();  
2116: gridBagConstraints.gridx = 5;  
2117: gridBagConstraints.gridy = 12;  
2118: panelLeves.add(tl83, gridBagConstraints);  
2119:  
2120: tl85.setColumns(5);  
2121: tl85.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2122: tl85.setToolTipText("");  
2123: tl85.setInputVerifier(intVerifier);  
2124: tl85.setMinimumSize(new java.awt.Dimension(50, 19));  
2125: tl85.setName(""); // NOI18N  
2126: tl85.setNextFocusableComponent(tl86);  
2127: gridBagConstraints = new java.awt.GridBagConstraints();  
2128: gridBagConstraints.gridx = 7;  
2129: gridBagConstraints.gridy = 12;  
2130: panelLeves.add(tl85, gridBagConstraints);  
2131:  
2132: tl87.setColumns(5);  
2133: tl87.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2134: tl87.setToolTipText("");  
2135: tl87.setInputVerifier(intVerifier);  
2136: tl87.setMinimumSize(new java.awt.Dimension(50, 19));  
2137: tl87.setName(""); // NOI18N  
2138: tl87.setNextFocusableComponent(tl88);  
2139: gridBagConstraints = new java.awt.GridBagConstraints();  
2140: gridBagConstraints.gridx = 9;  
2141: gridBagConstraints.gridy = 12;  
2142: panelLeves.add(tl87, gridBagConstraints);  
2143:  
2144: lbl3D.setBackground(new java.awt.Color(255, 255, 255));  
2145: lbl3D.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
2146: lbl3D.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/simples/c5.png"))); // NOI18N  
2147: lbl3D.setToolTipText("");  
2148: lbl3D.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
2149: lbl3D.setFocusable(false);  
2150: lbl3D.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
2151: lbl3D.setMinimumSize(new java.awt.Dimension(200, 76));  
2152: lbl3D.setOpaque(true);  
2153: gridBagConstraints = new java.awt.GridBagConstraints();  
2154: gridBagConstraints.gridx = 0;  
2155: gridBagConstraints.gridy = 13;  
2156: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
2157: panelLeves.add(lbl3D, gridBagConstraints);  
2158:  
2159: tl89.setColumns(5);  
2160: tl89.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2161: tl89.setToolTipText("");  
2162: tl89.setInputVerifier(intVerifier);  
2163: tl89.setMinimumSize(new java.awt.Dimension(50, 19));  
2164: tl89.setName(""); // NOI18N  
2165: tl89.setNextFocusableComponent(tl90);  
2166: gridBagConstraints = new java.awt.GridBagConstraints();  
2167: gridBagConstraints.gridx = 3;  
2168: gridBagConstraints.gridy = 13;  
2169: panelLeves.add(tl89, gridBagConstraints);  
2170:  
2171: tl91.setColumns(5);  
2172: tl91.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2173: tl91.setToolTipText("");  
2174: tl91.setInputVerifier(intVerifier);  
2175: tl91.setMinimumSize(new java.awt.Dimension(50, 19));  
2176: tl91.setName(""); // NOI18N  
2177: tl91.setNextFocusableComponent(tl92);  
2178: gridBagConstraints = new java.awt.GridBagConstraints();  
2179: gridBagConstraints.gridx = 5;  
2180: gridBagConstraints.gridy = 13;  
2181: panelLeves.add(tl91, gridBagConstraints);  
2182:  
2183: tl93.setColumns(5);  
2184: tl93.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2185: tl93.setToolTipText("");  
2186: tl93.setInputVerifier(intVerifier);  
2187: tl93.setMinimumSize(new java.awt.Dimension(50, 19));  
2188: tl93.setName(""); // NOI18N  
2189: tl93.setNextFocusableComponent(tl94);  
2190: gridBagConstraints = new java.awt.GridBagConstraints();  
2191: gridBagConstraints.gridx = 7;  
2192: gridBagConstraints.gridy = 13;  
2193: panelLeves.add(tl93, gridBagConstraints);  
2194:  
2195: tl95.setColumns(5);  
2196: tl95.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2197: tl95.setToolTipText("");  
2198: tl95.setInputVerifier(intVerifier);  
2199: tl95.setMinimumSize(new java.awt.Dimension(50, 19));  
2200: tl95.setName(""); // NOI18N  
2201: tl95.setNextFocusableComponent(txtPesquisador2);  
2202: gridBagConstraints = new java.awt.GridBagConstraints();  
2203: gridBagConstraints.gridx = 9;  
2204: gridBagConstraints.gridy = 13;  
2205: panelLeves.add(tl95, gridBagConstraints);  
2206:  
2207: jLabel5.setText("P1");  
2208: gridBagConstraints = new java.awt.GridBagConstraints();  
2209: gridBagConstraints.gridx = 1;  
2210: gridBagConstraints.gridy = 2;  
2211: panelLeves.add(jLabel5, gridBagConstraints);  
2212:  
2213: jLabel6.setText("P3");  
2214: gridBagConstraints = new java.awt.GridBagConstraints();  
2215: gridBagConstraints.gridx = 1;  
2216: gridBagConstraints.gridy = 3;  
2217: panelLeves.add(jLabel6, gridBagConstraints);  
2218:  
2219: jLabel7.setText("P2");  
2220: gridBagConstraints = new java.awt.GridBagConstraints();  
2221: gridBagConstraints.gridx = 1;  
2222: gridBagConstraints.gridy = 4;  
2223: panelLeves.add(jLabel7, gridBagConstraints);  
2224:  
2225: jLabel8.setText("M");  
2226: gridBagConstraints = new java.awt.GridBagConstraints();  
2227: gridBagConstraints.gridx = 1;  
2228: gridBagConstraints.gridy = 5;  
2229: panelLeves.add(jLabel8, gridBagConstraints);  
2230:  
2231: jLabel9.setText("2CB");  
2232: gridBagConstraints = new java.awt.GridBagConstraints();  
2233: gridBagConstraints.gridx = 1;  
2234: gridBagConstraints.gridy = 6;  
2235: panelLeves.add(jLabel9, gridBagConstraints);  
2236:  
2237: jLabel10.setText("3CB");  
2238: gridBagConstraints = new java.awt.GridBagConstraints();  
2239: gridBagConstraints.gridx = 1;  
2240: gridBagConstraints.gridy = 7;  
2241: panelLeves.add(jLabel10, gridBagConstraints);  
2242:  
2243: jLabel11.setText("4CB");  
2244: gridBagConstraints = new java.awt.GridBagConstraints();  
2245: gridBagConstraints.gridx = 1;  
2246: gridBagConstraints.gridy = 8;  
2247: panelLeves.add(jLabel11, gridBagConstraints);  
2248:  
2249: jLabel12.setText("2C");  
2250: gridBagConstraints = new java.awt.GridBagConstraints();  
2251: gridBagConstraints.gridx = 1;  
2252: gridBagConstraints.gridy = 9;  
2253: panelLeves.add(jLabel12, gridBagConstraints);  
2254:  
2255: jLabel13.setText("3C");  
2256: gridBagConstraints = new java.awt.GridBagConstraints();  
2257: gridBagConstraints.gridx = 1;  
2258: gridBagConstraints.gridy = 10;  
2259: panelLeves.add(jLabel13, gridBagConstraints);  
2260:  
2261: jLabel14.setText("4C");  
2262: gridBagConstraints = new java.awt.GridBagConstraints();  
2263: gridBagConstraints.gridx = 1;  
2264: gridBagConstraints.gridy = 11;  
2265: panelLeves.add(jLabel14, gridBagConstraints);  
2266:  
2267: jLabel15.setText("4CD");  
2268: gridBagConstraints = new java.awt.GridBagConstraints();  
2269: gridBagConstraints.gridx = 1;  
2270: gridBagConstraints.gridy = 12;  
2271: panelLeves.add(jLabel15, gridBagConstraints);  
2272:  
2273: jLabel16.setText("3D");  
2274: gridBagConstraints = new java.awt.GridBagConstraints();  
2275: gridBagConstraints.gridx = 1;  
2276: gridBagConstraints.gridy = 13;  
2277: panelLeves.add(jLabel16, gridBagConstraints);  
2278:  
2279: jPanel4.setToolTipText("");  
2280: jPanel4.setLayout(new java.awt.GridBagLayout());  
2281:  
2282: tl8.setColumns(5);  
2283: tl8.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2284: tl8.setToolTipText("");  
2285: tl8.setInputVerifier(intVerifier);  
2286: tl8.setMinimumSize(new java.awt.Dimension(50, 19));  
2287: tl8.setName(""); // NOI18N  
2288: tl8.setNextFocusableComponent(tl9);  
2289: gridBagConstraints = new java.awt.GridBagConstraints();  
2290: gridBagConstraints.gridx = 0;  
2291: gridBagConstraints.gridy = 0;  
2292: gridBagConstraints.weightx = 1.0;  
2293: jPanel4.add(tl8, gridBagConstraints);  
2294:  
2295: gridBagConstraints = new java.awt.GridBagConstraints();  
2296: gridBagConstraints.gridx = 2;  
2297: gridBagConstraints.gridy = 3;  
2298: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2299: gridBagConstraints.weightx = 1.0;  
2300: panelLeves.add(jPanel4, gridBagConstraints);  
2301:  
2302: jPanel5.setToolTipText("");  
2303: jPanel5.setLayout(new java.awt.GridBagLayout());  
2304:  
2305: tl16.setColumns(5);  
2306: tl16.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2307: tl16.setToolTipText("");  
2308: tl16.setInputVerifier(intVerifier);  
2309: tl16.setMinimumSize(new java.awt.Dimension(50, 19));  
2310: tl16.setName(""); // NOI18N  
2311: tl16.setNextFocusableComponent(tl17);  
2312: gridBagConstraints = new java.awt.GridBagConstraints();  
2313: gridBagConstraints.gridx = 0;  
2314: gridBagConstraints.gridy = 0;  
2315: gridBagConstraints.weightx = 1.0;  
2316: jPanel5.add(tl16, gridBagConstraints);  
2317:  
2318: gridBagConstraints = new java.awt.GridBagConstraints();  
2319: gridBagConstraints.gridx = 2;  
2320: gridBagConstraints.gridy = 4;  
2321: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2322: gridBagConstraints.weightx = 1.0;  
2323: panelLeves.add(jPanel5, gridBagConstraints);  
2324:  
2325: jPanel6.setToolTipText("");  
2326: jPanel6.setLayout(new java.awt.GridBagLayout());  
2327:  
2328: tl24.setColumns(5);  
2329: tl24.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2330: tl24.setToolTipText("");  
2331: tl24.setInputVerifier(intVerifier);  
2332: tl24.setMinimumSize(new java.awt.Dimension(50, 19));  
2333: tl24.setName(""); // NOI18N  
2334: tl24.setNextFocusableComponent(tl25);  
2335: gridBagConstraints = new java.awt.GridBagConstraints();  
2336: gridBagConstraints.gridx = 0;  
2337: gridBagConstraints.gridy = 0;  
2338: gridBagConstraints.weightx = 1.0;  
2339: jPanel6.add(tl24, gridBagConstraints);  
2340:  
2341: gridBagConstraints = new java.awt.GridBagConstraints();  
2342: gridBagConstraints.gridx = 2;  
2343: gridBagConstraints.gridy = 5;  
2344: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2345: gridBagConstraints.weightx = 1.0;  
2346: panelLeves.add(jPanel6, gridBagConstraints);  
2347:  
2348: jPanel7.setToolTipText("");  
2349: jPanel7.setLayout(new java.awt.GridBagLayout());  
2350:  
2351: tl32.setColumns(5);  
2352: tl32.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2353: tl32.setToolTipText("");  
2354: tl32.setInputVerifier(intVerifier);  
2355: tl32.setMinimumSize(new java.awt.Dimension(50, 19));  
2356: tl32.setName(""); // NOI18N  
2357: tl32.setNextFocusableComponent(tl33);  
2358: gridBagConstraints = new java.awt.GridBagConstraints();  
2359: gridBagConstraints.gridx = 0;  
2360: gridBagConstraints.gridy = 0;  
2361: jPanel7.add(tl32, gridBagConstraints);  
2362:  
2363: gridBagConstraints = new java.awt.GridBagConstraints();  
2364: gridBagConstraints.gridx = 2;  
2365: gridBagConstraints.gridy = 6;  
2366: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2367: gridBagConstraints.weightx = 1.0;  
2368: panelLeves.add(jPanel7, gridBagConstraints);  
2369:  
2370: jPanel8.setToolTipText("");  
2371: jPanel8.setLayout(new java.awt.GridBagLayout());  
2372:  
2373: tl40.setColumns(5);  
2374: tl40.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2375: tl40.setToolTipText("");  
2376: tl40.setInputVerifier(intVerifier);  
2377: tl40.setMinimumSize(new java.awt.Dimension(50, 19));  
2378: tl40.setName(""); // NOI18N  
2379: tl40.setNextFocusableComponent(tl41);  
2380: gridBagConstraints = new java.awt.GridBagConstraints();  
2381: gridBagConstraints.gridx = 0;  
2382: gridBagConstraints.gridy = 0;  
2383: jPanel8.add(tl40, gridBagConstraints);  
2384:  
2385: gridBagConstraints = new java.awt.GridBagConstraints();  
2386: gridBagConstraints.gridx = 2;  
2387: gridBagConstraints.gridy = 7;  
2388: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2389: gridBagConstraints.weightx = 1.0;  
2390: panelLeves.add(jPanel8, gridBagConstraints);  
2391:  
2392: jPanel9.setToolTipText("");  
2393: jPanel9.setLayout(new java.awt.GridBagLayout());  
2394:  
2395: tl48.setColumns(5);  
2396: tl48.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2397: tl48.setToolTipText("");  
2398: tl48.setInputVerifier(intVerifier);  
2399: tl48.setMinimumSize(new java.awt.Dimension(50, 19));  
2400: tl48.setName(""); // NOI18N  
2401: tl48.setNextFocusableComponent(tl49);  
2402: gridBagConstraints = new java.awt.GridBagConstraints();  
2403: gridBagConstraints.gridx = 0;  
2404: gridBagConstraints.gridy = 0;  
2405: jPanel9.add(tl48, gridBagConstraints);  
2406:  
2407: gridBagConstraints = new java.awt.GridBagConstraints();  
2408: gridBagConstraints.gridx = 2;  
2409: gridBagConstraints.gridy = 8;  
2410: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2411: gridBagConstraints.weightx = 1.0;  
2412: panelLeves.add(jPanel9, gridBagConstraints);  
2413:  
2414: jPanel10.setToolTipText("");  
2415: jPanel10.setLayout(new java.awt.GridBagLayout());  
2416:  
2417: tl56.setColumns(5);  
2418: tl56.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2419: tl56.setToolTipText("");  
2420: tl56.setInputVerifier(intVerifier);  
2421: tl56.setMinimumSize(new java.awt.Dimension(50, 19));  
2422: tl56.setName(""); // NOI18N  
2423: tl56.setNextFocusableComponent(tl57);  
2424: gridBagConstraints = new java.awt.GridBagConstraints();  
2425: gridBagConstraints.gridx = 0;  
2426: gridBagConstraints.gridy = 0;  
2427: jPanel10.add(tl56, gridBagConstraints);  
2428:  
2429: gridBagConstraints = new java.awt.GridBagConstraints();  
2430: gridBagConstraints.gridx = 2;  
2431: gridBagConstraints.gridy = 9;  
2432: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2433: gridBagConstraints.weightx = 1.0;  
2434: panelLeves.add(jPanel10, gridBagConstraints);  
2435:  
2436: jPanel11.setToolTipText("");  
2437: jPanel11.setLayout(new java.awt.GridBagLayout());  
2438:  
2439: tl64.setColumns(5);  
2440: tl64.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2441: tl64.setToolTipText("");  
2442: tl64.setInputVerifier(intVerifier);  
2443: tl64.setMinimumSize(new java.awt.Dimension(50, 19));  
2444: tl64.setName(""); // NOI18N  
2445: tl64.setNextFocusableComponent(tl65);  
2446: gridBagConstraints = new java.awt.GridBagConstraints();  
2447: gridBagConstraints.gridx = 0;  
2448: gridBagConstraints.gridy = 0;  
2449: jPanel11.add(tl64, gridBagConstraints);  
2450:  
2451: gridBagConstraints = new java.awt.GridBagConstraints();  
2452: gridBagConstraints.gridx = 2;  
2453: gridBagConstraints.gridy = 10;  
2454: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2455: gridBagConstraints.weightx = 1.0;  
2456: panelLeves.add(jPanel11, gridBagConstraints);  
2457:  
2458: jPanel12.setToolTipText("");  
2459: jPanel12.setLayout(new java.awt.GridBagLayout());  
2460:  
2461: tl72.setColumns(5);  
2462: tl72.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2463: tl72.setToolTipText("");  
2464: tl72.setInputVerifier(intVerifier);  
2465: tl72.setMinimumSize(new java.awt.Dimension(50, 19));  
2466: tl72.setName(""); // NOI18N  
2467: tl72.setNextFocusableComponent(tl73);  
2468: gridBagConstraints = new java.awt.GridBagConstraints();  
2469: gridBagConstraints.gridx = 0;  
2470: gridBagConstraints.gridy = 0;  
2471: jPanel12.add(tl72, gridBagConstraints);  
2472:  
2473: gridBagConstraints = new java.awt.GridBagConstraints();  
2474: gridBagConstraints.gridx = 2;  
2475: gridBagConstraints.gridy = 11;  
2476: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2477: gridBagConstraints.weightx = 1.0;  
2478: panelLeves.add(jPanel12, gridBagConstraints);  
2479:  
2480: jPanel13.setToolTipText("");  
2481: jPanel13.setLayout(new java.awt.GridBagLayout());  
2482:  
2483: tl80.setColumns(5);  
2484: tl80.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2485: tl80.setToolTipText("");  
2486: tl80.setInputVerifier(intVerifier);  
2487: tl80.setMinimumSize(new java.awt.Dimension(50, 19));  
2488: tl80.setName(""); // NOI18N  
2489: tl80.setNextFocusableComponent(tl81);  
2490: gridBagConstraints = new java.awt.GridBagConstraints();  
2491: gridBagConstraints.gridx = 0;  
2492: gridBagConstraints.gridy = 0;  
2493: jPanel13.add(tl80, gridBagConstraints);  
2494:  
2495: gridBagConstraints = new java.awt.GridBagConstraints();  
2496: gridBagConstraints.gridx = 2;  
2497: gridBagConstraints.gridy = 12;  
2498: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2499: gridBagConstraints.weightx = 1.0;  
2500: panelLeves.add(jPanel13, gridBagConstraints);  
2501:  
2502: jPanel14.setToolTipText("");  
2503: jPanel14.setLayout(new java.awt.GridBagLayout());  
2504:  
2505: tl88.setColumns(5);  
2506: tl88.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2507: tl88.setToolTipText("");  
2508: tl88.setInputVerifier(intVerifier);  
2509: tl88.setMinimumSize(new java.awt.Dimension(50, 19));  
2510: tl88.setName(""); // NOI18N  
2511: tl88.setNextFocusableComponent(tl89);  
2512: gridBagConstraints = new java.awt.GridBagConstraints();  
2513: gridBagConstraints.gridx = 0;  
2514: gridBagConstraints.gridy = 0;  
2515: jPanel14.add(tl88, gridBagConstraints);  
2516:  
2517: gridBagConstraints = new java.awt.GridBagConstraints();  
2518: gridBagConstraints.gridx = 2;  
2519: gridBagConstraints.gridy = 13;  
2520: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2521: gridBagConstraints.weightx = 1.0;  
2522: panelLeves.add(jPanel14, gridBagConstraints);  
2523:  
2524: jPanel15.setLayout(new java.awt.GridBagLayout());  
2525:  
2526: tl2.setColumns(5);  
2527: tl2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2528: tl2.setToolTipText("");  
2529: tl2.setInputVerifier(intVerifier);  
2530: tl2.setMinimumSize(new java.awt.Dimension(50, 19));  
2531: tl2.setName(""); // NOI18N  
2532: tl2.setNextFocusableComponent(tl3);  
2533: gridBagConstraints = new java.awt.GridBagConstraints();  
2534: gridBagConstraints.gridx = 0;  
2535: gridBagConstraints.gridy = 0;  
2536: gridBagConstraints.weightx = 1.0;  
2537: jPanel15.add(tl2, gridBagConstraints);  
2538:  
2539: gridBagConstraints = new java.awt.GridBagConstraints();  
2540: gridBagConstraints.gridx = 4;  
2541: gridBagConstraints.gridy = 2;  
2542: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2543: gridBagConstraints.weightx = 1.0;  
2544: panelLeves.add(jPanel15, gridBagConstraints);  
2545:  
2546: jPanel16.setLayout(new java.awt.GridBagLayout());  
2547:  
2548: tl10.setColumns(5);  
2549: tl10.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2550: tl10.setToolTipText("");  
2551: tl10.setInputVerifier(intVerifier);  
2552: tl10.setMinimumSize(new java.awt.Dimension(50, 19));  
2553: tl10.setName(""); // NOI18N  
2554: tl10.setNextFocusableComponent(tl11);  
2555: gridBagConstraints = new java.awt.GridBagConstraints();  
2556: gridBagConstraints.gridx = 0;  
2557: gridBagConstraints.gridy = 0;  
2558: gridBagConstraints.weightx = 1.0;  
2559: jPanel16.add(tl10, gridBagConstraints);  
2560:  
2561: gridBagConstraints = new java.awt.GridBagConstraints();  
2562: gridBagConstraints.gridx = 4;  
2563: gridBagConstraints.gridy = 3;  
2564: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2565: gridBagConstraints.weightx = 1.0;  
2566: panelLeves.add(jPanel16, gridBagConstraints);  
2567:  
2568: jPanel17.setLayout(new java.awt.GridBagLayout());  
2569:  
2570: tl18.setColumns(5);  
2571: tl18.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2572: tl18.setToolTipText("");  
2573: tl18.setInputVerifier(intVerifier);  
2574: tl18.setMinimumSize(new java.awt.Dimension(50, 19));  
2575: tl18.setName(""); // NOI18N  
2576: tl18.setNextFocusableComponent(tl19);  
2577: gridBagConstraints = new java.awt.GridBagConstraints();  
2578: gridBagConstraints.gridx = 0;  
2579: gridBagConstraints.gridy = 0;  
2580: gridBagConstraints.weightx = 1.0;  
2581: jPanel17.add(tl18, gridBagConstraints);  
2582:  
2583: gridBagConstraints = new java.awt.GridBagConstraints();  
2584: gridBagConstraints.gridx = 4;  
2585: gridBagConstraints.gridy = 4;  
2586: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2587: gridBagConstraints.weightx = 1.0;  
2588: panelLeves.add(jPanel17, gridBagConstraints);  
2589:  
2590: jPanel18.setLayout(new java.awt.GridBagLayout());  
2591:  
2592: tl26.setColumns(5);  
2593: tl26.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2594: tl26.setToolTipText("");  
2595: tl26.setInputVerifier(intVerifier);  
2596: tl26.setMinimumSize(new java.awt.Dimension(50, 19));  
2597: tl26.setName(""); // NOI18N  
2598: tl26.setNextFocusableComponent(tl27);  
2599: gridBagConstraints = new java.awt.GridBagConstraints();  
2600: gridBagConstraints.gridx = 0;  
2601: gridBagConstraints.gridy = 0;  
2602: gridBagConstraints.weightx = 1.0;  
2603: jPanel18.add(tl26, gridBagConstraints);  
2604:  
2605: gridBagConstraints = new java.awt.GridBagConstraints();  
2606: gridBagConstraints.gridx = 4;  
2607: gridBagConstraints.gridy = 5;  
2608: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2609: gridBagConstraints.weightx = 1.0;  
2610: panelLeves.add(jPanel18, gridBagConstraints);  
2611:  
2612: jPanel19.setLayout(new java.awt.GridBagLayout());  
2613:  
2614: tl34.setColumns(5);  
2615: tl34.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2616: tl34.setToolTipText("");  
2617: tl34.setInputVerifier(intVerifier);  
2618: tl34.setMinimumSize(new java.awt.Dimension(50, 19));  
2619: tl34.setName(""); // NOI18N  
2620: tl34.setNextFocusableComponent(tl35);  
2621: gridBagConstraints = new java.awt.GridBagConstraints();  
2622: gridBagConstraints.gridx = 0;  
2623: gridBagConstraints.gridy = 0;  
2624: jPanel19.add(tl34, gridBagConstraints);  
2625:  
2626: gridBagConstraints = new java.awt.GridBagConstraints();  
2627: gridBagConstraints.gridx = 4;  
2628: gridBagConstraints.gridy = 6;  
2629: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2630: gridBagConstraints.weightx = 1.0;  
2631: panelLeves.add(jPanel19, gridBagConstraints);  
2632:  
2633: jPanel20.setLayout(new java.awt.GridBagLayout());  
2634:  
2635: tl42.setColumns(5);  
2636: tl42.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2637: tl42.setToolTipText("");  
2638: tl42.setInputVerifier(intVerifier);  
2639: tl42.setMinimumSize(new java.awt.Dimension(50, 19));  
2640: tl42.setName(""); // NOI18N  
2641: tl42.setNextFocusableComponent(tl43);  
2642: gridBagConstraints = new java.awt.GridBagConstraints();  
2643: gridBagConstraints.gridx = 0;  
2644: gridBagConstraints.gridy = 0;  
2645: jPanel20.add(tl42, gridBagConstraints);  
2646:  
2647: gridBagConstraints = new java.awt.GridBagConstraints();  
2648: gridBagConstraints.gridx = 4;  
2649: gridBagConstraints.gridy = 7;  
2650: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2651: gridBagConstraints.weightx = 1.0;  
2652: panelLeves.add(jPanel20, gridBagConstraints);  
2653:  
2654: jPanel21.setLayout(new java.awt.GridBagLayout());  
2655:  
2656: tl50.setColumns(5);  
2657: tl50.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2658: tl50.setToolTipText("");  
2659: tl50.setInputVerifier(intVerifier);  
2660: tl50.setMinimumSize(new java.awt.Dimension(50, 19));  
2661: tl50.setName(""); // NOI18N  
2662: tl50.setNextFocusableComponent(tl51);  
2663: gridBagConstraints = new java.awt.GridBagConstraints();  
2664: gridBagConstraints.gridx = 0;  
2665: gridBagConstraints.gridy = 0;  
2666: jPanel21.add(tl50, gridBagConstraints);  
2667:  
2668: gridBagConstraints = new java.awt.GridBagConstraints();  
2669: gridBagConstraints.gridx = 4;  
2670: gridBagConstraints.gridy = 8;  
2671: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2672: gridBagConstraints.weightx = 1.0;  
2673: panelLeves.add(jPanel21, gridBagConstraints);  
2674:  
2675: jPanel22.setLayout(new java.awt.GridBagLayout());  
2676:  
2677: tl58.setColumns(5);  
2678: tl58.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2679: tl58.setToolTipText("");  
2680: tl58.setInputVerifier(intVerifier);  
2681: tl58.setMinimumSize(new java.awt.Dimension(50, 19));  
2682: tl58.setName(""); // NOI18N  
2683: tl58.setNextFocusableComponent(tl59);  
2684: gridBagConstraints = new java.awt.GridBagConstraints();  
2685: gridBagConstraints.gridx = 0;  
2686: gridBagConstraints.gridy = 0;  
2687: jPanel22.add(tl58, gridBagConstraints);  
2688:  
2689: gridBagConstraints = new java.awt.GridBagConstraints();  
2690: gridBagConstraints.gridx = 4;  
2691: gridBagConstraints.gridy = 9;  
2692: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2693: gridBagConstraints.weightx = 1.0;  
2694: panelLeves.add(jPanel22, gridBagConstraints);  
2695:  
2696: jPanel23.setLayout(new java.awt.GridBagLayout());  
2697:  
2698: tl66.setColumns(5);  
2699: tl66.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2700: tl66.setToolTipText("");  
2701: tl66.setInputVerifier(intVerifier);  
2702: tl66.setMinimumSize(new java.awt.Dimension(50, 19));  
2703: tl66.setName(""); // NOI18N  
2704: tl66.setNextFocusableComponent(tl67);  
2705: gridBagConstraints = new java.awt.GridBagConstraints();  
2706: gridBagConstraints.gridx = 0;  
2707: gridBagConstraints.gridy = 0;  
2708: jPanel23.add(tl66, gridBagConstraints);  
2709:  
2710: gridBagConstraints = new java.awt.GridBagConstraints();  
2711: gridBagConstraints.gridx = 4;  
2712: gridBagConstraints.gridy = 10;  
2713: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2714: gridBagConstraints.weightx = 1.0;  
2715: panelLeves.add(jPanel23, gridBagConstraints);  
2716:  
2717: jPanel24.setLayout(new java.awt.GridBagLayout());  
2718:  
2719: tl74.setColumns(5);  
2720: tl74.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2721: tl74.setToolTipText("");  
2722: tl74.setInputVerifier(intVerifier);  
2723: tl74.setMinimumSize(new java.awt.Dimension(50, 19));  
2724: tl74.setName(""); // NOI18N  
2725: tl74.setNextFocusableComponent(tl75);  
2726: gridBagConstraints = new java.awt.GridBagConstraints();  
2727: gridBagConstraints.gridx = 0;  
2728: gridBagConstraints.gridy = 0;  
2729: jPanel24.add(tl74, gridBagConstraints);  
2730:  
2731: gridBagConstraints = new java.awt.GridBagConstraints();  
2732: gridBagConstraints.gridx = 4;  
2733: gridBagConstraints.gridy = 11;  
2734: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2735: gridBagConstraints.weightx = 1.0;  
2736: panelLeves.add(jPanel24, gridBagConstraints);  
2737:  
2738: jPanel25.setLayout(new java.awt.GridBagLayout());  
2739:  
2740: tl82.setColumns(5);  
2741: tl82.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2742: tl82.setToolTipText("");  
2743: tl82.setInputVerifier(intVerifier);  
2744: tl82.setMinimumSize(new java.awt.Dimension(50, 19));  
2745: tl82.setName(""); // NOI18N  
2746: tl82.setNextFocusableComponent(tl83);  
2747: gridBagConstraints = new java.awt.GridBagConstraints();  
2748: gridBagConstraints.gridx = 0;  
2749: gridBagConstraints.gridy = 0;  
2750: jPanel25.add(tl82, gridBagConstraints);  
2751:  
2752: gridBagConstraints = new java.awt.GridBagConstraints();  
2753: gridBagConstraints.gridx = 4;  
2754: gridBagConstraints.gridy = 12;  
2755: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2756: gridBagConstraints.weightx = 1.0;  
2757: panelLeves.add(jPanel25, gridBagConstraints);  
2758:  
2759: jPanel26.setLayout(new java.awt.GridBagLayout());  
2760:  
2761: tl90.setColumns(5);  
2762: tl90.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2763: tl90.setToolTipText("");  
2764: tl90.setInputVerifier(intVerifier);  
2765: tl90.setMinimumSize(new java.awt.Dimension(50, 19));  
2766: tl90.setName(""); // NOI18N  
2767: tl90.setNextFocusableComponent(tl91);  
2768: gridBagConstraints = new java.awt.GridBagConstraints();  
2769: gridBagConstraints.gridx = 0;  
2770: gridBagConstraints.gridy = 0;  
2771: jPanel26.add(tl90, gridBagConstraints);  
2772:  
2773: gridBagConstraints = new java.awt.GridBagConstraints();  
2774: gridBagConstraints.gridx = 4;  
2775: gridBagConstraints.gridy = 13;  
2776: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2777: gridBagConstraints.weightx = 1.0;  
2778: panelLeves.add(jPanel26, gridBagConstraints);  
2779:  
2780: jPanel27.setLayout(new java.awt.GridBagLayout());  
2781:  
2782: tl4.setColumns(5);  
2783: tl4.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2784: tl4.setToolTipText("");  
2785: tl4.setInputVerifier(intVerifier);  
2786: tl4.setMinimumSize(new java.awt.Dimension(50, 19));  
2787: tl4.setName(""); // NOI18N  
2788: tl4.setNextFocusableComponent(tl5);  
2789: gridBagConstraints = new java.awt.GridBagConstraints();  
2790: gridBagConstraints.gridx = 0;  
2791: gridBagConstraints.gridy = 0;  
2792: gridBagConstraints.weightx = 1.0;  
2793: jPanel27.add(tl4, gridBagConstraints);  
2794:  
2795: gridBagConstraints = new java.awt.GridBagConstraints();  
2796: gridBagConstraints.gridx = 6;  
2797: gridBagConstraints.gridy = 2;  
2798: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2799: gridBagConstraints.weightx = 1.0;  
2800: panelLeves.add(jPanel27, gridBagConstraints);  
2801:  
2802: jPanel28.setLayout(new java.awt.GridBagLayout());  
2803:  
2804: tl12.setColumns(5);  
2805: tl12.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2806: tl12.setToolTipText("");  
2807: tl12.setInputVerifier(intVerifier);  
2808: tl12.setMinimumSize(new java.awt.Dimension(50, 19));  
2809: tl12.setName(""); // NOI18N  
2810: tl12.setNextFocusableComponent(tl13);  
2811: gridBagConstraints = new java.awt.GridBagConstraints();  
2812: gridBagConstraints.gridx = 0;  
2813: gridBagConstraints.gridy = 0;  
2814: gridBagConstraints.weightx = 1.0;  
2815: jPanel28.add(tl12, gridBagConstraints);  
2816:  
2817: gridBagConstraints = new java.awt.GridBagConstraints();  
2818: gridBagConstraints.gridx = 6;  
2819: gridBagConstraints.gridy = 3;  
2820: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2821: gridBagConstraints.weightx = 1.0;  
2822: panelLeves.add(jPanel28, gridBagConstraints);  
2823:  
2824: jPanel29.setLayout(new java.awt.GridBagLayout());  
2825:  
2826: tl20.setColumns(5);  
2827: tl20.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2828: tl20.setToolTipText("");  
2829: tl20.setInputVerifier(intVerifier);  
2830: tl20.setMinimumSize(new java.awt.Dimension(50, 19));  
2831: tl20.setName(""); // NOI18N  
2832: tl20.setNextFocusableComponent(tl21);  
2833: gridBagConstraints = new java.awt.GridBagConstraints();  
2834: gridBagConstraints.gridx = 0;  
2835: gridBagConstraints.gridy = 0;  
2836: gridBagConstraints.weightx = 1.0;  
2837: jPanel29.add(tl20, gridBagConstraints);  
2838:  
2839: gridBagConstraints = new java.awt.GridBagConstraints();  
2840: gridBagConstraints.gridx = 6;  
2841: gridBagConstraints.gridy = 4;  
2842: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2843: gridBagConstraints.weightx = 1.0;  
2844: panelLeves.add(jPanel29, gridBagConstraints);  
2845:  
2846: jPanel30.setLayout(new java.awt.GridBagLayout());  
2847:  
2848: tl28.setColumns(5);  
2849: tl28.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2850: tl28.setToolTipText("");  
2851: tl28.setInputVerifier(intVerifier);  
2852: tl28.setMinimumSize(new java.awt.Dimension(50, 19));  
2853: tl28.setName(""); // NOI18N  
2854: tl28.setNextFocusableComponent(tl29);  
2855: gridBagConstraints = new java.awt.GridBagConstraints();  
2856: gridBagConstraints.gridx = 0;  
2857: gridBagConstraints.gridy = 0;  
2858: gridBagConstraints.weightx = 1.0;  
2859: jPanel30.add(tl28, gridBagConstraints);  
2860:  
2861: gridBagConstraints = new java.awt.GridBagConstraints();  
2862: gridBagConstraints.gridx = 6;  
2863: gridBagConstraints.gridy = 5;  
2864: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2865: gridBagConstraints.weightx = 1.0;  
2866: panelLeves.add(jPanel30, gridBagConstraints);  
2867:  
2868: jPanel31.setLayout(new java.awt.GridBagLayout());  
2869:  
2870: tl36.setColumns(5);  
2871: tl36.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2872: tl36.setToolTipText("");  
2873: tl36.setInputVerifier(intVerifier);  
2874: tl36.setMinimumSize(new java.awt.Dimension(50, 19));  
2875: tl36.setName(""); // NOI18N  
2876: tl36.setNextFocusableComponent(tl37);  
2877: gridBagConstraints = new java.awt.GridBagConstraints();  
2878: gridBagConstraints.gridx = 0;  
2879: gridBagConstraints.gridy = 0;  
2880: jPanel31.add(tl36, gridBagConstraints);  
2881:  
2882: gridBagConstraints = new java.awt.GridBagConstraints();  
2883: gridBagConstraints.gridx = 6;  
2884: gridBagConstraints.gridy = 6;  
2885: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2886: gridBagConstraints.weightx = 1.0;  
2887: panelLeves.add(jPanel31, gridBagConstraints);  
2888:  
2889: jPanel32.setLayout(new java.awt.GridBagLayout());  
2890:  
2891: tl44.setColumns(5);  
2892: tl44.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2893: tl44.setToolTipText("");  
2894: tl44.setInputVerifier(intVerifier);  
2895: tl44.setMinimumSize(new java.awt.Dimension(50, 19));  
2896: tl44.setName(""); // NOI18N  
2897: tl44.setNextFocusableComponent(tl45);  
2898: gridBagConstraints = new java.awt.GridBagConstraints();  
2899: gridBagConstraints.gridx = 0;  
2900: gridBagConstraints.gridy = 0;  
2901: jPanel32.add(tl44, gridBagConstraints);  
2902:  
2903: gridBagConstraints = new java.awt.GridBagConstraints();  
2904: gridBagConstraints.gridx = 6;  
2905: gridBagConstraints.gridy = 7;  
2906: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2907: gridBagConstraints.weightx = 1.0;  
2908: panelLeves.add(jPanel32, gridBagConstraints);  
2909:  
2910: jPanel33.setLayout(new java.awt.GridBagLayout());  
2911:  
2912: tl52.setColumns(5);  
2913: tl52.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2914: tl52.setToolTipText("");  
2915: tl52.setInputVerifier(intVerifier);  
2916: tl52.setMinimumSize(new java.awt.Dimension(50, 19));  
2917: tl52.setName(""); // NOI18N  
2918: tl52.setNextFocusableComponent(tl53);  
2919: gridBagConstraints = new java.awt.GridBagConstraints();  
2920: gridBagConstraints.gridx = 0;  
2921: gridBagConstraints.gridy = 0;  
2922: jPanel33.add(tl52, gridBagConstraints);  
2923:  
2924: gridBagConstraints = new java.awt.GridBagConstraints();  
2925: gridBagConstraints.gridx = 6;  
2926: gridBagConstraints.gridy = 8;  
2927: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2928: gridBagConstraints.weightx = 1.0;  
2929: panelLeves.add(jPanel33, gridBagConstraints);  
2930:  
2931: jPanel34.setLayout(new java.awt.GridBagLayout());  
2932:  
2933: tl60.setColumns(5);  
2934: tl60.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2935: tl60.setToolTipText("");  
2936: tl60.setInputVerifier(intVerifier);  
2937: tl60.setMinimumSize(new java.awt.Dimension(50, 19));  
2938: tl60.setName(""); // NOI18N  
2939: tl60.setNextFocusableComponent(tl61);  
2940: gridBagConstraints = new java.awt.GridBagConstraints();  
2941: gridBagConstraints.gridx = 0;  
2942: gridBagConstraints.gridy = 0;  
2943: jPanel34.add(tl60, gridBagConstraints);  
2944:  
2945: gridBagConstraints = new java.awt.GridBagConstraints();  
2946: gridBagConstraints.gridx = 6;  
2947: gridBagConstraints.gridy = 9;  
2948: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2949: gridBagConstraints.weightx = 1.0;  
2950: panelLeves.add(jPanel34, gridBagConstraints);  
2951:  
2952: jPanel35.setLayout(new java.awt.GridBagLayout());  
2953:  
2954: tl68.setColumns(5);  
2955: tl68.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2956: tl68.setToolTipText("");  
2957: tl68.setInputVerifier(intVerifier);  
2958: tl68.setMinimumSize(new java.awt.Dimension(50, 19));  
2959: tl68.setName(""); // NOI18N  
2960: tl68.setNextFocusableComponent(tl69);  
2961: gridBagConstraints = new java.awt.GridBagConstraints();  
2962: gridBagConstraints.gridx = 0;  
2963: gridBagConstraints.gridy = 0;  
2964: jPanel35.add(tl68, gridBagConstraints);  
2965:  
2966: gridBagConstraints = new java.awt.GridBagConstraints();  
2967: gridBagConstraints.gridx = 6;  
2968: gridBagConstraints.gridy = 10;  
2969: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2970: gridBagConstraints.weightx = 1.0;  
2971: panelLeves.add(jPanel35, gridBagConstraints);  
2972:  
2973: jPanel36.setLayout(new java.awt.GridBagLayout());  
2974:  
2975: tl76.setColumns(5);  
2976: tl76.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2977: tl76.setToolTipText("");  
2978: tl76.setInputVerifier(intVerifier);  
2979: tl76.setMinimumSize(new java.awt.Dimension(50, 19));  
2980: tl76.setName(""); // NOI18N  
2981: tl76.setNextFocusableComponent(tl77);  
2982: gridBagConstraints = new java.awt.GridBagConstraints();  
2983: gridBagConstraints.gridx = 0;  
2984: gridBagConstraints.gridy = 0;  
2985: jPanel36.add(tl76, gridBagConstraints);  
2986:  
2987: gridBagConstraints = new java.awt.GridBagConstraints();  
2988: gridBagConstraints.gridx = 6;  
2989: gridBagConstraints.gridy = 11;  
2990: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
2991: gridBagConstraints.weightx = 1.0;  
2992: panelLeves.add(jPanel36, gridBagConstraints);  
2993:  
2994: jPanel37.setLayout(new java.awt.GridBagLayout());  
2995:  
2996: tl84.setColumns(5);  
2997: tl84.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
2998: tl84.setToolTipText("");  
2999: tl84.setInputVerifier(intVerifier);  
3000: tl84.setMinimumSize(new java.awt.Dimension(50, 19));  
3001: tl84.setName(""); // NOI18N  
3002: tl84.setNextFocusableComponent(tl85);  
3003: gridBagConstraints = new java.awt.GridBagConstraints();  
3004: gridBagConstraints.gridx = 0;  
3005: gridBagConstraints.gridy = 0;  
3006: jPanel37.add(tl84, gridBagConstraints);  
3007:  
3008: gridBagConstraints = new java.awt.GridBagConstraints();  
3009: gridBagConstraints.gridx = 6;  
3010: gridBagConstraints.gridy = 12;  
3011: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3012: gridBagConstraints.weightx = 1.0;  
3013: panelLeves.add(jPanel37, gridBagConstraints);  
3014:  
3015: jPanel38.setLayout(new java.awt.GridBagLayout());  
3016:  
3017: tl92.setColumns(5);  
3018: tl92.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3019: tl92.setToolTipText("");  
3020: tl92.setInputVerifier(intVerifier);  
3021: tl92.setMinimumSize(new java.awt.Dimension(50, 19));  
3022: tl92.setName(""); // NOI18N  
3023: tl92.setNextFocusableComponent(tl93);  
3024: gridBagConstraints = new java.awt.GridBagConstraints();  
3025: gridBagConstraints.gridx = 0;  
3026: gridBagConstraints.gridy = 0;  
3027: jPanel38.add(tl92, gridBagConstraints);  
3028:  
3029: gridBagConstraints = new java.awt.GridBagConstraints();  
3030: gridBagConstraints.gridx = 6;  
3031: gridBagConstraints.gridy = 13;  
3032: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3033: gridBagConstraints.weightx = 1.0;  
3034: panelLeves.add(jPanel38, gridBagConstraints);  
3035:  
3036: jPanel39.setLayout(new java.awt.GridBagLayout());  
3037:  
3038: tl6.setColumns(5);  
3039: tl6.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3040: tl6.setToolTipText("");  
3041: tl6.setInputVerifier(intVerifier);  
3042: tl6.setMinimumSize(new java.awt.Dimension(50, 19));  
3043: tl6.setName(""); // NOI18N  
3044: tl6.setNextFocusableComponent(tl7);  
3045: gridBagConstraints = new java.awt.GridBagConstraints();  
3046: gridBagConstraints.gridx = 0;  
3047: gridBagConstraints.gridy = 0;  
3048: gridBagConstraints.weightx = 1.0;  
3049: jPanel39.add(tl6, gridBagConstraints);  
3050:  
3051: gridBagConstraints = new java.awt.GridBagConstraints();  
3052: gridBagConstraints.gridx = 8;  
3053: gridBagConstraints.gridy = 2;  
3054: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3055: gridBagConstraints.weightx = 1.0;  
3056: panelLeves.add(jPanel39, gridBagConstraints);  
3057:  
3058: jPanel40.setLayout(new java.awt.GridBagLayout());  
3059:  
3060: tl14.setColumns(5);  
3061: tl14.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3062: tl14.setToolTipText("");  
3063: tl14.setInputVerifier(intVerifier);  
3064: tl14.setMinimumSize(new java.awt.Dimension(50, 19));  
3065: tl14.setName(""); // NOI18N  
3066: tl14.setNextFocusableComponent(tl15);  
3067: gridBagConstraints = new java.awt.GridBagConstraints();  
3068: gridBagConstraints.gridx = 0;  
3069: gridBagConstraints.gridy = 0;  
3070: gridBagConstraints.weightx = 1.0;  
3071: jPanel40.add(tl14, gridBagConstraints);  
3072:  
3073: gridBagConstraints = new java.awt.GridBagConstraints();  
3074: gridBagConstraints.gridx = 8;  
3075: gridBagConstraints.gridy = 3;  
3076: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3077: gridBagConstraints.weightx = 1.0;  
3078: panelLeves.add(jPanel40, gridBagConstraints);  
3079:  
3080: jPanel41.setLayout(new java.awt.GridBagLayout());  
3081:  
3082: tl22.setColumns(5);  
3083: tl22.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3084: tl22.setToolTipText("");  
3085: tl22.setInputVerifier(intVerifier);  
3086: tl22.setMinimumSize(new java.awt.Dimension(50, 19));  
3087: tl22.setName(""); // NOI18N  
3088: tl22.setNextFocusableComponent(tl23);  
3089: gridBagConstraints = new java.awt.GridBagConstraints();  
3090: gridBagConstraints.gridx = 0;  
3091: gridBagConstraints.gridy = 0;  
3092: gridBagConstraints.weightx = 1.0;  
3093: jPanel41.add(tl22, gridBagConstraints);  
3094:  
3095: gridBagConstraints = new java.awt.GridBagConstraints();  
3096: gridBagConstraints.gridx = 8;  
3097: gridBagConstraints.gridy = 4;  
3098: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3099: gridBagConstraints.weightx = 1.0;  
3100: panelLeves.add(jPanel41, gridBagConstraints);  
3101:  
3102: jPanel42.setLayout(new java.awt.GridBagLayout());  
3103:  
3104: tl30.setColumns(5);  
3105: tl30.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3106: tl30.setToolTipText("");  
3107: tl30.setInputVerifier(intVerifier);  
3108: tl30.setMinimumSize(new java.awt.Dimension(50, 19));  
3109: tl30.setName(""); // NOI18N  
3110: tl30.setNextFocusableComponent(tl31);  
3111: gridBagConstraints = new java.awt.GridBagConstraints();  
3112: gridBagConstraints.gridx = 0;  
3113: gridBagConstraints.gridy = 0;  
3114: gridBagConstraints.weightx = 1.0;  
3115: jPanel42.add(tl30, gridBagConstraints);  
3116:  
3117: gridBagConstraints = new java.awt.GridBagConstraints();  
3118: gridBagConstraints.gridx = 8;  
3119: gridBagConstraints.gridy = 5;  
3120: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3121: gridBagConstraints.weightx = 1.0;  
3122: panelLeves.add(jPanel42, gridBagConstraints);  
3123:  
3124: jPanel43.setLayout(new java.awt.GridBagLayout());  
3125:  
3126: tl38.setColumns(5);  
3127: tl38.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3128: tl38.setToolTipText("");  
3129: tl38.setInputVerifier(intVerifier);  
3130: tl38.setMinimumSize(new java.awt.Dimension(50, 19));  
3131: tl38.setName(""); // NOI18N  
3132: tl38.setNextFocusableComponent(tl39);  
3133: gridBagConstraints = new java.awt.GridBagConstraints();  
3134: gridBagConstraints.gridx = 0;  
3135: gridBagConstraints.gridy = 0;  
3136: jPanel43.add(tl38, gridBagConstraints);  
3137:  
3138: gridBagConstraints = new java.awt.GridBagConstraints();  
3139: gridBagConstraints.gridx = 8;  
3140: gridBagConstraints.gridy = 6;  
3141: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3142: gridBagConstraints.weightx = 1.0;  
3143: panelLeves.add(jPanel43, gridBagConstraints);  
3144:  
3145: jPanel44.setLayout(new java.awt.GridBagLayout());  
3146:  
3147: tl46.setColumns(5);  
3148: tl46.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3149: tl46.setToolTipText("");  
3150: tl46.setInputVerifier(intVerifier);  
3151: tl46.setMinimumSize(new java.awt.Dimension(50, 19));  
3152: tl46.setName(""); // NOI18N  
3153: tl46.setNextFocusableComponent(tl47);  
3154: gridBagConstraints = new java.awt.GridBagConstraints();  
3155: gridBagConstraints.gridx = 0;  
3156: gridBagConstraints.gridy = 0;  
3157: jPanel44.add(tl46, gridBagConstraints);  
3158:  
3159: gridBagConstraints = new java.awt.GridBagConstraints();  
3160: gridBagConstraints.gridx = 8;  
3161: gridBagConstraints.gridy = 7;  
3162: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3163: gridBagConstraints.weightx = 1.0;  
3164: panelLeves.add(jPanel44, gridBagConstraints);  
3165:  
3166: jPanel45.setLayout(new java.awt.GridBagLayout());  
3167:  
3168: tl54.setColumns(5);  
3169: tl54.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3170: tl54.setToolTipText("");  
3171: tl54.setInputVerifier(intVerifier);  
3172: tl54.setMinimumSize(new java.awt.Dimension(50, 19));  
3173: tl54.setName(""); // NOI18N  
3174: tl54.setNextFocusableComponent(tl55);  
3175: gridBagConstraints = new java.awt.GridBagConstraints();  
3176: gridBagConstraints.gridx = 0;  
3177: gridBagConstraints.gridy = 0;  
3178: jPanel45.add(tl54, gridBagConstraints);  
3179:  
3180: gridBagConstraints = new java.awt.GridBagConstraints();  
3181: gridBagConstraints.gridx = 8;  
3182: gridBagConstraints.gridy = 8;  
3183: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3184: gridBagConstraints.weightx = 1.0;  
3185: panelLeves.add(jPanel45, gridBagConstraints);  
3186:  
3187: jPanel46.setLayout(new java.awt.GridBagLayout());  
3188:  
3189: tl62.setColumns(5);  
3190: tl62.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3191: tl62.setToolTipText("");  
3192: tl62.setInputVerifier(intVerifier);  
3193: tl62.setMinimumSize(new java.awt.Dimension(50, 19));  
3194: tl62.setName(""); // NOI18N  
3195: tl62.setNextFocusableComponent(tl63);  
3196: gridBagConstraints = new java.awt.GridBagConstraints();  
3197: gridBagConstraints.gridx = 0;  
3198: gridBagConstraints.gridy = 0;  
3199: jPanel46.add(tl62, gridBagConstraints);  
3200:  
3201: gridBagConstraints = new java.awt.GridBagConstraints();  
3202: gridBagConstraints.gridx = 8;  
3203: gridBagConstraints.gridy = 9;  
3204: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3205: gridBagConstraints.weightx = 1.0;  
3206: panelLeves.add(jPanel46, gridBagConstraints);  
3207:  
3208: jPanel47.setLayout(new java.awt.GridBagLayout());  
3209:  
3210: tl70.setColumns(5);  
3211: tl70.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3212: tl70.setToolTipText("");  
3213: tl70.setInputVerifier(intVerifier);  
3214: tl70.setMinimumSize(new java.awt.Dimension(50, 19));  
3215: tl70.setName(""); // NOI18N  
3216: tl70.setNextFocusableComponent(tl71);  
3217: gridBagConstraints = new java.awt.GridBagConstraints();  
3218: gridBagConstraints.gridx = 0;  
3219: gridBagConstraints.gridy = 0;  
3220: jPanel47.add(tl70, gridBagConstraints);  
3221:  
3222: gridBagConstraints = new java.awt.GridBagConstraints();  
3223: gridBagConstraints.gridx = 8;  
3224: gridBagConstraints.gridy = 10;  
3225: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3226: gridBagConstraints.weightx = 1.0;  
3227: panelLeves.add(jPanel47, gridBagConstraints);  
3228:  
3229: jPanel48.setLayout(new java.awt.GridBagLayout());  
3230:  
3231: tl78.setColumns(5);  
3232: tl78.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3233: tl78.setToolTipText("");  
3234: tl78.setInputVerifier(intVerifier);  
3235: tl78.setMinimumSize(new java.awt.Dimension(50, 19));  
3236: tl78.setName(""); // NOI18N  
3237: tl78.setNextFocusableComponent(tl79);  
3238: gridBagConstraints = new java.awt.GridBagConstraints();  
3239: gridBagConstraints.gridx = 0;  
3240: gridBagConstraints.gridy = 0;  
3241: jPanel48.add(tl78, gridBagConstraints);  
3242:  
3243: gridBagConstraints = new java.awt.GridBagConstraints();  
3244: gridBagConstraints.gridx = 8;  
3245: gridBagConstraints.gridy = 11;  
3246: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3247: gridBagConstraints.weightx = 1.0;  
3248: panelLeves.add(jPanel48, gridBagConstraints);  
3249:  
3250: jPanel49.setLayout(new java.awt.GridBagLayout());  
3251:  
3252: tl86.setColumns(5);  
3253: tl86.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3254: tl86.setToolTipText("");  
3255: tl86.setInputVerifier(intVerifier);  
3256: tl86.setMinimumSize(new java.awt.Dimension(50, 19));  
3257: tl86.setName(""); // NOI18N  
3258: tl86.setNextFocusableComponent(tl87);  
3259: gridBagConstraints = new java.awt.GridBagConstraints();  
3260: gridBagConstraints.gridx = 0;  
3261: gridBagConstraints.gridy = 0;  
3262: jPanel49.add(tl86, gridBagConstraints);  
3263:  
3264: gridBagConstraints = new java.awt.GridBagConstraints();  
3265: gridBagConstraints.gridx = 8;  
3266: gridBagConstraints.gridy = 12;  
3267: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3268: gridBagConstraints.weightx = 1.0;  
3269: panelLeves.add(jPanel49, gridBagConstraints);  
3270:  
3271: jPanel50.setLayout(new java.awt.GridBagLayout());  
3272:  
3273: tl94.setColumns(5);  
3274: tl94.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3275: tl94.setToolTipText("");  
3276: tl94.setInputVerifier(intVerifier);  
3277: tl94.setMinimumSize(new java.awt.Dimension(50, 19));  
3278: tl94.setName(""); // NOI18N  
3279: tl94.setNextFocusableComponent(tl95);  
3280: gridBagConstraints = new java.awt.GridBagConstraints();  
3281: gridBagConstraints.gridx = 0;  
3282: gridBagConstraints.gridy = 0;  
3283: jPanel50.add(tl94, gridBagConstraints);  
3284:  
3285: gridBagConstraints = new java.awt.GridBagConstraints();  
3286: gridBagConstraints.gridx = 8;  
3287: gridBagConstraints.gridy = 13;  
3288: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
3289: gridBagConstraints.weightx = 1.0;  
3290: panelLeves.add(jPanel50, gridBagConstraints);  
3291:  
3292: javax.swing.GroupLayout jPanel119Layout = new javax.swing.GroupLayout(jPanel119);  
3293: jPanel119.setLayout(jPanel119Layout);  
3294: jPanel119Layout.setHorizontalGroup(  
3295: jPanel119Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
3296: .addComponent(panelLeves, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
3297: );  
3298: jPanel119Layout.setVerticalGroup(  
3299: jPanel119Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
3300: .addGroup(jPanel119Layout.createSequentialGroup()  
3301: .addComponent(panelLeves, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  
3302: .addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))  
3303: );  
3304:  
3305: jScrollPane3.setViewportView(jPanel119);  
3306:  
3307: txtTotalLeves.setEditable(false);  
3308: txtTotalLeves.setText("0");  
3309:  
3310: jLabel36.setText("Total:");  
3311:  
3312: lblFolha1.setText("-");  
3313:  
3314: jLabel39.setText("Folha:");  
3315:  
3316: javax.swing.GroupLayout tab\_levesLayout = new javax.swing.GroupLayout(tab\_leves);  
3317: tab\_leves.setLayout(tab\_levesLayout);  
3318: tab\_levesLayout.setHorizontalGroup(  
3319: tab\_levesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
3320: .addGroup(tab\_levesLayout.createSequentialGroup()  
3321: .addContainerGap()  
3322: .addComponent(lblPesquisador)  
3323: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
3324: .addComponent(txtPesquisador1, javax.swing.GroupLayout.PREFERRED\_SIZE, 234, javax.swing.GroupLayout.PREFERRED\_SIZE)  
3325: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
3326: .addComponent(jLabel39)  
3327: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
3328: .addComponent(lblFolha1)  
3329: .addGap(18, 18, 18)  
3330: .addComponent(jLabel36)  
3331: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
3332: .addComponent(txtTotalLeves, javax.swing.GroupLayout.PREFERRED\_SIZE, 64, javax.swing.GroupLayout.PREFERRED\_SIZE)  
3333: .addContainerGap())  
3334: .addComponent(jScrollPane3, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, 2914, Short.MAX\_VALUE)  
3335: );  
3336: tab\_levesLayout.setVerticalGroup(  
3337: tab\_levesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
3338: .addGroup(tab\_levesLayout.createSequentialGroup()  
3339: .addGap(2, 2, 2)  
3340: .addGroup(tab\_levesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
3341: .addComponent(lblPesquisador)  
3342: .addComponent(txtPesquisador1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  
3343: .addComponent(txtTotalLeves, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  
3344: .addComponent(jLabel36)  
3345: .addComponent(lblFolha1)  
3346: .addComponent(jLabel39))  
3347: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
3348: .addComponent(jScrollPane3))  
3349: );  
3350:  
3351: jTabbedPane1.addTab("Veículos e Caminhões Leves", tab\_leves);  
3352:  
3353: panelPesados.setBackground(new java.awt.Color(255, 255, 255));  
3354: panelPesados.setMinimumSize(new java.awt.Dimension(200, 76));  
3355: panelPesados.setLayout(new java.awt.GridBagLayout());  
3356:  
3357: pesados\_hora1.setEditable(false);  
3358: pesados\_hora1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3359: gridBagConstraints = new java.awt.GridBagConstraints();  
3360: gridBagConstraints.gridx = 2;  
3361: gridBagConstraints.gridy = 0;  
3362: gridBagConstraints.gridwidth = 4;  
3363: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3364: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3365: panelPesados.add(pesados\_hora1, gridBagConstraints);  
3366:  
3367: pes15\_1.setEditable(false);  
3368: pes15\_1.setBackground(new java.awt.Color(204, 204, 204));  
3369: pes15\_1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3370: pes15\_1.setText("15");  
3371: gridBagConstraints = new java.awt.GridBagConstraints();  
3372: gridBagConstraints.gridx = 2;  
3373: gridBagConstraints.gridy = 1;  
3374: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3375: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3376: panelPesados.add(pes15\_1, gridBagConstraints);  
3377:  
3378: pes30\_1.setEditable(false);  
3379: pes30\_1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3380: pes30\_1.setText("30");  
3381: gridBagConstraints = new java.awt.GridBagConstraints();  
3382: gridBagConstraints.gridx = 3;  
3383: gridBagConstraints.gridy = 1;  
3384: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3385: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3386: panelPesados.add(pes30\_1, gridBagConstraints);  
3387:  
3388: pes45\_1.setEditable(false);  
3389: pes45\_1.setBackground(new java.awt.Color(204, 204, 204));  
3390: pes45\_1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3391: pes45\_1.setText("45");  
3392: gridBagConstraints = new java.awt.GridBagConstraints();  
3393: gridBagConstraints.gridx = 4;  
3394: gridBagConstraints.gridy = 1;  
3395: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3396: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3397: panelPesados.add(pes45\_1, gridBagConstraints);  
3398:  
3399: pes60\_1.setEditable(false);  
3400: pes60\_1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3401: pes60\_1.setText("60");  
3402: gridBagConstraints = new java.awt.GridBagConstraints();  
3403: gridBagConstraints.gridx = 5;  
3404: gridBagConstraints.gridy = 1;  
3405: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3406: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3407: panelPesados.add(pes60\_1, gridBagConstraints);  
3408:  
3409: pesados\_hora2.setEditable(false);  
3410: pesados\_hora2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3411: gridBagConstraints = new java.awt.GridBagConstraints();  
3412: gridBagConstraints.gridx = 6;  
3413: gridBagConstraints.gridy = 0;  
3414: gridBagConstraints.gridwidth = 4;  
3415: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3416: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3417: panelPesados.add(pesados\_hora2, gridBagConstraints);  
3418:  
3419: pes15\_2.setEditable(false);  
3420: pes15\_2.setBackground(new java.awt.Color(204, 204, 204));  
3421: pes15\_2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3422: pes15\_2.setText("15");  
3423: gridBagConstraints = new java.awt.GridBagConstraints();  
3424: gridBagConstraints.gridx = 6;  
3425: gridBagConstraints.gridy = 1;  
3426: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3427: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3428: panelPesados.add(pes15\_2, gridBagConstraints);  
3429:  
3430: pes30\_2.setEditable(false);  
3431: pes30\_2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3432: pes30\_2.setText("30");  
3433: gridBagConstraints = new java.awt.GridBagConstraints();  
3434: gridBagConstraints.gridx = 7;  
3435: gridBagConstraints.gridy = 1;  
3436: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3437: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3438: panelPesados.add(pes30\_2, gridBagConstraints);  
3439:  
3440: peso45\_2.setEditable(false);  
3441: peso45\_2.setBackground(new java.awt.Color(204, 204, 204));  
3442: peso45\_2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3443: peso45\_2.setText("45");  
3444: gridBagConstraints = new java.awt.GridBagConstraints();  
3445: gridBagConstraints.gridx = 8;  
3446: gridBagConstraints.gridy = 1;  
3447: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3448: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3449: panelPesados.add(peso45\_2, gridBagConstraints);  
3450:  
3451: peso60\_2.setEditable(false);  
3452: peso60\_2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3453: peso60\_2.setText("60");  
3454: gridBagConstraints = new java.awt.GridBagConstraints();  
3455: gridBagConstraints.gridx = 9;  
3456: gridBagConstraints.gridy = 1;  
3457: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3458: gridBagConstraints.anchor = java.awt.GridBagConstraints.NORTHWEST;  
3459: panelPesados.add(peso60\_2, gridBagConstraints);  
3460:  
3461: lbl2S1.setBackground(new java.awt.Color(255, 255, 255));  
3462: lbl2S1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
3463: lbl2S1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/semireboques/s1.png"))); // NOI18N  
3464: lbl2S1.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
3465: lbl2S1.setFocusable(false);  
3466: lbl2S1.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
3467: lbl2S1.setMinimumSize(new java.awt.Dimension(200, 76));  
3468: lbl2S1.setName(""); // NOI18N  
3469: lbl2S1.setOpaque(true);  
3470: gridBagConstraints = new java.awt.GridBagConstraints();  
3471: gridBagConstraints.gridx = 0;  
3472: gridBagConstraints.gridy = 2;  
3473: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3474: panelPesados.add(lbl2S1, gridBagConstraints);  
3475: lbl2S1.getAccessibleContext().setAccessibleDescription("");  
3476:  
3477: tp1.setColumns(5);  
3478: tp1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3479: tp1.setToolTipText("");  
3480: tp1.setInputVerifier(intVerifier);  
3481: tp1.setMinimumSize(new java.awt.Dimension(50, 19));  
3482: tp1.setName(""); // NOI18N  
3483: tp1.setNextFocusableComponent(tp2);  
3484: gridBagConstraints = new java.awt.GridBagConstraints();  
3485: gridBagConstraints.gridx = 3;  
3486: gridBagConstraints.gridy = 2;  
3487: gridBagConstraints.weightx = 1.0;  
3488: panelPesados.add(tp1, gridBagConstraints);  
3489:  
3490: tp3.setColumns(5);  
3491: tp3.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3492: tp3.setToolTipText("");  
3493: tp3.setInputVerifier(intVerifier);  
3494: tp3.setMinimumSize(new java.awt.Dimension(50, 19));  
3495: tp3.setName(""); // NOI18N  
3496: tp3.setNextFocusableComponent(tp4);  
3497: gridBagConstraints = new java.awt.GridBagConstraints();  
3498: gridBagConstraints.gridx = 5;  
3499: gridBagConstraints.gridy = 2;  
3500: gridBagConstraints.weightx = 1.0;  
3501: panelPesados.add(tp3, gridBagConstraints);  
3502:  
3503: tp5.setColumns(5);  
3504: tp5.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3505: tp5.setToolTipText("");  
3506: tp5.setInputVerifier(intVerifier);  
3507: tp5.setMinimumSize(new java.awt.Dimension(50, 19));  
3508: tp5.setName(""); // NOI18N  
3509: tp5.setNextFocusableComponent(tp6);  
3510: gridBagConstraints = new java.awt.GridBagConstraints();  
3511: gridBagConstraints.gridx = 7;  
3512: gridBagConstraints.gridy = 2;  
3513: gridBagConstraints.weightx = 1.0;  
3514: panelPesados.add(tp5, gridBagConstraints);  
3515:  
3516: tp7.setColumns(5);  
3517: tp7.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3518: tp7.setToolTipText("");  
3519: tp7.setInputVerifier(intVerifier);  
3520: tp7.setMinimumSize(new java.awt.Dimension(50, 19));  
3521: tp7.setName(""); // NOI18N  
3522: tp7.setNextFocusableComponent(tp8);  
3523: gridBagConstraints = new java.awt.GridBagConstraints();  
3524: gridBagConstraints.gridx = 9;  
3525: gridBagConstraints.gridy = 2;  
3526: gridBagConstraints.weightx = 1.0;  
3527: panelPesados.add(tp7, gridBagConstraints);  
3528:  
3529: lbl2S2.setBackground(new java.awt.Color(255, 255, 255));  
3530: lbl2S2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
3531: lbl2S2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/semireboques/s2.png"))); // NOI18N  
3532: lbl2S2.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
3533: lbl2S2.setFocusable(false);  
3534: lbl2S2.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
3535: lbl2S2.setMinimumSize(new java.awt.Dimension(200, 76));  
3536: lbl2S2.setOpaque(true);  
3537: gridBagConstraints = new java.awt.GridBagConstraints();  
3538: gridBagConstraints.gridx = 0;  
3539: gridBagConstraints.gridy = 3;  
3540: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3541: panelPesados.add(lbl2S2, gridBagConstraints);  
3542:  
3543: tp9.setColumns(5);  
3544: tp9.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3545: tp9.setToolTipText("");  
3546: tp9.setInputVerifier(intVerifier);  
3547: tp9.setMinimumSize(new java.awt.Dimension(50, 19));  
3548: tp9.setName(""); // NOI18N  
3549: tp9.setNextFocusableComponent(tp10);  
3550: gridBagConstraints = new java.awt.GridBagConstraints();  
3551: gridBagConstraints.gridx = 3;  
3552: gridBagConstraints.gridy = 3;  
3553: gridBagConstraints.weightx = 1.0;  
3554: panelPesados.add(tp9, gridBagConstraints);  
3555:  
3556: tp11.setColumns(5);  
3557: tp11.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3558: tp11.setToolTipText("");  
3559: tp11.setInputVerifier(intVerifier);  
3560: tp11.setMinimumSize(new java.awt.Dimension(50, 19));  
3561: tp11.setName(""); // NOI18N  
3562: tp11.setNextFocusableComponent(tp12);  
3563: gridBagConstraints = new java.awt.GridBagConstraints();  
3564: gridBagConstraints.gridx = 5;  
3565: gridBagConstraints.gridy = 3;  
3566: gridBagConstraints.weightx = 1.0;  
3567: panelPesados.add(tp11, gridBagConstraints);  
3568:  
3569: tp13.setColumns(5);  
3570: tp13.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3571: tp13.setToolTipText("");  
3572: tp13.setInputVerifier(intVerifier);  
3573: tp13.setMinimumSize(new java.awt.Dimension(50, 19));  
3574: tp13.setName(""); // NOI18N  
3575: tp13.setNextFocusableComponent(tp14);  
3576: gridBagConstraints = new java.awt.GridBagConstraints();  
3577: gridBagConstraints.gridx = 7;  
3578: gridBagConstraints.gridy = 3;  
3579: gridBagConstraints.weightx = 1.0;  
3580: panelPesados.add(tp13, gridBagConstraints);  
3581:  
3582: tp15.setColumns(5);  
3583: tp15.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3584: tp15.setToolTipText("");  
3585: tp15.setInputVerifier(intVerifier);  
3586: tp15.setMinimumSize(new java.awt.Dimension(50, 19));  
3587: tp15.setName(""); // NOI18N  
3588: tp15.setNextFocusableComponent(tp16);  
3589: gridBagConstraints = new java.awt.GridBagConstraints();  
3590: gridBagConstraints.gridx = 9;  
3591: gridBagConstraints.gridy = 3;  
3592: gridBagConstraints.weightx = 1.0;  
3593: panelPesados.add(tp15, gridBagConstraints);  
3594:  
3595: lbl2S3.setBackground(new java.awt.Color(255, 255, 255));  
3596: lbl2S3.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
3597: lbl2S3.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/semireboques/s3.png"))); // NOI18N  
3598: lbl2S3.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
3599: lbl2S3.setFocusable(false);  
3600: lbl2S3.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
3601: lbl2S3.setMinimumSize(new java.awt.Dimension(200, 76));  
3602: lbl2S3.setOpaque(true);  
3603: gridBagConstraints = new java.awt.GridBagConstraints();  
3604: gridBagConstraints.gridx = 0;  
3605: gridBagConstraints.gridy = 4;  
3606: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3607: panelPesados.add(lbl2S3, gridBagConstraints);  
3608:  
3609: tp17.setColumns(5);  
3610: tp17.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3611: tp17.setToolTipText("");  
3612: tp17.setInputVerifier(intVerifier);  
3613: tp17.setMinimumSize(new java.awt.Dimension(50, 19));  
3614: tp17.setName(""); // NOI18N  
3615: tp17.setNextFocusableComponent(tp18);  
3616: gridBagConstraints = new java.awt.GridBagConstraints();  
3617: gridBagConstraints.gridx = 3;  
3618: gridBagConstraints.gridy = 4;  
3619: gridBagConstraints.weightx = 1.0;  
3620: panelPesados.add(tp17, gridBagConstraints);  
3621:  
3622: tp19.setColumns(5);  
3623: tp19.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3624: tp19.setToolTipText("");  
3625: tp19.setInputVerifier(intVerifier);  
3626: tp19.setMinimumSize(new java.awt.Dimension(50, 19));  
3627: tp19.setName(""); // NOI18N  
3628: tp19.setNextFocusableComponent(tp20);  
3629: gridBagConstraints = new java.awt.GridBagConstraints();  
3630: gridBagConstraints.gridx = 5;  
3631: gridBagConstraints.gridy = 4;  
3632: gridBagConstraints.weightx = 1.0;  
3633: panelPesados.add(tp19, gridBagConstraints);  
3634:  
3635: tp21.setColumns(5);  
3636: tp21.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3637: tp21.setToolTipText("");  
3638: tp21.setInputVerifier(intVerifier);  
3639: tp21.setMinimumSize(new java.awt.Dimension(50, 19));  
3640: tp21.setName(""); // NOI18N  
3641: tp21.setNextFocusableComponent(tp22);  
3642: gridBagConstraints = new java.awt.GridBagConstraints();  
3643: gridBagConstraints.gridx = 7;  
3644: gridBagConstraints.gridy = 4;  
3645: gridBagConstraints.weightx = 1.0;  
3646: panelPesados.add(tp21, gridBagConstraints);  
3647:  
3648: tp23.setColumns(5);  
3649: tp23.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3650: tp23.setToolTipText("");  
3651: tp23.setInputVerifier(intVerifier);  
3652: tp23.setMinimumSize(new java.awt.Dimension(50, 19));  
3653: tp23.setName(""); // NOI18N  
3654: tp23.setNextFocusableComponent(tp24);  
3655: gridBagConstraints = new java.awt.GridBagConstraints();  
3656: gridBagConstraints.gridx = 9;  
3657: gridBagConstraints.gridy = 4;  
3658: gridBagConstraints.weightx = 1.0;  
3659: panelPesados.add(tp23, gridBagConstraints);  
3660:  
3661: lbl3S1.setBackground(new java.awt.Color(255, 255, 255));  
3662: lbl3S1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
3663: lbl3S1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/semireboques/s4.png"))); // NOI18N  
3664: lbl3S1.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
3665: lbl3S1.setFocusable(false);  
3666: lbl3S1.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
3667: lbl3S1.setMinimumSize(new java.awt.Dimension(200, 76));  
3668: lbl3S1.setOpaque(true);  
3669: gridBagConstraints = new java.awt.GridBagConstraints();  
3670: gridBagConstraints.gridx = 0;  
3671: gridBagConstraints.gridy = 5;  
3672: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3673: panelPesados.add(lbl3S1, gridBagConstraints);  
3674:  
3675: tp25.setColumns(5);  
3676: tp25.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3677: tp25.setToolTipText("");  
3678: tp25.setInputVerifier(intVerifier);  
3679: tp25.setMinimumSize(new java.awt.Dimension(50, 19));  
3680: tp25.setName(""); // NOI18N  
3681: tp25.setNextFocusableComponent(tp26);  
3682: gridBagConstraints = new java.awt.GridBagConstraints();  
3683: gridBagConstraints.gridx = 3;  
3684: gridBagConstraints.gridy = 5;  
3685: gridBagConstraints.weightx = 1.0;  
3686: panelPesados.add(tp25, gridBagConstraints);  
3687:  
3688: tp27.setColumns(5);  
3689: tp27.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3690: tp27.setToolTipText("");  
3691: tp27.setInputVerifier(intVerifier);  
3692: tp27.setMinimumSize(new java.awt.Dimension(50, 19));  
3693: tp27.setName(""); // NOI18N  
3694: tp27.setNextFocusableComponent(tp28);  
3695: gridBagConstraints = new java.awt.GridBagConstraints();  
3696: gridBagConstraints.gridx = 5;  
3697: gridBagConstraints.gridy = 5;  
3698: gridBagConstraints.weightx = 1.0;  
3699: panelPesados.add(tp27, gridBagConstraints);  
3700:  
3701: tp29.setColumns(5);  
3702: tp29.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3703: tp29.setToolTipText("");  
3704: tp29.setInputVerifier(intVerifier);  
3705: tp29.setMinimumSize(new java.awt.Dimension(50, 19));  
3706: tp29.setName(""); // NOI18N  
3707: tp29.setNextFocusableComponent(tp30);  
3708: gridBagConstraints = new java.awt.GridBagConstraints();  
3709: gridBagConstraints.gridx = 7;  
3710: gridBagConstraints.gridy = 5;  
3711: gridBagConstraints.weightx = 1.0;  
3712: panelPesados.add(tp29, gridBagConstraints);  
3713:  
3714: tp31.setColumns(5);  
3715: tp31.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3716: tp31.setToolTipText("");  
3717: tp31.setInputVerifier(intVerifier);  
3718: tp31.setMinimumSize(new java.awt.Dimension(50, 19));  
3719: tp31.setName(""); // NOI18N  
3720: tp31.setNextFocusableComponent(tp32);  
3721: gridBagConstraints = new java.awt.GridBagConstraints();  
3722: gridBagConstraints.gridx = 9;  
3723: gridBagConstraints.gridy = 5;  
3724: gridBagConstraints.weightx = 1.0;  
3725: panelPesados.add(tp31, gridBagConstraints);  
3726:  
3727: lbl3S2.setBackground(new java.awt.Color(255, 255, 255));  
3728: lbl3S2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
3729: lbl3S2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/semireboques/s5.png"))); // NOI18N  
3730: lbl3S2.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
3731: lbl3S2.setFocusable(false);  
3732: lbl3S2.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
3733: lbl3S2.setMinimumSize(new java.awt.Dimension(200, 76));  
3734: lbl3S2.setOpaque(true);  
3735: gridBagConstraints = new java.awt.GridBagConstraints();  
3736: gridBagConstraints.gridx = 0;  
3737: gridBagConstraints.gridy = 6;  
3738: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3739: panelPesados.add(lbl3S2, gridBagConstraints);  
3740:  
3741: tp33.setColumns(5);  
3742: tp33.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3743: tp33.setToolTipText("");  
3744: tp33.setInputVerifier(intVerifier);  
3745: tp33.setMinimumSize(new java.awt.Dimension(50, 19));  
3746: tp33.setName(""); // NOI18N  
3747: tp33.setNextFocusableComponent(tp34);  
3748: gridBagConstraints = new java.awt.GridBagConstraints();  
3749: gridBagConstraints.gridx = 3;  
3750: gridBagConstraints.gridy = 6;  
3751: panelPesados.add(tp33, gridBagConstraints);  
3752:  
3753: tp35.setColumns(5);  
3754: tp35.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3755: tp35.setToolTipText("");  
3756: tp35.setInputVerifier(intVerifier);  
3757: tp35.setMinimumSize(new java.awt.Dimension(50, 19));  
3758: tp35.setName(""); // NOI18N  
3759: tp35.setNextFocusableComponent(tp36);  
3760: gridBagConstraints = new java.awt.GridBagConstraints();  
3761: gridBagConstraints.gridx = 5;  
3762: gridBagConstraints.gridy = 6;  
3763: panelPesados.add(tp35, gridBagConstraints);  
3764:  
3765: tp37.setColumns(5);  
3766: tp37.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3767: tp37.setToolTipText("");  
3768: tp37.setInputVerifier(intVerifier);  
3769: tp37.setMinimumSize(new java.awt.Dimension(50, 19));  
3770: tp37.setName(""); // NOI18N  
3771: tp37.setNextFocusableComponent(tp38);  
3772: gridBagConstraints = new java.awt.GridBagConstraints();  
3773: gridBagConstraints.gridx = 7;  
3774: gridBagConstraints.gridy = 6;  
3775: panelPesados.add(tp37, gridBagConstraints);  
3776:  
3777: tp39.setColumns(5);  
3778: tp39.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3779: tp39.setToolTipText("");  
3780: tp39.setInputVerifier(intVerifier);  
3781: tp39.setMinimumSize(new java.awt.Dimension(50, 19));  
3782: tp39.setName(""); // NOI18N  
3783: tp39.setNextFocusableComponent(tp40);  
3784: gridBagConstraints = new java.awt.GridBagConstraints();  
3785: gridBagConstraints.gridx = 9;  
3786: gridBagConstraints.gridy = 6;  
3787: panelPesados.add(tp39, gridBagConstraints);  
3788:  
3789: lbl3S3.setBackground(new java.awt.Color(255, 255, 255));  
3790: lbl3S3.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
3791: lbl3S3.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/semireboques/s6.png"))); // NOI18N  
3792: lbl3S3.setToolTipText("");  
3793: lbl3S3.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
3794: lbl3S3.setFocusable(false);  
3795: lbl3S3.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
3796: lbl3S3.setMinimumSize(new java.awt.Dimension(200, 76));  
3797: lbl3S3.setOpaque(true);  
3798: gridBagConstraints = new java.awt.GridBagConstraints();  
3799: gridBagConstraints.gridx = 0;  
3800: gridBagConstraints.gridy = 7;  
3801: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3802: panelPesados.add(lbl3S3, gridBagConstraints);  
3803:  
3804: tp41.setColumns(5);  
3805: tp41.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3806: tp41.setToolTipText("");  
3807: tp41.setInputVerifier(intVerifier);  
3808: tp41.setMinimumSize(new java.awt.Dimension(50, 19));  
3809: tp41.setName(""); // NOI18N  
3810: tp41.setNextFocusableComponent(tp42);  
3811: gridBagConstraints = new java.awt.GridBagConstraints();  
3812: gridBagConstraints.gridx = 3;  
3813: gridBagConstraints.gridy = 7;  
3814: panelPesados.add(tp41, gridBagConstraints);  
3815:  
3816: tp43.setColumns(5);  
3817: tp43.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3818: tp43.setToolTipText("");  
3819: tp43.setInputVerifier(intVerifier);  
3820: tp43.setMinimumSize(new java.awt.Dimension(50, 19));  
3821: tp43.setName(""); // NOI18N  
3822: tp43.setNextFocusableComponent(tp44);  
3823: gridBagConstraints = new java.awt.GridBagConstraints();  
3824: gridBagConstraints.gridx = 5;  
3825: gridBagConstraints.gridy = 7;  
3826: panelPesados.add(tp43, gridBagConstraints);  
3827:  
3828: tp45.setColumns(5);  
3829: tp45.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3830: tp45.setToolTipText("");  
3831: tp45.setInputVerifier(intVerifier);  
3832: tp45.setMinimumSize(new java.awt.Dimension(50, 19));  
3833: tp45.setName(""); // NOI18N  
3834: tp45.setNextFocusableComponent(tp46);  
3835: gridBagConstraints = new java.awt.GridBagConstraints();  
3836: gridBagConstraints.gridx = 7;  
3837: gridBagConstraints.gridy = 7;  
3838: panelPesados.add(tp45, gridBagConstraints);  
3839:  
3840: tp47.setColumns(5);  
3841: tp47.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3842: tp47.setToolTipText("");  
3843: tp47.setInputVerifier(intVerifier);  
3844: tp47.setMinimumSize(new java.awt.Dimension(50, 19));  
3845: tp47.setName(""); // NOI18N  
3846: tp47.setNextFocusableComponent(tp48);  
3847: gridBagConstraints = new java.awt.GridBagConstraints();  
3848: gridBagConstraints.gridx = 9;  
3849: gridBagConstraints.gridy = 7;  
3850: panelPesados.add(tp47, gridBagConstraints);  
3851:  
3852: lbl3T4.setBackground(new java.awt.Color(255, 255, 255));  
3853: lbl3T4.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
3854: lbl3T4.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/combinados/se1.png"))); // NOI18N  
3855: lbl3T4.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
3856: lbl3T4.setFocusable(false);  
3857: lbl3T4.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
3858: lbl3T4.setMinimumSize(new java.awt.Dimension(200, 76));  
3859: lbl3T4.setOpaque(true);  
3860: gridBagConstraints = new java.awt.GridBagConstraints();  
3861: gridBagConstraints.gridx = 0;  
3862: gridBagConstraints.gridy = 8;  
3863: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3864: panelPesados.add(lbl3T4, gridBagConstraints);  
3865:  
3866: tp49.setColumns(5);  
3867: tp49.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3868: tp49.setToolTipText("");  
3869: tp49.setInputVerifier(intVerifier);  
3870: tp49.setMinimumSize(new java.awt.Dimension(50, 19));  
3871: tp49.setName(""); // NOI18N  
3872: tp49.setNextFocusableComponent(tp50);  
3873: gridBagConstraints = new java.awt.GridBagConstraints();  
3874: gridBagConstraints.gridx = 3;  
3875: gridBagConstraints.gridy = 8;  
3876: panelPesados.add(tp49, gridBagConstraints);  
3877:  
3878: tp51.setColumns(5);  
3879: tp51.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3880: tp51.setToolTipText("");  
3881: tp51.setInputVerifier(intVerifier);  
3882: tp51.setMinimumSize(new java.awt.Dimension(50, 19));  
3883: tp51.setName(""); // NOI18N  
3884: tp51.setNextFocusableComponent(tp52);  
3885: gridBagConstraints = new java.awt.GridBagConstraints();  
3886: gridBagConstraints.gridx = 5;  
3887: gridBagConstraints.gridy = 8;  
3888: panelPesados.add(tp51, gridBagConstraints);  
3889:  
3890: tp53.setColumns(5);  
3891: tp53.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3892: tp53.setToolTipText("");  
3893: tp53.setInputVerifier(intVerifier);  
3894: tp53.setMinimumSize(new java.awt.Dimension(50, 19));  
3895: tp53.setName(""); // NOI18N  
3896: tp53.setNextFocusableComponent(tp54);  
3897: gridBagConstraints = new java.awt.GridBagConstraints();  
3898: gridBagConstraints.gridx = 7;  
3899: gridBagConstraints.gridy = 8;  
3900: panelPesados.add(tp53, gridBagConstraints);  
3901:  
3902: tp55.setColumns(5);  
3903: tp55.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3904: tp55.setToolTipText("");  
3905: tp55.setInputVerifier(intVerifier);  
3906: tp55.setMinimumSize(new java.awt.Dimension(50, 19));  
3907: tp55.setName(""); // NOI18N  
3908: tp55.setNextFocusableComponent(tp56);  
3909: gridBagConstraints = new java.awt.GridBagConstraints();  
3910: gridBagConstraints.gridx = 9;  
3911: gridBagConstraints.gridy = 8;  
3912: panelPesados.add(tp55, gridBagConstraints);  
3913:  
3914: lbl3T6.setBackground(new java.awt.Color(255, 255, 255));  
3915: lbl3T6.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
3916: lbl3T6.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/combinados/se2.png"))); // NOI18N  
3917: lbl3T6.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
3918: lbl3T6.setFocusable(false);  
3919: lbl3T6.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
3920: lbl3T6.setMinimumSize(new java.awt.Dimension(200, 76));  
3921: lbl3T6.setOpaque(true);  
3922: gridBagConstraints = new java.awt.GridBagConstraints();  
3923: gridBagConstraints.gridx = 0;  
3924: gridBagConstraints.gridy = 9;  
3925: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3926: panelPesados.add(lbl3T6, gridBagConstraints);  
3927:  
3928: tp57.setColumns(5);  
3929: tp57.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3930: tp57.setToolTipText("");  
3931: tp57.setInputVerifier(intVerifier);  
3932: tp57.setMinimumSize(new java.awt.Dimension(50, 19));  
3933: tp57.setName(""); // NOI18N  
3934: tp57.setNextFocusableComponent(tp58);  
3935: gridBagConstraints = new java.awt.GridBagConstraints();  
3936: gridBagConstraints.gridx = 3;  
3937: gridBagConstraints.gridy = 9;  
3938: panelPesados.add(tp57, gridBagConstraints);  
3939:  
3940: tp59.setColumns(5);  
3941: tp59.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3942: tp59.setToolTipText("");  
3943: tp59.setInputVerifier(intVerifier);  
3944: tp59.setMinimumSize(new java.awt.Dimension(50, 19));  
3945: tp59.setName(""); // NOI18N  
3946: tp59.setNextFocusableComponent(tp60);  
3947: gridBagConstraints = new java.awt.GridBagConstraints();  
3948: gridBagConstraints.gridx = 5;  
3949: gridBagConstraints.gridy = 9;  
3950: panelPesados.add(tp59, gridBagConstraints);  
3951:  
3952: tp61.setColumns(5);  
3953: tp61.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3954: tp61.setToolTipText("");  
3955: tp61.setInputVerifier(intVerifier);  
3956: tp61.setMinimumSize(new java.awt.Dimension(50, 19));  
3957: tp61.setName(""); // NOI18N  
3958: tp61.setNextFocusableComponent(tp62);  
3959: gridBagConstraints = new java.awt.GridBagConstraints();  
3960: gridBagConstraints.gridx = 7;  
3961: gridBagConstraints.gridy = 9;  
3962: panelPesados.add(tp61, gridBagConstraints);  
3963:  
3964: tp63.setColumns(5);  
3965: tp63.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3966: tp63.setToolTipText("");  
3967: tp63.setInputVerifier(intVerifier);  
3968: tp63.setMinimumSize(new java.awt.Dimension(50, 19));  
3969: tp63.setName(""); // NOI18N  
3970: tp63.setNextFocusableComponent(tp64);  
3971: gridBagConstraints = new java.awt.GridBagConstraints();  
3972: gridBagConstraints.gridx = 9;  
3973: gridBagConstraints.gridy = 9;  
3974: panelPesados.add(tp63, gridBagConstraints);  
3975:  
3976: lbl3T6B.setBackground(new java.awt.Color(255, 255, 255));  
3977: lbl3T6B.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
3978: lbl3T6B.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/combinados/se3.png"))); // NOI18N  
3979: lbl3T6B.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
3980: lbl3T6B.setFocusable(false);  
3981: lbl3T6B.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
3982: lbl3T6B.setMinimumSize(new java.awt.Dimension(200, 76));  
3983: lbl3T6B.setOpaque(true);  
3984: gridBagConstraints = new java.awt.GridBagConstraints();  
3985: gridBagConstraints.gridx = 0;  
3986: gridBagConstraints.gridy = 10;  
3987: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
3988: panelPesados.add(lbl3T6B, gridBagConstraints);  
3989:  
3990: tp65.setColumns(5);  
3991: tp65.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
3992: tp65.setToolTipText("");  
3993: tp65.setInputVerifier(intVerifier);  
3994: tp65.setMinimumSize(new java.awt.Dimension(50, 19));  
3995: tp65.setName(""); // NOI18N  
3996: tp65.setNextFocusableComponent(tp66);  
3997: gridBagConstraints = new java.awt.GridBagConstraints();  
3998: gridBagConstraints.gridx = 3;  
3999: gridBagConstraints.gridy = 10;  
4000: panelPesados.add(tp65, gridBagConstraints);  
4001:  
4002: tp67.setColumns(5);  
4003: tp67.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4004: tp67.setToolTipText("");  
4005: tp67.setInputVerifier(intVerifier);  
4006: tp67.setMinimumSize(new java.awt.Dimension(50, 19));  
4007: tp67.setName(""); // NOI18N  
4008: tp67.setNextFocusableComponent(tp68);  
4009: gridBagConstraints = new java.awt.GridBagConstraints();  
4010: gridBagConstraints.gridx = 5;  
4011: gridBagConstraints.gridy = 10;  
4012: panelPesados.add(tp67, gridBagConstraints);  
4013:  
4014: tp69.setColumns(5);  
4015: tp69.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4016: tp69.setToolTipText("");  
4017: tp69.setInputVerifier(intVerifier);  
4018: tp69.setMinimumSize(new java.awt.Dimension(50, 19));  
4019: tp69.setName(""); // NOI18N  
4020: tp69.setNextFocusableComponent(tp70);  
4021: gridBagConstraints = new java.awt.GridBagConstraints();  
4022: gridBagConstraints.gridx = 7;  
4023: gridBagConstraints.gridy = 10;  
4024: panelPesados.add(tp69, gridBagConstraints);  
4025:  
4026: tp71.setColumns(5);  
4027: tp71.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4028: tp71.setToolTipText("");  
4029: tp71.setInputVerifier(intVerifier);  
4030: tp71.setMinimumSize(new java.awt.Dimension(50, 19));  
4031: tp71.setName(""); // NOI18N  
4032: tp71.setNextFocusableComponent(tp72);  
4033: gridBagConstraints = new java.awt.GridBagConstraints();  
4034: gridBagConstraints.gridx = 9;  
4035: gridBagConstraints.gridy = 10;  
4036: panelPesados.add(tp71, gridBagConstraints);  
4037:  
4038: lbl3V5.setBackground(new java.awt.Color(255, 255, 255));  
4039: lbl3V5.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
4040: lbl3V5.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/combinados/se4.png"))); // NOI18N  
4041: lbl3V5.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
4042: lbl3V5.setFocusable(false);  
4043: lbl3V5.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
4044: lbl3V5.setMinimumSize(new java.awt.Dimension(200, 76));  
4045: lbl3V5.setOpaque(true);  
4046: gridBagConstraints = new java.awt.GridBagConstraints();  
4047: gridBagConstraints.gridx = 0;  
4048: gridBagConstraints.gridy = 11;  
4049: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
4050: panelPesados.add(lbl3V5, gridBagConstraints);  
4051:  
4052: tp73.setColumns(5);  
4053: tp73.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4054: tp73.setToolTipText("");  
4055: tp73.setInputVerifier(intVerifier);  
4056: tp73.setMinimumSize(new java.awt.Dimension(50, 19));  
4057: tp73.setName(""); // NOI18N  
4058: tp73.setNextFocusableComponent(tp74);  
4059: gridBagConstraints = new java.awt.GridBagConstraints();  
4060: gridBagConstraints.gridx = 3;  
4061: gridBagConstraints.gridy = 11;  
4062: panelPesados.add(tp73, gridBagConstraints);  
4063:  
4064: tp75.setColumns(5);  
4065: tp75.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4066: tp75.setToolTipText("");  
4067: tp75.setInputVerifier(intVerifier);  
4068: tp75.setMinimumSize(new java.awt.Dimension(50, 19));  
4069: tp75.setName(""); // NOI18N  
4070: tp75.setNextFocusableComponent(tp76);  
4071: gridBagConstraints = new java.awt.GridBagConstraints();  
4072: gridBagConstraints.gridx = 5;  
4073: gridBagConstraints.gridy = 11;  
4074: panelPesados.add(tp75, gridBagConstraints);  
4075:  
4076: tp77.setColumns(5);  
4077: tp77.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4078: tp77.setToolTipText("");  
4079: tp77.setInputVerifier(intVerifier);  
4080: tp77.setMinimumSize(new java.awt.Dimension(50, 19));  
4081: tp77.setName(""); // NOI18N  
4082: tp77.setNextFocusableComponent(tp78);  
4083: gridBagConstraints = new java.awt.GridBagConstraints();  
4084: gridBagConstraints.gridx = 7;  
4085: gridBagConstraints.gridy = 11;  
4086: panelPesados.add(tp77, gridBagConstraints);  
4087:  
4088: tp79.setColumns(5);  
4089: tp79.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4090: tp79.setToolTipText("");  
4091: tp79.setInputVerifier(intVerifier);  
4092: tp79.setMinimumSize(new java.awt.Dimension(50, 19));  
4093: tp79.setName(""); // NOI18N  
4094: tp79.setNextFocusableComponent(tp80);  
4095: gridBagConstraints = new java.awt.GridBagConstraints();  
4096: gridBagConstraints.gridx = 9;  
4097: gridBagConstraints.gridy = 11;  
4098: panelPesados.add(tp79, gridBagConstraints);  
4099:  
4100: lbl3M6.setBackground(new java.awt.Color(255, 255, 255));  
4101: lbl3M6.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
4102: lbl3M6.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/combinados/se5.png"))); // NOI18N  
4103: lbl3M6.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
4104: lbl3M6.setFocusable(false);  
4105: lbl3M6.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
4106: lbl3M6.setMinimumSize(new java.awt.Dimension(200, 76));  
4107: lbl3M6.setOpaque(true);  
4108: gridBagConstraints = new java.awt.GridBagConstraints();  
4109: gridBagConstraints.gridx = 0;  
4110: gridBagConstraints.gridy = 12;  
4111: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
4112: panelPesados.add(lbl3M6, gridBagConstraints);  
4113:  
4114: tp81.setColumns(5);  
4115: tp81.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4116: tp81.setToolTipText("");  
4117: tp81.setInputVerifier(intVerifier);  
4118: tp81.setMinimumSize(new java.awt.Dimension(50, 19));  
4119: tp81.setName(""); // NOI18N  
4120: tp81.setNextFocusableComponent(tp82);  
4121: gridBagConstraints = new java.awt.GridBagConstraints();  
4122: gridBagConstraints.gridx = 3;  
4123: gridBagConstraints.gridy = 12;  
4124: panelPesados.add(tp81, gridBagConstraints);  
4125:  
4126: tp83.setColumns(5);  
4127: tp83.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4128: tp83.setToolTipText("");  
4129: tp83.setInputVerifier(intVerifier);  
4130: tp83.setMinimumSize(new java.awt.Dimension(50, 19));  
4131: tp83.setName(""); // NOI18N  
4132: tp83.setNextFocusableComponent(tp84);  
4133: gridBagConstraints = new java.awt.GridBagConstraints();  
4134: gridBagConstraints.gridx = 5;  
4135: gridBagConstraints.gridy = 12;  
4136: panelPesados.add(tp83, gridBagConstraints);  
4137:  
4138: tp85.setColumns(5);  
4139: tp85.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4140: tp85.setToolTipText("");  
4141: tp85.setInputVerifier(intVerifier);  
4142: tp85.setMinimumSize(new java.awt.Dimension(50, 19));  
4143: tp85.setName(""); // NOI18N  
4144: tp85.setNextFocusableComponent(tp86);  
4145: gridBagConstraints = new java.awt.GridBagConstraints();  
4146: gridBagConstraints.gridx = 7;  
4147: gridBagConstraints.gridy = 12;  
4148: panelPesados.add(tp85, gridBagConstraints);  
4149:  
4150: tp87.setColumns(5);  
4151: tp87.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4152: tp87.setToolTipText("");  
4153: tp87.setInputVerifier(intVerifier);  
4154: tp87.setMinimumSize(new java.awt.Dimension(50, 19));  
4155: tp87.setName(""); // NOI18N  
4156: tp87.setNextFocusableComponent(tp88);  
4157: gridBagConstraints = new java.awt.GridBagConstraints();  
4158: gridBagConstraints.gridx = 9;  
4159: gridBagConstraints.gridy = 12;  
4160: panelPesados.add(tp87, gridBagConstraints);  
4161:  
4162: lbl3Q4.setBackground(new java.awt.Color(255, 255, 255));  
4163: lbl3Q4.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
4164: lbl3Q4.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/reboques/r1.png"))); // NOI18N  
4165: lbl3Q4.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
4166: lbl3Q4.setFocusable(false);  
4167: lbl3Q4.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
4168: lbl3Q4.setMinimumSize(new java.awt.Dimension(200, 76));  
4169: lbl3Q4.setOpaque(true);  
4170: gridBagConstraints = new java.awt.GridBagConstraints();  
4171: gridBagConstraints.gridx = 0;  
4172: gridBagConstraints.gridy = 13;  
4173: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
4174: panelPesados.add(lbl3Q4, gridBagConstraints);  
4175:  
4176: tp89.setColumns(5);  
4177: tp89.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4178: tp89.setToolTipText("");  
4179: tp89.setInputVerifier(intVerifier);  
4180: tp89.setMinimumSize(new java.awt.Dimension(50, 19));  
4181: tp89.setName(""); // NOI18N  
4182: tp89.setNextFocusableComponent(tp90);  
4183: gridBagConstraints = new java.awt.GridBagConstraints();  
4184: gridBagConstraints.gridx = 3;  
4185: gridBagConstraints.gridy = 13;  
4186: panelPesados.add(tp89, gridBagConstraints);  
4187:  
4188: tp91.setColumns(5);  
4189: tp91.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4190: tp91.setToolTipText("");  
4191: tp91.setInputVerifier(intVerifier);  
4192: tp91.setMinimumSize(new java.awt.Dimension(50, 19));  
4193: tp91.setName(""); // NOI18N  
4194: tp91.setNextFocusableComponent(tp92);  
4195: gridBagConstraints = new java.awt.GridBagConstraints();  
4196: gridBagConstraints.gridx = 5;  
4197: gridBagConstraints.gridy = 13;  
4198: panelPesados.add(tp91, gridBagConstraints);  
4199:  
4200: tp93.setColumns(5);  
4201: tp93.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4202: tp93.setToolTipText("");  
4203: tp93.setInputVerifier(intVerifier);  
4204: tp93.setMinimumSize(new java.awt.Dimension(50, 19));  
4205: tp93.setName(""); // NOI18N  
4206: tp93.setNextFocusableComponent(tp94);  
4207: gridBagConstraints = new java.awt.GridBagConstraints();  
4208: gridBagConstraints.gridx = 7;  
4209: gridBagConstraints.gridy = 13;  
4210: panelPesados.add(tp93, gridBagConstraints);  
4211:  
4212: tp95.setColumns(5);  
4213: tp95.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4214: tp95.setToolTipText("");  
4215: tp95.setInputVerifier(intVerifier);  
4216: tp95.setMinimumSize(new java.awt.Dimension(50, 19));  
4217: tp95.setName(""); // NOI18N  
4218: tp95.setNextFocusableComponent(tp96);  
4219: gridBagConstraints = new java.awt.GridBagConstraints();  
4220: gridBagConstraints.gridx = 9;  
4221: gridBagConstraints.gridy = 13;  
4222: panelPesados.add(tp95, gridBagConstraints);  
4223:  
4224: lbl2C2.setBackground(new java.awt.Color(255, 255, 255));  
4225: lbl2C2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
4226: lbl2C2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/reboques/r2.png"))); // NOI18N  
4227: lbl2C2.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
4228: lbl2C2.setFocusable(false);  
4229: lbl2C2.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
4230: lbl2C2.setMinimumSize(new java.awt.Dimension(200, 76));  
4231: lbl2C2.setOpaque(true);  
4232: gridBagConstraints = new java.awt.GridBagConstraints();  
4233: gridBagConstraints.gridx = 0;  
4234: gridBagConstraints.gridy = 14;  
4235: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
4236: panelPesados.add(lbl2C2, gridBagConstraints);  
4237:  
4238: tp97.setColumns(5);  
4239: tp97.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4240: tp97.setToolTipText("");  
4241: tp97.setInputVerifier(intVerifier);  
4242: tp97.setMinimumSize(new java.awt.Dimension(50, 19));  
4243: tp97.setName(""); // NOI18N  
4244: tp97.setNextFocusableComponent(tp98);  
4245: gridBagConstraints = new java.awt.GridBagConstraints();  
4246: gridBagConstraints.gridx = 3;  
4247: gridBagConstraints.gridy = 14;  
4248: panelPesados.add(tp97, gridBagConstraints);  
4249:  
4250: tp99.setColumns(5);  
4251: tp99.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4252: tp99.setToolTipText("");  
4253: tp99.setInputVerifier(intVerifier);  
4254: tp99.setMinimumSize(new java.awt.Dimension(50, 19));  
4255: tp99.setName(""); // NOI18N  
4256: tp99.setNextFocusableComponent(tp100);  
4257: gridBagConstraints = new java.awt.GridBagConstraints();  
4258: gridBagConstraints.gridx = 5;  
4259: gridBagConstraints.gridy = 14;  
4260: panelPesados.add(tp99, gridBagConstraints);  
4261:  
4262: tp101.setColumns(5);  
4263: tp101.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4264: tp101.setToolTipText("");  
4265: tp101.setInputVerifier(intVerifier);  
4266: tp101.setMinimumSize(new java.awt.Dimension(50, 19));  
4267: tp101.setName(""); // NOI18N  
4268: tp101.setNextFocusableComponent(tp102);  
4269: gridBagConstraints = new java.awt.GridBagConstraints();  
4270: gridBagConstraints.gridx = 7;  
4271: gridBagConstraints.gridy = 14;  
4272: panelPesados.add(tp101, gridBagConstraints);  
4273:  
4274: tp103.setColumns(5);  
4275: tp103.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4276: tp103.setToolTipText("");  
4277: tp103.setInputVerifier(intVerifier);  
4278: tp103.setMinimumSize(new java.awt.Dimension(50, 19));  
4279: tp103.setName(""); // NOI18N  
4280: tp103.setNextFocusableComponent(tp104);  
4281: gridBagConstraints = new java.awt.GridBagConstraints();  
4282: gridBagConstraints.gridx = 9;  
4283: gridBagConstraints.gridy = 14;  
4284: panelPesados.add(tp103, gridBagConstraints);  
4285:  
4286: lbl2C3.setBackground(new java.awt.Color(255, 255, 255));  
4287: lbl2C3.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
4288: lbl2C3.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/reboques/r3.png"))); // NOI18N  
4289: lbl2C3.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
4290: lbl2C3.setFocusable(false);  
4291: lbl2C3.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
4292: lbl2C3.setMinimumSize(new java.awt.Dimension(200, 76));  
4293: lbl2C3.setOpaque(true);  
4294: gridBagConstraints = new java.awt.GridBagConstraints();  
4295: gridBagConstraints.gridx = 0;  
4296: gridBagConstraints.gridy = 15;  
4297: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
4298: panelPesados.add(lbl2C3, gridBagConstraints);  
4299:  
4300: tp105.setColumns(5);  
4301: tp105.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4302: tp105.setToolTipText("");  
4303: tp105.setInputVerifier(intVerifier);  
4304: tp105.setMinimumSize(new java.awt.Dimension(50, 19));  
4305: tp105.setName(""); // NOI18N  
4306: tp105.setNextFocusableComponent(tp106);  
4307: gridBagConstraints = new java.awt.GridBagConstraints();  
4308: gridBagConstraints.gridx = 3;  
4309: gridBagConstraints.gridy = 15;  
4310: panelPesados.add(tp105, gridBagConstraints);  
4311:  
4312: tp107.setColumns(5);  
4313: tp107.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4314: tp107.setToolTipText("");  
4315: tp107.setInputVerifier(intVerifier);  
4316: tp107.setMinimumSize(new java.awt.Dimension(50, 19));  
4317: tp107.setName(""); // NOI18N  
4318: tp107.setNextFocusableComponent(tp108);  
4319: gridBagConstraints = new java.awt.GridBagConstraints();  
4320: gridBagConstraints.gridx = 5;  
4321: gridBagConstraints.gridy = 15;  
4322: panelPesados.add(tp107, gridBagConstraints);  
4323:  
4324: tp109.setColumns(5);  
4325: tp109.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4326: tp109.setToolTipText("");  
4327: tp109.setInputVerifier(intVerifier);  
4328: tp109.setMinimumSize(new java.awt.Dimension(50, 19));  
4329: tp109.setName(""); // NOI18N  
4330: tp109.setNextFocusableComponent(tp110);  
4331: gridBagConstraints = new java.awt.GridBagConstraints();  
4332: gridBagConstraints.gridx = 7;  
4333: gridBagConstraints.gridy = 15;  
4334: panelPesados.add(tp109, gridBagConstraints);  
4335:  
4336: tp111.setColumns(5);  
4337: tp111.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4338: tp111.setToolTipText("");  
4339: tp111.setInputVerifier(intVerifier);  
4340: tp111.setMinimumSize(new java.awt.Dimension(50, 19));  
4341: tp111.setName(""); // NOI18N  
4342: tp111.setNextFocusableComponent(tp112);  
4343: gridBagConstraints = new java.awt.GridBagConstraints();  
4344: gridBagConstraints.gridx = 9;  
4345: gridBagConstraints.gridy = 15;  
4346: panelPesados.add(tp111, gridBagConstraints);  
4347:  
4348: lbl3C2.setBackground(new java.awt.Color(255, 255, 255));  
4349: lbl3C2.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
4350: lbl3C2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/reboques/r4.png"))); // NOI18N  
4351: lbl3C2.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
4352: lbl3C2.setFocusable(false);  
4353: lbl3C2.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
4354: lbl3C2.setMinimumSize(new java.awt.Dimension(200, 76));  
4355: lbl3C2.setOpaque(true);  
4356: gridBagConstraints = new java.awt.GridBagConstraints();  
4357: gridBagConstraints.gridx = 0;  
4358: gridBagConstraints.gridy = 16;  
4359: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
4360: panelPesados.add(lbl3C2, gridBagConstraints);  
4361:  
4362: tp113.setColumns(5);  
4363: tp113.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4364: tp113.setToolTipText("");  
4365: tp113.setInputVerifier(intVerifier);  
4366: tp113.setMinimumSize(new java.awt.Dimension(50, 19));  
4367: tp113.setName(""); // NOI18N  
4368: tp113.setNextFocusableComponent(tp114);  
4369: gridBagConstraints = new java.awt.GridBagConstraints();  
4370: gridBagConstraints.gridx = 3;  
4371: gridBagConstraints.gridy = 16;  
4372: panelPesados.add(tp113, gridBagConstraints);  
4373:  
4374: tp115.setColumns(5);  
4375: tp115.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4376: tp115.setToolTipText("");  
4377: tp115.setInputVerifier(intVerifier);  
4378: tp115.setMinimumSize(new java.awt.Dimension(50, 19));  
4379: tp115.setName(""); // NOI18N  
4380: tp115.setNextFocusableComponent(tp116);  
4381: gridBagConstraints = new java.awt.GridBagConstraints();  
4382: gridBagConstraints.gridx = 5;  
4383: gridBagConstraints.gridy = 16;  
4384: panelPesados.add(tp115, gridBagConstraints);  
4385:  
4386: tp117.setColumns(5);  
4387: tp117.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4388: tp117.setToolTipText("");  
4389: tp117.setInputVerifier(intVerifier);  
4390: tp117.setMinimumSize(new java.awt.Dimension(50, 19));  
4391: tp117.setName(""); // NOI18N  
4392: tp117.setNextFocusableComponent(tp118);  
4393: gridBagConstraints = new java.awt.GridBagConstraints();  
4394: gridBagConstraints.gridx = 7;  
4395: gridBagConstraints.gridy = 16;  
4396: panelPesados.add(tp117, gridBagConstraints);  
4397:  
4398: tp119.setColumns(5);  
4399: tp119.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4400: tp119.setToolTipText("");  
4401: tp119.setInputVerifier(intVerifier);  
4402: tp119.setMinimumSize(new java.awt.Dimension(50, 19));  
4403: tp119.setName(""); // NOI18N  
4404: tp119.setNextFocusableComponent(tp120);  
4405: gridBagConstraints = new java.awt.GridBagConstraints();  
4406: gridBagConstraints.gridx = 9;  
4407: gridBagConstraints.gridy = 16;  
4408: panelPesados.add(tp119, gridBagConstraints);  
4409:  
4410: lbl3C3.setBackground(new java.awt.Color(255, 255, 255));  
4411: lbl3C3.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
4412: lbl3C3.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/reboques/r5.png"))); // NOI18N  
4413: lbl3C3.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
4414: lbl3C3.setFocusable(false);  
4415: lbl3C3.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
4416: lbl3C3.setMinimumSize(new java.awt.Dimension(200, 76));  
4417: lbl3C3.setOpaque(true);  
4418: gridBagConstraints = new java.awt.GridBagConstraints();  
4419: gridBagConstraints.gridx = 0;  
4420: gridBagConstraints.gridy = 17;  
4421: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
4422: panelPesados.add(lbl3C3, gridBagConstraints);  
4423:  
4424: tp121.setColumns(5);  
4425: tp121.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4426: tp121.setToolTipText("");  
4427: tp121.setInputVerifier(intVerifier);  
4428: tp121.setMinimumSize(new java.awt.Dimension(50, 19));  
4429: tp121.setName(""); // NOI18N  
4430: tp121.setNextFocusableComponent(tp122);  
4431: gridBagConstraints = new java.awt.GridBagConstraints();  
4432: gridBagConstraints.gridx = 3;  
4433: gridBagConstraints.gridy = 17;  
4434: panelPesados.add(tp121, gridBagConstraints);  
4435:  
4436: tp123.setColumns(5);  
4437: tp123.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4438: tp123.setToolTipText("");  
4439: tp123.setInputVerifier(intVerifier);  
4440: tp123.setMinimumSize(new java.awt.Dimension(50, 19));  
4441: tp123.setName(""); // NOI18N  
4442: tp123.setNextFocusableComponent(tp124);  
4443: gridBagConstraints = new java.awt.GridBagConstraints();  
4444: gridBagConstraints.gridx = 5;  
4445: gridBagConstraints.gridy = 17;  
4446: panelPesados.add(tp123, gridBagConstraints);  
4447:  
4448: tp125.setColumns(5);  
4449: tp125.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4450: tp125.setToolTipText("");  
4451: tp125.setInputVerifier(intVerifier);  
4452: tp125.setMinimumSize(new java.awt.Dimension(50, 19));  
4453: tp125.setName(""); // NOI18N  
4454: tp125.setNextFocusableComponent(tp126);  
4455: gridBagConstraints = new java.awt.GridBagConstraints();  
4456: gridBagConstraints.gridx = 7;  
4457: gridBagConstraints.gridy = 17;  
4458: panelPesados.add(tp125, gridBagConstraints);  
4459:  
4460: tp127.setColumns(5);  
4461: tp127.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4462: tp127.setToolTipText("");  
4463: tp127.setInputVerifier(intVerifier);  
4464: tp127.setMinimumSize(new java.awt.Dimension(50, 19));  
4465: tp127.setName(""); // NOI18N  
4466: tp127.setNextFocusableComponent(tp128);  
4467: gridBagConstraints = new java.awt.GridBagConstraints();  
4468: gridBagConstraints.gridx = 9;  
4469: gridBagConstraints.gridy = 17;  
4470: panelPesados.add(tp127, gridBagConstraints);  
4471:  
4472: lbl3D4.setBackground(new java.awt.Color(255, 255, 255));  
4473: lbl3D4.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
4474: lbl3D4.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/reboques/r6.png"))); // NOI18N  
4475: lbl3D4.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 3, 1));  
4476: lbl3D4.setFocusable(false);  
4477: lbl3D4.setHorizontalTextPosition(javax.swing.SwingConstants.RIGHT);  
4478: lbl3D4.setMinimumSize(new java.awt.Dimension(200, 76));  
4479: lbl3D4.setOpaque(true);  
4480: gridBagConstraints = new java.awt.GridBagConstraints();  
4481: gridBagConstraints.gridx = 0;  
4482: gridBagConstraints.gridy = 18;  
4483: gridBagConstraints.fill = java.awt.GridBagConstraints.HORIZONTAL;  
4484: panelPesados.add(lbl3D4, gridBagConstraints);  
4485:  
4486: tp129.setColumns(5);  
4487: tp129.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4488: tp129.setToolTipText("");  
4489: tp129.setInputVerifier(intVerifier);  
4490: tp129.setMinimumSize(new java.awt.Dimension(50, 19));  
4491: tp129.setName(""); // NOI18N  
4492: tp129.setNextFocusableComponent(tp130);  
4493: gridBagConstraints = new java.awt.GridBagConstraints();  
4494: gridBagConstraints.gridx = 3;  
4495: gridBagConstraints.gridy = 18;  
4496: panelPesados.add(tp129, gridBagConstraints);  
4497:  
4498: tp131.setColumns(5);  
4499: tp131.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4500: tp131.setToolTipText("");  
4501: tp131.setInputVerifier(intVerifier);  
4502: tp131.setMinimumSize(new java.awt.Dimension(50, 19));  
4503: tp131.setName(""); // NOI18N  
4504: tp131.setNextFocusableComponent(tp132);  
4505: gridBagConstraints = new java.awt.GridBagConstraints();  
4506: gridBagConstraints.gridx = 5;  
4507: gridBagConstraints.gridy = 18;  
4508: panelPesados.add(tp131, gridBagConstraints);  
4509:  
4510: tp133.setColumns(5);  
4511: tp133.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4512: tp133.setToolTipText("");  
4513: tp133.setInputVerifier(intVerifier);  
4514: tp133.setMinimumSize(new java.awt.Dimension(50, 19));  
4515: tp133.setName(""); // NOI18N  
4516: tp133.setNextFocusableComponent(tp134);  
4517: gridBagConstraints = new java.awt.GridBagConstraints();  
4518: gridBagConstraints.gridx = 7;  
4519: gridBagConstraints.gridy = 18;  
4520: panelPesados.add(tp133, gridBagConstraints);  
4521:  
4522: tp135.setColumns(5);  
4523: tp135.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4524: tp135.setToolTipText("");  
4525: tp135.setInputVerifier(intVerifier);  
4526: tp135.setMinimumSize(new java.awt.Dimension(50, 19));  
4527: tp135.setName(""); // NOI18N  
4528: gridBagConstraints = new java.awt.GridBagConstraints();  
4529: gridBagConstraints.gridx = 9;  
4530: gridBagConstraints.gridy = 18;  
4531: panelPesados.add(tp135, gridBagConstraints);  
4532:  
4533: jLabel17.setText("2S1");  
4534: gridBagConstraints = new java.awt.GridBagConstraints();  
4535: gridBagConstraints.gridx = 1;  
4536: gridBagConstraints.gridy = 2;  
4537: panelPesados.add(jLabel17, gridBagConstraints);  
4538:  
4539: jLabel18.setText("2S2");  
4540: gridBagConstraints = new java.awt.GridBagConstraints();  
4541: gridBagConstraints.gridx = 1;  
4542: gridBagConstraints.gridy = 3;  
4543: panelPesados.add(jLabel18, gridBagConstraints);  
4544:  
4545: jLabel19.setText("2S3");  
4546: gridBagConstraints = new java.awt.GridBagConstraints();  
4547: gridBagConstraints.gridx = 1;  
4548: gridBagConstraints.gridy = 4;  
4549: panelPesados.add(jLabel19, gridBagConstraints);  
4550:  
4551: jLabel20.setText("3S1");  
4552: gridBagConstraints = new java.awt.GridBagConstraints();  
4553: gridBagConstraints.gridx = 1;  
4554: gridBagConstraints.gridy = 5;  
4555: panelPesados.add(jLabel20, gridBagConstraints);  
4556:  
4557: jLabel21.setText("3S2");  
4558: gridBagConstraints = new java.awt.GridBagConstraints();  
4559: gridBagConstraints.gridx = 1;  
4560: gridBagConstraints.gridy = 6;  
4561: panelPesados.add(jLabel21, gridBagConstraints);  
4562:  
4563: jLabel22.setText("3S3");  
4564: gridBagConstraints = new java.awt.GridBagConstraints();  
4565: gridBagConstraints.gridx = 1;  
4566: gridBagConstraints.gridy = 7;  
4567: panelPesados.add(jLabel22, gridBagConstraints);  
4568:  
4569: jLabel23.setText("3T4");  
4570: gridBagConstraints = new java.awt.GridBagConstraints();  
4571: gridBagConstraints.gridx = 1;  
4572: gridBagConstraints.gridy = 8;  
4573: panelPesados.add(jLabel23, gridBagConstraints);  
4574:  
4575: jLabel24.setText("3T6");  
4576: gridBagConstraints = new java.awt.GridBagConstraints();  
4577: gridBagConstraints.gridx = 1;  
4578: gridBagConstraints.gridy = 9;  
4579: panelPesados.add(jLabel24, gridBagConstraints);  
4580:  
4581: jLabel25.setText("3R6");  
4582: gridBagConstraints = new java.awt.GridBagConstraints();  
4583: gridBagConstraints.gridx = 1;  
4584: gridBagConstraints.gridy = 10;  
4585: panelPesados.add(jLabel25, gridBagConstraints);  
4586:  
4587: jLabel26.setText("3V5");  
4588: gridBagConstraints = new java.awt.GridBagConstraints();  
4589: gridBagConstraints.gridx = 1;  
4590: gridBagConstraints.gridy = 11;  
4591: panelPesados.add(jLabel26, gridBagConstraints);  
4592:  
4593: jLabel27.setText("3M6");  
4594: gridBagConstraints = new java.awt.GridBagConstraints();  
4595: gridBagConstraints.gridx = 1;  
4596: gridBagConstraints.gridy = 12;  
4597: panelPesados.add(jLabel27, gridBagConstraints);  
4598:  
4599: jLabel28.setText("3Q4");  
4600: gridBagConstraints = new java.awt.GridBagConstraints();  
4601: gridBagConstraints.gridx = 1;  
4602: gridBagConstraints.gridy = 13;  
4603: panelPesados.add(jLabel28, gridBagConstraints);  
4604:  
4605: jLabel29.setText("2C2");  
4606: gridBagConstraints = new java.awt.GridBagConstraints();  
4607: gridBagConstraints.gridx = 1;  
4608: gridBagConstraints.gridy = 14;  
4609: panelPesados.add(jLabel29, gridBagConstraints);  
4610:  
4611: jLabel30.setText("2C3");  
4612: gridBagConstraints = new java.awt.GridBagConstraints();  
4613: gridBagConstraints.gridx = 1;  
4614: gridBagConstraints.gridy = 15;  
4615: panelPesados.add(jLabel30, gridBagConstraints);  
4616:  
4617: jLabel31.setText("3C2");  
4618: gridBagConstraints = new java.awt.GridBagConstraints();  
4619: gridBagConstraints.gridx = 1;  
4620: gridBagConstraints.gridy = 16;  
4621: panelPesados.add(jLabel31, gridBagConstraints);  
4622:  
4623: jLabel32.setText("3C3");  
4624: gridBagConstraints = new java.awt.GridBagConstraints();  
4625: gridBagConstraints.gridx = 1;  
4626: gridBagConstraints.gridy = 17;  
4627: panelPesados.add(jLabel32, gridBagConstraints);  
4628:  
4629: jLabel33.setText("3D4");  
4630: gridBagConstraints = new java.awt.GridBagConstraints();  
4631: gridBagConstraints.gridx = 1;  
4632: gridBagConstraints.gridy = 18;  
4633: panelPesados.add(jLabel33, gridBagConstraints);  
4634:  
4635: jPanel51.setLayout(new java.awt.GridBagLayout());  
4636:  
4637: tp0.setColumns(5);  
4638: tp0.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4639: tp0.setToolTipText("");  
4640: tp0.setInputVerifier(intVerifier);  
4641: tp0.setMinimumSize(new java.awt.Dimension(50, 19));  
4642: tp0.setName(""); // NOI18N  
4643: tp0.setNextFocusableComponent(tp1);  
4644: gridBagConstraints = new java.awt.GridBagConstraints();  
4645: gridBagConstraints.gridx = 0;  
4646: gridBagConstraints.gridy = 0;  
4647: gridBagConstraints.weightx = 1.0;  
4648: jPanel51.add(tp0, gridBagConstraints);  
4649:  
4650: gridBagConstraints = new java.awt.GridBagConstraints();  
4651: gridBagConstraints.gridx = 2;  
4652: gridBagConstraints.gridy = 2;  
4653: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4654: gridBagConstraints.weightx = 1.0;  
4655: panelPesados.add(jPanel51, gridBagConstraints);  
4656:  
4657: jPanel52.setLayout(new java.awt.GridBagLayout());  
4658:  
4659: tp8.setColumns(5);  
4660: tp8.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4661: tp8.setToolTipText("");  
4662: tp8.setInputVerifier(intVerifier);  
4663: tp8.setMinimumSize(new java.awt.Dimension(50, 19));  
4664: tp8.setName(""); // NOI18N  
4665: tp8.setNextFocusableComponent(tp9);  
4666: gridBagConstraints = new java.awt.GridBagConstraints();  
4667: gridBagConstraints.gridx = 0;  
4668: gridBagConstraints.gridy = 0;  
4669: gridBagConstraints.weightx = 1.0;  
4670: jPanel52.add(tp8, gridBagConstraints);  
4671:  
4672: gridBagConstraints = new java.awt.GridBagConstraints();  
4673: gridBagConstraints.gridx = 2;  
4674: gridBagConstraints.gridy = 3;  
4675: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4676: gridBagConstraints.weightx = 1.0;  
4677: panelPesados.add(jPanel52, gridBagConstraints);  
4678:  
4679: jPanel53.setLayout(new java.awt.GridBagLayout());  
4680:  
4681: tp16.setColumns(5);  
4682: tp16.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4683: tp16.setToolTipText("");  
4684: tp16.setInputVerifier(intVerifier);  
4685: tp16.setMinimumSize(new java.awt.Dimension(50, 19));  
4686: tp16.setName(""); // NOI18N  
4687: tp16.setNextFocusableComponent(tp17);  
4688: gridBagConstraints = new java.awt.GridBagConstraints();  
4689: gridBagConstraints.gridx = 0;  
4690: gridBagConstraints.gridy = 0;  
4691: gridBagConstraints.weightx = 1.0;  
4692: jPanel53.add(tp16, gridBagConstraints);  
4693:  
4694: gridBagConstraints = new java.awt.GridBagConstraints();  
4695: gridBagConstraints.gridx = 2;  
4696: gridBagConstraints.gridy = 4;  
4697: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4698: gridBagConstraints.weightx = 1.0;  
4699: panelPesados.add(jPanel53, gridBagConstraints);  
4700:  
4701: jPanel54.setLayout(new java.awt.GridBagLayout());  
4702:  
4703: tp24.setColumns(5);  
4704: tp24.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4705: tp24.setToolTipText("");  
4706: tp24.setInputVerifier(intVerifier);  
4707: tp24.setMinimumSize(new java.awt.Dimension(50, 19));  
4708: tp24.setName(""); // NOI18N  
4709: tp24.setNextFocusableComponent(tp25);  
4710: gridBagConstraints = new java.awt.GridBagConstraints();  
4711: gridBagConstraints.gridx = 0;  
4712: gridBagConstraints.gridy = 0;  
4713: gridBagConstraints.weightx = 1.0;  
4714: jPanel54.add(tp24, gridBagConstraints);  
4715:  
4716: gridBagConstraints = new java.awt.GridBagConstraints();  
4717: gridBagConstraints.gridx = 2;  
4718: gridBagConstraints.gridy = 5;  
4719: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4720: gridBagConstraints.weightx = 1.0;  
4721: panelPesados.add(jPanel54, gridBagConstraints);  
4722:  
4723: jPanel55.setLayout(new java.awt.GridBagLayout());  
4724:  
4725: tp32.setColumns(5);  
4726: tp32.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4727: tp32.setToolTipText("");  
4728: tp32.setInputVerifier(intVerifier);  
4729: tp32.setMinimumSize(new java.awt.Dimension(50, 19));  
4730: tp32.setName(""); // NOI18N  
4731: tp32.setNextFocusableComponent(tp33);  
4732: gridBagConstraints = new java.awt.GridBagConstraints();  
4733: gridBagConstraints.gridx = 0;  
4734: gridBagConstraints.gridy = 0;  
4735: jPanel55.add(tp32, gridBagConstraints);  
4736:  
4737: gridBagConstraints = new java.awt.GridBagConstraints();  
4738: gridBagConstraints.gridx = 2;  
4739: gridBagConstraints.gridy = 6;  
4740: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4741: gridBagConstraints.weightx = 1.0;  
4742: panelPesados.add(jPanel55, gridBagConstraints);  
4743:  
4744: jPanel56.setLayout(new java.awt.GridBagLayout());  
4745:  
4746: tp40.setColumns(5);  
4747: tp40.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4748: tp40.setToolTipText("");  
4749: tp40.setInputVerifier(intVerifier);  
4750: tp40.setMinimumSize(new java.awt.Dimension(50, 19));  
4751: tp40.setName(""); // NOI18N  
4752: tp40.setNextFocusableComponent(tp41);  
4753: gridBagConstraints = new java.awt.GridBagConstraints();  
4754: gridBagConstraints.gridx = 0;  
4755: gridBagConstraints.gridy = 0;  
4756: jPanel56.add(tp40, gridBagConstraints);  
4757:  
4758: gridBagConstraints = new java.awt.GridBagConstraints();  
4759: gridBagConstraints.gridx = 2;  
4760: gridBagConstraints.gridy = 7;  
4761: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4762: gridBagConstraints.weightx = 1.0;  
4763: panelPesados.add(jPanel56, gridBagConstraints);  
4764:  
4765: jPanel57.setLayout(new java.awt.GridBagLayout());  
4766:  
4767: tp48.setColumns(5);  
4768: tp48.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4769: tp48.setToolTipText("");  
4770: tp48.setInputVerifier(intVerifier);  
4771: tp48.setMinimumSize(new java.awt.Dimension(50, 19));  
4772: tp48.setName(""); // NOI18N  
4773: tp48.setNextFocusableComponent(tp49);  
4774: gridBagConstraints = new java.awt.GridBagConstraints();  
4775: gridBagConstraints.gridx = 0;  
4776: gridBagConstraints.gridy = 0;  
4777: jPanel57.add(tp48, gridBagConstraints);  
4778:  
4779: gridBagConstraints = new java.awt.GridBagConstraints();  
4780: gridBagConstraints.gridx = 2;  
4781: gridBagConstraints.gridy = 8;  
4782: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4783: gridBagConstraints.weightx = 1.0;  
4784: panelPesados.add(jPanel57, gridBagConstraints);  
4785:  
4786: jPanel58.setLayout(new java.awt.GridBagLayout());  
4787:  
4788: tp56.setColumns(5);  
4789: tp56.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4790: tp56.setToolTipText("");  
4791: tp56.setInputVerifier(intVerifier);  
4792: tp56.setMinimumSize(new java.awt.Dimension(50, 19));  
4793: tp56.setName(""); // NOI18N  
4794: tp56.setNextFocusableComponent(tp57);  
4795: gridBagConstraints = new java.awt.GridBagConstraints();  
4796: gridBagConstraints.gridx = 0;  
4797: gridBagConstraints.gridy = 0;  
4798: jPanel58.add(tp56, gridBagConstraints);  
4799:  
4800: gridBagConstraints = new java.awt.GridBagConstraints();  
4801: gridBagConstraints.gridx = 2;  
4802: gridBagConstraints.gridy = 9;  
4803: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4804: gridBagConstraints.weightx = 1.0;  
4805: panelPesados.add(jPanel58, gridBagConstraints);  
4806:  
4807: jPanel59.setLayout(new java.awt.GridBagLayout());  
4808:  
4809: tp64.setColumns(5);  
4810: tp64.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4811: tp64.setToolTipText("");  
4812: tp64.setInputVerifier(intVerifier);  
4813: tp64.setMinimumSize(new java.awt.Dimension(50, 19));  
4814: tp64.setName(""); // NOI18N  
4815: tp64.setNextFocusableComponent(tp65);  
4816: gridBagConstraints = new java.awt.GridBagConstraints();  
4817: gridBagConstraints.gridx = 0;  
4818: gridBagConstraints.gridy = 0;  
4819: jPanel59.add(tp64, gridBagConstraints);  
4820:  
4821: gridBagConstraints = new java.awt.GridBagConstraints();  
4822: gridBagConstraints.gridx = 2;  
4823: gridBagConstraints.gridy = 10;  
4824: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4825: gridBagConstraints.weightx = 1.0;  
4826: panelPesados.add(jPanel59, gridBagConstraints);  
4827:  
4828: jPanel60.setLayout(new java.awt.GridBagLayout());  
4829:  
4830: tp72.setColumns(5);  
4831: tp72.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4832: tp72.setToolTipText("");  
4833: tp72.setInputVerifier(intVerifier);  
4834: tp72.setMinimumSize(new java.awt.Dimension(50, 19));  
4835: tp72.setName(""); // NOI18N  
4836: tp72.setNextFocusableComponent(tp73);  
4837: gridBagConstraints = new java.awt.GridBagConstraints();  
4838: gridBagConstraints.gridx = 0;  
4839: gridBagConstraints.gridy = 0;  
4840: jPanel60.add(tp72, gridBagConstraints);  
4841:  
4842: gridBagConstraints = new java.awt.GridBagConstraints();  
4843: gridBagConstraints.gridx = 2;  
4844: gridBagConstraints.gridy = 11;  
4845: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4846: gridBagConstraints.weightx = 1.0;  
4847: panelPesados.add(jPanel60, gridBagConstraints);  
4848:  
4849: jPanel61.setLayout(new java.awt.GridBagLayout());  
4850:  
4851: tp80.setColumns(5);  
4852: tp80.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4853: tp80.setToolTipText("");  
4854: tp80.setInputVerifier(intVerifier);  
4855: tp80.setMinimumSize(new java.awt.Dimension(50, 19));  
4856: tp80.setName(""); // NOI18N  
4857: tp80.setNextFocusableComponent(tp81);  
4858: gridBagConstraints = new java.awt.GridBagConstraints();  
4859: gridBagConstraints.gridx = 0;  
4860: gridBagConstraints.gridy = 0;  
4861: jPanel61.add(tp80, gridBagConstraints);  
4862:  
4863: gridBagConstraints = new java.awt.GridBagConstraints();  
4864: gridBagConstraints.gridx = 2;  
4865: gridBagConstraints.gridy = 12;  
4866: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4867: gridBagConstraints.weightx = 1.0;  
4868: panelPesados.add(jPanel61, gridBagConstraints);  
4869:  
4870: jPanel62.setLayout(new java.awt.GridBagLayout());  
4871:  
4872: tp88.setColumns(5);  
4873: tp88.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4874: tp88.setToolTipText("");  
4875: tp88.setInputVerifier(intVerifier);  
4876: tp88.setMinimumSize(new java.awt.Dimension(50, 19));  
4877: tp88.setName(""); // NOI18N  
4878: tp88.setNextFocusableComponent(tp89);  
4879: gridBagConstraints = new java.awt.GridBagConstraints();  
4880: gridBagConstraints.gridx = 0;  
4881: gridBagConstraints.gridy = 0;  
4882: jPanel62.add(tp88, gridBagConstraints);  
4883:  
4884: gridBagConstraints = new java.awt.GridBagConstraints();  
4885: gridBagConstraints.gridx = 2;  
4886: gridBagConstraints.gridy = 13;  
4887: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4888: gridBagConstraints.weightx = 1.0;  
4889: panelPesados.add(jPanel62, gridBagConstraints);  
4890:  
4891: jPanel63.setLayout(new java.awt.GridBagLayout());  
4892:  
4893: tp96.setColumns(5);  
4894: tp96.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4895: tp96.setToolTipText("");  
4896: tp96.setInputVerifier(intVerifier);  
4897: tp96.setMinimumSize(new java.awt.Dimension(50, 19));  
4898: tp96.setName(""); // NOI18N  
4899: tp96.setNextFocusableComponent(tp97);  
4900: gridBagConstraints = new java.awt.GridBagConstraints();  
4901: gridBagConstraints.gridx = 0;  
4902: gridBagConstraints.gridy = 0;  
4903: jPanel63.add(tp96, gridBagConstraints);  
4904:  
4905: gridBagConstraints = new java.awt.GridBagConstraints();  
4906: gridBagConstraints.gridx = 2;  
4907: gridBagConstraints.gridy = 14;  
4908: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4909: gridBagConstraints.weightx = 1.0;  
4910: panelPesados.add(jPanel63, gridBagConstraints);  
4911:  
4912: jPanel64.setLayout(new java.awt.GridBagLayout());  
4913:  
4914: tp104.setColumns(5);  
4915: tp104.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4916: tp104.setToolTipText("");  
4917: tp104.setInputVerifier(intVerifier);  
4918: tp104.setMinimumSize(new java.awt.Dimension(50, 19));  
4919: tp104.setName(""); // NOI18N  
4920: tp104.setNextFocusableComponent(tp105);  
4921: gridBagConstraints = new java.awt.GridBagConstraints();  
4922: gridBagConstraints.gridx = 0;  
4923: gridBagConstraints.gridy = 0;  
4924: jPanel64.add(tp104, gridBagConstraints);  
4925:  
4926: gridBagConstraints = new java.awt.GridBagConstraints();  
4927: gridBagConstraints.gridx = 2;  
4928: gridBagConstraints.gridy = 15;  
4929: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4930: gridBagConstraints.weightx = 1.0;  
4931: panelPesados.add(jPanel64, gridBagConstraints);  
4932:  
4933: jPanel65.setLayout(new java.awt.GridBagLayout());  
4934:  
4935: tp112.setColumns(5);  
4936: tp112.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4937: tp112.setToolTipText("");  
4938: tp112.setInputVerifier(intVerifier);  
4939: tp112.setMinimumSize(new java.awt.Dimension(50, 19));  
4940: tp112.setName(""); // NOI18N  
4941: tp112.setNextFocusableComponent(tp113);  
4942: gridBagConstraints = new java.awt.GridBagConstraints();  
4943: gridBagConstraints.gridx = 0;  
4944: gridBagConstraints.gridy = 0;  
4945: jPanel65.add(tp112, gridBagConstraints);  
4946:  
4947: gridBagConstraints = new java.awt.GridBagConstraints();  
4948: gridBagConstraints.gridx = 2;  
4949: gridBagConstraints.gridy = 16;  
4950: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4951: gridBagConstraints.weightx = 1.0;  
4952: panelPesados.add(jPanel65, gridBagConstraints);  
4953:  
4954: jPanel66.setLayout(new java.awt.GridBagLayout());  
4955:  
4956: tp120.setColumns(5);  
4957: tp120.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4958: tp120.setToolTipText("");  
4959: tp120.setInputVerifier(intVerifier);  
4960: tp120.setMinimumSize(new java.awt.Dimension(50, 19));  
4961: tp120.setName(""); // NOI18N  
4962: tp120.setNextFocusableComponent(tp121);  
4963: gridBagConstraints = new java.awt.GridBagConstraints();  
4964: gridBagConstraints.gridx = 0;  
4965: gridBagConstraints.gridy = 0;  
4966: jPanel66.add(tp120, gridBagConstraints);  
4967:  
4968: gridBagConstraints = new java.awt.GridBagConstraints();  
4969: gridBagConstraints.gridx = 2;  
4970: gridBagConstraints.gridy = 17;  
4971: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4972: gridBagConstraints.weightx = 1.0;  
4973: panelPesados.add(jPanel66, gridBagConstraints);  
4974:  
4975: jPanel67.setLayout(new java.awt.GridBagLayout());  
4976:  
4977: tp128.setColumns(5);  
4978: tp128.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
4979: tp128.setToolTipText("");  
4980: tp128.setInputVerifier(intVerifier);  
4981: tp128.setMinimumSize(new java.awt.Dimension(50, 19));  
4982: tp128.setName(""); // NOI18N  
4983: tp128.setNextFocusableComponent(tp129);  
4984: gridBagConstraints = new java.awt.GridBagConstraints();  
4985: gridBagConstraints.gridx = 0;  
4986: gridBagConstraints.gridy = 0;  
4987: jPanel67.add(tp128, gridBagConstraints);  
4988:  
4989: gridBagConstraints = new java.awt.GridBagConstraints();  
4990: gridBagConstraints.gridx = 2;  
4991: gridBagConstraints.gridy = 18;  
4992: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
4993: gridBagConstraints.weightx = 1.0;  
4994: panelPesados.add(jPanel67, gridBagConstraints);  
4995:  
4996: jPanel68.setLayout(new java.awt.GridBagLayout());  
4997:  
4998: tp2.setColumns(5);  
4999: tp2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5000: tp2.setToolTipText("");  
5001: tp2.setInputVerifier(intVerifier);  
5002: tp2.setMinimumSize(new java.awt.Dimension(50, 19));  
5003: tp2.setName(""); // NOI18N  
5004: tp2.setNextFocusableComponent(tp3);  
5005: gridBagConstraints = new java.awt.GridBagConstraints();  
5006: gridBagConstraints.gridx = 0;  
5007: gridBagConstraints.gridy = 0;  
5008: gridBagConstraints.weightx = 1.0;  
5009: jPanel68.add(tp2, gridBagConstraints);  
5010:  
5011: gridBagConstraints = new java.awt.GridBagConstraints();  
5012: gridBagConstraints.gridx = 4;  
5013: gridBagConstraints.gridy = 2;  
5014: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5015: gridBagConstraints.weightx = 1.0;  
5016: panelPesados.add(jPanel68, gridBagConstraints);  
5017:  
5018: jPanel69.setLayout(new java.awt.GridBagLayout());  
5019:  
5020: tp10.setColumns(5);  
5021: tp10.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5022: tp10.setToolTipText("");  
5023: tp10.setInputVerifier(intVerifier);  
5024: tp10.setMinimumSize(new java.awt.Dimension(50, 19));  
5025: tp10.setName(""); // NOI18N  
5026: tp10.setNextFocusableComponent(tp11);  
5027: gridBagConstraints = new java.awt.GridBagConstraints();  
5028: gridBagConstraints.gridx = 0;  
5029: gridBagConstraints.gridy = 0;  
5030: gridBagConstraints.weightx = 1.0;  
5031: jPanel69.add(tp10, gridBagConstraints);  
5032:  
5033: gridBagConstraints = new java.awt.GridBagConstraints();  
5034: gridBagConstraints.gridx = 4;  
5035: gridBagConstraints.gridy = 3;  
5036: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5037: gridBagConstraints.weightx = 1.0;  
5038: panelPesados.add(jPanel69, gridBagConstraints);  
5039:  
5040: jPanel70.setLayout(new java.awt.GridBagLayout());  
5041:  
5042: tp18.setColumns(5);  
5043: tp18.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5044: tp18.setToolTipText("");  
5045: tp18.setInputVerifier(intVerifier);  
5046: tp18.setMinimumSize(new java.awt.Dimension(50, 19));  
5047: tp18.setName(""); // NOI18N  
5048: tp18.setNextFocusableComponent(tp19);  
5049: gridBagConstraints = new java.awt.GridBagConstraints();  
5050: gridBagConstraints.gridx = 0;  
5051: gridBagConstraints.gridy = 0;  
5052: gridBagConstraints.weightx = 1.0;  
5053: jPanel70.add(tp18, gridBagConstraints);  
5054:  
5055: gridBagConstraints = new java.awt.GridBagConstraints();  
5056: gridBagConstraints.gridx = 4;  
5057: gridBagConstraints.gridy = 4;  
5058: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5059: gridBagConstraints.weightx = 1.0;  
5060: panelPesados.add(jPanel70, gridBagConstraints);  
5061:  
5062: jPanel71.setLayout(new java.awt.GridBagLayout());  
5063:  
5064: tp26.setColumns(5);  
5065: tp26.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5066: tp26.setToolTipText("");  
5067: tp26.setInputVerifier(intVerifier);  
5068: tp26.setMinimumSize(new java.awt.Dimension(50, 19));  
5069: tp26.setName(""); // NOI18N  
5070: tp26.setNextFocusableComponent(tp27);  
5071: gridBagConstraints = new java.awt.GridBagConstraints();  
5072: gridBagConstraints.gridx = 0;  
5073: gridBagConstraints.gridy = 0;  
5074: gridBagConstraints.weightx = 1.0;  
5075: jPanel71.add(tp26, gridBagConstraints);  
5076:  
5077: gridBagConstraints = new java.awt.GridBagConstraints();  
5078: gridBagConstraints.gridx = 4;  
5079: gridBagConstraints.gridy = 5;  
5080: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5081: gridBagConstraints.weightx = 1.0;  
5082: panelPesados.add(jPanel71, gridBagConstraints);  
5083:  
5084: jPanel72.setLayout(new java.awt.GridBagLayout());  
5085:  
5086: tp34.setColumns(5);  
5087: tp34.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5088: tp34.setToolTipText("");  
5089: tp34.setInputVerifier(intVerifier);  
5090: tp34.setMinimumSize(new java.awt.Dimension(50, 19));  
5091: tp34.setName(""); // NOI18N  
5092: tp34.setNextFocusableComponent(tp35);  
5093: gridBagConstraints = new java.awt.GridBagConstraints();  
5094: gridBagConstraints.gridx = 0;  
5095: gridBagConstraints.gridy = 0;  
5096: jPanel72.add(tp34, gridBagConstraints);  
5097:  
5098: gridBagConstraints = new java.awt.GridBagConstraints();  
5099: gridBagConstraints.gridx = 4;  
5100: gridBagConstraints.gridy = 6;  
5101: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5102: gridBagConstraints.weightx = 1.0;  
5103: panelPesados.add(jPanel72, gridBagConstraints);  
5104:  
5105: jPanel73.setLayout(new java.awt.GridBagLayout());  
5106:  
5107: tp42.setColumns(5);  
5108: tp42.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5109: tp42.setToolTipText("");  
5110: tp42.setInputVerifier(intVerifier);  
5111: tp42.setMinimumSize(new java.awt.Dimension(50, 19));  
5112: tp42.setName(""); // NOI18N  
5113: tp42.setNextFocusableComponent(tp43);  
5114: gridBagConstraints = new java.awt.GridBagConstraints();  
5115: gridBagConstraints.gridx = 0;  
5116: gridBagConstraints.gridy = 0;  
5117: jPanel73.add(tp42, gridBagConstraints);  
5118:  
5119: gridBagConstraints = new java.awt.GridBagConstraints();  
5120: gridBagConstraints.gridx = 4;  
5121: gridBagConstraints.gridy = 7;  
5122: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5123: gridBagConstraints.weightx = 1.0;  
5124: panelPesados.add(jPanel73, gridBagConstraints);  
5125:  
5126: jPanel74.setLayout(new java.awt.GridBagLayout());  
5127:  
5128: tp50.setColumns(5);  
5129: tp50.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5130: tp50.setToolTipText("");  
5131: tp50.setInputVerifier(intVerifier);  
5132: tp50.setMinimumSize(new java.awt.Dimension(50, 19));  
5133: tp50.setName(""); // NOI18N  
5134: tp50.setNextFocusableComponent(tp51);  
5135: gridBagConstraints = new java.awt.GridBagConstraints();  
5136: gridBagConstraints.gridx = 0;  
5137: gridBagConstraints.gridy = 0;  
5138: jPanel74.add(tp50, gridBagConstraints);  
5139:  
5140: gridBagConstraints = new java.awt.GridBagConstraints();  
5141: gridBagConstraints.gridx = 4;  
5142: gridBagConstraints.gridy = 8;  
5143: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5144: gridBagConstraints.weightx = 1.0;  
5145: panelPesados.add(jPanel74, gridBagConstraints);  
5146:  
5147: jPanel75.setLayout(new java.awt.GridBagLayout());  
5148:  
5149: tp58.setColumns(5);  
5150: tp58.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5151: tp58.setToolTipText("");  
5152: tp58.setInputVerifier(intVerifier);  
5153: tp58.setMinimumSize(new java.awt.Dimension(50, 19));  
5154: tp58.setName(""); // NOI18N  
5155: tp58.setNextFocusableComponent(tp59);  
5156: gridBagConstraints = new java.awt.GridBagConstraints();  
5157: gridBagConstraints.gridx = 0;  
5158: gridBagConstraints.gridy = 0;  
5159: jPanel75.add(tp58, gridBagConstraints);  
5160:  
5161: gridBagConstraints = new java.awt.GridBagConstraints();  
5162: gridBagConstraints.gridx = 4;  
5163: gridBagConstraints.gridy = 9;  
5164: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5165: gridBagConstraints.weightx = 1.0;  
5166: panelPesados.add(jPanel75, gridBagConstraints);  
5167:  
5168: jPanel76.setLayout(new java.awt.GridBagLayout());  
5169:  
5170: tp66.setColumns(5);  
5171: tp66.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5172: tp66.setToolTipText("");  
5173: tp66.setInputVerifier(intVerifier);  
5174: tp66.setMinimumSize(new java.awt.Dimension(50, 19));  
5175: tp66.setName(""); // NOI18N  
5176: tp66.setNextFocusableComponent(tp67);  
5177: gridBagConstraints = new java.awt.GridBagConstraints();  
5178: gridBagConstraints.gridx = 0;  
5179: gridBagConstraints.gridy = 0;  
5180: jPanel76.add(tp66, gridBagConstraints);  
5181:  
5182: gridBagConstraints = new java.awt.GridBagConstraints();  
5183: gridBagConstraints.gridx = 4;  
5184: gridBagConstraints.gridy = 10;  
5185: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5186: gridBagConstraints.weightx = 1.0;  
5187: panelPesados.add(jPanel76, gridBagConstraints);  
5188:  
5189: jPanel77.setLayout(new java.awt.GridBagLayout());  
5190:  
5191: tp74.setColumns(5);  
5192: tp74.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5193: tp74.setToolTipText("");  
5194: tp74.setInputVerifier(intVerifier);  
5195: tp74.setMinimumSize(new java.awt.Dimension(50, 19));  
5196: tp74.setName(""); // NOI18N  
5197: tp74.setNextFocusableComponent(tp75);  
5198: gridBagConstraints = new java.awt.GridBagConstraints();  
5199: gridBagConstraints.gridx = 0;  
5200: gridBagConstraints.gridy = 0;  
5201: jPanel77.add(tp74, gridBagConstraints);  
5202:  
5203: gridBagConstraints = new java.awt.GridBagConstraints();  
5204: gridBagConstraints.gridx = 4;  
5205: gridBagConstraints.gridy = 11;  
5206: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5207: gridBagConstraints.weightx = 1.0;  
5208: panelPesados.add(jPanel77, gridBagConstraints);  
5209:  
5210: jPanel78.setLayout(new java.awt.GridBagLayout());  
5211:  
5212: tp82.setColumns(5);  
5213: tp82.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5214: tp82.setToolTipText("");  
5215: tp82.setInputVerifier(intVerifier);  
5216: tp82.setMinimumSize(new java.awt.Dimension(50, 19));  
5217: tp82.setName(""); // NOI18N  
5218: tp82.setNextFocusableComponent(tp83);  
5219: gridBagConstraints = new java.awt.GridBagConstraints();  
5220: gridBagConstraints.gridx = 0;  
5221: gridBagConstraints.gridy = 0;  
5222: jPanel78.add(tp82, gridBagConstraints);  
5223:  
5224: gridBagConstraints = new java.awt.GridBagConstraints();  
5225: gridBagConstraints.gridx = 4;  
5226: gridBagConstraints.gridy = 12;  
5227: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5228: gridBagConstraints.weightx = 1.0;  
5229: panelPesados.add(jPanel78, gridBagConstraints);  
5230:  
5231: jPanel79.setLayout(new java.awt.GridBagLayout());  
5232:  
5233: tp90.setColumns(5);  
5234: tp90.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5235: tp90.setToolTipText("");  
5236: tp90.setInputVerifier(intVerifier);  
5237: tp90.setMinimumSize(new java.awt.Dimension(50, 19));  
5238: tp90.setName(""); // NOI18N  
5239: tp90.setNextFocusableComponent(tp91);  
5240: gridBagConstraints = new java.awt.GridBagConstraints();  
5241: gridBagConstraints.gridx = 0;  
5242: gridBagConstraints.gridy = 0;  
5243: jPanel79.add(tp90, gridBagConstraints);  
5244:  
5245: gridBagConstraints = new java.awt.GridBagConstraints();  
5246: gridBagConstraints.gridx = 4;  
5247: gridBagConstraints.gridy = 13;  
5248: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5249: gridBagConstraints.weightx = 1.0;  
5250: panelPesados.add(jPanel79, gridBagConstraints);  
5251:  
5252: jPanel80.setLayout(new java.awt.GridBagLayout());  
5253:  
5254: tp98.setColumns(5);  
5255: tp98.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5256: tp98.setToolTipText("");  
5257: tp98.setInputVerifier(intVerifier);  
5258: tp98.setMinimumSize(new java.awt.Dimension(50, 19));  
5259: tp98.setName(""); // NOI18N  
5260: tp98.setNextFocusableComponent(tp99);  
5261: gridBagConstraints = new java.awt.GridBagConstraints();  
5262: gridBagConstraints.gridx = 0;  
5263: gridBagConstraints.gridy = 0;  
5264: jPanel80.add(tp98, gridBagConstraints);  
5265:  
5266: gridBagConstraints = new java.awt.GridBagConstraints();  
5267: gridBagConstraints.gridx = 4;  
5268: gridBagConstraints.gridy = 14;  
5269: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5270: gridBagConstraints.weightx = 1.0;  
5271: panelPesados.add(jPanel80, gridBagConstraints);  
5272:  
5273: jPanel81.setLayout(new java.awt.GridBagLayout());  
5274:  
5275: tp106.setColumns(5);  
5276: tp106.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5277: tp106.setToolTipText("");  
5278: tp106.setInputVerifier(intVerifier);  
5279: tp106.setMinimumSize(new java.awt.Dimension(50, 19));  
5280: tp106.setName(""); // NOI18N  
5281: tp106.setNextFocusableComponent(tp107);  
5282: gridBagConstraints = new java.awt.GridBagConstraints();  
5283: gridBagConstraints.gridx = 0;  
5284: gridBagConstraints.gridy = 0;  
5285: jPanel81.add(tp106, gridBagConstraints);  
5286:  
5287: gridBagConstraints = new java.awt.GridBagConstraints();  
5288: gridBagConstraints.gridx = 4;  
5289: gridBagConstraints.gridy = 15;  
5290: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5291: gridBagConstraints.weightx = 1.0;  
5292: panelPesados.add(jPanel81, gridBagConstraints);  
5293:  
5294: jPanel82.setLayout(new java.awt.GridBagLayout());  
5295:  
5296: tp114.setColumns(5);  
5297: tp114.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5298: tp114.setToolTipText("");  
5299: tp114.setInputVerifier(intVerifier);  
5300: tp114.setMinimumSize(new java.awt.Dimension(50, 19));  
5301: tp114.setName(""); // NOI18N  
5302: tp114.setNextFocusableComponent(tp115);  
5303: gridBagConstraints = new java.awt.GridBagConstraints();  
5304: gridBagConstraints.gridx = 0;  
5305: gridBagConstraints.gridy = 0;  
5306: jPanel82.add(tp114, gridBagConstraints);  
5307:  
5308: gridBagConstraints = new java.awt.GridBagConstraints();  
5309: gridBagConstraints.gridx = 4;  
5310: gridBagConstraints.gridy = 16;  
5311: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5312: gridBagConstraints.weightx = 1.0;  
5313: panelPesados.add(jPanel82, gridBagConstraints);  
5314:  
5315: jPanel83.setLayout(new java.awt.GridBagLayout());  
5316:  
5317: tp122.setColumns(5);  
5318: tp122.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5319: tp122.setToolTipText("");  
5320: tp122.setInputVerifier(intVerifier);  
5321: tp122.setMinimumSize(new java.awt.Dimension(50, 19));  
5322: tp122.setName(""); // NOI18N  
5323: tp122.setNextFocusableComponent(tp123);  
5324: gridBagConstraints = new java.awt.GridBagConstraints();  
5325: gridBagConstraints.gridx = 0;  
5326: gridBagConstraints.gridy = 0;  
5327: jPanel83.add(tp122, gridBagConstraints);  
5328:  
5329: gridBagConstraints = new java.awt.GridBagConstraints();  
5330: gridBagConstraints.gridx = 4;  
5331: gridBagConstraints.gridy = 17;  
5332: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5333: gridBagConstraints.weightx = 1.0;  
5334: panelPesados.add(jPanel83, gridBagConstraints);  
5335:  
5336: jPanel84.setLayout(new java.awt.GridBagLayout());  
5337:  
5338: tp130.setColumns(5);  
5339: tp130.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5340: tp130.setToolTipText("");  
5341: tp130.setInputVerifier(intVerifier);  
5342: tp130.setMinimumSize(new java.awt.Dimension(50, 19));  
5343: tp130.setName(""); // NOI18N  
5344: tp130.setNextFocusableComponent(tp131);  
5345: gridBagConstraints = new java.awt.GridBagConstraints();  
5346: gridBagConstraints.gridx = 0;  
5347: gridBagConstraints.gridy = 0;  
5348: jPanel84.add(tp130, gridBagConstraints);  
5349:  
5350: gridBagConstraints = new java.awt.GridBagConstraints();  
5351: gridBagConstraints.gridx = 4;  
5352: gridBagConstraints.gridy = 18;  
5353: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5354: gridBagConstraints.weightx = 1.0;  
5355: panelPesados.add(jPanel84, gridBagConstraints);  
5356:  
5357: jPanel85.setLayout(new java.awt.GridBagLayout());  
5358:  
5359: tp4.setColumns(5);  
5360: tp4.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5361: tp4.setToolTipText("");  
5362: tp4.setInputVerifier(intVerifier);  
5363: tp4.setMinimumSize(new java.awt.Dimension(50, 19));  
5364: tp4.setName(""); // NOI18N  
5365: tp4.setNextFocusableComponent(tp5);  
5366: gridBagConstraints = new java.awt.GridBagConstraints();  
5367: gridBagConstraints.gridx = 0;  
5368: gridBagConstraints.gridy = 0;  
5369: gridBagConstraints.weightx = 1.0;  
5370: jPanel85.add(tp4, gridBagConstraints);  
5371:  
5372: gridBagConstraints = new java.awt.GridBagConstraints();  
5373: gridBagConstraints.gridx = 6;  
5374: gridBagConstraints.gridy = 2;  
5375: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5376: gridBagConstraints.weightx = 1.0;  
5377: panelPesados.add(jPanel85, gridBagConstraints);  
5378:  
5379: jPanel86.setLayout(new java.awt.GridBagLayout());  
5380:  
5381: tp12.setColumns(5);  
5382: tp12.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5383: tp12.setToolTipText("");  
5384: tp12.setInputVerifier(intVerifier);  
5385: tp12.setMinimumSize(new java.awt.Dimension(50, 19));  
5386: tp12.setName(""); // NOI18N  
5387: tp12.setNextFocusableComponent(tp13);  
5388: gridBagConstraints = new java.awt.GridBagConstraints();  
5389: gridBagConstraints.gridx = 0;  
5390: gridBagConstraints.gridy = 0;  
5391: gridBagConstraints.weightx = 1.0;  
5392: jPanel86.add(tp12, gridBagConstraints);  
5393:  
5394: gridBagConstraints = new java.awt.GridBagConstraints();  
5395: gridBagConstraints.gridx = 6;  
5396: gridBagConstraints.gridy = 3;  
5397: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5398: gridBagConstraints.weightx = 1.0;  
5399: panelPesados.add(jPanel86, gridBagConstraints);  
5400:  
5401: jPanel87.setLayout(new java.awt.GridBagLayout());  
5402:  
5403: tp20.setColumns(5);  
5404: tp20.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5405: tp20.setToolTipText("");  
5406: tp20.setInputVerifier(intVerifier);  
5407: tp20.setMinimumSize(new java.awt.Dimension(50, 19));  
5408: tp20.setName(""); // NOI18N  
5409: tp20.setNextFocusableComponent(tp21);  
5410: gridBagConstraints = new java.awt.GridBagConstraints();  
5411: gridBagConstraints.gridx = 0;  
5412: gridBagConstraints.gridy = 0;  
5413: gridBagConstraints.weightx = 1.0;  
5414: jPanel87.add(tp20, gridBagConstraints);  
5415:  
5416: gridBagConstraints = new java.awt.GridBagConstraints();  
5417: gridBagConstraints.gridx = 6;  
5418: gridBagConstraints.gridy = 4;  
5419: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5420: gridBagConstraints.weightx = 1.0;  
5421: panelPesados.add(jPanel87, gridBagConstraints);  
5422:  
5423: jPanel88.setLayout(new java.awt.GridBagLayout());  
5424:  
5425: tp28.setColumns(5);  
5426: tp28.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5427: tp28.setToolTipText("");  
5428: tp28.setInputVerifier(intVerifier);  
5429: tp28.setMinimumSize(new java.awt.Dimension(50, 19));  
5430: tp28.setName(""); // NOI18N  
5431: tp28.setNextFocusableComponent(tp29);  
5432: gridBagConstraints = new java.awt.GridBagConstraints();  
5433: gridBagConstraints.gridx = 0;  
5434: gridBagConstraints.gridy = 0;  
5435: gridBagConstraints.weightx = 1.0;  
5436: jPanel88.add(tp28, gridBagConstraints);  
5437:  
5438: gridBagConstraints = new java.awt.GridBagConstraints();  
5439: gridBagConstraints.gridx = 6;  
5440: gridBagConstraints.gridy = 5;  
5441: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5442: gridBagConstraints.weightx = 1.0;  
5443: panelPesados.add(jPanel88, gridBagConstraints);  
5444:  
5445: jPanel89.setLayout(new java.awt.GridBagLayout());  
5446:  
5447: tp36.setColumns(5);  
5448: tp36.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5449: tp36.setToolTipText("");  
5450: tp36.setInputVerifier(intVerifier);  
5451: tp36.setMinimumSize(new java.awt.Dimension(50, 19));  
5452: tp36.setName(""); // NOI18N  
5453: tp36.setNextFocusableComponent(tp37);  
5454: gridBagConstraints = new java.awt.GridBagConstraints();  
5455: gridBagConstraints.gridx = 0;  
5456: gridBagConstraints.gridy = 0;  
5457: jPanel89.add(tp36, gridBagConstraints);  
5458:  
5459: gridBagConstraints = new java.awt.GridBagConstraints();  
5460: gridBagConstraints.gridx = 6;  
5461: gridBagConstraints.gridy = 6;  
5462: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5463: gridBagConstraints.weightx = 1.0;  
5464: panelPesados.add(jPanel89, gridBagConstraints);  
5465:  
5466: jPanel90.setLayout(new java.awt.GridBagLayout());  
5467:  
5468: tp44.setColumns(5);  
5469: tp44.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5470: tp44.setToolTipText("");  
5471: tp44.setInputVerifier(intVerifier);  
5472: tp44.setMinimumSize(new java.awt.Dimension(50, 19));  
5473: tp44.setName(""); // NOI18N  
5474: tp44.setNextFocusableComponent(tp45);  
5475: gridBagConstraints = new java.awt.GridBagConstraints();  
5476: gridBagConstraints.gridx = 0;  
5477: gridBagConstraints.gridy = 0;  
5478: jPanel90.add(tp44, gridBagConstraints);  
5479:  
5480: gridBagConstraints = new java.awt.GridBagConstraints();  
5481: gridBagConstraints.gridx = 6;  
5482: gridBagConstraints.gridy = 7;  
5483: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5484: gridBagConstraints.weightx = 1.0;  
5485: panelPesados.add(jPanel90, gridBagConstraints);  
5486:  
5487: jPanel91.setLayout(new java.awt.GridBagLayout());  
5488:  
5489: tp52.setColumns(5);  
5490: tp52.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5491: tp52.setToolTipText("");  
5492: tp52.setInputVerifier(intVerifier);  
5493: tp52.setMinimumSize(new java.awt.Dimension(50, 19));  
5494: tp52.setName(""); // NOI18N  
5495: tp52.setNextFocusableComponent(tp53);  
5496: gridBagConstraints = new java.awt.GridBagConstraints();  
5497: gridBagConstraints.gridx = 0;  
5498: gridBagConstraints.gridy = 0;  
5499: jPanel91.add(tp52, gridBagConstraints);  
5500:  
5501: gridBagConstraints = new java.awt.GridBagConstraints();  
5502: gridBagConstraints.gridx = 6;  
5503: gridBagConstraints.gridy = 8;  
5504: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5505: gridBagConstraints.weightx = 1.0;  
5506: panelPesados.add(jPanel91, gridBagConstraints);  
5507:  
5508: jPanel92.setLayout(new java.awt.GridBagLayout());  
5509:  
5510: tp60.setColumns(5);  
5511: tp60.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5512: tp60.setToolTipText("");  
5513: tp60.setInputVerifier(intVerifier);  
5514: tp60.setMinimumSize(new java.awt.Dimension(50, 19));  
5515: tp60.setName(""); // NOI18N  
5516: tp60.setNextFocusableComponent(tp61);  
5517: gridBagConstraints = new java.awt.GridBagConstraints();  
5518: gridBagConstraints.gridx = 0;  
5519: gridBagConstraints.gridy = 0;  
5520: jPanel92.add(tp60, gridBagConstraints);  
5521:  
5522: gridBagConstraints = new java.awt.GridBagConstraints();  
5523: gridBagConstraints.gridx = 6;  
5524: gridBagConstraints.gridy = 9;  
5525: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5526: gridBagConstraints.weightx = 1.0;  
5527: panelPesados.add(jPanel92, gridBagConstraints);  
5528:  
5529: jPanel93.setLayout(new java.awt.GridBagLayout());  
5530:  
5531: tp68.setColumns(5);  
5532: tp68.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5533: tp68.setToolTipText("");  
5534: tp68.setInputVerifier(intVerifier);  
5535: tp68.setMinimumSize(new java.awt.Dimension(50, 19));  
5536: tp68.setName(""); // NOI18N  
5537: tp68.setNextFocusableComponent(tp69);  
5538: gridBagConstraints = new java.awt.GridBagConstraints();  
5539: gridBagConstraints.gridx = 0;  
5540: gridBagConstraints.gridy = 0;  
5541: jPanel93.add(tp68, gridBagConstraints);  
5542:  
5543: gridBagConstraints = new java.awt.GridBagConstraints();  
5544: gridBagConstraints.gridx = 6;  
5545: gridBagConstraints.gridy = 10;  
5546: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5547: gridBagConstraints.weightx = 1.0;  
5548: panelPesados.add(jPanel93, gridBagConstraints);  
5549:  
5550: jPanel94.setLayout(new java.awt.GridBagLayout());  
5551:  
5552: tp76.setColumns(5);  
5553: tp76.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5554: tp76.setToolTipText("");  
5555: tp76.setInputVerifier(intVerifier);  
5556: tp76.setMinimumSize(new java.awt.Dimension(50, 19));  
5557: tp76.setName(""); // NOI18N  
5558: tp76.setNextFocusableComponent(tp77);  
5559: gridBagConstraints = new java.awt.GridBagConstraints();  
5560: gridBagConstraints.gridx = 0;  
5561: gridBagConstraints.gridy = 0;  
5562: jPanel94.add(tp76, gridBagConstraints);  
5563:  
5564: gridBagConstraints = new java.awt.GridBagConstraints();  
5565: gridBagConstraints.gridx = 6;  
5566: gridBagConstraints.gridy = 11;  
5567: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5568: gridBagConstraints.weightx = 1.0;  
5569: panelPesados.add(jPanel94, gridBagConstraints);  
5570:  
5571: jPanel95.setLayout(new java.awt.GridBagLayout());  
5572:  
5573: tp84.setColumns(5);  
5574: tp84.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5575: tp84.setToolTipText("");  
5576: tp84.setInputVerifier(intVerifier);  
5577: tp84.setMinimumSize(new java.awt.Dimension(50, 19));  
5578: tp84.setName(""); // NOI18N  
5579: tp84.setNextFocusableComponent(tp85);  
5580: gridBagConstraints = new java.awt.GridBagConstraints();  
5581: gridBagConstraints.gridx = 0;  
5582: gridBagConstraints.gridy = 0;  
5583: jPanel95.add(tp84, gridBagConstraints);  
5584:  
5585: gridBagConstraints = new java.awt.GridBagConstraints();  
5586: gridBagConstraints.gridx = 6;  
5587: gridBagConstraints.gridy = 12;  
5588: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5589: gridBagConstraints.weightx = 1.0;  
5590: panelPesados.add(jPanel95, gridBagConstraints);  
5591:  
5592: jPanel96.setLayout(new java.awt.GridBagLayout());  
5593:  
5594: tp92.setColumns(5);  
5595: tp92.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5596: tp92.setToolTipText("");  
5597: tp92.setInputVerifier(intVerifier);  
5598: tp92.setMinimumSize(new java.awt.Dimension(50, 19));  
5599: tp92.setName(""); // NOI18N  
5600: tp92.setNextFocusableComponent(tp93);  
5601: gridBagConstraints = new java.awt.GridBagConstraints();  
5602: gridBagConstraints.gridx = 0;  
5603: gridBagConstraints.gridy = 0;  
5604: jPanel96.add(tp92, gridBagConstraints);  
5605:  
5606: gridBagConstraints = new java.awt.GridBagConstraints();  
5607: gridBagConstraints.gridx = 6;  
5608: gridBagConstraints.gridy = 13;  
5609: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5610: gridBagConstraints.weightx = 1.0;  
5611: panelPesados.add(jPanel96, gridBagConstraints);  
5612:  
5613: jPanel97.setLayout(new java.awt.GridBagLayout());  
5614:  
5615: tp100.setColumns(5);  
5616: tp100.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5617: tp100.setToolTipText("");  
5618: tp100.setInputVerifier(intVerifier);  
5619: tp100.setMinimumSize(new java.awt.Dimension(50, 19));  
5620: tp100.setName(""); // NOI18N  
5621: tp100.setNextFocusableComponent(tp101);  
5622: gridBagConstraints = new java.awt.GridBagConstraints();  
5623: gridBagConstraints.gridx = 0;  
5624: gridBagConstraints.gridy = 0;  
5625: jPanel97.add(tp100, gridBagConstraints);  
5626:  
5627: gridBagConstraints = new java.awt.GridBagConstraints();  
5628: gridBagConstraints.gridx = 6;  
5629: gridBagConstraints.gridy = 14;  
5630: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5631: gridBagConstraints.weightx = 1.0;  
5632: panelPesados.add(jPanel97, gridBagConstraints);  
5633:  
5634: jPanel98.setLayout(new java.awt.GridBagLayout());  
5635:  
5636: tp108.setColumns(5);  
5637: tp108.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5638: tp108.setToolTipText("");  
5639: tp108.setInputVerifier(intVerifier);  
5640: tp108.setMinimumSize(new java.awt.Dimension(50, 19));  
5641: tp108.setName(""); // NOI18N  
5642: tp108.setNextFocusableComponent(tp109);  
5643: gridBagConstraints = new java.awt.GridBagConstraints();  
5644: gridBagConstraints.gridx = 0;  
5645: gridBagConstraints.gridy = 0;  
5646: jPanel98.add(tp108, gridBagConstraints);  
5647:  
5648: gridBagConstraints = new java.awt.GridBagConstraints();  
5649: gridBagConstraints.gridx = 6;  
5650: gridBagConstraints.gridy = 15;  
5651: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5652: gridBagConstraints.weightx = 1.0;  
5653: panelPesados.add(jPanel98, gridBagConstraints);  
5654:  
5655: jPanel99.setLayout(new java.awt.GridBagLayout());  
5656:  
5657: tp116.setColumns(5);  
5658: tp116.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5659: tp116.setToolTipText("");  
5660: tp116.setInputVerifier(intVerifier);  
5661: tp116.setMinimumSize(new java.awt.Dimension(50, 19));  
5662: tp116.setName(""); // NOI18N  
5663: tp116.setNextFocusableComponent(tp117);  
5664: gridBagConstraints = new java.awt.GridBagConstraints();  
5665: gridBagConstraints.gridx = 0;  
5666: gridBagConstraints.gridy = 0;  
5667: jPanel99.add(tp116, gridBagConstraints);  
5668:  
5669: gridBagConstraints = new java.awt.GridBagConstraints();  
5670: gridBagConstraints.gridx = 6;  
5671: gridBagConstraints.gridy = 16;  
5672: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5673: gridBagConstraints.weightx = 1.0;  
5674: panelPesados.add(jPanel99, gridBagConstraints);  
5675:  
5676: jPanel100.setLayout(new java.awt.GridBagLayout());  
5677:  
5678: tp124.setColumns(5);  
5679: tp124.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5680: tp124.setToolTipText("");  
5681: tp124.setInputVerifier(intVerifier);  
5682: tp124.setMinimumSize(new java.awt.Dimension(50, 19));  
5683: tp124.setName(""); // NOI18N  
5684: tp124.setNextFocusableComponent(tp125);  
5685: gridBagConstraints = new java.awt.GridBagConstraints();  
5686: gridBagConstraints.gridx = 0;  
5687: gridBagConstraints.gridy = 0;  
5688: jPanel100.add(tp124, gridBagConstraints);  
5689:  
5690: gridBagConstraints = new java.awt.GridBagConstraints();  
5691: gridBagConstraints.gridx = 6;  
5692: gridBagConstraints.gridy = 17;  
5693: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5694: gridBagConstraints.weightx = 1.0;  
5695: panelPesados.add(jPanel100, gridBagConstraints);  
5696:  
5697: jPanel101.setLayout(new java.awt.GridBagLayout());  
5698:  
5699: tp132.setColumns(5);  
5700: tp132.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5701: tp132.setToolTipText("");  
5702: tp132.setInputVerifier(intVerifier);  
5703: tp132.setMinimumSize(new java.awt.Dimension(50, 19));  
5704: tp132.setName(""); // NOI18N  
5705: tp132.setNextFocusableComponent(tp133);  
5706: gridBagConstraints = new java.awt.GridBagConstraints();  
5707: gridBagConstraints.gridx = 0;  
5708: gridBagConstraints.gridy = 0;  
5709: jPanel101.add(tp132, gridBagConstraints);  
5710:  
5711: gridBagConstraints = new java.awt.GridBagConstraints();  
5712: gridBagConstraints.gridx = 6;  
5713: gridBagConstraints.gridy = 18;  
5714: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5715: gridBagConstraints.weightx = 1.0;  
5716: panelPesados.add(jPanel101, gridBagConstraints);  
5717:  
5718: jPanel102.setLayout(new java.awt.GridBagLayout());  
5719:  
5720: tp6.setColumns(5);  
5721: tp6.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5722: tp6.setToolTipText("");  
5723: tp6.setInputVerifier(intVerifier);  
5724: tp6.setMinimumSize(new java.awt.Dimension(50, 19));  
5725: tp6.setName(""); // NOI18N  
5726: tp6.setNextFocusableComponent(tp7);  
5727: gridBagConstraints = new java.awt.GridBagConstraints();  
5728: gridBagConstraints.gridx = 0;  
5729: gridBagConstraints.gridy = 0;  
5730: gridBagConstraints.weightx = 1.0;  
5731: jPanel102.add(tp6, gridBagConstraints);  
5732:  
5733: gridBagConstraints = new java.awt.GridBagConstraints();  
5734: gridBagConstraints.gridx = 8;  
5735: gridBagConstraints.gridy = 2;  
5736: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5737: gridBagConstraints.weightx = 1.0;  
5738: panelPesados.add(jPanel102, gridBagConstraints);  
5739:  
5740: jPanel103.setLayout(new java.awt.GridBagLayout());  
5741:  
5742: tp14.setColumns(5);  
5743: tp14.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5744: tp14.setToolTipText("");  
5745: tp14.setInputVerifier(intVerifier);  
5746: tp14.setMinimumSize(new java.awt.Dimension(50, 19));  
5747: tp14.setName(""); // NOI18N  
5748: tp14.setNextFocusableComponent(tp15);  
5749: gridBagConstraints = new java.awt.GridBagConstraints();  
5750: gridBagConstraints.gridx = 0;  
5751: gridBagConstraints.gridy = 0;  
5752: gridBagConstraints.weightx = 1.0;  
5753: jPanel103.add(tp14, gridBagConstraints);  
5754:  
5755: gridBagConstraints = new java.awt.GridBagConstraints();  
5756: gridBagConstraints.gridx = 8;  
5757: gridBagConstraints.gridy = 3;  
5758: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5759: gridBagConstraints.weightx = 1.0;  
5760: panelPesados.add(jPanel103, gridBagConstraints);  
5761:  
5762: jPanel104.setLayout(new java.awt.GridBagLayout());  
5763:  
5764: tp22.setColumns(5);  
5765: tp22.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5766: tp22.setToolTipText("");  
5767: tp22.setInputVerifier(intVerifier);  
5768: tp22.setMinimumSize(new java.awt.Dimension(50, 19));  
5769: tp22.setName(""); // NOI18N  
5770: tp22.setNextFocusableComponent(tp23);  
5771: gridBagConstraints = new java.awt.GridBagConstraints();  
5772: gridBagConstraints.gridx = 0;  
5773: gridBagConstraints.gridy = 0;  
5774: gridBagConstraints.weightx = 1.0;  
5775: jPanel104.add(tp22, gridBagConstraints);  
5776:  
5777: gridBagConstraints = new java.awt.GridBagConstraints();  
5778: gridBagConstraints.gridx = 8;  
5779: gridBagConstraints.gridy = 4;  
5780: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5781: gridBagConstraints.weightx = 1.0;  
5782: panelPesados.add(jPanel104, gridBagConstraints);  
5783:  
5784: jPanel105.setLayout(new java.awt.GridBagLayout());  
5785:  
5786: tp30.setColumns(5);  
5787: tp30.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5788: tp30.setToolTipText("");  
5789: tp30.setInputVerifier(intVerifier);  
5790: tp30.setMinimumSize(new java.awt.Dimension(50, 19));  
5791: tp30.setName(""); // NOI18N  
5792: tp30.setNextFocusableComponent(tp31);  
5793: gridBagConstraints = new java.awt.GridBagConstraints();  
5794: gridBagConstraints.gridx = 0;  
5795: gridBagConstraints.gridy = 0;  
5796: gridBagConstraints.weightx = 1.0;  
5797: jPanel105.add(tp30, gridBagConstraints);  
5798:  
5799: gridBagConstraints = new java.awt.GridBagConstraints();  
5800: gridBagConstraints.gridx = 8;  
5801: gridBagConstraints.gridy = 5;  
5802: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5803: gridBagConstraints.weightx = 1.0;  
5804: panelPesados.add(jPanel105, gridBagConstraints);  
5805:  
5806: jPanel106.setLayout(new java.awt.GridBagLayout());  
5807:  
5808: tp38.setColumns(5);  
5809: tp38.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5810: tp38.setToolTipText("");  
5811: tp38.setInputVerifier(intVerifier);  
5812: tp38.setMinimumSize(new java.awt.Dimension(50, 19));  
5813: tp38.setName(""); // NOI18N  
5814: tp38.setNextFocusableComponent(tp39);  
5815: gridBagConstraints = new java.awt.GridBagConstraints();  
5816: gridBagConstraints.gridx = 0;  
5817: gridBagConstraints.gridy = 0;  
5818: jPanel106.add(tp38, gridBagConstraints);  
5819:  
5820: gridBagConstraints = new java.awt.GridBagConstraints();  
5821: gridBagConstraints.gridx = 8;  
5822: gridBagConstraints.gridy = 6;  
5823: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5824: gridBagConstraints.weightx = 1.0;  
5825: panelPesados.add(jPanel106, gridBagConstraints);  
5826:  
5827: jPanel107.setLayout(new java.awt.GridBagLayout());  
5828:  
5829: tp46.setColumns(5);  
5830: tp46.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5831: tp46.setToolTipText("");  
5832: tp46.setInputVerifier(intVerifier);  
5833: tp46.setMinimumSize(new java.awt.Dimension(50, 19));  
5834: tp46.setName(""); // NOI18N  
5835: tp46.setNextFocusableComponent(tp47);  
5836: gridBagConstraints = new java.awt.GridBagConstraints();  
5837: gridBagConstraints.gridx = 0;  
5838: gridBagConstraints.gridy = 0;  
5839: jPanel107.add(tp46, gridBagConstraints);  
5840:  
5841: gridBagConstraints = new java.awt.GridBagConstraints();  
5842: gridBagConstraints.gridx = 8;  
5843: gridBagConstraints.gridy = 7;  
5844: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5845: gridBagConstraints.weightx = 1.0;  
5846: panelPesados.add(jPanel107, gridBagConstraints);  
5847:  
5848: jPanel108.setLayout(new java.awt.GridBagLayout());  
5849:  
5850: tp54.setColumns(5);  
5851: tp54.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5852: tp54.setToolTipText("");  
5853: tp54.setInputVerifier(intVerifier);  
5854: tp54.setMinimumSize(new java.awt.Dimension(50, 19));  
5855: tp54.setName(""); // NOI18N  
5856: tp54.setNextFocusableComponent(tp55);  
5857: gridBagConstraints = new java.awt.GridBagConstraints();  
5858: gridBagConstraints.gridx = 0;  
5859: gridBagConstraints.gridy = 0;  
5860: jPanel108.add(tp54, gridBagConstraints);  
5861:  
5862: gridBagConstraints = new java.awt.GridBagConstraints();  
5863: gridBagConstraints.gridx = 8;  
5864: gridBagConstraints.gridy = 8;  
5865: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5866: gridBagConstraints.weightx = 1.0;  
5867: panelPesados.add(jPanel108, gridBagConstraints);  
5868:  
5869: jPanel109.setLayout(new java.awt.GridBagLayout());  
5870:  
5871: tp62.setColumns(5);  
5872: tp62.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5873: tp62.setToolTipText("");  
5874: tp62.setInputVerifier(intVerifier);  
5875: tp62.setMinimumSize(new java.awt.Dimension(50, 19));  
5876: tp62.setName(""); // NOI18N  
5877: tp62.setNextFocusableComponent(tp63);  
5878: gridBagConstraints = new java.awt.GridBagConstraints();  
5879: gridBagConstraints.gridx = 0;  
5880: gridBagConstraints.gridy = 0;  
5881: jPanel109.add(tp62, gridBagConstraints);  
5882:  
5883: gridBagConstraints = new java.awt.GridBagConstraints();  
5884: gridBagConstraints.gridx = 8;  
5885: gridBagConstraints.gridy = 9;  
5886: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5887: gridBagConstraints.weightx = 1.0;  
5888: panelPesados.add(jPanel109, gridBagConstraints);  
5889:  
5890: jPanel110.setLayout(new java.awt.GridBagLayout());  
5891:  
5892: tp70.setColumns(5);  
5893: tp70.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5894: tp70.setToolTipText("");  
5895: tp70.setInputVerifier(intVerifier);  
5896: tp70.setMinimumSize(new java.awt.Dimension(50, 19));  
5897: tp70.setName(""); // NOI18N  
5898: tp70.setNextFocusableComponent(tp71);  
5899: gridBagConstraints = new java.awt.GridBagConstraints();  
5900: gridBagConstraints.gridx = 0;  
5901: gridBagConstraints.gridy = 0;  
5902: jPanel110.add(tp70, gridBagConstraints);  
5903:  
5904: gridBagConstraints = new java.awt.GridBagConstraints();  
5905: gridBagConstraints.gridx = 8;  
5906: gridBagConstraints.gridy = 10;  
5907: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5908: gridBagConstraints.weightx = 1.0;  
5909: panelPesados.add(jPanel110, gridBagConstraints);  
5910:  
5911: jPanel111.setLayout(new java.awt.GridBagLayout());  
5912:  
5913: tp78.setColumns(5);  
5914: tp78.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5915: tp78.setToolTipText("");  
5916: tp78.setInputVerifier(intVerifier);  
5917: tp78.setMinimumSize(new java.awt.Dimension(50, 19));  
5918: tp78.setName(""); // NOI18N  
5919: tp78.setNextFocusableComponent(tp79);  
5920: gridBagConstraints = new java.awt.GridBagConstraints();  
5921: gridBagConstraints.gridx = 0;  
5922: gridBagConstraints.gridy = 0;  
5923: jPanel111.add(tp78, gridBagConstraints);  
5924:  
5925: gridBagConstraints = new java.awt.GridBagConstraints();  
5926: gridBagConstraints.gridx = 8;  
5927: gridBagConstraints.gridy = 11;  
5928: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5929: gridBagConstraints.weightx = 1.0;  
5930: panelPesados.add(jPanel111, gridBagConstraints);  
5931:  
5932: jPanel112.setLayout(new java.awt.GridBagLayout());  
5933:  
5934: tp86.setColumns(5);  
5935: tp86.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5936: tp86.setToolTipText("");  
5937: tp86.setInputVerifier(intVerifier);  
5938: tp86.setMinimumSize(new java.awt.Dimension(50, 19));  
5939: tp86.setName(""); // NOI18N  
5940: tp86.setNextFocusableComponent(tp87);  
5941: gridBagConstraints = new java.awt.GridBagConstraints();  
5942: gridBagConstraints.gridx = 0;  
5943: gridBagConstraints.gridy = 0;  
5944: jPanel112.add(tp86, gridBagConstraints);  
5945:  
5946: gridBagConstraints = new java.awt.GridBagConstraints();  
5947: gridBagConstraints.gridx = 8;  
5948: gridBagConstraints.gridy = 12;  
5949: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5950: gridBagConstraints.weightx = 1.0;  
5951: panelPesados.add(jPanel112, gridBagConstraints);  
5952:  
5953: jPanel113.setLayout(new java.awt.GridBagLayout());  
5954:  
5955: tp94.setColumns(5);  
5956: tp94.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5957: tp94.setToolTipText("");  
5958: tp94.setInputVerifier(intVerifier);  
5959: tp94.setMinimumSize(new java.awt.Dimension(50, 19));  
5960: tp94.setName(""); // NOI18N  
5961: tp94.setNextFocusableComponent(tp95);  
5962: gridBagConstraints = new java.awt.GridBagConstraints();  
5963: gridBagConstraints.gridx = 0;  
5964: gridBagConstraints.gridy = 0;  
5965: jPanel113.add(tp94, gridBagConstraints);  
5966:  
5967: gridBagConstraints = new java.awt.GridBagConstraints();  
5968: gridBagConstraints.gridx = 8;  
5969: gridBagConstraints.gridy = 13;  
5970: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5971: gridBagConstraints.weightx = 1.0;  
5972: panelPesados.add(jPanel113, gridBagConstraints);  
5973:  
5974: jPanel114.setLayout(new java.awt.GridBagLayout());  
5975:  
5976: tp102.setColumns(5);  
5977: tp102.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5978: tp102.setToolTipText("");  
5979: tp102.setInputVerifier(intVerifier);  
5980: tp102.setMinimumSize(new java.awt.Dimension(50, 19));  
5981: tp102.setName(""); // NOI18N  
5982: tp102.setNextFocusableComponent(tp103);  
5983: gridBagConstraints = new java.awt.GridBagConstraints();  
5984: gridBagConstraints.gridx = 0;  
5985: gridBagConstraints.gridy = 0;  
5986: jPanel114.add(tp102, gridBagConstraints);  
5987:  
5988: gridBagConstraints = new java.awt.GridBagConstraints();  
5989: gridBagConstraints.gridx = 8;  
5990: gridBagConstraints.gridy = 14;  
5991: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
5992: gridBagConstraints.weightx = 1.0;  
5993: panelPesados.add(jPanel114, gridBagConstraints);  
5994:  
5995: jPanel115.setLayout(new java.awt.GridBagLayout());  
5996:  
5997: tp110.setColumns(5);  
5998: tp110.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
5999: tp110.setToolTipText("");  
6000: tp110.setInputVerifier(intVerifier);  
6001: tp110.setMinimumSize(new java.awt.Dimension(50, 19));  
6002: tp110.setName(""); // NOI18N  
6003: tp110.setNextFocusableComponent(tp111);  
6004: gridBagConstraints = new java.awt.GridBagConstraints();  
6005: gridBagConstraints.gridx = 0;  
6006: gridBagConstraints.gridy = 0;  
6007: jPanel115.add(tp110, gridBagConstraints);  
6008:  
6009: gridBagConstraints = new java.awt.GridBagConstraints();  
6010: gridBagConstraints.gridx = 8;  
6011: gridBagConstraints.gridy = 15;  
6012: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
6013: gridBagConstraints.weightx = 1.0;  
6014: panelPesados.add(jPanel115, gridBagConstraints);  
6015:  
6016: jPanel116.setLayout(new java.awt.GridBagLayout());  
6017:  
6018: tp118.setColumns(5);  
6019: tp118.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6020: tp118.setToolTipText("");  
6021: tp118.setInputVerifier(intVerifier);  
6022: tp118.setMinimumSize(new java.awt.Dimension(50, 19));  
6023: tp118.setName(""); // NOI18N  
6024: tp118.setNextFocusableComponent(tp119);  
6025: gridBagConstraints = new java.awt.GridBagConstraints();  
6026: gridBagConstraints.gridx = 0;  
6027: gridBagConstraints.gridy = 0;  
6028: jPanel116.add(tp118, gridBagConstraints);  
6029:  
6030: gridBagConstraints = new java.awt.GridBagConstraints();  
6031: gridBagConstraints.gridx = 8;  
6032: gridBagConstraints.gridy = 16;  
6033: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
6034: gridBagConstraints.weightx = 1.0;  
6035: panelPesados.add(jPanel116, gridBagConstraints);  
6036:  
6037: jPanel117.setLayout(new java.awt.GridBagLayout());  
6038:  
6039: tp126.setColumns(5);  
6040: tp126.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6041: tp126.setToolTipText("");  
6042: tp126.setInputVerifier(intVerifier);  
6043: tp126.setMinimumSize(new java.awt.Dimension(50, 19));  
6044: tp126.setName(""); // NOI18N  
6045: tp126.setNextFocusableComponent(tp127);  
6046: gridBagConstraints = new java.awt.GridBagConstraints();  
6047: gridBagConstraints.gridx = 0;  
6048: gridBagConstraints.gridy = 0;  
6049: jPanel117.add(tp126, gridBagConstraints);  
6050:  
6051: gridBagConstraints = new java.awt.GridBagConstraints();  
6052: gridBagConstraints.gridx = 8;  
6053: gridBagConstraints.gridy = 17;  
6054: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
6055: gridBagConstraints.weightx = 1.0;  
6056: panelPesados.add(jPanel117, gridBagConstraints);  
6057:  
6058: jPanel118.setLayout(new java.awt.GridBagLayout());  
6059:  
6060: tp134.setColumns(5);  
6061: tp134.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6062: tp134.setToolTipText("");  
6063: tp134.setInputVerifier(intVerifier);  
6064: tp134.setMinimumSize(new java.awt.Dimension(50, 19));  
6065: tp134.setName(""); // NOI18N  
6066: tp134.setNextFocusableComponent(tp135);  
6067: gridBagConstraints = new java.awt.GridBagConstraints();  
6068: gridBagConstraints.gridx = 0;  
6069: gridBagConstraints.gridy = 0;  
6070: jPanel118.add(tp134, gridBagConstraints);  
6071:  
6072: gridBagConstraints = new java.awt.GridBagConstraints();  
6073: gridBagConstraints.gridx = 8;  
6074: gridBagConstraints.gridy = 18;  
6075: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
6076: gridBagConstraints.weightx = 1.0;  
6077: panelPesados.add(jPanel118, gridBagConstraints);  
6078:  
6079: jScrollPane4.setViewportView(panelPesados);  
6080:  
6081: lblPesquisador1.setText("Pesquisador\_2:");  
6082:  
6083: txtPesquisador2.setNextFocusableComponent(tp0);  
6084:  
6085: txtTotalPesados.setEditable(false);  
6086: txtTotalPesados.setText("0");  
6087:  
6088: jLabel37.setText("Total:");  
6089:  
6090: lblFolha2.setText("-");  
6091:  
6092: jLabel40.setText("Folha:");  
6093:  
6094: javax.swing.GroupLayout tab\_pesadosLayout = new javax.swing.GroupLayout(tab\_pesados);  
6095: tab\_pesados.setLayout(tab\_pesadosLayout);  
6096: tab\_pesadosLayout.setHorizontalGroup(  
6097: tab\_pesadosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6098: .addGroup(tab\_pesadosLayout.createSequentialGroup()  
6099: .addContainerGap()  
6100: .addComponent(lblPesquisador1)  
6101: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6102: .addComponent(txtPesquisador2, javax.swing.GroupLayout.PREFERRED\_SIZE, 234, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6103: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6104: .addComponent(jLabel40)  
6105: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6106: .addComponent(lblFolha2)  
6107: .addGap(18, 18, 18)  
6108: .addComponent(jLabel37)  
6109: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6110: .addComponent(txtTotalPesados, javax.swing.GroupLayout.PREFERRED\_SIZE, 64, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6111: .addContainerGap())  
6112: .addComponent(jScrollPane4, javax.swing.GroupLayout.DEFAULT\_SIZE, 2914, Short.MAX\_VALUE)  
6113: );  
6114: tab\_pesadosLayout.setVerticalGroup(  
6115: tab\_pesadosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6116: .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, tab\_pesadosLayout.createSequentialGroup()  
6117: .addGap(2, 2, 2)  
6118: .addGroup(tab\_pesadosLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6119: .addComponent(txtPesquisador2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6120: .addComponent(lblPesquisador1)  
6121: .addComponent(txtTotalPesados, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6122: .addComponent(jLabel37)  
6123: .addComponent(lblFolha2)  
6124: .addComponent(jLabel40))  
6125: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6126: .addComponent(jScrollPane4, javax.swing.GroupLayout.DEFAULT\_SIZE, 1365, Short.MAX\_VALUE))  
6127: );  
6128:  
6129: jTabbedPane1.addTab("Caminhões Pesados", tab\_pesados);  
6130:  
6131: btnSalvarForms.setText("Gravar Formulários");  
6132: btnSalvarForms.addActionListener(new java.awt.event.ActionListener() {  
6133: public void actionPerformed(java.awt.event.ActionEvent evt) {  
6134: btnSalvarFormsActionPerformed(evt);  
6135: }  
6136: });  
6137:  
6138: jLabel34.setText("Trecho:");  
6139:  
6140: lblTrecho\_dados.setText("------------------------------------------------------------------------------------------");  
6141:  
6142: btnApagar.setText("Apagar");  
6143: btnApagar.addActionListener(new java.awt.event.ActionListener() {  
6144: public void actionPerformed(java.awt.event.ActionEvent evt) {  
6145: btnApagarActionPerformed(evt);  
6146: }  
6147: });  
6148:  
6149: cmbData.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "Item 1", "Item 2", "Item 3", "Item 4" }));  
6150: cmbData.addActionListener(new java.awt.event.ActionListener() {  
6151: public void actionPerformed(java.awt.event.ActionEvent evt) {  
6152: cmbDataActionPerformed(evt);  
6153: }  
6154: });  
6155:  
6156: javax.swing.GroupLayout pnl\_volumetricaLayout = new javax.swing.GroupLayout(pnl\_volumetrica);  
6157: pnl\_volumetrica.setLayout(pnl\_volumetricaLayout);  
6158: pnl\_volumetricaLayout.setHorizontalGroup(  
6159: pnl\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6160: .addGroup(pnl\_volumetricaLayout.createSequentialGroup()  
6161: .addContainerGap()  
6162: .addGroup(pnl\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)  
6163: .addComponent(lblPosto)  
6164: .addComponent(lblSentido))  
6165: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
6166: .addGroup(pnl\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)  
6167: .addGroup(pnl\_volumetricaLayout.createSequentialGroup()  
6168: .addComponent(rdo\_SentidoAB)  
6169: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
6170: .addComponent(rdo\_SentidoBA)  
6171: .addGap(33, 33, 33)  
6172: .addComponent(lblPista)  
6173: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6174: .addComponent(rdo\_PistaSimples)  
6175: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6176: .addComponent(rdo\_PistaDupla)  
6177: .addGap(53, 53, 53)  
6178: .addComponent(lblHora)  
6179: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6180: .addComponent(cmbHora, javax.swing.GroupLayout.PREFERRED\_SIZE, 105, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6181: .addGroup(pnl\_volumetricaLayout.createSequentialGroup()  
6182: .addComponent(lblPosto\_dados)  
6183: .addGap(62, 62, 62)  
6184: .addComponent(jLabel34)  
6185: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6186: .addComponent(lblTrecho\_dados)))  
6187: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6188: .addGroup(pnl\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6189: .addComponent(lblData, javax.swing.GroupLayout.Alignment.TRAILING)  
6190: .addComponent(lblLocal, javax.swing.GroupLayout.Alignment.TRAILING))  
6191: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
6192: .addGroup(pnl\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)  
6193: .addComponent(txtLocal, javax.swing.GroupLayout.DEFAULT\_SIZE, 234, Short.MAX\_VALUE)  
6194: .addComponent(cmbData, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))  
6195: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6196: .addGroup(pnl\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6197: .addComponent(btnSalvarForms, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED\_SIZE, 185, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6198: .addComponent(btnApagar, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED\_SIZE, 185, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6199: .addContainerGap())  
6200: .addComponent(jTabbedPane1, javax.swing.GroupLayout.Alignment.TRAILING)  
6201: );  
6202: pnl\_volumetricaLayout.setVerticalGroup(  
6203: pnl\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6204: .addGroup(pnl\_volumetricaLayout.createSequentialGroup()  
6205: .addContainerGap()  
6206: .addGroup(pnl\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6207: .addComponent(lblLocal, javax.swing.GroupLayout.PREFERRED\_SIZE, 27, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6208: .addComponent(txtLocal, javax.swing.GroupLayout.PREFERRED\_SIZE, 18, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6209: .addComponent(lblPosto\_dados)  
6210: .addComponent(jLabel34)  
6211: .addComponent(lblTrecho\_dados)  
6212: .addComponent(lblPosto)  
6213: .addComponent(btnApagar))  
6214: .addGap(10, 10, 10)  
6215: .addGroup(pnl\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6216: .addComponent(rdo\_SentidoBA)  
6217: .addComponent(rdo\_SentidoAB)  
6218: .addComponent(lblPista)  
6219: .addComponent(rdo\_PistaSimples)  
6220: .addComponent(rdo\_PistaDupla)  
6221: .addComponent(lblData)  
6222: .addComponent(btnSalvarForms)  
6223: .addComponent(lblSentido)  
6224: .addComponent(lblHora)  
6225: .addComponent(cmbHora, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6226: .addComponent(cmbData, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6227: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6228: .addComponent(jTabbedPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 1419, Short.MAX\_VALUE))  
6229: );  
6230:  
6231: jTabbedPane1.getAccessibleContext().setAccessibleName("tab\_leves");  
6232: jTabbedPane1.getAccessibleContext().setAccessibleDescription("");  
6233:  
6234: tabRelatorio.addTab("Formulário de pesquisa volumétrica", pnl\_volumetrica);  
6235:  
6236: jLabel1.setText("Porta");  
6237:  
6238: txtAreaLog.setEditable(false);  
6239: txtAreaLog.setColumns(20);  
6240: txtAreaLog.setLineWrap(true);  
6241: txtAreaLog.setRows(5);  
6242: txtAreaLog.setWrapStyleWord(true);  
6243: jScrollPane1.setViewportView(txtAreaLog);  
6244: txtAreaLog.setEditable(false);  
6245:  
6246: txtPorta.setEditable(false);  
6247:  
6248: jLabel2.setText("Registro das operações do servidor");  
6249:  
6250: javax.swing.GroupLayout pnl\_servidorLayout = new javax.swing.GroupLayout(pnl\_servidor);  
6251: pnl\_servidor.setLayout(pnl\_servidorLayout);  
6252: pnl\_servidorLayout.setHorizontalGroup(  
6253: pnl\_servidorLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6254: .addComponent(jScrollPane1)  
6255: .addGroup(pnl\_servidorLayout.createSequentialGroup()  
6256: .addContainerGap()  
6257: .addGroup(pnl\_servidorLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)  
6258: .addGroup(pnl\_servidorLayout.createSequentialGroup()  
6259: .addComponent(jLabel1)  
6260: .addGap(4, 4, 4)  
6261: .addComponent(txtPorta))  
6262: .addComponent(jLabel2))  
6263: .addContainerGap(2654, Short.MAX\_VALUE))  
6264: );  
6265: pnl\_servidorLayout.setVerticalGroup(  
6266: pnl\_servidorLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6267: .addGroup(pnl\_servidorLayout.createSequentialGroup()  
6268: .addContainerGap()  
6269: .addGroup(pnl\_servidorLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6270: .addComponent(txtPorta, javax.swing.GroupLayout.PREFERRED\_SIZE, 19, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6271: .addComponent(jLabel1))  
6272: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
6273: .addComponent(jLabel2)  
6274: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6275: .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 1435, Short.MAX\_VALUE))  
6276: );  
6277:  
6278: tabRelatorio.addTab("Servidor Web", pnl\_servidor);  
6279:  
6280: jLabel3.setText("Captura de Dados");  
6281:  
6282: btnInDados.setText("Importar dados do Coletor");  
6283: btnInDados.addActionListener(new java.awt.event.ActionListener() {  
6284: public void actionPerformed(java.awt.event.ActionEvent evt) {  
6285: btnInDadosActionPerformed(evt);  
6286: }  
6287: });  
6288:  
6289: btnExpVolDate.setText("Exportar");  
6290: btnExpVolDate.addActionListener(new java.awt.event.ActionListener() {  
6291: public void actionPerformed(java.awt.event.ActionEvent evt) {  
6292: btnExpVolDateActionPerformed(evt);  
6293: }  
6294: });  
6295:  
6296: jLabel4.setText("Exportação de Dados Pesquisa Volumétrica");  
6297:  
6298: jLabel35.setText("Exportação de Dados Pesquisa OD");  
6299:  
6300: cmbDateExpVol.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "--" }));  
6301: cmbDateExpVol.addActionListener(new java.awt.event.ActionListener() {  
6302: public void actionPerformed(java.awt.event.ActionEvent evt) {  
6303: cmbDateExpVolActionPerformed(evt);  
6304: }  
6305: });  
6306:  
6307: jLabel71.setText("Data:");  
6308:  
6309: cmbDateExpOD.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "--" }));  
6310: cmbDateExpOD.setToolTipText("");  
6311: cmbDateExpOD.addActionListener(new java.awt.event.ActionListener() {  
6312: public void actionPerformed(java.awt.event.ActionEvent evt) {  
6313: cmbDateExpODActionPerformed(evt);  
6314: }  
6315: });  
6316:  
6317: jLabel72.setText("Data:");  
6318:  
6319: btnExpODDate.setText("Exportar");  
6320: btnExpODDate.addActionListener(new java.awt.event.ActionListener() {  
6321: public void actionPerformed(java.awt.event.ActionEvent evt) {  
6322: btnExpODDateActionPerformed(evt);  
6323: }  
6324: });  
6325:  
6326: javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);  
6327: jPanel1.setLayout(jPanel1Layout);  
6328: jPanel1Layout.setHorizontalGroup(  
6329: jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6330: .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()  
6331: .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)  
6332: .addGroup(jPanel1Layout.createSequentialGroup()  
6333: .addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6334: .addComponent(jLabel4))  
6335: .addGroup(jPanel1Layout.createSequentialGroup()  
6336: .addGap(29, 29, 29)  
6337: .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)  
6338: .addComponent(jLabel35, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6339: .addGroup(jPanel1Layout.createSequentialGroup()  
6340: .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6341: .addComponent(jLabel72, javax.swing.GroupLayout.Alignment.TRAILING)  
6342: .addComponent(jLabel71))  
6343: .addGap(5, 5, 5)  
6344: .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6345: .addGroup(jPanel1Layout.createSequentialGroup()  
6346: .addComponent(cmbDateExpVol, javax.swing.GroupLayout.PREFERRED\_SIZE, 123, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6347: .addGap(18, 18, 18)  
6348: .addComponent(btnExpVolDate, javax.swing.GroupLayout.PREFERRED\_SIZE, 120, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6349: .addGroup(jPanel1Layout.createSequentialGroup()  
6350: .addComponent(cmbDateExpOD, javax.swing.GroupLayout.PREFERRED\_SIZE, 123, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6351: .addGap(18, 18, 18)  
6352: .addComponent(btnExpODDate, javax.swing.GroupLayout.PREFERRED\_SIZE, 120, javax.swing.GroupLayout.PREFERRED\_SIZE)))))))  
6353: .addGap(81, 81, 81))  
6354: );  
6355: jPanel1Layout.setVerticalGroup(  
6356: jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6357: .addGroup(jPanel1Layout.createSequentialGroup()  
6358: .addGap(25, 25, 25)  
6359: .addComponent(jLabel4)  
6360: .addGap(18, 18, 18)  
6361: .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6362: .addComponent(btnExpVolDate)  
6363: .addComponent(cmbDateExpVol, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6364: .addComponent(jLabel71))  
6365: .addGap(51, 51, 51)  
6366: .addComponent(jLabel35)  
6367: .addGap(18, 18, 18)  
6368: .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6369: .addComponent(cmbDateExpOD, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6370: .addComponent(jLabel72)  
6371: .addComponent(btnExpODDate))  
6372: .addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))  
6373: );  
6374:  
6375: javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);  
6376: jPanel2.setLayout(jPanel2Layout);  
6377: jPanel2Layout.setHorizontalGroup(  
6378: jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6379: .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()  
6380: .addContainerGap()  
6381: .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6382: .addComponent(jLabel3)  
6383: .addComponent(btnInDados))  
6384: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 1832, Short.MAX\_VALUE)  
6385: .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6386: .addGap(116, 116, 116))  
6387: );  
6388: jPanel2Layout.setVerticalGroup(  
6389: jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6390: .addGroup(jPanel2Layout.createSequentialGroup()  
6391: .addGap(26, 26, 26)  
6392: .addComponent(jLabel3)  
6393: .addGap(18, 18, 18)  
6394: .addComponent(btnInDados)  
6395: .addContainerGap(1415, Short.MAX\_VALUE))  
6396: .addGroup(jPanel2Layout.createSequentialGroup()  
6397: .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6398: .addGap(95, 95, 95))  
6399: );  
6400:  
6401: javax.swing.GroupLayout pnl\_envioLayout = new javax.swing.GroupLayout(pnl\_envio);  
6402: pnl\_envio.setLayout(pnl\_envioLayout);  
6403: pnl\_envioLayout.setHorizontalGroup(  
6404: pnl\_envioLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6405: .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnl\_envioLayout.createSequentialGroup()  
6406: .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6407: .addGap(315, 315, 315))  
6408: );  
6409: pnl\_envioLayout.setVerticalGroup(  
6410: pnl\_envioLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6411: .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6412: );  
6413:  
6414: tabRelatorio.addTab("captura e exportação de dados", pnl\_envio);  
6415:  
6416: jLabel38.setText("Data");  
6417:  
6418: cmbDataSumVol.addActionListener(new java.awt.event.ActionListener() {  
6419: public void actionPerformed(java.awt.event.ActionEvent evt) {  
6420: cmbDataSumVolActionPerformed(evt);  
6421: }  
6422: });  
6423:  
6424: chk\_exportadas\_sumVol.setText("Exportadas");  
6425:  
6426: chk\_nao\_exportadas\_sumVol.setText("Não exportadas");  
6427:  
6428: jLabel41.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6429: jLabel41.setText("P1");  
6430:  
6431: jLabel42.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6432: jLabel42.setText("P2");  
6433:  
6434: jLabel43.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6435: jLabel43.setText("P3");  
6436:  
6437: jLabel44.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6438: jLabel44.setText("M");  
6439:  
6440: jLabel45.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6441: jLabel45.setText("2CB");  
6442:  
6443: jLabel46.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6444: jLabel46.setText("3CB");  
6445:  
6446: jLabel47.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6447: jLabel47.setText("4CB");  
6448:  
6449: jLabel48.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6450: jLabel48.setText("2C");  
6451:  
6452: jLabel49.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6453: jLabel49.setText("3C");  
6454:  
6455: jLabel50.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6456: jLabel50.setText("4C");  
6457:  
6458: jLabel51.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6459: jLabel51.setText("4CD");  
6460:  
6461: jLabel52.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6462: jLabel52.setText("3D");  
6463:  
6464: txtSumVolP1.setEditable(false);  
6465: txtSumVolP1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6466:  
6467: txtSumVolP2.setEditable(false);  
6468: txtSumVolP2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6469:  
6470: txtSumVolP3.setEditable(false);  
6471: txtSumVolP3.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6472:  
6473: txtSumVolM.setEditable(false);  
6474: txtSumVolM.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6475:  
6476: txtSumVolO1.setEditable(false);  
6477: txtSumVolO1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6478:  
6479: txtSumVolO2.setEditable(false);  
6480: txtSumVolO2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6481:  
6482: txtSumVolO3.setEditable(false);  
6483: txtSumVolO3.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6484:  
6485: txtSumVolC1.setEditable(false);  
6486: txtSumVolC1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6487:  
6488: txtSumVolC2.setEditable(false);  
6489: txtSumVolC2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6490:  
6491: txtSumVolC3.setEditable(false);  
6492: txtSumVolC3.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6493:  
6494: txtSumVolC4.setEditable(false);  
6495: txtSumVolC4.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6496:  
6497: txtSumVolC5.setEditable(false);  
6498: txtSumVolC5.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6499:  
6500: jLabel53.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6501: jLabel53.setText("2S1");  
6502:  
6503: jLabel54.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6504: jLabel54.setText("2S2");  
6505:  
6506: jLabel55.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6507: jLabel55.setText("2S3");  
6508:  
6509: jLabel56.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6510: jLabel56.setText("3S1");  
6511:  
6512: jLabel57.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6513: jLabel57.setText("3S2");  
6514:  
6515: jLabel58.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6516: jLabel58.setText("3S3");  
6517:  
6518: jLabel59.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6519: jLabel59.setText("3T4");  
6520:  
6521: jLabel60.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6522: jLabel60.setText("3T6");  
6523:  
6524: jLabel61.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6525: jLabel61.setText("3R6");  
6526:  
6527: jLabel62.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6528: jLabel62.setText("3V5");  
6529:  
6530: jLabel63.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6531: jLabel63.setText("3M6");  
6532:  
6533: jLabel64.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6534: jLabel64.setText("3Q4");  
6535:  
6536: jLabel65.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6537: jLabel65.setText("2C2");  
6538:  
6539: jLabel66.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6540: jLabel66.setText("2C3");  
6541:  
6542: jLabel67.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6543: jLabel67.setText("3C2");  
6544:  
6545: jLabel68.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6546: jLabel68.setText("3C3");  
6547:  
6548: jLabel69.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
6549: jLabel69.setText("3D4");  
6550:  
6551: txtSumVolS1.setEditable(false);  
6552: txtSumVolS1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6553:  
6554: txtSumVolS2.setEditable(false);  
6555: txtSumVolS2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6556:  
6557: txtSumVolS3.setEditable(false);  
6558: txtSumVolS3.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6559:  
6560: txtSumVolS4.setEditable(false);  
6561: txtSumVolS4.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6562:  
6563: txtSumVolS5.setEditable(false);  
6564: txtSumVolS5.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6565:  
6566: txtSumVolS6.setEditable(false);  
6567: txtSumVolS6.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6568:  
6569: txtSumVolSE1.setEditable(false);  
6570: txtSumVolSE1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6571:  
6572: txtSumVolSE2.setEditable(false);  
6573: txtSumVolSE2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6574:  
6575: txtSumVolSE3.setEditable(false);  
6576: txtSumVolSE3.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6577:  
6578: txtSumVolSE4.setEditable(false);  
6579: txtSumVolSE4.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6580:  
6581: txtSumVolSE5.setEditable(false);  
6582: txtSumVolSE5.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6583:  
6584: txtSumVolR1.setEditable(false);  
6585: txtSumVolR1.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6586:  
6587: txtSumVolR2.setEditable(false);  
6588: txtSumVolR2.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6589:  
6590: txtSumVolR3.setEditable(false);  
6591: txtSumVolR3.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6592:  
6593: txtSumVolR4.setEditable(false);  
6594: txtSumVolR4.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6595:  
6596: txtSumVolR5.setEditable(false);  
6597: txtSumVolR5.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6598:  
6599: txtSumVolR6.setEditable(false);  
6600: txtSumVolR6.setHorizontalAlignment(javax.swing.JTextField.CENTER);  
6601:  
6602: jLabel70.setText("Totais por tipo de veículo");  
6603:  
6604: tblSumVol.setModel(new javax.swing.table.DefaultTableModel(  
6605: new Object [][] {  
6606: {"P1", null, null, null, null, null, null, null, null, null, null, null, null},  
6607: {"P3", null, null, null, null, null, null, null, null, null, null, null, null},  
6608: {"P2", null, null, null, null, null, null, null, null, null, null, null, null},  
6609: {"M", null, null, null, null, null, null, null, null, null, null, null, null},  
6610: {"2CB", null, null, null, null, null, null, null, null, null, null, null, null},  
6611: {"3CB", null, null, null, null, null, null, null, null, null, null, null, null},  
6612: {"4CB", null, null, null, null, null, null, null, null, null, null, null, null},  
6613: {"2C", null, null, null, null, null, null, null, null, null, null, null, null},  
6614: {"3C", null, null, null, null, null, null, null, null, null, null, null, null},  
6615: {"4C", null, null, null, null, null, null, null, null, null, null, null, null},  
6616: {"4CD", null, null, null, null, null, null, null, null, null, null, null, null},  
6617: {"3D", null, null, null, null, null, null, null, null, null, null, null, null},  
6618: {"2S1", null, null, null, null, null, null, null, null, null, null, null, null},  
6619: {"2S2", null, null, null, null, null, null, null, null, null, null, null, null},  
6620: {"2S3", null, null, null, null, null, null, null, null, null, null, null, null},  
6621: {"3S1", null, null, null, null, null, null, null, null, null, null, null, null},  
6622: {"3S2", null, null, null, null, null, null, null, null, null, null, null, null},  
6623: {"3S3", null, null, null, null, null, null, null, null, null, null, null, null},  
6624: {"3T4", null, null, null, null, null, null, null, null, null, null, null, null},  
6625: {"3T6", null, null, null, null, null, null, null, null, null, null, null, null},  
6626: {"3R6", null, null, null, null, null, null, null, null, null, null, null, null},  
6627: {"3V5", null, null, null, null, null, null, null, null, null, null, null, null},  
6628: {"3M6", null, null, null, null, null, null, null, null, null, null, null, null},  
6629: {"3Q4", null, null, null, null, null, null, null, null, null, null, null, null},  
6630: {"2C2", null, null, null, null, null, null, null, null, null, null, null, null},  
6631: {"2C3", null, null, null, null, null, null, null, null, null, null, null, null},  
6632: {"3C2", null, null, null, null, null, null, null, null, null, null, null, null},  
6633: {"3C3", null, null, null, null, null, null, null, null, null, null, null, null},  
6634: {"3D4", null, null, null, null, null, null, null, null, null, null, null, null}  
6635: },  
6636: new String [] {  
6637: "Tipos", "0:00 - 2:00", "2:00 - 4:00", "4:00 - 6:00", "6:00 - 8:00", "8:00 - 10:00", "10:00 - 12:00", "12:00 - 14:00", "14:00 - 16:00", "16:00 - 18:00", "18:00 - 20:00", "20:00 - 22:00", "22:00 - 24:00"  
6638: }  
6639: ) {  
6640: Class[] types = new Class [] {  
6641: java.lang.String.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class, java.lang.Integer.class  
6642: };  
6643: boolean[] canEdit = new boolean [] {  
6644: false, false, false, false, false, false, false, false, false, false, false, false, false  
6645: };  
6646:  
6647: public Class getColumnClass(int columnIndex) {  
6648: return types [columnIndex];  
6649: }  
6650:  
6651: public boolean isCellEditable(int rowIndex, int columnIndex) {  
6652: return canEdit [columnIndex];  
6653: }  
6654: });  
6655: jScrollPane2.setViewportView(tblSumVol);  
6656:  
6657: javax.swing.GroupLayout pnl\_sumario\_volumetricaLayout = new javax.swing.GroupLayout(pnl\_sumario\_volumetrica);  
6658: pnl\_sumario\_volumetrica.setLayout(pnl\_sumario\_volumetricaLayout);  
6659: pnl\_sumario\_volumetricaLayout.setHorizontalGroup(  
6660: pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6661: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6662: .addContainerGap()  
6663: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6664: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6665: .addComponent(jLabel42, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6666: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6667: .addComponent(txtSumVolP2, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6668: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6669: .addComponent(jLabel44, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6670: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6671: .addComponent(txtSumVolM, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6672: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6673: .addComponent(jLabel45, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6674: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6675: .addComponent(txtSumVolO1, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6676: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6677: .addComponent(jLabel46, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6678: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6679: .addComponent(txtSumVolO2, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6680: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6681: .addComponent(jLabel47, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6682: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6683: .addComponent(txtSumVolO3, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6684: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6685: .addComponent(jLabel48, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6686: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6687: .addComponent(txtSumVolC1, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6688: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6689: .addComponent(jLabel49, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6690: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6691: .addComponent(txtSumVolC2, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6692: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6693: .addComponent(jLabel50, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6694: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6695: .addComponent(txtSumVolC3, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6696: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6697: .addComponent(jLabel51, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6698: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6699: .addComponent(txtSumVolC4, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6700: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6701: .addComponent(jLabel52, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6702: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6703: .addComponent(txtSumVolC5, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6704: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6705: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)  
6706: .addComponent(jLabel41, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)  
6707: .addComponent(jLabel43, javax.swing.GroupLayout.DEFAULT\_SIZE, 34, Short.MAX\_VALUE))  
6708: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6709: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6710: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6711: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6712: .addComponent(txtSumVolP1, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6713: .addComponent(txtSumVolP3, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6714: .addGap(83, 83, 83)  
6715: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6716: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6717: .addComponent(jLabel69, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6718: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6719: .addComponent(txtSumVolR6, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6720: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6721: .addComponent(jLabel68, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6722: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6723: .addComponent(txtSumVolR5, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6724: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6725: .addComponent(jLabel67, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6726: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6727: .addComponent(txtSumVolR4, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6728: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6729: .addComponent(jLabel66, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6730: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6731: .addComponent(txtSumVolR3, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6732: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6733: .addComponent(jLabel65, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6734: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6735: .addComponent(txtSumVolR2, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6736: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6737: .addComponent(jLabel64, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6738: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6739: .addComponent(txtSumVolR1, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6740: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6741: .addComponent(jLabel63, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6742: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6743: .addComponent(txtSumVolSE5, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6744: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6745: .addComponent(jLabel62, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6746: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6747: .addComponent(txtSumVolSE4, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6748: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6749: .addComponent(jLabel61, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6750: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6751: .addComponent(txtSumVolSE3, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6752: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6753: .addComponent(jLabel60, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6754: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6755: .addComponent(txtSumVolSE2, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6756: .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6757: .addComponent(jLabel59, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6758: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6759: .addComponent(txtSumVolSE1, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6760: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6761: .addComponent(jLabel58, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6762: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6763: .addComponent(txtSumVolS6, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6764: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6765: .addComponent(jLabel57, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6766: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6767: .addComponent(txtSumVolS5, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6768: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6769: .addComponent(jLabel56, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6770: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6771: .addComponent(txtSumVolS4, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6772: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6773: .addComponent(jLabel55, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6774: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6775: .addComponent(txtSumVolS3, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6776: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6777: .addComponent(jLabel54, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6778: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6779: .addComponent(txtSumVolS2, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6780: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6781: .addComponent(jLabel53, javax.swing.GroupLayout.PREFERRED\_SIZE, 34, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6782: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6783: .addComponent(txtSumVolS1, javax.swing.GroupLayout.PREFERRED\_SIZE, 52, javax.swing.GroupLayout.PREFERRED\_SIZE))))  
6784: .addComponent(jLabel70)))  
6785: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6786: .addComponent(jLabel38)  
6787: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6788: .addComponent(cmbDataSumVol, javax.swing.GroupLayout.PREFERRED\_SIZE, 167, javax.swing.GroupLayout.PREFERRED\_SIZE)))  
6789: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6790: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6791: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6792: .addComponent(chk\_exportadas\_sumVol)  
6793: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
6794: .addComponent(chk\_nao\_exportadas\_sumVol)  
6795: .addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))  
6796: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6797: .addGap(61, 61, 61)  
6798: .addComponent(jScrollPane2, javax.swing.GroupLayout.DEFAULT\_SIZE, 2555, Short.MAX\_VALUE)  
6799: .addContainerGap())))  
6800: );  
6801: pnl\_sumario\_volumetricaLayout.setVerticalGroup(  
6802: pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6803: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6804: .addContainerGap()  
6805: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6806: .addComponent(jLabel38)  
6807: .addComponent(cmbDataSumVol, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6808: .addComponent(chk\_exportadas\_sumVol)  
6809: .addComponent(chk\_nao\_exportadas\_sumVol))  
6810: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6811: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6812: .addGap(24, 24, 24)  
6813: .addComponent(jLabel70)  
6814: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
6815: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6816: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6817: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6818: .addComponent(jLabel41)  
6819: .addComponent(txtSumVolP1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6820: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6821: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6822: .addComponent(jLabel43)  
6823: .addComponent(txtSumVolP3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6824: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6825: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6826: .addComponent(jLabel42)  
6827: .addComponent(txtSumVolP2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6828: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6829: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6830: .addComponent(jLabel44)  
6831: .addComponent(txtSumVolM, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6832: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6833: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6834: .addComponent(jLabel45)  
6835: .addComponent(txtSumVolO1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6836: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6837: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6838: .addComponent(jLabel46)  
6839: .addComponent(txtSumVolO2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6840: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6841: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6842: .addComponent(jLabel47)  
6843: .addComponent(txtSumVolO3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6844: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6845: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6846: .addComponent(jLabel48)  
6847: .addComponent(txtSumVolC1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6848: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6849: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6850: .addComponent(jLabel49)  
6851: .addComponent(txtSumVolC2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6852: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6853: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6854: .addComponent(jLabel50)  
6855: .addComponent(txtSumVolC3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6856: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6857: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6858: .addComponent(jLabel51)  
6859: .addComponent(txtSumVolC4, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6860: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6861: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6862: .addComponent(jLabel52)  
6863: .addComponent(txtSumVolC5, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)))  
6864: .addGroup(pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6865: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6866: .addComponent(jLabel53)  
6867: .addComponent(txtSumVolS1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6868: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6869: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6870: .addComponent(jLabel54)  
6871: .addComponent(txtSumVolS2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6872: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6873: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6874: .addComponent(jLabel55)  
6875: .addComponent(txtSumVolS3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6876: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6877: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6878: .addComponent(jLabel56)  
6879: .addComponent(txtSumVolS4, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6880: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6881: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6882: .addComponent(jLabel57)  
6883: .addComponent(txtSumVolS5, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6884: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6885: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6886: .addComponent(jLabel58)  
6887: .addComponent(txtSumVolS6, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6888: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6889: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6890: .addComponent(jLabel59)  
6891: .addComponent(txtSumVolSE1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6892: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6893: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6894: .addComponent(jLabel60)  
6895: .addComponent(txtSumVolSE2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6896: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6897: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6898: .addComponent(jLabel61)  
6899: .addComponent(txtSumVolSE3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6900: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6901: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6902: .addComponent(jLabel62)  
6903: .addComponent(txtSumVolSE4, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6904: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6905: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6906: .addComponent(jLabel63)  
6907: .addComponent(txtSumVolSE5, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6908: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6909: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6910: .addComponent(jLabel64)  
6911: .addComponent(txtSumVolR1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6912: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6913: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6914: .addComponent(jLabel65)  
6915: .addComponent(txtSumVolR2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6916: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6917: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6918: .addComponent(jLabel66)  
6919: .addComponent(txtSumVolR3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6920: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6921: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6922: .addComponent(jLabel67)  
6923: .addComponent(txtSumVolR4, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6924: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6925: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6926: .addComponent(jLabel68)  
6927: .addComponent(txtSumVolR5, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
6928: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6929: .addGroup(pnl\_sumario\_volumetricaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)  
6930: .addComponent(jLabel69)  
6931: .addComponent(txtSumVolR6, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))))  
6932: .addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))  
6933: .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnl\_sumario\_volumetricaLayout.createSequentialGroup()  
6934: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
6935: .addComponent(jScrollPane2)  
6936: .addContainerGap())))  
6937: );  
6938:  
6939: tabRelatorio.addTab("Sumário Volumétrica", pnl\_sumario\_volumetrica);  
6940:  
6941: relatorio.setModel(new javax.swing.table.DefaultTableModel(  
6942: new Object [][] {  
6943: {null, null, null, null},  
6944: {null, null, null, null},  
6945: {null, null, null, null},  
6946: {null, null, null, null}  
6947: },  
6948: new String [] {  
6949: "Title 1", "Title 2", "Title 3", "Title 4"  
6950: }  
6951: ));  
6952: jScrollPane5.setViewportView(relatorio);  
6953:  
6954: javax.swing.GroupLayout pnl\_relatorioLayout = new javax.swing.GroupLayout(pnl\_relatorio);  
6955: pnl\_relatorio.setLayout(pnl\_relatorioLayout);  
6956: pnl\_relatorioLayout.setHorizontalGroup(  
6957: pnl\_relatorioLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6958: .addGroup(pnl\_relatorioLayout.createSequentialGroup()  
6959: .addGap(38, 38, 38)  
6960: .addComponent(odStatus, javax.swing.GroupLayout.PREFERRED\_SIZE, 169, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6961: .addContainerGap(2712, Short.MAX\_VALUE))  
6962: .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnl\_relatorioLayout.createSequentialGroup()  
6963: .addContainerGap()  
6964: .addComponent(jScrollPane5))  
6965: );  
6966: pnl\_relatorioLayout.setVerticalGroup(  
6967: pnl\_relatorioLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
6968: .addGroup(pnl\_relatorioLayout.createSequentialGroup()  
6969: .addGap(14, 14, 14)  
6970: .addComponent(odStatus, javax.swing.GroupLayout.PREFERRED\_SIZE, 18, javax.swing.GroupLayout.PREFERRED\_SIZE)  
6971: .addGap(18, 18, 18)  
6972: .addComponent(jScrollPane5, javax.swing.GroupLayout.DEFAULT\_SIZE, 1437, Short.MAX\_VALUE)  
6973: .addContainerGap())  
6974: );  
6975:  
6976: tabRelatorio.addTab("Sumário OD", pnl\_relatorio);  
6977:  
6978: gridBagConstraints = new java.awt.GridBagConstraints();  
6979: gridBagConstraints.gridx = 0;  
6980: gridBagConstraints.gridy = 1;  
6981: gridBagConstraints.fill = java.awt.GridBagConstraints.BOTH;  
6982: gridBagConstraints.ipady = 229;  
6983: gridBagConstraints.weightx = 1.0;  
6984: gridBagConstraints.weighty = 1.0;  
6985: getContentPane().add(tabRelatorio, gridBagConstraints);  
6986: tabRelatorio.getAccessibleContext().setAccessibleName("tabna1");  
6987:  
6988: pack();  
6989: }// </editor-fold>//GEN-END:initComponents  
6990:   
6991: /\*\*  
6992: \* Método disparado quando o usuário escolhe o sentido BA no formulário.  
6993: \* Verifica se os dados informados já são suficientes para identificar o registro.  
6994: \* @param evt evento de clique na opção do sentido BA.  
6995: \*/  
6996: private void rdo\_SentidoBAActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_rdo\_SentidoBAActionPerformed  
6997: askForDataRetrieve();  
6998: }// GEN-LAST:event\_rdo\_SentidoBAActionPerformed  
6999:  
7000: /\*\*  
7001: \* Método disparado quando o usuário escolhe o sentido AB no formulário.  
7002: \* Verifica se os dados informados já são suficientes para identificar o registro.  
7003: \* @param evt evento de clique na opção do sentido AB.  
7004: \*/  
7005: private void rdo\_SentidoABActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_rdo\_SentidoABActionPerformed  
7006: askForDataRetrieve();  
7007: }// GEN-LAST:event\_rdo\_SentidoABActionPerformed  
7008:  
7009: /\*\*  
7010: \* Método disparado quano o usuário clica no botão Apagar.  
7011: \* Na hipótese de dados de uma pesquisa volumétrica terem sido recuperados do banco de dados, o botão de apagar fica   
7012: \* visível para o usuário. Ao clicar nesse botão, o usuário recebe um diálogo de confirmação informando que a operação de   
7013: \* de apagar os dados não pode ser revertida. No caso de uma resposta afirmativa, a senha de operação para o posto atual   
7014: \* é solicitada. Após as confirmações, o método apaga o registro do banco de dados.  
7015: \* @param evt evento de clique no botão  
7016: \*/  
7017: private void btnApagarActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_btnApagarActionPerformed  
7018: int returnedValue = JOptionPane.showConfirmDialog(Janela.this,  
7019: "Os dados apagados não poderão ser recuperados.\nVocê deseja continuar?", "Apagar dados.",  
7020: JOptionPane.YES\_NO\_OPTION, JOptionPane.WARNING\_MESSAGE);  
7021: if (returnedValue == JOptionPane.YES\_OPTION) {  
7022: // solicita senha  
7023: JPanel panel = new JPanel();  
7024: JLabel label = new JLabel("Entre com o senha para apagar o registro.");  
7025: JPasswordField pass = new JPasswordField(10);  
7026: panel.add(label);  
7027: panel.add(pass);  
7028: String[] options = new String[] { "Cancel", "OK" };  
7029: int option = JOptionPane.showOptionDialog(Janela.this, panel, "Senha para apagar registro.", JOptionPane.NO\_OPTION,  
7030: JOptionPane.WARNING\_MESSAGE, null, options, options[1]);  
7031: if (option == 1) { // ok button  
7032: LoginController controller = new LoginController("/logins/users.json");  
7033: if (controller.validateLogin(Concentrador.posto, String.copyValueOf(pass.getPassword()))) {  
7034: try {  
7035: myDB database = Concentrador.database;  
7036: database.deletePV(makePVKey());  
7037: this.clearForm();  
7038: //btnApagar.setVisible(false);  
7039: JOptionPane.showMessageDialog(Janela.this, "Operação concluída", "Operação concluída",  
7040: JOptionPane.INFORMATION\_MESSAGE);  
7041:  
7042: } catch (Exception e) {  
7043: logger.error("Erro ao tentar apagar um registro.", e);  
7044: JOptionPane.showMessageDialog(Janela.this, "Erro ao tentar apagar um registro:\n" + e.getMessage(),  
7045: "Erro de conexão com o banco de dados.", JOptionPane.ERROR\_MESSAGE);  
7046: }  
7047:  
7048: } else {  
7049: JOptionPane.showMessageDialog(Janela.this, "A senha informada não confere.", "Senha incorreta",  
7050: JOptionPane.ERROR\_MESSAGE);  
7051: }  
7052: }  
7053: } else {  
7054: return; // sai sem apagar  
7055: }  
7056: }// GEN-LAST:event\_btnApagarActionPerformed  
7057:  
7058: /\*\*  
7059: \* Monta e exibe a tabela com os dados do sumário das pesquisas do tipo Origem/Destino já realizadas.  
7060: \*/  
7061: private synchronized void relatorioOD() {  
7062: odStatus.setText("Carregando dados ...");  
7063: Vector<String> cols = null;  
7064: Vector<String> rows = null;  
7065: DefaultTableModel model = new DefaultTableModel() {  
7066: public boolean isCellEditable(int row, int column) {  
7067: return false;  
7068: }  
7069: };  
7070:  
7071: try {  
7072: cols = Concentrador.database.fetchReportODColumns();  
7073: rows = Concentrador.database.fetchReportODRows(Concentrador.getPostoInt());  
7074: } catch (Exception ex) {  
7075: logger.error("Error ao ler dados para relatorio OD.", ex);  
7076: }  
7077: if (cols != null) {  
7078: relatorio.setAutoCreateColumnsFromModel(true);  
7079: model.setColumnCount(cols.size());  
7080: model.setColumnIdentifiers(cols);  
7081: model.addColumn("Total");  
7082: HashMap<String, Integer> totals = new HashMap<String, Integer>();  
7083: for (String ipad : cols) {  
7084: if (!ipad.equals("") && !ipad.equals("Total")) {  
7085: totals.put(ipad, 0);  
7086: }  
7087: }  
7088: for (String date : rows) {  
7089: try {  
7090: Vector<String> rowData = new Vector<String>();  
7091: Map<String, Integer> ipads = Concentrador.database.fetchReportODData(date,  
7092: Integer.parseInt(Concentrador.posto));  
7093: rowData.add(date);  
7094: int total = 0;  
7095: if (ipads != null) {  
7096: for (String ipad : cols) {  
7097: if (!ipad.equals("") && !ipad.equals("Total")) {  
7098: Integer counter = ipads.get(ipad);  
7099: if (counter != null) {  
7100: rowData.add(String.valueOf(counter));  
7101: total += counter;  
7102: int tmp = totals.get(ipad);  
7103: totals.put(ipad, tmp + counter);  
7104: } else  
7105: rowData.add("0");  
7106: }  
7107: }  
7108: }  
7109: rowData.add(String.valueOf(total));  
7110: model.addRow(rowData);  
7111: } catch (Exception e) {  
7112: e.printStackTrace();  
7113: }  
7114: }  
7115:  
7116: Vector<String> totalsRow = new Vector<String>();  
7117: totalsRow.add("Total");  
7118: int sumTotal = 0;  
7119: for (String ipad : cols) {  
7120: if (!ipad.equals("") && !ipad.equals("Total")) {  
7121: totalsRow.add(String.valueOf(totals.get(ipad)));  
7122: sumTotal += totals.get(ipad);  
7123: }  
7124: }  
7125: totalsRow.add(String.valueOf(sumTotal));  
7126: model.addRow(totalsRow);  
7127: model.fireTableStructureChanged();  
7128: relatorio.setModel(model);  
7129: relatorio.setCellSelectionEnabled(false);  
7130: relatorio.setIntercellSpacing(new Dimension(0, 2));  
7131: odStatus.setText("");  
7132: }  
7133: }  
7134:  
7135: /\*\*  
7136: \* Verifica se o painel exibido mudou e recarrega os componentes necessários em caso de mudança.  
7137: \* @param evt mudança do painel exibido  
7138: \*/  
7139: private void tabRelatorioStateChanged(javax.swing.event.ChangeEvent evt) {// GEN-FIRST:event\_tabRelatorioStateChanged  
7140: if (tabRelatorio.getSelectedComponent().equals(pnl\_sumario\_volumetrica)) {  
7141: this.fillCmbDatesSumVol();  
7142: } else if (tabRelatorio.getSelectedComponent().equals(pnl\_relatorio)) {  
7143: this.relatorioOD();  
7144: } else if (tabRelatorio.getSelectedComponent().equals(pnl\_envio)) {  
7145: this.fillCmbDatesExp();  
7146:  
7147: }  
7148: }// GEN-LAST:event\_tabRelatorioStateChanged  
7149:  
7150: /\*\*  
7151: \* Filtra o sumário das pesquisas volumétricas pela data selecionada na respectiva \_caixa de seleção\_.  
7152: \* @param evt evento de seleção de uma data  
7153: \*/  
7154: private void cmbDataSumVolActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_cmbDataSumVolActionPerformed  
7155: try {  
7156: String dateStr = getSelectedDateFromCombo(cmbDataSumVol);  
7157: if (dateStr != null && dateStr.equals(TODAS)){  
7158: dateStr = null;  
7159: }  
7160:   
7161: if (dateStr == null && cmbDataSumVol.getSelectedIndex() == 0){ // "--" option selected so clean the data fields  
7162: this.clearSumVolData();  
7163: this.clearSumVolTable();  
7164: }else{  
7165: this.setSumVolData(dateStr);  
7166: this.setSumVolTable(dateStr);  
7167: }  
7168: } catch (Exception e) {  
7169: logger.error("Erro na conversão de datas para consulta e construção do sumário da pesquisa volumétrica.", e);  
7170: }  
7171:  
7172: }// GEN-LAST:event\_cmbDataSumVolActionPerformed  
7173:  
7174: @Deprecated  
7175: private void cmbDateExpVolActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_cmbDateExpVolActionPerformed  
7176: // TO-DO add your handling code here:  
7177: }// GEN-LAST:event\_cmbDateExpVolActionPerformed  
7178:  
7179: /\*\*  
7180: \* Método disparado pelo clique no botão para exportar pesquisas Origem/Destino.  
7181: \* @param evt evento de clique no botão  
7182: \*/  
7183: private void btnExpODDateActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_btnExpODDateActionPerformed  
7184: exportData(JSONExporter.DbTable.OD);  
7185: }// GEN-LAST:event\_btnExpODDateActionPerformed  
7186:  
7187: @Deprecated  
7188: private void cmbDateExpODActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_cmbDateExpODActionPerformed  
7189: // TO-DO add your handling code here:  
7190: }// GEN-LAST:event\_cmbDateExpODActionPerformed  
7191:  
7192: /\*\*  
7193: \* Método disparado quando o usuário escolhe uma data para a pesquisa volumétrica.  
7194: \* Verifica se os dados informados já são suficientes para identificar o registro.  
7195: \* @param evt evento de escolha de data na \_caixa de seleção\_.  
7196: \*/  
7197: private void cmbDataActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_cmbDataActionPerformed  
7198: askForDataRetrieve();  
7199: }// GEN-LAST:event\_cmbDataActionPerformed  
7200:  
7201: /\*\*  
7202: \* Método disparado pelo botão de importar dados.  
7203: \*   
7204: \* @param evt evento de clique no botão.  
7205: \*/  
7206: private void btnInDadosActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_btnInDadosActionPerformed  
7207: FileSystemView filesys = FileSystemView.getFileSystemView();  
7208: // File[] roots = filesys.getRoots();  
7209: File desktop = filesys.getHomeDirectory();  
7210:  
7211: File inputFolder = new File(desktop.getAbsolutePath() + File.separator  
7212: + Concentrador.configuration.getProperty("dataInputFolder") + File.separator);  
7213:  
7214: logger.debug(String.format("Procurando arquivos para importação em %s", inputFolder.getAbsolutePath()));  
7215:  
7216: if (!inputFolder.exists()) {  
7217: inputFolder.mkdir();  
7218: logger.debug(String.format("Diretório criado: %s", inputFolder.getAbsolutePath()));  
7219: JOptionPane.showMessageDialog(this,  
7220: "O diretório \"" + inputFolder.getName() + "\" foi criado na área de trabalho.\n"  
7221: + "O Concentrador de dados vasculha esse diretório procurando pelos arquivos.\n"  
7222: + "Armazene nele os arquivos que deseja importar.",  
7223: "Diretório de importação criado.", JOptionPane.INFORMATION\_MESSAGE);  
7224: }  
7225:  
7226: try {  
7227: DBFileImporter dbfile = new DBFileImporter(inputFolder);  
7228: dbfile.readNewFiles();  
7229: String modoOperacao = (Concentrador.treinamento)? "treinamento" : "produção";   
7230: JOptionPane.showMessageDialog(this, "Total de registros inseridos para o posto " + Concentrador.posto + ", no modo de " + modoOperacao + ": " + Integer.toString(dbfile.getCounter()));  
7231: this.fillCmbDatesExp();  
7232: } catch (Exception e) {  
7233: logger.error(String.format("Erro ao importar arquivos em %s", inputFolder.getAbsolutePath()), e);  
7234: }  
7235:  
7236: }  
7237:   
7238: /\*\*  
7239: \* Conferte a data selecionada numa \_caixa de selecao\_ do formato de exibição para o formato SQL.  
7240: \*   
7241: \* @param combo \_Caixa de seleção\_ contendo as datas  
7242: \* @return data no formato SQL  
7243: \* @throws ParseException   
7244: \*/  
7245: private String getSelectedDateFromCombo(javax.swing.JComboBox<String> combo) throws ParseException {  
7246: if (combo.getSelectedItem() == null) {  
7247: return null;  
7248: }  
7249: String dateStr = combo.getSelectedItem().toString();  
7250: if (dateStr.equals("--")) {  
7251: return null;  
7252: }  
7253: if (dateStr.equals(TODAS)){  
7254: return TODAS;  
7255: }  
7256: return Util.sdfSQL.format(Util.sdfBrazil.parse(dateStr));  
7257: }  
7258:  
7259: /\*\*  
7260: \* Constrói o nome do arquivo de exportação de dados.  
7261: \*   
7262: \* O nome do arquivo é formado pelo tipo de pesquisa [OD | VOL], concatenado com o número do posto e   
7263: \* a data selecionada para exportação.  
7264: \*   
7265: \* @param t Tabela do banco de dados relativa ao tipo de pesquisa [PV | OD]  
7266: \* @param date data selecionada para exportação  
7267: \* @return nome do arquivo  
7268: \* @throws ParseException   
7269: \*/  
7270: private String buildExportName(JSONExporter.DbTable t, String date) throws ParseException {  
7271: String name = "";  
7272: if (t.toString().equals(JSONExporter.DbTable.OD.toString()))  
7273: name = "OD-";  
7274: else if (t.toString().equals(JSONExporter.DbTable.PV.toString()))  
7275: name = "VOL-";  
7276: name += Concentrador.posto + "-" + date + ".zip";  
7277: return name;  
7278: }  
7279:  
7280: /\*\*  
7281: \* Método de exportação dos dados.  
7282: \*   
7283: \* Exibe ao usuário um diálogo para escolha do diretório onde será salvo o arquivo com os dados exportados.  
7284: \* Com base na data selecionada na tela e no tipo de dados recebido como entrada, monta e salva o arquivo no   
7285: \* local escolhido.  
7286: \*   
7287: \* @param t Tabela do banco de dados relativa ao tipo de pesquisa [PV | OD]  
7288: \*/  
7289: private void exportData(JSONExporter.DbTable t) {  
7290: try {  
7291:  
7292: String date = null;  
7293: if (t.toString().equals(JSONExporter.DbTable.OD.toString()))  
7294: date = getSelectedDateFromCombo(cmbDateExpOD);  
7295: else if (t.toString().equals(JSONExporter.DbTable.PV.toString()))  
7296: date = getSelectedDateFromCombo(cmbDateExpVol);  
7297:  
7298: if(date != null){  
7299:   
7300: exporterFileChooser.setSelectedFile(new File(buildExportName(t, date)));  
7301:  
7302: int returnVal = exporterFileChooser.showSaveDialog(this);  
7303:  
7304: if (returnVal == javax.swing.JFileChooser.APPROVE\_OPTION) {  
7305: JSONExporter exporter = new JSONExporter(exporterFileChooser.getSelectedFile(), t, this);  
7306: exporter.export(Integer.parseInt(Concentrador.posto), date);  
7307: }  
7308: }  
7309: } catch (ParseException e) {  
7310: logger.info("Erro na conversão de datas para exportar pesquisa volumétrica.", e);  
7311: JOptionPane.showMessageDialog(Janela.this, "Error de conversão de data.", "Formatado de data.",  
7312: JOptionPane.ERROR\_MESSAGE);  
7313: }  
7314: }  
7315:  
7316: /\*\*  
7317: \* Método disparado pelo clique no botão para exportar pesquisas volumétricas.  
7318: \* @param evt evento de clique no botão.  
7319: \*/  
7320: private void btnExpVolDateActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_btnVolNotSentActionPerformed  
7321: exportData(JSONExporter.DbTable.PV);  
7322: }  
7323:  
7324: // private void btnODNotSentActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_btnODNotSentActionPerformed  
7325: // exporterFileChooser.setSelectedFile(new File("novos\_od.zip"));  
7326: // int returnVal = exporterFileChooser.showSaveDialog(this);  
7327: //  
7328: // if (returnVal == javax.swing.JFileChooser.APPROVE\_OPTION) {  
7329: // JSONExporter exporter = new JSONExporter(exporterFileChooser.getSelectedFile(), JSONExporter.DbTable.OD, this);  
7330: // exporter.export(Integer.parseInt(Concentrador.posto));  
7331: // }  
7332: // }// GEN-LAST:event\_btnODNotSentActionPerformed  
7333:  
7334: /\*\*  
7335: \* Método disparado quando o usuário clique no botão para salvar uma pesquisa volumétrica.  
7336: \*   
7337: \* O método verifica e valida os campos do formulário. Constrói a representação dos dados no formato a ser armazenado no banco   
7338: \* de dados, verifica se já existe um registro com os mesmos dados de identificação (posto, data, hora, sentido) e salva o  
7339: \* registro no banco.  
7340: \* @param evt evento de clique no botão.  
7341: \*/  
7342: private void btnSalvarFormsActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_btnSalvarFormsActionPerformed  
7343: String posto = Concentrador.posto;  
7344: String pista;  
7345: String date = "";  
7346: String sentido;  
7347: int hora\_inicial;  
7348: String pesquisador1;  
7349: String pesquisador2;  
7350: String local;  
7351: myDB database = Concentrador.database;  
7352:  
7353: // valida preenchimento de dados  
7354: String emptyMessage = null;  
7355: boolean showEmptyMessage = false;  
7356: if (txtTotalLeves.getText().equals("0") && txtTotalPesados.getText().equals("0")) {  
7357: emptyMessage = "O formulário não contém dados de contagem.\n\n";  
7358: showEmptyMessage = true;  
7359: } else if (txtTotalLeves.getText().equals("0")) {  
7360: emptyMessage = "O formulário não contém dados de\ncontagem de veículos e caminhões leves.\n\n";  
7361: showEmptyMessage = true;  
7362: } else if (txtTotalPesados.getText().equals("0")) {  
7363: emptyMessage = "O formulário não contém dados de\ncontagem de caminhões pesados.\n\n";  
7364: showEmptyMessage = true;  
7365: }  
7366:  
7367: if (showEmptyMessage) {  
7368: int returnedValue = JOptionPane.showConfirmDialog(Janela.this, emptyMessage + "Você deseja gravá-lo assim mesmo?",  
7369: "Dados não preenchidos.", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE);  
7370: if (returnedValue == JOptionPane.NO\_OPTION) {  
7371: return;  
7372: }  
7373: }  
7374:  
7375: // validate pista  
7376: if (!rdo\_PistaDupla.isSelected() && !rdo\_PistaSimples.isSelected()) {  
7377: JOptionPane.showMessageDialog(Janela.this, "O tipo de pista precisa ser selecionado.",  
7378: "Tipo de pista não selecionado.", JOptionPane.ERROR\_MESSAGE);  
7379: return;  
7380: } else if (rdo\_PistaDupla.isSelected()) {  
7381: pista = "D";  
7382: } else {  
7383: pista = "S";  
7384: }  
7385:  
7386: // valida e formata data  
7387: if (cmbData.getSelectedIndex() == 0) {  
7388: JOptionPane.showMessageDialog(Janela.this, "A data do formulário precisa ser preenchida.", "Data não preenchida.",  
7389: JOptionPane.ERROR\_MESSAGE);  
7390: return;  
7391: } else {  
7392: try {  
7393: date = Util.sdfSQL.format(Util.sdfBrazil.parse(cmbData.getSelectedItem().toString()));  
7394: } catch (Exception e) {  
7395: logger.info("Erro na conversão de datas para salvar o registro da pesquisa volumétrica.", e);  
7396: }  
7397:  
7398: }  
7399:  
7400: // valida sentido  
7401: if (!rdo\_SentidoAB.isSelected() && !rdo\_SentidoBA.isSelected()) {  
7402: JOptionPane.showMessageDialog(Janela.this, "O sentido precisa ser selecionado.", "Sentido da pista.",  
7403: JOptionPane.ERROR\_MESSAGE);  
7404: return;  
7405: } else if (rdo\_SentidoAB.isSelected()) {  
7406: sentido = "AB";  
7407: } else {  
7408: sentido = "BA";  
7409: }  
7410:  
7411: // valida hora inicial  
7412: if (cmbHora.getSelectedItem() == null) {  
7413: JOptionPane.showMessageDialog(Janela.this, "Selecione a hora inicial.", "Hora inicial.", JOptionPane.ERROR\_MESSAGE);  
7414: return;  
7415: } else {  
7416: hora\_inicial = Integer.parseInt(cmbHora.getSelectedItem().toString());  
7417: }  
7418:  
7419: // valida pesquisador  
7420: if (txtPesquisador1.getText().trim().isEmpty()) {  
7421: JOptionPane.showMessageDialog(Janela.this, "A identificação do pesquisador\_1 não foi preenchida.", "Pesquisador.",  
7422: JOptionPane.ERROR\_MESSAGE);  
7423: return;  
7424: } else {  
7425: pesquisador1 = txtPesquisador1.getText();  
7426: }  
7427:  
7428: // valida pesquisador  
7429: if (txtPesquisador2.getText().trim().isEmpty()) {  
7430: JOptionPane.showMessageDialog(Janela.this, "A identificação do pesquisador\_2 não foi preenchida.", "Pesquisador.",  
7431: JOptionPane.ERROR\_MESSAGE);  
7432: return;  
7433: } else {  
7434: pesquisador2 = txtPesquisador2.getText();  
7435: }  
7436: // valida local  
7437: if (txtLocal.getText().trim().isEmpty()) {  
7438: JOptionPane.showMessageDialog(Janela.this, "O local da pesquisa precisa ser preenchido.",  
7439: "Preenchimento do local da pesquisa.", JOptionPane.ERROR\_MESSAGE);  
7440: return;  
7441: } else {  
7442: local = txtLocal.getText();  
7443: }  
7444:  
7445: // montando o registro para gravação  
7446: PVregister reg = new PVregister();  
7447: reg.data = date;  
7448: reg.sentido = sentido;  
7449: reg.pista = pista;  
7450: reg.posto = Integer.parseInt(posto);  
7451: reg.local = local;  
7452: reg.sentido = sentido;  
7453: reg.pesquisador1 = pesquisador1;  
7454: reg.pesquisador2 = pesquisador2;  
7455: reg.hora = hora\_inicial;  
7456:  
7457: // reg fields for types  
7458: for (String s : volFieldsNames) {  
7459: try {  
7460: PVregister.class.getField(s.toLowerCase()).set(reg, buildValues(s));  
7461: } catch (Exception e) {  
7462: logger.error("Erro ao gravar os dados.", e);  
7463: JOptionPane.showMessageDialog(Janela.this, "Erro gravando dados:\n" + e.getMessage(),  
7464: "Erro na gravação dos dados.", JOptionPane.ERROR\_MESSAGE);  
7465: return;  
7466: }  
7467:  
7468: }  
7469:  
7470: try {  
7471: int alreadyInDataBase = database.verifyPV(reg);  
7472: if (alreadyInDataBase != 0) {  
7473: int returnedValue = JOptionPane.showConfirmDialog(Janela.this,  
7474: "Já existe um registro no Banco de Dados com o mesmo posto, sentido, data e hora.\nVocê deseja sobrescrever os dados já gravados?",  
7475: "Dados já existentes.", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE);  
7476: if (returnedValue == JOptionPane.YES\_OPTION) {  
7477: database.updatePV(reg, alreadyInDataBase);  
7478: clearForm();  
7479: }  
7480: return;// exit and don't save  
7481: }  
7482:  
7483: database.inputPV(reg);  
7484: clearForm();  
7485: JOptionPane.showMessageDialog(Janela.this, "Registro gravado com sucesso!", "Registro gravado.",  
7486: JOptionPane.INFORMATION\_MESSAGE);  
7487:  
7488: } catch (Exception e) {  
7489: logger.error("Erro ao conectar com o BD.", e);  
7490: JOptionPane.showMessageDialog(Janela.this, "Erro ao conectar com o banco de dados:\n" + e.getMessage(),  
7491: "Erro de conexão com o banco de dados.", JOptionPane.ERROR\_MESSAGE);  
7492: }  
7493:  
7494: return;  
7495: }// GEN-LAST:event\_btnSalvarFormsActionPerformed  
7496:  
7497: /\*\*  
7498: \* Método disparado quando o usuário escolhe um horário na \_caixa de seleção\_ de horas para o cadastro de   
7499: \* pesquisa volumétrica.  
7500: \* Verifica se os dados informados já são suficientes para identificar o registro. Atualiza o formulário para   
7501: \* exibir o horário selecionado.  
7502: \* @param evt evento de escolha de data na \_caixa de seleção\_.  
7503: \*/  
7504: private void cmbHoraActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_cmbHoraActionPerformed  
7505: int hora\_inicial;  
7506: if (cmbHora.getSelectedIndex() != -1) {  
7507: try {  
7508: hora\_inicial = Integer.parseInt((String) cmbHora.getSelectedItem());  
7509: } catch (Exception e) {  
7510: logger.error("Erro ao configurar a hora inicial da pesquisa.", e);  
7511: JOptionPane.showMessageDialog(null, "Erro configurando a hora inicial da pesquisa: \n" + e.getMessage());  
7512: return;  
7513: }  
7514:  
7515: String hora1 = String.valueOf(hora\_inicial) + ":00 a " + String.valueOf(hora\_inicial + 1) + ":00";  
7516: String hora2 = String.valueOf(hora\_inicial + 1) + ":00 a " + String.valueOf(hora\_inicial + 2) + ":00";  
7517:  
7518: pesados\_hora1.setText(hora1);  
7519: leves\_hora1.setText(hora1);  
7520: lblFolha1.setText(Integer.toString(cmbHora.getSelectedIndex() + 1));  
7521: pesados\_hora2.setText(hora2);  
7522: leves\_hora2.setText(hora2);  
7523: lblFolha2.setText(Integer.toString(cmbHora.getSelectedIndex() + 1));  
7524: askForDataRetrieve();  
7525: }  
7526: }// GEN-LAST:event\_cmbHoraActionPerformed  
7527:  
7528: /\*\*  
7529: \* Limpa os campos e restaura o estado inicial do formulário de entrada de dados da pesquisa volumétrica.  
7530: \*/  
7531: private void clearForm() {  
7532: btnApagar.setVisible(false);  
7533: cmbData.setSelectedIndex(0);  
7534: txtPesquisador1.setText(null);  
7535: txtPesquisador2.setText(null);  
7536: txtLocal.setText(null);  
7537: pesados\_hora1.setText(null);  
7538: pesados\_hora2.setText(null);  
7539: leves\_hora1.setText(null);  
7540: leves\_hora2.setText(null);  
7541: grpPista.clearSelection();  
7542: grpSentido.clearSelection();  
7543: cmbHora.setSelectedIndex(-1);  
7544: // txtTotalLeves.setText("0");  
7545: // txtTotalPesados.setText("0");  
7546: lblFolha1.setText("-");  
7547: lblFolha2.setText("-");  
7548: for (String s : volFieldsNames) {  
7549: for (JTextField f : fieldsMap.get(s)) {  
7550: f.setText("0");  
7551: }  
7552: }  
7553: }  
7554:  
7555: /\*\*  
7556: \* Monta a representação de dados no formato JSON a ser utilizada no armazenamento e transmissão de um registro de   
7557: \* pesquisa volumétrica.  
7558: \* @param field  
7559: \* @return   
7560: \*/  
7561: private String buildValues(String field) {  
7562: String r = "{\"Veiculo\": \"" + field + "\", \"Valores\": \"";  
7563: JTextField[] inputFields = fieldsMap.get(field);  
7564: int h1 = 0;  
7565:  
7566: int h2 = 0;  
7567:  
7568: for (int i = 0; i < inputFields.length; i++) {  
7569: r += inputFields[i].getText();  
7570: if (i + 1 != inputFields.length) {  
7571: r += ",";  
7572: }  
7573: if (i <= 3) {  
7574: h1 += Integer.parseInt(inputFields[i].getText());  
7575: } else {  
7576: h2 += Integer.parseInt(inputFields[i].getText());  
7577: }  
7578: }  
7579:  
7580: r += "\", \"Hora1\": " + String.valueOf(h1) + ", \"Hora2\": " + String.valueOf(h2) + "}";  
7581:  
7582: return r;  
7583: }  
7584:  
7585: /\*\*  
7586: \* Método utilizado apenas durante o desenvolvimento para testes.  
7587: \*   
7588: \* @param args método executado sem parâmetros  
7589: \*/  
7590: @Deprecated  
7591: public static void main(String args[]) {  
7592: /\* Set the Nimbus look and feel \*/  
7593: // <editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">  
7594: /\*  
7595: \* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel. For details see  
7596: \* http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html  
7597: \*/  
7598: try {  
7599: for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {  
7600: if ("Nimbus".equals(info.getName())) {  
7601: javax.swing.UIManager.setLookAndFeel(info.getClassName());  
7602: break;  
7603: }  
7604: }  
7605: } catch (ClassNotFoundException ex) {  
7606: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
7607: } catch (InstantiationException ex) {  
7608: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
7609: } catch (IllegalAccessException ex) {  
7610: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
7611: } catch (javax.swing.UnsupportedLookAndFeelException ex) {  
7612: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
7613: }  
7614: // </editor-fold>  
7615:  
7616: /\* Create and display the form \*/  
7617: java.awt.EventQueue.invokeLater(new Runnable() {  
7618: public void run() {  
7619: new Janela().setVisible(true);  
7620: }  
7621: });  
7622: }  
7623:  
7624: private DefaultTableCellRenderer numeros; ///< Formatador da célula para exibição dos números.  
7625: private ImagemRenderer imagens; ///< Formatador para exibição de imagens.  
7626:   
7627:   
7628: private InputVerifier intVerifier; ///< Verificador do formato de entrada dos valores inteiros.  
7629:  
7630: // Variables declaration - do not modify//GEN-BEGIN:variables  
7631: private javax.swing.JPanel alertaTreinamento;  
7632: private javax.swing.JButton btnApagar;  
7633: private javax.swing.JButton btnExpODDate;  
7634: private javax.swing.JButton btnExpVolDate;  
7635: private javax.swing.JButton btnInDados;  
7636: private javax.swing.JButton btnSalvarForms;  
7637: private javax.swing.JCheckBox chk\_exportadas\_sumVol;  
7638: private javax.swing.JCheckBox chk\_nao\_exportadas\_sumVol;  
7639: private javax.swing.JComboBox cmbData;  
7640: private javax.swing.JComboBox cmbDataSumVol;  
7641: private javax.swing.JComboBox<String> cmbDateExpOD;  
7642: private javax.swing.JComboBox<String> cmbDateExpVol;  
7643: private javax.swing.JComboBox<String> cmbHora;  
7644: private javax.swing.JFileChooser dadosFileChooser;  
7645: private javax.swing.JFileChooser exporterFileChooser;  
7646: private javax.swing.ButtonGroup grpPista;  
7647: private javax.swing.ButtonGroup grpSentido;  
7648: private javax.swing.JLabel jLabel1;  
7649: private javax.swing.JLabel jLabel10;  
7650: private javax.swing.JLabel jLabel11;  
7651: private javax.swing.JLabel jLabel12;  
7652: private javax.swing.JLabel jLabel13;  
7653: private javax.swing.JLabel jLabel14;  
7654: private javax.swing.JLabel jLabel15;  
7655: private javax.swing.JLabel jLabel16;  
7656: private javax.swing.JLabel jLabel17;  
7657: private javax.swing.JLabel jLabel18;  
7658: private javax.swing.JLabel jLabel19;  
7659: private javax.swing.JLabel jLabel2;  
7660: private javax.swing.JLabel jLabel20;  
7661: private javax.swing.JLabel jLabel21;  
7662: private javax.swing.JLabel jLabel22;  
7663: private javax.swing.JLabel jLabel23;  
7664: private javax.swing.JLabel jLabel24;  
7665: private javax.swing.JLabel jLabel25;  
7666: private javax.swing.JLabel jLabel26;  
7667: private javax.swing.JLabel jLabel27;  
7668: private javax.swing.JLabel jLabel28;  
7669: private javax.swing.JLabel jLabel29;  
7670: private javax.swing.JLabel jLabel3;  
7671: private javax.swing.JLabel jLabel30;  
7672: private javax.swing.JLabel jLabel31;  
7673: private javax.swing.JLabel jLabel32;  
7674: private javax.swing.JLabel jLabel33;  
7675: private javax.swing.JLabel jLabel34;  
7676: private javax.swing.JLabel jLabel35;  
7677: private javax.swing.JLabel jLabel36;  
7678: private javax.swing.JLabel jLabel37;  
7679: private javax.swing.JLabel jLabel38;  
7680: private javax.swing.JLabel jLabel39;  
7681: private javax.swing.JLabel jLabel4;  
7682: private javax.swing.JLabel jLabel40;  
7683: private javax.swing.JLabel jLabel41;  
7684: private javax.swing.JLabel jLabel42;  
7685: private javax.swing.JLabel jLabel43;  
7686: private javax.swing.JLabel jLabel44;  
7687: private javax.swing.JLabel jLabel45;  
7688: private javax.swing.JLabel jLabel46;  
7689: private javax.swing.JLabel jLabel47;  
7690: private javax.swing.JLabel jLabel48;  
7691: private javax.swing.JLabel jLabel49;  
7692: private javax.swing.JLabel jLabel5;  
7693: private javax.swing.JLabel jLabel50;  
7694: private javax.swing.JLabel jLabel51;  
7695: private javax.swing.JLabel jLabel52;  
7696: private javax.swing.JLabel jLabel53;  
7697: private javax.swing.JLabel jLabel54;  
7698: private javax.swing.JLabel jLabel55;  
7699: private javax.swing.JLabel jLabel56;  
7700: private javax.swing.JLabel jLabel57;  
7701: private javax.swing.JLabel jLabel58;  
7702: private javax.swing.JLabel jLabel59;  
7703: private javax.swing.JLabel jLabel6;  
7704: private javax.swing.JLabel jLabel60;  
7705: private javax.swing.JLabel jLabel61;  
7706: private javax.swing.JLabel jLabel62;  
7707: private javax.swing.JLabel jLabel63;  
7708: private javax.swing.JLabel jLabel64;  
7709: private javax.swing.JLabel jLabel65;  
7710: private javax.swing.JLabel jLabel66;  
7711: private javax.swing.JLabel jLabel67;  
7712: private javax.swing.JLabel jLabel68;  
7713: private javax.swing.JLabel jLabel69;  
7714: private javax.swing.JLabel jLabel7;  
7715: private javax.swing.JLabel jLabel70;  
7716: private javax.swing.JLabel jLabel71;  
7717: private javax.swing.JLabel jLabel72;  
7718: private javax.swing.JLabel jLabel8;  
7719: private javax.swing.JLabel jLabel9;  
7720: private javax.swing.JPanel jPanel1;  
7721: private javax.swing.JPanel jPanel10;  
7722: private javax.swing.JPanel jPanel100;  
7723: private javax.swing.JPanel jPanel101;  
7724: private javax.swing.JPanel jPanel102;  
7725: private javax.swing.JPanel jPanel103;  
7726: private javax.swing.JPanel jPanel104;  
7727: private javax.swing.JPanel jPanel105;  
7728: private javax.swing.JPanel jPanel106;  
7729: private javax.swing.JPanel jPanel107;  
7730: private javax.swing.JPanel jPanel108;  
7731: private javax.swing.JPanel jPanel109;  
7732: private javax.swing.JPanel jPanel11;  
7733: private javax.swing.JPanel jPanel110;  
7734: private javax.swing.JPanel jPanel111;  
7735: private javax.swing.JPanel jPanel112;  
7736: private javax.swing.JPanel jPanel113;  
7737: private javax.swing.JPanel jPanel114;  
7738: private javax.swing.JPanel jPanel115;  
7739: private javax.swing.JPanel jPanel116;  
7740: private javax.swing.JPanel jPanel117;  
7741: private javax.swing.JPanel jPanel118;  
7742: private javax.swing.JPanel jPanel119;  
7743: private javax.swing.JPanel jPanel12;  
7744: private javax.swing.JPanel jPanel13;  
7745: private javax.swing.JPanel jPanel14;  
7746: private javax.swing.JPanel jPanel15;  
7747: private javax.swing.JPanel jPanel16;  
7748: private javax.swing.JPanel jPanel17;  
7749: private javax.swing.JPanel jPanel18;  
7750: private javax.swing.JPanel jPanel19;  
7751: private javax.swing.JPanel jPanel2;  
7752: private javax.swing.JPanel jPanel20;  
7753: private javax.swing.JPanel jPanel21;  
7754: private javax.swing.JPanel jPanel22;  
7755: private javax.swing.JPanel jPanel23;  
7756: private javax.swing.JPanel jPanel24;  
7757: private javax.swing.JPanel jPanel25;  
7758: private javax.swing.JPanel jPanel26;  
7759: private javax.swing.JPanel jPanel27;  
7760: private javax.swing.JPanel jPanel28;  
7761: private javax.swing.JPanel jPanel29;  
7762: private javax.swing.JPanel jPanel3;  
7763: private javax.swing.JPanel jPanel30;  
7764: private javax.swing.JPanel jPanel31;  
7765: private javax.swing.JPanel jPanel32;  
7766: private javax.swing.JPanel jPanel33;  
7767: private javax.swing.JPanel jPanel34;  
7768: private javax.swing.JPanel jPanel35;  
7769: private javax.swing.JPanel jPanel36;  
7770: private javax.swing.JPanel jPanel37;  
7771: private javax.swing.JPanel jPanel38;  
7772: private javax.swing.JPanel jPanel39;  
7773: private javax.swing.JPanel jPanel4;  
7774: private javax.swing.JPanel jPanel40;  
7775: private javax.swing.JPanel jPanel41;  
7776: private javax.swing.JPanel jPanel42;  
7777: private javax.swing.JPanel jPanel43;  
7778: private javax.swing.JPanel jPanel44;  
7779: private javax.swing.JPanel jPanel45;  
7780: private javax.swing.JPanel jPanel46;  
7781: private javax.swing.JPanel jPanel47;  
7782: private javax.swing.JPanel jPanel48;  
7783: private javax.swing.JPanel jPanel49;  
7784: private javax.swing.JPanel jPanel5;  
7785: private javax.swing.JPanel jPanel50;  
7786: private javax.swing.JPanel jPanel51;  
7787: private javax.swing.JPanel jPanel52;  
7788: private javax.swing.JPanel jPanel53;  
7789: private javax.swing.JPanel jPanel54;  
7790: private javax.swing.JPanel jPanel55;  
7791: private javax.swing.JPanel jPanel56;  
7792: private javax.swing.JPanel jPanel57;  
7793: private javax.swing.JPanel jPanel58;  
7794: private javax.swing.JPanel jPanel59;  
7795: private javax.swing.JPanel jPanel6;  
7796: private javax.swing.JPanel jPanel60;  
7797: private javax.swing.JPanel jPanel61;  
7798: private javax.swing.JPanel jPanel62;  
7799: private javax.swing.JPanel jPanel63;  
7800: private javax.swing.JPanel jPanel64;  
7801: private javax.swing.JPanel jPanel65;  
7802: private javax.swing.JPanel jPanel66;  
7803: private javax.swing.JPanel jPanel67;  
7804: private javax.swing.JPanel jPanel68;  
7805: private javax.swing.JPanel jPanel69;  
7806: private javax.swing.JPanel jPanel7;  
7807: private javax.swing.JPanel jPanel70;  
7808: private javax.swing.JPanel jPanel71;  
7809: private javax.swing.JPanel jPanel72;  
7810: private javax.swing.JPanel jPanel73;  
7811: private javax.swing.JPanel jPanel74;  
7812: private javax.swing.JPanel jPanel75;  
7813: private javax.swing.JPanel jPanel76;  
7814: private javax.swing.JPanel jPanel77;  
7815: private javax.swing.JPanel jPanel78;  
7816: private javax.swing.JPanel jPanel79;  
7817: private javax.swing.JPanel jPanel8;  
7818: private javax.swing.JPanel jPanel80;  
7819: private javax.swing.JPanel jPanel81;  
7820: private javax.swing.JPanel jPanel82;  
7821: private javax.swing.JPanel jPanel83;  
7822: private javax.swing.JPanel jPanel84;  
7823: private javax.swing.JPanel jPanel85;  
7824: private javax.swing.JPanel jPanel86;  
7825: private javax.swing.JPanel jPanel87;  
7826: private javax.swing.JPanel jPanel88;  
7827: private javax.swing.JPanel jPanel89;  
7828: private javax.swing.JPanel jPanel9;  
7829: private javax.swing.JPanel jPanel90;  
7830: private javax.swing.JPanel jPanel91;  
7831: private javax.swing.JPanel jPanel92;  
7832: private javax.swing.JPanel jPanel93;  
7833: private javax.swing.JPanel jPanel94;  
7834: private javax.swing.JPanel jPanel95;  
7835: private javax.swing.JPanel jPanel96;  
7836: private javax.swing.JPanel jPanel97;  
7837: private javax.swing.JPanel jPanel98;  
7838: private javax.swing.JPanel jPanel99;  
7839: private javax.swing.JScrollPane jScrollPane1;  
7840: private javax.swing.JScrollPane jScrollPane2;  
7841: private javax.swing.JScrollPane jScrollPane3;  
7842: private javax.swing.JScrollPane jScrollPane4;  
7843: private javax.swing.JScrollPane jScrollPane5;  
7844: private javax.swing.JTabbedPane jTabbedPane1;  
7845: private javax.swing.JLabel lbl2C;  
7846: private javax.swing.JLabel lbl2C2;  
7847: private javax.swing.JLabel lbl2C3;  
7848: private javax.swing.JLabel lbl2CB;  
7849: private javax.swing.JLabel lbl2S1;  
7850: private javax.swing.JLabel lbl2S2;  
7851: private javax.swing.JLabel lbl2S3;  
7852: private javax.swing.JLabel lbl3C;  
7853: private javax.swing.JLabel lbl3C2;  
7854: private javax.swing.JLabel lbl3C3;  
7855: private javax.swing.JLabel lbl3CB;  
7856: private javax.swing.JLabel lbl3D;  
7857: private javax.swing.JLabel lbl3D4;  
7858: private javax.swing.JLabel lbl3M6;  
7859: private javax.swing.JLabel lbl3Q4;  
7860: private javax.swing.JLabel lbl3S1;  
7861: private javax.swing.JLabel lbl3S2;  
7862: private javax.swing.JLabel lbl3S3;  
7863: private javax.swing.JLabel lbl3T4;  
7864: private javax.swing.JLabel lbl3T6;  
7865: private javax.swing.JLabel lbl3T6B;  
7866: private javax.swing.JLabel lbl3V5;  
7867: private javax.swing.JLabel lbl4C;  
7868: private javax.swing.JLabel lbl4CB;  
7869: private javax.swing.JLabel lbl4CD;  
7870: private javax.swing.JLabel lblData;  
7871: private javax.swing.JLabel lblFolha1;  
7872: private javax.swing.JLabel lblFolha2;  
7873: private javax.swing.JLabel lblHora;  
7874: private javax.swing.JLabel lblLocal;  
7875: private javax.swing.JLabel lblM;  
7876: private javax.swing.JLabel lblP1;  
7877: private javax.swing.JLabel lblP2;  
7878: private javax.swing.JLabel lblP3;  
7879: private javax.swing.JLabel lblPesquisador;  
7880: private javax.swing.JLabel lblPesquisador1;  
7881: private javax.swing.JLabel lblPista;  
7882: private javax.swing.JLabel lblPosto;  
7883: public javax.swing.JLabel lblPosto\_dados;  
7884: private javax.swing.JLabel lblSentido;  
7885: public javax.swing.JLabel lblTrecho\_dados;  
7886: private javax.swing.JTextField lev15\_1;  
7887: private javax.swing.JTextField lev15\_2;  
7888: private javax.swing.JTextField lev30\_1;  
7889: private javax.swing.JTextField lev30\_2;  
7890: private javax.swing.JTextField lev45\_1;  
7891: private javax.swing.JTextField lev45\_2;  
7892: private javax.swing.JTextField lev60\_1;  
7893: private javax.swing.JTextField lev60\_2;  
7894: private javax.swing.JTextField leves\_hora1;  
7895: private javax.swing.JTextField leves\_hora2;  
7896: private javax.swing.JLabel odStatus;  
7897: private javax.swing.JPanel panelLeves;  
7898: private javax.swing.JPanel panelPesados;  
7899: private javax.swing.JTextField pes15\_1;  
7900: private javax.swing.JTextField pes15\_2;  
7901: private javax.swing.JTextField pes30\_1;  
7902: private javax.swing.JTextField pes30\_2;  
7903: private javax.swing.JTextField pes45\_1;  
7904: private javax.swing.JTextField pes60\_1;  
7905: private javax.swing.JTextField pesados\_hora1;  
7906: private javax.swing.JTextField pesados\_hora2;  
7907: private javax.swing.JTextField peso45\_2;  
7908: private javax.swing.JTextField peso60\_2;  
7909: private javax.swing.JPanel pnl\_envio;  
7910: private javax.swing.JPanel pnl\_relatorio;  
7911: private javax.swing.JPanel pnl\_servidor;  
7912: private javax.swing.JPanel pnl\_sumario\_volumetrica;  
7913: private javax.swing.JPanel pnl\_volumetrica;  
7914: private javax.swing.JRadioButton rdo\_PistaDupla;  
7915: private javax.swing.JRadioButton rdo\_PistaSimples;  
7916: private javax.swing.JRadioButton rdo\_SentidoAB;  
7917: private javax.swing.JRadioButton rdo\_SentidoBA;  
7918: private javax.swing.JTable relatorio;  
7919: private javax.swing.JTabbedPane tabRelatorio;  
7920: private javax.swing.JPanel tab\_leves;  
7921: private javax.swing.JPanel tab\_pesados;  
7922: private javax.swing.JTable tblSumVol;  
7923: private javax.swing.JTextField tl0;  
7924: private javax.swing.JTextField tl1;  
7925: private javax.swing.JTextField tl10;  
7926: private javax.swing.JTextField tl11;  
7927: private javax.swing.JTextField tl12;  
7928: private javax.swing.JTextField tl13;  
7929: private javax.swing.JTextField tl14;  
7930: private javax.swing.JTextField tl15;  
7931: private javax.swing.JTextField tl16;  
7932: private javax.swing.JTextField tl17;  
7933: private javax.swing.JTextField tl18;  
7934: private javax.swing.JTextField tl19;  
7935: private javax.swing.JTextField tl2;  
7936: private javax.swing.JTextField tl20;  
7937: private javax.swing.JTextField tl21;  
7938: private javax.swing.JTextField tl22;  
7939: private javax.swing.JTextField tl23;  
7940: private javax.swing.JTextField tl24;  
7941: private javax.swing.JTextField tl25;  
7942: private javax.swing.JTextField tl26;  
7943: private javax.swing.JTextField tl27;  
7944: private javax.swing.JTextField tl28;  
7945: private javax.swing.JTextField tl29;  
7946: private javax.swing.JTextField tl3;  
7947: private javax.swing.JTextField tl30;  
7948: private javax.swing.JTextField tl31;  
7949: private javax.swing.JTextField tl32;  
7950: private javax.swing.JTextField tl33;  
7951: private javax.swing.JTextField tl34;  
7952: private javax.swing.JTextField tl35;  
7953: private javax.swing.JTextField tl36;  
7954: private javax.swing.JTextField tl37;  
7955: private javax.swing.JTextField tl38;  
7956: private javax.swing.JTextField tl39;  
7957: private javax.swing.JTextField tl4;  
7958: private javax.swing.JTextField tl40;  
7959: private javax.swing.JTextField tl41;  
7960: private javax.swing.JTextField tl42;  
7961: private javax.swing.JTextField tl43;  
7962: private javax.swing.JTextField tl44;  
7963: private javax.swing.JTextField tl45;  
7964: private javax.swing.JTextField tl46;  
7965: private javax.swing.JTextField tl47;  
7966: private javax.swing.JTextField tl48;  
7967: private javax.swing.JTextField tl49;  
7968: private javax.swing.JTextField tl5;  
7969: private javax.swing.JTextField tl50;  
7970: private javax.swing.JTextField tl51;  
7971: private javax.swing.JTextField tl52;  
7972: private javax.swing.JTextField tl53;  
7973: private javax.swing.JTextField tl54;  
7974: private javax.swing.JTextField tl55;  
7975: private javax.swing.JTextField tl56;  
7976: private javax.swing.JTextField tl57;  
7977: private javax.swing.JTextField tl58;  
7978: private javax.swing.JTextField tl59;  
7979: private javax.swing.JTextField tl6;  
7980: private javax.swing.JTextField tl60;  
7981: private javax.swing.JTextField tl61;  
7982: private javax.swing.JTextField tl62;  
7983: private javax.swing.JTextField tl63;  
7984: private javax.swing.JTextField tl64;  
7985: private javax.swing.JTextField tl65;  
7986: private javax.swing.JTextField tl66;  
7987: private javax.swing.JTextField tl67;  
7988: private javax.swing.JTextField tl68;  
7989: private javax.swing.JTextField tl69;  
7990: private javax.swing.JTextField tl7;  
7991: private javax.swing.JTextField tl70;  
7992: private javax.swing.JTextField tl71;  
7993: private javax.swing.JTextField tl72;  
7994: private javax.swing.JTextField tl73;  
7995: private javax.swing.JTextField tl74;  
7996: private javax.swing.JTextField tl75;  
7997: private javax.swing.JTextField tl76;  
7998: private javax.swing.JTextField tl77;  
7999: private javax.swing.JTextField tl78;  
8000: private javax.swing.JTextField tl79;  
8001: private javax.swing.JTextField tl8;  
8002: private javax.swing.JTextField tl80;  
8003: private javax.swing.JTextField tl81;  
8004: private javax.swing.JTextField tl82;  
8005: private javax.swing.JTextField tl83;  
8006: private javax.swing.JTextField tl84;  
8007: private javax.swing.JTextField tl85;  
8008: private javax.swing.JTextField tl86;  
8009: private javax.swing.JTextField tl87;  
8010: private javax.swing.JTextField tl88;  
8011: private javax.swing.JTextField tl89;  
8012: private javax.swing.JTextField tl9;  
8013: private javax.swing.JTextField tl90;  
8014: private javax.swing.JTextField tl91;  
8015: private javax.swing.JTextField tl92;  
8016: private javax.swing.JTextField tl93;  
8017: private javax.swing.JTextField tl94;  
8018: private javax.swing.JTextField tl95;  
8019: private javax.swing.JTextField tp0;  
8020: private javax.swing.JTextField tp1;  
8021: private javax.swing.JTextField tp10;  
8022: private javax.swing.JTextField tp100;  
8023: private javax.swing.JTextField tp101;  
8024: private javax.swing.JTextField tp102;  
8025: private javax.swing.JTextField tp103;  
8026: private javax.swing.JTextField tp104;  
8027: private javax.swing.JTextField tp105;  
8028: private javax.swing.JTextField tp106;  
8029: private javax.swing.JTextField tp107;  
8030: private javax.swing.JTextField tp108;  
8031: private javax.swing.JTextField tp109;  
8032: private javax.swing.JTextField tp11;  
8033: private javax.swing.JTextField tp110;  
8034: private javax.swing.JTextField tp111;  
8035: private javax.swing.JTextField tp112;  
8036: private javax.swing.JTextField tp113;  
8037: private javax.swing.JTextField tp114;  
8038: private javax.swing.JTextField tp115;  
8039: private javax.swing.JTextField tp116;  
8040: private javax.swing.JTextField tp117;  
8041: private javax.swing.JTextField tp118;  
8042: private javax.swing.JTextField tp119;  
8043: private javax.swing.JTextField tp12;  
8044: private javax.swing.JTextField tp120;  
8045: private javax.swing.JTextField tp121;  
8046: private javax.swing.JTextField tp122;  
8047: private javax.swing.JTextField tp123;  
8048: private javax.swing.JTextField tp124;  
8049: private javax.swing.JTextField tp125;  
8050: private javax.swing.JTextField tp126;  
8051: private javax.swing.JTextField tp127;  
8052: private javax.swing.JTextField tp128;  
8053: private javax.swing.JTextField tp129;  
8054: private javax.swing.JTextField tp13;  
8055: private javax.swing.JTextField tp130;  
8056: private javax.swing.JTextField tp131;  
8057: private javax.swing.JTextField tp132;  
8058: private javax.swing.JTextField tp133;  
8059: private javax.swing.JTextField tp134;  
8060: private javax.swing.JTextField tp135;  
8061: private javax.swing.JTextField tp14;  
8062: private javax.swing.JTextField tp15;  
8063: private javax.swing.JTextField tp16;  
8064: private javax.swing.JTextField tp17;  
8065: private javax.swing.JTextField tp18;  
8066: private javax.swing.JTextField tp19;  
8067: private javax.swing.JTextField tp2;  
8068: private javax.swing.JTextField tp20;  
8069: private javax.swing.JTextField tp21;  
8070: private javax.swing.JTextField tp22;  
8071: private javax.swing.JTextField tp23;  
8072: private javax.swing.JTextField tp24;  
8073: private javax.swing.JTextField tp25;  
8074: private javax.swing.JTextField tp26;  
8075: private javax.swing.JTextField tp27;  
8076: private javax.swing.JTextField tp28;  
8077: private javax.swing.JTextField tp29;  
8078: private javax.swing.JTextField tp3;  
8079: private javax.swing.JTextField tp30;  
8080: private javax.swing.JTextField tp31;  
8081: private javax.swing.JTextField tp32;  
8082: private javax.swing.JTextField tp33;  
8083: private javax.swing.JTextField tp34;  
8084: private javax.swing.JTextField tp35;  
8085: private javax.swing.JTextField tp36;  
8086: private javax.swing.JTextField tp37;  
8087: private javax.swing.JTextField tp38;  
8088: private javax.swing.JTextField tp39;  
8089: private javax.swing.JTextField tp4;  
8090: private javax.swing.JTextField tp40;  
8091: private javax.swing.JTextField tp41;  
8092: private javax.swing.JTextField tp42;  
8093: private javax.swing.JTextField tp43;  
8094: private javax.swing.JTextField tp44;  
8095: private javax.swing.JTextField tp45;  
8096: private javax.swing.JTextField tp46;  
8097: private javax.swing.JTextField tp47;  
8098: private javax.swing.JTextField tp48;  
8099: private javax.swing.JTextField tp49;  
8100: private javax.swing.JTextField tp5;  
8101: private javax.swing.JTextField tp50;  
8102: private javax.swing.JTextField tp51;  
8103: private javax.swing.JTextField tp52;  
8104: private javax.swing.JTextField tp53;  
8105: private javax.swing.JTextField tp54;  
8106: private javax.swing.JTextField tp55;  
8107: private javax.swing.JTextField tp56;  
8108: private javax.swing.JTextField tp57;  
8109: private javax.swing.JTextField tp58;  
8110: private javax.swing.JTextField tp59;  
8111: private javax.swing.JTextField tp6;  
8112: private javax.swing.JTextField tp60;  
8113: private javax.swing.JTextField tp61;  
8114: private javax.swing.JTextField tp62;  
8115: private javax.swing.JTextField tp63;  
8116: private javax.swing.JTextField tp64;  
8117: private javax.swing.JTextField tp65;  
8118: private javax.swing.JTextField tp66;  
8119: private javax.swing.JTextField tp67;  
8120: private javax.swing.JTextField tp68;  
8121: private javax.swing.JTextField tp69;  
8122: private javax.swing.JTextField tp7;  
8123: private javax.swing.JTextField tp70;  
8124: private javax.swing.JTextField tp71;  
8125: private javax.swing.JTextField tp72;  
8126: private javax.swing.JTextField tp73;  
8127: private javax.swing.JTextField tp74;  
8128: private javax.swing.JTextField tp75;  
8129: private javax.swing.JTextField tp76;  
8130: private javax.swing.JTextField tp77;  
8131: private javax.swing.JTextField tp78;  
8132: private javax.swing.JTextField tp79;  
8133: private javax.swing.JTextField tp8;  
8134: private javax.swing.JTextField tp80;  
8135: private javax.swing.JTextField tp81;  
8136: private javax.swing.JTextField tp82;  
8137: private javax.swing.JTextField tp83;  
8138: private javax.swing.JTextField tp84;  
8139: private javax.swing.JTextField tp85;  
8140: private javax.swing.JTextField tp86;  
8141: private javax.swing.JTextField tp87;  
8142: private javax.swing.JTextField tp88;  
8143: private javax.swing.JTextField tp89;  
8144: private javax.swing.JTextField tp9;  
8145: private javax.swing.JTextField tp90;  
8146: private javax.swing.JTextField tp91;  
8147: private javax.swing.JTextField tp92;  
8148: private javax.swing.JTextField tp93;  
8149: private javax.swing.JTextField tp94;  
8150: private javax.swing.JTextField tp95;  
8151: private javax.swing.JTextField tp96;  
8152: private javax.swing.JTextField tp97;  
8153: private javax.swing.JTextField tp98;  
8154: private javax.swing.JTextField tp99;  
8155: private javax.swing.JTextField txtAlertTreinamento;  
8156: public javax.swing.JTextArea txtAreaLog;  
8157: private javax.swing.JTextField txtLocal;  
8158: private javax.swing.JTextField txtPesquisador1;  
8159: private javax.swing.JTextField txtPesquisador2;  
8160: public javax.swing.JTextField txtPorta;  
8161: private javax.swing.JTextField txtSumVolC1;  
8162: private javax.swing.JTextField txtSumVolC2;  
8163: private javax.swing.JTextField txtSumVolC3;  
8164: private javax.swing.JTextField txtSumVolC4;  
8165: private javax.swing.JTextField txtSumVolC5;  
8166: private javax.swing.JTextField txtSumVolM;  
8167: private javax.swing.JTextField txtSumVolO1;  
8168: private javax.swing.JTextField txtSumVolO2;  
8169: private javax.swing.JTextField txtSumVolO3;  
8170: private javax.swing.JTextField txtSumVolP1;  
8171: private javax.swing.JTextField txtSumVolP2;  
8172: private javax.swing.JTextField txtSumVolP3;  
8173: private javax.swing.JTextField txtSumVolR1;  
8174: private javax.swing.JTextField txtSumVolR2;  
8175: private javax.swing.JTextField txtSumVolR3;  
8176: private javax.swing.JTextField txtSumVolR4;  
8177: private javax.swing.JTextField txtSumVolR5;  
8178: private javax.swing.JTextField txtSumVolR6;  
8179: private javax.swing.JTextField txtSumVolS1;  
8180: private javax.swing.JTextField txtSumVolS2;  
8181: private javax.swing.JTextField txtSumVolS3;  
8182: private javax.swing.JTextField txtSumVolS4;  
8183: private javax.swing.JTextField txtSumVolS5;  
8184: private javax.swing.JTextField txtSumVolS6;  
8185: private javax.swing.JTextField txtSumVolSE1;  
8186: private javax.swing.JTextField txtSumVolSE2;  
8187: private javax.swing.JTextField txtSumVolSE3;  
8188: private javax.swing.JTextField txtSumVolSE4;  
8189: private javax.swing.JTextField txtSumVolSE5;  
8190: private javax.swing.JTextField txtTotalLeves;  
8191: private javax.swing.JTextField txtTotalPesados;  
8192: // End of variables declaration//GEN-END:variables  
8193:  
8194: private String[] volFieldsNames; ///< Nomes dos campos do formulário da pesquisa volumétrica.  
8195:  
8196: private String[] volFieldsNamesLeves = { "P1", "P3", "P2", "M", "O1", "O2",   
8197: "O3", "C1", "C2", "C3", "C4", "C5" }; ///< Nomes dos campos para veículos leves.  
8198: private String[] volFieldsNamesPesados = { "S1", "S2", "S3", "S4", "S5", "S6",  
8199: "SE1", "SE2", "SE3", "SE4", "SE5",   
8200: "R1", "R2", "R3", "R4", "R5", "R6" }; ///< Nomes dos campos para veículos pesados.  
8201:  
8202: private HashMap<String, JTextField[]> fieldsMap; ///< Mapeamento entre os nomes e conjuntos de campos de entrada de dados.  
8203:  
8204: private enum fieldTypes {  
8205: LEVES, PESADOS  
8206: } ///< tipos de campos de entradas de dados de veículos [LEVES | PESADOS].  
8207:  
8208: private HashMap<String, JTextField> txtFree; ///< Mapeamento entre nome e campo de entrada de dados.  
8209:  
8210: private boolean ctlGetValuesFromDataBase = true; ///< Informa se os dados editados no formulário da pesquisa volumétrica são novos ou oriundos do banco de dados.  
8211:  
8212:}  
8213:  
8214:/\*\*  
8215: \* Classe que representa a formatação de uma célula para a exibição de uma imagem no formulário de entrada de   
8216: \* dados das pesquisas volumétricas.  
8217: \*   
8218: \* @author ludes - PESC - COPPE - ufrj  
8219: \* @author Eduardo Mangeli  
8220: \* @author Marcelo Areas  
8221: \* @author Fabrício Pereira  
8222: \* @author Geraldo Xexéo  
8223: \*/  
8224:@SuppressWarnings("serial")  
8225:class ImagemRenderer extends DefaultTableCellRenderer {  
8226:   
8227: /\*\*  
8228: \* Construtor que determina o espaçamento constuíndo uma borda transparente.  
8229: \*/  
8230: public ImagemRenderer() {  
8231: super();  
8232: setBorder(BorderFactory.createEmptyBorder(5, 0, 5, 0));  
8233: // setBackground(new Color(178,178,178));  
8234: }  
8235:  
8236: /\*\*  
8237: \* Método que exibe a imagem na célula.  
8238: \*   
8239: \* @param value imagem a ser exibida   
8240: \*/  
8241: public void setValue(Object value) {  
8242: if (value == null) {  
8243: setText("");  
8244: } else {  
8245: setIcon((ImageIcon) value);  
8246: }  
8247:  
8248: }  
8249:}  
8250:  
8251:/\*\*  
8252: \* Filtro para exibição de diretórios para a importação de arquivos.  
8253: \*   
8254: \* @author ludes - PESC - COPPE - ufrj  
8255: \* @author Eduardo Mangeli  
8256: \* @author Marcelo Areas  
8257: \* @author Fabrício Pereira  
8258: \* @author Geraldo Xexéo  
8259: \*/  
8260:class SQLiteFilter extends FileFilter {  
8261:  
8262: /\*\*  
8263: \* Verifica se um arquivo ou diretório atende ao critério de exibição.  
8264: \*   
8265: \* @param f arquivo/diretório a ser verificado.  
8266: \* @return resultado da verificação.  
8267: \*/  
8268: @Override  
8269: public boolean accept(File f) {  
8270: return f.isDirectory(); // || f.getAbsolutePath().endsWith(".db");  
8271: }  
8272:  
8273: /\*\*  
8274: \* Retorna a descrição do filtro.  
8275: \*   
8276: \* @return descrição.  
8277: \*/  
8278: @Override  
8279: public String getDescription() {  
8280: return "Dados pesquisa OD (\*.db)";  
8281: }  
8282:  
8283:}  
8284:  
8285:/\*\*  
8286: \* Filtro para exibição de diretórios e arquivos para exportação de arquivos NÃO compactados.  
8287: \*   
8288: \* @author ludes - PESC - COPPE - ufrj  
8289: \* @author Eduardo Mangeli  
8290: \* @author Marcelo Areas  
8291: \* @author Fabrício Pereira  
8292: \* @author Geraldo Xexéo  
8293: \*/  
8294:class JsonFilter extends FileFilter {  
8295:  
8296: /\*\*  
8297: \* Verifica se um arquivo ou diretório atende ao critério de exibição.  
8298: \*   
8299: \* @param f arquivo/diretório a ser verificado.  
8300: \* @return resultado da verificação.  
8301: \*/  
8302: @Override  
8303: public boolean accept(File f) {  
8304: return f.isDirectory() || f.getAbsolutePath().endsWith(".json");  
8305: }  
8306:  
8307: /\*\*  
8308: \* Retorna a descrição do filtro.  
8309: \*   
8310: \* @return descrição.  
8311: \*/  
8312: @Override  
8313: public String getDescription() {  
8314: return "Arquivos JSON (\*.json)";  
8315: }  
8316:  
8317:}  
8318:  
8319:/\*\*  
8320: \* Filtro para exibição de diretórios e arquivos para exportação de arquivos compactados.  
8321: \*   
8322: \* @author ludes - PESC - COPPE - ufrj  
8323: \* @author Eduardo Mangeli  
8324: \* @author Marcelo Areas  
8325: \* @author Fabrício Pereira  
8326: \* @author Geraldo Xexéo  
8327: \*/  
8328:class ZipFilter extends FileFilter {  
8329:   
8330: /\*\*  
8331: \* Verifica se um arquivo ou diretório atende ao critério de exibição.  
8332: \*   
8333: \* @param f arquivo/diretório a ser verificado.  
8334: \* @return resultado da verificação.  
8335: \*/  
8336: @Override  
8337: public boolean accept(File f) {  
8338: return f.isDirectory() || f.getAbsolutePath().endsWith(".zip");  
8339: }  
8340:  
8341: /\*\*  
8342: \* Retorna a descrição do filtro.  
8343: \*   
8344: \* @return descrição.  
8345: \*/  
8346: @Override  
8347: public String getDescription() {  
8348: return "Arquivos ZIP (\*.zip)";  
8349: }  
8350:  
8351:}  
8352:  
8353:/\*\*  
8354: \* Classe para formatação da célula de exibição do sumário das pesquisas do tipo Origem/Destino.  
8355: \*   
8356: \* @author ludes - PESC - COPPE - ufrj  
8357: \* @author Eduardo Mangeli  
8358: \* @author Marcelo Areas  
8359: \* @author Fabrício Pereira  
8360: \* @author Geraldo Xexéo  
8361: \*/  
8362:@SuppressWarnings("serial")  
8363:class RelatorioODRender extends DefaultTableCellRenderer {  
8364: public Component getTableCellRendererComponent(JTable table, Object value, boolean isSelected, boolean hasFocus, int row,  
8365: int column) {  
8366: Component c = super.getTableCellRendererComponent(table, value, isSelected, hasFocus, row, column);  
8367:  
8368: if (column == 0)  
8369: c.setBackground(Color.LIGHT\_GRAY);  
8370: else  
8371: c.setBackground(table.getBackground());  
8372:  
8373: if (column == 0 || column == table.getColumnCount() - 1 || row == table.getRowCount() - 1)  
8374: c.setFont(c.getFont().deriveFont(Font.BOLD));  
8375: else  
8376: c.setFont(c.getFont().deriveFont(Font.PLAIN));  
8377:  
8378: if (row != 0) {  
8379: super.setHorizontalAlignment(DefaultTableCellRenderer.CENTER);  
8380: }  
8381:  
8382: if (table.isRowSelected(row) || table.isColumnSelected(column)) {  
8383: if (column != 0 && (row < table.getRowCount() - 1 || column == table.getSelectedColumn()))  
8384: c.setBackground(new Color(240, 255, 128));  
8385: }  
8386:  
8387: return c;  
8388: }  
8389:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\LoginController.java

1:package br.ufrj.coppetec.concentrador;  
 2:  
 3:import java.io.InputStream;  
 4:import java.util.LinkedHashMap;  
 5:import java.util.Map;  
 6:import java.util.Scanner;  
 7:import java.util.TreeMap;  
 8:  
 9:import javax.swing.JOptionPane;  
 10:  
 11:import org.apache.logging.log4j.LogManager;  
 12:import org.apache.logging.log4j.Logger;  
 13:import org.json.JSONArray;  
 14:import org.json.JSONObject;  
 15:  
 16:/\*  
 17: \* To change this license header, choose License Headers in Project Properties.  
 18: \* To change this template file, choose Tools | Templates  
 19: \* and open the template in the editor.  
 20: \*/  
 21:  
 22:/\*\*  
 23: \*  
 24: \* @author mangeli  
 25: \*/  
 26:public class LoginController {  
 27:  
 28: private JSONObject loginsJson;  
 29: private Map<String, Map<String, String>> logins;  
 30: private static Logger logger = LogManager.getLogger(LoginController.class);  
 31: private static int LEN\_LOGIN\_USER = 3;  
 32:  
 33: public LoginController() {  
 34: this("/logins/users.json");  
 35: }  
 36:  
 37: public LoginController(String loginsFile) {  
 38: this.buildLoginsObject(loginsFile);  
 39: }  
 40:  
 41: private void buildLoginsObject(String loginsFile) {  
 42: try {  
 43: StringBuilder result = new StringBuilder("");  
 44: InputStream loginsFileStream = this.getClass().getResourceAsStream(loginsFile);  
 45: Scanner scanner = new Scanner(loginsFileStream);  
 46: while (scanner.hasNextLine()) {  
 47: String line = scanner.nextLine();  
 48: result.append(line).append("\n");  
 49: }  
 50: scanner.close();  
 51: loginsJson = new JSONObject(result.toString());  
 52: loadMapLogins(loginsJson.getJSONArray("logins"));  
 53: } catch (Exception e) {  
 54: logger.error("Erro ao ler o arquivo de login", e);  
 55: JOptionPane.showMessageDialog(null, "Erro lendo o arquivo de login." + e.getMessage());  
 56: }  
 57: }  
 58:  
 59: private void loadMapLogins(JSONArray arrayLoginsJson) {  
 60: logins = new TreeMap<String, Map<String, String>>();  
 61: for (Object objLoginJson : arrayLoginsJson) {  
 62: JSONObject loginJson = (JSONObject) objLoginJson;  
 63: Map<String, String> mapInfoByLogin = new LinkedHashMap<String, String>(loginJson.length() - 1);  
 64: String usr = loginJson.getString("usr");  
 65: for (String key : loginJson.keySet()) {  
 66: if (!key.equals("usr")) {  
 67: Object value = loginJson.get(key);  
 68: if (value instanceof String) {  
 69: mapInfoByLogin.put(key, (String) value);  
 70: } else if (value instanceof Boolean) {  
 71: mapInfoByLogin.put(key, Boolean.toString(loginJson.getBoolean(key)));  
 72: }  
 73: }  
 74: }  
 75: logins.put(usr, mapInfoByLogin);  
 76: }  
 77:  
 78: }  
 79:  
 80: public boolean validateLogin(String user, String pwd) {  
 81: boolean isValidLogin = false;  
 82:  
 83: if (verificaUserPwdProducao(user, pwd)) {  
 84: Concentrador.posto = user;  
 85: Concentrador.trecho = logins.get(user).get("trecho");  
 86: isValidLogin = true;  
 87: } else if (verificaUserPwdTreinamento(user, pwd)) {  
 88: user = user.substring(0, 3);  
 89: Concentrador.treinamento = true;  
 90: Concentrador.posto = user;  
 91: Concentrador.trecho = String.format("TREINAMENTO (%s)",  
 92: (!user.equals(Concentrador.configuration.getProperty("trainingUser")) ? logins.get(user).get("trecho")  
 93: : "Trecho de teste"));  
 94: isValidLogin = true;  
 95: }  
 96:  
 97: if (isValidLogin == true) {  
 98: logger.info("Login" + (Concentrador.treinamento ? " de TREINAMENTO " : " ") + "do usuário COM sucesso: " + user);  
 99: } else {  
 100: logger.info("Tentativa de login SEM sucesso para o usuário: " + user);  
 101: }  
 102: return isValidLogin;  
 103: }  
 104:  
 105: private boolean verificaUserPwdProducao(String user, String pwd) {  
 106: if (user.length() == LEN\_LOGIN\_USER) {  
 107: // try {  
 108: // /\* O IF abaixo está comentado por não termos confiança na data do sistema. \*/  
 109: // if (Util.isInValidPeriod(new Date())) {  
 110: return verificaUserPwd(user, pwd);  
 111: // }  
 112: // } catch (ParseException e) { /\* O TRY/CATCH acompanha a condição do IF. \*/  
 113: // JOptionPane.showMessageDialog(null, "Erro ao carregar datas válidas." + e.getMessage());  
 114: // }  
 115: }  
 116: return false;  
 117: }  
 118:  
 119: private boolean verificaUserPwdTreinamento(String user, String pwd) {  
 120: if ((user.equals(Concentrador.configuration.getProperty("trainingUser"))  
 121: || user.equalsIgnoreCase(Concentrador.configuration.getProperty("trainingUser") + "T"))  
 122: && pwd.equals(Concentrador.configuration.getProperty("trainingPassword"))) {  
 123: return true;  
 124: } else if ((user.length() == LEN\_LOGIN\_USER + 1) && (Character.toUpperCase(user.charAt(LEN\_LOGIN\_USER)) == 'T')) {  
 125: // try {  
 126: // /\* O IF abaixo está comentado por não termos confiança na data do sistema. \*/  
 127: // if (!Util.isInValidPeriod(new Date())) {  
 128: return verificaUserPwd(user.substring(0, LEN\_LOGIN\_USER), pwd);  
 129: // }  
 130: // } catch (ParseException e) { /\* O TRY/CATCH acompanha a condição do IF. \*/  
 131: // JOptionPane.showMessageDialog(null, "Erro ao carregar datas de treinamento." + e.getMessage());  
 132: // }  
 133: }  
 134: return false;  
 135: }  
 136:  
 137: private boolean verificaUserPwd(String user, String pwd) {  
 138: Map<String, String> loginInfos = logins.get(user);  
 139: if (loginInfos != null) {  
 140: return pwd.equals(loginInfos.get("pwd"));  
 141: }  
 142: return false;  
 143: }  
 144:  
 145:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\LoginJanela.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador;  
 7:  
 8:import javax.swing.JOptionPane;  
 9:  
 10:import org.apache.logging.log4j.LogManager;  
 11:import org.apache.logging.log4j.Logger;  
 12:  
 13:/\*\*  
 14: \*  
 15: \* @author mangeli  
 16: \*/  
 17:@SuppressWarnings("serial")  
 18:public class LoginJanela extends javax.swing.JDialog {  
 19:  
 20: private static Logger logger = LogManager.getLogger(LoginJanela.class);  
 21:  
 22: /\*\*  
 23: \* Creates new form LoginJanela  
 24: \*/  
 25: public LoginJanela(java.awt.Frame parent, boolean modal) {  
 26: super(parent, modal);  
 27: initComponents();  
 28: }  
 29:  
 30: private void exitProgram() {  
 31: logger.info("Usuário cancelou o login");  
 32: System.exit(0);  
 33: }  
 34:  
 35: private boolean validateLogin() {  
 36: LoginController controller = new LoginController("/logins/users.json");  
 37: return controller.validateLogin(txtUser.getText(), String.copyValueOf(txtPassword.getPassword()));  
 38: }  
 39:  
 40: /\*\*  
 41: \* This method is called from within the constructor to initialize the form. WARNING: Do NOT modify this code. The content of  
 42: \* this method is always regenerated by the Form Editor.  
 43: \*/  
 44: // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents  
 45: private void initComponents() {  
 46:  
 47: jLabel1 = new javax.swing.JLabel();  
 48: jLabel2 = new javax.swing.JLabel();  
 49: txtUser = new javax.swing.JTextField();  
 50: txtPassword = new javax.swing.JPasswordField();  
 51: btnOK = new javax.swing.JToggleButton();  
 52: btnCancel = new javax.swing.JButton();  
 53:  
 54: setDefaultCloseOperation(javax.swing.WindowConstants.DO\_NOTHING\_ON\_CLOSE);  
 55: setTitle("Login PNCT");  
 56: setAlwaysOnTop(true);  
 57: setIconImage(null);  
 58: setLocationByPlatform(true);  
 59: setModal(true);  
 60:  
 61: jLabel1.setText("Usuário:");  
 62:  
 63: jLabel2.setText("Senha:");  
 64:  
 65: btnOK.setText("OK");  
 66: btnOK.addActionListener(new java.awt.event.ActionListener() {  
 67: public void actionPerformed(java.awt.event.ActionEvent evt) {  
 68: btnOKActionPerformed(evt);  
 69: }  
 70: });  
 71:  
 72: btnCancel.setText("Cancel");  
 73: btnCancel.addActionListener(new java.awt.event.ActionListener() {  
 74: public void actionPerformed(java.awt.event.ActionEvent evt) {  
 75: btnCancelActionPerformed(evt);  
 76: }  
 77: });  
 78:  
 79: javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());  
 80: getContentPane().setLayout(layout);  
 81: layout.setHorizontalGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
 82: .addGroup(layout.createSequentialGroup().addContainerGap()  
 83: .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
 84: .addGroup(layout.createSequentialGroup()  
 85: .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
 86: .addComponent(jLabel1).addComponent(jLabel2))  
 87: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
 88: .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
 89: .addComponent(txtUser).addComponent(txtPassword)))  
 90: .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,  
 91: layout.createSequentialGroup().addGap(0, 245, Short.MAX\_VALUE).addComponent(btnCancel)  
 92: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED).addComponent(  
 93: btnOK)))  
 94: .addContainerGap()));  
 95: layout.setVerticalGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
 96: .addGroup(layout.createSequentialGroup().addContainerGap()  
 97: .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE).addComponent(jLabel1)  
 98: .addComponent(txtUser, javax.swing.GroupLayout.PREFERRED\_SIZE,  
 99: javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
 100: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
 101: .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE).addComponent(jLabel2)  
 102: .addComponent(txtPassword, javax.swing.GroupLayout.PREFERRED\_SIZE,  
 103: javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))  
 104: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
 105: .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE).addComponent(btnOK)  
 106: .addComponent(btnCancel))  
 107: .addContainerGap(13, Short.MAX\_VALUE)));  
 108:  
 109: pack();  
 110: }// </editor-fold>//GEN-END:initComponents  
 111:  
 112: private void btnOKActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_btnOKActionPerformed  
 113:  
 114: if (validateLogin()) {  
 115: LoginJanela.this.dispose();  
 116: } else {  
 117: JOptionPane.showMessageDialog(LoginJanela.this, "Usuário ou login informados não puderam ser validados.",  
 118: "Erro no login", JOptionPane.ERROR\_MESSAGE);  
 119: }  
 120:  
 121: }// GEN-LAST:event\_btnOKActionPerformed  
 122:  
 123: private void btnCancelActionPerformed(java.awt.event.ActionEvent evt) {// GEN-FIRST:event\_btnCancelActionPerformed  
 124: this.exitProgram();  
 125: }// GEN-LAST:event\_btnCancelActionPerformed  
 126:  
 127: /\*\*  
 128: \* @param args  
 129: \* the command line arguments  
 130: \*/  
 131: public static void main(String args[]) {  
 132: /\* Set the Nimbus look and feel \*/  
 133: // <editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">  
 134: /\*  
 135: \* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel. For details see  
 136: \* http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html  
 137: \*/  
 138: try {  
 139: for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {  
 140: if ("Nimbus".equals(info.getName())) {  
 141: javax.swing.UIManager.setLookAndFeel(info.getClassName());  
 142: break;  
 143: }  
 144: }  
 145: } catch (ClassNotFoundException ex) {  
 146: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
 147: } catch (InstantiationException ex) {  
 148: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
 149: } catch (IllegalAccessException ex) {  
 150: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
 151: } catch (javax.swing.UnsupportedLookAndFeelException ex) {  
 152: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
 153: }  
 154: // </editor-fold>  
 155:  
 156: /\* Create and display the dialog \*/  
 157: java.awt.EventQueue.invokeLater(new Runnable() {  
 158: public void run() {  
 159: LoginJanela dialog = new LoginJanela(new javax.swing.JFrame(), true);  
 160: dialog.addWindowListener(new java.awt.event.WindowAdapter() {  
 161: @Override  
 162: public void windowClosing(java.awt.event.WindowEvent e) {  
 163: System.exit(0);  
 164: }  
 165: });  
 166: dialog.setVisible(true);  
 167: }  
 168: });  
 169: }  
 170:  
 171: // Variables declaration - do not modify//GEN-BEGIN:variables  
 172: private javax.swing.JButton btnCancel;  
 173: private javax.swing.JToggleButton btnOK;  
 174: private javax.swing.JLabel jLabel1;  
 175: private javax.swing.JLabel jLabel2;  
 176: private javax.swing.JPasswordField txtPassword;  
 177: private javax.swing.JTextField txtUser;  
 178: // End of variables declaration//GEN-END:variables  
 179:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\package-info.java

1:/\*\*  
 2: \* Pacote raiz da aplicação  
 3: \*   
 4: \*/  
 5:package br.ufrj.coppetec.concentrador;

## .\concentrador\src\br\ufrj\coppetec\concentrador\PropertyLoader.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador;  
 7:  
 8:import java.io.FileNotFoundException;  
 9:import java.io.IOException;  
 10:import java.io.InputStream;  
 11:import java.util.Properties;  
 12:  
 13:import org.apache.logging.log4j.LogManager;  
 14:import org.apache.logging.log4j.Logger;  
 15:  
 16:/\*\*  
 17: \*  
 18: \* @author mangeli  
 19: \*/  
 20:public class PropertyLoader {  
 21: private static Logger logger = LogManager.getLogger(PropertyLoader.class);  
 22: String propertyFile;  
 23: Properties properties;  
 24:  
 25: PropertyLoader() throws IOException {  
 26: this("config.properties");  
 27: }  
 28:  
 29: PropertyLoader(String fileName) throws IOException {  
 30: propertyFile = fileName;  
 31: setProperty();  
 32: }  
 33:  
 34: private void setProperty() throws IOException {  
 35: InputStream inputStream = null;  
 36: Properties prop = new Properties();  
 37: try {  
 38: inputStream = getClass().getClassLoader().getResourceAsStream(propertyFile);  
 39:  
 40: if (inputStream != null) {  
 41: prop.load(inputStream);  
 42: } else {  
 43: throw new FileNotFoundException("property file '" + propertyFile + "' not found in the classpath");  
 44: }  
 45: } catch (Exception e) {  
 46: logger.catching(e);  
 47: throw e;  
 48: } finally {  
 49: inputStream.close();  
 50: }  
 51:  
 52: this.properties = prop;  
 53: logger.info("Properties file readed.");  
 54: }  
 55:  
 56: public Properties getProperties() {  
 57: return properties;  
 58: }  
 59:  
 60:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\Splash.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador;  
 7:  
 8:import javax.swing.ImageIcon;  
 9:  
 10:import org.apache.logging.log4j.LogManager;  
 11:import org.apache.logging.log4j.Logger;  
 12:  
 13:/\*\*  
 14: \* Tela exibida durante a inicialização da aplicação  
 15: \*   
 16: \* @author ludes - PESC - COPPE - ufrj  
 17: \* @author Eduardo Mangeli  
 18: \* @author Marcelo Areas  
 19: \* @author Fabrício Pereira  
 20: \* @author Geraldo Xexéo  
 21: \*/  
 22:public class Splash extends javax.swing.JFrame {  
 23:  
 24: private static Logger logger = LogManager.getLogger(Splash.class);  
 25:  
 26: /\*\*  
 27: \* Cria a tela de inicialização  
 28: \*/  
 29: public Splash() {  
 30: initComponents();  
 31:  
 32: }  
 33:  
 34: /\*\*  
 35: \* Ajusta a versão do programa a ser exibida na tela inicial  
 36: \*   
 37: \* @param version Versão do programa  
 38: \*/  
 39: public void setVersionLabet(String version) {  
 40: lblVersao.setText("Versão: " + version);  
 41: }  
 42:  
 43: /\*\*  
 44: \* Método utilizado pelo construtor para iniciar os componentes da tela  
 45: \*   
 46: \* WARNING: Não modifique esse método. O seu conteúdo é sempre gerado automaticamente pelo gerador de formulários  
 47: \* da IDE NetBeans.  
 48: \*/  
 49: @SuppressWarnings("unchecked")  
 50: // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents  
 51: private void initComponents() {  
 52:  
 53: jLabel1 = new javax.swing.JLabel();  
 54: lblVersao = new javax.swing.JLabel();  
 55:  
 56: setDefaultCloseOperation(javax.swing.WindowConstants.DO\_NOTHING\_ON\_CLOSE);  
 57: setIconImage(new ImageIcon(Splash.this.getClass().getResource("/images/icon.png")).getImage());  
 58: setUndecorated(true);  
 59:  
 60: jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/images/splash.png"))); // NOI18N  
 61:  
 62: lblVersao.setText("Versão: xxx");  
 63: lblVersao.setAlignmentX(0.5F);  
 64:  
 65: javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());  
 66: getContentPane().setLayout(layout);  
 67: layout.setHorizontalGroup(  
 68: layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
 69: .addComponent(jLabel1)  
 70: .addGroup(layout.createSequentialGroup()  
 71: .addContainerGap()  
 72: .addComponent(lblVersao, javax.swing.GroupLayout.PREFERRED\_SIZE, 125, javax.swing.GroupLayout.PREFERRED\_SIZE))  
 73: );  
 74: layout.setVerticalGroup(  
 75: layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
 76: .addGroup(layout.createSequentialGroup()  
 77: .addComponent(jLabel1)  
 78: .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
 79: .addComponent(lblVersao)  
 80: .addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))  
 81: );  
 82:  
 83: pack();  
 84: }// </editor-fold>//GEN-END:initComponents  
 85:  
 86: /\*\*  
 87: \* Método utilizado apenas durante o desenvolvimento para a realização de testes  
 88: \*   
 89: \* @param args Executado sem argumentos  
 90: \*   
 91: \*/  
 92: @Deprecated  
 93: public static void main(String args[]) {  
 94: /\* Set the Nimbus look and feel \*/  
 95: // <editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">  
 96: /\*  
 97: \* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel. For details see  
 98: \* http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html  
 99: \*/  
 100: try {  
 101: for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {  
 102: if ("Nimbus".equals(info.getName())) {  
 103: javax.swing.UIManager.setLookAndFeel(info.getClassName());  
 104: break;  
 105: }  
 106: }  
 107: } catch (ClassNotFoundException ex) {  
 108: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
 109: } catch (InstantiationException ex) {  
 110: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
 111: } catch (IllegalAccessException ex) {  
 112: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
 113: } catch (javax.swing.UnsupportedLookAndFeelException ex) {  
 114: logger.fatal("ERRO FATAL: a aplicação será finalizada.", ex);  
 115: }  
 116: // </editor-fold>  
 117:  
 118: /\* Create and display the form \*/  
 119: java.awt.EventQueue.invokeLater(new Runnable() {  
 120: public void run() {  
 121: new Splash().setVisible(true);  
 122: }  
 123: });  
 124: }  
 125:  
 126: // Variables declaration - do not modify//GEN-BEGIN:variables  
 127: private javax.swing.JLabel jLabel1;  
 128: private javax.swing.JLabel lblVersao;  
 129: // End of variables declaration//GEN-END:variables  
 130:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\Util.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador;  
 7:  
 8:import java.io.File;  
 9:import java.io.FileInputStream;  
 10:import java.io.IOException;  
 11:import java.lang.reflect.Array;  
 12:import java.security.MessageDigest;  
 13:import java.text.ParseException;  
 14:import java.text.SimpleDateFormat;  
 15:import java.util.Arrays;  
 16:import java.util.Calendar;  
 17:import java.util.Date;  
 18:import java.util.HashMap;  
 19:import java.util.HashSet;  
 20:import java.util.Set;  
 21:  
 22:import org.apache.logging.log4j.LogManager;  
 23:import org.apache.logging.log4j.Logger;  
 24:  
 25:/\*\*  
 26: \*  
 27: \* @author mangeli  
 28: \*/  
 29:public final class Util {  
 30:  
 31: public static final String TODAS = "Todas";  
 32: private static Logger logger = LogManager.getLogger(Util.class);  
 33:  
 34: private static HashMap<String, Integer> inputLimits = new HashMap<String, Integer>();  
 35:  
 36: public static final SimpleDateFormat sdfSQL = new SimpleDateFormat("yyyy-MM-dd");  
 37: public static final SimpleDateFormat sdfBrazil = new SimpleDateFormat("dd/MM/yyyy");  
 38: public static final SimpleDateFormat sdfArq = new SimpleDateFormat("yyyy-MM-dd-HH-mm-ss");  
 39:  
 40: private Util() {  
 41: }  
 42:  
 43: public static <T> T[] concatArrays(T[] a, T[] b) {  
 44: int aLen = a.length;  
 45: int bLen = b.length;  
 46:  
 47: @SuppressWarnings("unchecked")  
 48: T[] c = (T[]) Array.newInstance(a.getClass().getComponentType(), aLen + bLen);  
 49: System.arraycopy(a, 0, c, 0, aLen);  
 50: System.arraycopy(b, 0, c, aLen, bLen);  
 51: return c;  
 52: }  
 53:  
 54: public static String[] getValidDatesStr() {  
 55: String[] r;  
 56: r = Concentrador.configuration.getProperty("validDays").split(",");  
 57: for (int i = 0; i < r.length; i++) {  
 58: r[i] = r[i].trim();  
 59: }  
 60: return r;  
 61: }  
 62:  
 63: public static String[] getTrainingDatesStr() {  
 64: String[] r;  
 65: r = Concentrador.configuration.getProperty("trainingDays").split(",");  
 66: for (int i = 0; i < r.length; i++) {  
 67: r[i] = r[i].trim();  
 68: }  
 69: return r;  
 70: }  
 71:  
 72: public static boolean isInValidPeriod(Date date) throws ParseException {  
 73: return getValidDates().contains(lowDateTime(date));  
 74: }  
 75:  
 76: // public static boolean isOutValidPeriod(Date date) throws ParseException {  
 77: // return (getMinValidDate().compareTo(date) > 0) || (getMaxValidDate().compareTo(date) < 0);  
 78: // }  
 79: //  
 80: // public static boolean isBeforeValidPeriod(Date date) throws ParseException {  
 81: // return getMinValidDate().compareTo(date) > 0;  
 82: // }  
 83: //  
 84: // public static boolean isAfterValidPeriod(Date date) throws ParseException {  
 85: // return getMaxValidDate().compareTo(date) < 0;  
 86: // }  
 87: //  
 88: // public static boolean isInOrAfterValidPeriod(Date date) throws ParseException {  
 89: // // return getMinValidDate().compareTo(date) <= 0; /\* Se o intervalo não for contínuo, essa linha não é adequada! \*/  
 90: // return isInValidPeriod(date) || isAfterValidPeriod(date);  
 91: // }  
 92:  
 93: public static Set<Date> getValidDates() throws ParseException {  
 94: return parseDates(getValidDatesStr());  
 95: }  
 96:  
 97: public static Set<Date> getTrainingDates() throws ParseException {  
 98: return parseDates(getTrainingDatesStr());  
 99: }  
 100:  
 101: protected static Set<Date> parseDates(String[] datesStr) throws ParseException {  
 102: Set<Date> dates = new HashSet<Date>(datesStr.length);  
 103: for (int i = 0; i < datesStr.length; i++) {  
 104: try {  
 105: dates.add(sdfBrazil.parse(datesStr[i].trim()));  
 106: } catch (ParseException e) {  
 107: logger.error(String.format("Erro ao carregar datas (formato inválido da data: %s).", datesStr[i]), e);  
 108: throw e;  
 109: }  
 110: }  
 111: return dates;  
 112: }  
 113:  
 114: public static String getValidDatesListInSQL() throws ParseException {  
 115: return getDatesListInSQL(getValidDates());  
 116: }  
 117:  
 118: public static String getTrainingDatesListInSQL() throws ParseException {  
 119: return getDatesListInSQL(getTrainingDates());  
 120: }  
 121:  
 122: protected static String getDatesListInSQL(Set<Date> dates) throws ParseException {  
 123: String dateListInSQL = "";  
 124: for (Date date : dates) {  
 125: dateListInSQL += ", '" + sdfSQL.format(date) + "'";  
 126: }  
 127: dateListInSQL += ")";  
 128: return dateListInSQL.replaceFirst(", ", "(");  
 129: }  
 130:  
 131: // public static String getMinValidDateSQL() throws ParseException {  
 132: // return SDF\_SQL\_DATE\_ONLY.format(getMinValidDate());  
 133: // }  
 134: //  
 135: // public static String getMaxValidDateSQL() throws ParseException {  
 136: // return SDF\_SQL\_DATE\_ONLY.format(getMaxValidDate());  
 137: // }  
 138: //  
 139: // public static Date getMinValidDate() throws ParseException {  
 140: // Set<Date> sortedValidDates = new TreeSet<Date>(getValidDates());  
 141: // return sortedValidDates.iterator().next();  
 142: // }  
 143: //  
 144: // public static Date getMaxValidDate() throws ParseException {  
 145: // Set<Date> sortedValidDates = new TreeSet<Date>(Collections.reverseOrder());  
 146: // sortedValidDates.addAll(getValidDates());  
 147: // return highDateTime(sortedValidDates.iterator().next());  
 148: // }  
 149:  
 150: /\*\*  
 151: \* Retorna o valor do horário minimo para a data de referencia passada. <BR>  
 152: \* <BR>  
 153: \* Por exemplo se a data for "30/01/2017 as 17h:33m:12s e 299ms" a data retornada por este metodo será "30/01/2017 as  
 154: \* 00h:00m:00s e 000ms".  
 155: \*   
 156: \* @param date  
 157: \* de referencia.  
 158: \* @return {@link Date} que representa o horário minimo para dia informado.  
 159: \*/  
 160: public static Date lowDateTime(Date date) {  
 161: Calendar aux = Calendar.getInstance();  
 162: aux.setTime(date);  
 163: toOnlyDate(aux); // zera os parametros de hour,min,sec,millisec  
 164: return aux.getTime();  
 165: }  
 166:  
 167: // /\*\*  
 168: // \* Retorna o valor do horário maximo para a data de referencia passada. <BR>  
 169: // \* <BR>  
 170: // \* Por exemplo se a data for "30/01/2017 as 17h:33m:12s e 299ms" a data retornada por este metodo será "30/01/2017 as  
 171: // \* 23h:59m:59s e 999ms".  
 172: // \*  
 173: // \* @param date  
 174: // \* de referencia.  
 175: // \* @return {@link Date} que representa o horário maximo para dia informado.  
 176: // \*/  
 177: // public static Date highDateTime(Date date) {  
 178: // Calendar aux = Calendar.getInstance();  
 179: // aux.setTime(date);  
 180: // toFinalDate(aux); // maximiza os parametros de hour,min,sec,millisec  
 181: // return aux.getTime();  
 182: // }  
 183:  
 184: /\*\*  
 185: \* Zera todas as referencias de hora, minuto, segundo e milissegundo do {@link Calendar}.  
 186: \*   
 187: \* @param date  
 188: \* a ser modificado.  
 189: \*/  
 190: private static void toOnlyDate(Calendar date) {  
 191: date.set(Calendar.HOUR\_OF\_DAY, 0);  
 192: date.set(Calendar.MINUTE, 0);  
 193: date.set(Calendar.SECOND, 0);  
 194: date.set(Calendar.MILLISECOND, 0);  
 195: }  
 196:  
 197: // /\*\*  
 198: // \* Maximiza todas as referencias de hora, minuto, segundo e milissegundo do {@link Calendar}.  
 199: // \*  
 200: // \* @param date  
 201: // \* a ser modificado.  
 202: // \*/  
 203: // private static void toFinalDate(Calendar date) {  
 204: // date.set(Calendar.HOUR\_OF\_DAY, 23);  
 205: // date.set(Calendar.MINUTE, 59);  
 206: // date.set(Calendar.SECOND, 59);  
 207: // date.set(Calendar.MILLISECOND, 999);  
 208: // }  
 209: //  
 210: // public static Date incrementDay(Date date, int amount) {  
 211: // return incrementDayOfMonth(date, amount);  
 212: // }  
 213: //  
 214: // public static Date incrementDayOfMonth(Date date, int amount) {  
 215: // return incrementDate(date, Calendar.DAY\_OF\_MONTH, amount);  
 216: // }  
 217: //  
 218: // private static Date incrementDate(Date date, int field, int amount) {  
 219: // Calendar calendar = new GregorianCalendar();  
 220: // calendar.setTime(date);  
 221: // calendar.add(field, amount);  
 222: // return calendar.getTime();  
 223: // }  
 224:  
 225: public static void populateInputLimits() {  
 226: inputLimits.put("placaEstrangeira",  
 227: Integer.parseInt(Concentrador.configuration.getProperty("placaEstrangeira", "20").trim()));  
 228: inputLimits.put("frequenciaQtd", Integer.parseInt(Concentrador.configuration.getProperty("frequenciaQtd", "3").trim()));  
 229: inputLimits.put("numeroDePessoasNoVeiculo",  
 230: Integer.parseInt(Concentrador.configuration.getProperty("numeroDePessoasNoVeiculo", "2").trim()));  
 231: inputLimits.put("pesoDaCarga", Integer.parseInt(Concentrador.configuration.getProperty("pesoDaCarga", "5").trim()));  
 232: inputLimits.put("valorDoFrete", Integer.parseInt(Concentrador.configuration.getProperty("valorDoFrete", "5").trim()));  
 233: inputLimits.put("valorDaCarga", Integer.parseInt(Concentrador.configuration.getProperty("valorDaCarga", "15").trim()));  
 234: inputLimits.put("pesquisador\_1", Integer.parseInt(Concentrador.configuration.getProperty("pesquisador\_1", "20").trim()));  
 235: inputLimits.put("pesquisador\_2", Integer.parseInt(Concentrador.configuration.getProperty("pesquisador\_2", "20").trim()));  
 236: inputLimits.put("local", Integer.parseInt(Concentrador.configuration.getProperty("local", "20").trim()));  
 237: }  
 238:  
 239: public static String getSQLiteBoolean(String val) throws Exception {  
 240: if (inputLimits.isEmpty()) {  
 241: String msg = "inputLimits variable is not initialized.";  
 242: logger.info(msg);  
 243: throw new Exception(msg);  
 244: }  
 245: String r;  
 246: if (val != null) {  
 247: if (val.equals("false") || val.equals("0")) {  
 248: r = "0";  
 249: } else if (val.equals("true") || val.equals("1")) {  
 250: r = "1";  
 251: } else {  
 252: r = null;  
 253: }  
 254: } else {  
 255: r = null;  
 256: }  
 257: return r;  
 258: }  
 259:  
 260: public static String getSQLiteIntLimited(String val, String fieldName) throws Exception {  
 261: if (inputLimits.isEmpty()) {  
 262: String msg = "inputLimits variable is not initialized.";  
 263: logger.info(msg);  
 264: throw new Exception(msg);  
 265: }  
 266: String r = null;  
 267: if (val != null) {  
 268: Integer limit = inputLimits.get(fieldName);  
 269: if (limit == null) {  
 270: limit = 2;  
 271: }  
 272: r = cutString(val, limit);  
 273:  
 274: try {  
 275: Integer.parseInt(r);  
 276: } catch (NumberFormatException e) {  
 277: logger.debug("Erro na conversão do campo " + fieldName, e);  
 278: r = new String(new char[limit]).replace("\0", "9");  
 279: }  
 280: }  
 281: return r;  
 282: }  
 283:  
 284: public static String getSQLiteRealLimited(String val, String fieldName) throws Exception {  
 285: if (inputLimits.isEmpty()) {  
 286: String msg = "inputLimits variable is not initialized.";  
 287: logger.info(msg);  
 288: throw new Exception(msg);  
 289: }  
 290: String r = null;  
 291: if (val != null) {  
 292: Integer limit = inputLimits.get(fieldName);  
 293: if (limit == null) {  
 294: limit = 2;  
 295: }  
 296: // if is a decimal separator  
 297: if (val.indexOf('.') != -1) {  
 298: String intPart = cutString(val.split("\\.")[0], limit);  
 299: r = intPart + "." + cutString(val.split("\\.")[1], 2);  
 300:  
 301: } else {  
 302: r = cutString(val, limit);  
 303: }  
 304:  
 305: try {  
 306: Float.parseFloat(r);  
 307: } catch (NumberFormatException e) {  
 308: logger.debug("Erro na conversão do campo " + fieldName, e);  
 309: r = new String(new char[limit]).replace("\0", "9");  
 310: }  
 311: }  
 312: return r;  
 313: }  
 314:  
 315: public static String getStringLimited(String val, String fieldName) throws Exception {  
 316: if (inputLimits.isEmpty()) {  
 317: String msg = "inputLimits variable is not initialized.";  
 318: logger.info(msg);  
 319: throw new Exception(msg);  
 320: }  
 321: String r = null;  
 322: if (val != null) {  
 323: Integer limit = inputLimits.get(fieldName);  
 324: if (limit == null) {  
 325: limit = 20;  
 326: }  
 327: if (val.length() > limit) {  
 328: r = val.substring(0, limit - 1);  
 329: } else {  
 330: r = val;  
 331: }  
 332: }  
 333: return r;  
 334: }  
 335:  
 336: private static String cutString(String str, int maxLength) {  
 337: String r;  
 338: if (str.length() > maxLength) {  
 339: char[] chars = new char[maxLength];  
 340: Arrays.fill(chars, '9');  
 341: r = new String(chars);  
 342: } else {  
 343: r = str;  
 344: }  
 345: return r;  
 346: }  
 347:  
 348: public static String getFileChecksum(MessageDigest digest, File file) throws IOException {  
 349: // Get file input stream for reading the file content  
 350: FileInputStream fis = new FileInputStream(file);  
 351:  
 352: // Create byte array to read data in chunks  
 353: byte[] byteArray = new byte[1024];  
 354: int bytesCount = 0;  
 355:  
 356: // Read file data and update in message digest  
 357: while ((bytesCount = fis.read(byteArray)) != -1) {  
 358: digest.update(byteArray, 0, bytesCount);  
 359: }  
 360: ;  
 361:  
 362: // close the stream; We don't need it now.  
 363: fis.close();  
 364:  
 365: // Get the hash's bytes  
 366: byte[] bytes = digest.digest();  
 367:  
 368: // This bytes[] has bytes in decimal format;  
 369: // Convert it to hexadecimal format  
 370: StringBuilder sb = new StringBuilder();  
 371: for (int i = 0; i < bytes.length; i++) {  
 372: sb.append(Integer.toString((bytes[i] & 0xff) + 0x100, 16).substring(1));  
 373: }  
 374:  
 375: // return complete hash  
 376: return sb.toString();  
 377: }  
 378:  
 379:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\WebServer.java

1:package br.ufrj.coppetec.concentrador;  
 2:  
 3:import java.io.IOException;  
 4:import java.io.PrintStream;  
 5:import java.net.InetSocketAddress;  
 6:import java.net.SocketAddress;  
 7:import java.text.SimpleDateFormat;  
 8:import java.util.Date;  
 9:  
 10:import javax.swing.JTextArea;  
 11:import javax.swing.JTextField;  
 12:  
 13:import org.apache.logging.log4j.LogManager;  
 14:import org.apache.logging.log4j.Logger;  
 15:import org.simpleframework.http.Query;  
 16:import org.simpleframework.http.Request;  
 17:import org.simpleframework.http.Response;  
 18:import org.simpleframework.http.Status;  
 19:import org.simpleframework.http.core.Container;  
 20:import org.simpleframework.http.core.ContainerSocketProcessor;  
 21:import org.simpleframework.transport.SocketProcessor;  
 22:import org.simpleframework.transport.connect.Connection;  
 23:import org.simpleframework.transport.connect.SocketConnection;  
 24:  
 25:/\*\*  
 26: \*  
 27: \* @author mangeli  
 28: \*/  
 29:public class WebServer implements Container {  
 30:  
 31: private static Logger logger = LogManager.getLogger(WebServer.class);  
 32:  
 33: private JTextArea observer;  
 34: private JTextField txtPort;  
 35: private SocketProcessor realServer;  
 36: private SocketAddress address;  
 37: private Connection conn;  
 38: private int listeningPort;  
 39:  
 40: public WebServer(JTextArea observer, JTextField txtPort) throws IOException {  
 41: this.observer = observer;  
 42: this.txtPort = txtPort;  
 43: this.realServer = new ContainerSocketProcessor(this);  
 44: this.conn = new SocketConnection(this.realServer);  
 45:  
 46: int initialPort = 8000;  
 47: int finalPort = 8090;  
 48: this.listeningPort = 0;  
 49:  
 50: for (int port = initialPort; port <= finalPort; port++) {  
 51: try {  
 52: this.address = new InetSocketAddress(port);  
 53: this.conn.connect(address);  
 54: this.listeningPort = port;  
 55: break;  
 56: } catch (Exception e) {  
 57: logger.error(String.format("Erro ao escutar a porta %d", port), e);  
 58: observer.append("Erro tentando escutar a porta " + port + ": " + e.getMessage() + "\n");  
 59: txtPort.setText("");  
 60: }  
 61: }  
 62: if (listeningPort != 0) {  
 63: observer.append("Servidor escutando a porta " + this.listeningPort + "\n");  
 64: txtPort.setText(Integer.toString(listeningPort));  
 65: }  
 66:  
 67: }  
 68:  
 69: @Override  
 70: public void handle(Request req, Response resp) {  
 71: try {  
 72: PrintStream body = resp.getPrintStream();  
 73: Query query = req.getQuery();  
 74: long time = System.currentTimeMillis();  
 75: String caminho = "";  
 76:  
 77: resp.setValue("Content-Type", "text/plain");  
 78: resp.setValue("Server", "Concentrador/1.0");  
 79: resp.setDate("Date", time);  
 80: resp.setDate("Last-Modified", time);  
 81:  
 82: // se existe um caminho na url  
 83: if (req.getPath().getSegments().length > 0) {  
 84: caminho = req.getPath().getSegments()[0];  
 85: }  
 86:  
 87: switch (caminho) {  
 88: case "":  
 89: this.sendHello(body);  
 90: break;  
 91: case "dados":  
 92: if (query.size() > 0 && query.get("team").equals("9393")) {  
 93: this.storeValues(body, query);  
 94: } else {  
 95: this.sendError(body, resp);  
 96: }  
 97: break;  
 98: default:  
 99: this.sendError(body, resp);  
 100: }  
 101:  
 102: } catch (Exception e) {  
 103: logger.error("Erro ao receber dados.", e);  
 104: observer.append("ERRO recebendo dados: " + e.getMessage() + "\n");  
 105: } finally {  
 106:  
 107: }  
 108: }  
 109:  
 110: private void sendHello(PrintStream body) {  
 111: body.println("Servidor do concentrador");  
 112: body.close();  
 113: }  
 114:  
 115: private void storeValues(PrintStream body, Query query) {  
 116: this.observer.append("Dados corretos");  
 117: }  
 118:  
 119: private void sendError(PrintStream body, Response resp) throws Exception {  
 120: resp.setStatus(Status.NOT\_FOUND);  
 121: body.close();  
 122: throw new Exception("tentativa de acesso não autorizado " + new SimpleDateFormat("dd/MM/yyyy k:m:s").format(new Date()));  
 123: }  
 124:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\database\Db.java

1:package br.ufrj.coppetec.concentrador.database;  
 2:  
 3:import java.sql.Connection;  
 4:import java.sql.DriverManager;  
 5:import java.sql.ResultSet;  
 6:import java.sql.SQLException;  
 7:import java.sql.Statement;  
 8:  
 9:import org.apache.logging.log4j.LogManager;  
 10:import org.apache.logging.log4j.Logger;  
 11:  
 12:/\*\*  
 13: \*  
 14: \* @author mangeli  
 15: \*/  
 16:abstract class Db {  
 17:  
 18: private static Logger logger = LogManager.getLogger(Db.class);  
 19:  
 20: private String driver = "";  
 21: private String url = null;  
 22: private int timeout = 30;  
 23: private Connection conn = null;  
 24: private Statement statement = null;  
 25:  
 26: public Db(String driver, String url) throws Exception {  
 27: this.driver = driver;  
 28: this.url = url;  
 29: setConnection();  
 30: setStatement();  
 31: }  
 32:  
 33: public Db(String driver) throws Exception {  
 34: this.driver = driver;  
 35: this.conn = null;  
 36: }  
 37:  
 38: public void rollback() {  
 39: try {  
 40: if (this.conn != null) {  
 41: this.conn.rollback();  
 42: logger.info("Rollback!");  
 43: }  
 44: } catch (Exception e) {  
 45: logger.error("Erro ao fazer rollback.", e);  
 46: }  
 47: }  
 48:  
 49: public void setConnection() throws Exception {  
 50: this.setConnection(this.url);  
 51: }  
 52:  
 53: public void setConnection(String url) throws Exception {  
 54: logger.info("Driver: " + this.driver);  
 55: Class.forName(this.driver);  
 56: this.conn = DriverManager.getConnection(url);  
 57: this.conn.setAutoCommit(false);  
 58: logger.info("Conexão com o BD estabelecida.");  
 59: }  
 60:  
 61: public Connection getConnection() throws Exception {  
 62: return this.conn;  
 63: }  
 64:  
 65: public void openTransaction() throws Exception {  
 66: this.openTransaction(this.url);  
 67: }  
 68:  
 69: public void openTransaction(String url) throws Exception {  
 70: if (this.conn == null)  
 71: setConnection(url);  
 72: this.conn.setAutoCommit(false);  
 73: }  
 74:  
 75: public void commit() throws Exception {  
 76: this.conn.commit();  
 77: }  
 78:  
 79: public final void setStatement() throws Exception {  
 80: if (this.conn == null) {  
 81: setConnection();  
 82: }  
 83: this.statement = conn.createStatement();  
 84: this.statement.setQueryTimeout(this.timeout);  
 85: }  
 86:  
 87: public Statement getStatement() {  
 88: return this.statement;  
 89: }  
 90:  
 91: public int executeStatement(String instruction) throws SQLException {  
 92: return this.statement.executeUpdate(instruction);  
 93: }  
 94:  
 95: public ResultSet executeQuery(String instruction) throws SQLException {  
 96: return this.statement.executeQuery(instruction);  
 97: }  
 98:  
 99: public void closeConnection() {  
 100: if (this.statement != null) {  
 101: try {  
 102: this.statement.close();  
 103: logger.info("Database statement closed.");  
 104: } catch (Exception e) {  
 105: logger.error("Erro ao encerrar a transação com o BD.", e);  
 106: }  
 107: }  
 108: if (this.conn != null) {  
 109: try {  
 110: this.conn.close();  
 111: logger.info("Database connection closed.");  
 112: } catch (Exception e) {  
 113: logger.error("Erro ao fechar a conexão com o BD.", e);  
 114: }  
 115: }  
 116:  
 117: }  
 118:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\database\DBFileImporter.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador.database;  
 7:  
 8:import java.io.File;  
 9:import java.io.FileFilter;  
 10:import java.io.IOException;  
 11:import java.security.NoSuchAlgorithmException;  
 12:import java.sql.SQLException;  
 13:import java.util.Map;  
 14:  
 15:import javax.naming.NoPermissionException;  
 16:  
 17:import org.apache.logging.log4j.LogManager;  
 18:import org.apache.logging.log4j.Logger;  
 19:  
 20:/\*\*  
 21: \*  
 22: \* @author mangeli  
 23: \*/  
 24:public class DBFileImporter {  
 25: private static Logger logger = LogManager.getLogger(DBFileImporter.class);  
 26: private File folder;  
 27: private Map<String, String> storedDbFiles; // <path, hasg>  
 28: private Map<File, String> newDbFiles;  
 29: private int counter;  
 30:  
 31: public DBFileImporter(File folder) throws Exception {  
 32: this.folder = folder;  
 33: getStoredFiles();  
 34: counter = 0;  
 35: }  
 36:  
 37: private void getStoredFiles() throws Exception {  
 38: // myDB database = new myDB();  
 39: // ResultSet result;  
 40: // String qry = "SELECT \* FROM importedFiles;";  
 41: // database.setStatement();  
 42: // result = database.executeQuery(qry);  
 43: // database.setStatement();  
 44: // while (result.next()) {  
 45: // storedDbFiles.put(result.getString("path"), result.getString("hash"));  
 46: // }  
 47: // result.close();  
 48: }  
 49:  
 50: public void saveFile(myDB concentrador, String path, String fileName) throws Exception {  
 51: String sqlInsert = "INSERT INTO " + myDB.TABLE\_IMPORTED\_FILES + " (" + myDB.TABLE\_IMPORTED\_FILES\_ID + ","  
 52: + myDB.TABLE\_IMPORTED\_FILES\_PATH + ") VALUES ('" + fileName + "','" + path + "')";  
 53: logger.debug("SQL: " + sqlInsert);  
 54: concentrador.setStatement();  
 55: concentrador.executeStatement(sqlInsert);  
 56: }  
 57:  
 58: public void readNewFiles() throws NoSuchAlgorithmException, IOException, NoPermissionException {  
 59: FileFilter fileFilter = new FileFilter() {  
 60: @Override  
 61: public boolean accept(File f) {  
 62: return f.getAbsolutePath().endsWith(".db");  
 63: }  
 64: };  
 65:  
 66: String hash = null;  
 67:  
 68: File[] fileList = folder.listFiles(fileFilter);  
 69: if (fileList == null)  
 70: throw new NoPermissionException("Verifique a permissao da pasta!");  
 71:  
 72: for (File f : fileList) {  
 73: myDB concentrador = null;  
 74: ImportedDB db = null;  
 75: boolean novoArquivo = true;  
 76: try {  
 77: concentrador = myDB.getInstance();  
 78: concentrador.openTransaction();  
 79: logger.info("Importando arquivo: " + f.getAbsolutePath());  
 80: novoArquivo = false;  
 81: saveFile(concentrador, f.getAbsolutePath(), f.getName());  
 82: novoArquivo = true;  
 83: db = new ImportedDB(f.getAbsolutePath(), null);  
 84: counter += db.importData(concentrador);  
 85: db.closeConnection();  
 86: db = null;  
 87: concentrador.commit();  
 88: logger.info("Arquivo importado: " + f.getAbsolutePath());  
 89: } catch (IOException ex) {  
 90: logger.error(String.format("Erro ao acessar o arquivo %s", f.getAbsolutePath()));  
 91: if (concentrador != null)  
 92: concentrador.rollback();  
 93: } catch (SQLException e) {  
 94: if (novoArquivo)  
 95: logger.error(String.format("Erro ao importar os dados do arquivo %s", f.getAbsolutePath()), e);  
 96: if (concentrador != null)  
 97: concentrador.rollback();  
 98: } catch (Exception e) {  
 99: logger.error(String.format("Erro ao importar os dados do arquivo %s", f.getAbsolutePath()), e);  
 100: if (concentrador != null)  
 101: concentrador.rollback();  
 102: } finally {  
 103: if (db != null)  
 104: db.closeConnection();  
 105: }  
 106: }  
 107: }  
 108:  
 109: public int getCounter() {  
 110: return counter;  
 111: }  
 112:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\database\ImportedDB.java

1:package br.ufrj.coppetec.concentrador.database;  
 2:  
 3:import java.awt.BorderLayout;  
 4:import java.awt.Dialog.ModalityType;  
 5:import java.beans.PropertyChangeEvent;  
 6:import java.beans.PropertyChangeListener;  
 7:import java.sql.ResultSet;  
 8:  
 9:import javax.swing.JDialog;  
 10:import javax.swing.JLabel;  
 11:import javax.swing.JPanel;  
 12:import javax.swing.JProgressBar;  
 13:import javax.swing.SwingWorker;  
 14:import javax.swing.WindowConstants;  
 15:  
 16:import org.apache.logging.log4j.LogManager;  
 17:import org.apache.logging.log4j.Logger;  
 18:  
 19:import br.ufrj.coppetec.concentrador.Concentrador;  
 20:import br.ufrj.coppetec.concentrador.Janela;  
 21:import br.ufrj.coppetec.concentrador.Util;  
 22:  
 23:/\*\*  
 24: \*  
 25: \* @author mangeli  
 26: \*/  
 27:public class ImportedDB extends Db {  
 28:  
 29: private static Logger logger = LogManager.getLogger(ImportedDB.class);  
 30: private Janela janela;  
 31: private int counter;  
 32:  
 33: public ImportedDB(String path, Janela janela) throws Exception {  
 34: super("org.sqlite.JDBC", "jdbc:sqlite:" + path);  
 35: this.janela = janela;  
 36: counter = 0;  
 37: }  
 38:  
 39: public void sanitize() throws Exception {  
 40: openTransaction();  
 41: this.executeStatement("DELETE FROM tblDados WHERE id is null or id = 'null';");  
 42: commit();  
 43: }  
 44:  
 45: public int importData() throws Exception {  
 46: this.sanitize();  
 47: this.openTransaction();  
 48: this.counter = importData(Concentrador.database);  
 49: this.commit();  
 50: // final JDialog dialog = new JDialog(this.janela, "Dialog", ModalityType.APPLICATION\_MODAL);  
 51: // JOptionPane.showMessageDialog(this.janela, "Total de registros inseridos: " + Integer.toString(this.counter));  
 52: logger.debug("Novos registros: " + Integer.toString(this.counter));  
 53: return counter;  
 54: }  
 55:  
 56: public int importData(myDB concentradorDb) throws Exception {  
 57: Db db = this;  
 58: this.sanitize();  
 59: SwingWorker<Integer, Void> mySwingWorker = new SwingWorker<Integer, Void>() {  
 60:  
 61: @Override  
 62: protected Integer doInBackground() throws Exception {  
 63: db.setStatement();  
 64: // String queryToImport = String.format("SELECT \* FROM tblDados WHERE treinamento = %d;",  
 65: // (Concentrador.treinamento ? 1 : 0));  
 66: /\* Outra possibilidade para a mesma consulta acima: \*/  
 67: // String queryToImport = String.format("SELECT \* FROM tblDados WHERE %s;",  
 68: // myDB.getConditionByValidDate(myDB.TABLE\_NAME\_OD));  
 69: /\* Mas como o ideal é importar tudo, e deixar que o concentrador exiba as informações conforme o contexto. \*/  
 70: String queryToImport = "SELECT \* FROM tblDados;";  
 71:  
 72: ResultSet rs = db.executeQuery(queryToImport);  
 73: String sqlbase = "INSERT OR IGNORE INTO odTable (";  
 74:  
 75: sqlbase += "id, ";  
 76: sqlbase += "enviado, ";  
 77: sqlbase += "estaNoNote, ";  
 78: sqlbase += "cancelado, ";  
 79: sqlbase += "idPosto, ";  
 80: sqlbase += "sentido, ";  
 81: sqlbase += "idIpad, ";  
 82: // sqlbase += "uuid, "  
 83: sqlbase += "login, ";  
 84: sqlbase += "dataIniPesq, ";  
 85: sqlbase += "dataFimPesq, ";  
 86: sqlbase += "placa, ";  
 87: sqlbase += "anoDeFabricacao, ";  
 88: sqlbase += "tipo, ";  
 89: sqlbase += "idOrigemPais, ";  
 90: sqlbase += "idOrigemMunicipio, ";  
 91: sqlbase += "idDestinoPais, ";  
 92: sqlbase += "idDestinoMunicipio, ";  
 93: sqlbase += "idMotivoDeEscolhaDaRota, ";  
 94: sqlbase += "frequenciaQtd, ";  
 95: sqlbase += "frequenciaPeriodo, ";  
 96: sqlbase += "idPropriedadesDoVeiculo, ";  
 97: sqlbase += "placaEstrangeira, ";  
 98: sqlbase += "idPaisPlacaEstrangeira, ";  
 99: sqlbase += "idCombustivel, ";  
 100: sqlbase += "categoria, ";  
 101: sqlbase += "possuiReboque, ";  
 102: sqlbase += "numeroDePessoasNoVeiculo, ";  
 103: sqlbase += "numeroDePessoasATrabalho, ";  
 104: sqlbase += "idRendaMedia, ";  
 105: sqlbase += "idMotivoDaViagem, ";  
 106: sqlbase += "tipoCaminhao, ";  
 107: sqlbase += "idTipoDeContainer, ";  
 108: sqlbase += "idTipoCarroceria, ";  
 109: sqlbase += "rntrc, ";  
 110: sqlbase += "possuiCargaPerigosa, ";  
 111: sqlbase += "idNumeroDeRisco, ";  
 112: sqlbase += "idNumeroDaOnu, ";  
 113: sqlbase += "idAgregado, ";  
 114: sqlbase += "placaVermelha, ";  
 115: sqlbase += "idTipoDeViagemOuServico, ";  
 116: sqlbase += "pesoDaCarga, ";  
 117: sqlbase += "valorDoFrete, ";  
 118: sqlbase += "utilizaParadaEspecial, ";  
 119: sqlbase += "idProduto, ";  
 120: sqlbase += "idCargaAnterior, ";  
 121: sqlbase += "valorDaCarga, ";  
 122: sqlbase += "municipioEmbarqueCarga, ";  
 123: sqlbase += "idLocalEmbarqueCarga, ";  
 124: sqlbase += "municipioDesembarqueCarga, ";  
 125: sqlbase += "idLocalDesembarqueCarga, ";  
 126: sqlbase += "indoPegarCarga, ";  
 127: sqlbase += "paradaObrigatoriaMunicipio1, ";  
 128: sqlbase += "paradaObrigatoriaMunicipio2, ";  
 129: sqlbase += "idPerguntaExtra, ";  
 130: sqlbase += "idPerguntaExtra2, ";  
 131: sqlbase += "duracaoPesq, ";  
 132: sqlbase += "treinamento";  
 133: // IMPORTANTE: se adicionar mais campos ao final, não esquecer do separador no campo anterior  
 134: sqlbase += ") ";  
 135: String sql = "";  
 136: int idPosto = 0;  
 137: String idCombustivel = "";  
 138: String treinamento = "";  
 139: String cancelado = "";  
 140: boolean placaEstrangeira;  
 141: int counter = 0;  
 142: int counterProducao = 0;  
 143: int counterTraining = 0;  
 144: int counterProducaoCancelados = 0;  
 145: int counterTrainingCancelados = 0;  
 146: int counterPostoProducao = 0;  
 147: int counterPostoTraining = 0;  
 148: int counterPostoProducaoCancelados = 0;  
 149: int counterPostoTrainingCancelados = 0;  
 150: int rowsAffected = 0;  
 151: int counterInconsistentes = 0;  
 152: int counterInconsistentesTreinamento = 0;  
 153: while (rs.next()) {  
 154: try {  
 155: idPosto = 0;  
 156: cancelado = "";  
 157: treinamento = "";  
 158: idPosto = Integer.parseInt(rs.getString("idPosto"));  
 159: cancelado = Util.getSQLiteBoolean(rs.getString("cancelado"));  
 160: treinamento = Util.getSQLiteBoolean(rs.getString("treinamento"));  
 161:  
 162: /\*  
 163: \* IMPORTANTE: não se deve fazer nenhum outro tratamento nos dados recuperados (além dos três acima) antes  
 164: \* de começar a contruir o SQL com pelo menos os campos identificadores. Portanto é imprescindível que o  
 165: \* tratamento do campo seja feito logo antes de usá-lo para montrar a string SQL. Isso impacta no log caso  
 166: \* ocorra algum erro/exceção nesse tratamento. Exemplos: placaEstrangeira e idCombustivel.  
 167: \*/  
 168:  
 169: sql = sqlbase + " VALUES (";  
 170: sql += " '" + rs.getString("id") + "', ";  
 171: sql += "0, "; // não enviado  
 172: sql += "1, "; // esta no note  
 173: sql += cancelado + ", ";  
 174: sql += idPosto + ", ";  
 175: sql += "'" + rs.getString("sentido") + "', ";  
 176: sql += "'" + rs.getString("idIpad") + "', ";  
 177: // sql += "'" + rs.getString("uuid") + "', ";  
 178: sql += "'" + rs.getString("login") + "', ";  
 179: sql += "'" + rs.getString("dataIniPesq") + "', ";  
 180: sql += "'" + rs.getString("dataFimPesq") + "', ";  
 181: placaEstrangeira = Util.getSQLiteBoolean(rs.getString("placaEstrangeira")).equals("1");  
 182: if (placaEstrangeira) {  
 183: sql += "'" + Util.getStringLimited(rs.getString("placa"), "placaEstrangeira") + "', ";  
 184: } else {  
 185: sql += "'" + rs.getString("placa") + "', ";  
 186: }  
 187: sql += rs.getString("anoDeFabricacao") + ", ";  
 188: sql += "'" + rs.getString("tipo") + "', ";  
 189: sql += rs.getString("idOrigemPais") + ", ";  
 190: sql += rs.getString("idOrigemMunicipio") + ", ";  
 191: sql += rs.getString("idDestinoPais") + ", ";  
 192: sql += rs.getString("idDestinoMunicipio") + ", ";  
 193: sql += rs.getString("idMotivoDeEscolhaDaRota") + ", ";  
 194: sql += Util.getSQLiteIntLimited(rs.getString("frequenciaQtd"), "frequenciaQtd") + ", ";  
 195: sql += "'" + rs.getString("frequenciaPeriodo") + "', ";  
 196: sql += rs.getString("idPropriedadesDoVeiculo") + ", ";  
 197: sql += Util.getSQLiteBoolean(rs.getString("placaEstrangeira")) + ", ";  
 198: sql += rs.getString("idPaisPlacaEstrangeira") + ", ";  
 199: idCombustivel = (rs.getString("idCombustivel") == null && cancelado.equals("0")) ? "3"  
 200: : rs.getString("idCombustivel");  
 201: // sql += rs.getString("idCombustivel") + ", ";  
 202: sql += idCombustivel + ", ";  
 203: sql += "'" + rs.getString("categoria") + "', ";  
 204: sql += Util.getSQLiteBoolean(rs.getString("possuiReboque")) + ", ";  
 205: sql += Util.getSQLiteIntLimited(rs.getString("numeroDePessoasNoVeiculo"), "numeroDePessoasNoVeiculo")  
 206: + ", ";  
 207: sql += Util.getSQLiteIntLimited(rs.getString("numeroDePessoasATrabalho"), "numeroDePessoasATrabalho")  
 208: + ", ";  
 209: sql += rs.getString("idRendaMedia") + ", ";  
 210: sql += rs.getString("idMotivoDaViagem") + ", ";  
 211: sql += "'" + rs.getString("tipoCaminhao") + "', ";  
 212: sql += rs.getString("idTipoDeContainer") + ", ";  
 213: sql += rs.getString("idTipoCarroceria") + ", ";  
 214: sql += "'" + rs.getString("rntrc") + "', ";  
 215: sql += Util.getSQLiteBoolean(rs.getString("possuiCargaPerigosa")) + ", ";  
 216: sql += rs.getString("idNumeroDeRisco") + ", ";  
 217: sql += rs.getString("idNumeroDaOnu") + ", ";  
 218: sql += rs.getString("idAgregado") + ", ";  
 219: sql += Util.getSQLiteBoolean(rs.getString("placaVermelha")) + ", ";  
 220: sql += rs.getString("idTipoDeViagemOuServico") + ", ";  
 221: sql += Util.getSQLiteRealLimited(rs.getString("pesoDaCarga"), "pesoDaCarga") + ", ";  
 222: sql += Util.getSQLiteRealLimited(rs.getString("valorDoFrete"), "valorDoFrete") + ", ";  
 223: sql += Util.getSQLiteBoolean(rs.getString("utilizaParadaEspecial")) + ", ";  
 224: sql += rs.getString("idProduto") + ", ";  
 225: sql += rs.getString("idCargaAnterior") + ", ";  
 226: sql += Util.getSQLiteRealLimited(rs.getString("valorDaCarga"), "valorDaCarga") + ", ";  
 227: sql += rs.getString("municipioEmbarqueCarga") + ", ";  
 228: sql += rs.getString("idLocalEmbarqueCarga") + ", ";  
 229: sql += rs.getString("municipioDesembarqueCarga") + ", ";  
 230: sql += rs.getString("idLocalDesembarqueCarga") + ", ";  
 231: sql += Util.getSQLiteBoolean(rs.getString("indoPegarCarga")) + ", ";  
 232: sql += rs.getString("paradaObrigatoriaMunicipio1") + ", ";  
 233: sql += rs.getString("paradaObrigatoriaMunicipio2") + ", ";  
 234: sql += rs.getString("idPerguntaExtra") + ", ";  
 235: sql += rs.getString("idPerguntaExtra2") + ", ";  
 236: sql += rs.getString("duracaoPesq") + ", ";  
 237: sql += treinamento;  
 238: /\* IMPORTANTE: se adicionar mais campos ao final, não esquecer do separador no campo anterior. \*/  
 239: sql += "); ";  
 240: concentradorDb.setStatement();  
 241:  
 242: rowsAffected = concentradorDb.executeStatement(sql);  
 243:  
 244: if (treinamento.equals("0")) {  
 245: counterProducao += rowsAffected;  
 246: if (cancelado.equals("1")) {  
 247: counterProducaoCancelados += rowsAffected;  
 248: }  
 249: if (idPosto == Concentrador.getPostoInt()) {  
 250: counterPostoProducao += rowsAffected;  
 251: if (cancelado.equals("1")) {  
 252: counterPostoProducaoCancelados += rowsAffected;  
 253: }  
 254: }  
 255: } else {  
 256: counterTraining += rowsAffected;  
 257: if (cancelado.equals("1")) {  
 258: counterTrainingCancelados += rowsAffected;  
 259: }  
 260: if (idPosto == Concentrador.getPostoInt()) {  
 261: counterPostoTraining += rowsAffected;  
 262: if (cancelado.equals("1")) {  
 263: counterPostoTrainingCancelados += rowsAffected;  
 264: }  
 265: }  
 266: }  
 267: //conta apenas os registros não cancelados do posto logado e do modo atual, apesar de importar TODOS!  
 268: if(idPosto == Concentrador.getPostoInt() && !cancelado.equals("1")){  
 269: if ((Concentrador.treinamento && treinamento.equals("1")) || (!Concentrador.treinamento && treinamento.equals("0"))){  
 270: counter += rowsAffected;  
 271: }  
 272: }  
 273:   
 274: } catch (Exception e) {  
 275: // do nothing only ignore error  
 276: logger.error(String.format("Erro a inserir o registro:%s%s", System.lineSeparator(), sql), e);  
 277: counterInconsistentes++;  
 278: if (treinamento.equals("1")) {  
 279: counterInconsistentesTreinamento++;  
 280: }  
 281: }  
 282: // System.out.println(sql);  
 283: }  
 284: rs.close();  
 285: String logImports = "Foram importados %d registros do período de %s, em que %d são válidos,"  
 286: + " e %d são do POSTO %s, com %d registros válidos para este posto.";  
 287: logger.info(String.format(logImports, counterProducao, "PRODUÇÃO", counterProducao - counterProducaoCancelados,  
 288: counterPostoProducao, Concentrador.posto, counterPostoProducao - counterPostoProducaoCancelados));  
 289: logger.info(String.format(logImports, counterTraining, "TREINAMENTO", counterTraining - counterTrainingCancelados,  
 290: counterPostoTraining, Concentrador.posto, counterPostoTraining - counterPostoTrainingCancelados));  
 291: logger.info(String.format(  
 292: "Foram detectados %d registros inconsistentes dentre os quais, pelo menos %d são da fase de treinamento.",  
 293: counterInconsistentes, counterInconsistentesTreinamento));  
 294: return counter;  
 295: }  
 296:  
 297: };  
 298:  
 299: final JDialog dialog = new JDialog(janela, "Aguarde", ModalityType.APPLICATION\_MODAL);  
 300: dialog.setDefaultCloseOperation(WindowConstants.DO\_NOTHING\_ON\_CLOSE);  
 301: mySwingWorker.addPropertyChangeListener(new PropertyChangeListener() {  
 302:  
 303: @Override  
 304: public void propertyChange(PropertyChangeEvent evt) {  
 305: if (evt.getPropertyName().equals("state")) {  
 306: if (evt.getNewValue() == SwingWorker.StateValue.DONE) {  
 307: dialog.dispose();  
 308: }  
 309: }  
 310: }  
 311:  
 312: });  
 313: mySwingWorker.execute();  
 314:  
 315: JProgressBar progressBar = new JProgressBar();  
 316: progressBar.setIndeterminate(true);  
 317: JPanel panel = new JPanel(new BorderLayout());  
 318: panel.add(progressBar, BorderLayout.CENTER);  
 319: panel.add(new JLabel("Aguarde a importação dos dados......."), BorderLayout.PAGE\_START);  
 320: dialog.add(panel);  
 321: dialog.pack();  
 322: dialog.setLocationRelativeTo(janela);  
 323: dialog.setVisible(true);  
 324:  
 325: return mySwingWorker.get();  
 326: }  
 327:  
 328: public int getCounter() {  
 329: return counter;  
 330: }  
 331:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\database\myDB.java

1:package br.ufrj.coppetec.concentrador.database;  
 2:  
 3:import java.sql.ResultSet;  
 4:import java.text.ParseException;  
 5:import java.util.Date;  
 6:import java.util.HashMap;  
 7:import java.util.Map;  
 8:import java.util.Vector;  
 9:  
 10:import org.apache.logging.log4j.LogManager;  
 11:import org.apache.logging.log4j.Logger;  
 12:import org.json.JSONObject;  
 13:  
 14:import br.ufrj.coppetec.concentrador.Concentrador;  
 15:import br.ufrj.coppetec.concentrador.Util;  
 16:import java.sql.SQLException;  
 17:  
 18:/\*\*  
 19: \*  
 20: \* @author mangeli  
 21: \*/  
 22:public class myDB extends Db {  
 23:  
 24: public static final String TABLE\_IMPORTED\_FILES = "importedFiles";  
 25: public static final String TABLE\_IMPORTED\_FILES\_ID = "id";  
 26: public static final String TABLE\_IMPORTED\_FILES\_PATH = "path";  
 27: public static final String TABLE\_IMPORTED\_FILES\_DATE = "DATE";  
 28:  
 29: public static final String TABLE\_NAME\_OD = "odtable";  
 30: public static final String TABLE\_NAME\_VOL = "voltable";  
 31:  
 32: private static Logger logger = LogManager.getLogger(myDB.class);  
 33:  
 34: private static myDB instance = null;  
 35:  
 36: private myDB() throws Exception {  
 37: super("org.sqlite.JDBC", "jdbc:sqlite:dados.db");  
 38:  
 39: }  
 40:  
 41: public static myDB getInstance() throws Exception {  
 42: if (instance == null) {  
 43: instance = new myDB();  
 44: }  
 45: return instance;  
 46: }  
 47:  
 48: public static String getConditionByValidDate(String tableName) throws ParseException {  
 49: return String.format("%s %s IN %s", (Concentrador.treinamento ? "NOT" : ""),  
 50: (tableName.equalsIgnoreCase(TABLE\_NAME\_OD) ? "DATE(dataIniPesq)" : "data"), Util.getValidDatesListInSQL());  
 51: }  
 52:  
 53: public void keepAlive() throws Exception {  
 54: openTransaction();  
 55: this.executeQuery("Select 1;");  
 56: commit();  
 57: }  
 58:  
 59: public void sanitize() throws Exception {  
 60: try {  
 61: openTransaction();  
 62: this.executeStatement("DELETE FROM odtable WHERE id is null OR id = 'null';");  
 63: commit();  
 64: } catch (Exception e) {  
 65: rollback();  
 66: throw e;  
 67: }  
 68: }  
 69:  
 70: public int verifyPV(PVKey regKey) throws Exception {  
 71: int r = 0;  
 72: try {  
 73: this.openTransaction();  
 74: this.setStatement();  
 75: String qry = "SELECT \* FROM voltable WHERE posto = " + Integer.toString(regKey.posto);  
 76: // qry += " AND pista='" + reg.pista + "'";  
 77: qry += " AND data='" + regKey.data + "'";  
 78: qry += " AND hora=" + Integer.toString(regKey.hora);  
 79: qry += " AND sentido='" + regKey.sentido + "'";  
 80: // qry += " AND posto='" + Integer.toString(reg.posto) + "'";  
 81: // qry += " AND pesquisador1='" + reg.pesquisador1 + "'";  
 82: // qry += " AND pesquisador2='" + reg.pesquisador2 + "'";  
 83:  
 84: ResultSet result = this.executeQuery(qry);  
 85: r = ((result.next()) ? result.getInt("id") : 0);  
 86: result.close();  
 87: this.commit();  
 88: } catch (Exception e) {  
 89: rollback();  
 90: throw e;  
 91: }  
 92: return r;  
 93: }  
 94:  
 95: public int verifyPV(PVregister reg) throws Exception {  
 96: return verifyPV(new PVKey(reg));  
 97: }  
 98:  
 99: public PVregister getPVRegister(PVKey key) throws Exception {  
 100: return getPVRegister(verifyPV(key));  
 101: }  
 102:  
 103: public PVregister getPVRegister(int id) throws Exception {  
 104: PVregister pvR = new PVregister();  
 105: String qry = "SELECT \* FROM voltable WHERE id = " + "'" + id + "'";  
 106:  
 107: this.setStatement();  
 108: ResultSet result = this.executeQuery(qry);  
 109: if (result.next()) {  
 110: pvR.posto = result.getInt("posto");  
 111: pvR.pista = result.getString("pista");  
 112: pvR.data = result.getString("data");  
 113: pvR.hora = result.getInt("hora");  
 114: pvR.sentido = result.getString("sentido");  
 115: pvR.local = result.getString("local");  
 116: pvR.pesquisador1 = result.getString("pesquisador1");  
 117: pvR.pesquisador2 = result.getString("pesquisador2");  
 118: pvR.p1 = result.getString("P1");  
 119: pvR.p2 = result.getString("P2");  
 120: pvR.p3 = result.getString("P3");  
 121: pvR.m = result.getString("M");  
 122: pvR.o1 = result.getString("O1");  
 123: pvR.o2 = result.getString("O2");  
 124: pvR.o3 = result.getString("O3");  
 125: pvR.c1 = result.getString("C1");  
 126: pvR.c2 = result.getString("C2");  
 127: pvR.c3 = result.getString("C3");  
 128: pvR.c4 = result.getString("C4");  
 129: pvR.c5 = result.getString("C5");  
 130: pvR.s1 = result.getString("S1");  
 131: pvR.s2 = result.getString("S2");  
 132: pvR.s3 = result.getString("S3");  
 133: pvR.s4 = result.getString("S4");  
 134: pvR.s5 = result.getString("S5");  
 135: pvR.s6 = result.getString("S6");  
 136: pvR.se1 = result.getString("SE1");  
 137: pvR.se2 = result.getString("SE2");  
 138: pvR.se3 = result.getString("SE3");  
 139: pvR.se4 = result.getString("SE4");  
 140: pvR.se5 = result.getString("SE5");  
 141: pvR.r1 = result.getString("R1");  
 142: pvR.r2 = result.getString("R2");  
 143: pvR.r3 = result.getString("R3");  
 144: pvR.r4 = result.getString("R4");  
 145: pvR.r5 = result.getString("R5");  
 146: pvR.r6 = result.getString("R6");  
 147: }  
 148: result.close();  
 149: return pvR;  
 150: }  
 151:  
 152: public void deletePV(PVKey key) throws Exception {  
 153: try {  
 154: openTransaction();  
 155: this.setStatement();  
 156: String qry = "DELETE FROM voltable WHERE posto = " + Integer.toString(key.posto);  
 157: qry += " AND data='" + key.data + "'";  
 158: qry += " AND hora=" + Integer.toString(key.hora);  
 159: qry += " AND sentido='" + key.sentido + "'";  
 160:  
 161: this.executeStatement(qry);  
 162: commit();  
 163: } catch (Exception e) {  
 164: rollback();  
 165: throw e;  
 166: }  
 167: }  
 168:  
 169: private String[] getDates(String table, String nameFieldDate, String postoField, Integer posto, String extra\_condition)  
 170: throws Exception {  
 171: String validPeriodCondition = " AND " + getConditionByValidDate(table) + " ";  
 172: this.setStatement();  
 173: ResultSet result = null;  
 174: String[] datesReturn = null;  
 175: int qtd = 0;  
 176: String qry = "SELECT COUNT(DISTINCT DATE(" + nameFieldDate + ")) AS qtd FROM " + table + " WHERE " + postoField + "="  
 177: + posto + " " + validPeriodCondition + extra\_condition;  
 178: result = this.executeQuery(qry);  
 179: if (result.next()) {  
 180: qtd = result.getInt("qtd");  
 181: }  
 182: result.close();  
 183: if (qtd != 0) {  
 184: datesReturn = new String[qtd];  
 185: this.setStatement();  
 186: qry = "SELECT DISTINCT DATE(" + nameFieldDate + ") AS d FROM " + table + " WHERE " + postoField + "=" + posto + " "  
 187: + validPeriodCondition + extra\_condition + " ORDER BY d DESC";  
 188: result = this.executeQuery(qry);  
 189: int count = 0;  
 190: while (result.next()) {  
 191: datesReturn[count++] = result.getString("d");  
 192: }  
 193: }  
 194: return datesReturn;  
 195: }  
 196:  
 197: public String[] getODDates(Integer posto) throws Exception {  
 198: return getDates("odTable", "dataIniPesq", "idPosto", posto, " AND cancelado=0");  
 199: }  
 200:  
 201: public String[] getVolDates(Integer posto) throws Exception {  
 202: return getDates("voltable", "data", "posto", posto, "");  
 203: }  
 204:  
 205: public Map<String, Integer> getSumVol(String[] fieldNames, String date) throws Exception {  
 206: Map<String, Integer> returnData = new HashMap<String, Integer>();  
 207: try {  
 208: openTransaction();  
 209: this.setStatement();  
 210: String qry;  
 211: JSONObject jsonObject = null;  
 212: for (int i = 0; i < fieldNames.length; i++) {  
 213: returnData.put(fieldNames[i], 0);  
 214: }  
 215: if (date == null) {  
 216: qry = "SELECT \* FROM voltable WHERE " + getConditionByValidDate(TABLE\_NAME\_VOL);  
 217: } else {  
 218: qry = "SELECT \* FROM voltable WHERE data = '" + date + "'";  
 219: }  
 220:  
 221: ResultSet result = this.executeQuery(qry);  
 222: while (result.next()) {  
 223: for (String field : returnData.keySet()) {  
 224: jsonObject = new JSONObject(result.getString(field));  
 225: returnData.put(field, returnData.get(field) + jsonObject.getInt("Hora1") + jsonObject.getInt("Hora2"));  
 226: }  
 227: }  
 228: result.close();  
 229: commit();  
 230: } catch (Exception e) {  
 231: rollback();  
 232: throw e;  
 233: }  
 234: return returnData;  
 235:  
 236: }  
 237:  
 238: public Map<String, Map<Integer, Integer>> getSumVolPerHour(String[] fieldNames, String date) throws Exception {  
 239: Map<String, Map<Integer, Integer>> returnData = new HashMap<String, Map<Integer, Integer>>();  
 240:  
 241: try {  
 242: openTransaction();  
 243: this.setStatement();  
 244: String qry;  
 245: JSONObject jsonObject = null;  
 246: for (int i = 0; i < fieldNames.length; i++) {  
 247: returnData.put(fieldNames[i], new HashMap<Integer, Integer>());  
 248: for (int j = 0; j < 24; j += 2) {  
 249: returnData.get(fieldNames[i]).put(j, 0);  
 250: }  
 251: }  
 252: if (date == null) {  
 253: qry = "SELECT \* FROM voltable WHERE " + getConditionByValidDate(TABLE\_NAME\_VOL);  
 254: } else {  
 255: qry = "SELECT \* FROM voltable WHERE data = '" + date + "'";  
 256: }  
 257: ResultSet result = this.executeQuery(qry);  
 258: while (result.next()) {  
 259: for (String field : returnData.keySet()) {  
 260: jsonObject = new JSONObject(result.getString(field));  
 261: returnData.get(field).put(result.getInt("hora"), returnData.get(field).get(result.getInt("hora"))  
 262: + jsonObject.getInt("Hora1") + jsonObject.getInt("Hora2"));  
 263: }  
 264: }  
 265: result.close();  
 266: commit();  
 267: } catch (Exception e) {  
 268: rollback();  
 269: throw e;  
 270: }  
 271: return returnData;  
 272: }  
 273:  
 274: public void updatePV(PVregister reg, int id) throws Exception {  
 275: try {  
 276: openTransaction();  
 277: String sql = "UPDATE voltable SET ";  
 278: sql += "enviado=" + Integer.toString(reg.enviado) + ",";  
 279: sql += "posto=" + Integer.toString(reg.posto) + ",";  
 280: sql += "pista=" + "'" + reg.pista + "', ";  
 281: sql += "data=" + "'" + reg.data + "',";  
 282: sql += "hora=" + Integer.toString(reg.hora) + ", ";  
 283: sql += "sentido=" + "" + "'" + reg.sentido + "', ";  
 284: sql += "local=" + "'" + reg.local + "', ";  
 285: sql += "pesquisador1=" + "'" + reg.pesquisador1 + "', ";  
 286: sql += "pesquisador2=" + "'" + reg.pesquisador2 + "', ";  
 287: sql += "P1=" + "'" + reg.p1 + "', ";  
 288: sql += "P2=" + "'" + reg.p2 + "', ";  
 289: sql += "P3=" + "'" + reg.p3 + "', ";  
 290: sql += "M=" + "'" + reg.m + "', ";  
 291: sql += "O1=" + "'" + reg.o1 + "', ";  
 292: sql += "O2=" + "'" + reg.o2 + "',";  
 293: sql += "O3=" + "'" + reg.o3 + "', ";  
 294: sql += "C1=" + "'" + reg.c1 + "', ";  
 295: sql += "C2=" + "'" + reg.c2 + "', ";  
 296: sql += "C3=" + "'" + reg.c3 + "', ";  
 297: sql += "C4=" + "'" + reg.c4 + "', ";  
 298: sql += "C5=" + "'" + reg.c5 + "', ";  
 299: sql += "S1=" + "'" + reg.s1 + "', ";  
 300: sql += "S2=" + "'" + reg.s2 + "', ";  
 301: sql += "S3=" + "'" + reg.s3 + "', ";  
 302: sql += "S4=" + "'" + reg.s4 + "', ";  
 303: sql += "S5=" + "'" + reg.s5 + "', ";  
 304: sql += "S6=" + "'" + reg.s6 + "', ";  
 305: sql += "SE1=" + "'" + reg.se1 + "', ";  
 306: sql += "SE2=" + "'" + reg.se2 + "', ";  
 307: sql += "SE3=" + "'" + reg.se3 + "', ";  
 308: sql += "SE4=" + "'" + reg.se4 + "', ";  
 309: sql += "SE5=" + "'" + reg.se5 + "', ";  
 310: sql += "R1=" + "'" + reg.r1 + "', ";  
 311: sql += "R2=" + "'" + reg.r2 + "', ";  
 312: sql += "R3=" + "'" + reg.r3 + "', ";  
 313: sql += "R4=" + "'" + reg.r4 + "', ";  
 314: sql += "R5=" + "'" + reg.r5 + "', ";  
 315: sql += "R6=" + "'" + reg.r6 + "'";  
 316: sql += " WHERE id=" + "'" + id + "'";  
 317: this.setStatement();  
 318: this.executeStatement(sql);  
 319: commit();  
 320: } catch (Exception e) {  
 321: rollback();  
 322: throw e;  
 323: }  
 324: }  
 325:  
 326: public int inputPV(PVregister reg) throws Exception {  
 327: int r = -1;  
 328: try {  
 329: openTransaction();  
 330: int timeId = (int) (System.currentTimeMillis() / 1000L);  
 331:  
 332: this.setStatement();  
 333: String sql = "INSERT OR IGNORE INTO voltable (";  
 334: sql += "id, enviado, posto, pista, data, hora, sentido, local, pesquisador1, pesquisador2, ";  
 335: sql += "P1, P2, P3, M, O1, O2, O3, C1, C2, C3, C4, C5, ";  
 336: sql += "S1, S2, S3, S4, S5, S6, SE1, SE2, SE3, SE4, SE5, R1, R2, R3, R4, R5, R6";  
 337: sql += ") VALUES (";  
 338: sql += Integer.toString(timeId) + ", ";  
 339: sql += Integer.toString(reg.enviado) + ",";  
 340: sql += Integer.toString(reg.posto) + ",";  
 341: sql += "'" + reg.pista + "', ";  
 342: sql += "'" + reg.data + "',";  
 343: sql += Integer.toString(reg.hora) + ", ";  
 344: sql += "'" + reg.sentido + "', ";  
 345: sql += "'" + reg.local + "', ";  
 346: sql += "'" + reg.pesquisador1 + "', ";  
 347: sql += "'" + reg.pesquisador2 + "', ";  
 348: sql += "'" + reg.p1 + "', ";  
 349: sql += "'" + reg.p2 + "', ";  
 350: sql += "'" + reg.p3 + "', ";  
 351: sql += "'" + reg.m + "', ";  
 352: sql += "'" + reg.o1 + "', ";  
 353: sql += "'" + reg.o2 + "',";  
 354: sql += "'" + reg.o3 + "', ";  
 355: sql += "'" + reg.c1 + "', ";  
 356: sql += "'" + reg.c2 + "', ";  
 357: sql += "'" + reg.c3 + "', ";  
 358: sql += "'" + reg.c4 + "', ";  
 359: sql += "'" + reg.c5 + "', ";  
 360: sql += "'" + reg.s1 + "', ";  
 361: sql += "'" + reg.s2 + "', ";  
 362: sql += "'" + reg.s3 + "', ";  
 363: sql += "'" + reg.s4 + "', ";  
 364: sql += "'" + reg.s5 + "', ";  
 365: sql += "'" + reg.s6 + "', ";  
 366: sql += "'" + reg.se1 + "', ";  
 367: sql += "'" + reg.se2 + "', ";  
 368: sql += "'" + reg.se3 + "', ";  
 369: sql += "'" + reg.se4 + "', ";  
 370: sql += "'" + reg.se5 + "', ";  
 371: sql += "'" + reg.r1 + "', ";  
 372: sql += "'" + reg.r2 + "', ";  
 373: sql += "'" + reg.r3 + "', ";  
 374: sql += "'" + reg.r4 + "', ";  
 375: sql += "'" + reg.r5 + "', ";  
 376: sql += "'" + reg.r6 + "') ";  
 377: r = this.executeStatement(sql);  
 378: commit();  
 379: } catch (Exception e) {  
 380: rollback();  
 381: throw e;  
 382: }  
 383: return r;  
 384: }  
 385:  
 386: public void initDatabaseTables() throws Exception {  
 387: boolean newDB;  
 388: try {  
 389: openTransaction();  
 390: newDB = isClean();  
 391: createVolTable();  
 392: createODTable();  
 393: createImportedFilesTable();  
 394: updateSchema(newDB);  
 395: commit();  
 396: } catch (Exception e) {  
 397: rollback();  
 398: throw e;  
 399: }  
 400: }  
 401:   
 402: /\*\*  
 403: \* Verifica se a tabale odTable já existe no banco  
 404: \* @return true se o banco não possuía a tabela odTable (banco recém criado)  
 405: \* @throws Exception   
 406: \*/  
 407: private boolean isClean() throws Exception{  
 408: boolean r = true;  
 409: this.setStatement();  
 410: ResultSet result = this.executeQuery("SELECT \* FROM SQLITE\_MASTER WHERE TYPE='table' AND name='odTable';");  
 411: if (result.next()){  
 412: r = false;  
 413: }  
 414: return r;  
 415: }  
 416:  
 417: private void createVolTable() throws Exception {  
 418: this.setStatement();  
 419: String qry = "CREATE TABLE IF NOT EXISTS voltable ";  
 420: qry += " (id int PRIMARY KEY, ";  
 421: qry += "enviado integer, ";  
 422: qry += "posto integer, ";  
 423: qry += "pista text, ";  
 424: qry += "data text, ";  
 425: qry += "hora integer, ";  
 426: qry += "sentido text, ";  
 427: qry += "local text, ";  
 428: qry += "pesquisador1 text, ";  
 429: qry += "pesquisador2 text, ";  
 430: qry += "P1 text, ";  
 431: qry += "P2 text, ";  
 432: qry += "P3 text, ";  
 433: qry += "M text, ";  
 434: qry += "O1 text, ";  
 435: qry += "O2 text, ";  
 436: qry += "O3 text, ";  
 437: qry += "C1 text, ";  
 438: qry += "C2 text, ";  
 439: qry += "C3 text, ";  
 440: qry += "C4 text, ";  
 441: qry += "C5 text, ";  
 442: qry += "R1 text, ";  
 443: qry += "R2 text, ";  
 444: qry += "R3 text, ";  
 445: qry += "R4 text, ";  
 446: qry += "R5 text, ";  
 447: qry += "R6 text, ";  
 448: qry += "S1 text, ";  
 449: qry += "S2 text, ";  
 450: qry += "S3 text, ";  
 451: qry += "S4 text, ";  
 452: qry += "S5 text, ";  
 453: qry += "S6 text, ";  
 454: qry += "SE1 text, ";  
 455: qry += "SE2 text, ";  
 456: qry += "SE3 text, ";  
 457: qry += "SE4 text, ";  
 458: qry += "SE5 text ";  
 459: qry += "); ";  
 460: this.executeStatement(qry);  
 461:  
 462: }  
 463:  
 464: private void createImportedFilesTable() throws Exception {  
 465: this.setStatement();  
 466: String qry = "CREATE TABLE IF NOT EXISTS " + TABLE\_IMPORTED\_FILES + " (";  
 467: qry += TABLE\_IMPORTED\_FILES\_ID + " text primary key, ";  
 468: qry += TABLE\_IMPORTED\_FILES\_PATH + " text unique,";  
 469: qry += TABLE\_IMPORTED\_FILES\_DATE + " datetime DEFAULT CURRENT\_TIMESTAMP";  
 470: qry += "); ";  
 471: this.executeStatement(qry);  
 472: }  
 473:  
 474: private void createODTable() throws Exception {  
 475: this.setStatement();  
 476: String qry = "CREATE TABLE IF NOT EXISTS odTable (";  
 477: qry += "id text primary key NOT NULL, ";  
 478: qry += "enviado integer, ";  
 479: qry += "estaNoNote integer, ";  
 480: qry += "cancelado integer, ";  
 481: qry += "idPosto integer, ";  
 482: qry += "sentido text, ";  
 483: qry += "idIpad text, ";  
 484: // qry += "uuid text, ";  
 485: qry += "login text, ";  
 486: qry += "dataIniPesq text, ";  
 487: qry += "dataFimPesq text, ";  
 488: qry += "placa text, ";  
 489: qry += "anoDeFabricacao integer, ";  
 490: qry += "tipo text, ";  
 491: qry += "idOrigemPais integer, ";  
 492: qry += "idOrigemMunicipio integer, ";  
 493: qry += "idDestinoPais integer, ";  
 494: qry += "idDestinoMunicipio integer, ";  
 495: qry += "idMotivoDeEscolhaDaRota integer, ";  
 496: qry += "frequenciaQtd integer, ";  
 497: qry += "frequenciaPeriodo text, ";  
 498: qry += "idPropriedadesDoVeiculo integer, ";  
 499: qry += "placaEstrangeira integer, ";  
 500: qry += "idPaisPlacaEstrangeira integer, ";  
 501: qry += "idCombustivel integer, ";  
 502: qry += "categoria text, ";  
 503: qry += "possuiReboque integer, ";  
 504: qry += "numeroDePessoasNoVeiculo integer, ";  
 505: qry += "numeroDePessoasATrabalho integer, ";  
 506: qry += "idRendaMedia integer, ";  
 507: qry += "idMotivoDaViagem integer, ";  
 508: qry += "tipoCaminhao text, ";  
 509: qry += "idTipoDeContainer integer, ";  
 510: qry += "idTipoCarroceria integer, ";  
 511: qry += "rntrc text, ";  
 512: qry += "possuiCargaPerigosa integer, ";  
 513: qry += "idNumeroDeRisco integer, ";  
 514: qry += "idNumeroDaOnu integer, ";  
 515: qry += "idAgregado integer, ";  
 516: qry += "placaVermelha integer, ";  
 517: qry += "idTipoDeViagemOuServico integer, ";  
 518: qry += "pesoDaCarga real, ";  
 519: qry += "valorDoFrete real, ";  
 520: qry += "utilizaParadaEspecial integer, ";  
 521: qry += "idProduto integer, ";  
 522: qry += "idCargaAnterior integer, ";  
 523: qry += "valorDaCarga real, ";  
 524: qry += "municipioEmbarqueCarga integer, ";  
 525: qry += "idLocalEmbarqueCarga integer, ";  
 526: qry += "municipioDesembarqueCarga integer, ";  
 527: qry += "idLocalDesembarqueCarga integer, ";  
 528: qry += "indoPegarCarga integer, ";  
 529: qry += "paradaObrigatoriaMunicipio1 integer, ";  
 530: qry += "paradaObrigatoriaMunicipio2 integer, ";  
 531: qry += "idPerguntaExtra integer,";  
 532: qry += "duracaoPesq integer, ";  
 533: qry += "treinamento integer, ";  
 534: qry += "idPerguntaExtra2 integer";  
 535: // IMPORTANTE: se adicionar mais campos ao final, não esquecer do separador no campo anterior  
 536: // mudanças no banco devem ser refletidas com método updateSchemaScript()  
 537: qry += "); ";  
 538: this.executeStatement(qry);  
 539: this.sanitize();  
 540:  
 541: }  
 542:   
 543: private void updateSchema(boolean newDB) throws Exception{  
 544: int schemaVersion = 0;  
 545: ResultSet schemaResult;  
 546: this.setStatement();  
 547: this.executeStatement("CREATE TABLE IF NOT EXISTS versaoSchema (versao integer);");  
 548: schemaResult = this.executeQuery("SELECT versao FROM versaoSchema;");  
 549: if (schemaResult.next()){  
 550: schemaVersion = schemaResult.getInt("versao");  
 551: }  
 552:   
 553: logger.info("Versão do schema: " + schemaVersion);  
 554: if (schemaVersion != Concentrador.dbVersion){  
 555: logger.info("Atualizando schema do banco de dados");  
 556: this.updateSchemaScript(schemaVersion, newDB);  
 557: }  
 558:   
 559: }  
 560:  
 561: private void updateSchemaScript(int oldVersion, boolean newDB) throws Exception{  
 562: this.setStatement();  
 563:   
 564: if(!newDB){  
 565: if (oldVersion < 1){  
 566: //rodando atualização  
 567: this.executeStatement("ALTER TABLE odTable ADD COLUMN treinamento integer;");  
 568: }  
 569: if (oldVersion < 2){  
 570: this.executeStatement("ALTER TABLE odTable ADD COLUMN idPerguntaExtra2 integer;");  
 571: }  
 572: }  
 573:   
 574:   
 575: //atualizando a tabela de versão  
 576: if(oldVersion == 0){ //a tabela de versão foi recem criada e não tem um registro  
 577: this.executeStatement("INSERT INTO versaoSchema VALUES (" + Concentrador.dbVersion + ");");  
 578: } else {  
 579: this.executeStatement("UPDATE versaoSchema SET versao = " + Concentrador.dbVersion + ";");  
 580: }  
 581:   
 582:   
 583:   
 584: logger.info("Versão do schema: " + Concentrador.dbVersion);  
 585: }  
 586:   
 587: public Vector<String> fetchReportODColumns() throws Exception {  
 588: Vector<String> cols = new Vector<String>();  
 589: openTransaction();  
 590: String sel\_sql = String.format("SELECT DISTINCT idIpad FROM odTable WHERE %s ORDER BY idIpad ASC",  
 591: getConditionByValidDate(TABLE\_NAME\_OD));  
 592: ResultSet result = this.executeQuery(sel\_sql);  
 593: cols.add("");  
 594: while (result.next())  
 595: cols.add(result.getString("idIpad"));  
 596: commit();  
 597: return cols;  
 598: }  
 599:  
 600: public Vector<String> fetchReportODRows(Integer posto) throws Exception {  
 601: Vector<String> rows = new Vector<String>();  
 602: openTransaction();  
 603: String validPeriodCondition = " AND " + getConditionByValidDate(TABLE\_NAME\_OD);  
 604: String sel\_sql = "SELECT DISTINCT DATE(dataIniPesq) AS data FROM odTable WHERE idPosto=" + posto + " AND cancelado=0 "  
 605: + validPeriodCondition + " ORDER BY date(dataIniPesq) DESC";  
 606: ResultSet result = this.executeQuery(sel\_sql);  
 607: while (result.next()) {  
 608: Date day = Util.sdfSQL.parse(result.getString("data"));  
 609: rows.add(Util.sdfBrazil.format(day));  
 610: }  
 611: commit();  
 612: return rows;  
 613: }  
 614:  
 615: public Map<String, Integer> fetchReportODData(String strData, Integer posto) throws Exception {  
 616: Date data = Util.sdfBrazil.parse(strData);  
 617: openTransaction();  
 618: String sqlData = Util.sdfSQL.format(data);  
 619: String sel\_sql = "SELECT idIpad, COUNT(idIpad) AS times FROM odTable " + " WHERE DATE(dataIniPesq)='" + sqlData  
 620: + "' AND cancelado=0 AND idPosto=" + posto + " GROUP BY idIpad ";  
 621:  
 622: HashMap<String, Integer> map = new HashMap<String, Integer>();  
 623: ResultSet result = this.executeQuery(sel\_sql);  
 624: while (result.next()) {  
 625: String ipad = result.getString("idIpad");  
 626: Integer count = result.getInt("times");  
 627: map.put(ipad, count);  
 628: }  
 629: commit();  
 630: return map;  
 631: }  
 632:  
 633: public Map<String, Map<String, Integer>> fetchReportODData(Integer posto) throws Exception {  
 634: openTransaction();  
 635: String validPeriodCondition = " AND " + getConditionByValidDate(TABLE\_NAME\_OD);  
 636: String sel\_sql = "SELECT DATE(dataIniPesq) AS dia, idIpad, COUNT(idIpad) AS times FROM odTable "  
 637: + "WHERE cancelado=0 AND idPosto=" + posto + validPeriodCondition + " GROUP BY date(dataIniPesq), idIpad "  
 638: + "ORDER BY date(dataIniPesq) ASC, idIpad ASC";  
 639: HashMap<String, Map<String, Integer>> data = new HashMap<String, Map<String, Integer>>();  
 640: ResultSet result = this.executeQuery(sel\_sql);  
 641: while (result.next()) {  
 642: Date day = Util.sdfSQL.parse(result.getString("dia"));  
 643: String ipad = result.getString("idIpad");  
 644: Integer times = result.getInt("times");  
 645:  
 646: logger.debug(day.getTime() + ":" + ipad + "(" + times + ")");  
 647: if (data.containsKey(day)) {  
 648: Map<String, Integer> regs = data.get(day);  
 649: regs.put(ipad, times);  
 650: data.put(Util.sdfBrazil.format(day), regs);  
 651: } else {  
 652: HashMap<String, Integer> regs = new HashMap<String, Integer>();  
 653: regs.put(ipad, times);  
 654: data.put(Util.sdfBrazil.format(day), regs);  
 655: }  
 656: }  
 657: result.close();  
 658: commit();  
 659: return data;  
 660: }  
 661:  
 662: // public final static BigInteger MAX\_SAFE\_INT = BigInteger.valueOf(99999999);  
 663: //  
 664: // /\*\*  
 665: // \* Retorna o valor inteiro, limitando ao valor máximo permitido (99999999).  
 666: // \*  
 667: // \* @param valueStr  
 668: // \* - O valor em uma string.  
 669: // \* @return O valor em inteiro.  
 670: // \*/  
 671: // public static Integer getIntValue(String valueStr) {  
 672: // Integer valueInt = null;  
 673: // if (valueStr != null) {  
 674: // BigInteger value = new BigInteger(valueStr);  
 675: // if (value.compareTo(MAX\_SAFE\_INT) > 0) {  
 676: // value = MAX\_SAFE\_INT;  
 677: // logger.warn(String.format("Valor inteiro maior que o permitido (%s > %d)", valueStr, MAX\_SAFE\_INT.intValue()));  
 678: // }  
 679: // valueInt = value.intValue();  
 680: // }  
 681: // return valueInt;  
 682: // }  
 683:  
 684: // public static String getStrFromIntValue(String valueStr) {  
 685: // Integer valueInt = getIntValue(valueStr);  
 686: // return (valueInt == null ? null : String.valueOf(valueInt.intValue()));  
 687: // }  
 688:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\database\PVKey.java

1:package br.ufrj.coppetec.concentrador.database;  
 2:  
 3:/\*\*  
 4: \*  
 5: \* @author mangeli  
 6: \*/  
 7:public class PVKey {  
 8: public int posto;  
 9: public String data;  
 10: public int hora;  
 11: public String sentido;  
 12:   
 13: public PVKey(){  
 14: }  
 15:   
 16: public PVKey(PVregister reg){  
 17: this.posto = reg.posto;  
 18: this.data = reg.data;  
 19: this.hora = reg.hora;  
 20: this.sentido = reg.sentido;  
 21: }  
 22:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\database\PVregister.java

1:package br.ufrj.coppetec.concentrador.database;  
 2:  
 3:public class PVregister {  
 4: public int enviado;  
 5: public int posto;  
 6: public String pista;  
 7: public String data;  
 8: public int hora;  
 9: public String sentido;  
 10: public String local;  
 11: public String pesquisador1;  
 12: public String pesquisador2;  
 13: public String p1;  
 14: public String p2;  
 15: public String p3;  
 16: public String m;  
 17: public String o1;  
 18: public String o2;  
 19: public String o3;  
 20: public String c1;  
 21: public String c2;  
 22: public String c3;  
 23: public String c4;  
 24: public String c5;  
 25: public String s1;  
 26: public String s2;  
 27: public String s3;  
 28: public String s4;  
 29: public String s5;  
 30: public String s6;  
 31: public String se1;  
 32: public String se2;  
 33: public String se3;  
 34: public String se4;  
 35: public String se5;  
 36: public String r1;  
 37: public String r2;  
 38: public String r3;  
 39: public String r4;  
 40: public String r5;  
 41: public String r6;  
 42:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\exporter\JSONBuilder.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador.exporter;  
 7:  
 8:import java.sql.ResultSet;  
 9:  
 10:/\*\*  
 11: \*  
 12: \* @author mangeli  
 13: \*/  
 14:public interface JSONBuilder {  
 15: public String build(ResultSet result) throws Exception;  
 16:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\exporter\JSONExporter.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador.exporter;  
 7:  
 8:import java.awt.BorderLayout;  
 9:import java.awt.Dialog;  
 10:import java.beans.PropertyChangeEvent;  
 11:import java.beans.PropertyChangeListener;  
 12:import java.io.File;  
 13:import java.io.FileWriter;  
 14:import java.sql.ResultSet;  
 15:import java.text.ParseException;  
 16:  
 17:import javax.swing.JDialog;  
 18:import javax.swing.JLabel;  
 19:import javax.swing.JOptionPane;  
 20:import javax.swing.JPanel;  
 21:import javax.swing.JProgressBar;  
 22:import javax.swing.SwingWorker;  
 23:  
 24:import org.apache.logging.log4j.LogManager;  
 25:import org.apache.logging.log4j.Logger;  
 26:import org.json.JSONException;  
 27:import org.json.JSONObject;  
 28:  
 29:import br.ufrj.coppetec.concentrador.Janela;  
 30:import static br.ufrj.coppetec.concentrador.Util.TODAS;  
 31:import br.ufrj.coppetec.concentrador.database.myDB;  
 32:  
 33:/\*\*  
 34: \*  
 35: \* @author mangeli  
 36: \*/  
 37:public class JSONExporter {  
 38:  
 39: private static Logger logger = LogManager.getLogger(JSONExporter.class);  
 40:  
 41: private File file;  
 42: private File tmpFile;  
 43: private DbTable table;  
 44: private Janela janela;  
 45:  
 46: static public enum WhatToExport {  
 47: ALL, NOT\_SENT, DATE  
 48: }  
 49:  
 50: private JSONBuilder jsonBuilder;  
 51:  
 52: static public enum DbTable {  
 53: PV("voltable"), OD("odtable");  
 54: private final String text;  
 55:  
 56: DbTable(final String s) {  
 57: this.text = s;  
 58: }  
 59:  
 60: public String toString() {  
 61: return text;  
 62: }  
 63:  
 64: public String sql(Integer posto) throws ParseException {  
 65: String r = "SELECT \* FROM ";  
 66: if (text.equals(myDB.TABLE\_NAME\_OD)) {  
 67: r += "odtable WHERE cancelado=0 and idPosto=" + posto.intValue();  
 68: } else {  
 69: r += "voltable WHERE posto=" + posto.intValue();  
 70: }  
 71: r += " AND " + myDB.getConditionByValidDate(text);  
 72: return r;  
 73: }  
 74:  
 75: }  
 76:  
 77: public JSONExporter(File f, DbTable t, Janela janela) {  
 78: this.file = f;  
 79: this.table = t;  
 80: this.janela = janela;  
 81: this.tmpFile = new File("temp.json");  
 82: switch (t) {  
 83: case PV:  
 84: jsonBuilder = new PvJSONbuilder();  
 85: break;  
 86: case OD:  
 87: jsonBuilder = new OdJSONbuilder();  
 88: break;  
 89: }  
 90: }  
 91:  
 92: public void export(Integer posto) {  
 93: export(posto, null);  
 94: }  
 95:  
 96: public void export(Integer posto, String date) {  
 97:  
 98: SwingWorker<Boolean, Void> mySwingWorker = new SwingWorker<Boolean, Void>() {  
 99:  
 100: @Override  
 101: protected Boolean doInBackground() throws Exception {  
 102: return JSONExporter.this.backgroundExport(posto, date);  
 103: }  
 104:  
 105: };  
 106:  
 107: final JDialog dialog = new JDialog(this.janela, "Dialog", Dialog.ModalityType.APPLICATION\_MODAL);  
 108:  
 109: mySwingWorker.addPropertyChangeListener(new PropertyChangeListener() {  
 110:  
 111: @Override  
 112: public void propertyChange(PropertyChangeEvent evt) {  
 113: if (evt.getPropertyName().equals("state")) {  
 114: if (evt.getNewValue() == SwingWorker.StateValue.DONE) {  
 115: dialog.dispose();  
 116: try {  
 117: if (mySwingWorker.get() == true) {  
 118: String msgSucesso = "Arquivo " + JSONExporter.this.file.getName() + " exportado com sucesso.";  
 119: JOptionPane.showMessageDialog(janela, msgSucesso,  
 120: "Exportação de dados.", JOptionPane.INFORMATION\_MESSAGE);  
 121: logger.info (msgSucesso);  
 122: }  
 123: } catch (Exception e) {  
 124: logger.error("Error exportando dados: ", e);  
 125: }  
 126:  
 127: }  
 128: }  
 129: }  
 130:  
 131: });  
 132:  
 133: mySwingWorker.execute();  
 134:  
 135: JProgressBar progressBar = new JProgressBar();  
 136: progressBar.setIndeterminate(true);  
 137: JPanel panel = new JPanel(new BorderLayout());  
 138: panel.add(progressBar, BorderLayout.CENTER);  
 139: panel.add(new JLabel("Aguarde a exportação dos dados......."), BorderLayout.PAGE\_START);  
 140: dialog.add(panel);  
 141: dialog.pack();  
 142: dialog.setLocationRelativeTo(this.janela);  
 143: dialog.setVisible(true);  
 144:  
 145: }  
 146:  
 147: private Boolean backgroundExport(Integer posto) {  
 148: return backgroundExport(posto, null);  
 149: }  
 150:  
 151: private Boolean backgroundExport(Integer posto, String date) {  
 152: myDB database = null;  
 153: ResultSet result;  
 154: FileWriter writer;  
 155: try {  
 156: String qry = this.table.sql(posto);  
 157:  
 158: if (date != null) {  
 159: if (date.equals(TODAS)){  
 160: qry += " AND " + myDB.getConditionByValidDate(this.table.toString()) + ";";  
 161: } else {  
 162: if (this.table.text.equals(myDB.TABLE\_NAME\_OD)) {  
 163: qry += " AND dataIniPesq like '" + date + "%';";  
 164: } else {  
 165: qry += " AND data like '" + date + "%';";  
 166: }  
 167: }  
 168:   
 169: }  
 170:  
 171: database = myDB.getInstance();  
 172: database.openTransaction();  
 173: database.setStatement();  
 174: result = database.executeQuery(qry);  
 175:  
 176: // writer = new FileWriter(file);  
 177: writer = new FileWriter(tmpFile);  
 178: writer.write(jsonBuilder.build(result));  
 179: result.close();  
 180: writer.close();  
 181: Zipper zipper = new Zipper(tmpFile, file);  
 182: zipper.zipIt();  
 183: database.executeStatement("UPDATE " + this.table.toString() + " SET enviado=1 WHERE enviado=0;");  
 184: database.commit();  
 185: tmpFile.delete();  
 186: } catch (JSONException je) {  
 187: logger.warn("Não existem dados para exportação", je);  
 188: // probably empty table  
 189: JOptionPane.showMessageDialog(janela, "Não existem dados para exportação.", "Exportação de dados.",  
 190: JOptionPane.INFORMATION\_MESSAGE);  
 191: if (database != null)  
 192: database.rollback();  
 193: return false;  
 194: } catch (Exception e) {  
 195: logger.error("Erro ao exportar dados.", e);  
 196: JOptionPane.showMessageDialog(janela, "Erro ao exportar dados:\n" + e.getMessage(), "Erro na exportação de dados.",  
 197: JOptionPane.ERROR\_MESSAGE);  
 198: if (database != null)  
 199: database.rollback();  
 200: return false;  
 201: }  
 202:  
 203: return true;  
 204: }  
 205:  
 206: static public Object getJSONInteger(String val) {  
 207: Object r;  
 208: if (val == null) {  
 209: r = JSONObject.NULL;  
 210: } else {  
 211: r = new Integer(val);  
 212: }  
 213:  
 214: return r;  
 215: }  
 216:  
 217: static public Object getJSONDouble(String val) {  
 218: Object r;  
 219: if (val == null) {  
 220: r = JSONObject.NULL;  
 221: } else {  
 222: r = JSONObject.stringToValue(val);  
 223: }  
 224: return r;  
 225: }  
 226:  
 227: static public Object getJSONString(String val) {  
 228: Object r;  
 229: if (val == null || val.equals("null")) {  
 230: r = JSONObject.NULL;  
 231: } else {  
 232: r = val;  
 233: }  
 234:  
 235: return r;  
 236: }  
 237:  
 238:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\exporter\OdJSONbuilder.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador.exporter;  
 7:  
 8:import br.ufrj.coppetec.concentrador.Util;  
 9:import java.sql.ResultSet;  
 10:import java.util.Date;  
 11:  
 12:import javax.swing.JOptionPane;  
 13:  
 14:import org.apache.logging.log4j.LogManager;  
 15:import org.apache.logging.log4j.Logger;  
 16:import org.json.JSONObject;  
 17:  
 18:/\*\*  
 19: \*  
 20: \* @author mangeli  
 21: \*/  
 22:public class OdJSONbuilder implements JSONBuilder {  
 23:  
 24: private static Logger logger = LogManager.getLogger(OdJSONbuilder.class);  
 25:  
 26: @Override  
 27: public String build(ResultSet result) throws Exception {  
 28: String s;  
 29: JSONObject json = new JSONObject();  
 30: JSONObject reg;  
 31: while (result.next()) {  
 32: try {  
 33: reg = new JSONObject();  
 34: // reg.put("id",JSONExporter.getJSONString(result.getString("id"));  
 35: reg.put("estaNoNote", JSONExporter.getJSONInteger(result.getString("estaNoNote")));  
 36: reg.put("cancelado", JSONExporter.getJSONInteger(result.getString("cancelado")));  
 37: reg.put("idPosto", JSONExporter.getJSONInteger(result.getString("idPosto")));  
 38: reg.put("sentido", JSONExporter.getJSONString(result.getString("sentido")));  
 39: reg.put("idIpad", JSONExporter.getJSONString(result.getString("idIpad")));  
 40: // reg.put("login" ,JSONExporter.getJSONString(result.getString("login"));  
 41: reg.put("dataIniPesq", JSONExporter.getJSONString(result.getString("dataIniPesq")));  
 42: reg.put("dataFimPesq", JSONExporter.getJSONString(result.getString("dataFimPesq")));  
 43: reg.put("placa", JSONExporter.getJSONString(result.getString("placa")));  
 44: reg.put("anoDeFabricacao", JSONExporter.getJSONInteger(result.getString("anoDeFabricacao")));  
 45: reg.put("tipo", JSONExporter.getJSONString(result.getString("tipo")));  
 46: reg.put("idOrigemPais", JSONExporter.getJSONInteger(result.getString("idOrigemPais")));  
 47: reg.put("idOrigemMunicipio", JSONExporter.getJSONInteger(result.getString("idOrigemMunicipio")));  
 48: reg.put("idDestinoPais", JSONExporter.getJSONInteger(result.getString("idDestinoPais")));  
 49: reg.put("idDestinoMunicipio", JSONExporter.getJSONInteger(result.getString("idDestinoMunicipio")));  
 50: reg.put("idMotivoDeEscolhaDaRota", JSONExporter.getJSONInteger(result.getString("idMotivoDeEscolhaDaRota")));  
 51: reg.put("frequenciaQtd", JSONExporter.getJSONInteger(result.getString("frequenciaQtd")));  
 52: reg.put("frequenciaPeriodo", JSONExporter.getJSONString(result.getString("frequenciaPeriodo")));  
 53: reg.put("idPropriedadesDoVeiculo", JSONExporter.getJSONInteger(result.getString("idPropriedadesDoVeiculo")));  
 54: reg.put("placaEstrangeira", JSONExporter.getJSONInteger(result.getString("placaEstrangeira")));  
 55: reg.put("idPaisPlacaEstrangeira", JSONExporter.getJSONInteger(result.getString("idPaisPlacaEstrangeira")));  
 56: reg.put("idCombustivel", JSONExporter.getJSONInteger(result.getString("idCombustivel")));  
 57: reg.put("categoria", JSONExporter.getJSONString(result.getString("categoria")));  
 58: reg.put("possuiReboque", JSONExporter.getJSONInteger(result.getString("possuiReboque")));  
 59: reg.put("numeroDePessoasNoVeiculo", JSONExporter.getJSONInteger(result.getString("numeroDePessoasNoVeiculo")));  
 60: reg.put("numeroDePessoasATrabalho", JSONExporter.getJSONInteger(result.getString("numeroDePessoasATrabalho")));  
 61: reg.put("idRendaMedia", JSONExporter.getJSONInteger(result.getString("idRendaMedia")));  
 62: reg.put("idMotivoDaViagem", JSONExporter.getJSONInteger(result.getString("idMotivoDaViagem")));  
 63: reg.put("tipoCaminhao", JSONExporter.getJSONString(result.getString("tipoCaminhao")));  
 64: reg.put("idTipoDeContainer", JSONExporter.getJSONInteger(result.getString("idTipoDeContainer")));  
 65: reg.put("idTipoCarroceria", JSONExporter.getJSONInteger(result.getString("idTipoCarroceria")));  
 66: reg.put("rntrc", JSONExporter.getJSONString(result.getString("rntrc")));  
 67: reg.put("possuiCargaPerigosa", JSONExporter.getJSONInteger(result.getString("possuiCargaPerigosa")));  
 68: reg.put("idNumeroDeRisco", JSONExporter.getJSONInteger(result.getString("idNumeroDeRisco")));  
 69: reg.put("idNumeroDaOnu", JSONExporter.getJSONInteger(result.getString("idNumeroDaOnu")));  
 70: reg.put("idAgregado", JSONExporter.getJSONInteger(result.getString("idAgregado")));  
 71: reg.put("placaVermelha", JSONExporter.getJSONInteger(result.getString("placaVermelha")));  
 72: reg.put("idTipoDeViagemOuServico", JSONExporter.getJSONInteger(result.getString("idTipoDeViagemOuServico")));  
 73: reg.put("pesoDaCarga", JSONExporter.getJSONDouble(result.getString("pesoDaCarga")));  
 74: reg.put("valorDoFrete", JSONExporter.getJSONDouble(result.getString("valorDoFrete")));  
 75: reg.put("utilizaParadaEspecial", JSONExporter.getJSONInteger(result.getString("utilizaParadaEspecial")));  
 76: reg.put("idProduto", JSONExporter.getJSONInteger(result.getString("idProduto")));  
 77: reg.put("idCargaAnterior", JSONExporter.getJSONInteger(result.getString("idCargaAnterior")));  
 78: reg.put("valorDaCarga", JSONExporter.getJSONDouble(result.getString("valorDaCarga")));  
 79: reg.put("municipioEmbarqueCarga", JSONExporter.getJSONInteger(result.getString("municipioEmbarqueCarga")));  
 80: reg.put("idLocalEmbarqueCarga", JSONExporter.getJSONInteger(result.getString("idLocalEmbarqueCarga")));  
 81: reg.put("municipioDesembarqueCarga", JSONExporter.getJSONInteger(result.getString("municipioDesembarqueCarga")));  
 82: reg.put("idLocalDesembarqueCarga", JSONExporter.getJSONInteger(result.getString("idLocalDesembarqueCarga")));  
 83: reg.put("indoPegarCarga", JSONExporter.getJSONInteger(result.getString("indoPegarCarga")));  
 84: reg.put("paradaObrigatoriaMunicipio1",  
 85: JSONExporter.getJSONInteger(result.getString("paradaObrigatoriaMunicipio1")));  
 86: reg.put("paradaObrigatoriaMunicipio2",  
 87: JSONExporter.getJSONInteger(result.getString("paradaObrigatoriaMunicipio2")));  
 88: reg.put("idPerguntaExtra", JSONExporter.getJSONInteger(result.getString("idPerguntaExtra")));  
 89: reg.put("idPerguntaExtra2", JSONExporter.getJSONInteger(result.getString("idPerguntaExtra2")));  
 90: reg.put("dataExportacao", Util.sdfArq.format(new Date()));  
 91: // reg.put("duracaoPesq", JSONExporter.getJSONInteger(result.getString("duracaoPesq")));  
 92:  
 93: json.append("dados", reg);  
 94: } catch (Exception e) {  
 95: logger.error(String.format("Erro ao exportar dados da pesquisa OD no tratamento do registro: %s",  
 96: result.getString("id")), e);  
 97: JOptionPane.showMessageDialog(null, "Erro ao exportar dados:\n" + e.getMessage(), "Erro na exportação de dados.",  
 98: JOptionPane.ERROR\_MESSAGE);  
 99: }  
 100: }  
 101:  
 102: s = json.get("dados").toString().replace("},", "},\n").replace(":[", ":\n[");  
 103: return s;  
 104: }  
 105:  
 106:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\exporter\PvJSONbuilder.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador.exporter;  
 7:  
 8:import java.sql.ResultSet;  
 9:  
 10:import javax.swing.JOptionPane;  
 11:  
 12:import org.apache.logging.log4j.LogManager;  
 13:import org.apache.logging.log4j.Logger;  
 14:import org.json.JSONObject;  
 15:  
 16:/\*\*  
 17: \*  
 18: \* @author mangeli  
 19: \*/  
 20:public class PvJSONbuilder implements JSONBuilder {  
 21:  
 22: private static Logger logger = LogManager.getLogger(PvJSONbuilder.class);  
 23:  
 24: @Override  
 25: public String build(ResultSet result) throws Exception {  
 26: String s;  
 27: JSONObject json = new JSONObject();  
 28: JSONObject reg;  
 29: while (result.next()) {  
 30: try {  
 31: reg = new JSONObject();  
 32: reg.put("Posto", result.getInt("posto"));  
 33: reg.put("Pista", result.getString("pista"));  
 34: reg.put("Data", result.getString("data"));  
 35: reg.put("Hora", result.getString("hora"));  
 36: reg.put("Sentido", result.getString("Sentido"));  
 37: reg.put("Local", result.getString("local"));  
 38: reg.put("Pesquisador1", result.getString("pesquisador1"));  
 39: reg.put("Pesquisador2", result.getString("pesquisador2"));  
 40: reg.append("DadosVolumetrica", new JSONObject(result.getString("P1")));  
 41: reg.append("DadosVolumetrica", new JSONObject(result.getString("P3")));  
 42: reg.append("DadosVolumetrica", new JSONObject(result.getString("P2")));  
 43: reg.append("DadosVolumetrica", new JSONObject(result.getString("M")));  
 44: reg.append("DadosVolumetrica", new JSONObject(result.getString("O1")));  
 45: reg.append("DadosVolumetrica", new JSONObject(result.getString("O2")));  
 46: reg.append("DadosVolumetrica", new JSONObject(result.getString("O3")));  
 47: reg.append("DadosVolumetrica", new JSONObject(result.getString("C1")));  
 48: reg.append("DadosVolumetrica", new JSONObject(result.getString("C2")));  
 49: reg.append("DadosVolumetrica", new JSONObject(result.getString("C3")));  
 50: reg.append("DadosVolumetrica", new JSONObject(result.getString("C4")));  
 51: reg.append("DadosVolumetrica", new JSONObject(result.getString("C5")));  
 52: reg.append("DadosVolumetrica", new JSONObject(result.getString("S1")));  
 53: reg.append("DadosVolumetrica", new JSONObject(result.getString("S2")));  
 54: reg.append("DadosVolumetrica", new JSONObject(result.getString("S3")));  
 55: reg.append("DadosVolumetrica", new JSONObject(result.getString("S4")));  
 56: reg.append("DadosVolumetrica", new JSONObject(result.getString("S5")));  
 57: reg.append("DadosVolumetrica", new JSONObject(result.getString("S6")));  
 58: reg.append("DadosVolumetrica", new JSONObject(result.getString("SE1")));  
 59: reg.append("DadosVolumetrica", new JSONObject(result.getString("SE2")));  
 60: reg.append("DadosVolumetrica", new JSONObject(result.getString("SE3")));  
 61: reg.append("DadosVolumetrica", new JSONObject(result.getString("SE4")));  
 62: reg.append("DadosVolumetrica", new JSONObject(result.getString("SE5")));  
 63: reg.append("DadosVolumetrica", new JSONObject(result.getString("R1")));  
 64: reg.append("DadosVolumetrica", new JSONObject(result.getString("R2")));  
 65: reg.append("DadosVolumetrica", new JSONObject(result.getString("R3")));  
 66: reg.append("DadosVolumetrica", new JSONObject(result.getString("R4")));  
 67: reg.append("DadosVolumetrica", new JSONObject(result.getString("R5")));  
 68: reg.append("DadosVolumetrica", new JSONObject(result.getString("R6")));  
 69:  
 70: json.append("dados", reg);  
 71: } catch (Exception e) {  
 72: logger.error("Erro ao exportar dados da PV.", e);  
 73: JOptionPane.showMessageDialog(null, "Erro ao exportar dados:\n" + e.getMessage(), "Erro na exportação de dados.",  
 74: JOptionPane.ERROR\_MESSAGE);  
 75: }  
 76: }  
 77: s = json.get("dados").toString().replace("},", "},\n").replace(":[", ":\n[");  
 78: return s;  
 79: }  
 80:  
 81:}

## .\concentrador\src\br\ufrj\coppetec\concentrador\exporter\Zipper.java

1:/\*  
 2: \* To change this license header, choose License Headers in Project Properties.  
 3: \* To change this template file, choose Tools | Templates  
 4: \* and open the template in the editor.  
 5: \*/  
 6:package br.ufrj.coppetec.concentrador.exporter;  
 7:  
 8:import java.io.File;  
 9:import java.io.FileInputStream;  
 10:import java.io.FileOutputStream;  
 11:import java.util.regex.Pattern;  
 12:import java.util.zip.ZipEntry;  
 13:import java.util.zip.ZipOutputStream;  
 14:  
 15:/\*\*  
 16: \*  
 17: \* @author mangeli  
 18: \*/  
 19:public class Zipper {  
 20: private File jsonFile;  
 21: private File zipFile;  
 22:  
 23: public Zipper(File f, File z) {  
 24: jsonFile = f;  
 25: zipFile = z;  
 26: }  
 27:  
 28: public void zipIt() throws Exception {  
 29: byte[] buffer = new byte[1024];  
 30: FileOutputStream fos = new FileOutputStream(zipFile);  
 31: ZipOutputStream zos = new ZipOutputStream(fos);  
 32: ZipEntry ze = new ZipEntry(zipFile.getName().split(Pattern.quote("."))[0] + ".json");  
 33: zos.putNextEntry(ze);  
 34: FileInputStream in = new FileInputStream(jsonFile);  
 35: int len;  
 36: while ((len = in.read(buffer)) > 0) {  
 37: zos.write(buffer, 0, len);  
 38: }  
 39: in.close();  
 40: zos.closeEntry();  
 41: // remember close it  
 42: zos.close();  
 43:  
 44: }  
 45:  
 46:}

## .\pnctcheck\src\main\java\br\ufrj\coppetec\pnct\pnctcheck\PNCTCheck.java

1:package br.ufrj.coppetec.pnct.pnctcheck;  
 2:  
 3:import java.io.BufferedInputStream;  
 4:import java.io.BufferedOutputStream;  
 5:import java.io.File;  
 6:import java.io.FileInputStream;  
 7:import java.io.FileOutputStream;  
 8:import java.io.IOException;  
 9:import java.util.ArrayList;  
 10:import java.util.Arrays;  
 11:import java.util.Iterator;  
 12:import java.util.List;  
 13:import java.util.regex.Pattern;  
 14:  
 15:import org.apache.commons.compress.archivers.ArchiveEntry;  
 16:import org.apache.commons.compress.archivers.zip.ZipArchiveInputStream;  
 17:import org.apache.commons.io.FileUtils;  
 18:import org.apache.logging.log4j.LogManager;  
 19:import org.apache.logging.log4j.Logger;  
 20:  
 21:/\*\*  
 22: \* Hello world!  
 23: \*  
 24: \*/  
 25:public class PNCTCheck {  
 26:  
 27: private static Logger logger = LogManager.getLogger(PNCTCheck.class);  
 28:  
 29: private static final String FOLDER\_NAME = "pnct\_data";  
 30: private static final String[] END\_FILES\_NAMES = { "\_dadosOD.db", "\_dadosOD.json", "\_logOD.txt" };  
 31:  
 32: private final static int BUFFER = 512 \* 1024;  
 33: private static final String TMP\_IMPORT\_ZIP = System.getProperty("java.io.tmpdir") + "/processZip/";  
 34: private static File FOLDER\_TMP;  
 35:  
 36: public static void main(String[] args) {  
 37: try {  
 38: processFolder(FOLDER\_NAME);  
 39: } catch (Exception e) {  
 40: e.printStackTrace();  
 41: } finally {  
 42: if (FOLDER\_TMP.exists()) {  
 43: try {  
 44: FileUtils.deleteDirectory(FOLDER\_TMP);  
 45: } catch (IOException e) {  
 46: e.printStackTrace();  
 47: }  
 48: }  
 49: }  
 50: }  
 51:  
 52: private static void processFolder(String folderName) throws IOException, InterruptedException {  
 53: processFolder(new File(folderName));  
 54: }  
 55:  
 56: private static void processFolder(File folder) throws IOException, InterruptedException {  
 57: FOLDER\_TMP = new File(TMP\_IMPORT\_ZIP);  
 58: if (FOLDER\_TMP.exists()) {  
 59: FileUtils.deleteDirectory(FOLDER\_TMP);  
 60: }  
 61: FOLDER\_TMP.mkdir();  
 62: FOLDER\_TMP.deleteOnExit();  
 63:  
 64: if ((folder != null) && folder.exists()) {  
 65: // logger.info(folder.getAbsolutePath());  
 66: if (folder.isDirectory()) {  
 67: File[] files = folder.listFiles();  
 68: Arrays.sort(files);  
 69: for (File file : files) {  
 70: processFolder(file);  
 71: }  
 72: files = null;  
 73: } else if (folder.isFile()) {  
 74: if (folder.getName().endsWith(".zip")) {  
 75: processFileZip(folder);  
 76: }  
 77: }  
 78: }  
 79: }  
 80:  
 81: /\*\*  
 82: \* @param folder  
 83: \* @throws IOException  
 84: \*/  
 85: private static void processFileZip(File fileZip) throws IOException {  
 86: String posto = fileZip.getName().substring("posto".length(), fileZip.getName().length() - ".zip".length());  
 87: // logger.info("POSTO: " + posto);  
 88: int countLogs = 0;  
 89: Long lengthDadosDB = null;  
 90: List<File> files = extractFileZip(fileZip);  
 91: Pattern pattern = Pattern.compile("^" + posto + "0\\d{2}\_[a-z]{3,5}OD\\.[a-z]{2,4}$");  
 92: for (Iterator<File> iterator = files.iterator(); iterator.hasNext();) {  
 93: File file = iterator.next();  
 94: if (file.getParentFile().getName().equals("logs") && file.getName().startsWith("concentrador")) {  
 95: countLogs++;  
 96: } else if (file.getName().equals("dados.db")) {  
 97: lengthDadosDB = file.length();  
 98: if (lengthDadosDB == 0) {  
 99: logger.info("POSTO: " + posto + "\tERROR: " + file.getAbsolutePath());  
 100: }  
 101: } else {  
 102: boolean error = true;  
 103: for (String endFileName : END\_FILES\_NAMES) {  
 104: if (file.getName().endsWith(endFileName)) {  
 105: error = !pattern.matcher(file.getName()).find();  
 106: break;  
 107: }  
 108: }  
 109: if (error) {  
 110: logger.info("POSTO: " + posto + "\tERROR: " + file.getAbsolutePath());  
 111: } else {  
 112: // logger.info("POSTO: " + posto + "\tOK");  
 113: }  
 114: }  
 115: // logger.info(file.getAbsolutePath());  
 116: file.delete();  
 117: }  
 118: if (lengthDadosDB == null) {  
 119: logger.info("POSTO: " + posto + "\tERROR: não existe o arquivo dados.db");  
 120: }  
 121: if (countLogs == 0) {  
 122: logger.info("POSTO: " + posto + "\tERROR: não existe nenhum arquivo de log do concentrador");  
 123: }  
 124: }  
 125:  
 126: /\*\*  
 127: \* @param folder  
 128: \* @return  
 129: \* @throws IOException  
 130: \*/  
 131: private static List<File> extractFileZip(File fileZip) throws IOException {  
 132: List<File> files = new ArrayList<File>();  
 133:  
 134: FileInputStream fin = new FileInputStream(fileZip);  
 135: BufferedInputStream in = new BufferedInputStream(fin);  
 136: ZipArchiveInputStream zipIn = new ZipArchiveInputStream(in);  
 137:  
 138: ArchiveEntry entry = null;  
 139:  
 140: /\*\* Read the tar entries using the getNextEntry method \*\*/  
 141: while ((entry = (ArchiveEntry) zipIn.getNextEntry()) != null) {  
 142:  
 143: /\*\* If the entry is a directory, create the directory. \*\*/  
 144: File fileDest = new File(FOLDER\_TMP, entry.getName());  
 145: if (!fileDest.getParentFile().exists()) {  
 146: fileDest.getParentFile().mkdirs();  
 147: } else if (fileDest.exists()) {  
 148: fileDest.delete();  
 149: }  
 150: if (entry.isDirectory()) {  
 151: fileDest.mkdirs();  
 152: }  
 153: /\*\*  
 154: \* If the entry is a file,write the decompressed file to the disk and close destination stream.  
 155: \*\*/  
 156: else {  
 157: fileDest.createNewFile();  
 158: int count;  
 159: byte data[] = new byte[BUFFER];  
 160:  
 161: FileOutputStream fos = new FileOutputStream(fileDest);  
 162: BufferedOutputStream dest = new BufferedOutputStream(fos, BUFFER);  
 163: while ((count = zipIn.read(data, 0, BUFFER)) != -1) {  
 164: dest.write(data, 0, count);  
 165: }  
 166: dest.close();  
 167: files.add(fileDest);  
 168: }  
 169: }  
 170:  
 171: /\*\* Close the input stream \*\*/  
 172:  
 173: zipIn.close();  
 174: return files;  
 175: }  
 176:  
 177:}

## .\pnctcheck\src\test\java\br\ufrj\coppetec\pnct\pnctcheck\AppTest.java

1:package br.ufrj.coppetec.pnct.pnctcheck;  
 2:  
 3:import junit.framework.Test;  
 4:import junit.framework.TestCase;  
 5:import junit.framework.TestSuite;  
 6:  
 7:/\*\*  
 8: \* Unit test for simple PNCTCheck.  
 9: \*/  
 10:public class AppTest   
 11: extends TestCase  
 12:{  
 13: /\*\*  
 14: \* Create the test case  
 15: \*  
 16: \* @param testName name of the test case  
 17: \*/  
 18: public AppTest( String testName )  
 19: {  
 20: super( testName );  
 21: }  
 22:  
 23: /\*\*  
 24: \* @return the suite of tests being tested  
 25: \*/  
 26: public static Test suite()  
 27: {  
 28: return new TestSuite( AppTest.class );  
 29: }  
 30:  
 31: /\*\*  
 32: \* Rigourous Test :-)  
 33: \*/  
 34: public void testApp()  
 35: {  
 36: assertTrue( true );  
 37: }  
 38:}

## .\mobile\platforms\ios\www\controllers\pergunta\textunderscore extra.js

1:controllers.pergunta\_extra = {  
 2: config : function() {  
 3: var me = controllers.pergunta\_extra;  
 4: me.inicializaElementos();  
 5: me.progressoTela();  
 6: me.buttons();  
 7: },  
 8:  
 9: buttons : function() {  
 10: $("#pergunta\_extra\_avancar").click(function() {  
 11: var ok = controllers.pergunta\_extra.validar\_componentes();  
 12: if (ok) {  
 13: app.finalizaRegistro(function() {  
 14: app.trocaPagina('views/menu.html', controllers.menu);  
 15: });  
 16: }  
 17: })  
 18: },  
 19:  
 20: // Inicializa os elementos da tela  
 21: inicializaElementos : function() {  
 22:  
 23: util.inicializaSelectFrequencia('simples');  
 24:  
 25: var lista\_opcoes = [ 'Sim, trocaria minha rota atual por esta', 'Não, prefiro usar a rota atual',  
 26: 'Minha rota não seria afetada', 'Não sei informar' ];  
 27:   
 28: if (logins.user\_logado != null && (logins.user\_logado.usr == '019' || logins.user\_logado.usr == '149'  
 29: || logins.user\_logado.usr == '019t' || logins.user\_logado.usr == '149t')) {  
 30: $("#grupo\_pergunta\_extra\_p1").show();  
 31: util.inicializaSelect("select\_pergunta\_extra\_p1", lista\_opcoes);  
 32: if (logins.user\_logado != null && (logins.user\_logado.usr == '149' || logins.user\_logado.usr == '149t')) {  
 33: util.inicializaSelect("select\_pergunta\_extra\_p2", lista\_opcoes);  
 34: }  
 35: }  
 36: else if (logins.user\_logado != null && (logins.user\_logado.usr == '040' || logins.user\_logado.usr == '040t')) {  
 37: $("#grupo\_pergunta\_extra\_p3").show();  
 38: util.inicializaSelect("select\_pergunta\_extra\_p3", lista\_opcoes);  
 39: }  
 40: },  
 41:  
 42: // Controla o show e hide dos elementos da tela  
 43: progressoTela : function() {  
 44:  
 45: if (logins.user\_logado != null && (logins.user\_logado.usr == '019' || logins.user\_logado.usr == '019t')) {  
 46: util.progressoSelect("idPerguntaExtra", "select\_pergunta\_extra\_p1", "grupo\_pergunta\_extra\_avancar");  
 47: }  
 48: else if (logins.user\_logado != null && (logins.user\_logado.usr == '149' || logins.user\_logado.usr == '149t')) {  
 49: util.progressoSelect("idPerguntaExtra", "select\_pergunta\_extra\_p1", "grupo\_pergunta\_extra\_p2");  
 50: util.progressoSelect("idPerguntaExtra2", "select\_pergunta\_extra\_p2", "grupo\_pergunta\_extra\_avancar");  
 51: }  
 52: else if (logins.user\_logado != null && (logins.user\_logado.usr == '040' || logins.user\_logado.usr == '040t')) {  
 53: util.progressoSelect("idPerguntaExtra", "select\_pergunta\_extra\_p3", "grupo\_pergunta\_extra\_avancar");  
 54: }  
 55: else {  
 56: // Não deveria entrar aqui. Mostra o botão avançar para finalizar a pesquisa e não ficar travado.  
 57: $("#grupo\_pergunta\_extra\_avancar").show();  
 58: }  
 59: },  
 60:  
 61: // Controla as validações dos componentes de tela após clicar em AVANÇAR  
 62: validar\_componentes : function(id\_avancar) {  
 63: if (logins.user\_logado != null && (logins.user\_logado.usr == '019' || logins.user\_logado.usr == '019t')) {  
 64: return util.validaSelect("select\_pergunta\_extra\_p1", "Pergunta Extra");  
 65: }  
 66: else if (logins.user\_logado != null && (logins.user\_logado.usr == '149' || logins.user\_logado.usr == '149t')) {  
 67: var ok = util.validaSelect("select\_pergunta\_extra\_p1", "Pergunta Extra 1");  
 68: ok = ok && util.validaSelect("select\_pergunta\_extra\_p2", "Pergunta Extra 2");  
 69: return ok;  
 70: }  
 71: else if (logins.user\_logado != null && (logins.user\_logado.usr == '040' || logins.user\_logado.usr == '040t')) {  
 72: return util.validaSelect("select\_pergunta\_extra\_p3", "Pergunta Extra");  
 73: }  
 74: return false;  
 75: }  
 76:  
 77:};

## .\mobile\platforms\ios\www\js\index.js

1:/\* global util, ipadID \*/  
 2:  
 3:var app = {  
 4:  
 5: versao : '2.4.2',  
 6:   
 7: debugMode : true,  
 8:  
 9: login : function() {  
 10: var usuario = $("#usuario").val().trim();  
 11: var senha = $("#senha").val().trim();  
 12: if(app.sentido === null){  
 13: alert("Selecione o sentido!");  
 14: return;  
 15: }  
 16:  
 17: // configura identificador do ipad, para executar uma única vez(durante a instalação)  
 18: if (logins.autenticaMaster(usuario, senha)) {  
 19: ipadID.requestID(function(id) {  
 20: $("#ipadID").html(id);  
 21: });  
 22: } else if (logins.autentica(usuario, senha)) {  
 23: app.logger.log("Login efetuado pelo usuário: " + usuario);  
 24: // navega para página e executa o script de configuração depois do carregamento  
 25: app.trocaPagina("views/menu.html", controllers.menu)  
 26: // set: user\_login, senha\_login, posto e sentido  
 27: app.user\_login = usuario;  
 28: app.senha\_login = senha;  
 29: app.posto = String(usuario).substr(0, 3);  
 30: //app.sentido = String(usuario).substr(3, 2).toUpperCase();  
 31: if (isNaN(app.posto)) { // apenas para efeitos ao usuário de testes  
 32: app.posto = '000';  
 33: }  
 34: if ((app.sentido != 'AB') && (app.sentido != 'BA')) { // apenas para efeitos ao usuário de testes  
 35: app.sentido = 'AB';  
 36: }  
 37: // limpa o registro  
 38: app.limpaRegistro();  
 39: }  
 40: },  
 41:  
 42: logout : function() {  
 43: // limpa o registro  
 44: app.limpaRegistro();  
 45: // realiza logout  
 46: $("#usuario").val('').textinput("refresh");  
 47: $("#senha").val('').textinput("refresh");  
 48: $("#agregador input[type='radio']").prop('checked', false);  
 49: $("#entrar").prop('disabled', true);  
 50: $(":mobile-pagecontainer").pagecontainer("change", $("#page\_login"));  
 51: app.logger.log('Logout');  
 52: app.user\_login = null;  
 53: app.senha\_login = null;  
 54: app.posto = null;  
 55: app.sentido = null;  
 56: app.restart(); //problably the only line we need  
 57: },  
 58:  
 59: exportaDbToJson : function() {  
 60: if (app.filePaths) {  
 61: resolveLocalFileSystemURL(app.filePaths.externalFolder, function(dir) {  
 62: var newJsonName = ipadID.id + "\_" + app.jsonName;  
 63: app.removeFile(newJsonName, app.filePaths.externalFolder, function() {  
 64: realExporter(newJsonName, dir);  
 65: }, function() {  
 66: realExporter(newJsonName, dir);  
 67: });  
 68: }, function(err) {  
 69: app.logger.log('ERRO ao acessar o folder externo ' + app.filePaths.externalFolder + ' ' + JSON.stringify(err));  
 70: });  
 71: } else {  
 72: alert('Operação não realizada, o sistema de arquivos não foi definido');  
 73: }  
 74:  
 75: function realExporter(jsonName, dir) {  
 76: dir.getFile(jsonName, {  
 77: create : true  
 78: }, function(file) {  
 79: app.logger.log("arquivo JSON: ", file);  
 80: jsonWriter.setJsonFile(file);  
 81: file.createWriter(function(fileWriter) {  
 82: jsonWriter.setJsonWriter(fileWriter);  
 83: myDb.exportaDbToJson(jsonWriter);  
 84: }, function() {  
 85: app.logger.log('ERRO criando o escritor do JSON a ser exportado.');  
 86: });  
 87: }, function(e) {  
 88: app.logger.log('ERRO ao criar o arquivo JSON a ser exportado: ' + e.message);  
 89: });  
 90: }  
 91: },  
 92:  
 93: duplicaDb : function() {  
 94: if (app.filePaths) {  
 95: app.openDB(function(){  
 96: app.database.close(function() {  
 97: app.copyFile(app.dbName, app.filePaths.dbFolder, app.filePaths.externalFolder, false, function(newName) {  
 98: alert('Banco de dados ' + newName + ' exportado com sucesso.');  
 99: //app.openDB(); a conexão com o banco de dados permanece fechada  
 100: });  
 101: }, function(err) {  
 102: //falha na cópia do banco  
 103: //copia banco com problema  
 104: app.logger.log('Exportando bando de dados corrompido');  
 105: app.copyFile(app.dbName, app.filePaths.dbFolder, app.filePaths.externalFolder, true, function(newName) {  
 106: alert('Banco de dados ' + newName + ' exportado!\n\nEntre em contato com o suporte.',  
 107: "Banco da Dados Corrompido", null, "info");  
 108: //apaga banco atual  
 109: app.removeFile(app.dbName, app.filePaths.dbFolder,  
 110: function(){ //removeu com sucesso  
 111: app.logger.log("Banco de dados corrompido removido do sistema de arquivos do app.");  
 112: //app.openDB(); a conexão com o banco de dados permanece fechada  
 113: }, function(){//falhou quando removia  
 114: app.logger.log("Falha removendo banco de dados corrompido");  
 115: alert('Houve uma falha grava no sistema. \n Continue o processo de exportação, entre em contato com o suporte e NÃO CONTINUE A USAR ESSE IPAD.',  
 116: "Erro no Banco da Dados", null, "error");  
 117: //app.openDB(); a conexão com o banco de dados permanece fechada  
 118: });  
 119:  
 120: });  
 121: app.logger.log(JSON.stringify(err));  
 122: });  
 123: // Exporta Log  
 124: app.duplicaLog();   
 125: });  
 126:   
 127: } else {  
 128: alert('Operação não realizada, o sistema de arquivos não foi definido');  
 129: }  
 130: },  
 131:  
 132: duplicaLog : function() {  
 133: app.copyFile(app.logFileName, cordova.file.dataDirectory, app.filePaths.externalFolder, false, function(newName) {  
 134: alert('Arquivo de log ' + newName + ' exportado com sucesso.');  
 135: });  
 136: },  
 137:  
 138: /\*\*  
 139: \* Validates interview's cancellation  
 140: \*   
 141: \* @param Function  
 142: \* cb receives a Boolean representing the success in validation  
 143: \*/  
 144: validaCancelamento : function(cb) {  
 145: prompt("Insira a senha para cancelar a entrevista.", function(result) {  
 146: // botão prosseguir cancelamento  
 147: if (result == app.senha\_login) {  
 148: cb(true);  
 149: } else {  
 150: confirm("Senha incorreta para o cancelamento.\nDeseja tentar novamente?",  
 151: // ok button pressed  
 152: function() {  
 153: app.validaCancelamento(cb);  
 154:  
 155: }, function() {  
 156: cb(false);  
 157: }, "Senha incorreta.");  
 158:  
 159: }  
 160: }, "Cancelamento de entrevista", "Cancelar entrevista", "Voltar", "Senha", 'password');  
 161: },  
 162:  
 163: validaOperacoes : function(operacao, txtPrompt, titlePrompt, txtConfirm, titleConfirm, btnOKPrompt, btnCancelPrompt) {  
 164: prompt(txtPrompt, function(result) {  
 165: if (result == app.senha\_login) {  
 166: operacao();  
 167: } else {  
 168: confirm(txtConfirm,  
 169: function() {  
 170: app.validaOperacoes(operacao, txtPrompt, titlePrompt, txtConfirm, titleConfirm, btnOKPrompt,  
 171: btnCancelPrompt); // try again  
 172: }, null, titleConfirm);  
 173: }  
 174: }, titlePrompt, btnOKPrompt, btnCancelPrompt, "Senha", 'password');  
 175: },  
 176:  
 177: /\*  
 178: \* Application constructor  
 179: \*/  
 180: initialize : function() {  
 181: this.bindEvents();  
 182: this.extraConfig();  
 183: this.baseUrl = window.location.href.replace("index.html", "");  
 184: // configuring dialogs  
 185: window.alert = myDialogs.alert;  
 186: window.prompt = myDialogs.prompt;  
 187: window.confirm = myDialogs.confirm;  
 188:  
 189: },  
 190:  
 191: /\*  
 192: \* bind any events that are required on startup to listeners:  
 193: \*/  
 194: bindEvents : function() {  
 195: document.addEventListener('deviceready', app.onDeviceReady);  
 196: console.log('bindindEvents');  
 197: },  
 198:  
 199: /\*  
 200: \* this runs when the device is ready for user interaction:  
 201: \*/  
 202: onDeviceReady : function() {  
 203: app.logger.log('device ready');  
 204: window.addEventListener('batterycritical',function(status){  
 205: alert("A bateria atingiu nível crítco! \nCarregue o ipad imediatamente.\nNível de bateria: " + status.level + "%",  
 206: 'Alerta de Bateria', null, 'info');  
 207: }, false);  
 208: window.addEventListener('batterylow', function(status){  
 209: alert("O nível da bateria está baixo. \nCarregue o ipad assim que possível.\nNível de bateria: " + status.level + "%",  
 210: 'Alerta de Bateria', null, 'info');  
 211: }, false);  
 212:   
 213: document.addEventListener('pause', function(){  
 214: console.log("[" + (new Date()) + "]" + 'Ocorrência de evento: PAUSE');  
 215: }, false);  
 216:   
 217: document.addEventListener('resume', function(){  
 218: console.log("[" + (new Date()) + "]" + 'Ocorrência de evento: RESUME');  
 219: }, false);  
 220:   
 221: document.addEventListener('resign', function(){  
 222: console.log("[" + (new Date()) + "]" + 'Ocorrência de evento: RESIGN');  
 223: }, false);  
 224:  
 225: // a plataforma Browser não permite o desenvolvimento das escritas em arquivo  
 226: if (device.platform == 'iOS' || device.platform == 'Android') {  
 227: app.filePaths = {}; // inicia a variável  
 228:  
 229: setTimeout(function() {  
 230: navigator.splashscreen.hide();  
 231: console.log("esperando " + device.platform);  
 232: }, 3000);  
 233:  
 234:  
 235: // durante o debug, inicia o sistema de arquivos  
 236: if (app.debugMode){  
 237: navigator.notification.alert("App em modo debug. \nConecte o debugger", app.onFileSystemReady, 'Alerta de desenvolvimento', 'OK');  
 238: }else{  
 239: app.onFileSystemReady();  
 240: }   
 241: }  
 242:  
 243: // nessa altura do desenvolvimento não faz sentido  
 244: /\*  
 245: \* if (device.platform == 'browser' && device.model != 'Firefox') { alert('ATENÇÃO!!! \n Use o Firefox para fazer a  
 246: \* simulação (cordova run browser --target=firefox)'); }  
 247: \*/  
 248:  
 249: if (device.platform == 'iOS') {  
 250: // configurando a statusBar  
 251: StatusBar.overlaysWebView(false);  
 252: StatusBar.backgroundColorByName("black");  
 253: // black, darkGray, lightGray, white, gray, red, green, blue, cyan, yellow, magenta, orange, purple, brown  
 254:  
 255: // ajusta os caminhos para o iOS  
 256: app.filePaths['externalFolder'] = cordova.file.documentsDirectory;  
 257: app.filePaths['dbFolder'] = 'cdvfile://localhost/library/LocalDatabase';  
 258:  
 259: } else if (device.platform == 'Android') {  
 260: StatusBar.hide();  
 261: StatusBar.backgroundColorByName("darkGray");  
 262:  
 263: // ajusta os caminhos para o Android  
 264: app.filePaths['externalFolder'] = cordova.file.externalDataDirectory;  
 265: app.filePaths['dbFolder'] = cordova.file.applicationStorageDirectory + "databases";  
 266: }  
 267:  
 268: // alert sem a página como título  
 269: /\*  
 270: \* window.alert = function(txt, cb) { navigator.notification.alert(txt, cb, "Aviso", "Fechar"); }  
 271: \*/  
 272:  
 273: // novo valor para variavel criada no extraConfig()  
 274: app.uuid\_device = device.uuid;  
 275:  
 276: },  
 277:  
 278: onFileSystemReady : function() {  
 279:  
 280: app.logger.log("folder dos dados: ", cordova.file.dataDirectory);  
 281:  
 282: // setting ipadID file and value  
 283: window.resolveLocalFileSystemURL(cordova.file.dataDirectory,  
 284: // success resolving dir  
 285: function(dir) {  
 286: app.logger.log('file system ready: ', dir);  
 287: // modifica o atributo ipadId.id que havia sido configurado no extraConfig() como 'browser'  
 288: // reading ipadID file  
 289: dir.getFile(ipadID.ipadIDFileName, {  
 290: create : false  
 291: }, function(file) {  
 292: ipadID.readId(file, function(id) {  
 293: $("#ipadID").html(id);  
 294: afterIpadID(id, dir);  
 295: });  
 296: }, function(e) {  
 297: ipadID.requestID(function(id) {  
 298: $("#ipadID").html(id);  
 299: afterIpadID(id, dir);  
 300: });  
 301: });  
 302: },  
 303: // error resolving dir  
 304: function(err) {  
 305: app.logger.log("erro no sistema de arquivos: " + err.name + " -> " + err.message);  
 306: alert("erro no sistema de arquivos: " + err.name + " -> " + err.message);  
 307: });  
 308:  
 309: // internal function  
 310: function afterIpadID(id, dir) {  
 311:  
 312: dir.getFile(app.logFileName, {  
 313: create : true  
 314: },  
 315: // success getting file  
 316: function(file) {  
 317: app.logger.log("arquivo de log: ", file);  
 318: myLogger.setLogFile(file);  
 319: file.createWriter(function(fileWriter) {  
 320: myLogger.setLogWriter(fileWriter);  
 321: // novo valor para variável criada no extraConfig  
 322: app.logger = myLogger;  
 323: app.openDB(function(){// if successfully create fileWriter, open database for testing  
 324: app.logger.log("Conexão inicial com o banco para verificação");  
 325: app.database.close(function(){  
 326: app.logger.log("Conexão com o banco fechada");  
 327: });  
 328: });   
 329:   
 330: },  
 331: // error creating fileWriter  
 332: function() {  
 333: app.logger.log('erro criando o escritor do log');  
 334: });  
 335: });  
 336: }  
 337: ;  
 338:  
 339: },  
 340:  
 341: openDB : function(cb) {  
 342: app.database = sqlitePlugin.openDatabase({  
 343: name : app.dbName,  
 344: iosDatabaseLocation : 'default'  
 345: },  
 346: // sucsess  
 347: function() {  
 348: app.logger.log('Conexão com o banco de dados criada com sucesso.');  
 349: myDb.cretateTblDados(function(){  
 350: myDb.sanitize(function(){  
 351: if(util.isFunction(cb)){  
 352: cb();  
 353: }  
 354: });  
 355: });   
 356: },  
 357: // fail  
 358: function(err) {  
 359: app.logger.log('ERRO ao tentar conectar com o banco de dados.');  
 360: app.logger.log(JSON.stringify(err));  
 361: alert("O Banco da dados foi corrompido. \n Exporte todos os dados e entre em contato com o suporte.",  
 362: "Erro no Banco da Dados", null, "error");  
 363: });  
 364: },  
 365:   
 366: extraConfig : function() {  
 367: // initialize panel  
 368: $(function() {  
 369: $("[data-role=panel]").panel().enhanceWithin();  
 370: });  
 371: $('#dataHoraLogin').html(util.formatDateTimeToDisplay());  
 372: var updater\_dataTimeToDisplay = setInterval(function(){  
 373: $('#dataHoraLogin').html(util.formatDateTimeToDisplay());  
 374: },10000);  
 375:   
 376: $("#versao").html(app.versao);  
 377: $("#entrar").click(function(){  
 378: clearInterval(updater\_dataTimeToDisplay);  
 379: app.login();  
 380: });  
 381: $("#entrar").prop('disabled', true);  
 382: $("#agregador input[type='radio']").prop('checked', false).change(function(event, ui){  
 383: $("#entrar").prop('disabled', false);  
 384: app.sentido = $(this).val();  
 385: });  
 386:  
 387: // valores iniciais (vão ficar assim se estiver usando o browser)  
 388: app.uuid\_device = "browser";  
 389:   
 390: ipadID.id = 'browser';  
 391:   
 392: // botões do menu  
 393: $("#btn\_sair").click(  
 394: function() {  
 395: app.validaOperacoes(app.logout, "Insira a senha para realizar o logout.", "Logout"  
 396:, "Senha incorreta.\nDeseja tentar novamente?", "Senha Incorreta", "Logout", "Voltar");  
 397: });  
 398:   
 399: /\*  
 400: \* A partir da Fase 2 o log passou a ser sempre exportado junto ao DB  
 401: $("#duplica\_log").click(  
 402: function() {  
 403: app.validaOperacoes(app.duplicaLog, "Insira a senha para exportar o Log de operações.",  
 404: "Exportar log de operações", "Senha incorreta para a exportação.\nDeseja tentar novamente?",  
 405: "Senha Incorreta", "Exportar", "Voltar");  
 406: });  
 407: \*/  
 408: $("#duplica\_db").click(  
 409: function() {  
 410: app.validaOperacoes(app.duplicaDb, "Insira a senha para exportar o banco de dados.",  
 411: "Exportar banco de dados", "Senha incorreta para a exportação.\nDeseja tentar novamente?",  
 412: "Senha Incorreta", "Exportar", "Voltar");  
 413: });  
 414:   
 415: /\*  
 416: \* Funçao retirada a partir da fase 2 do projeto  
 417: $("#exporta\_db\_to\_json").click(  
 418: function() {  
 419: app.validaOperacoes(app.exportaDbToJson, "Insira a senha para exportar os dados no formato JSON.",  
 420: "Exportar JSON", "Senha incorreta para a exportação.\nDeseja tentar novamente?", "Senha Incorreta",  
 421: "Exportar", "Voltar");  
 422: });  
 423: \*/  
 424: // remove o filtro original do autocomplete para poder filtrar acentos  
 425: $.mobile.filterable.prototype.options.filterCallback = function(index, value) {  
 426: return false  
 427: };  
 428:   
 429: //configurando o spinner  
 430: $.mobile.loader.prototype.options.text = "Aguarde";  
 431: $.mobile.loader.prototype.options.textVisible = true;  
 432: $.mobile.loader.prototype.options.theme = "b";  
 433: $.mobile.loader.prototype.options.html = "";  
 434: },  
 435: /\*\*  
 436: \* Change pages within app  
 437: \* @param view -> new view  
 438: \* @param controller -> controller that will run on view change  
 439: \* @param String changeFunction -> ['old', 'new'] -- default new  
 440: \*   
 441: \*/  
 442: trocaPagina : function(view, controller, changeFunction) {  
 443:   
 444: var changeF = (util.isEmpty(changeFunction) || changeFunction != 'old')? newChange : oldChange;  
 445: var abortOperation = false;  
 446:   
 447: if (view == null) {  
 448: app.logger.log("[ERRO] trocaPagina: view null");  
 449: abortOperation = true;  
 450: }   
 451:   
 452: if (controller == null){  
 453: app.logger.log("[ERRO] trocaPagina: controller null");  
 454: abortOperation = true;  
 455: }  
 456:   
 457: if (abortOperation){  
 458: myDialogs.alert("Ocorreu um erro interno na aplicação. A entrevista será cancelada.",  
 459: "Erro interno da aplicação.",  
 460: function(){  
 461: app.cancelaRegistro();  
 462: app.trocaPagina('views/menu.html', controllers.menu);  
 463: }, 'error');  
 464: } else {  
 465: try {  
 466: app.onChangeHandler.controller = controller.config;  
 467: $(":mobile-pagecontainer").off("pagecontainershow", app.onChangeHandler.handler).on("pagecontainershow",  
 468: app.onChangeHandler.handler);  
 469: } catch(exc) {  
 470: app.logger.log("[ERRO] trocaPagina: excecao no changeHandler. Detalhes: ")  
 471: app.logger.log(exc.message)  
 472: }  
 473:  
 474: try {  
 475: changeF(view);  
 476: } catch(exc) {  
 477: app.logger.log("[ERRO] trocaPagina: excecao ao chamar pagecontainer. Detalhes: ");  
 478: app.logger.log(exc.message);  
 479: }  
 480:   
 481: app.logger.log(view);  
 482: app.logger.log('número de mudanças de página: ' + ++app.changesCounter);  
 483:   
 484: }  
 485:   
 486: function oldChange(v){  
 487: $(":mobile-pagecontainer").pagecontainer("change", app.baseUrl + v);  
 488: app.logger.log('oldTrocaPagina');  
 489: }  
 490:   
 491: function newChange(v){  
 492: $(":mobile-pagecontainer").pagecontainer("change", app.baseUrl + v, {reload : true, changeHash : false, transition : 'none'});  
 493:// app.logger.log('newTrocaPagina');  
 494: }  
 495: },  
 496:  
 497: // para apendar coisas aos controllers  
 498: onChangeHandler : {  
 499: handler : function() {  
 500: try {  
 501:// $("section[data-role=page]").append(  
 502:// "<footer data-role='footer' data-position='fixed'> \  
 503:// <span class='versao'>Versão: <span id='versao'></span></span> \  
 504:// <span class='data\_atual'>Data: <span id='data\_atual'></span></span> \  
 505:// <span class='posto'>Posto: <span id='posto'></span></span> \  
 506:// <span class='sentido'>Sentido: <span id='sentido'></span></span> \  
 507:// <span class='ipad'>id-iPad: <span id='ipadID'></span></span> \  
 508:// </footer>"  
 509:// );  
 510:// $("footer[data-role='footer']").toolbar({  
 511:// defaults : true,  
 512:// position: "fixed"  
 513:// });  
 514:   
 515: app.onChangeHandler.controller();  
 516:  
 517: // Botão "Cancelar"  
 518: $("#btn\_cancelar").click(app.cancelar);  
 519:  
 520: // Input radio  
 521: $('input[type="radio"]').click(function() {  
 522: $(this).focus();  
 523: });  
 524:  
 525: // Input integer  
 526: $('input[typeMask="integer"]').each(function(key, input) {  
 527: var minValue = 0;  
 528: var maxValue = 99999999;  
 529: if (!util.isEmpty($(input).attr('min'))) {  
 530: minValue = Number($(input).attr('min'));  
 531: }  
 532: if (!util.isEmpty($(input).attr('max'))) {  
 533: maxValue = Number($(input).attr('max'));  
 534: }  
 535: $(input).inputmask('integer', {  
 536: min : minValue,  
 537: max : maxValue  
 538: });  
 539: });  
 540:  
 541: // Input money  
 542: $('input[typeMask="money-BRL"]').each(function(key, input) {  
 543: $(input).maskMoney({  
 544: prefix : "R$ ",  
 545: affixesStay : "true",  
 546: thousands : ".",  
 547: decimal : ","  
 548: });  
 549: });  
 550:  
 551: $('input[mask]').each(function(key, input) {  
 552: var decimalSeparator = $(input).attr('mask-decimal-separator');  
 553: if (util.isEmpty(decimalSeparator)) {  
 554: $(input).inputmask($(input).attr('mask'), {  
 555: 'autoUnmask' : true  
 556: });  
 557: } else {  
 558: $(input).inputmask($(input).attr('mask'), {  
 559: 'autoUnmask' : true,  
 560: numericInput : true,  
 561: radixPoint : decimalSeparator  
 562: });  
 563: }  
 564: });  
 565:   
 566:   
 567: $(".versao #versao").html(app.versao);  
 568: $(".data\_atual #data\_atual").html(util.formateDateOnly(new Date()));  
 569: $(".ipad #ipadID").html(ipadID.id);  
 570: $(".posto #posto").html(app.posto);  
 571: $(".sentido #sentido").html(app.sentido);  
 572:  
 573: if (typeof device != 'undefined' && device.platform == "Android") {  
 574: StatusBar.hide();  
 575: }  
 576: //removendo outras páginas do DOM  
 577: $('div[data-role=page]:hidden').remove();  
 578: } catch (e) {  
 579: app.logger.log(e);  
 580: alert(e.message);  
 581: }  
 582: },  
 583:  
 584: controller : null,  
 585: },  
 586:  
 587: getAtributo : function(nome) {  
 588: return registro[nome];  
 589: },  
 590:  
 591: setAtributo : function(nome, valor) {  
 592: registro[nome] = valor;  
 593: app.logger.log(JSON.stringify(registro));  
 594: },  
 595:  
 596: splitAtributo : function(nome) {  
 597: var valor = registro[nome];  
 598: if (!util.isEmpty(valor)) {  
 599: var splited = String(valor).split("|");  
 600: if (splited.length > 1) {  
 601: valor = splited[splited.length - 1].trim();  
 602: if (!util.isEmpty(valor)) {  
 603: if (isNaN(valor)) {  
 604: app.setAtributo(nome, valor);  
 605: } else {  
 606: app.setAtributo(nome, parseInt(valor));  
 607: }  
 608: }  
 609: }  
 610: }  
 611: },  
 612:  
 613: cancelar : function() {  
 614: app.validaCancelamento(function(result) {  
 615: if (result) {  
 616: app.cancelaRegistro(function() {  
 617: app.trocaPagina('views/menu.html', controllers.menu);  
 618: });  
 619: }  
 620: });  
 621: },  
 622:  
 623: limpaRegistro : function() {  
 624: app.logger.log('Limpando registro', true);  
 625: registro = {};  
 626: },  
 627:  
 628: iniciaRegistro : function() {  
 629: try {  
 630: var now = new Date();  
 631: app.logger.log('Iniciando registro');  
 632: app.limpaRegistro();  
 633: app.setAtributo('id', ipadID.id + util.getTimeUnixTimestamp(now));  
 634: app.setAtributo('login', app.user\_login);  
 635: app.setAtributo('idPosto', app.posto);  
 636: app.setAtributo('sentido', app.sentido);  
 637: app.setAtributo('uuid', app.uuid\_device);  
 638: app.setAtributo('dataIniPesq', util.getTimeDefaultFormated(now));  
 639: app.setAtributo('dataIniPesqAbsolute', util.getTimeUnixTimestamp(now));  
 640: app.setAtributo('idIpad', ipadID.id);  
 641: app.logger.log(JSON.stringify(registro));  
 642: app.logger.log('Registro iniciado: ' + registro.id);  
 643: } catch (e) {  
 644: app.logger.log(e.message);  
 645: }  
 646: },  
 647:  
 648: setCamposDerivados : function() {  
 649: try {  
 650: //Treinamento  
 651: if (app.isTreinamento){  
 652: app.setAtributo('treinamento', 1);  
 653: }else{  
 654: app.setAtributo('treinamento', 0);   
 655: }  
 656:   
 657:   
 658: // TIPO VEICULO  
 659: if (!util.isEmpty(app.getAtributo('tipo'))) {  
 660: var tipoReal = app.getAtributo('tipo').split("\_")[0];  
 661: app.setAtributo('tipo', tipoReal.toUpperCase());  
 662: } else if (registro.cancelado != 1) {  
 663: app.setAtributo('erro', "ERRO (tipo vazio)");  
 664: app.logger.log(registro.erro + " no registro: ", registro.id);  
 665: }  
 666:   
 667: // PLACA  
 668: if ((!util.isEmpty(registro.placaEstrangeira)) && (!registro.placaEstrangeira)) {  
 669: if ((!util.isEmpty(registro.placa\_letras)) && (!util.isEmpty(registro.placa\_numeros))) {  
 670: app.setAtributo('placa', registro.placa\_letras.toUpperCase() + "-" + registro.placa\_numeros);  
 671: }  
 672: } else if ((!util.isEmpty(registro.placaEstrangeira)) && registro.placaEstrangeira) {  
 673: if (!util.isEmpty(registro.placa\_unica)) {  
 674: app.setAtributo('placa', registro.placa\_unica.toUpperCase());  
 675: }  
 676: }  
 677: if (util.isEmpty(registro.placa)) {  
 678: if (registro.cancelado != 1) {  
 679: app.setAtributo('erro', "ERRO (placa vazio)");  
 680: app.logger.log(registro.erro + " no registro: ", registro.id);  
 681: }  
 682: }  
 683:   
 684: // TIPO DE COMBUSTÍVEL  
 685: if (util.isEmpty(registro.idCombustivel) && (!util.isEmpty(registro.categoria))  
 686: && (registro.categoria == 'Onibus' || registro.categoria == 'Pesado')) {  
 687: app.setAtributo('idCombustivel', util.getIdFromTabelaAuxiliar('Diesel', lista\_combustivel));  
 688: }  
 689: if (registro.cancelado != 1) {  
 690: if (util.isEmpty(registro.categoria)) {  
 691: app.setAtributo('erro', "ERRO (categoria vazio)");  
 692: app.logger.log(registro.erro + " no registro: ", registro.id);  
 693: }  
 694: if (util.isEmpty(registro.idCombustivel)) {  
 695: app.setAtributo('erro', "ERRO (idCombustivel vazio)");  
 696: app.logger.log(registro.erro + " no registro: ", registro.id);  
 697: }  
 698: }  
 699:   
 700: // FRENQUENCIA  
 701: if (!util.isEmpty(registro.frequenciaPeriodo)) {  
 702: app.setAtributo('frequenciaPeriodo', util.getListaFrequencias()[Number(registro.frequenciaPeriodo) - 1]);  
 703: if (registro.frequenciaPeriodo == 'Eventualmente') {  
 704: app.setAtributo('frequenciaQtd', 1);  
 705: } else if (util.isEmpty(registro.frequenciaQtd) || (Number(registro.frequenciaQtd) <= 0)) {  
 706: if (registro.cancelado != 1) {  
 707: app.setAtributo('erro', "ERRO (frequenciaQtd vazio)");  
 708: app.logger.log(registro.erro + " no registro: ", registro.id);  
 709: }  
 710: }  
 711: } else if (registro.cancelado != 1) {  
 712: app.setAtributo('erro', "ERRO (frequenciaPeriodo vazio)");  
 713: app.logger.log(registro.erro + " no registro: ", registro.id);  
 714: }  
 715:   
 716: // ORIGEM: MUNICÍPIO  
 717: app.splitAtributo('idOrigemMunicipio');  
 718: if (util.isEmpty(registro.idOrigemMunicipio) && (registro.idOrigemPais == 1)) { // Brasil  
 719: if (registro.cancelado != 1) {  
 720: app.setAtributo('erro', "ERRO (idOrigemMunicipio vazio)");  
 721: app.logger.log(registro.erro + " no registro: ", registro.id);  
 722: }  
 723: }  
 724:   
 725: // DESTINO: MUNICÍPIO  
 726: app.splitAtributo('idDestinoMunicipio');  
 727: if (util.isEmpty(registro.idDestinoMunicipio) && (registro.idDestinoPais == 1)) { // Brasil  
 728: if (registro.cancelado != 1) {  
 729: app.setAtributo('erro', "ERRO (idDestinoMunicipio vazio)");  
 730: app.logger.log(registro.erro + " no registro: ", registro.id);  
 731: }  
 732: }  
 733:   
 734: // ESPECÍFICOS PARA FLUXO DE CARGA  
 735: if (app.getAtributo('classeVeiculo') == 'carga') {  
 736:   
 737: // RNTRC  
 738: if ((!util.isEmpty(registro.placaVermelha)) && registro.placaVermelha) {  
 739: if (!util.isEmpty(registro.placa\_vermelha\_rntrc\_sel) && !util.isEmpty(registro.placa\_vermelha\_rntrc\_num)) {  
 740: app.setAtributo('rntrc', String(registro.placa\_vermelha\_rntrc\_sel)  
 741: + String(registro.placa\_vermelha\_rntrc\_num));  
 742: } else {  
 743: if (registro.cancelado != 1) {  
 744: app.setAtributo('erro', "ERRO (rntrc vazio)");  
 745: app.logger.log(registro.erro + " no registro: ", registro.id);  
 746: }  
 747: }  
 748: }  
 749:   
 750: // RISCO E ONU  
 751: if (registro.possuiCargaPerigosa) {  
 752: var idNumRisco = util.findValueInList(registro.idNumeroDeRisco, lista\_numero\_risco);  
 753: if (!util.isEmpty(idNumRisco)) {  
 754: app.setAtributo('idNumeroDeRisco',idNumRisco);  
 755: } else {  
 756: if (registro.cancelado != 1) {  
 757: app.setAtributo('erro', "ERRO (idNumeroDeRisco não encontrado)");  
 758: app.logger.log(registro.erro + " no registro: ", registro.id);  
 759: }  
 760: }  
 761:  
 762: var idNumOnu = util.findValueInList(registro.idNumeroDaOnu, lista\_numero\_onu);  
 763: if (!util.isEmpty(idNumOnu)) {  
 764: app.setAtributo('idNumeroDaOnu',idNumOnu);  
 765: } else {  
 766: if (registro.cancelado != 1) {  
 767: app.setAtributo('erro', "ERRO (idNumeroDaOnu não encontrado)");  
 768: app.logger.log(registro.erro + " no registro: ", registro.id);  
 769: }  
 770: }  
 771: }  
 772:   
 773: //POSSUI CARGA  
 774: if (registro.possui\_carga) {  
 775: // TIPO DE PRODUTO (CARGA)  
 776: app.splitAtributo('idProduto');  
 777: if (util.isEmpty(registro.idProduto) && (registro.cancelado != 1)) {  
 778: app.setAtributo('erro', "ERRO (idProduto vazio)");  
 779: app.logger.log(registro.erro + " no registro: ", registro.id);  
 780: }  
 781:  
 782: // PESO DA CARGA ('ton' -> 'kg')  
 783: if (!util.isEmpty(registro.pesoDaCarga) && ((typeof registro.pesoDaCarga) != 'number')) {  
 784: var peso = Number(registro.pesoDaCarga);  
 785: if (registro.unidadePesoDaCarga == 'kg') {  
 786: peso = peso / 10;  
 787: } else {  
 788: peso = peso \* 100;  
 789: }  
 790: app.setAtributo('pesoDaCarga', peso);  
 791: }  
 792: if (util.isEmpty(registro.pesoDaCarga) && (registro.cancelado != 1)) {  
 793: app.setAtributo('erro', "ERRO (pesoDaCarga vazio)");  
 794: app.logger.log(registro.erro + " no registro: ", registro.id);  
 795: }  
 796:   
 797: // MUNICÍPIO EMBARQUE DA CARGA  
 798: app.splitAtributo('municipioEmbarqueCarga');  
 799: if (util.isEmpty(registro.municipioEmbarqueCarga) && registro.sabe\_embarque && (registro.cancelado != 1)) {  
 800: app.setAtributo('erro', "ERRO (municipioEmbarqueCarga vazio)");  
 801: app.logger.log(registro.erro + " no registro: ", registro.id);  
 802: }  
 803:   
 804: // MUNICÍPIO DESEMBARQUE DA CARGA  
 805: app.splitAtributo('municipioDesembarqueCarga');  
 806: if (util.isEmpty(registro.municipioDesembarqueCarga) && registro.sabe\_desembarque && (registro.cancelado != 1)) {  
 807: app.setAtributo('erro', "ERRO (municipioDesembarqueCarga vazio)");  
 808: app.logger.log(registro.erro + " no registro: ", registro.id);  
 809: }  
 810: } else {  
 811: // NAO POSSUI CARGA  
 812: app.setAtributo('idProduto', 3000); // Produto VAZIO  
 813: app.setAtributo('pesoDaCarga', null);  
 814: app.setAtributo('unidadePesoDaCarga', null);  
 815: app.setAtributo('valorDoFrete', null);  
 816: app.setAtributo('valorDaCarga', null);  
 817: app.setAtributo('embarqueCargaNaoSabe', null);  
 818: app.setAtributo('embarque\_uf', null);  
 819: app.setAtributo('municipioEmbarqueCarga', null);  
 820: app.setAtributo('idLocalEmbarqueCarga', null);  
 821: app.setAtributo('desembarqueCargaNaoSabe', null);  
 822: app.setAtributo('desembarque\_uf', null);  
 823: app.setAtributo('municipioDesembarqueCarga', null);  
 824: app.setAtributo('idLocalDesembarqueCarga', null);  
 825:  
 826: // POSSUI CARGA ANTERIOR  
 827: if (registro.carga\_anterior) {  
 828: app.splitAtributo('idCargaAnterior');  
 829: if (util.isEmpty(registro.idCargaAnterior) && registro.carga\_anterior && (registro.cancelado != 1)) {  
 830: app.setAtributo('erro', "ERRO (idCargaAnterior vazio)");  
 831: app.logger.log(registro.erro + " no registro: ", registro.id);  
 832: }  
 833: } else {  
 834: app.setAtributo('idCargaAnterior', 3000); // Produto VAZIO  
 835: }  
 836: }  
 837:   
 838: // CARGA SUGESTÃO PARADA OBRIGATÓRIA MUNICÍPIOS  
 839: app.splitAtributo('paradaObrigatoriaMunicipio1');  
 840: app.splitAtributo('paradaObrigatoriaMunicipio2');  
 841:   
 842: if (util.isEmpty(registro.paradaObrigatoriaMunicipio1) && util.isEmpty(registro.paradaObrigatoriaMunicipio1)) {  
 843: if ((!registro.municipiosParadaNaoSabe) && (registro.cancelado != 1)) {  
 844: app.setAtributo('erro', "ERRO (paradaObrigatoriaMunicipio1 ou paradaObrigatoriaMunicipio2 vazio)");  
 845: app.logger.log(registro.erro + " no registro: ", registro.id);  
 846: }  
 847: }  
 848: }  
 849: }  
 850: catch(exc) {  
 851: app.logger.log("[ERRO] Excecao em setCamposDerivados; Detalhes: ");  
 852: app.logger.log(exc);  
 853: }  
 854:   
 855: var now = new Date();  
 856: app.setAtributo('dataFimPesq', util.getTimeDefaultFormated(now));  
 857: var duracao = util.getTimeUnixTimestamp(now) - app.getAtributo('dataIniPesqAbsolute');  
 858: app.setAtributo('duracaoPesq', duracao);  
 859: },  
 860:  
 861: finalizaRegistro : function(cb) {  
 862: app.logger.log('Finalizando registro: ' + registro.id);  
 863: app.setAtributo('cancelado', 0);  
 864: app.setCamposDerivados();  
 865: app.setAtributo('finalizado', 1); //para facilitar a recuperação do registro final pelo log  
 866: $.mobile.loading("show");  
 867: app.openDB(function(){  
 868: app.inserirRegistro(function(){  
 869: $.mobile.loading('hide');  
 870: app.database.close(function(){  
 871: app.logger.log("Conexão com o banco fechada");  
 872: cb();  
 873: });//fechando a conexão com o banco depois de inserir o registro;  
 874:   
 875: });  
 876: });  
 877: },  
 878:   
 879: inserirRegistro : function(cb){ //função é chamada por finalizaRegistro  
 880: myDb.insertRegistro(registro,  
 881: // erro  
 882: function(error) {  
 883: app.logger.log('Erro ao inserir registro: ' + error.message);  
 884: // confirma se tenta outra vez  
 885: confirm("Houve uma falha ao inserir o registro.\nDeseja tentar novamente?",  
 886: // button ok  
 887: function() {  
 888: app.inserirRegistro(cb);  
 889: },  
 890: // button cancel  
 891: function() {  
 892: app.logger.log("Descartando registro: " + registro.id);  
 893: if (cb != null) {  
 894: cb();  
 895: }  
 896: }, "Falha na gravação.", // título  
 897: "Sim", "Não (irá descartar o registro)");  
 898: },  
 899: // ok  
 900: function() {  
 901: app.logger.log('Registro finalizado: ' + registro.id);  
 902: app.limpaRegistro();  
 903:   
 904: alert("Entrevista registrada.");  
 905: if (util.isFunction(cb)) {  
 906: cb();  
 907: }  
 908: });  
 909: },  
 910:   
 911: buscaDuracoesRegistros : function(cb) {  
 912: app.openDB(function(){  
 913: myDb.selectDuracoesDiaRegistro(  
 914: // erro  
 915: function(error) {  
 916: app.logger.log('Erro ao buscar duracoes: ' + error.message);  
 917: // confirma se tenta outra vez  
 918: confirm("Houve uma falha ao buscar os registros.\nDeseja tentar novamente?",  
 919: // button ok  
 920: function() {  
 921: app.buscaDuracoesRegistros();  
 922: },  
 923: // button cancel  
 924: function() {  
 925: app.logger.log("buscaDuracoesRegistros cancelado");  
 926: app.database.close(function(){  
 927: app.logger.log('conexão com o banco fechada');  
 928: });  
 929: }, "Falha na busca.", // título  
 930: "Sim", "Não");  
 931: },  
 932: // ok  
 933: function() {  
 934: app.logger.log('buscaDuracoesRegistros executado com sucesso');  
 935: app.database.close(function(){  
 936: app.logger.log('conexão com o banco fechada');  
 937: if(util.isFunction(cb)){  
 938: cb();  
 939: }  
 940: });  
 941:   
 942: });  
 943: });  
 944:   
 945: },  
 946:  
 947: buscaUltimaPesquisa : function(cb) {  
 948: app.openDB(function(){  
 949: myDb.selectUltimaPesquisaValida(  
 950: // erro  
 951: function(error) {  
 952: app.logger.log('Erro ao buscar última pesquisa: ' + error.message);  
 953: // confirma se tenta outra vez  
 954: confirm("Houve uma falha ao buscar a última pesquisa realizada.\nDeseja tentar novamente?",  
 955: // button ok  
 956: function() {  
 957: app.buscaUltimaPesquisa();  
 958: },  
 959: // button cancel  
 960: function() {  
 961: app.logger.log("buscaUltimaPesquisa cancelado");  
 962: app.database.close(function(){  
 963: app.logger.log('conexão com o banco fechada');  
 964: });  
 965: }, "Falha na busca.", // título  
 966: "Sim", "Não");  
 967: },  
 968: // ok  
 969: function() {  
 970: app.logger.log('buscaUltimaPesquisa executado com sucesso');  
 971: app.database.close(function(){  
 972: app.logger.log('Conexão com o banco fechada.');  
 973: if(util.isFunction(cb)){  
 974: cb();  
 975: }  
 976: });  
 977: });  
 978: });  
 979:   
 980: },  
 981:  
 982: buscaRegistrosCancelados : function(cb) {  
 983: app.openDB(function(){  
 984: myDb.selectRegistrosCancelados(  
 985: // erro  
 986: function(error) {  
 987: app.logger.log('Erro ao buscar cancelados: ' + error.message);  
 988: // confirma se tenta outra vez  
 989: confirm("Houve uma falha ao buscar os registros cancelados.\nDeseja tentar novamente?",  
 990: // button ok  
 991: function() {  
 992: app.buscaRegistrosCancelados();  
 993: },  
 994: // button cancel  
 995: function() {  
 996: app.logger.log("buscaRegistrosCancelados cancelado");  
 997: app.database.close(function(){  
 998: app.logger.log('conexão com o banco fechada');  
 999: });  
1000: }, "Falha na busca.", // título  
1001: "Sim", "Não");  
1002: },  
1003: // ok  
1004: function() {  
1005: app.logger.log('buscaRegistrosCancelados executado com sucesso');  
1006: app.database.close(function(){  
1007: app.logger.log('Conexão com o banco fechada');  
1008: if(util.isFunction(cb)){  
1009: cb();  
1010: }  
1011: });  
1012:   
1013: });   
1014: });  
1015:   
1016: },  
1017:  
1018: cancelaRegistro : function(cb) {  
1019: app.logger.log('Cancelando registro: ' + registro.id);  
1020: app.setAtributo('cancelado', 1);  
1021: app.setCamposDerivados();  
1022: app.openDB(function(){  
1023: myDb.insertRegistro(registro, function(error) {  
1024: app.logger.log(error.message);  
1025: // confirma se tenta outra vez  
1026: confirm("Houve uma falha ao registrar o cancelamento.\nDeseja tentar novamente?",  
1027: // button ok  
1028: function() {  
1029: app.cancelaRegistro(cb);  
1030: },  
1031: // button cancel  
1032: function() {  
1033: app.logger.log("Descartando registro: " + registro.id);  
1034: if (cb != null) {  
1035: cb();  
1036: }  
1037:  
1038: }, "Falha na gravação.", // título  
1039: "Sim", "Não (irá descartar o registro)");  
1040: }, function() {  
1041: app.logger.log('Registro cancelado: ' + registro.id);  
1042: app.limpaRegistro();  
1043: app.database.close(function(){  
1044: app.logger.log("Conexão com o banco fechada.");  
1045: });  
1046: alert("Entrevista cancelada.");  
1047: if (cb != null) {  
1048: cb();  
1049: }  
1050: });  
1051: });  
1052:   
1053: },  
1054:  
1055: /\*\*  
1056: \* Remove a file from fs, if it exists  
1057: \*   
1058: \* @param String  
1059: \* fileName  
1060: \* @param String  
1061: \* dir -> prefers cordova.file.{dir}  
1062: \* @param function  
1063: \* cbSuccess success callback  
1064: \* @param function  
1065: \* cbFail error callback  
1066: \*/  
1067: removeFile : function(fileName, dirURI, cbSuccess, cbFail) {  
1068: window.resolveLocalFileSystemURL(dirURI,  
1069: // success  
1070: function(folder) {  
1071: app.logger.log('Removendo arquivo ' + fileName + ' do folder ' + JSON.stringify(folder));  
1072: folder.getFile(fileName, {  
1073: create : false  
1074: }, function(arquivo) {  
1075: arquivo.remove(function() {  
1076: app.logger.log('Arquivo ' + fileName + ' removido');  
1077: if (util.isFunction(cbSuccess)) {  
1078: cbSuccess();  
1079: }  
1080: });  
1081: }, function(err) {  
1082: app.logger.log('Arquivo ' + fileName + ' não existe no folder ' + JSON.stringify(folder));  
1083: if (util.isFunction(cbFail)) {  
1084: cbFail();  
1085: }  
1086:  
1087: });  
1088: },  
1089: // fail  
1090: function(err) {  
1091: app.logger.log('Erro resolvendo o diretório ' + dir + JSON.stringify(err));  
1092: });  
1093: },  
1094:  
1095: /\*\*  
1096: \* Copy a file overwriting the destination file, if it exists  
1097: \*   
1098: \* @param String  
1099: \* fileName  
1100: \* @param String  
1101: \* originDir  
1102: \* @param String  
1103: \* destDir  
1104: \* @param Boolean  
1105: \* isCorrupted  
1106: \* @param Function  
1107: \* cb success callback function  
1108: \*/  
1109: copyFile : function(fileName, originDirURI, destDirURI, isCorrupted, cb) {  
1110: var now = new Date();  
1111: var newName = (isCorrupted)? "CORROMPIDO" + ipadID.id + "\_" + fileName : ipadID.id + "\_" + fileName;  
1112: var extension = "";  
1113: if(newName.lastIndexOf(".txt") > -1) {  
1114: newName = newName.replace(".txt","");  
1115: extension = ".txt";  
1116: }  
1117: else if (newName.lastIndexOf(".db") > -1) {  
1118: newName = newName.replace(".db","");  
1119: extension = ".db";  
1120: }  
1121: newName = newName + "\_" + util.getTimeToExportFormated(now) + extension;  
1122: // get original dir  
1123: resolveLocalFileSystemURL(originDirURI,  
1124: // success  
1125: function(dir) {  
1126: // get original file  
1127: dir.getFile(fileName, {},  
1128: // success getting original file  
1129: function(file) {  
1130: // resolving destination  
1131: resolveLocalFileSystemURL(destDirURI,  
1132: // success resolving destination  
1133: function(destDir) {  
1134: app.logger.log('Copiando o arquivo ' + newName + ' do folder ' + dir.nativeURL + ' para o folder '  
1135: + destDir.nativeURL);  
1136: //copying new file  
1137: realCopier(file, newName, destDir);  
1138:   
1139: // Deixou de sobrescrever arquivo a partir da Fase 2  
1140: /\*  
1141: // removing destination file  
1142: app.removeFile(newName, destDirURI, function() {  
1143: realCopier(file, newName, destDir);  
1144: },  
1145: // mesmo se não conseguir remover  
1146: function() {   
1147: realCopier(file, newName, destDir);  
1148: });  
1149: \*/  
1150: }, function(err) {  
1151: app.logger.log('Erro acessando o folder de destino ' + destDir + ' ' + JSON.stringify(err));  
1152: });  
1153:  
1154: }, function(err) {  
1155: app.logger.log('Erro acessando arquivo original ' + fileName + ' ' + JSON.stringify(err));  
1156: });  
1157:  
1158: }, function(err) {  
1159: app.logger.log('Erro acessando o folder original ' + originDirURI + ' ' + JSON.stringify(err));  
1160: });  
1161: // internal function  
1162: function realCopier(f, newName, d) {  
1163: f.copyTo(d, newName, function() {  
1164: // success  
1165: app.logger.log('Arquivo ' + newName + ' copiado.');  
1166: if (util.isFunction(cb)) {  
1167: cb(newName);  
1168: }  
1169: }, function(err) {  
1170: app.logger.log('Erro copiando arquivo ' + newName + ' ' + JSON.stringify(err));  
1171:  
1172: });  
1173: }  
1174: ;  
1175: },  
1176:  
1177: restart : function(){  
1178: window.location.href = app.baseUrl + 'index.html'; //href para a página inicial  
1179: window.setTimeout(function(){  
1180: window.location.reload();  
1181: },1000);  
1182: },  
1183:   
1184: baseUrl : null,  
1185:  
1186: logFileName : "logOD.txt",  
1187:  
1188: dbName : "dadosOD.db",  
1189:  
1190: jsonName : "dadosOD.json",  
1191:  
1192: user\_login : null,  
1193:  
1194: senha\_login : null,  
1195:  
1196: posto : null,  
1197:  
1198: sentido : null,  
1199:   
1200: isTreinamento : false,  
1201:  
1202: filePaths : null, // { externalFolder : null, dbFolder : null, }  
1203:   
1204: changesCounter : 0,  
1205:   
1206: sumario\_lista : [],  
1207:   
1208: sumario\_lista\_cancelados : [],  
1209:   
1210: ultima\_pesquisa : [],  
1211:   
1212: logger : window.console //initial value, changed to use logger.js on file system initialization  
1213: ,  
1214:  
1215:}; // end of app  
1216:  
1217:// Registro do momento  
1218:var registro;  
1219:  
1220:$(document).ready(function() {  
1221:   
1222: //captura todos os erros   
1223: window.onerror = function(msg, url, line, col, error) {  
1224: // Note that col & error are new to the HTML 5 spec and may not be   
1225: // supported in every browser. It worked for me in Chrome.  
1226: var extra = !col ? '' : '\ncolumn: ' + col;  
1227: extra += !error ? '' : '\nerror: ' + error;  
1228:   
1229: var mesageString = "Error: " + msg + "\nurl: " + url + "\nline: " + line + extra;  
1230:  
1231: if (app == null){  
1232: console.log(mesageString);  
1233: } else {  
1234: app.logger.log(mesageString);  
1235: }  
1236:  
1237: // You can view the information in an alert to see things working like this:  
1238: if (msg.indexOf("TypeError: null is not an object (evaluating 'input.val().replace')")>-1){  
1239: //ignores  
1240: } else if (msg.indexOf("SecurityError: DOM Exception 18")>-1){ //app freezes  
1241:   
1242: if (app != null){  
1243: alert("Ocorreu uma condição que impede \no prosseguimento do programa.\nO aplicativo será reiniciado",  
1244: 'Erro fatal.',  
1245: app.restart,  
1246: 'error');  
1247: } else {  
1248: alert("Ocorreu uma condição que impede \no prosseguimento do programa.\nO aplicativo será reiniciado");  
1249: window.location.reload()  
1250: }  
1251:   
1252: } else {  
1253: alert("Ocorreu uma condição inesperada anote \n a mensagem abaixo\n" + mesageString);  
1254: }  
1255:   
1256:   
1257: var suppressErrorAlert = true;  
1258: // If you return true, then error alerts (like in older versions of   
1259: // Internet Explorer) will be suppressed.  
1260: return suppressErrorAlert;  
1261: };  
1262: insert\_controllers.insert();  
1263: app.initialize();  
1264:});

## .\mobile\platforms\ios\www\js\logins.js

1:var logins = {  
 2:  
 3: autenticaMaster : function(usuario, senha) {  
 4: if ((usuario == logins.user\_master.usr) && (senha == logins.user\_master.pwd)) {  
 5: return true;  
 6: }  
 7: return false;  
 8: },  
 9:  
 10: autentica : function(usuario, senha) {  
 11: var posto = null;  
 12: var msg = null;  
 13: if ((logins.user\_tester != undefined) && (logins.user\_tester.usr != undefined) && (usuario == logins.user\_tester.usr)  
 14: && (senha == logins.user\_tester.pwd)) {  
 15: logins.user\_logado = logins.user\_tester;  
 16: return true;  
 17: } else if (String(usuario).length == 3 && logins.verificaPostoSenha(String(usuario), senha)) {  
 18: if(datas.verificaData()){  
 19: app.isTreinamento = false;  
 20: app.logger.log("Sistema em modo de produção.");  
 21: return true;  
 22: }else{  
 23: msg = "Logins de produção não funcionam fora das datas da pesquisa real!";  
 24: alert(msg ,"Erro!",null,'error');  
 25: app.logger.log(msg);  
 26: return false;  
 27: }  
 28:   
 29: //treinamento  
 30: } else if (String(usuario).length == 4 && String(usuario).substr(3).toUpperCase() == 'T'){  
 31: if (datas.verificaData()){  
 32: msg = "Logins de treinamento não funcionam em datas da pesquisa real!";  
 33: alert(msg ,"Erro!",null,'error');  
 34: app.logger.log(msg);  
 35: return false;  
 36: }  
 37: posto = String(usuario).substr(0,3);  
 38: if (logins.verificaPostoSenha(posto, senha)){  
 39: app.isTreinamento = true;  
 40: msg = "O sistema funcionará em modo de treinamento!";  
 41: alert(msg,null,null,'error');  
 42: app.logger.log(msg);  
 43: return true;  
 44: }  
 45: }  
 46: msg = "Usuário e/ou Senha informados não estão cadastrados no sistema!";  
 47: alert(msg);  
 48: app.logger.log(msg);  
 49: return false;  
 50: },  
 51:   
 52: verificaPostoSenha : function(posto, senha){  
 53: if (posto.length == 3 && Number(posto) > 0) {  
 54: var i = Number(posto) - 1;  
 55: // for (i = 0; i < logins.users.length; i++) {  
 56: /\* comentar o 'for' se o sequencial do posto for igual ao posicionamento dele na lista. \*/  
 57: var login = logins.users[i];  
 58: if ((posto == login.usr) && (senha == login.pwd)) {  
 59: logins.user\_logado = login;  
 60: return true;  
 61: }  
 62: // }  
 63: }  
 64: return false;  
 65: },  
 66:   
 67: user\_master : { // Usuário exclusivo para configurar o idIpad na instalação, ele não faz login no aplicativo.  
 68: usr : 'Master', // usuário mestre  
 69: pwd : "434723" // senha mestra (ver teclado numérico para a string 'idIPad')  
 70: },  
 71:  
 72: user\_tester : { // Usuário destinado aos testes do aplicativo.  
 73: // Antes de ir para produção, basta comentar as duas linhas abaixo se quiser inativar para esse ambiente.  
 74: usr : 'Teste', // usuário de teste  
 75: pwd : "837837", // senha de teste (ver teclado numérico para a string 'tester')  
 76: perguntaExtra : true  
 77: },  
 78:   
 79: // Postos com pergunta extra na 1ª fase: 211  
 80: // Postos com pergunta extra na 2ª fase: 020, 184, 185   
 81: users : [  
 82: // usr = número do posto  
 83: // pwd = 4 últimos números do SNV  
 84: {  
 85: usr : '001',  
 86: pwd : '1590',  
 87: fase : '3',  
 88: ladoA : '',  
 89: ladoB : '',  
 90: trecho : 'ENTR AC-463/465 (BOM FUTURO) - ENTR BR-317'  
 91: }, {  
 92: usr : '002',  
 93: pwd : '0808',  
 94: fase : '1',  
 95: ladoA : 'LUCAS DO RIO VERDE',  
 96: ladoB : 'INÍCIO DA TRAVESSIA URBANA DE SORRISO',  
 97: trecho : 'LUCAS DO RIO VERDE – INÍCIO DA TRAVESSIA URBANA DE SORRISO'  
 98: }, {  
 99: usr : '003',  
 100: pwd : '0630',  
 101: fase : '2',  
 102: ladoA : '',  
 103: ladoB : '',  
 104: trecho : 'ENTR BR-104(A) - ENTR BR-104(B)'  
 105: }, {  
 106: usr : '004',  
 107: pwd : '0890',  
 108: fase : '2',  
 109: ladoA : '',  
 110: ladoB : '',  
 111: trecho : 'ENTR AL-225(A) - ENTR AL-225(B) (P/PORTO REAL DO COLÉGIO)'  
 112: }, {  
 113: usr : '005',  
 114: pwd : '0950',  
 115: fase : '3',  
 116: ladoA : '',  
 117: ladoB : '',  
 118: trecho : 'ENTR AL-115(A) - ENTR AL-115(B) (PALMEIRA DOS ÍNDIOS)'  
 119: }, {  
 120: usr : '006',  
 121: pwd : '0750',  
 122: fase : '4',  
 123: ladoA : '',  
 124: ladoB : '',  
 125: trecho : 'ENTR BR-101(A) - ENTR BR-101(B) (P/ALAGOINHAS)'  
 126: }, {  
 127: usr : '007',  
 128: pwd : '1810',  
 129: fase : '3',  
 130: ladoA : '',  
 131: ladoB : '',  
 132: trecho : 'ENTR BA-120(A)/262(B) (P/ITAJUÍPE) - ENTR BR-415/BA-120(B) (ITABUNA)'  
 133: }, {  
 134: usr : '008',  
 135: pwd : '0330',  
 136: fase : '2',  
 137: ladoA : '',  
 138: ladoB : '',  
 139: trecho : 'ENTR BR-324(A) - ENTR BR-324(B)/BA-502/503 (FEIRA DE SANTANA)'  
 140: }, {  
 141: usr : '009',  
 142: pwd : '0934',  
 143: fase : '3',  
 144: ladoA : '',  
 145: ladoB : '',  
 146: trecho : 'ENTR BA-643 (PLANALTO) - ENTR BA-641'  
 147: }, {  
 148: usr : '010',  
 149: pwd : '0365',  
 150: fase : '3',  
 151: ladoA : '',  
 152: ladoB : '',  
 153: trecho : 'ENTR BR-242(A) - ENTR BR-330/BA-148 (SEABRA)'  
 154: }, {  
 155: usr : '011',  
 156: pwd : '0300',  
 157: fase : '2',  
 158: ladoA : '',  
 159: ladoB : '',  
 160: trecho : 'JAGUARARI - ENTR BA-220 (SENHOR DO BONFIM)'  
 161: }, {  
 162: usr : '012',  
 163: pwd : '0530',  
 164: fase : '3',  
 165: ladoA : '',  
 166: ladoB : '',  
 167: trecho : 'ENTR CE-187 (P/MARRECOS) - ENTR BR-404/CE-176/187/363 (P/TAUÁ)'  
 168: }, {  
 169: usr : '013',  
 170: pwd : '0090',  
 171: fase : '3',  
 172: ladoA : '',  
 173: ladoB : '',  
 174: trecho : 'ENTR CE-253 (PACAJÚS) - ENTR BR-122(A)/CE-354 (CHORÓZINHO)'  
 175: }, {  
 176: usr : '014',  
 177: pwd : '0260',  
 178: fase : '2',  
 179: ladoA : '',  
 180: ladoB : '',  
 181: trecho : 'ENTR BR-226(B)/CE-275(B) (P/PEREIRO) - ENTR BR-404/434 (ICÓ)'  
 182: }, {  
 183: usr : '015',  
 184: pwd : '2195',  
 185: fase : '2',  
 186: ladoA : '',  
 187: ladoB : '',  
 188: trecho : 'ENTR ES-245(B) - ENTR ES-440'  
 189: }, {  
 190: usr : '016',  
 191: pwd : '2470',  
 192: fase : '3',  
 193: ladoA : '',  
 194: ladoB : '',  
 195: trecho : 'ENTR ES-375(A) (P/PIÚMA) - ENTR ES-375(B) (ICONHA)'  
 196: }, {  
 197: usr : '017',  
 198: pwd : '0070',  
 199: fase : '3',  
 200: ladoA : '',  
 201: ladoB : '',  
 202: trecho : 'ENTR BR-101(B) - ENTR ES-465 (P/DOMINGOS MARTINS)'  
 203: }, {  
 204: usr : '018',  
 205: pwd : '0015',  
 206: fase : '2',  
 207: ladoA : '',  
 208: ladoB : '',  
 209: trecho : 'KM 15,3 - ENTR BR-484 (P/PONTE SOBRE RIO DOCE)'  
 210: }, {  
 211: usr : '019',  
 212: pwd : '0110',  
 213: fase : '2',  
 214: ladoA : '',  
 215: ladoB : '',  
 216: trecho : 'ENTR GO-116 (P/FORMOSA) - ENTR GO-346 (P/CABEÇEIRAS)',  
 217: perguntaExtra : true // Possui pergunta extra na 4ª fase  
 218: }, {  
 219: usr : '020',  
 220: pwd : '0090',  
 221: fase : '2',  
 222: ladoA : '',  
 223: ladoB : '',  
 224: trecho : 'ENTR BR-457(B)/GO-219 - ENTR GO-020(A)'  
 225: // perguntaExtra : true // Possuiu pergunta extra na 2ª fase  
 226: }, {  
 227: usr : '021',  
 228: pwd : '0112',  
 229: fase : '4',  
 230: ladoA : '',  
 231: ladoB : '',  
 232: trecho : 'ENTR GO-338 (ABADIÂNIA - FIM TRAVESSIA URBANA) - ENTR BR-153(A) (P/ANÁPOLIS)'  
 233: }, {  
 234: usr : '022',  
 235: pwd : '0340',  
 236: fase : '3',  
 237: ladoA : '',  
 238: ladoB : '',  
 239: trecho : 'ENTR GO-353(A) (LINDA VISTA) - ENTR BR-414(A)/GO-151/244/353(B) (PORANGATU)'  
 240: }, {  
 241: usr : '023',  
 242: pwd : '0770',  
 243: fase : '2',  
 244: ladoA : '',  
 245: ladoB : '',  
 246: trecho : 'ENTR BR-154/452(A)/483 - ENTR BR-452(B) (DIV GO/MG) (ITUMBIARA)'  
 247: }, {  
 248: usr : '024',  
 249: pwd : '0510',  
 250: fase : '2',  
 251: ladoA : '',  
 252: ladoB : '',  
 253: trecho : 'ENTR BR-060(B)/158(B)/GO-184(B) (P/ESTREITO) - ENTR GO-050(B)'  
 254: }, {  
 255: usr : '025',  
 256: pwd : '0050',  
 257: fase : '2',  
 258: ladoA : '',  
 259: ladoB : '',  
 260: trecho : 'FIM DA DUPLICAÇÃO - ENTR BR-402/MA-110 (BACABEIRA)'  
 261: }, {  
 262: usr : '026',  
 263: pwd : '0150',  
 264: fase : '3',  
 265: ladoA : '',  
 266: ladoB : '',  
 267: trecho : 'ENTR BR-316(A) (CACHUCHA) - ENTR BR-316(B) (PERITORÓ)'  
 268: }, {  
 269: usr : '027',  
 270: pwd : '0680',  
 271: fase : '3',  
 272: ladoA : '',  
 273: ladoB : '',  
 274: trecho : 'RIO PINDARÉ - ENTR BR-010(A) (AÇAILÂNDIA)'  
 275: }, {  
 276: usr : '028',  
 277: pwd : '0670',  
 278: fase : '2',  
 279: ladoA : '',  
 280: ladoB : '',  
 281: trecho : 'ENTR BR-251(A) (CANACÍ) - ENTR BR-135/251(B)/308/365 (MONTES CLAROS)'  
 282: }, {  
 283: usr : '029',  
 284: pwd : '0370',  
 285: fase : '3',  
 286: ladoA : '',  
 287: ladoB : '',  
 288: trecho : 'ENTR MA-026 (DEZESSETE) - ENTR MA-034(A)/127/349 (CAXIAS)'  
 289: }, {  
 290: usr : '030',  
 291: pwd : '0130',  
 292: fase : '2',  
 293: ladoA : '',  
 294: ladoB : '',  
 295: trecho : 'ENTR MG-410 (P/PONTO DIAMANTE) - ENTR MG-181 (JOÃO PINHEIRO)'  
 296: }, {  
 297: usr : '031',  
 298: pwd : '0330',  
 299: fase : '3',  
 300: ladoA : '',  
 301: ladoB : '',  
 302: trecho : 'ENTR MG-238 (P/SETE LAGOAS) - ENTR MG-432 (P/ESMERALDAS)'  
 303: }, {  
 304: usr : '032',  
 305: pwd : '0040',  
 306: fase : '2',  
 307: ladoA : '',  
 308: ladoB : '',  
 309: trecho : 'ENTR BR-040(A) (ANEL RODOVIÁRIO DE BELO HORIZONTE) - ENTR BR-040(B)'  
 310: }, {  
 311: usr : '033',  
 312: pwd : '0570',  
 313: fase : '3',  
 314: ladoA : '',  
 315: ladoB : '',  
 316: trecho : 'ENTR BR-499 (SANTOS DUMONT) - ENTR ANT UNIÃO E INDÚSTRIA (B. TRIUNFO)'  
 317: }, {  
 318: usr : '034',  
 319: pwd : '0250',  
 320: fase : '4',  
 321: ladoA : '',  
 322: ladoB : '',  
 323: trecho : 'ENTR BR-365(B)/452(B) - RIO TIJUCO'  
 324: }, {  
 325: usr : '035',  
 326: pwd : '1130',  
 327: fase : '2',  
 328: ladoA : '',  
 329: ladoB : '',  
 330: trecho : 'ENTR BR-342(B)/418/MG-217 (RIB STO ANTÔNIO) (TEÓFILO OTONI) - ACESSO ITAMBACURI'  
 331: }, {  
 332: usr : '036',  
 333: pwd : '1280',  
 334: fase : '4',  
 335: ladoA : '',  
 336: ladoB : '',  
 337: trecho : 'FIM PISTA DUPLA - DOM CORRÊA'  
 338: }, {  
 339: usr : '037',  
 340: pwd : '1430',  
 341: fase : '3',  
 342: ladoA : '',  
 343: ladoB : '',  
 344: trecho : 'ENTR BR-116(A)/120 (LEOPOLDINA) - ENTR BR-116(B)'  
 345: }, {  
 346: usr : '038',  
 347: pwd : '0850',  
 348: fase : '3',  
 349: ladoA : '',  
 350: ladoB : '',  
 351: trecho : 'ENTR BR-259(B) (CURVELO) - ENTR BR-040(A)'  
 352: }, {  
 353: usr : '039',  
 354: pwd : '0290',  
 355: fase : '2',  
 356: ladoA : '',  
 357: ladoB : '',  
 358: trecho : 'ENTR BR-364(A)/262(A) (P/COMENDADOR GOMES) - ENTR BR-364(B)/262(B) (P/FRUTAL)'  
 359: }, {  
 360: usr : '040',  
 361: pwd : '0890',  
 362: fase : '4',  
 363: ladoA : '',  
 364: ladoB : '',  
 365: trecho : 'ENTR MG-187 (P/IBIÁ) - ENTR BR-146 (P/ARAXÁ)',  
 366: perguntaExtra : true // Possui pergunta extra na 4ª fase  
 367: }, {  
 368: usr : '041',  
 369: pwd : '0230',  
 370: fase : '3',  
 371: ladoA : '',  
 372: ladoB : '',  
 373: trecho : 'ENTR MG-451 (ITUTINGA) - ENTR BR-354 (LAVRAS)'  
 374: }, {  
 375: usr : '042',  
 376: pwd : '0215',  
 377: fase : '4',  
 378: ladoA : '',  
 379: ladoB : '',  
 380: trecho : 'ACESSO CARMO DO PARANAIBA - ENTR MG-230'  
 381: }, {  
 382: usr : '043',  
 383: pwd : '0100',  
 384: fase : '3',  
 385: ladoA : '',  
 386: ladoB : '',  
 387: trecho : 'PRESIDENTE OLEGÁRIO - ENTR BR-146(A)/365 (PATOS DE MINAS)'  
 388: }, {  
 389: usr : '044',  
 390: pwd : '0370',  
 391: fase : '3',  
 392: ladoA : '',  
 393: ladoB : '',  
 394: trecho : 'ENTR BR-153 - ENTR BR-154(A)'  
 395: }, {  
 396: usr : '045',  
 397: pwd : '0230',  
 398: fase : '2',  
 399: ladoA : '',  
 400: ladoB : '',  
 401: trecho : 'ENTR MG-425 (CORONEL FABRICIANO) - ENTR MG-320 (P/JAGUARAÇU)'  
 402: }, {  
 403: usr : '046',  
 404: pwd : '0490',  
 405: fase : '2',  
 406: ladoA : '',  
 407: ladoB : '',  
 408: trecho : 'ENTR BR-262(B) (BETIM) - ENTR MG-155'  
 409: }, {  
 410: usr : '047',  
 411: pwd : '0750',  
 412: fase : '3',  
 413: ladoA : '',  
 414: ladoB : '',  
 415: trecho : 'ENTR MG-458 (CAREAÇU) - ENTR BR-459 (P/POUSO ALEGRE)'  
 416: }, {  
 417: usr : '048',  
 418: pwd : '0324',  
 419: fase : '1',  
 420: ladoA : 'VILA SÃO PEDRO',  
 421: ladoB : 'ACESSO COLÔNIA AGRÍCOLA SEXTA LINHA',  
 422: trecho : 'ACESSO COLÂNIA AGRICOLA SEXTA LINHA - FINAL PISTA DUPLA (VILA VARGAS)'  
 423: }, {  
 424: usr : '049',  
 425: pwd : '0450',  
 426: fase : '1',  
 427: ladoA : 'ENTR MS-441(B) (P/BANDEIRANTES)',  
 428: ladoB : 'ENTR MS-244 (P/BONFIM)',  
 429: trecho : 'ENTR MS-441(B) (P/BANDEIRANTES) - ENTR MS-244 (P/BONFIM)'  
 430: }, {  
 431: usr : '050',  
 432: pwd : '1290',  
 433: fase : '2',  
 434: ladoA : '',  
 435: ladoB : '',  
 436: trecho : 'ENTR BR-158(B)/MS-395 (P/BRASILÂNDIA) - ENTR MS-453/459 (P/ARAPUÁ)'  
 437: }, {  
 438: usr : '051',  
 439: pwd : '0880',  
 440: fase : '2',  
 441: ladoA : '',  
 442: ladoB : '',  
 443: trecho : 'FIM TRAV RIO PARANÁ (PONTE M. JOPPERT) - INCÍO DA PISTA DUPLA'  
 444: }, {  
 445: usr : '052',  
 446: pwd : '0030',  
 447: fase : '1',  
 448: ladoA : 'ENTR BR-070(B)',  
 449: ladoB : 'ENTR MT-170 (CARAMÚJO)',  
 450: trecho : 'ENTR BR-070(B) - ENTR MT-170 (CARAMÚJO)'  
 451: }, {  
 452: usr : '053',  
 453: pwd : '0592',  
 454: fase : '1',  
 455: ladoA : 'ENTR MT-270(B)',  
 456: ladoB : 'ENTR MT-469(A)',  
 457: trecho : 'ENTR MT-270(B) - ENTR MT-469(A)'  
 458: }, {  
 459: usr : '054',  
 460: pwd : '0840',  
 461: fase : '3',  
 462: ladoA : '',  
 463: ladoB : '',  
 464: trecho : 'JANGADA - ENTR MT-246(B)'  
 465: }, {  
 466: usr : '055',  
 467: pwd : '0510',  
 468: fase : '3',  
 469: ladoA : '',  
 470: ladoB : '',  
 471: trecho : 'ENTR BR-222(B)/PA-332 (DOM ELISEU) - ENTR PA-125/263 (GURUPIZINHO)'  
 472: }, {  
 473: usr : '056',  
 474: pwd : '0335',  
 475: fase : '3',  
 476: ladoA : '',  
 477: ladoB : '',  
 478: trecho : 'ENTR PB-018 (P/CONDE) - ENTR PB-034'  
 479: }, {  
 480: usr : '057',  
 481: pwd : '0010',  
 482: fase : '4',  
 483: ladoA : '',  
 484: ladoB : '',  
 485: trecho : 'ENTR PB-100 (P/GALANTE) - ENTR BR-104(A)/408(B)/PB-095 (CAMPINA GRANDE)'  
 486: }, {  
 487: usr : '058',  
 488: pwd : '0510',  
 489: fase : '3',  
 490: ladoA : '',  
 491: ladoB : '',  
 492: trecho : 'ENTR PE-063 (P/AMARAJI) - ENTR PE-064/085 (RIBEIRÃO)'  
 493: }, {  
 494: usr : '059',  
 495: pwd : '0460',  
 496: fase : '3',  
 497: ladoA : '',  
 498: ladoB : '',  
 499: trecho : 'ENTR PE-460 - ENTR BR-316/428 (P/CABROBÓ)'  
 500: }, {  
 501: usr : '060',  
 502: pwd : '0140',  
 503: fase : '3',  
 504: ladoA : '',  
 505: ladoB : '',  
 506: trecho : 'ENTR PE-103 (P/BONITO) - ENTR BR-104/423(A) (CARUARÚ)'  
 507: }, {  
 508: usr : '061',  
 509: pwd : '0420',  
 510: fase : '4',  
 511: ladoA : '',  
 512: ladoB : '',  
 513: trecho : 'ENTR PE-450 (P/VERDEJANTE) - ENTR BR-116/361 (SALGUEIRO)'  
 514: }, {  
 515: usr : '062',  
 516: pwd : '0370',  
 517: fase : '3',  
 518: ladoA : '',  
 519: ladoB : '',  
 520: trecho : 'ENTR BR-428(A) (LAGOA GRANDE) - ENTR BR-235/407/423/428(B) (DIV PE/BA) (PETROLINA/JUAZEIRO)'  
 521: }, {  
 522: usr : '063',  
 523: pwd : '0830',  
 524: fase : '4',  
 525: ladoA : '',  
 526: ladoB : '',  
 527: trecho : 'ENTR PI-242 - ENTR BR-407(A)/PI-238/245(A) (PICOS)'  
 528: }, {  
 529: usr : '064',  
 530: pwd : '0130',  
 531: fase : '3',  
 532: ladoA : '',  
 533: ladoB : '',  
 534: trecho : 'ENTR BR-404(B)/407(B) - INÍCIO PISTA DUPLA (CAPITÃO CAMPOS)'  
 535: }, {  
 536: usr : '065',  
 537: pwd : '0420',  
 538: fase : '3',  
 539: ladoA : '',  
 540: ladoB : '',  
 541: trecho : 'DEMERVAL LOBÃO - ENTR PI-223 (MONSENHOR GIL)'  
 542: }, {  
 543: usr : '066',  
 544: pwd : '2710',  
 545: fase : '1',  
 546: ladoA : 'DIV SP/PR (CAB NORTE PONTE S/ RIO PARDINHO)',  
 547: ladoB : 'INÍCIO VARIANTE DO ALPINO(PISTA DIREITA)',  
 548: trecho : 'DIV SP/PR (CAB NORTE PONTE S/ RIO PARDINHO) - INÍCIO VARIANTE DO ALPINO(PISTA DIREITA)'  
 549: }, {  
 550: usr : '067',  
 551: pwd : '2810',  
 552: fase : '1',  
 553: ladoA : 'ENTR PR-427 (CAMPO DO TENENTE)',  
 554: ladoB : 'DIV PR/SC (RIO NEGRO/MAFRA)',  
 555: trecho : 'ENTR PR-427 (CAMPO DO TENENTE) - DIV PR/SC (RIO NEGRO/MAFRA)'  
 556: }, {  
 557: usr : '068',  
 558: pwd : '0205',  
 559: fase : '1',  
 560: ladoA : 'ACESSO GUARAPUAVA',  
 561: ladoB : 'ENTR BR-466 (P/GUARAPUAVA)',  
 562: trecho : 'ACESSO GUARAPUAVA - ENTR BR-466 (P/GUARAPUAVA)'  
 563: }, {  
 564: usr : '069',  
 565: pwd : '0310',  
 566: fase : '1',  
 567: ladoA : 'ENTR PR-182 (P/CAPITÃO LEÔNIDAS MARQUES)',  
 568: ladoB : 'ENTR BR-163',  
 569: trecho : 'ACESSO OESTE CASCAVEL - ENTR BR-163/PR-182 (P/CAPITÃO LEÔNIDAS MARQUES)'  
 570: }, {  
 571: usr : '070',  
 572: pwd : '0480',  
 573: fase : '2',  
 574: ladoA : '',  
 575: ladoB : '',  
 576: trecho : 'FIM PISTA DUPLA - ENTR PR-431 (P/JACAREZINHO)'  
 577: }, {  
 578: usr : '071',  
 579: pwd : '0490',  
 580: fase : '1',  
 581: ladoA : 'ENTR PR-281 (P/TIJUCAS DO SUL)',  
 582: ladoB : 'DIV PR/SC (ENTR BR-101)',  
 583: trecho : 'ENTR PR-281 (P/TIJUCAS DO SUL) - DIV PR/SC (ENTR BR-101)'  
 584: }, {  
 585: usr : '072',  
 586: pwd : '0095',  
 587: fase : '1',  
 588: ladoA : 'ENTR PR-510/511 (P/CONTENDA)',  
 589: ladoB : 'ENTR PR-428 (LAPA)',  
 590: trecho : 'ENTR PR-510/511 (P/CONTENDA) - ENTR PR-428 (LAPA)'  
 591: }, {  
 592: usr : '073',  
 593: pwd : '0730',  
 594: fase : '2',  
 595: ladoA : '',  
 596: ladoB : '',  
 597: trecho : 'ENTR BR-393(B) (TREVO MOURA BRASIL) - ENTR BR-492(A) (TREVO P/ AREAL)'  
 598: }, {  
 599: usr : '074',  
 600: pwd : '2930',  
 601: fase : '3',  
 602: ladoA : '',  
 603: ladoB : '',  
 604: trecho : 'ENTR RJ-162 (P/RIO DOURADO) - ENTR BR-120 (CASIMIRO DE ABREU)'  
 605: }, {  
 606: usr : '075',  
 607: pwd : '2130',  
 608: fase : '2',  
 609: ladoA : '',  
 610: ladoB : '',  
 611: trecho : 'ENTR RJ-159 (FLORIANO) - ENTR RJ-161 (RESENDE)'  
 612: }, {  
 613: usr : '076',  
 614: pwd : '0310',  
 615: fase : '2',  
 616: ladoA : '',  
 617: ladoB : '',  
 618: trecho : 'ENTR BR-356(A)/393(B) - ENTR BR-356(B) (ITAPERUNA)'  
 619: }, {  
 620: usr : '077',  
 621: pwd : '0330',  
 622: fase : '3',  
 623: ladoA : '',  
 624: ladoB : '',  
 625: trecho : 'SAPUCAIA - ENTR BR-040(A)'  
 626: }, {  
 627: usr : '078',  
 628: pwd : '0167',  
 629: fase : '3',  
 630: ladoA : '',  
 631: ladoB : '',  
 632: trecho : 'ACESSO À LAGOA DO BONFIM - ENTR RN-315/002(A) (SÃO JOSÉ DE MIPIBÚ/NÍSIA FLORESTA)'  
 633: }, {  
 634: usr : '079',  
 635: pwd : '0070',  
 636: fase : '3',  
 637: ladoA : '',  
 638: ladoB : '',  
 639: trecho : 'ENTR RN-203 (P/SÃO PEDRO) - ACESSO BOM JESUS'  
 640: }, {  
 641: usr : '080',  
 642: pwd : '0270',  
 643: fase : '3',  
 644: ladoA : '',  
 645: ladoB : '',  
 646: trecho : 'ENTR BR-104(B) (LAJES) - ENTR RN-023 (CAIÇARA DO RIO DO VENTO)'  
 647: }, {  
 648: usr : '081',  
 649: pwd : '0070',  
 650: fase : '3',  
 651: ladoA : '',  
 652: ladoB : '',  
 653: trecho : 'ENTR RN-233 (P/CARAÚBAS) - ENTR RN-031/177(A)'  
 654: }, {  
 655: usr : '082',  
 656: pwd : '1210',  
 657: fase : '3',  
 658: ladoA : '',  
 659: ladoB : '',  
 660: trecho : 'ENTR BR-429(B) (JI-PARANÁ) - ENTR RO-473 (P/URUPÁ)'  
 661: }, {  
 662: usr : '083',  
 663: pwd : '3275',  
 664: fase : '3',  
 665: ladoA : '',  
 666: ladoB : '',  
 667: trecho : 'ENTR ERS-703 (GUAIBA) - ENTR ERS-709 (P/ BARRA DO RIBEIRO)'  
 668: }, {  
 669: usr : '084',  
 670: pwd : '1150',  
 671: fase : '1',  
 672: ladoA : 'ENTR BR-472 (P/PALMITINHO)',  
 673: ladoB : 'ENTR RS-585/587 (SEBERI)',  
 674: trecho : 'ENTR RSC-472 (P/ PALMITINHO) - ENTR ERS-585/587 (SEBERI)'  
 675: }, {  
 676: usr : '085',  
 677: pwd : '1310',  
 678: fase : '1',  
 679: ladoA : 'ENTR RS-348 (VAL DE SERRA)',  
 680: ladoB : 'ENTR RS-509 (P/SANTA MARIA)',  
 681: trecho : 'ENTR ERS-348 (VAL DE SERRA) - ENTR ERS-509 (P/ SANTA MARIA)'  
 682: }, {  
 683: usr : '086',  
 684: pwd : '0180',  
 685: fase : '1',  
 686: ladoA : 'ACESSO A MINAS DO LEÃO',  
 687: ladoB : 'ENTR BR-471 (PANTANO GRANDE)',  
 688: trecho : 'ACESSO A MINAS DO LEAO - ENTR BRS-471 (PANTANO GRANDE)'  
 689: }, {  
 690: usr : '087',  
 691: pwd : '1840',  
 692: fase : '1',  
 693: ladoA : 'ENTR RS-705 (P/GERIBA)',  
 694: ladoB : 'ENTR BR-290(B)',  
 695: trecho : 'ENTR ERS-705 (GERIBA) - ENTR BRS-153(B) (P/ BAGE)'  
 696: }, {  
 697: usr : '088',  
 698: pwd : '0410',  
 699: fase : '1',  
 700: ladoA : 'ENTR BR-293(A)',  
 701: ladoB : 'ENTR BR-472(A)',  
 702: trecho : 'ENTR BRS-293(A) (P/ QUARAI) - ENTR BRS-472(A) (P/ BARRA DO QUARAI)'  
 703: }, {  
 704: usr : '089',  
 705: pwd : '0130',  
 706: fase : '1',  
 707: ladoA : 'ENTR BR-473 (P/BAGÉ)',  
 708: ladoB : 'ENTR RS-630 (P/DOM PEDRITO)',  
 709: trecho : 'ENTR RSC-473 (P/ BAGE) - ENTR ERS-630/634 (P/ DOM PEDRITO)'  
 710: }, {  
 711: usr : '090',  
 712: pwd : '0070',  
 713: fase : '1',  
 714: ladoA : 'ENTR BR-471(A) (QUINTA)',  
 715: ladoB : 'ACESSO PELOTAS',  
 716: trecho : 'ENTR BRS-471(A) (QUINTA) - ACESSO SUL A PELOTAS'  
 717: }, {  
 718: usr : '091',  
 719: pwd : '0210',  
 720: fase : '1',  
 721: ladoA : 'ACESSO LESTE A ITAQUI',  
 722: ladoB : 'ENTR BR-290(A)/293(A) (URUGUAIANA)',  
 723: trecho : 'ACESSO LESTE A ITAQUI - ENTR BR-290(A)/293(A) (URUGUAIANA)'  
 724: }, {  
 725: usr : '092',  
 726: pwd : '3830',  
 727: fase : '1',  
 728: ladoA : 'GARUVA',  
 729: ladoB : 'ENTR SC-430 (PIRABEIRABA)',  
 730: trecho : 'GARUVA - ENTR SC-430 (PIRABEIRABA)'  
 731: }, {  
 732: usr : '093',  
 733: pwd : '4010',  
 734: fase : '1',  
 735: ladoA : 'BALNEÁRIO DE CAMBORIÚ',  
 736: ladoB : 'P/PORTO BELO',  
 737: trecho : 'BALNEÁRIO DE CAMBORIÚ - P/PORTO BELO'  
 738: }, {  
 739: usr : '094',  
 740: pwd : '4125',  
 741: fase : '1',  
 742: ladoA : 'PONTE SOBRE O RIO DA MADRE',  
 743: ladoB : 'PONTE SOBRE O RIO ARAÇATUBA',  
 744: trecho : 'PONTE SOBRE O RIO DA MADRE - PONTE SOBRE O RIO ARAÇATUBA'  
 745: }, {  
 746: usr : '095',  
 747: pwd : '4240',  
 748: fase : '1',  
 749: ladoA : 'ENTR BR-285(A) (ARARANGUÁ)',  
 750: ladoB : 'ENTR BR-285(B)/SC-285 (SANGA DA TOCA)',  
 751: trecho : 'ENTR BR-285(A)/SC-449 (ARARANGUÁ) - ENTR BR-285(B)/SC-285 (P/ERMO)'  
 752: }, {  
 753: usr : '096',  
 754: pwd : '2910',  
 755: fase : '2',  
 756: ladoA : '',  
 757: ladoB : '',  
 758: trecho : 'ENTR SC-352(B) (P/TAIÓ) - SÃO CRISTOVÃO DO SUL'  
 759: }, {  
 760: usr : '097',  
 761: pwd : '1590',  
 762: fase : '3',  
 763: ladoA : '',  
 764: ladoB : '',  
 765: trecho : 'ENTR SC-463 (P/JABORÁ) - ENTR BR-283 (P/CONCÓRDIA)'  
 766: }, {  
 767: usr : '098',  
 768: pwd : '0020',  
 769: fase : '3',  
 770: ladoA : '',  
 771: ladoB : '',  
 772: trecho : 'ENTR SC-114(A) (ÍNDIOS) - ENTR SC-114(B) (P/PAINEL)'  
 773: }, {  
 774: usr : '099',  
 775: pwd : '0377',  
 776: fase : '1',  
 777: ladoA : 'ENTR SC-157(B) (P/MODELO)',  
 778: ladoB : 'ENTR BR-158(A) (P/CUNHA PORÃ)',  
 779: trecho : 'ENTR SC-157(B) (P/MODELO) - ENTR BR-158(A) (P/CUNHA PORÃ)'  
 780: }, {  
 781: usr : '100',  
 782: pwd : '0190',  
 783: fase : '4',  
 784: ladoA : '',  
 785: ladoB : '',  
 786: trecho : 'ENTR SC-114(B) (P/OTACÍLIO COSTA) - ENTR BR-116'  
 787: }, {  
 788: usr : '101',  
 789: pwd : '0040',  
 790: fase : '3',  
 791: ladoA : '',  
 792: ladoB : '',  
 793: trecho : 'ENTR SE-464 (P/SÃO CRISTOVÃO) - ENTR BR-349(B)/SE-265 (ITAPORANGA D`AJUDA)'  
 794: }, {  
 795: usr : '102',  
 796: pwd : '0060',  
 797: fase : '3',  
 798: ladoA : '',  
 799: ladoB : '',  
 800: trecho : 'ENTR SE-160 - ENTR SE-170(A)'  
 801: }, {  
 802: usr : '103',  
 803: pwd : '2410',  
 804: fase : '3',  
 805: ladoA : '',  
 806: ladoB : '',  
 807: trecho : 'ENTR SP-070 - ENTR SP-103 (CAÇAPAVA)'  
 808: }, {  
 809: usr : '104',  
 810: pwd : '2570',  
 811: fase : '3',  
 812: ladoA : '',  
 813: ladoB : '',  
 814: trecho : 'ENTR SP-228 - ENTR SP-057 (P/SIDERÚRGICA)'  
 815: }, {  
 816: usr : '105',  
 817: pwd : '1043',  
 818: fase : '3',  
 819: ladoA : '',  
 820: ladoB : '',  
 821: trecho : 'ACESSO GUAIÇARA - ENTR BR-154/267 (LINS)'  
 822: }, {  
 823: usr : '106',  
 824: pwd : '0810',  
 825: fase : '3',  
 826: ladoA : '',  
 827: ladoB : '',  
 828: trecho : 'DIV MG/SP - ENTR SP-063 (P/BRAGANÇA PAULISTA)'  
 829: }, {  
 830: usr : '107',  
 831: pwd : '0105',  
 832: fase : '2',  
 833: ladoA : '',  
 834: ladoB : '',  
 835: trecho : 'ENTR TO-420 - INICIO PISTA DUPLA ARAGUAINA'  
 836: }, {  
 837: usr : '108',  
 838: pwd : '0205',  
 839: fase : '3',  
 840: ladoA : '',  
 841: ladoB : '',  
 842: trecho : 'ENTR TO-342(B) (MIRANORTE) - ENTR TO-348 (BARROLÂNDIA)'  
 843: }, {  
 844: usr : '109',  
 845: pwd : '0300',  
 846: fase : '3',  
 847: ladoA : '',  
 848: ladoB : '',  
 849: trecho : 'ENTR TO-483 (FIGUEIRÓPOLIS) - ENTR TO-296(A)/373 (ALVORADA)'  
 850: }, {  
 851: usr : '110',  
 852: pwd : '0495',  
 853: fase : '3',  
 854: ladoA : '',  
 855: ladoB : '',  
 856: trecho : 'ENTR PI-260 (P/BARREIRAS DO PIAUÍ) - ENTR PI-255 (CORRENTE)'  
 857: }, {  
 858: usr : '111',  
 859: pwd : '0150',  
 860: fase : '3',  
 861: ladoA : '',  
 862: ladoB : '',  
 863: trecho : 'FIM PISTA DUPLA - ENTR GO-156(B)'  
 864: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
 865: }, {  
 866: usr : '112',  
 867: pwd : '0025',  
 868: fase : '1',  
 869: ladoA : 'ENTR PR-508 (P/MATINHOS)',  
 870: ladoB : 'ENTR BR-101/PR-408 (P/MORRETES)',  
 871: trecho : 'ENTR PR-508 (P/MATINHOS) - ENTR BR-101/PR-408 (P/MORRETES)'  
 872: }, {  
 873: usr : '113',  
 874: pwd : '0075',  
 875: fase : '1',  
 876: ladoA : 'ACESSO SANTA',  
 877: ladoB : 'ENTR BR-376(B)/PR-428 (SÃO LUIS PURUNÃ)',  
 878: trecho : 'ACESSO SANTA - ENTR BR-376(B)/PR-428 (SÃO LUIS PURUNÃ)'  
 879: }, {  
 880: usr : '114',  
 881: pwd : '0950',  
 882: fase : '1',  
 883: ladoA : 'ENTR MS-375 (ZUZU)',  
 884: ladoB : 'ENTR BR-163(A) (NOVA ALVORADA DO SUL)',  
 885: trecho : 'ENTR MS-375 (ZUZU) - ENTR BR-163(A) (NOVA ALVORADA)'  
 886: }, {  
 887: usr : '115',  
 888: pwd : '0440',  
 889: fase : '1',  
 890: ladoA : 'ENTR MS-434 (RAIMUNDO)',  
 891: ladoB : 'ENTR BR-483/497 (INÍCIO DA PISTA DUPLA)',  
 892: trecho : 'ENTR MS-434 (RAIMUNDO) - ENTR BR-483/497 (INÍCIO DA PISTA DUPLA)'  
 893: }, {  
 894: usr : '116',  
 895: pwd : '0810',  
 896: fase : '3',  
 897: ladoA : '',  
 898: ladoB : '',  
 899: trecho : 'ENTR PISTA INVERSA (B) - ENTR BR-101'  
 900: }, {  
 901: usr : '117',  
 902: pwd : '0070',  
 903: fase : '3',  
 904: ladoA : '',  
 905: ladoB : '',  
 906: trecho : 'ENTR BR-493(A) (P/MAGÉ) - ENTR RJ-107 (IMBARIÊ)'  
 907: }, {  
 908: usr : '118',  
 909: pwd : '0040',  
 910: fase : '1',  
 911: ladoA : 'ENTR BR-101',  
 912: ladoB : 'ENTR SC-108(A) (NEUDOR)',  
 913: trecho : 'ENTR BR-101 - ENTR SC-108(A) (NEUDOR)'  
 914: }, {  
 915: usr : '119',  
 916: pwd : '1490',  
 917: fase : '1',  
 918: ladoA : 'ENTR BR-280(A)/PR-446',  
 919: ladoB : 'ENTR PR-170',  
 920: trecho : 'ENTR BR-153(A)/PR-446 - ENTR PR-170'  
 921: }, {  
 922: usr : '120',  
 923: pwd : '0060',  
 924: fase : '2',  
 925: ladoA : '',  
 926: ladoB : '',  
 927: trecho : 'ENTR BR-262(B) - ENTR BR-050(B)'  
 928: }, {  
 929: usr : '121',  
 930: pwd : '0834',  
 931: fase : '1',  
 932: ladoA : 'ENTR MT-423',  
 933: ladoB : 'ITAUBA',  
 934: trecho : 'ENTR MT-423 - ITAUBA'  
 935: }, {  
 936: usr : '122',  
 937: pwd : '0852',  
 938: fase : '3',  
 939: ladoA : '',  
 940: ladoB : '',  
 941: trecho : 'MATUPÁ - GUARANTÃ DO NORTE'  
 942: }, {  
 943: usr : '123',  
 944: pwd : '0790',  
 945: fase : '3',  
 946: ladoA : '',  
 947: ladoB : '',  
 948: trecho : 'ENTR AL-215(B) - ENTR AL-220(A) (P/BARRA DE SÃO MIGUEL)'  
 949: }, {  
 950: usr : '124',  
 951: pwd : '0260',  
 952: fase : '4',  
 953: ladoA : '',  
 954: ladoB : '',  
 955: trecho : 'ENTR PE-270 (P/BUÍQUE) - ENTR BR-110 (CRUZEIRO DO NORDESTE)'  
 956: }, {  
 957: usr : '125',  
 958: pwd : '0090',  
 959: fase : '4',  
 960: ladoA : '',  
 961: ladoB : '',  
 962: trecho : 'ENTR PE-218 (P/BOM CONSELHO) - ENTR PE-187 (P/PALMEIRINA)'  
 963: }, {  
 964: usr : '126',  
 965: pwd : '1695',  
 966: fase : '3',  
 967: ladoA : '',  
 968: ladoB : '',  
 969: trecho : 'ENTR BR-420(B) (P/LAJE) - ENTR BA-542 (P/GUERÉM)'  
 970: }, {  
 971: usr : '127',  
 972: pwd : '0790',  
 973: fase : '4',  
 974: ladoA : '',  
 975: ladoB : '',  
 976: trecho : 'ENTR BA-493 (ITATIM) - ENTR BA-245'  
 977: }, {  
 978: usr : '128',  
 979: pwd : '0334',  
 980: fase : '2',  
 981: ladoA : '',  
 982: ladoB : '',  
 983: trecho : 'MAIRI - ENTR BR-349'  
 984: }, {  
 985: usr : '129',  
 986: pwd : '0270',  
 987: fase : '2',  
 988: ladoA : '',  
 989: ladoB : '',  
 990: trecho : 'GAVIÃO - ENTR BR-349 (NOVA FÁTIMA)'  
 991: }, {  
 992: usr : '130',  
 993: pwd : '1010',  
 994: fase : '2',  
 995: ladoA : '',  
 996: ladoB : '',  
 997: trecho : 'DIV BA/MG - ENTR BR-251(A) (P/SALINAS)'  
 998: }, {  
 999: usr : '131',  
1000: pwd : '0530',  
1001: fase : '2',  
1002: ladoA : '',  
1003: ladoB : '',  
1004: trecho : 'ENTR BA-937 (P/PAJEÚ DO VENTO) - ENTR BR-122(B)/430/BA-569 (CAETITÉ)'  
1005: }, {  
1006: usr : '132',  
1007: pwd : '0245',  
1008: fase : '3',  
1009: ladoA : '',  
1010: ladoB : '',  
1011: trecho : 'ENTR BA-131 (P/CAÉM) - ENTR BA-417'  
1012: }, {  
1013: usr : '133',  
1014: pwd : '0550',  
1015: fase : '2',  
1016: ladoA : '',  
1017: ladoB : '',  
1018: trecho : 'ENTR BR-235 - ENTR BA-220 (EUCLIDES DA CUNHA)'  
1019: }, {  
1020: usr : '134',  
1021: pwd : '0690',  
1022: fase : '3',  
1023: ladoA : '',  
1024: ladoB : '',  
1025: trecho : 'ENTR CE-373 (CARMELÓPOLIS) - ENTR CE-187/292 (CAMPOS SALES)'  
1026: }, {  
1027: usr : '135',  
1028: pwd : '0190',  
1029: fase : '2',  
1030: ladoA : '',  
1031: ladoB : '',  
1032: trecho : 'ENTR CE-266 (SUCESSO) - ENTR BR-226/404 (CRATEÚS)'  
1033: }, {  
1034: usr : '136',  
1035: pwd : '0610',  
1036: fase : '4',  
1037: ladoA : '',  
1038: ladoB : '',  
1039: trecho : 'ENTR CE-060(B) (MINEIROLÂNDIA) - ENTR CE-168 (PEDRA BRANCA)'  
1040: }, {  
1041: usr : '137',  
1042: pwd : '0050',  
1043: fase : '4',  
1044: ladoA : '',  
1045: ladoB : '',  
1046: trecho : 'ENTR CE-154 (PIRANGI) - ENTR CE-060(A)/265 (QUIXADÁ)'  
1047: }, {  
1048: usr : '138',  
1049: pwd : '0590',  
1050: fase : '2',  
1051: ladoA : '',  
1052: ladoB : '',  
1053: trecho : 'ENTR CE-456 (P/TARGINOS) - ENTR CE-257 (CANINDÉ)'  
1054: }, {  
1055: usr : '139',  
1056: pwd : '0070',  
1057: fase : '4',  
1058: ladoA : '',  
1059: ladoB : '',  
1060: trecho : 'ENTR CE-341 (CROATÁ) - ENTR CE-162 (SÃO LUÍS DO CURU)'  
1061: }, {  
1062: usr : '140',  
1063: pwd : '0160',  
1064: fase : '3',  
1065: ladoA : '',  
1066: ladoB : '',  
1067: trecho : 'ENTR CE-263 (P/JAGUARUANA) - RUSSAS'  
1068: }, {  
1069: usr : '141',  
1070: pwd : '2130',  
1071: fase : '2',  
1072: ladoA : '',  
1073: ladoB : '',  
1074: trecho : 'ENTR BR-381 (SÃO MATEUS) - ENTR ES-430 (P/JAGUARÉ)'  
1075: }, {  
1076: usr : '142',  
1077: pwd : '0300',  
1078: fase : '4',  
1079: ladoA : '',  
1080: ladoB : '',  
1081: trecho : 'ENTR ES-137 - ENTR ES-130 (P/NOVA VENÉCIA)'  
1082: }, {  
1083: usr : '143',  
1084: pwd : '1990',  
1085: fase : '3',  
1086: ladoA : '',  
1087: ladoB : '',  
1088: trecho : 'ENTR BA-290 (TEIXEIRA DE FREITAS) - ENTR BR-418 (P/POSTO DA MATA)'  
1089: }, {  
1090: usr : '144',  
1091: pwd : '0030',  
1092: fase : '2',  
1093: ladoA : '',  
1094: ladoB : '',  
1095: trecho : 'MORRO GRANDE (ACESSO IV C. ITAPEMIRIM) - ENTR ES-166 (COUTINHO)'  
1096: }, {  
1097: usr : '145',  
1098: pwd : '0130',  
1099: fase : '2',  
1100: ladoA : '',  
1101: ladoB : '',  
1102: trecho : 'ENTR MG-422 (RESPLENDOR) - ENTR BR-458 (CONSELHEIRO PENA)'  
1103: }, {  
1104: usr : '146',  
1105: pwd : '0125',  
1106: fase : '3',  
1107: ladoA : '',  
1108: ladoB : '',  
1109: trecho : 'ENTR GO-230 - ENTR GO-237'  
1110: }, {  
1111: usr : '147',  
1112: pwd : '0160',  
1113: fase : '2',  
1114: ladoA : '',  
1115: ladoB : '',  
1116: trecho : 'ENTR GO-114 (P/FLORES DE GOIÁS) - SANTA MARIA'  
1117: }, {  
1118: usr : '148',  
1119: pwd : '0470',  
1120: fase : '3',  
1121: ladoA : '',  
1122: ladoB : '',  
1123: trecho : 'DIV MG/GO - ENTR GO-010(A)'  
1124: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
1125: }, {  
1126: usr : '149',  
1127: pwd : '0050',  
1128: fase : '4',  
1129: ladoA : '',  
1130: ladoB : '',  
1131: trecho : 'ENTR GO-010 (P/LUZIÂNIA) - ENTR BR-050(B)/354/457/GO-309 (CRISTALINA)',  
1132: perguntaExtra : true // Possui pergunta extra na 4ª fase  
1133: }, {  
1134: usr : '150',  
1135: pwd : '0080',  
1136: fase : '3',  
1137: ladoA : '',  
1138: ladoB : '',  
1139: trecho : 'URUTAÍ - ENTR BR-490/GO-213/307 (IPAMERI)'  
1140: }, {  
1141: usr : '151',  
1142: pwd : '0550',  
1143: fase : '2',  
1144: ladoA : '',  
1145: ladoB : '',  
1146: trecho : 'ENTR GO-080(B) (P/SAO FRANCISCO DE GOIÁS) - ENTR GO-431 (P/PIRENOPOLIS)'  
1147: }, {  
1148: usr : '152',  
1149: pwd : '0190',  
1150: fase : '2',  
1151: ladoA : '',  
1152: ladoB : '',  
1153: trecho : 'ENTR BR-153(A)/GO-342(B) - ENTR BR-153(B)/GO-237 (URUAÇÚ)'  
1154: }, {  
1155: usr : '153',  
1156: pwd : '0750',  
1157: fase : '2',  
1158: ladoA : '',  
1159: ladoB : '',  
1160: trecho : 'ENTR GO-210(B) (P/PANAMÁ) - ENTR BR-154/452(A)/483'  
1161: }, {  
1162: usr : '154',  
1163: pwd : '0050',  
1164: fase : '2',  
1165: ladoA : '',  
1166: ladoB : '',  
1167: trecho : 'ENTR GO-040 - ENTR BR-154(A)/483(A)/GO-206(A) (P/CACHOEIRA DOURADA)'  
1168: }, {  
1169: usr : '155',  
1170: pwd : '0570',  
1171: fase : '3',  
1172: ladoA : '',  
1173: ladoB : '',  
1174: trecho : 'ENTR GO-194 (P/PORTELÂNDIA) - INÍCIO PISTA DUPLA'  
1175: }, {  
1176: usr : '156',  
1177: pwd : '0440',  
1178: fase : '3',  
1179: ladoA : '',  
1180: ladoB : '',  
1181: trecho : 'ENTR GO-174(B) - ENTR GO-206 (P/CAÇU)'  
1182: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
1183: }, {  
1184: usr : '157',  
1185: pwd : '0280',  
1186: fase : '2',  
1187: ladoA : '',  
1188: ladoB : '',  
1189: trecho : 'ENTR MA-006(A) (COCALINHO) - ENTR MA-216 (BOM JARDIM)'  
1190: }, {  
1191: usr : '158',  
1192: pwd : '0860',  
1193: fase : '3',  
1194: ladoA : '',  
1195: ladoB : '',  
1196: trecho : 'ENTR MA-259 (P/TUNTUM) - ENTR MA-012/272 (BARRA DO CORDA)'  
1197: }, {  
1198: usr : '159',  
1199: pwd : '0300',  
1200: fase : '2',  
1201: ladoA : '',  
1202: ladoB : '',  
1203: trecho : 'ENTR BR-222(B) (ESTACA ZERO) - ENTR MA-240 (PIO XII)'  
1204: }, {  
1205: usr : '160',  
1206: pwd : '0730',  
1207: fase : '3',  
1208: ladoA : '',  
1209: ladoB : '',  
1210: trecho : 'BOM JESUS DO TOCANTINS - RIO JACUNDÁ (R MÃE MARIA)'  
1211: }, {  
1212: usr : '161',  
1213: pwd : '0750',  
1214: fase : '3',  
1215: ladoA : '',  
1216: ladoB : '',  
1217: trecho : 'ENTR BR-122/251/365 (MONTES CLAROS) - ENTR BR-451 (BOCAIÚVA)'  
1218: }, {  
1219: usr : '162',  
1220: pwd : '0240',  
1221: fase : '4',  
1222: ladoA : '',  
1223: ladoB : '',  
1224: trecho : 'ENTR BR-342/MG-404 (SALINAS) - RIO VACARIA'  
1225: }, {  
1226: usr : '163',  
1227: pwd : '0290',  
1228: fase : '2',  
1229: ladoA : '',  
1230: ladoB : '',  
1231: trecho : 'ENTR MA-132/270 (COLINAS) - ENTR MA-134 (PEIXE)'  
1232: }, {  
1233: usr : '164',  
1234: pwd : '0030',  
1235: fase : '3',  
1236: ladoA : '',  
1237: ladoB : '',  
1238: trecho : 'ENTR BR-040(B) - ENTR MG-155 (SÃO BRÁS DO SUAÇUÍ)'  
1239: }, {  
1240: usr : '165',  
1241: pwd : '0335',  
1242: fase : '2',  
1243: ladoA : '',  
1244: ladoB : '',  
1245: trecho : 'ENTR MG-434 - ENTR MG-435 (P/CAETÉ)'  
1246: }, {  
1247: usr : '166',  
1248: pwd : '0510',  
1249: fase : '2',  
1250: ladoA : '',  
1251: ladoB : '',  
1252: trecho : 'ENTR MG-275 (P/CARANDAÍ) - ACESSO ALTO DOCE (INÍCIO PISTA DUPLA)'  
1253: }, {  
1254: usr : '167',  
1255: pwd : '0130',  
1256: fase : '3',  
1257: ladoA : '',  
1258: ladoB : '',  
1259: trecho : 'ENTR BR-040(B) - ENTR MG-135'  
1260: }, {  
1261: usr : '168',  
1262: pwd : '0250',  
1263: fase : '3',  
1264: ladoA : '',  
1265: ladoB : '',  
1266: trecho : 'ENTR MG-190 (RIO BAGAGEM) - ACESSO INDIANÓPOLIS'  
1267: }, {  
1268: usr : '169',  
1269: pwd : '0152',  
1270: fase : '3',  
1271: ladoA : '',  
1272: ladoB : '',  
1273: trecho : 'ENTR BR-352(B)/GO-210(B)/330 (CATALÃO) - ENTR GO-402'  
1274: }, {  
1275: usr : '170',  
1276: pwd : '0010',  
1277: fase : '3',  
1278: ladoA : '',  
1279: ladoB : '',  
1280: trecho : 'ENTR BR-050/365/452/455 (UBERLÂNDIA) - ENTR BR-153/464 (PRATA)'  
1281: }, {  
1282: usr : '171',  
1283: pwd : '0310',  
1284: fase : '4',  
1285: ladoA : '',  
1286: ladoB : '',  
1287: trecho : 'ENTR BR-050(B)/455/497 (UBERLÂNDIA) - ENTR BR-452(B)'  
1288: }, {  
1289: usr : '172',  
1290: pwd : '0230',  
1291: fase : '4',  
1292: ladoA : '',  
1293: ladoB : '',  
1294: trecho : 'ENTR MG-190 (P/NOVA PONTE) - ACESSO SANTA JULIANA'  
1295: }, {  
1296: usr : '173',  
1297: pwd : '0110',  
1298: fase : '3',  
1299: ladoA : '',  
1300: ladoB : '',  
1301: trecho : 'ACESSO CARLOS CHAGAS - ENTR MG-412'  
1302: }, {  
1303: usr : '174',  
1304: pwd : '0110',  
1305: fase : '2',  
1306: ladoA : '',  
1307: ladoB : '',  
1308: trecho : 'ACESSO SÃO JOSÉ DO JACURÍ - ENTR MG-117 (SÃO JOÃO EVANGELISTA)'  
1309: }, {  
1310: usr : '175',  
1311: pwd : '1310',  
1312: fase : '4',  
1313: ladoA : '',  
1314: ladoB : '',  
1315: trecho : 'TREVO DE ACESSO SÃO JOÃO DO MANHUAÇU - ENTR MG-265 (P/DIVINO)'  
1316: }, {  
1317: usr : '176',  
1318: pwd : '0210',  
1319: fase : '2',  
1320: ladoA : '',  
1321: ladoB : '',  
1322: trecho : 'ENTR ES-185(B) (P/IÚNA) - DIV ES/MG'  
1323: }, {  
1324: usr : '177',  
1325: pwd : '1195',  
1326: fase : '3',  
1327: ladoA : '',  
1328: ladoB : '',  
1329: trecho : 'ACESSO ITANHOMI - ENTR BR-458(A) (TURUAÇÚ)'  
1330: }, {  
1331: usr : '178',  
1332: pwd : '0030',  
1333: fase : '2',  
1334: ladoA : '',  
1335: ladoB : '',  
1336: trecho : 'ENTR BR-116(B) - ENTR MG-126 (BICAS)'  
1337: }, {  
1338: usr : '179',  
1339: pwd : '1370',  
1340: fase : '2',  
1341: ladoA : '',  
1342: ladoB : '',  
1343: trecho : 'ENTR BR-265/356 (MURIAÉ) - ENTR MG-285 (LARANJAL)'  
1344: }, {  
1345: usr : '180',  
1346: pwd : '0310',  
1347: fase : '3',  
1348: ladoA : '',  
1349: ladoB : '',  
1350: trecho : 'PONTE S/ RIO PARÁ - ENTR MG-423(A) (PITANGUI)'  
1351: }, {  
1352: usr : '181',  
1353: pwd : '0195',  
1354: fase : '2',  
1355: ladoA : '',  
1356: ladoB : '',  
1357: trecho : 'ACESSO MORADA NOVA DE MINAS - ACESSO SÃO JOSÉ DO BURITI'  
1358: }, {  
1359: usr : '182',  
1360: pwd : '0330',  
1361: fase : '2',  
1362: ladoA : '',  
1363: ladoB : '',  
1364: trecho : 'ENTR BR-367(B) (GOUVEIA) - ACESSO NOSSA SENHORA DA GLÓRIA'  
1365: }, {  
1366: usr : '183',  
1367: pwd : '0910',  
1368: fase : '2',  
1369: ladoA : '',  
1370: ladoB : '',  
1371: trecho : 'ENTR MG-255 - ENTR BR-262(B) (DIV MG/SP)'  
1372: }, {  
1373: usr : '184',  
1374: pwd : '0250',  
1375: fase : '3',  
1376: ladoA : '',  
1377: ladoB : '',  
1378: trecho : 'ENTR BR-462(B) - ENTR BR-262'  
1379: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
1380: }, {  
1381: usr : '185',  
1382: pwd : '0930',  
1383: fase : '4',  
1384: ladoA : '',  
1385: ladoB : '',  
1386: trecho : 'ENTR BR-452 (P/UBERLÂNDIA) - ENTR BR-462 (PERDIZES)'  
1387: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
1388: }, {  
1389: usr : '186',  
1390: pwd : '0070',  
1391: fase : '2',  
1392: ladoA : '',  
1393: ladoB : '',  
1394: trecho : 'ENTR BR-265(B) - CAMPOS GERAIS'  
1395: }, {  
1396: usr : '187',  
1397: pwd : '0030',  
1398: fase : '2',  
1399: ladoA : '',  
1400: ladoB : '',  
1401: trecho : 'ENTR MG-354(B) - ENTR BR-265(A)'  
1402: }, {  
1403: usr : '188',  
1404: pwd : '0190',  
1405: fase : '2',  
1406: ladoA : '',  
1407: ladoB : '',  
1408: trecho : 'ENTR MG-135/338 (BARBACENA) - ACESSO TIRADENTES'  
1409: }, {  
1410: usr : '189',  
1411: pwd : '0670',  
1412: fase : '4',  
1413: ladoA : '',  
1414: ladoB : '',  
1415: trecho : 'ENTR BR-265 (LAVRAS) - ENTR MG-167(A) (P/TRÊS CORAÇÕES)'  
1416: }, {  
1417: usr : '190',  
1418: pwd : '0350',  
1419: fase : '2',  
1420: ladoA : '',  
1421: ladoB : '',  
1422: trecho : 'ACESSO FORMIGA - ENTR MG-164 (CANDEIAS)'  
1423: }, {  
1424: usr : '191',  
1425: pwd : '0590',  
1426: fase : '2',  
1427: ladoA : '',  
1428: ladoB : '',  
1429: trecho : 'ENTR MG-270 (CARMÓPOLIS DE MINAS) - ENTR BR-494(A) (P/OLIVEIRA)'  
1430: }, {  
1431: usr : '192',  
1432: pwd : '0230',  
1433: fase : '3',  
1434: ladoA : '',  
1435: ladoB : '',  
1436: trecho : 'ENTR BR-354(A) (P/BAMBUÍ) - ENTR BR-354(B)'  
1437: }, {  
1438: usr : '193',  
1439: pwd : '0205',  
1440: fase : '2',  
1441: ladoA : '',  
1442: ladoB : '',  
1443: trecho : 'ACESSO GUIMARÂNIA - ENTR MG-188'  
1444: }, {  
1445: usr : '194',  
1446: pwd : '0090',  
1447: fase : '3',  
1448: ladoA : '',  
1449: ladoB : '',  
1450: trecho : 'ENTR MG-410 - PRESIDENTE OLEGÁRIO'  
1451: }, {  
1452: usr : '195',  
1453: pwd : '0150',  
1454: fase : '4',  
1455: ladoA : '',  
1456: ladoB : '',  
1457: trecho : 'ENTR MG-060 (P/SÃO GONÇALO DO ABAETÉ) - ACESSO GALENA'  
1458: }, {  
1459: usr : '196',  
1460: pwd : '0410',  
1461: fase : '2',  
1462: ladoA : '',  
1463: ladoB : '',  
1464: trecho : 'ENTR BR-154(B)/461/464 (ITUIUTABA) - ACESSO GURINHATÃ'  
1465: }, {  
1466: usr : '197',  
1467: pwd : '0830',  
1468: fase : '2',  
1469: ladoA : '',  
1470: ladoB : '',  
1471: trecho : 'ENTR BR-365 (P/MONTE ALEGRE DE MINAS) - ENTR BR-464/497 (P/PRATA)'  
1472: }, {  
1473: usr : '198',  
1474: pwd : '0170',  
1475: fase : '3',  
1476: ladoA : '',  
1477: ladoB : '',  
1478: trecho : 'ENTR MG-229 (P/SENHORA DO PORTO) - ENTR MG-232 (P/CARMESIA)'  
1479: }, {  
1480: usr : '199',  
1481: pwd : '0370',  
1482: fase : '2',  
1483: ladoA : '',  
1484: ladoB : '',  
1485: trecho : 'ENTR MG-320 (P/SÃO JOSÉ DO GOIABAL) - ENTR BR-120 (VARGEM LINDA)'  
1486: }, {  
1487: usr : '200',  
1488: pwd : '0170',  
1489: fase : '2',  
1490: ladoA : '',  
1491: ladoB : '',  
1492: trecho : 'ACESSO À GOV. VALADARES - PERIQUITO'  
1493: }, {  
1494: usr : '201',  
1495: pwd : '0685',  
1496: fase : '3',  
1497: ladoA : '',  
1498: ladoB : '',  
1499: trecho : 'ACESSO FLORESTAL - ENTR BR-352 (PARÁ DE MINAS)'  
1500: }, {  
1501: usr : '202',  
1502: pwd : '0050',  
1503: fase : '2',  
1504: ladoA : '',  
1505: ladoB : '',  
1506: trecho : 'ENTR MG-050(B) (P/DIVINOPÓLIS) - ENTR MG-260 (P/CLÁUDIO)'  
1507: }, {  
1508: usr : '203',  
1509: pwd : '0270',  
1510: fase : '2',  
1511: ladoA : '',  
1512: ladoB : '',  
1513: trecho : 'ACESSO CONCEIÇÃO DO RIO VERDE - ENTR BR-460'  
1514: }, {  
1515: usr : '204',  
1516: pwd : '0023',  
1517: fase : '3',  
1518: ladoA : '',  
1519: ladoB : '',  
1520: trecho : 'ACESSO SANTA RITA DAS CALDAS - ENTR MG-179 (POUSO ALEGRE)'  
1521: }, {  
1522: usr : '205',  
1523: pwd : '0270',  
1524: fase : '3',  
1525: ladoA : '',  
1526: ladoB : '',  
1527: trecho : 'ENTR BR-369 (BOTELHOS) - ENTR BR-267(A) (P/BANDEIRA DO SUL)'  
1528: }, {  
1529: usr : '206',  
1530: pwd : '0212',  
1531: fase : '3',  
1532: ladoA : '',  
1533: ladoB : '',  
1534: trecho : 'ENTR BR-487(A) - ACESSO NAVIRAÍ I'  
1535: }, {  
1536: usr : '207',  
1537: pwd : '0612',  
1538: fase : '1',  
1539: ladoA : 'ENTR BR-267(A)',  
1540: ladoB : 'INÍCIO DUPLICAÇÃO',  
1541: trecho : 'ENTR BR-267(A) - INÍCIO DUPLICAÇÃO'  
1542: }, {  
1543: usr : '208',  
1544: pwd : '0090',  
1545: fase : '3',  
1546: ladoA : '',  
1547: ladoB : '',  
1548: trecho : 'ENTR MS-141 (IVINHEMA) - VILA AMANDINA'  
1549: }, {  
1550: usr : '209',  
1551: pwd : '1390',  
1552: fase : '1',  
1553: ladoA : 'ENTR MS-339/446/448 (MIRANDA)',  
1554: ladoB : 'ENTR MS-185/243 (GUAICURUS)',  
1555: trecho : 'ENTR MS-339/446/448 (MIRANDA) - ENTR MS-185/243 (GUAICURUS)'  
1556: }, {  
1557: usr : '210',  
1558: pwd : '0058',  
1559: fase : '1',  
1560: ladoA : 'BURITIZINHO',  
1561: ladoB : 'ENTR MS-223(A) (SILVOLÂNDIA)',  
1562: trecho : 'BURITIZINHO - ENTR MS-223(A) (SILVOLÂNDIA)'  
1563: }, {  
1564: usr : '211',  
1565: pwd : '0390',  
1566: fase : '1',  
1567: ladoA : 'ENTR MS-316 (PARAÍSO DAS ÁGUAS)',  
1568: ladoB : 'ENTR MS-324(A)',  
1569: trecho : 'ENTR MS-316 (PARAÍSO) - ENTR MS-324(A)'  
1570: // perguntaExtra : true // Possuiu pergunta extra na 1ª fase  
1571: }, {  
1572: usr : '212',  
1573: pwd : '0180',  
1574: fase : '3',  
1575: ladoA : '',  
1576: ladoB : '',  
1577: trecho : 'ENTR BR-158(B) (P/TAMBOARA) - ACESSO ALTO PARANÁ/MARISTELA'  
1578: }, {  
1579: usr : '213',  
1580: pwd : '0050',  
1581: fase : '2',  
1582: ladoA : '',  
1583: ladoB : '',  
1584: trecho : 'ENTR SP-425 (PRESIDENTE PRUDENTE) - ENTR SP-425'  
1585: }, {  
1586: usr : '214',  
1587: pwd : '0330',  
1588: fase : '1',  
1589: ladoA : 'ENTR MT-373',  
1590: ladoB : 'ENTR MT-130(A) (PRIMAVERA DO LESTE)',  
1591: trecho : 'ENTR MT-373 - ENTR MT-130(A) (PRIMAVERA DO LESTE)'  
1592: }, {  
1593: usr : '215',  
1594: pwd : '0765',  
1595: fase : '2',  
1596: ladoA : '',  
1597: ladoB : '',  
1598: trecho : 'FIM VARIANTE II SERRA DE SÃO VICENTE - ACESSO DISTRITO INDUSTRIAL'  
1599: }, {  
1600: usr : '216',  
1601: pwd : '0150',  
1602: fase : '3',  
1603: ladoA : '',  
1604: ladoB : '',  
1605: trecho : 'ENTR BR-010(B) (P/SANTA MARIA DO PARÁ) - ENTR PA-324 (P/SALINÓPOLIS)'  
1606: }, {  
1607: usr : '217',  
1608: pwd : '0270',  
1609: fase : '3',  
1610: ladoA : '',  
1611: ladoB : '',  
1612: trecho : 'ENTR PB-177 (SOLEDADE) - ENTR PB-195 (JUAZEIRINHO)'  
1613: }, {  
1614: usr : '218',  
1615: pwd : '0322',  
1616: fase : '2',  
1617: ladoA : '',  
1618: ladoB : '',  
1619: trecho : 'BARRA DE SANTANA - ENTR PB-196 (P/RIACHO DE SANTO ANTÔNIO)'  
1620: }, {  
1621: usr : '219',  
1622: pwd : '0280',  
1623: fase : '2',  
1624: ladoA : '',  
1625: ladoB : '',  
1626: trecho : 'ENTR PE-263 (P/ITAPETIM) - ENTR PE-320 (SÃO JOSÉ DO EGITO)'  
1627: }, {  
1628: usr : '220',  
1629: pwd : '0410',  
1630: fase : '3',  
1631: ladoA : '',  
1632: ladoB : '',  
1633: trecho : 'ENTR PE-015 (P/PAULISTA - VIADUTO SOBRE PE-015) - VIADUTO SOBRE AVENIDA CAXANGÁ'  
1634: }, {  
1635: usr : '221',  
1636: pwd : '0530',  
1637: fase : '2',  
1638: ladoA : '',  
1639: ladoB : '',  
1640: trecho : 'DIV PE/AL - ENTR BR-423(A)'  
1641: }, {  
1642: usr : '222',  
1643: pwd : '0630',  
1644: fase : '2',  
1645: ladoA : '',  
1646: ladoB : '',  
1647: trecho : 'ENTR PE-590 (P/IPUBÍ) - ENTR BR-122 (OURICURÍ)'  
1648: }, {  
1649: usr : '223',  
1650: pwd : '0250',  
1651: fase : '3',  
1652: ladoA : '',  
1653: ladoB : '',  
1654: trecho : 'ENTR CE-292(B) - DIV CE/PE'  
1655: }, {  
1656: usr : '224',  
1657: pwd : '0300',  
1658: fase : '3',  
1659: ladoA : '',  
1660: ladoB : '',  
1661: trecho : 'ENTR PE-310/312 (CUSTÓDIA) - ENTR BR-426/PE-340 (SÍTIO DOS NUNES)'  
1662: }, {  
1663: usr : '225',  
1664: pwd : '0250',  
1665: fase : '2',  
1666: ladoA : '',  
1667: ladoB : '',  
1668: trecho : 'RAJADA - ENTR BR-235(A)'  
1669: }, {  
1670: usr : '226',  
1671: pwd : '0115',  
1672: fase : '2',  
1673: ladoA : '',  
1674: ladoB : '',  
1675: trecho : 'ENTR PI-394 - ENTR PI-140 (CANTO DO BURITI)'  
1676: }, {  
1677: usr : '227',  
1678: pwd : '0070',  
1679: fase : '3',  
1680: ladoA : '',  
1681: ladoB : '',  
1682: trecho : 'ENTR PI-211 (CAXINGÓ) - ENTR PI-213(A)'  
1683: }, {  
1684: usr : '228',  
1685: pwd : '0190',  
1686: fase : '2',  
1687: ladoA : '',  
1688: ladoB : '',  
1689: trecho : 'ACESSO OESTE SOBRAL - ENTR BR-403(B)/CE-183'  
1690: }, {  
1691: usr : '229',  
1692: pwd : '0330',  
1693: fase : '2',  
1694: ladoA : '',  
1695: ladoB : '',  
1696: trecho : 'ENTR MA-034 (SÃO JOÃO DOS PATOS) - ENTR BR-230(B)/MA-364(A) (DOIS IRMÃOS)'  
1697: }, {  
1698: usr : '230',  
1699: pwd : '0290',  
1700: fase : '2',  
1701: ladoA : '',  
1702: ladoB : '',  
1703: trecho : 'ENTR PI-130 (P/AMARANTE) - ENTR PI-120'  
1704: }, {  
1705: usr : '231',  
1706: pwd : '0210',  
1707: fase : '3',  
1708: ladoA : '',  
1709: ladoB : '',  
1710: trecho : 'ENTR SP-252 (GUAPIARA) - ENTR SP-165/249 (APIAÍ)'  
1711: }, {  
1712: usr : '232',  
1713: pwd : '0180',  
1714: fase : '1',  
1715: ladoA : 'ENTR PR-549 (P/LUIZIANA)',  
1716: ladoB : 'ENTR PR-462 (P/IRETAMA)',  
1717: trecho : 'ENTR PR-549 (P/LUIZIANA) - ENTR PR-462 (P/IRETAMA)'  
1718: }, {  
1719: usr : '233',  
1720: pwd : '0080',  
1721: fase : '3',  
1722: ladoA : '',  
1723: ladoB : '',  
1724: trecho : 'ACESSO IVAIPORÃ - ENTR BR-487 (P/MANOEL RIBAS)'  
1725: }, {  
1726: usr : '234',  
1727: pwd : '0250',  
1728: fase : '1',  
1729: ladoA : 'ENTR BR-158(B) (LARANJEIRAS DO SUL)',  
1730: ladoB : 'ENTR PR-473',  
1731: trecho : 'ENTR BR-158(B) (LARANJEIRAS DO SUL) - ENTR PR-473'  
1732: }, {  
1733: usr : '235',  
1734: pwd : '0455',  
1735: fase : '1',  
1736: ladoA : 'ACESSO ÁGUA MINERAL SANTA CLARA',  
1737: ladoB : 'ENTR PR-281(A) (P/CHOPINZINHO)',  
1738: trecho : 'ACESSO ÁGUA MINERAL SANTA CLARA - ENTR PR-281(A) (P/CHOPINZINHO)'  
1739: }, {  
1740: usr : '236',  
1741: pwd : '0500',  
1742: fase : '1',  
1743: ladoA : 'ENTR PR-468(B)',  
1744: ladoB : 'ENTR PR-180 (GOIO ERÊ)',  
1745: trecho : 'ENTR PR-468(B) - ENTR PR-180 (GOIO ERÊ)'  
1746: }, {  
1747: usr : '237',  
1748: pwd : '1230',  
1749: fase : '1',  
1750: ladoA : 'ENTR PR-431(B)',  
1751: ladoB : 'ENTR PR-092(A) (P/BARRA DO JACARÉ)',  
1752: trecho : 'ENTR PR-431(B) - ENTR PR-092(A) (P/BARRA DO JACARÉ)'  
1753: }, {  
1754: usr : '238',  
1755: pwd : '1140',  
1756: fase : '2',  
1757: ladoA : '',  
1758: ladoB : '',  
1759: trecho : 'ENTR SP-331(A) - ENTR SP-331(B)'  
1760: }, {  
1761: usr : '239',  
1762: pwd : '0230',  
1763: fase : '3',  
1764: ladoA : '',  
1765: ladoB : '',  
1766: trecho : 'ENTR BR-369 - ENTR SP-261 (P/ÁGUAS DE SANTA BÁRBARA)'  
1767: }, {  
1768: usr : '240',  
1769: pwd : '0710',  
1770: fase : '1',  
1771: ladoA : 'ACESSO PIRAPÓ',  
1772: ladoB : 'ACESSO CAMBIRA',  
1773: trecho : 'ACESSO PIRAPÓ - ACESSO CAMBIRA'  
1774: }, {  
1775: usr : '241',  
1776: pwd : '0410',  
1777: fase : '3',  
1778: ladoA : '',  
1779: ladoB : '',  
1780: trecho : 'ACESSO PARAÍBA DO SUL - ENTR RJ-135 (ANDRADE PINTO)'  
1781: }, {  
1782: usr : '242',  
1783: pwd : '2850',  
1784: fase : '4',  
1785: ladoA : '',  
1786: ladoB : '',  
1787: trecho : 'ENTR RJ-196 - ENTR RJ-182'  
1788: }, {  
1789: usr : '243',  
1790: pwd : '0590',  
1791: fase : '3',  
1792: ladoA : '',  
1793: ladoB : '',  
1794: trecho : 'ITAMONTE - ENTR BR-485 (GARGANTA DO REGISTRO)'  
1795: }, {  
1796: usr : '244',  
1797: pwd : '3350',  
1798: fase : '3',  
1799: ladoA : '',  
1800: ladoB : '',  
1801: trecho : 'ENTR RJ-149 (P/RIO CLARO) - ENTR BR-494(A) (ANGRA DOS REIS)'  
1802: }, {  
1803: usr : '245',  
1804: pwd : '0530',  
1805: fase : '3',  
1806: ladoA : '',  
1807: ladoB : '',  
1808: trecho : 'ENTR RJ-145 (P/BARRA DO PIRAÍ) - ENTR RJ-141 (DORANDIA)'  
1809: }, {  
1810: usr : '246',  
1811: pwd : '0170',  
1812: fase : '4',  
1813: ladoA : '',  
1814: ladoB : '',  
1815: trecho : 'ENTR BR-494 (ARANTINA) - AIURUOCA'  
1816: }, {  
1817: usr : '247',  
1818: pwd : '0170',  
1819: fase : '2',  
1820: ladoA : '',  
1821: ladoB : '',  
1822: trecho : 'ENTR RJ-198(B) - ENTR RJ-200 (MONTE ALEGRE)'  
1823: }, {  
1824: usr : '248',  
1825: pwd : '0350',  
1826: fase : '4',  
1827: ladoA : '',  
1828: ladoB : '',  
1829: trecho : 'ENTR RJ-202 - ENTR RJ-234 (P/PENEDO)'  
1830: }, {  
1831: usr : '249',  
1832: pwd : '0150',  
1833: fase : '3',  
1834: ladoA : '',  
1835: ladoB : '',  
1836: trecho : 'ENTR BR-482(B) - ENTR ES-181'  
1837: }, {  
1838: usr : '250',  
1839: pwd : '0210',  
1840: fase : '3',  
1841: ladoA : '',  
1842: ladoB : '',  
1843: trecho : 'ENTR BR-116/265 (MURIAÉ) - DIV MG/RJ'  
1844: }, {  
1845: usr : '251',  
1846: pwd : '1530',  
1847: fase : '2',  
1848: ladoA : '',  
1849: ladoB : '',  
1850: trecho : 'ENTR RJ-156 (P/VOLTA DO PIÃO) - ENTR BR-492(A)'  
1851: }, {  
1852: usr : '252',  
1853: pwd : '0065',  
1854: fase : '3',  
1855: ladoA : '',  
1856: ladoB : '',  
1857: trecho : 'ACESSO À JENIPABU - ENTR RN-160(B) (P/EXTREMOZ)'  
1858: }, {  
1859: usr : '253',  
1860: pwd : '0110',  
1861: fase : '3',  
1862: ladoA : '',  
1863: ladoB : '',  
1864: trecho : 'TAIPÚ - ENTR RN-064 (P/CEARÁ MIRIM)'  
1865: }, {  
1866: usr : '254',  
1867: pwd : '0055',  
1868: fase : '3',  
1869: ladoA : '',  
1870: ladoB : '',  
1871: trecho : 'ENTR CE-261 - DIV CE/RN'  
1872: }, {  
1873: usr : '255',  
1874: pwd : '0210',  
1875: fase : '3',  
1876: ladoA : '',  
1877: ladoB : '',  
1878: trecho : 'ENTR BR-230(A) (SANTA GERTRUDES) - ENTR PB-275(A) (P/SÃO JOSÉ DE ESPINHARA)'  
1879: }, {  
1880: usr : '256',  
1881: pwd : '1375',  
1882: fase : '3',  
1883: ladoA : '',  
1884: ladoB : '',  
1885: trecho : 'ACESSO USINA DE SAMUEL - INICIO PISTA DUPLA'  
1886: }, {  
1887: usr : '257',  
1888: pwd : '1510',  
1889: fase : '3',  
1890: ladoA : '',  
1891: ladoB : '',  
1892: trecho : 'VISTA ALEGRE DO ABUANÃ - VILA EXTREMA'  
1893: }, {  
1894: usr : '258',  
1895: pwd : '1300',  
1896: fase : '2',  
1897: ladoA : '',  
1898: ladoB : '',  
1899: trecho : 'ENTR BR-421 (ARIQUEMES) - ENTR RO-459(A) (ALTO PARAISO)'  
1900: }, {  
1901: usr : '259',  
1902: pwd : '0031',  
1903: fase : '2',  
1904: ladoA : '',  
1905: ladoB : '',  
1906: trecho : 'ENTR RO-135 (P/NOVA LONDRINA) - FIM DO TRECHO PAVIMENTADO'  
1907: }, {  
1908: usr : '260',  
1909: pwd : '1148',  
1910: fase : '2',  
1911: ladoA : '',  
1912: ladoB : '',  
1913: trecho : 'ENTR RO-471 (P/MINISTRO ANDREAZZA) - ENTR RO-479(A) (P/ROLIM DE MOURA)'  
1914: }, {  
1915: usr : '261',  
1916: pwd : '1090',  
1917: fase : '2',  
1918: ladoA : '',  
1919: ladoB : '',  
1920: trecho : 'ENTR RO-391 (POSTO GUAPORÉ) - MARCO RONDON'  
1921: }, {  
1922: usr : '262',  
1923: pwd : '0132',  
1924: fase : '1',  
1925: ladoA : 'PADRONAL',  
1926: ladoB : 'RIO 12 DE OUTUBRO',  
1927: trecho : 'PADRONAL - RIO 12 DE OUTUBRO'  
1928: }, {  
1929: usr : '263',  
1930: pwd : '4425',  
1931: fase : '1',  
1932: ladoA : 'ENTR ACESSO SUL DE CAPÃO DA CANOA',  
1933: ladoB : 'ENTR BR-290(A) (OSÓRIO)',  
1934: trecho : 'ENTR ACESSO SUL DE CAPÃO DA CANOA - ENTR BR-290(A) (OSÓRIO)'  
1935: }, {  
1936: usr : '264',  
1937: pwd : '0330',  
1938: fase : '1',  
1939: ladoA : 'ENTR BR-470',  
1940: ladoB : 'ENTR BR-470/116(A) (CANOAS)',  
1941: trecho : 'ENTR RSC-470 (P/ MONTENEGRO) - ENTR ERS-124 (P/ POLO PETROQUIMICO)'  
1942: }, {  
1943: usr : '265',  
1944: pwd : '0210',  
1945: fase : '1',  
1946: ladoA : 'ENTR BR-153(B)/RS-324 (P/PONTÃO)',  
1947: ladoB : 'ENTR BR-377(A)/386 (CARAZINHO)',  
1948: trecho : 'ENTR BRS-153(B)/ERS-324 (P/ PONTAO) - ENTR BRS-377(A)/386 (P/ CARAZINHO)'  
1949: }, {  
1950: usr : '266',  
1951: pwd : '0470',  
1952: fase : '1',  
1953: ladoA : 'ENTR RS-168 (P/SÃO PAULO DAS MISSÕES)',  
1954: ladoB : 'ENTR BR-472 (FRONT BRASIL/ARGENTINA) (PORTO XAVIER)',  
1955: trecho : 'ENTR RS-168 (P/SÃO PAULO DAS MISSÕES) - ENTR BR-472 (FRONT BRASIL/ARGENTINA) (PORTO XAVIER)'  
1956: }, {  
1957: usr : '267',  
1958: pwd : '0140',  
1959: fase : '1',  
1960: ladoA : 'A. BELTRÃO',  
1961: ladoB : 'ENTR BR-287 (SANTIAGO)',  
1962: trecho : 'ENTR ERS-533 (P/ CAPAO CIPO) - ENTR BRS-287 (CONTORNO DE SANTIAGO)'  
1963: }, {  
1964: usr : '268',  
1965: pwd : '1850',  
1966: fase : '3',  
1967: ladoA : 'ENTR BR-290(B)',  
1968: ladoB : 'ENTR BR-392',  
1969: trecho : 'ENTR BRS-290(B) (P/ SAO GABRIEL) - ENTR BRS-392 (P/ CACAPAVA DO SUL)'  
1970: }, {  
1971: usr : '269',  
1972: pwd : '1350',  
1973: fase : '1',  
1974: ladoA : 'ENTR BR-290(A) (P/ROSÁRIO DO SUL)',  
1975: ladoB : 'ENTR RS-640 (P/CACEQUI)',  
1976: trecho : 'ENTR BRS-158(A) (P/ AZEVEDO SODRE) - ENTR ERS-640 (P/ CACEQUI)'  
1977: }, {  
1978: usr : '270',  
1979: pwd : '0230',  
1980: fase : '1',  
1981: ladoA : 'ENTR BR-473 (SARANDI)',  
1982: ladoB : 'CURRAL ALTO',  
1983: trecho : 'ENTR RSC-473 (VILA SARANDI) - CURRAL ALTO'  
1984: }, {  
1985: usr : '271',  
1986: pwd : '0310',  
1987: fase : '1',  
1988: ladoA : 'ENTR RS-476 (LAJEADO GRANDE)',  
1989: ladoB : 'ENTR RS-110 (VÁRZEA DO CEDRO)',  
1990: trecho : 'ENTR ERS-476 (LAJEADO GRANDE) - ENTR ERS-110 (VARZEA DO CEDRO)'  
1991: }, {  
1992: usr : '272',  
1993: pwd : '0015',  
1994: fase : '1',  
1995: ladoA : 'ENTR BR-282(B)/470(A)',  
1996: ladoB : 'ENTR SC-455 (P/IBICUÍ)',  
1997: trecho : 'ENTR BR-282(B)/470(A) - ENTR SC-455 (P/IBICUÍ)'  
1998: }, {  
1999: usr : '273',  
2000: pwd : '3010',  
2001: fase : '1',  
2002: ladoA : 'DIV SC/RS (RIO PELOTAS)',  
2003: ladoB : 'ENTR BR-285(A) (P/VACARIA)',  
2004: trecho : 'DIVISA SC/RS (RIO PELOTAS) - ENTR BRS-285(A) (P/ VACARIA)'  
2005: }, {  
2006: usr : '274',  
2007: pwd : '0110',  
2008: fase : '1',  
2009: ladoA : 'ENTR BR-486 (BOM RETIRO)',  
2010: ladoB : 'ENTR SC-345(B) (P/URUBICI)',  
2011: trecho : 'ENTR BR-486 (BOM RETIRO) - ENTR SC-345(B) (P/URUBICI)'  
2012: }, {  
2013: usr : '275',  
2014: pwd : '0027',  
2015: fase : '1',  
2016: ladoA : 'ENTR SC-473 (P/ANCHIETA)',  
2017: ladoB : 'SÃO JOSÉ DO CEDRO',  
2018: trecho : 'ENTR SC-473 (P/ANCHIETA) - SÃO JOSÉ DO CEDRO'  
2019: }, {  
2020: usr : '276',  
2021: pwd : '0290',  
2022: fase : '3',  
2023: ladoA : '',  
2024: ladoB : '',  
2025: trecho : 'ENTR BR-280(A) - ENTR BR-280(B) (VITORINO)'  
2026: }, {  
2027: usr : '277',  
2028: pwd : '0612',  
2029: fase : '2',  
2030: ladoA : '',  
2031: ladoB : '',  
2032: trecho : 'ENTR BA-392 (ANTAS) - ENTR BA-220 (CÍCERO DANTAS)'  
2033: }, {  
2034: usr : '278',  
2035: pwd : '0190',  
2036: fase : '3',  
2037: ladoA : '',  
2038: ladoB : '',  
2039: trecho : 'ENTR SP-183 - ENTR BR-116(A) (P/LORENA)'  
2040: }, {  
2041: usr : '279',  
2042: pwd : '2360',  
2043: fase : '3',  
2044: ladoA : '',  
2045: ladoB : '',  
2046: trecho : 'ENTR SP-062 (P/ROSEIRA) - ENTR BR-383(A)/SP-132 (PINDAMONHANGABA)'  
2047: }, {  
2048: usr : '280',  
2049: pwd : '2470',  
2050: fase : '3',  
2051: ladoA : '',  
2052: ladoB : '',  
2053: trecho : 'ENTR SP-065 (P/IGARATÁ) - ENTR SP-056 (ARUJÁ)'  
2054: }, {  
2055: usr : '281',  
2056: pwd : '2630',  
2057: fase : '3',  
2058: ladoA : '',  
2059: ladoB : '',  
2060: trecho : 'ENTR BR-116(A) (JUQUIÁ) - ENTR SP-139 (REGISTRO)'  
2061: }, {  
2062: usr : '282',  
2063: pwd : '0130',  
2064: fase : '3',  
2065: ladoA : '',  
2066: ladoB : '',  
2067: trecho : 'ENTR BR-373(A) (ITAPETININGA) - ENTR BR-373(B)'  
2068: }, {  
2069: usr : '283',  
2070: pwd : '1077',  
2071: fase : '3',  
2072: ladoA : '',  
2073: ladoB : '',  
2074: trecho : 'ACESSO GUAIMBÉ - ENTR SP-333(A)'  
2075: }, {  
2076: usr : '284',  
2077: pwd : '1010',  
2078: fase : '2',  
2079: ladoA : '',  
2080: ladoB : '',  
2081: trecho : 'ENTR SP-355 - ENTR SP-425'  
2082: }, {  
2083: usr : '285',  
2084: pwd : '0110',  
2085: fase : '3',  
2086: ladoA : '',  
2087: ladoB : '',  
2088: trecho : 'ENTR SP-255 (ARARAQUARA) - ENTR BR-364(B)'  
2089: }, {  
2090: usr : '286',  
2091: pwd : '0783',  
2092: fase : '3',  
2093: ladoA : '',  
2094: ladoB : '',  
2095: trecho : 'ACESSO RINÓPOLIS - ENTR SP-294 (PARAPUÃ)'  
2096: }, {  
2097: usr : '287',  
2098: pwd : '0850',  
2099: fase : '3',  
2100: ladoA : '',  
2101: ladoB : '',  
2102: trecho : 'ENTR SP-065 (ATIBAIA) - ENTR SP-023 (MARIPORÃ)'  
2103: }, {  
2104: usr : '288',  
2105: pwd : '0010',  
2106: fase : '3',  
2107: ladoA : '',  
2108: ladoB : '',  
2109: trecho : 'ENTR BR-222/230(A)/PA-150 (MARABÁ) - ENTR PA-405'  
2110: }, {  
2111: usr : '289',  
2112: pwd : '0390',  
2113: fase : '2',  
2114: ladoA : '',  
2115: ladoB : '',  
2116: trecho : 'ENTR BR-010(A) (PORTO FRANCO) - ENTR BR-010(B)/230(A)/MA-138 (ESTREITO)'  
2117: }, {  
2118: usr : '290',  
2119: pwd : '1010',  
2120: fase : '3',  
2121: ladoA : '',  
2122: ladoB : '',  
2123: trecho : 'ENTR MA-012/375 (SÃO RAIMUNDO DAS MANGABEIRAS) - ENTR MA-006(A)'  
2124: }, {  
2125: usr : '291',  
2126: pwd : '0290',  
2127: fase : '3',  
2128: ladoA : '',  
2129: ladoB : '',  
2130: trecho : 'FIM PISTA DUPLA - GURUPÍ - ENTR BR-242(B)/TO-280'  
2131: }, {  
2132: usr : '292',  
2133: pwd : '0330',  
2134: fase : '4',  
2135: ladoA : '',  
2136: ladoB : '',  
2137: trecho : 'ENTR BR-020(A)/135(B)/BA-020 - ENTR BR-020(B)'  
2138: }, {  
2139: usr : '293',  
2140: pwd : '0272',  
2141: fase : '3',  
2142: ladoA : '',  
2143: ladoB : '',  
2144: trecho : 'INDIANÓPOLIS - VALE DO SONHO'  
2145: }, {  
2146: usr : '294',  
2147: pwd : '0210',  
2148: fase : '4',  
2149: ladoA : '',  
2150: ladoB : '',  
2151: trecho : 'ENTR GO-217 (P/MAIRIPOTABA) - ENTR GO-320 (INDIARA)'  
2152: }, {  
2153: usr : '295',  
2154: pwd : '0370',  
2155: fase : '1',  
2156: ladoA : 'ENTR BR-153 (P/TIBAGI/IPIRANGA)',  
2157: ladoB : 'ENTR BR-373(A)/487(A) (CAETANO)',  
2158: trecho : 'ENTR BR-153 (P/TIBAGI/IPIRANGA) - ENTR BR-373(A)/487(A) (CAETANO)'  
2159: }, {  
2160: usr : '296',  
2161: pwd : '0390',  
2162: fase : '1',  
2163: ladoA : 'RIO ANHANDUÍ',  
2164: ladoB : 'ENTR BR-262(A) (CAMPO GRANDE)',  
2165: trecho : 'RIO ANHANDUÍ - ENTR BR-262(A) (CAMPO GRANDE)'  
2166: }, {  
2167: usr : '297',  
2168: pwd : '0434',  
2169: fase : '3',  
2170: ladoA : '',  
2171: ladoB : '',  
2172: trecho : 'ACESSO ITAJÁ (GO) - ENTR MS-431 (P/SÃO JOÃO DO APORÉ)'  
2173: }, {  
2174: usr : '298',  
2175: pwd : '0930',  
2176: fase : '3',  
2177: ladoA : '',  
2178: ladoB : '',  
2179: trecho : 'FNM (ENTR PISTA INVERSA) - ENTR BR-116(A)/493/RJ-109'  
2180: }, {  
2181: usr : '299',  
2182: pwd : '0110',  
2183: fase : '1',  
2184: ladoA : 'ENTR SC-418 (P/POMERODE)',  
2185: ladoB : 'ENTR BR-477(B) (P/TIMBÓ)',  
2186: trecho : 'ENTR SC-418 (P/POMERODE) - ENTR BR-477(B) (P/TIMBÓ)'  
2187: }, {  
2188: usr : '300',  
2189: pwd : '0110',  
2190: fase : '1',  
2191: ladoA : 'ENTR BR-116(B)',  
2192: ladoB : 'P/TRES BARRAS',  
2193: trecho : 'ENTR BR-116(B) - P/TRES BARRAS'  
2194: }, {  
2195: usr : '301',  
2196: pwd : '0150',  
2197: fase : '2',  
2198: ladoA : '',  
2199: ladoB : '',  
2200: trecho : 'DIV PI/BA - ENTR BR-235/BA-161 (REMANSO)'  
2201: }, {  
2202: usr : '302',  
2203: pwd : '0770',  
2204: fase : '2',  
2205: ladoA : '',  
2206: ladoB : '',  
2207: trecho : 'ENTR BR-369(A) (BARBOSA FERRAZ) - ENTR BR-158(A)/369(B) (ANEL VIÁRIO CAMPO MOURÃO)'  
2208: } ],  
2209:  
2210: user\_logado : {}  
2211:};

## .\mobile\platforms\ios\www\js\myDb.js

1:myDb = {  
 2:  
 3: camposNaoExportaveisJson : [ 'id', 'uuid', 'login', 'erro', 'duracaoPesq' ],  
 4:  
 5: tabelaOD : [  
 6: {field : 'id', type : 'text primary key'},  
 7: {field : 'estaNoNote', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 8: {field : 'cancelado', type : 'integer'},  
 9: {field : 'idPosto', type : 'integer', notNull: true},  
 10: {field : 'sentido', type : 'text', notNull: true},  
 11: {field : 'idIpad', type : 'text', notNull: true},  
 12: // {field : 'uuid', type : 'text'},  
 13: {field : 'login', type : 'text'}, // idPost + sentido (redundante?)  
 14: {field : 'dataIniPesq', type : 'text'},  
 15: {field : 'dataFimPesq', type : 'text'},  
 16: {field : 'placa', type : 'text'},  
 17: {field : 'anoDeFabricacao', type : 'integer', notNull: true},  
 18: {field : 'tipo', type : 'text', notNull: true},  
 19: {field : 'idOrigemPais', type : 'integer', notNull: true},  
 20: {field : 'idOrigemMunicipio', type : 'integer'},  
 21: {field : 'idDestinoPais', type : 'integer', notNull: true},  
 22: {field : 'idDestinoMunicipio', type : 'integer'},  
 23: {field : 'idMotivoDeEscolhaDaRota', type : 'integer'},  
 24: {field : 'frequenciaQtd', type : 'integer'},  
 25: {field : 'frequenciaPeriodo', type : 'text', notNull: true},  
 26: {field : 'idPropriedadesDoVeiculo', type : 'integer', notNull: true},  
 27: {field : 'placaEstrangeira', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 28: {field : 'idPaisPlacaEstrangeira', type : 'integer'},  
 29: {field : 'idCombustivel', type : 'integer', notNull: true},  
 30: {field : 'categoria', type : 'text', notNull: true},  
 31: {field : 'possuiReboque', type : 'integer'},  
 32: {field : 'numeroDePessoasNoVeiculo', type : 'integer'},  
 33: {field : 'numeroDePessoasATrabalho', type : 'integer'},  
 34: {field : 'idRendaMedia', type : 'integer'},  
 35: {field : 'idMotivoDaViagem', type : 'integer'},  
 36: {field : 'tipoCaminhao', type : 'text'},  
 37: {field : 'idTipoDeContainer', type : 'integer'},  
 38: {field : 'idTipoCarroceria', type : 'integer'},  
 39: {field : 'rntrc', type : 'text'},  
 40: {field : 'possuiCargaPerigosa', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 41: {field : 'idNumeroDeRisco', type : 'integer'},  
 42: {field : 'idNumeroDaOnu', type : 'integer'},  
 43: {field : 'idAgregado', type : 'integer'},  
 44: {field : 'placaVermelha', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 45: {field : 'idTipoDeViagemOuServico', type : 'integer'},  
 46: {field : 'pesoDaCarga', type : 'real'},  
 47: {field : 'valorDoFrete', type : 'real'},  
 48: {field : 'utilizaParadaEspecial', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 49: {field : 'idProduto', type : 'integer'},  
 50: {field : 'idCargaAnterior', type : 'integer'},  
 51: {field : 'valorDaCarga', type : 'real'},  
 52: {field : 'municipioEmbarqueCarga', type : 'integer'},  
 53: {field : 'idLocalEmbarqueCarga', type : 'integer'},  
 54: {field : 'municipioDesembarqueCarga', type : 'integer'},  
 55: {field : 'idLocalDesembarqueCarga', type : 'integer'},  
 56: {field : 'indoPegarCarga', type : 'integer'},  
 57: {field : 'paradaObrigatoriaMunicipio1', type : 'integer'},  
 58: {field : 'paradaObrigatoriaMunicipio2', type : 'integer'},  
 59: {field : 'idPerguntaExtra', type : 'integer'},  
 60: {field : 'idPerguntaExtra2', type : 'integer'},  
 61: {field : 'erro', type : 'text'},  
 62: {field : 'duracaoPesq', type : 'integer'},  
 63: {field : 'treinamento', type : 'integer', notNull: true}// Boolean 1->true || false, otherwise;  
 64: ],  
 65:  
 66: fieldExists : function(str) {  
 67: var found = false;  
 68: $.each(myDb.tabelaOD, function(index, item) {  
 69: if (item.field == str) {  
 70: found = true;  
 71: return false; // breaks $.each  
 72: }  
 73: });  
 74: return found;  
 75: },  
 76:  
 77: cretateTblDados : function(cb) {  
 78: app.logger.log("criando tabela: tblDados");  
 79: app.database.transaction(function(tx) {  
 80: var sql = "CREATE TABLE IF NOT EXISTS tblDados ( ";  
 81: $.each(myDb.tabelaOD, function(index, item) {  
 82: sql += item.field + " " + item.type;  
 83: if (index < myDb.tabelaOD.length - 1)  
 84: sql += ", ";  
 85: });  
 86: sql += ");";  
 87: tx.executeSql(sql);  
 88: }, function(e) {  
 89: app.logger.log('ERRO: ' + e.message);  
 90: }, function() {  
 91: app.logger.log("tabela criada: tblDados");  
 92: if(util.isFunction(cb)){  
 93: cb();  
 94: }  
 95: });  
 96:  
 97: },  
 98:  
 99: /\*\*  
 100: \* Inserts a registro variable into database  
 101: \* @param Registro reg registro to be inserted into database  
 102: \* @param Function fail error callback  
 103: \* @param Function success callback  
 104: \*/  
 105: insertRegistro : function(reg, fail, success) {  
 106: if (reg.id != null){  
 107: app.logger.log("inserindo registro: " + reg.id);  
 108: try {  
 109: app.database.transaction(function(tx) {  
 110:  
 111: var fields = "(";  
 112: var places = "(";  
 113: var values = [];  
 114: $.each(myDb.tabelaOD, function(index, item) {  
 115: fields += item.field;  
 116: places += "?";  
 117: var value = reg[item.field];  
 118: values.push((value == undefined ? null : value));  
 119: if (index < myDb.tabelaOD.length - 1) {  
 120: fields += ", ";  
 121: places += ", ";  
 122: }  
 123: });  
 124: fields += ")";  
 125: places += ");";  
 126:  
 127: var sql = "INSERT INTO tblDados " + fields + " VALUES " + places;  
 128:  
 129: tx.executeSql(sql, values, function(tx, res) {  
 130: app.logger.log('id inserido no banco de dados: ' + res.insertId);  
 131: });  
 132: },  
 133: // transaction fail  
 134: function(e) {  
 135: // inseriu = false;  
 136: app.logger.log('ERRO ao inserir registro (' + reg.id + '): ' + e.message);  
 137: fail(e);  
 138: },  
 139: // transaction success  
 140: function() {  
 141: // inseriu = true;  
 142: app.logger.log("registro inserido: " + reg.id);  
 143: success();  
 144: });  
 145: } catch (e) {  
 146: fail(e);  
 147: }  
 148: } else {  
 149: app.logger.log("Tentativa de inserção de registro sem id");  
 150: }  
 151:   
 152: },  
 153:   
 154: sanitize: function(cb){  
 155: app.database.transaction(function(tx) {  
 156: var sql = "DELETE from tblDados WHERE id is null;";  
 157: tx.executeSql(sql);  
 158: },  
 159: //fail  
 160: function(e){  
 161: app.logger.log('ERRO ao limpar a base de dados: ' + e.message);  
 162: },  
 163: //success  
 164: function(){  
 165: app.logger.log('Base de dados limpa');  
 166: if (util.isFunction(cb)){  
 167: cb();  
 168: }  
 169: })  
 170: },  
 171:  
 172:/\*  
 173:select date(dataIniPesq) as 'diaPesq',AVG(duracaoPesq) as 'mediaDia',SUM(duracaoPesq) as 'somaDia',COUNT(id) as 'qtdDia', MAX(duracaoPesq) as 'maxTempoDia', MIN(duracaoPesq) as 'minTempoDia'  
 174:from tblDados  
 175:WHERE cancelado = 0  
 176:GROUP by date(dataIniPesq)  
 177:ORDER BY diaPesq DESC;  
 178:\*/  
 179: selectDuracoesDiaRegistro : function(fail, success) {  
 180: app.logger.log("(selectDuracoesDiaRegistro) buscando no registro");  
 181: var linhas = [];  
 182: var treinamento = (app.isTreinamento)? 1 : 0;  
 183: try {  
 184: app.database.transaction(function(tx) {  
 185:  
 186: var sql = "SELECT DATE(dataIniPesq) as 'diaPesq'," +  
 187: " AVG(duracaoPesq) as 'mediaDia'," +  
 188: " SUM(duracaoPesq) as 'somaDia',"+  
 189: " COUNT(id) as 'qtdDia'," +  
 190: " MAX(duracaoPesq) as 'maxTempoDia'," +  
 191: " MIN(duracaoPesq) as 'minTempoDia'" +  
 192: " FROM tblDados WHERE cancelado = 0" +  
 193: " AND treinamento = " + treinamento +  
 194: " AND idPosto = " + app.posto +  
 195: " GROUP by DATE(dataIniPesq) ORDER BY diaPesq DESC";  
 196:  
 197: tx.executeSql(sql, [], function(tx, res) {  
 198: app.logger.log('qtd linhas select: ' + res.rows.length);  
 199: for (i = 0; i < res.rows.length; i++) {  
 200: var elem = res.rows.item(i);  
 201: linhas.push(elem);  
 202: app.logger.log('item ' + i + ':' + elem.diaPesq + '; ' + elem.mediaDia + '; ' + elem.somaDia + '; ' + elem.qtdDia + '; ' + elem.maxTempoDia + '; ' + elem.minTempoDia);  
 203: }  
 204: });  
 205: },  
 206: // transaction fail  
 207: function(e) {  
 208: app.logger.log('ERRO ao buscar tempos das pesquisas: ' + e.message);  
 209: fail(e);  
 210: },  
 211: // transaction success  
 212: function() {  
 213: app.logger.log("Busca dos tempos das pesquisas realizado com sucesso");  
 214: app.sumario\_lista = linhas;  
 215: success();  
 216: });  
 217: } catch (e) {  
 218: fail(e);  
 219: }  
 220: },  
 221:  
 222: selectUltimaPesquisaValida : function(fail, success) {  
 223: app.logger.log("(selectUltimaPesquisaValida) buscando no registro");  
 224: var ultimoRegistro = [];  
 225: var treinamento = (app.isTreinamento)? 1 : 0;  
 226: try {  
 227: app.database.transaction(function(tx) {  
 228:  
 229: var sql = "SELECT dataIniPesq," +  
 230: " duracaoPesq" +  
 231: " FROM tblDados WHERE cancelado = 0" +  
 232: " AND treinamento = " + treinamento +  
 233: " AND idPosto = " + app.posto +  
 234: " ORDER BY dataIniPesq DESC" +  
 235: " LIMIT 1";  
 236:  
 237: tx.executeSql(sql, [], function(tx, res) {  
 238: if( res.rows.length > 0) {  
 239: ultimoRegistro.push(res.rows.item(0).dataIniPesq);  
 240: ultimoRegistro.push(res.rows.item(0).duracaoPesq);  
 241: app.logger.log('Ultimo registro: dataIniPesq = ' + ultimoRegistro[0] + '; duracaoPesq = ' + ultimoRegistro[1]);  
 242: }  
 243: });  
 244: },  
 245: // transaction fail  
 246: function(e) {  
 247: app.logger.log('ERRO ao buscar última pesquisa: ' + e.message);  
 248: fail(e);  
 249: },  
 250: // transaction success  
 251: function() {  
 252: app.logger.log("Busca da última pesquisa realizada com sucesso");  
 253: app.ultima\_pesquisa = ultimoRegistro;  
 254: success();  
 255: });  
 256: } catch (e) {  
 257: fail(e);  
 258: }  
 259: },  
 260:  
 261:/\*  
 262:select date(dataIniPesq) as 'diaPesq',AVG(duracaoPesq) as 'mediaDia',SUM(duracaoPesq) as 'somaDia',COUNT(id) as 'qtdDia'  
 263:from tblDados  
 264:WHERE cancelado = 1  
 265:GROUP by date(dataIniPesq)  
 266:ORDER BY diaPesq DESC;  
 267:\*/  
 268: selectRegistrosCancelados : function(fail, success) {  
 269: app.logger.log("(selectRegistrosCancelados) buscando no registro");  
 270: var linhas = [];  
 271: var treinamento = (app.isTreinamento)? 1 : 0;  
 272: try {  
 273: app.database.transaction(function(tx) {  
 274:  
 275: var sql = "SELECT DATE(dataIniPesq) as 'diaPesq'," +  
 276: " AVG(duracaoPesq) as 'mediaDia'," +  
 277: " SUM(duracaoPesq) as 'somaDia',"+  
 278: " COUNT(id) as 'qtdDia'" +  
 279: " FROM tblDados WHERE cancelado = 1" +  
 280: " AND treinamento = " + treinamento +  
 281: " AND idPosto = " + app.posto +  
 282: " GROUP by DATE(dataIniPesq) ORDER BY diaPesq DESC";  
 283:  
 284: tx.executeSql(sql, [], function(tx, res) {  
 285: app.logger.log('qtd linhas select: ' + res.rows.length);  
 286: for (i = 0; i < res.rows.length; i++) {  
 287: var elem = res.rows.item(i);  
 288: linhas.push(elem);  
 289: app.logger.log('item ' + i + ':' + elem.diaPesq + '; ' + elem.mediaDia + '; ' + elem.somaDia + '; ' + elem.qtdDia);  
 290: }  
 291: });  
 292: },  
 293: // transaction fail  
 294: function(e) {  
 295: app.logger.log('ERRO ao buscar pesquisas canceladas: ' + e.message);  
 296: fail(e);  
 297: },  
 298: // transaction success  
 299: function() {  
 300: app.logger.log("Busca das pesquisas canceladas realizada com sucesso");  
 301: app.sumario\_lista\_cancelados = linhas;  
 302: success();  
 303: });  
 304: } catch (e) {  
 305: fail(e);  
 306: }  
 307: },  
 308:  
 309: exportaDbToJson : function(writer, fail, success) {  
 310: app.logger.log("exportando Json: ");  
 311: var treinamento = (app.isTreinamento)? 1 : 0;  
 312: try {  
 313: app.database.transaction(function(tx) {  
 314: var fields = "";  
 315: $.each(myDb.tabelaOD, function(index, item) {  
 316: fields += item.field;  
 317: if (index < myDb.tabelaOD.length - 1) {  
 318: fields += ", ";  
 319: }  
 320: });  
 321:  
 322: var sql = "SELECT " + fields + " FROM tblDados WHERE cancelado = 0 AND treinamento = " + treinamento + " ;";  
 323:  
 324: tx.executeSql(sql, [], function(tx, res) {  
 325: for (var rowIndex = 0; rowIndex < res.rows.length; rowIndex++) {  
 326: var rowDB = res.rows.item(rowIndex);  
 327: var rowJson = "{";  
 328: if (rowIndex == 0) {  
 329: rowJson = "[{";  
 330: }  
 331: $.each(myDb.tabelaOD, function(index, item) {  
 332: if (!util.contains(item.field, myDb.camposNaoExportaveisJson)) {  
 333: var value = rowDB[item.field];  
 334: if (value != null) {  
 335: // app.logger.log('FIELD (' + item.type + '): ' + item.field + '\tVALUE (' + typeof value  
 336: // + '): ' + value);  
 337: var typeValue = typeof value;  
 338: if (item.type == 'text') {  
 339: value = '"' + value + '"';  
 340: } else if (item.type == 'integer' && (typeValue != 'number')) {  
 341: if (typeValue == 'boolean') {  
 342: value = Number(value);  
 343: } else if (typeValue == 'string') {  
 344: value = ((value == 'true') ? 1 : ((value == 'false') ? 0 : (isNaN(value) ? ('"'  
 345: + value + '"') : Number(value))));  
 346: }  
 347: }  
 348: } else if (item.notNull) {  
 349: if (item.type == 'text') {  
 350: value = '""';  
 351: } else {  
 352: value = 0;  
 353: }  
 354: }  
 355: rowJson += '"' + item.field + '":' + value;  
 356: if (index < myDb.tabelaOD.length - 2) { // (length - 2) pq o último campo ('erro') tb não será exportado  
 357: rowJson += ",";  
 358: }  
 359: }  
 360: });  
 361: rowJson += "}";  
 362: if (rowIndex < res.rows.length - 1) {  
 363: rowJson += ",";  
 364: } else {  
 365: rowJson += "]";  
 366: }  
 367: writer.appendRow(rowJson);  
 368: }  
 369: });  
 370:  
 371: },  
 372: // transaction fail  
 373: function(e) {  
 374: app.logger.log('ERRO ao exportar os dados para o formato JSON: ' + e.message);  
 375: if (util.isFunction(fail)) {  
 376: fail(e);  
 377: }  
 378: },  
 379: // transaction success  
 380: function() {  
 381: app.logger.log("Dados exportados para o formato JSON.");  
 382: if (util.isFunction(success)) {  
 383: success();  
 384: }  
 385: });  
 386: } catch (e) {  
 387: if (util.isFunction(fail)) {  
 388: fail(e);  
 389: }  
 390: }  
 391: }  
 392:  
 393:};

## .\mobile\platforms\ios\www\js\produtos.js

1:var lista\_produtos = [  
 2: {id : '1', nome : "Abacaxi (a Granel)"},  
 3: {id : '2', nome : "Abacaxi (Doces e Conservas)"},  
 4: {id : '3', nome : "Abacaxi (Embalado)"},  
 5: {id : '4', nome : "Abre-cartas para escritorio"},  
 6: {id : '5', nome : "Açaí (em Conserva)"},  
 7: {id : '6', nome : "Açaí (In Natura)"},  
 8: {id : '7', nome : "Acessórios de Outros Metais não Ferrosos"},  
 9: {id : '8', nome : "Acessórios e Peças para Instrumentos Musicais"},  
 10: {id : '9', nome : "Açúcar a Granel (Cristal, Demerara e Refinado)"},  
 11: {id : '10', nome : "Açúcar Embalado (Cristal, Demerara e Refinado)"},  
 12: {id : '11', nome : "Açucar Invertido (Xarope)"},  
 13: {id : '12', nome : "Adesivos e Selantes"},  
 14: {id : '13', nome : "Aditivos de Uso Industrial (Quimicos)"},  
 15: {id : '14', nome : "Adubos e Fertilizantes"},  
 16: {id : '15', nome : "Aeronaves"},  
 17: {id : '16', nome : "Agrotóxicos"},  
 18: {id : '17', nome : "Aguardente de Cana (Envasado)"},  
 19: {id : '18', nome : "Aguardente de Cana (Granéis Líquidos)"},  
 20: {id : '19', nome : "Águas Minerais"},  
 21: {id : '20', nome : "Agulhas de Costura, Croche, Trico"},  
 22: {id : '21', nome : "Álcool (Embalado)"},  
 23: {id : '22', nome : "Álcool (Etanol)"},  
 24: {id : '23', nome : "Alfinetes"},  
 25: {id : '24', nome : "Algodão (a Granel)"},  
 26: {id : '25', nome : "Algodão Beneficiado"},  
 27: {id : '26', nome : "Algodão em Caroço (a Granel)"},  
 28: {id : '27', nome : "Alho (a Granel)"},  
 29: {id : '28', nome : "Alho (Embalado)"},  
 30: {id : '29', nome : "Alicates"},  
 31: {id : '30', nome : "Alicates para Unhas"},  
 32: {id : '31', nome : "Alizares e Soleiras de Aço, Aluminio ou Ferro"},  
 33: {id : '32', nome : "Alumínio e suas Ligas"},  
 34: {id : '33', nome : "Alviões (Ferramentas)"},  
 35: {id : '34', nome : "Amendoim (a Granel)"},  
 36: {id : '35', nome : "Amendoim (Doces, Conservas e Salgados)"},  
 37: {id : '36', nome : "Amianto"},  
 38: {id : '37', nome : "Amido de Milho"},  
 39: {id : '38', nome : "Amidos e Féculas"},  
 40: {id : '39', nome : "Amônia"},  
 41: {id : '40', nome : "Ancinhos"},  
 42: {id : '41', nome : "Âncoras Metálicas (Peças e Volumes)"},  
 43: {id : '42', nome : "Andaimes tubulares de Metal"},  
 44: {id : '43', nome : "Animais Domésticos"},  
 45: {id : '44', nome : "Animais vivos (outros)"},  
 46: {id : '45', nome : "Aparelhos de barbear"},  
 47: {id : '46', nome : "Aparelhos de Som e Imagem (televisão, DVD player, blue ray)"},  
 48: {id : '47', nome : "Aparelhos e Equipamentos não-Elétricos para Instalações Térmicas"},  
 49: {id : '48', nome : "Aparelhos e Equipamentos Para Sinalização e Alarme"},  
 50: {id : '49', nome : "Aparelhos/ Instrumentos de Medida"},  
 51: {id : '50', nome : "Aparelhos/ Instrumentos Médico-Hospitalar"},  
 52: {id : '51', nome : "Aparelhos/ Instrumentos Ópticos"},  
 53: {id : '52', nome : "Aplainadeiras"},  
 54: {id : '53', nome : "Arames de Aço"},  
 55: {id : '54', nome : "Arames Farpados (aço, ferro)"},  
 56: {id : '55', nome : "Arames Lisos (aço, ferro)"},  
 57: {id : '56', nome : "Arcos de pua (ferramenta)"},  
 58: {id : '57', nome : "Ardósia"},  
 59: {id : '58', nome : "Areia (a Granel)"},  
 60: {id : '59', nome : "Areia (Embalada)"},  
 61: {id : '60', nome : "Armas brancas (martelos, os machados, etc.)"},  
 62: {id : '61', nome : "Armas e Munições ( revólveres, pistolas, rifles)"},  
 63: {id : '62', nome : "Arroz (a Granel)"},  
 64: {id : '63', nome : "Arroz Beneficiado"},  
 65: {id : '64', nome : "Arroz Beneficiado - Produtos Derivados (pré-cozido, risoto, doce, biscoito)"},  
 66: {id : '65', nome : "Arroz em Casca"},  
 67: {id : '66', nome : "Arruelas não roscados de Aço, Cobre e Ferro"},  
 68: {id : '67', nome : "Artefatos de Couro"},  
 69: {id : '68', nome : "Artefatos de Joalheria e Ourivesaria (jóias)"},  
 70: {id : '69', nome : "Artefatos de Zinco para Construção (Peças e Volumes)"},  
 71: {id : '70', nome : "Artefatos para Caça, Pesca e Esporte"},  
 72: {id : '71', nome : "Artigos de Borracha (outros)"},  
 73: {id : '72', nome : "Artigos de Funilaria e de Metal para Usos Doméstico e Pessoal"},  
 74: {id : '73', nome : "Artigos de Serralheria (exceto esquadrias)"},  
 75: {id : '74', nome : "Artigos plasticos para Escritório"},  
 76: {id : '75', nome : "Artigos plasticos para Higiene"},  
 77: {id : '76', nome : "Artigos plasticos para uso doméstico"},  
 78: {id : '77', nome : "Artigos plasticos para uso escolar"},  
 79: {id : '78', nome : "Assadeiras"},  
 80: {id : '79', nome : "Assentos plasticos"},  
 81: {id : '80', nome : "Automóveis, Camionetas e Utilitários"},  
 82: {id : '81', nome : "Aves Abatidas"},  
 83: {id : '82', nome : "Aves vivas (Outras)"},  
 84: {id : '83', nome : "Avestruz"},  
 85: {id : '84', nome : "Aviamentos Para Costura"},  
 86: {id : '85', nome : "Azulejos e Pisos"},  
 87: {id : '86', nome : "Bacias de Aluminio"},  
 88: {id : '87', nome : "Baionetas"},  
 89: {id : '88', nome : "Baixelas de Aço, Aluminio"},  
 90: {id : '89', nome : "Baixelas de Cobre, Estanho, Ferro"},  
 91: {id : '90', nome : "Balas"},  
 92: {id : '91', nome : "Balcoes e Balaustradas (material metálico)"},  
 93: {id : '92', nome : "Baldes (metal)"},  
 94: {id : '93', nome : "Banana ( a Granel)"},  
 95: {id : '94', nome : "Banana (Doces e Conservas)"},  
 96: {id : '95', nome : "Bandejas (metal)"},  
 97: {id : '96', nome : "Banheiras de ferro ou aço"},  
 98: {id : '97', nome : "Banheiras plasticas"},  
 99: {id : '98', nome : "Barcos"},  
 100: {id : '99', nome : "Barras de Aço"},  
 101: {id : '100', nome : "Barras de Aluminio"},  
 102: {id : '101', nome : "Barras de Outros Metais não Ferrosos"},  
 103: {id : '102', nome : "Barris de Metal"},  
 104: {id : '103', nome : "Barris de Plastico"},  
 105: {id : '104', nome : "Batata (Doces, Conservas e Salgados)"},  
 106: {id : '105', nome : "Batata (a Granel)"},  
 107: {id : '106', nome : "Baterias de Veículos"},  
 108: {id : '107', nome : "Bebidas Outras (Envasadas)"},  
 109: {id : '108', nome : "Bebidas Outras (Granéis Líquidos)"},  
 110: {id : '109', nome : "Bezerros ou Bezerras"},  
 111: {id : '110', nome : "Bicicletas e Triciclos não Motorizados"},  
 112: {id : '111', nome : "Metais para Bidês (metal)"},  
 113: {id : '112', nome : "Bigornas"},  
 114: {id : '113', nome : "Bisnagas flexiveis para embalagem (metal)"},  
 115: {id : '114', nome : "Blocos de ferro"},  
 116: {id : '115', nome : "Bobinas de Aço"},  
 117: {id : '116', nome : "Bodes"},  
 118: {id : '117', nome : "Bois"},  
 119: {id : '118', nome : "Bolachas e Biscoitos"},  
 120: {id : '119', nome : "Bolsas de Plastico"},  
 121: {id : '120', nome : "Bolsas e Cintos"},  
 122: {id : '121', nome : "Bovinos vivos"},  
 123: {id : '122', nome : "Braçadeiras"},  
 124: {id : '123', nome : "Brinquedos e outros Jogos Recreativos"},  
 125: {id : '124', nome : "Búfalos"},  
 126: {id : '125', nome : "Cabines, Carrocerias e Reboques"},  
 127: {id : '126', nome : "Fios, Cabos e Condutores Elétricos Isolados"},  
 128: {id : '127', nome : "Cabos para facas"},  
 129: {id : '128', nome : "Cabras"},  
 130: {id : '129', nome : "Cabritos e Cabritas"},  
 131: {id : '130', nome : "Cacau"},  
 132: {id : '131', nome : "Cachorros"},  
 133: {id : '132', nome : "Cadeados"},  
 134: {id : '133', nome : "Café em grão"},  
 135: {id : '134', nome : "Café solúvel"},  
 136: {id : '135', nome : "Café torrado e moído"},  
 137: {id : '136', nome : "Caixa de descarga de plastico"},  
 138: {id : '137', nome : "Caixas de Correio de Ferro e Aço Fundidas"},  
 139: {id : '138', nome : "Caixas de Metal"},  
 140: {id : '139', nome : "Caixas e Caixotes de Plastico"},  
 141: {id : '140', nome : "Caixas plasticas"},  
 142: {id : '141', nome : "Cajú (a Granel)"},  
 143: {id : '142', nome : "Cajú (Doces, Conservas e Salgados)"},  
 144: {id : '143', nome : "Cal"},  
 145: {id : '144', nome : "Cal hidratado"},  
 146: {id : '145', nome : "Cal Virgem"},  
 147: {id : '146', nome : "Calçados e Tênis"},  
 148: {id : '147', nome : "Calcário"},  
 149: {id : '148', nome : "Calcário Agrícola"},  
 150: {id : '149', nome : "Calcário Siderúrgico"},  
 151: {id : '150', nome : "Caldeiras"},  
 152: {id : '151', nome : "Calhas Metálicas (Peças e Volumes)"},  
 153: {id : '152', nome : "Camarões"},  
 154: {id : '153', nome : "Caminhões"},  
 155: {id : '154', nome : "Campainhas Metálicas (Peças e Volumes)"},  
 156: {id : '155', nome : "Cana-de-açúcar"},  
 157: {id : '156', nome : "Canecas de plastico"},  
 158: {id : '157', nome : "Canecas para cozinha (metal, aço, ferro, aluminio)"},  
 159: {id : '158', nome : "Canecas plasticas"},  
 160: {id : '159', nome : "Canos de Ferro, Aço"},  
 161: {id : '160', nome : "Canos e Perfis Ocos (ferro, aç, inox)"},  
 162: {id : '161', nome : "Cantoneiras de plastico"},  
 163: {id : '162', nome : "Capacetes de plastico"},  
 164: {id : '163', nome : "Caprinos"},  
 165: {id : '164', nome : "Capsulas Plasticas"},  
 166: {id : '165', nome : "Carga fracionada"},  
 167: {id : '166', nome : "Carne bovina"},  
 168: {id : '167', nome : "Carne de aves"},  
 169: {id : '168', nome : "Carne de Bode"},  
 170: {id : '169', nome : "Carne de Cabra"},  
 171: {id : '170', nome : "Carne de Ovelha"},  
 172: {id : '171', nome : "Carne de Ovinos e Caprinos"},  
 173: {id : '172', nome : "Carne suína"},  
 174: {id : '173', nome : "Carneiros Abatidos (resfriado, frigorificado, congelado)"},  
 175: {id : '174', nome : "Carnes (outras)"},  
 176: {id : '175', nome : "Carretéis de Plastico"},  
 177: {id : '177', nome : "Carrocerias para Ônibus"},  
 178: {id : '178', nome : "Cartuchos de Plastico"},  
 179: {id : '179', nome : "Carvão Mineral"},  
 180: {id : '180', nome : "Carvão vegetal a granel"},  
 181: {id : '181', nome : "Carvão vegetal embalado"},  
 182: {id : '182', nome : "Cascalho"},  
 183: {id : '183', nome : "Castanha de Caju (a Granel)"},  
 184: {id : '184', nome : "Castanha de Caju (Conservas)"},  
 185: {id : '185', nome : "Castanha de Caju (Embalada)"},  
 186: {id : '186', nome : "Castanha do Pará (a Granel)"},  
 187: {id : '187', nome : "Castanha do Pará (Conservas)"},  
 188: {id : '188', nome : "Castanha do Pará (Embalada)"},  
 189: {id : '189', nome : "Catalisadores"},  
 190: {id : '190', nome : "Cavaco de madeira"},  
 191: {id : '191', nome : "Cavalos"},  
 192: {id : '192', nome : "Cebola (a Granel)"},  
 193: {id : '193', nome : "Cebola (Conservas)"},  
 194: {id : '194', nome : "Cebola (Embalada)"},  
 195: {id : '195', nome : "Celulose"},  
 196: {id : '196', nome : "Centrífugas (Eletrodoméstico)"},  
 197: {id : '197', nome : "Cerâmicas"},  
 198: {id : '198', nome : "Cerveja e Chopp (Tanque)"},  
 199: {id : '199', nome : "Cervejas (Em Barris)"},  
 200: {id : '200', nome : "Cervejas (Envasadas)"},  
 201: {id : '201', nome : "Cestas de plastico para lixo"},  
 202: {id : '202', nome : "Chá e outras Ervas para Infusão"},  
 203: {id : '203', nome : "Chafarizes para Moinhos de Ferro e Aço Fundidas"},  
 204: {id : '204', nome : "Chapas de Aço"},  
 205: {id : '205', nome : "Chapas de Aluminio"},  
 206: {id : '206', nome : "Chapas de Granito"},  
 207: {id : '207', nome : "Chapas de Outros Metais não Ferrosos"},  
 208: {id : '208', nome : "Chapas de Plástico"},  
 209: {id : '209', nome : "Chapas e tiras, distendidas, de cobre e niquel"},  
 210: {id : '210', nome : "Chapas e tiras, distendidas, de ferro e aço"},  
 211: {id : '211', nome : "Charutos"},  
 212: {id : '212', nome : "Chassis com Motor para Automóveis, Camionetas e Utilitários"},  
 213: {id : '213', nome : "Chaves"},  
 214: {id : '214', nome : "Chaves de caixa intercambiáveis"},  
 215: {id : '215', nome : "Chaves de fenda"},  
 216: {id : '216', nome : "Chaves Metálicas (Embalados)"},  
 217: {id : '217', nome : "Chaves para porcas"},  
 218: {id : '218', nome : "Chester"},  
 219: {id : '219', nome : "Chinelos e Sandálias"},  
 220: {id : '220', nome : "Chocolates e Doces"},  
 221: {id : '221', nome : "Chopp (Em Barris)"},  
 222: {id : '222', nome : "Chumbo"},  
 223: {id : '223', nome : "Cigarros"},  
 224: {id : '224', nome : "Cimento a granel"},  
 225: {id : '225', nome : "Cimento ensacado"},  
 226: {id : '226', nome : "Escória de cimento"},  
 227: {id : '227', nome : "Cinzas de Aluminio"},  
 228: {id : '228', nome : "Cinzas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 229: {id : '229', nome : "Cisalhas para metais (ferramenta)"},  
 230: {id : '230', nome : "Clarabóias Metálicas (Peças e Volumes)"},  
 231: {id : '231', nome : "Classificadores Metálicos (Embalados)"},  
 232: {id : '232', nome : "Clinquer"},  
 233: {id : '233', nome : "Clipes Metálicos (Embalados)"},  
 234: {id : '234', nome : "Cloreto de Potássio"},  
 235: {id : '235', nome : "Cloro e Álcalis"},  
 236: {id : '236', nome : "Coalhadas"},  
 237: {id : '237', nome : "Cobalto"},  
 238: {id : '238', nome : "Cobre"},  
 239: {id : '239', nome : "Côco (a Granel)"},  
 240: {id : '240', nome : "Côco (Doces e Conservas)"},  
 241: {id : '241', nome : "Codornas"},  
 242: {id : '242', nome : "Cofres e Caixas de Segurança e Artefatos Semelhantes Metálicos (Peças e Volumes)"},  
 243: {id : '243', nome : "Colheres"},  
 244: {id : '244', nome : "Comadres de Aluminio"},  
 245: {id : '245', nome : "Computadores e Notebooks, Impressoras"},  
 246: {id : '246', nome : "Conchas para cozinha"},  
 247: {id : '247', nome : "Condensadores de máquinas à vapor"},  
 248: {id : '248', nome : "Cabos Elétricos"},  
 249: {id : '249', nome : "Conexões de Aluminio"},  
 250: {id : '250', nome : "Conexões para encanamento (metal, ferro, aço, inox)"},  
 251: {id : '251', nome : "Conexões para encanamento (plástico, pvc)"},  
 252: {id : '252', nome : "Conservas e Doces de Outros Legumes e Vegetais"},  
 253: {id : '253', nome : "Construcoes pre-fabricadas de Ferro e Aço"},  
 254: {id : '254', nome : "Construcoes pré-fabricadas de Madeira"},  
 255: {id : '255', nome : "Copa (Salsicharia)"},  
 256: {id : '256', nome : "Copos de Alumínio"},  
 257: {id : '257', nome : "Copos e outros pequenos Artefatos de Vidro"},  
 258: {id : '258', nome : "Copos plasticos"},  
 259: {id : '259', nome : "Coque (Carvão)"},  
 260: {id : '260', nome : "Correias de transmissão - Plastico"},  
 261: {id : '261', nome : "Correio, Malote e Demais Serviços de Entrega Expressa"},  
 262: {id : '262', nome : "Correntes cortantes para serra"},  
 263: {id : '263', nome : "Cortadores para unhas"},  
 264: {id : '264', nome : "Corta-papeis para escritorio"},  
 265: {id : '265', nome : "Corta-pinos (ferramenta)"},  
 266: {id : '266', nome : "Corta-tubos (ferramenta)"},  
 267: {id : '267', nome : "Cotovelos de Aluminio"},  
 268: {id : '268', nome : "Cotovelos para Tubos de Ferro e Aço"},  
 269: {id : '269', nome : "Couros"},  
 270: {id : '270', nome : "Cromo"},  
 271: {id : '271', nome : "Crustáceos e Moluscos"},  
 272: {id : '272', nome : "Crustáceos industrializado"},  
 273: {id : '273', nome : "Cubas e recipientes"},  
 274: {id : '274', nome : "Cumeeiras Metálicas (Peças e Volumes)"},  
 275: {id : '275', nome : "Defensivos agrícolas (agrotóxicos)"},  
 276: {id : '276', nome : "Dendê (a Granel)"},  
 277: {id : '277', nome : "Desperdícios de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 278: {id : '278', nome : "Dextrina"},  
 279: {id : '279', nome : "Dobradiças"},  
 280: {id : '280', nome : "Eguas"},  
 281: {id : '281', nome : "Eletrodomésticos"},  
 282: {id : '282', nome : "Eletrodos"},  
 283: {id : '283', nome : "Embalagens de Papel ou Papelão"},  
 284: {id : '284', nome : "Embalagens de Plástico"},  
 285: {id : '285', nome : "Embalagens de Vidro"},  
 286: {id : '286', nome : "Embalagens Metálicas"},  
 287: {id : '287', nome : "Embalagens Plasticas"},  
 288: {id : '288', nome : "Embarcações"},  
 289: {id : '289', nome : "Engradados de Plastico"},  
 290: {id : '290', nome : "Enlatados"},  
 291: {id : '291', nome : "Entulho"},  
 292: {id : '292', nome : "Enxofre"},  
 293: {id : '293', nome : "Equinos"},  
 294: {id : '294', nome : "Equipamento Bélico (grande porte)"},  
 295: {id : '295', nome : "Equipamentos de Ar Condicionado"},  
 296: {id : '296', nome : "Equipamentos de Informática"},  
 297: {id : '297', nome : "Erva-Mate (a Granel)"},  
 298: {id : '298', nome : "Erva-Mate (Embalada)"},  
 299: {id : '299', nome : "Escadas Metálicas (Peças e Volumes)"},  
 300: {id : '300', nome : "Escamas de Aluminio"},  
 301: {id : '301', nome : "Escamas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 302: {id : '302', nome : "Escórias de Metal (restos de funilaria)"},  
 303: {id : '303', nome : "Escorredores de Louças"},  
 304: {id : '304', nome : "Escovas, Pincéis e Vassouras"},  
 305: {id : '305', nome : "Escumadeiras para cozinha"},  
 306: {id : '306', nome : "Esferas para Moinhos de Ferro e Aço Fundidas"},  
 307: {id : '307', nome : "Espadas"},  
 308: {id : '308', nome : "Espatulas para escritorio"},  
 309: {id : '309', nome : "Espatulas para Unhas"},  
 310: {id : '310', nome : "Esponjas de Aço para limpeza e polimento"},  
 311: {id : '311', nome : "Espumas de Poliuretano"},  
 312: {id : '312', nome : "Esquadrias de Metal"},  
 313: {id : '313', nome : "Esquadrias e Venezianas de Madeira"},  
 314: {id : '314', nome : "Estamparia de Metais"},  
 315: {id : '315', nome : "Estanho"},  
 316: {id : '316', nome : "Estatuetas e Outros Objetos Metálicos para Ornamentação (Embalados)"},  
 317: {id : '317', nome : "Estruturas de Farois Maritimos"},  
 318: {id : '318', nome : "Estruturas Metálicas"},  
 319: {id : '319', nome : "Estruturas para navios"},  
 320: {id : '320', nome : "Estruturas Pré-Moldadas de Concreto Armado, em Série ou Sob Encomenda"},  
 321: {id : '321', nome : "Estufas Elétricas Industriais"},  
 322: {id : '322', nome : "Artefatos de trefilados metálicos (peças pequenas para industria, residencia, escritorio)"},  
 323: {id : '323', nome : "Artefatos diversos de plástico (escritorio, doméstico, uso industrial)"},  
 324: {id : '324', nome : "Outros Produtos Alimentícios"},  
 325: {id : '325', nome : "Facas cortantes para máquinas"},  
 326: {id : '326', nome : "Facas e Artigos de Cutelaria"},  
 327: {id : '327', nome : "Facas para uso profissional (ferramenta)"},  
 328: {id : '328', nome : "Farelo de Soja"},  
 329: {id : '329', nome : "Farinha de Mandioca"},  
 330: {id : '330', nome : "Farinha de Milho"},  
 331: {id : '331', nome : "Farinha de Trigo"},  
 332: {id : '332', nome : "Fateixas Metálicas e Suas Partes (Peças e Volumes)"},  
 333: {id : '333', nome : "Fechaduras"},  
 334: {id : '334', nome : "Feijão (em Sacos)"},  
 335: {id : '335', nome : "Feijão Beneficiado (Embalado)"},  
 336: {id : '336', nome : "Fermentos, Leveduras e Coalhos"},  
 337: {id : '337', nome : "Ferragens para Construção Civil"},  
 338: {id : '338', nome : "Ferragens para Encadernação Metálicos (Embalados)"},  
 339: {id : '339', nome : "Ferragens para Linhas Elétricas"},  
 340: {id : '340', nome : "Ferramentas de Chanfrar"},  
 341: {id : '341', nome : "Ferramentas de Fresar"},  
 342: {id : '342', nome : "Ferramentas de moldurar, ranchurar"},  
 343: {id : '343', nome : "Ferramentas de roscar"},  
 344: {id : '344', nome : "Ferramentas Manuais"},  
 345: {id : '345', nome : "Ferramentas para jardinagem"},  
 346: {id : '346', nome : "Ferramentas para pedreiros"},  
 347: {id : '347', nome : "Ferramentas para tornear"},  
 348: {id : '348', nome : "Ferrolhos"},  
 349: {id : '349', nome : "Ferroligas"},  
 350: {id : '350', nome : "Fertilizantes"},  
 351: {id : '351', nome : "Fichários Metálicos (Embalados)"},  
 352: {id : '352', nome : "Fio-máquina de Aço"},  
 353: {id : '353', nome : "Fios de Aluminio"},  
 354: {id : '354', nome : "Fios de níquel e de fios de ferro e aço"},  
 355: {id : '355', nome : "Fios e Folhas de Outros Metais não Ferrosos"},  
 356: {id : '356', nome : "Fios e têxteis"},  
 357: {id : '357', nome : "Fios Elétricos"},  
 358: {id : '358', nome : "Flores e Plantas Ornamentais"},  
 359: {id : '359', nome : "Fogões"},  
 360: {id : '360', nome : "Fogos de Artifício"},  
 361: {id : '361', nome : "Foices"},  
 362: {id : '362', nome : "Folhas de Aluminio"},  
 363: {id : '363', nome : "Folhas de Plástico"},  
 364: {id : '364', nome : "Folhas de serra (ferramenta)"},  
 365: {id : '365', nome : "Folhas-de-flandres de Aço"},  
 366: {id : '366', nome : "Forcados (ferramenta)"},  
 367: {id : '367', nome : "Forjados de Aço"},  
 368: {id : '368', nome : "Forjados de Metais Não Ferrosos"},  
 369: {id : '369', nome : "Forjaria de Metais"},  
 370: {id : '370', nome : "Forjas portateis"},  
 371: {id : '371', nome : "Formões"},  
 372: {id : '372', nome : "Fornos Elétricos Industriais"},  
 373: {id : '373', nome : "Fornos Industriais não-Elétricos"},  
 374: {id : '374', nome : "Forquilhas (ferramenta)"},  
 375: {id : '375', nome : "Fosfato"},  
 376: {id : '376', nome : "Fósforos de Segurança"},  
 377: {id : '377', nome : "Frango Abatido (resfriado, frigorificado, congelado)"},  
 378: {id : '378', nome : "Frangos para Corte"},  
 379: {id : '379', nome : "Frascos Plasticos"},  
 380: {id : '380', nome : "Frigideiras"},  
 381: {id : '381', nome : "Frutose"},  
 382: {id : '382', nome : "Fumo em folha"},  
 383: {id : '383', nome : "Furadeiras"},  
 384: {id : '384', nome : "Furadores para bordar"},  
 385: {id : '385', nome : "Galinhas"},  
 386: {id : '386', nome : "Galos"},  
 387: {id : '387', nome : "Ganchos e Pinos"},  
 388: {id : '388', nome : "Garfos"},  
 389: {id : '389', nome : "Garrafas e Garrafoes de Plastico"},  
 390: {id : '390', nome : "Gás de cozinha"},  
 391: {id : '391', nome : "Gás liquefeito de petróleo (GLP)"},  
 392: {id : '392', nome : "Gases Industriais"},  
 393: {id : '393', nome : "Gasoálcool"},  
 394: {id : '394', nome : "Gasolina"},  
 395: {id : '395', nome : "Gatos"},  
 396: {id : '396', nome : "Geladeiras"},  
 397: {id : '397', nome : "Gêlo / alimentos e sucos congelados"},  
 398: {id : '398', nome : "Gesso"},  
 399: {id : '399', nome : "Gipsita"},  
 400: {id : '400', nome : "Glicose"},  
 401: {id : '401', nome : "Glútem de Trigo"},  
 402: {id : '402', nome : "Gonzos Metálicas (Peças e Volumes)"},  
 403: {id : '403', nome : "Grades de Esgotos de Ferro e Aço Fundidas"},  
 404: {id : '404', nome : "Grades e redes de fios de cobre"},  
 405: {id : '405', nome : "Grampeadores e Semelhantes Metálicos (Embalados)"},  
 406: {id : '406', nome : "Grampos em Barretas e Artefatos Semelhantes Metálicos (Embalados)"},  
 407: {id : '407', nome : "Granalhas de Ferro"},  
 408: {id : '408', nome : "Granalhas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 409: {id : '409', nome : "Grandes Estruturas e Peças de Caldeiraria Pesada"},  
 410: {id : '410', nome : "Granitos"},  
 411: {id : '411', nome : "Gravuras e Semelhantes Metálicos (Embalados)"},  
 412: {id : '412', nome : "Guaraná (In Natura)"},  
 413: {id : '413', nome : "Guaraná (Pó, Sementes e Doces)"},  
 414: {id : '414', nome : "Guaraná (Refrigerante)"},  
 415: {id : '415', nome : "Guarniçoes de Aço"},  
 416: {id : '416', nome : "Hélices Metálicas para Embarcações (Peças e Volumes)"},  
 417: {id : '417', nome : "Hortaliças (a Granel)"},  
 418: {id : '418', nome : "Hortaliças (Embalado)"},  
 419: {id : '419', nome : "Instrumentos Musicais"},  
 420: {id : '420', nome : "Intermediários para Fertilizantes (Outros)"},  
 421: {id : '421', nome : "Iogurtes"},  
 422: {id : '422', nome : "Jogos Recreativos"},  
 423: {id : '423', nome : "Jornais e Revistas"},  
 424: {id : '424', nome : "Jumentos e Jumentas"},  
 425: {id : '425', nome : "Juta"},  
 426: {id : '426', nome : "Lagostas"},  
 427: {id : '427', nome : "Lagostins"},  
 428: {id : '428', nome : "Laminados de Alumínio"},  
 429: {id : '429', nome : "Laminados de Outros Metais não Ferrosos"},  
 430: {id : '430', nome : "Laminados e Semi Acabados de Aço"},  
 431: {id : '431', nome : "Laminados e Semi Acabados de Aço Inoxidável"},  
 432: {id : '432', nome : "Laminados e Semi Acabados de Ferro"},  
 433: {id : '433', nome : "Laminados e Semi Acabados de Outras Ligas de Aço"},  
 434: {id : '434', nome : "Laminados Planos de Aço"},  
 435: {id : '435', nome : "Laminados Planos e Tubulares de Plástico"},  
 436: {id : '436', nome : "Laminas cortantes para máquinas"},  
 437: {id : '437', nome : "Lâminas de Plástico"},  
 438: {id : '438', nome : "Laminas para barbear"},  
 439: {id : '439', nome : "Laminas para facas"},  
 440: {id : '440', nome : "Lâmpadas"},  
 441: {id : '441', nome : "Lâmpadas, lamparina para soldagem"},  
 442: {id : '442', nome : "Lanças"},  
 443: {id : '443', nome : "Lanchas"},  
 444: {id : '444', nome : "Laranja (a Granel)"},  
 445: {id : '445', nome : "Laranja (Doces e Conservas)"},  
 446: {id : '446', nome : "Laranja (Embalada)"},  
 447: {id : '447', nome : "Latas de Aluminio, Aço, Ferro"},  
 448: {id : '448', nome : "Látex (Borracha)"},  
 449: {id : '449', nome : "Laticínios (Derivados do Leite)"},  
 450: {id : '450', nome : "Lavatórios de ferro ou aço"},  
 451: {id : '451', nome : "Lavatórios de plastico"},  
 452: {id : '452', nome : "Leite em Pó"},  
 453: {id : '453', nome : "Leite In Natura (em Latões ou Tarros)"},  
 454: {id : '454', nome : "Leite longa vida"},  
 455: {id : '455', nome : "Leite Pasteurizado (Envasado ou Ensacado)"},  
 456: {id : '456', nome : "Leite Pasteurizado (Granéis Líquidos)"},  
 457: {id : '457', nome : "Leitões e Leitoas vivos"},  
 458: {id : '458', nome : "Letras, Números e Sinais diversos Metálicos (Embalados)"},  
 459: {id : '459', nome : "Limão (a Granel)"},  
 460: {id : '460', nome : "Limão (Doces e Conservas)"},  
 461: {id : '461', nome : "Limão (Embalado)"},  
 462: {id : '462', nome : "Limas para afiar"},  
 463: {id : '463', nome : "Lingotes de Alumínio"},  
 464: {id : '464', nome : "Lingotes de Ferro"},  
 465: {id : '465', nome : "Lingotes de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 466: {id : '466', nome : "Linhas e Fios Para Costurar e Bordar"},  
 467: {id : '467', nome : "Liquidificadores"},  
 468: {id : '468', nome : "Lítio"},  
 469: {id : '469', nome : "Livros"},  
 470: {id : '470', nome : "Lixo Doméstico"},  
 471: {id : '471', nome : "Lixos e Resíduos Hospitalares"},  
 472: {id : '472', nome : "Lixos e Resíduos Industriais"},  
 473: {id : '473', nome : "Locomotivas"},  
 474: {id : '474', nome : "Louças Sanitárias"},  
 475: {id : '475', nome : "Luminárias e Equipamentos de Iluminação - Exceto Para Veículos"},  
 476: {id : '476', nome : "Maçã (a Granel)"},  
 477: {id : '477', nome : "Maçã (Doces e Conservas)"},  
 478: {id : '478', nome : "Maçã (Embalada)"},  
 479: {id : '479', nome : "Maçaricos"},  
 480: {id : '480', nome : "Macarrão e Massas Alimentícias"},  
 481: {id : '481', nome : "Machados"},  
 482: {id : '482', nome : "Madeira (Lenha)"},  
 483: {id : '483', nome : "Madeira Desdobrada (serrada)"},  
 484: {id : '484', nome : "Madeira em toras"},  
 485: {id : '485', nome : "Madeira Industrializada (Compensados e Aglomerados)"},  
 486: {id : '486', nome : "Madeira para Indústria de Celulose"},  
 487: {id : '487', nome : "Malte"},  
 488: {id : '488', nome : "Mamão (a Granel)"},  
 489: {id : '489', nome : "Mamão (Doces e Conservas)"},  
 490: {id : '490', nome : "Mamão (Embalado)"},  
 491: {id : '491', nome : "Mandioca ou Aipim (a Granel)"},  
 492: {id : '492', nome : "Mandioca ou Aipim (Embalada)"},  
 493: {id : '493', nome : "Manga (a Granel)"},  
 494: {id : '494', nome : "Manga (Doces e Conservas)"},  
 495: {id : '495', nome : "Manga (Embalada)"},  
 496: {id : '496', nome : "Manganês"},  
 497: {id : '497', nome : "Manteiga"},  
 498: {id : '498', nome : "Máquinas agrícolas"},  
 499: {id : '499', nome : "Máquinas de Apontar Lápis Metálicos (Embalados)"},  
 500: {id : '500', nome : "Máquinas de Lavar e Secar para Uso Doméstico"},  
 501: {id : '501', nome : "Máquinas e equipamentos (outros)"},  
 502: {id : '502', nome : "Máquinas para Escritório"},  
 503: {id : '503', nome : "Maracujá (a Granel)"},  
 504: {id : '504', nome : "Maracujá (Doces e Conservas)"},  
 505: {id : '505', nome : "Maracujá (Embalado)"},  
 506: {id : '506', nome : "Margarina"},  
 507: {id : '507', nome : "Mariscos"},  
 508: {id : '508', nome : "Mármores"},  
 509: {id : '509', nome : "Marretas"},  
 510: {id : '510', nome : "Martelos"},  
 511: {id : '511', nome : "Materiais de construção (outros)"},  
 512: {id : '512', nome : "Materiais de escritório (outros)"},  
 513: {id : '513', nome : "Materiais Elétricos (outros)"},  
 514: {id : '514', nome : "Materiais Impressos"},  
 515: {id : '515', nome : "Material Elétrico para Veículos (outros)"},  
 516: {id : '516', nome : "Material Eletrônico Básico"},  
 517: {id : '517', nome : "Material para Andaimes, Armacoes e Escoramento"},  
 518: {id : '518', nome : "Medicamentos Para Uso Humano"},  
 519: {id : '519', nome : "Medicamentos Para Uso Veterinário"},  
 520: {id : '520', nome : "Melancia (a Granel)"},  
 521: {id : '521', nome : "Melancia (Doces e Conservas)"},  
 522: {id : '522', nome : "Melancia (Embalada)"},  
 523: {id : '523', nome : "Melão (a Granel)"},  
 524: {id : '524', nome : "Melão (Doces e Conservas)"},  
 525: {id : '525', nome : "Melão (Embalado)"},  
 526: {id : '526', nome : "Metalurgia de Metais"},  
 527: {id : '527', nome : "Metalurgia de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 528: {id : '528', nome : "Milho (a Granel)"},  
 529: {id : '529', nome : "Milho (Doces, Conservas e Salgados)"},  
 530: {id : '530', nome : "Milho (Embalado)"},  
 531: {id : '531', nome : "Minerais (outros)"},  
 532: {id : '532', nome : "Minério de Alumínio (Bauxita)"},  
 533: {id : '533', nome : "Minério de Ferro"},  
 534: {id : '534', nome : "Molas de cobre e de níquel"},  
 535: {id : '535', nome : "Molas e folhas de molas de ferro e aço"},  
 536: {id : '536', nome : "Moldadores (ferramenta)"},  
 537: {id : '537', nome : "Molduras e Semelhantes Metálicos (Embalados)"},  
 538: {id : '538', nome : "Molhos, Temperos e Condimentos"},  
 539: {id : '539', nome : "Moluscos industrializado"},  
 540: {id : '540', nome : "Morango (a Granel)"},  
 541: {id : '541', nome : "Morango (Doces e Conservas)"},  
 542: {id : '542', nome : "Morango (Embalado)"},  
 543: {id : '543', nome : "Mortadela"},  
 544: {id : '544', nome : "Mos com armação (ferramenta)"},  
 545: {id : '545', nome : "Motocicletas"},  
 546: {id : '546', nome : "Motores Elétricos, Inclusive Peças"},  
 547: {id : '547', nome : "Motores para Automóveis, Camionetas e Utilitários"},  
 548: {id : '548', nome : "Motores para Caminhões e Ônibus"},  
 549: {id : '549', nome : "Motores, Bombas e Compressores para Máquinas e Equipamentos"},  
 550: {id : '550', nome : "Moveis de Aço, Aluminio ou Ferro"},  
 551: {id : '551', nome : "Móveis de Madeira"},  
 552: {id : '552', nome : "Mulas e Burros"},  
 553: {id : '553', nome : "Navalhas para barbear"},  
 554: {id : '554', nome : "Nióbio"},  
 555: {id : '555', nome : "Níquel"},  
 556: {id : '556', nome : "Novilhos ou Novilhas"},  
 557: {id : '557', nome : "Óleo combustível"},  
 558: {id : '558', nome : "Óleo de Algodão (Granéis Líquidos)"},  
 559: {id : '559', nome : "Óleo de Algodão (Envasado)"},  
 560: {id : '560', nome : "Óleo de Amendoim (Granéis Líquidos)"},  
 561: {id : '561', nome : "Óleo de Amendoim (Envasado)"},  
 562: {id : '562', nome : "Óleo de Dendê (Granéis Líquidos)"},  
 563: {id : '563', nome : "Óleo de Dendê (Envasado)"},  
 564: {id : '564', nome : "Óleo de Girassol (Granéis Líquidos)"},  
 565: {id : '565', nome : "Óleo de Girassol (Envasado)"},  
 566: {id : '566', nome : "Óleo de Mamona"},  
 567: {id : '567', nome : "Óleo de Milho (Envasado)"},  
 568: {id : '568', nome : "Óleo de Milho (Granéis Líquidos)"},  
 569: {id : '569', nome : "Óleo de Soja (Envasado)"},  
 570: {id : '570', nome : "Óleo de Soja (Granéis Líquidos)"},  
 571: {id : '571', nome : "Óleo Diesel"},  
 572: {id : '572', nome : "Óleos envazados (milho, amidos e féculas vegetais)"},  
 573: {id : '573', nome : "Óleos envasados (vegetal e animal)"},  
 574: {id : '574', nome : "Óleos outros (Granéis Líquidos)"},  
 575: {id : '575', nome : "Ônibus"},  
 576: {id : '576', nome : "Ostras"},  
 577: {id : '577', nome : "Ouro"},  
 578: {id : '578', nome : "Outras Frutas (Doces e Conservas)"},  
 579: {id : '579', nome : "Outras Frutas (a Granel)"},  
 580: {id : '580', nome : "Outras Pedras em Placas"},  
 581: {id : '581', nome : "Outros Animais Domésticos"},  
 582: {id : '582', nome : "Outros Produtos Metálicos (Embalados)"},  
 583: {id : '583', nome : "Outros Produtos Metálicos (Peças e Volumes)"},  
 584: {id : '584', nome : "Ovinos"},  
 585: {id : '585', nome : "Ovos de galinha e de outras aves"},  
 586: {id : '586', nome : "Pães e Bolos Industrializados"},  
 587: {id : '587', nome : "Palha (lã) de ferro e aço"},  
 588: {id : '588', nome : "Palmito (In Natura)"},  
 589: {id : '589', nome : "Palmito em Conserva (embalado)"},  
 590: {id : '590', nome : "Panelas"},  
 591: {id : '591', nome : "Papel e papelão e artefatos"},  
 592: {id : '592', nome : "Parafusos"},  
 593: {id : '593', nome : "Pás"},  
 594: {id : '594', nome : "Peças de Caldeiraria Pesada"},  
 595: {id : '595', nome : "Peças de plastico"},  
 596: {id : '597', nome : "Peças e acessórios para automóveis"},  
 597: {id : '598', nome : "Peças e acessórios para caminhões"},  
 598: {id : '599', nome : "Peças e Acessórios para Ferrovias"},  
 599: {id : '600', nome : "Peças e acessórios para motocicletas"},  
 600: {id : '601', nome : "Peças e acessórios para ônibus"},  
 601: {id : '602', nome : "Peças e acessórios para veículos automotores"},  
 602: {id : '603', nome : "Peças e acessórios para veículos utilitários"},  
 603: {id : '604', nome : "Peças Fundidas de Ferro e Aço"},  
 604: {id : '605', nome : "Peças Fundidas de Metais Não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 605: {id : '606', nome : "Peças e equipamentos para Embarcações"},  
 606: {id : '607', nome : "Peças para Reatores nucleares"},  
 607: {id : '608', nome : "Pedra Britada"},  
 608: {id : '609', nome : "Pedras para Revestimento/ Calçamento (Piso)"},  
 609: {id : '610', nome : "Peito de Perú"},  
 610: {id : '611', nome : "Peixe industrializado"},  
 611: {id : '612', nome : "Peixes"},  
 612: {id : '613', nome : "Peixes vivos"},  
 613: {id : '614', nome : "Películas de Plástico"},  
 614: {id : '615', nome : "Pellets Cítricos (Farinhas)"},  
 615: {id : '616', nome : "Pequenos Roedores"},  
 616: {id : '617', nome : "Perdizes"},  
 617: {id : '618', nome : "Perfis de Aço"},  
 618: {id : '619', nome : "Perfis de Outros Metais não Ferrosos"},  
 619: {id : '620', nome : "Cosméticos e artigos de perfumaria"},  
 620: {id : '621', nome : "Perfumes"},  
 621: {id : '622', nome : "Perfuradoras (ferramenta)"},  
 622: {id : '623', nome : "Perfuradores Metálicos (Embalados)"},  
 623: {id : '624', nome : "Peruas (ave)"},  
 624: {id : '625', nome : "Perus"},  
 625: {id : '626', nome : "Pescado industrializado"},  
 626: {id : '627', nome : "Pêssego (a Granel)"},  
 627: {id : '628', nome : "Pêssego (Doces e Conservas)"},  
 628: {id : '629', nome : "Pêssego (Embalado)"},  
 629: {id : '630', nome : "Petróleo Bruto"},  
 630: {id : '631', nome : "Petroquimicos Básicos (Granéis Líquidos)"},  
 631: {id : '632', nome : "Petroquimicos Básicos (Granéis Sólidos)"},  
 632: {id : '633', nome : "Pias (metal)"},  
 633: {id : '634', nome : "Picaretas"},  
 634: {id : '635', nome : "Pimenta do Reino (a Granel)"},  
 635: {id : '636', nome : "Pimenta do Reino (Embalada)"},  
 636: {id : '637', nome : "Pinças e artefatos"},  
 637: {id : '638', nome : "Pinças e ferramentas semelhantes"},  
 638: {id : '639', nome : "Pintos"},  
 639: {id : '640', nome : "Placas de Aço"},  
 640: {id : '641', nome : "Placas de Aluminio"},  
 641: {id : '642', nome : "Placas de Ferro"},  
 642: {id : '643', nome : "Placas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 643: {id : '644', nome : "Placas Indicadoras Metálicas (Peças e Volumes)"},  
 644: {id : '645', nome : "Placas Metálicas para Veículos (Embalados)"},  
 645: {id : '646', nome : "Placas Sinalizadoras e Semelhantes Metálicas (Peças e Volumes)"},  
 646: {id : '647', nome : "Plainas"},  
 647: {id : '648', nome : "Mudas de Plantas ou Árvores"},  
 648: {id : '649', nome : "Plaquetas de Aluminio"},  
 649: {id : '650', nome : "Plaquetas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 650: {id : '651', nome : "Plásticos (outros)"},  
 651: {id : '652', nome : "Pneus e Câmaras de Ar Novos"},  
 652: {id : '653', nome : "Pneus Recondicionados"},  
 653: {id : '654', nome : "Pneus Usados"},  
 654: {id : '655', nome : "Pólvoras, Explosivos e Detonantes"},  
 655: {id : '656', nome : "Pontes e elementos de pontes, de ferro e aço"},  
 656: {id : '657', nome : "Porcas e Arruelas"},  
 657: {id : '658', nome : "Porcos e Porcas vivos"},  
 658: {id : '659', nome : "Porta-canetas e Semelhantes Metálicos (Embalados)"},  
 659: {id : '660', nome : "Portas Blindadas e Compartimentos para Casas-Fortes (Peças e Volumes)"},  
 660: {id : '661', nome : "Portas de diques"},  
 661: {id : '662', nome : "Portas de Eclusas"},  
 662: {id : '663', nome : "Portas e janelas de Madeida"},  
 663: {id : '664', nome : "Portas e Janelas de Metal"},  
 664: {id : '665', nome : "Portas e Portoes de Aço, Aluminio ou Ferro"},  
 665: {id : '666', nome : "Pós de Ferro"},  
 666: {id : '667', nome : "Pós de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 667: {id : '668', nome : "Potros"},  
 668: {id : '669', nome : "Prata"},  
 669: {id : '670', nome : "Pratos (metal)"},  
 670: {id : '671', nome : "Pré-Moldados de concreto"},  
 671: {id : '672', nome : "Presunto"},  
 672: {id : '673', nome : "Processadores de Alimentos (Eletrodoméstico)"},  
 673: {id : '674', nome : "Produtos agrícolas (outros)"},  
 674: {id : '675', nome : "Produtos da floresta (outros)"},  
 675: {id : '676', nome : "Produtos de Higiene"},  
 676: {id : '677', nome : "Produtos de Madeira"},  
 677: {id : '678', nome : "Produtos de Salsicharia"},  
 678: {id : '679', nome : "Produtos Dietéticos"},  
 679: {id : '680', nome : "Produtos farmacêuticos"},  
 680: {id : '681', nome : "Produtos para fotografia"},  
 681: {id : '682', nome : "Produtos químicos outros (Granéis Líquidos)"},  
 682: {id : '683', nome : "Produtos químicos outros (Granéis Sólidos)"},  
 683: {id : '684', nome : "Produtos Têxteis (outros)"},  
 684: {id : '685', nome : "Quadros e Artigos para decoração"},  
 685: {id : '686', nome : "Queijos"},  
 686: {id : '687', nome : "Querosene de aviação"},  
 687: {id : '688', nome : "Ração Animal"},  
 688: {id : '689', nome : "Rádios e Aparelhos de Som"},  
 689: {id : '690', nome : "Raspadeiras (ferramenta)"},  
 690: {id : '691', nome : "Reatores nucleares"},  
 691: {id : '692', nome : "Rebites de Aço, Cobre e Ferro"},  
 692: {id : '694', nome : "Recipientes de Aço, Aluminio ou Ferro"},  
 693: {id : '695', nome : "Refrigerantes (Envasados)"},  
 694: {id : '696', nome : "Refrigerantes (Granéis Líquidos)"},  
 695: {id : '697', nome : "Regadores e semelhantes, de zinco"},  
 696: {id : '698', nome : "Regadores e similares, de plastico"},  
 697: {id : '699', nome : "Reservatorios Metalicos"},  
 698: {id : '700', nome : "Resíduos de Aluminio"},  
 699: {id : '701', nome : "Resíduos de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 700: {id : '702', nome : "Resinas e Elastômeros"},  
 701: {id : '703', nome : "Rocha Fosfática"},  
 702: {id : '704', nome : "Rocha Ornamental"},  
 703: {id : '705', nome : "Rodízios de Aço, Aluminio ou Ferro"},  
 704: {id : '706', nome : "Rolhas e Tampas Metalicas"},  
 705: {id : '707', nome : "Rolhas e Tampas Plasticas"},  
 706: {id : '708', nome : "Rosbife"},  
 707: {id : '709', nome : "Roupas e confecções"},  
 708: {id : '710', nome : "Sabões, Detergentes Produtos de Limpeza"},  
 709: {id : '711', nome : "Saboneteiras de Aluminio"},  
 710: {id : '712', nome : "Sabonetes e Shampoos"},  
 711: {id : '713', nome : "Saca-bocados (ferramenta)"},  
 712: {id : '714', nome : "Sachos (ferramenta)"},  
 713: {id : '715', nome : "Sacos e Sacolas Plasticas"},  
 714: {id : '716', nome : "Sal (a Granel)"},  
 715: {id : '717', nome : "Sal (Embalado)"},  
 716: {id : '718', nome : "Salame"},  
 717: {id : '719', nome : "Saleiros (metal)"},  
 718: {id : '720', nome : "Salsicha"},  
 719: {id : '721', nome : "Sapatos"},  
 720: {id : '722', nome : "Sargentos (ferramentas)"},  
 721: {id : '723', nome : "Semeadores manuais"},  
 722: {id : '724', nome : "Sementes Para Lavoura Temporária"},  
 723: {id : '725', nome : "Sementes Para Pastagem"},  
 724: {id : '726', nome : "Serras manuais"},  
 725: {id : '727', nome : "Siderúrgicos (outros)"},  
 726: {id : '728', nome : "Sinos Metálicas (Peças e Volumes)"},  
 727: {id : '729', nome : "Soja em grão"},  
 728: {id : '730', nome : "Sorgo"},  
 729: {id : '731', nome : "Sucatas de Ferro e Aço"},  
 730: {id : '732', nome : "Sucatas Metálicas"},  
 731: {id : '733', nome : "Suco de Laranja (Envasado)"},  
 732: {id : '734', nome : "Suco de Laranja (Granéis Líquidos)"},  
 733: {id : '735', nome : "Suco de Uva (Envasado)"},  
 734: {id : '736', nome : "Suco de Uva (Granéis Líquidos)"},  
 735: {id : '737', nome : "Sucos Outros (Envasado)"},  
 736: {id : '738', nome : "Sucos Outros (Granéis Líquidos)"},  
 737: {id : '739', nome : "Suínos vivos"},  
 738: {id : '740', nome : "Suportes (ganchos) Metálicos para Vestuário (Embalados)"},  
 739: {id : '741', nome : "Suportes de Metal"},  
 740: {id : '742', nome : "Suportes de Plastico"},  
 741: {id : '743', nome : "Tambores de Metal"},  
 742: {id : '744', nome : "Tambores de Plastico"},  
 743: {id : '745', nome : "Tampas plasticas para vaso Sanitário"},  
 744: {id : '746', nome : "Tangerina (a Granel)"},  
 745: {id : '747', nome : "Tangerina (Doces e Conservas)"},  
 746: {id : '748', nome : "Tangerina (Embalada)"},  
 747: {id : '749', nome : "Tanques de Cozinha (metal)"},  
 748: {id : '750', nome : "Tanques de grande capacidade (metal)"},  
 749: {id : '751', nome : "Tântalo"},  
 750: {id : '752', nome : "Tapetes"},  
 751: {id : '753', nome : "Tapioca e seus sucedândeos"},  
 752: {id : '754', nome : "Tecidos e Seus Artefatos (Exceto Vestuário)"},  
 753: {id : '755', nome : "Tecidos têxteis"},  
 754: {id : '756', nome : "Telas de fios de cobre, niquel"},  
 755: {id : '757', nome : "Telas metálicas tecidas"},  
 756: {id : '758', nome : "Telefones e Sistemas de Comunicação"},  
 757: {id : '759', nome : "Televisores"},  
 758: {id : '760', nome : "Telhas"},  
 759: {id : '761', nome : "Tenazes (ferramenta)"},  
 760: {id : '762', nome : "Tesouras"},  
 761: {id : '763', nome : "Tesouras-serrote"},  
 762: {id : '764', nome : "Tijolos"},  
 763: {id : '765', nome : "Tintas, vernizes, esmaltes e lacas"},  
 764: {id : '766', nome : "Tiras de Aço"},  
 765: {id : '767', nome : "Tiras de Aluminio"},  
 766: {id : '768', nome : "Tiras de Outros Metais não Ferrosos"},  
 767: {id : '769', nome : "Tiras de Plástico"},  
 768: {id : '770', nome : "Titânio"},  
 769: {id : '771', nome : "Tomate (a Granel)"},  
 770: {id : '772', nome : "Tomate (Doces e Conservas)"},  
 771: {id : '773', nome : "Tomate (Embalado)"},  
 772: {id : '774', nome : "Tonéis"},  
 773: {id : '775', nome : "Torneiras e Válvulas"},  
 774: {id : '776', nome : "Tornos"},  
 775: {id : '777', nome : "Torres de telegrafia"},  
 776: {id : '778', nome : "Torres e pórticos (pilares), de ferro e aço"},  
 777: {id : '779', nome : "Trailers"},  
 778: {id : '780', nome : "Tratores e Implementos para Agricultura (Inclusive Peças)"},  
 779: {id : '781', nome : "Travessas"},  
 780: {id : '782', nome : "Trigo em grão"},  
 781: {id : '783', nome : "Trilhos de Aço"},  
 782: {id : '784', nome : "Tubos de Ferro e Aço"},  
 783: {id : '785', nome : "Tubos de Outros Metais não Ferrosos"},  
 784: {id : '786', nome : "Tubos de Plástico"},  
 785: {id : '787', nome : "Tubos e Conexões de Ferro, Aço e Aço Inoxidável"},  
 786: {id : '788', nome : "Tubos Flexiveis de Chumbo para Embalagens"},  
 787: {id : '789', nome : "Tubos Flexiveis de Metal"},  
 788: {id : '790', nome : "Tungstênio"},  
 789: {id : '791', nome : "Uniões de Aluminio"},  
 790: {id : '792', nome : "Uniões para Tubos de Ferro e Aço"},  
 791: {id : '793', nome : "Uréia"},  
 792: {id : '794', nome : "Utensílios de plastico"},  
 793: {id : '795', nome : "Utensílios p/Cozinha Movidos a Eletricidade"},  
 794: {id : '796', nome : "Uva (a Granel)"},  
 795: {id : '797', nome : "Uva (Doces e Conservas)"},  
 796: {id : '798', nome : "Uva (Embalado)"},  
 797: {id : '799', nome : "Vacas"},  
 798: {id : '800', nome : "Vagões Ferroviarios"},  
 799: {id : '801', nome : "Veículos e Material Rodante para Ferrovias"},  
 800: {id : '802', nome : "Veículos e Utilitários"},  
 801: {id : '803', nome : "Vergalhões de Aço"},  
 802: {id : '804', nome : "Vidros e Espelhos"},  
 803: {id : '805', nome : "Vidros Laminados para Veículos Automotores, Aeronaves, etc"},  
 804: {id : '806', nome : "Vidros Planos"},  
 805: {id : '807', nome : "Vinhos (Envasados)"},  
 806: {id : '808', nome : "Vinhos (Granéis Líquidos)"},  
 807: {id : '809', nome : "Vôngoles"},  
 808: {id : '810', nome : "Xarope de Glicose"},  
 809: {id : '811', nome : "Xícaras (metal)"},  
 810: {id : '812', nome : "Xicaras de plastico"},  
 811: {id : '813', nome : "Zinco"},  
 812: {id : '814', nome : "Zircônio"},  
 813: {id : '815', nome : "Equipamento Bélico (pequeno porte, armas e munição)"},  
 814: {id : '816', nome : "Balcoes e Balaustradas (madeira)"},  
 815: {id : '817', nome : "Figo (a Granel)"},  
 816: {id : '818', nome : "Figo (Doces e Conservas)"},  
 817: {id : '819', nome : "Figo (Embalada)"},  
 818: {id : '820', nome : "Goiaba (a Granel)"},  
 819: {id : '821', nome : "Goiaba (Doces e Conservas)"},  
 820: {id : '822', nome : "Goiaba (Embalada)"},  
 821: {id : '823', nome : "Outras frutas (Embalada)"},  
 822: {id : '824', nome : "Baldes (plástico)"},  
 823: {id : '825', nome : "Bidês (louça, porcelana, cerâmica)"},  
 824: {id : '826', nome : "Pias (louça, porcelana, cerâmica)"},  
 825: {id : '827', nome : "Pratos (louça, porcelana, cerâmica)"},  
 826: {id : '828', nome : "Artigos banheiro - saboneteiras, porta escova (louça, porcelana, cerâmica )"},  
 827: {id : '829', nome : "Saleiros (louça, porcelana, cerâmica)"},  
 828: {id : '830', nome : "Artigos banheiro - saboneteiras, porta escova (plástico)"},  
 829: {id : '831', nome : "Saleiros (plástico)"},  
 830: {id : '832', nome : "Tanques de Cozinha (plástico)"},  
 831: {id : '833', nome : "Tanques de Cozinha (louça, porcelana, cerâmica )"},  
 832: {id : '834', nome : "Tanques de grande capacidade (plástico, fibra de vidro, outros)"},  
 833: {id : '835', nome : "Xicaras (louça, porcelana, cerâmica )"},  
 834: {id : '836', nome : "Cajú (embalado)"},  
 835: {id : '837', nome : "Bisnagas flexiveis para embalagem (plástico)"},  
 836: {id : '838', nome : "Louças"},  
 837: {id : '839', nome : "Aparelhos Celulares"},  
 838: {id : '840', nome : "Cristaleria"},  
 839: {id : '841', nome : "Frutas (a Granel)"},  
 840: {id : '842', nome : "Remédios"},  
 841: {id : '844', nome : "Objetos pessoais"},  
 842: {id : '845', nome : "Malas"},  
 843: {id : '846', nome : "Mobiliários (mudanças)"},  
 844: {id : '848', nome : "Cadáver (Corpo humano)"},  
 845: {id : '849', nome : "Restos incinerados"},  
 846: {id : '850', nome : "Cinzas do corpo humano"},  
 847: {id : '851', nome : "Platina"},  
 848: {id : '853', nome : "Bônus"},  
 849: {id : '855', nome : "Obras de arte"},  
 850: {id : '856', nome : "Esculturas"},  
 851: {id : '857', nome : "Jóias"},  
 852: {id : '858', nome : "Pedras Preciosas ou Semi-preciosas"},  
 853: {id : '859', nome : "Jaulas"},  
 854: {id : '860', nome : "Gaiolas"},  
 855: {id : '861', nome : "Petardos"},  
 856: {id : '863', nome : "Extintores"},  
 857: {id : '864', nome : "Aerossóis"},  
 858: {id : '865', nome : "Inaladores"},  
 859: {id : '866', nome : "Isqueiros"},  
 860: {id : '867', nome : "Garrafas de Oxigênio"},  
 861: {id : '868', nome : "Garrafas de Ar Comprimido"},  
 862: {id : '869', nome : "Líquidos Inflamáveis"},  
 863: {id : '870', nome : "Colas"},  
 864: {id : '871', nome : "Acetona"},  
 865: {id : '873', nome : "Produtos Petrolíferos"},  
 866: {id : '874', nome : "Sólidos Inflamáveis"},  
 867: {id : '875', nome : "Magnésio"},  
 868: {id : '876', nome : "Sódio"},  
 869: {id : '877', nome : "Peróxidos Orgânicos"},  
 870: {id : '878', nome : "Substâncias Oxidantes"},  
 871: {id : '879', nome : "Nitrato Amoníaco"},  
 872: {id : '880', nome : "Geradores de Oxigênio"},  
 873: {id : '881', nome : "Tóxicos e Substâncias infecciosas"},  
 874: {id : '884', nome : "Desinfetantes"},  
 875: {id : '885', nome : "Sangue Contaminado"},  
 876: {id : '886', nome : "Amostras Infectadas"},  
 877: {id : '887', nome : "Material Radioativo"},  
 878: {id : '888', nome : "Detectores de fumaça"},  
 879: {id : '889', nome : "Plutônio"},  
 880: {id : '890', nome : "Urânio"},  
 881: {id : '891', nome : "Materiais Corrosivos"},  
 882: {id : '892', nome : "Ácidos"},  
 883: {id : '893', nome : "Mercúrio"},  
 884: {id : '894', nome : "Amoníaco"},  
 885: {id : '895', nome : "Produtos de Limpeza"},  
 886: {id : '896', nome : "Baterias e pilhas"},  
 887: {id : '897', nome : "Mercadorias Perigosas"},  
 888: {id : '898', nome : "Material Magnetizado"},  
 889: {id : '899', nome : "Equipamentos de Salvamento"},  
 890: {id : '900', nome : "Combustíveis e seus derivados"},  
 891: {id : '902', nome : "Cristais"},  
 892: {id : '903', nome : "Documentos, Títulos e Apólices"},  
 893: {id : '904', nome : "Peróxido de Hidrogênio"},  
 894: {id : '905', nome : "Água de Côco"},  
 895: {id : '906', nome : "Água Oxigenada"},  
 896: {id : '907', nome : "Alimentos Industrializados"},  
 897: {id : '908', nome : "Argamassa"},  
 898: {id : '910', nome : "Amianto"},  
 899: {id : '911', nome : "Antenas e Equipamentos de Telecomunicações"},  
 900: {id : '912', nome : "Massa Asfáltica"},  
 901: {id : '913', nome : "Elevadores"},  
 902: {id : '914', nome : "Aveia"},  
 903: {id : '915', nome : "Azeite"},  
 904: {id : '916', nome : "Azeitonas"},  
 905: {id : '917', nome : "Balanças"},  
 906: {id : '918', nome : "Bauxita"},  
 907: {id : '919', nome : "Paletes de Madeira"},  
 908: {id : '920', nome : "Bambu"},  
 909: {id : '921', nome : "Estofamentos / Poltronas / bancos para veículos"},  
 910: {id : '922', nome : "Estofados / Sofás / Poltronas / Almofadas"},  
 911: {id : '923', nome : "Barracas e Equipamentos para Festas"},  
 912: {id : '924', nome : "Botija ou Botijão (Vazios)"},  
 913: {id : '925', nome : "Bicarbonato de Sódio"},  
 914: {id : '926', nome : "Betoneiras"},  
 915: {id : '927', nome : "Beterrabas"},  
 916: {id : '928', nome : "Cenouras"},  
 917: {id : '929', nome : "Legumes / Verduras"},  
 918: {id : '930', nome : "Camas ou Beliches"},  
 919: {id : '931', nome : "Extrato de Tomate / Polpa de Tomate"},  
 920: {id : '932', nome : "Ervilhas"},  
 921: {id : '933', nome : "Adubo Orgânico / Esterco"},  
 922: {id : '934', nome : "Polpa de Frutas"},  
 923: {id : '935', nome : "Polietileno"},  
 924: {id : '936', nome : "Pó de Serra / Serragem"},  
 925: {id : '937', nome : "Cevada"},  
 926: {id : '938', nome : "Soda Cáustica ou Hidróxido de Sódio"},  
 927: {id : '939', nome : "Sulfato de Amônia"},  
 928: {id : '940', nome : "Sulfato de Alumínio"},  
 929: {id : '941', nome : "Sulfato de Cálcio"},  
 930: {id : '942', nome : "Tolueno ou metil benzeno"},  
 931: {id : '943', nome : "Solvente, Dissolvente ou Dispersante"},  
 932: {id : '944', nome : "Cereais"},  
 933: {id : '945', nome : "Sebo bovino"},  
 934: {id : '946', nome : "Sebo Ácido"},  
 935: {id : '948', nome : "Cesta Básica"},  
 936: {id : '949', nome : "Caixa Eletrônico"},  
 937: {id : '950', nome : "Caixão / Urna Funerária"},  
 938: {id : '951', nome : "Caixa de Papelão"},  
 939: {id : '952', nome : "Material Escolar / Cadernos"},  
 940: {id : '953', nome : "Cabos para Vassouras / Rodos"},  
 941: {id : '954', nome : "Cabides"},  
 942: {id : '955', nome : "Móveis / Cadeiras / Mesas"},  
 943: {id : '956', nome : "Mudanças"},  
 944: {id : '957', nome : "Cadeiras de Plástico"},  
 945: {id : '958', nome : "Caixas Térmicas"},  
 946: {id : '959', nome : "Embalagens ou Caixas de Isopor"},  
 947: {id : '960', nome : "Câmara Frigorífica"},  
 948: {id : '961', nome : "Capim, Grama ou Sementes de Capim"},  
 949: {id : '962', nome : "Carbonatos - Sódio / Cálcio / Cobre / Zinco / Guanidina / Cobalto/ Magnésio / Bário / Silício"},  
 950: {id : '963', nome : "Caixotes de Madeira"},  
 951: {id : '964', nome : "Churrasqueiras"},  
 952: {id : '965', nome : "Silagem ou Silos"},  
 953: {id : '966', nome : "Cianeto de sódio"},  
 954: {id : '967', nome : "Chaminés"},  
 955: {id : '968', nome : "Cilindros"},  
 956: {id : '969', nome : "Climatizadores / Aquecedores / Condicionadores de Ar"},  
 957: {id : '970', nome : "Colchões"},  
 958: {id : '971', nome : "Cobertores e Tecidos Cama, Mesa e Banho / Travesseiros"},  
 959: {id : '972', nome : "Congelados ou Frios"},  
 960: {id : '973', nome : "Glicerina"},  
 961: {id : '974', nome : "Cogumelo / Champagnion"},  
 962: {id : '975', nome : "Caixas para Entulho"},  
 963: {id : '976', nome : "Conservas"},  
 964: {id : '977', nome : "Cordas e Cabos"},  
 965: {id : '978', nome : "Dióxido de carbono / Anidrido Carbônico / Gás carbônico"},  
 966: {id : '979', nome : "Foscálcio"},  
 967: {id : '980', nome : "Formol"},  
 968: {id : '981', nome : "Lubrificantes, Graxas"},  
 969: {id : '982', nome : "Farelo ou Gérmem de Milho"},  
 970: {id : '983', nome : "Hipocloreto ou Cloreto de Sódio"},  
 971: {id : '984', nome : "Equipamentos de Informática"},  
 972: {id : '985', nome : "Lã de Vidro"},  
 973: {id : '986', nome : "Nitrato de Alumínio"},  
 974: {id : '987', nome : "Lisina"},  
 975: {id : '988', nome : "Betume ou Piche"},  
 976: {id : '989', nome : "Piscinas"},  
 977: {id : '990', nome : "Policloreto de Alumínio"},  
 978: {id : '991', nome : "Polvilho"},  
 979: {id : '992', nome : "Pregos e Grampos"},  
 980: {id : '993', nome : "Material / produtos para Reciclagem"},  
 981: {id : '994', nome : "Proteína de Soja"},  
 982: {id : '995', nome : "Sacaria - Sacos"},  
 983: {id : '996', nome : "Vasilhames"},  
 984: {id : '997', nome : "Vinagre"},  
 985: {id : '998', nome : "Artesanatos"},  
 986: {id : '999', nome : "Biodiesel"},  
 987: {id : '1000', nome : "Bijouterias e Acessórios"},  
 988: {id : '1001', nome : "Biomassas"},  
 989: {id : '1002', nome : "Bobinas de Plástico"},  
 990: {id : '1003', nome : "Borras Algodão / Soja"},  
 991: {id : '1004', nome : "Emulsão"},  
 992: {id : '1005', nome : "Etileno Glicol Mono Butil Éter"},  
 993: {id : '1006', nome : "Carrinhos para Compras"},  
 994: {id : '1007', nome : "Descartáveis"},  
 995: {id : '1008', nome : "Dormentes"},  
 996: {id : '1009', nome : "Embutidos / Enlatados"},  
 997: {id : '1010', nome : "Aparelhos para Ginástica"},  
 998: {id : '1011', nome : "Farelo de Algodão"},  
 999: {id : '1012', nome : "Farelo de Arroz"},  
1000: {id : '1013', nome : "Farelo ou Farinha de Osso"},  
1001: {id : '1014', nome : "Farelo ou Farinha de Carne"},  
1002: {id : '1015', nome : "Feno"},  
1003: {id : '1016', nome : "Fibra de Côco"},  
1004: {id : '1017', nome : "Gôndolas / Prateleiras"},  
1005: {id : '1018', nome : "Dolomita"},  
1006: {id : '1019', nome : "Metanol / Hidroximetano / Álcool metílico / Carbinol"},  
1007: {id : '1020', nome : "Ferro Gusa"},  
1008: {id : '1021', nome : "Nitrogênio ou Azoto"},  
1009: {id : '1022', nome : "Água (Granel Líquido)"},  
1010:// {id : '3000', nome : "VAZIO"},  
1011: {id : '4000', nome : "NÃO SOUBE INFORMAR"},  
1012: {id : '5000', nome : "OUTROS"}  
1013:];

## .\mobile\www\controllers\confirmar\textunderscore veiculo.js

1:controllers.confirmar\_veiculo = {  
 2: config : function() {  
 3:  
 4: $("#img\_confirmar\_veiculo").attr('src', app.imagemPath);  
 5: $("#conf\_veic\_nao").click(function() {  
 6: // clear  
 7: app.iniciaRegistro();  
 8: app.trocaPagina('views/selecionar\_tipo.html', controllers.selecionar\_tipo);  
 9: });  
 10: $("#conf\_veic\_sim").click(  
 11: function() {  
 12: var imgFileName = app.imagemPath.split("/").reverse()[0];  
 13: var catVeiculo;  
 14: if (imgFileName.match(/(m)[\_0-9]\*\.png/g) != null) {  
 15: catVeiculo = 'Moto';  
 16: } else if (imgFileName.match(/(p)[\_0-9]\*\.png/g) != null) {  
 17: catVeiculo = 'Leve';  
 18: } else if (imgFileName.match(/(o)[\_0-9]\*\.png/g) != null) {  
 19: catVeiculo = 'Onibus';  
 20: } else // if (imgFileName.match(/(c|r|s|se)[\_0-9]\*\.png/g) != null)  
 21: {  
 22: catVeiculo = 'Pesado';  
 23: var tipoCaminhao;  
 24: if (imgFileName.match(/(c)[\_0-9]\*\.png/g) != null) {  
 25: tipoCaminhao = 'Leve';  
 26: }  
 27: if (imgFileName.match(/(s)[\_0-9]\*\.png/g) != null) {  
 28: tipoCaminhao = 'Semirreboque';  
 29: }  
 30: if (imgFileName.match(/(se)[\_0-9]\*\.png/g) != null) {  
 31: tipoCaminhao = 'Semirreboque Especial';  
 32: }  
 33: if (imgFileName.match(/(r)[\_0-9]\*\.png/g) != null) {  
 34: tipoCaminhao = 'Reboque';  
 35: }  
 36: app.setAtributo('tipoCaminhao', tipoCaminhao);  
 37: }  
 38: app.setAtributo('categoria', catVeiculo);  
 39: app.trocaPagina("views/" + registro.classeVeiculo + '/identificacao\_visual.html',  
 40: controllers['identificacao\_visual\_' + registro.classeVeiculo]);  
 41: });  
 42: }  
 43:}

## .\mobile\www\controllers\menu.js

1:/\* global util, datas, app, ipadID \*/  
 2:  
 3:controllers.menu = {  
 4: config : function() {  
 5: $('#display\_posto').html(app.posto);  
 6: $('#display\_idIpad').html(ipadID.id);  
 7: $('#display\_sentido').html(app.sentido);  
 8: $('#display\_data\_hora').html(util.formatDateTimeToDisplay());  
 9: var updater\_dataTimeToDisplay = setInterval(function(){  
 10: $('#display\_data\_hora').html(util.formatDateTimeToDisplay());  
 11: },10000);  
 12:   
 13: if (app.isTreinamento){  
 14: $('#banner\_supra').html("<br>O sistema está em modo de treinamento").addClass('avisoTreinamento');  
 15: }else{  
 16: $('#banner\_supra').html("").removeClass('avisoTreinamento');  
 17: }  
 18:   
 19: $('#menu\_nova\_pesquisa').click(function() {  
 20: clearInterval(updater\_dataTimeToDisplay);  
 21:   
 22: if (datas.verificaData() && app.isTreinamento) {  
 23: alert("Login de treinamento não podem ser \nutilizado em dias de produção de pesquisas reais.\nA aplicação será reiniciada",  
 24: "Controle de acesso", app.restart, 'error');  
 25: } else if (!datas.verificaData() && !app.isTreinamento){  
 26: alert("Logins de produção não podem ser \nutilizados fora das datas de produção de pesquisas reais.\nA aplicação será reiniciada",  
 27: "Controle de acesso", app.restart, 'error');  
 28: } else {  
 29: // clear registro  
 30: app.iniciaRegistro();  
 31: app.trocaPagina("views/selecionar\_tipo.html", controllers.selecionar\_tipo);  
 32: }  
 33: });  
 34:  
 35: $('#menu\_sumario').click(function() {  
 36: $.mobile.loading("show");  
 37: app.buscaDuracoesRegistros(function(){  
 38: app.buscaUltimaPesquisa(function(){  
 39: app.buscaRegistrosCancelados(function(){  
 40: $.mobile.loading("hide");  
 41: clearInterval(updater\_dataTimeToDisplay);  
 42: app.trocaPagina("views/sumario.html", controllers.sumario);  
 43: });  
 44: });  
 45: });  
 46:   
 47: });  
 48:   
 49: $("#menu\_trocar\_sentido").click(function(){  
 50: var novo\_sentido;  
 51: if (app.sentido == 'AB'){  
 52: novo\_sentido = 'BA'  
 53: }else{  
 54: novo\_sentido = 'AB'  
 55: }  
 56: app.validaOperacoes(function(){  
 57: app.sentido = novo\_sentido;  
 58: app.trocaPagina("views/menu.html", controllers.menu);  
 59: },   
 60: "Insira a senha para alterar o sentido para <span style='font-weight: bolder;'>" + novo\_sentido + "</span>.",  
 61: "Alterar Sentido",  
 62: "Senha incorreta.\nDeseja tentar novamente?",  
 63: "Senha Incorreta",   
 64: "Alterar Sentido",  
 65: "Voltar"  
 66: );  
 67: });  
 68:  
 69: $('#testaBugOld').click(function(){  
 70: for (i = 0; i < 51; i++){  
 71: app.trocaPagina("views/selecionar\_tipo.html", controllers.selecionar\_tipo, 'old');  
 72: app.trocaPagina("views/menu.html", controllers.menu, 'old');  
 73: }  
 74: });  
 75:  
 76: $('#testaBugNew').click(function(){  
 77: for (i = 0; i < 51; i++){  
 78: app.trocaPagina("views/selecionar\_tipo.html", controllers.selecionar\_tipo);  
 79: app.trocaPagina("views/menu.html", controllers.menu);  
 80: }  
 81: });  
 82:  
 83: /\*$("#duplica\_log").click(function() {  
 84: if (app.filePaths){  
 85: app.copyFile(app.logFileName,  
 86: cordova.file.dataDirectory,  
 87: app.filePaths.externalFolder,  
 88: function(newName){  
 89: alert('Arquivo de log ' + newName + ' exportado com sucesso.');  
 90: });  
 91: } else {  
 92: alert('Operação não realizada, o sistema de arquivos não foi definido');  
 93: }  
 94:   
 95: });\*/  
 96:  
 97: /\*$("#duplica\_db").click(function() {  
 98: if(app.filePaths){  
 99: app.database.close(function() {  
 100: app.copyFile(app.dbName,   
 101: app.filePaths.dbFolder,  
 102: app.filePaths.externalFolder,  
 103: function(newName){  
 104: alert('Banco de dados ' + newName + ' exportado com sucesso.');  
 105: app.openDB();  
 106: });  
 107: }, function(err) {  
 108: app.logger.log(JSON.stringify(err));  
 109: });  
 110: } else {  
 111: alert('Operação não realizada, o sistema de arquivos não foi definido');  
 112: }  
 113: });  
 114:   
 115: $("#menu\_logout").click(function () {  
 116: app.logout();  
 117: });\*/  
 118:  
 119: },  
 120:};

## .\mobile\www\controllers\pergunta\textunderscore extra.js

1:controllers.pergunta\_extra = {  
 2: config : function() {  
 3: var me = controllers.pergunta\_extra;  
 4: me.inicializaElementos();  
 5: me.progressoTela();  
 6: me.buttons();  
 7: },  
 8:  
 9: buttons : function() {  
 10: $("#pergunta\_extra\_avancar").click(function() {  
 11: var ok = controllers.pergunta\_extra.validar\_componentes();  
 12: if (ok) {  
 13: app.finalizaRegistro(function() {  
 14: app.trocaPagina('views/menu.html', controllers.menu);  
 15: });  
 16: }  
 17: })  
 18: },  
 19:  
 20: // Inicializa os elementos da tela  
 21: inicializaElementos : function() {  
 22:  
 23: util.inicializaSelectFrequencia('simples');  
 24:  
 25: var lista\_opcoes = [ 'Sim, trocaria minha rota atual por esta', 'Não, prefiro usar a rota atual',  
 26: 'Minha rota não seria afetada', 'Não sei informar' ];  
 27:   
 28: if (logins.user\_logado != null && (logins.user\_logado.usr == '019' || logins.user\_logado.usr == '149'  
 29: || logins.user\_logado.usr == '019t' || logins.user\_logado.usr == '149t')) {  
 30: $("#grupo\_pergunta\_extra\_p1").show();  
 31: util.inicializaSelect("select\_pergunta\_extra\_p1", lista\_opcoes);  
 32: if (logins.user\_logado != null && (logins.user\_logado.usr == '149' || logins.user\_logado.usr == '149t')) {  
 33: util.inicializaSelect("select\_pergunta\_extra\_p2", lista\_opcoes);  
 34: }  
 35: }  
 36: else if (logins.user\_logado != null && (logins.user\_logado.usr == '040' || logins.user\_logado.usr == '040t')) {  
 37: $("#grupo\_pergunta\_extra\_p3").show();  
 38: util.inicializaSelect("select\_pergunta\_extra\_p3", lista\_opcoes);  
 39: }  
 40: },  
 41:  
 42: // Controla o show e hide dos elementos da tela  
 43: progressoTela : function() {  
 44:  
 45: if (logins.user\_logado != null && (logins.user\_logado.usr == '019' || logins.user\_logado.usr == '019t')) {  
 46: util.progressoSelect("idPerguntaExtra", "select\_pergunta\_extra\_p1", "grupo\_pergunta\_extra\_avancar");  
 47: }  
 48: else if (logins.user\_logado != null && (logins.user\_logado.usr == '149' || logins.user\_logado.usr == '149t')) {  
 49: util.progressoSelect("idPerguntaExtra", "select\_pergunta\_extra\_p1", "grupo\_pergunta\_extra\_p2");  
 50: util.progressoSelect("idPerguntaExtra2", "select\_pergunta\_extra\_p2", "grupo\_pergunta\_extra\_avancar");  
 51: }  
 52: else if (logins.user\_logado != null && (logins.user\_logado.usr == '040' || logins.user\_logado.usr == '040t')) {  
 53: util.progressoSelect("idPerguntaExtra", "select\_pergunta\_extra\_p3", "grupo\_pergunta\_extra\_avancar");  
 54: }  
 55: else {  
 56: // Não deveria entrar aqui. Mostra o botão avançar para finalizar a pesquisa e não ficar travado.  
 57: $("#grupo\_pergunta\_extra\_avancar").show();  
 58: }  
 59: },  
 60:  
 61: // Controla as validações dos componentes de tela após clicar em AVANÇAR  
 62: validar\_componentes : function(id\_avancar) {  
 63: if (logins.user\_logado != null && (logins.user\_logado.usr == '019' || logins.user\_logado.usr == '019t')) {  
 64: return util.validaSelect("select\_pergunta\_extra\_p1", "Pergunta Extra");  
 65: }  
 66: else if (logins.user\_logado != null && (logins.user\_logado.usr == '149' || logins.user\_logado.usr == '149t')) {  
 67: var ok = util.validaSelect("select\_pergunta\_extra\_p1", "Pergunta Extra 1");  
 68: ok = ok && util.validaSelect("select\_pergunta\_extra\_p2", "Pergunta Extra 2");  
 69: return ok;  
 70: }  
 71: else if (logins.user\_logado != null && (logins.user\_logado.usr == '040' || logins.user\_logado.usr == '040t')) {  
 72: return util.validaSelect("select\_pergunta\_extra\_p3", "Pergunta Extra");  
 73: }  
 74: return false;  
 75: }  
 76:  
 77:};

## .\mobile\www\controllers\selecionar\textunderscore tipo.js

1:controllers.selecionar\_tipo = {  
 2: config : function() {  
 3:  
 4: $(".switch\_classe\_veiculo").click(function() {  
 5: $('#sel\_tipo\_veiculo\_1').toggle();  
 6: $('#sel\_tipo\_veiculo\_2').toggle();  
 7: });  
 8:  
 9: $(".img\_sel\_veiculo").on("click", controllers.selecionar\_tipo.image\_click);  
 10:  
 11: },  
 12:   
 13: image\_click : function(){  
 14: app.imagemPath = $(this).attr('src');  
 15: app.setAtributo('tipo', $(this).attr('id'));  
 16: var limitePessoas = $(this).attr('limitePessoas');  
 17: if ((!util.isEmpty(limitePessoas)) && (!isNaN(limitePessoas))) {  
 18: app.setAtributo('limitePessoas', parseInt(limitePessoas));  
 19: } else {  
 20: app.setAtributo('limitePessoas', 999); // máximo de pessoas indefinido  
 21: }  
 22:  
 23: if ($(this).hasClass('v-simples')) {  
 24: app.setAtributo('classeVeiculo', 'simples');  
 25: } else if ($(this).hasClass('v-onibus')) {  
 26: app.setAtributo('classeVeiculo', 'onibus');  
 27: } else if ($(this).hasClass('v-carga')) {  
 28: app.setAtributo('classeVeiculo', 'carga');  
 29: } else if ($(this).hasClass('v-semireboque')) {  
 30: app.setAtributo('classeVeiculo', 'carga');  
 31: } else if ($(this).hasClass('v-combinado')) {  
 32: app.setAtributo('classeVeiculo', 'carga');  
 33: } else if ($(this).hasClass('v-reboque')) {  
 34: app.setAtributo('classeVeiculo', 'carga');  
 35: }  
 36: $(".img\_sel\_veiculo").off("click",controllers.selecionar\_tipo.image\_click);  
 37: app.trocaPagina("views/confirmar\_veiculo.html", controllers.confirmar\_veiculo);  
 38: },  
 39:  
 40:}

## .\mobile\www\controllers\sumario.js

1:controllers.sumario = {  
 2: config : function() {  
 3:  
 4: var dataSet = [];  
 5: for (i = 0; i < app.sumario\_lista.length; i++) {  
 6: var elem = app.sumario\_lista[i];  
 7: dataSet.push([util.formatDateSumario(elem.diaPesq),elem.qtdDia,util.secToFrase(elem.somaDia),  
 8: util.secToFrase(Math.round(elem.mediaDia \* 100) / 100),  
 9: util.secToFrase(elem.maxTempoDia),util.secToFrase(elem.minTempoDia)]);  
 10: }  
 11:  
 12: $('#sumario\_tabela').DataTable({  
 13: "data": dataSet,  
 14: "paging": false,  
 15: "info": false,  
 16: "ordering" : false,  
 17: "searching" : false,  
 18: "columns": [  
 19: {   
 20: title: "Dia",  
 21: className : "cell\_dia"},  
 22: {   
 23: title: "Quantidade",  
 24: className : "cell\_quantidade"},  
 25: {   
 26: title: "Tempo total",  
 27: className : "cell\_tempo\_total"},  
 28: {   
 29: title: "Tempo médio",  
 30: className : "cell\_tempo\_medio"},  
 31: {   
 32: title: "Tempo máximo",  
 33: className : "cell\_tempo\_maximo"},  
 34: {   
 35: title: "Tempo mínimo",  
 36: className : "cell\_tempo\_minimo"}  
 37: ]  
 38: });  
 39:   
 40: if(app.ultima\_pesquisa.length == 2) {  
 41: $('#sumario\_ult\_pesq').text(  
 42: "A última pesquisa válida foi realizada em " + util.formatDateAndTimeSumario(app.ultima\_pesquisa[0]) +  
 43: " e durou " + util.secToFrase(app.ultima\_pesquisa[1],true));  
 44: }  
 45:  
 46: var dataCancelados = [];  
 47: for (i = 0; i < app.sumario\_lista\_cancelados.length; i++) {  
 48: var elem = app.sumario\_lista\_cancelados[i];  
 49: dataCancelados.push([util.formatDateSumario(elem.diaPesq),elem.qtdDia,util.secToFrase(elem.somaDia),  
 50: util.secToFrase(Math.round(elem.mediaDia \* 100) / 100)]);  
 51: }  
 52:  
 53: $('#sumario\_tab\_cancelados').DataTable({  
 54: "data": dataCancelados,  
 55: "paging": false,  
 56: "info": false,  
 57: "ordering" : false,  
 58: "searching" : false,  
 59: "columns": [  
 60: {  
 61: title: "Dia",  
 62: className : "cell\_dia"},  
 63: {  
 64: title: "Quantidade pesquisas canceladas",  
 65: className : "cell\_quantidade"},  
 66: {  
 67: title: "Tempo total",  
 68: className : "cell\_tempo\_total"},  
 69: {  
 70: title: "Tempo médio",  
 71: className : "cell\_tempo\_medio"}  
 72: ]  
 73: });  
 74:   
 75: myDb.sanitize();  
 76:  
 77: $('#sumario\_retornar\_menu').click(function() {  
 78: app.trocaPagina('views/menu.html', controllers.menu);  
 79: });  
 80: },  
 81:};

## .\mobile\www\controllers\carga\caracterizacao\textunderscore carga.js

1:/\* global util, registro \*/  
 2:  
 3:controllers.caracterizacao\_carga = {  
 4: config : function() {  
 5: var me = controllers.caracterizacao\_carga;  
 6: me.inicializaElementos();  
 7: me.progressoTela();  
 8: me.buttons();  
 9: },  
 10:  
 11: // TODO Acertar essa constante após alterar para o id correto  
 12: idPropriedadePropriaParticularAutônomo : 2,  
 13:  
 14: buttons : function() {  
 15: $("#caracterizacao\_carga\_avancar").click(function() {  
 16: var ok = controllers.caracterizacao\_carga.validar\_componentes();  
 17: if (ok) {  
 18: app.trocaPagina('views/carga/caracterizacao\_viagem\_carga.html', controllers.caracterizacao\_viagem\_carga);  
 19: }  
 20: })  
 21: },  
 22:  
 23: // Inicializa os elementos da tela  
 24: inicializaElementos : function() {  
 25:  
 26:// var lista\_anos = [];  
 27:// for (var ano = 2017; ano > 1899; ano--) {  
 28:// lista\_anos = lista\_anos.concat(ano);  
 29:// }  
 30:  
 31: var lista\_anos = util.geraListaAnos();  
 32:   
 33: util.inicializaSelectCustomValueAsIndex("ano\_carga", lista\_anos, "Selecione");  
 34:  
 35: // placa cinza  
 36: if (app.getAtributo('placaVermelha') == true) {  
 37: // placa vermelha  
 38: util.inicializaTabelaAuxiliar("propriedade\_carga", "Selecione", lista\_propriedade\_veiculo, "carga\_vermelha");  
 39: } else {  
 40: // placa cinza  
 41: util.inicializaTabelaAuxiliar("propriedade\_carga", "Selecione", lista\_propriedade\_veiculo, "carga\_cinza");  
 42: }  
 43:  
 44: var lista\_agregado = [ 'Transportadora', 'Empresa/Frota própria', 'Não' ];  
 45: util.inicializaSelect("agregado\_carga", lista\_agregado);  
 46: },  
 47:  
 48: // Controla o show e hide dos elementos da tela  
 49: progressoTela : function() {  
 50: var me = controllers.caracterizacao\_carga;  
 51:  
 52: util.progressoSelect("anoDeFabricacao", "ano\_carga", "grupo\_propriedade\_carga");  
 53:  
 54: $('#propriedade\_carga').change(function() {  
 55: var selecionado = Number($(this).val());  
 56: if (selecionado != -1) {  
 57: app.setAtributo('idPropriedadesDoVeiculo', $(this).val());  
 58:  
 59: if (selecionado == me.idPropriedadePropriaParticularAutônomo) {  
 60: $("#grupo\_agregado\_carga").show();  
 61: $("#grupo\_caracterizacao\_carga\_avancar").hide();  
 62: } else {  
 63: $("#grupo\_agregado\_carga").hide();  
 64: $("#grupo\_caracterizacao\_carga\_avancar").show();  
 65:  
 66: app.setAtributo('idAgregado', null);  
 67: util.progressoRestartSelect("agregado\_carga", "Selecione");  
 68: }  
 69: } else {  
 70: $("#grupo\_agregado\_carga").hide();  
 71: $("#grupo\_caracterizacao\_carga\_avancar").hide();  
 72: app.setAtributo('idPropriedadesDoVeiculo', null);  
 73: app.setAtributo('idAgregado', null);  
 74: util.progressoRestartSelect("agregado\_carga", "Selecione");  
 75: }  
 76: });  
 77:  
 78: util.progressoSelect("idAgregado", "agregado\_carga", "grupo\_caracterizacao\_carga\_avancar");  
 79: },  
 80:  
 81: // Controla as validações dos componentes de tela após clicar em AVANÇAR  
 82: validar\_componentes : function(id\_avancar) {  
 83: var me = controllers.caracterizacao\_carga;  
 84:  
 85: if (util.validaSelect("ano\_carga", "Ano de fabricação") && util.validaSelect("propriedade\_carga", "Propriedade")) {  
 86:  
 87: if (Number($("#propriedade\_carga").val()) == me.idPropriedadePropriaParticularAutônomo) {  
 88: return util.validaSelect("agregado\_carga", "Agregado");  
 89: }  
 90:  
 91: return true;  
 92: }  
 93: return false;  
 94: }  
 95:  
 96:};

## .\mobile\www\controllers\carga\caracterizacao\textunderscore viagem2\textunderscore carga.js

1:controllers.caracterizacao\_viagem2\_carga = {  
 2: config : function() {  
 3: var me = controllers.caracterizacao\_viagem2\_carga;  
 4: me.inicializaElementos();  
 5: me.progressoTela();  
 6: me.buttons();  
 7: },  
 8:  
 9: buttons : function() {  
 10: $("#caracterizacao\_viagem2\_carga\_avancar").click(function() {  
 11: var ok = controllers.caracterizacao\_viagem2\_carga.validar\_componentes();  
 12: if (ok) {  
 13: if (logins.user\_logado != null && logins.user\_logado.perguntaExtra) {  
 14: app.trocaPagina('views/pergunta\_extra.html', controllers.pergunta\_extra);  
 15: } else {  
 16: app.finalizaRegistro(function() {  
 17: app.trocaPagina('views/menu.html', controllers.menu);  
 18: });  
 19: }  
 20: }  
 21: })  
 22: },  
 23:  
 24: // Inicializa os elementos da tela  
 25: inicializaElementos : function() {  
 26:  
 27: // Verifica se possui Pergunta Extra  
 28: if (logins.user\_logado != null && logins.user\_logado.perguntaExtra) {  
 29: $("#caracterizacao\_viagem2\_carga\_avancar").html("Avançar");  
 30: }  
 31:  
 32: // EMBARQUE DA CARGA  
 33: util.inicializaSelectCustomValueAsIndex("embarque\_uf\_carga", lista\_estados, "UF");  
 34: var lista\_locais = [ 'Porto Seco', 'Porto Marítimo', 'Terminal Ferroviário', 'Terminal Hidroviário',  
 35: 'Empresa Particular', 'Atacadista/Distribuidor', 'Outros' ];  
 36: util.inicializaSelectCustom("embarque\_local\_carga", lista\_locais, "Tipo do Local");  
 37:  
 38: // DESEMBARQUE DA CARGA  
 39: util.inicializaSelectCustomValueAsIndex("desembarque\_uf\_carga", lista\_estados, "UF");  
 40: util.inicializaSelectCustom("desembarque\_local\_carga", lista\_locais, "Tipo do Local");  
 41:  
 42: // SUGESTAO DE MUNICIPIOS PARADA OBRIGATORIA  
 43: util.inicializaSelectCustomValueAsIndex("municipios\_parada\_uf1\_carga", lista\_estados, "UF");  
 44: util.inicializaSelectCustomValueAsIndex("municipios\_parada\_uf2\_carga", lista\_estados, "UF");  
 45:  
 46: controllers.caracterizacao\_viagem2\_carga.limitePesoCargaInputText();  
 47: controllers.caracterizacao\_viagem2\_carga.limitePrecoInputText('valor\_frete\_carga',7);  
 48: controllers.caracterizacao\_viagem2\_carga.limitePrecoInputText('valor\_nota\_carga',17);  
 49: },  
 50:  
 51: // Controla o show e hide dos elementos da tela  
 52: progressoTela : function() {  
 53:  
 54: util.progressoRadioSimNaoAlternado("possui\_carga", "possui\_carga", "div\_possui\_carga\_sim", "div\_possui\_carga\_nao");  
 55:  
 56: $('#produto\_carga').click(function() {  
 57: util.autocomplete("produto\_carga", lista\_produtos, "Tipo de carga", "Entre com o produto da carga");  
 58: })  
 59:  
 60: util.progressoInputText("idProduto", "produto\_carga", "grupo\_peso\_carga", true);  
 61:  
 62: util.progressoInputText("pesoDaCarga", "peso\_carga", "grupo\_valor\_frete\_carga");  
 63:  
 64: // UNIDADE DO PESO DA CARGA  
 65: app.setAtributo("unidadePesoDaCarga", $('input[name="unidade\_peso\_carga"]:checked').val());  
 66: $('input[name=unidade\_peso\_carga]').change(function() {  
 67: app.setAtributo("unidadePesoDaCarga", $('input[name="unidade\_peso\_carga"]:checked').val());  
 68: controllers.caracterizacao\_viagem2\_carga.limitePesoCargaInputText();  
 69: });  
 70:  
 71: util.progressoInputMoney("valorDoFrete", "valor\_frete\_carga", "grupo\_valor\_nota\_carga");  
 72:  
 73: util.progressoInputMoney("valorDaCarga", "valor\_nota\_carga", "grupo\_sabe\_embarque\_carga");  
 74:  
 75: util.progressoCheckboxAlternado("embarqueCargaNaoSabe", "embarque\_carga\_nao\_sei", "grupo\_sabe\_desembarque\_carga",  
 76: "div\_sabe\_embarque\_sim");  
 77: $('#embarque\_carga\_nao\_sei').click(function() {  
 78: util.progressoRestartSelect("embarque\_uf\_carga", "UF");  
 79: util.progressoRestartSelect("embarque\_local\_carga", "Tipo do Local");  
 80: $('#embarque\_mun\_carga').val("");  
 81: app.setAtributo("embarque\_uf", null);  
 82: app.setAtributo("municipioEmbarqueCarga", null);  
 83: app.setAtributo("idLocalEmbarqueCarga", null);  
 84: if ($(this).is(':checked')) {  
 85: $('#grupo\_embarque\_mun\_carga').hide();  
 86: $('#grupo\_embarque\_local\_carga').hide();  
 87: }  
 88: });  
 89: util.progressoSelect("embarque\_uf", "embarque\_uf\_carga", "grupo\_embarque\_mun\_carga");  
 90: $('#embarque\_uf\_carga').change(  
 91: function() {  
 92: var estado = $(this).val();  
 93: if (estado != '-1') {  
 94: $('#embarque\_mun\_carga').off("click").click(  
 95: function() {  
 96: util.autocomplete("embarque\_mun\_carga", lista\_municipios[estado], "Município de embarque",  
 97: "Entre com o município de embarque");  
 98: }).trigger('click');  
 99: }  
 100: });  
 101: util.progressoInputText("municipioEmbarqueCarga", "embarque\_mun\_carga", "grupo\_embarque\_local\_carga", true);  
 102: util.progressoSelect("idLocalEmbarqueCarga", "embarque\_local\_carga", "grupo\_sabe\_desembarque\_carga");  
 103:  
 104: var cargaPerigosa = app.getAtributo('possuiCargaPerigosa');  
 105: if (cargaPerigosa == true) {  
 106: util.progressoCheckboxAlternado("desembarqueCargaNaoSabe", "desembarque\_carga\_nao\_sei",  
 107: "grupo\_parada\_especial\_carga", "div\_sabe\_desembarque\_sim");  
 108:  
 109: } else {  
 110: util.progressoCheckboxAlternado("desembarqueCargaNaoSabe", "desembarque\_carga\_nao\_sei",  
 111: "grupo\_municipios\_parada\_carga", "div\_sabe\_desembarque\_sim");  
 112: ;  
 113: }  
 114: $('#desembarque\_carga\_nao\_sei').click(function() {  
 115: util.progressoRestartSelect("desembarque\_uf\_carga", "UF");  
 116: util.progressoRestartSelect("desembarque\_local\_carga", "Tipo do Local");  
 117: $('#desembarque\_mun\_carga').val("");  
 118: app.setAtributo("desembarque\_uf", null);  
 119: app.setAtributo("municipioDesembarqueCarga", null);  
 120: app.setAtributo("idLocalDesembarqueCarga", null);  
 121: if ($(this).is(':checked')) {  
 122: $('#grupo\_desembarque\_mun\_carga').hide();  
 123: $('#grupo\_desembarque\_local\_carga').hide();  
 124: }  
 125: });  
 126:  
 127: util.progressoSelect("desembarque\_uf", "desembarque\_uf\_carga", "grupo\_desembarque\_mun\_carga");  
 128: $('#desembarque\_uf\_carga').change(  
 129: function() {  
 130: var estado = $(this).val();  
 131: if (estado != '-1') {  
 132: $('#desembarque\_mun\_carga').off("click").click(  
 133: function() {  
 134: util.autocomplete("desembarque\_mun\_carga", lista\_municipios[estado],  
 135: "Município de desembarque", "Entre com o município de desembarque");  
 136: }).trigger('click');  
 137: }  
 138: });  
 139: util.progressoInputText("municipioDesembarqueCarga", "desembarque\_mun\_carga", "grupo\_desembarque\_local\_carga", true);  
 140: if (cargaPerigosa == true) {  
 141: util.progressoSelect("idLocalDesembarqueCarga", "desembarque\_local\_carga", "grupo\_parada\_especial\_carga");  
 142: } else {  
 143: util.progressoSelect("idLocalDesembarqueCarga", "desembarque\_local\_carga", "grupo\_municipios\_parada\_carga");  
 144: }  
 145:  
 146: util.progressoRadioSimNaoAlternado("carga\_anterior", "carga\_anterior\_carga", "grupo\_produto\_anterior\_carga",  
 147: "grupo\_indo\_pegar\_carga");  
 148: $('#carga\_anterior\_carga\_nao').click(function() {  
 149: $('#produto\_anterior\_carga').val("");  
 150: app.setAtributo("idCargaAnterior", null);  
 151: });  
 152: $('#produto\_anterior\_carga').click(function() {  
 153: util.autocomplete("produto\_anterior\_carga", lista\_produtos, "Tipo de carga", "Entre com o produto da carga");  
 154: })  
 155: util.progressoInputText("idCargaAnterior", "produto\_anterior\_carga", "grupo\_indo\_pegar\_carga", true);  
 156:  
 157: if (cargaPerigosa == true) {  
 158: util.progressoRadioSimNao("indoPegarCarga", "indo\_pegar\_carga", "grupo\_parada\_especial\_carga");  
 159: } else {  
 160: util.progressoRadioSimNao("indoPegarCarga", "indo\_pegar\_carga", "grupo\_municipios\_parada\_carga");  
 161: }  
 162:  
 163: util.progressoRadioSimNao("utilizaParadaEspecial", "parada\_especial\_carga", "grupo\_municipios\_parada\_carga");  
 164:  
 165: util.progressoCheckboxAlternado("municipiosParadaNaoSabe", "municipios\_parada\_nao\_sei\_carga",  
 166: "grupo\_caracterizacao\_viagem2\_carga\_avancar", "div\_municipios\_parada\_carga");  
 167:  
 168: $('#municipios\_parada\_nao\_sei\_carga').click(function() {  
 169: util.progressoRestartSelect("municipios\_parada\_uf1\_carga", "UF");  
 170: util.progressoRestartSelect("municipios\_parada\_uf2\_carga", "UF");  
 171: $('#municipios\_parada\_mun1\_carga').val("");  
 172: $('#municipios\_parada\_mun2\_carga').val("");  
 173: app.setAtributo("municipios\_parada\_uf1", null);  
 174: app.setAtributo("municipios\_parada\_uf2", null);  
 175: app.setAtributo("paradaObrigatoriaMunicipio1", null);  
 176: app.setAtributo("paradaObrigatoriaMunicipio2", null);  
 177: if ($(this).is(':checked')) {  
 178: $('#div\_municipios\_parada\_carga').hide();  
 179: }  
 180: });  
 181:  
 182: // Parada obrigatoria UF1 MUN1  
 183: util.progressoSelect("municipios\_parada\_uf1", "municipios\_parada\_uf1\_carga", "grupo\_municipios\_parada\_mun1\_carga");  
 184: $('#municipios\_parada\_uf1\_carga').change(  
 185: function() {  
 186: var estado = $(this).val();  
 187: if (estado != '-1') {  
 188: $('#municipios\_parada\_mun1\_carga').off("click").click(  
 189: function() {  
 190: util.autocomplete("municipios\_parada\_mun1\_carga", lista\_municipios[estado], "Município",  
 191: "Entre com o município");  
 192: }).trigger('click');  
 193: }  
 194: });  
 195: util.progressoInputText("paradaObrigatoriaMunicipio1", "municipios\_parada\_mun1\_carga",  
 196: "grupo\_caracterizacao\_viagem2\_carga\_avancar", true);  
 197:  
 198: // Parada obrigatoria UF2 MUN2  
 199: util.progressoSelect("municipios\_parada\_uf2", "municipios\_parada\_uf2\_carga", "grupo\_municipios\_parada\_mun2\_carga");  
 200: $('#municipios\_parada\_uf2\_carga').change(  
 201: function() {  
 202: var estado = $(this).val();  
 203: if (estado != '-1') {  
 204: $('#municipios\_parada\_mun2\_carga').off("click").click(  
 205: function() {  
 206: util.autocomplete("municipios\_parada\_mun2\_carga", lista\_municipios[estado], "Município",  
 207: "Entre com o município");  
 208: }).trigger('click');  
 209: }  
 210: });  
 211: util.progressoInputText("paradaObrigatoriaMunicipio2", "municipios\_parada\_mun2\_carga",  
 212: "grupo\_caracterizacao\_viagem2\_carga\_avancar", true);  
 213: },  
 214:  
 215: // Controla as validações dos componentes de tela após clicar em AVANÇAR  
 216: validar\_componentes : function(id\_avancar) {  
 217:  
 218: if (util.validaRadioSimNao("possui\_carga", "Possui carga")) {  
 219: var opt\_possui\_carga = $('input[name=possui\_carga]:checked').val();  
 220: if (opt\_possui\_carga == 'sim') {  
 221: // POSSUI CARGA = SIM  
 222: var ok\_possui\_carga\_sim = true;  
 223: if (!util.validaInputText("produto\_carga", "Tipo de carga")  
 224: || !util.validaInputText("peso\_carga", "Peso da carga")  
 225: || !util.validaInputText("valor\_frete\_carga", "Valor do frete")  
 226: || !util.validaInputText("valor\_nota\_carga", "Valor da carga/nota fiscal")) {  
 227: ok\_possui\_carga\_sim = false;  
 228: }  
 229:  
 230: var ok\_local\_embarque = true;  
 231: var ckb\_embarque = $('#embarque\_carga\_nao\_sei').is(':checked');  
 232: if (!ckb\_embarque) {  
 233: // LOCAL EMBARQUE  
 234: if (!util.validaSelect("embarque\_uf\_carga", "Embarque da carga - UF")  
 235: || !util.validaInputText("embarque\_mun\_carga", "Embarque da carga - Município")  
 236: || !util.validaSelect("embarque\_local\_carga", "Embarque da carga - Local")) {  
 237: ok\_local\_embarque = false;  
 238: }  
 239: }  
 240:  
 241: var ok\_local\_desmbarque = true;  
 242: var ckb\_desembarque = $('#desembarque\_carga\_nao\_sei').is(':checked');  
 243: if (!ckb\_desembarque) {  
 244: // LOCAL DESEMBARQUE  
 245: if (!util.validaSelect("desembarque\_uf\_carga", "Desembarque da carga - UF")  
 246: || !util.validaInputText("desembarque\_mun\_carga", "Desembarque da carga - Município")  
 247: || !util.validaSelect("desembarque\_local\_carga", "Desembarque da carga - Local")) {  
 248: ok\_local\_desmbarque = false;  
 249: }  
 250: }  
 251:  
 252: if (!(ok\_possui\_carga\_sim && ok\_local\_embarque && ok\_local\_desmbarque)) {  
 253: return false;  
 254: }  
 255: } else if (opt\_possui\_carga == 'nao') {  
 256: // POSSUI CARGA = NAO  
 257: if (util.validaRadioSimNao("carga\_anterior\_carga", "Carga anterior")  
 258: && util.validaRadioSimNao("indo\_pegar\_carga", "Indo pegar uma carga")) {  
 259: var opt\_carga\_anterior = $('input[name=carga\_anterior\_carga]:checked').val();  
 260: if (opt\_carga\_anterior == 'sim') {  
 261: if (!util.validaInputText("produto\_anterior\_carga", "Tipo de carga")) {  
 262: return false;  
 263: }  
 264: }  
 265: } else {  
 266: return false;  
 267: }  
 268: }  
 269: }  
 270:  
 271: var cargaPerigosa = app.getAtributo('possuiCargaPerigosa');  
 272: if (cargaPerigosa == true) {  
 273: if (!util.validaRadioSimNao("parada\_especial\_carga", "Utiliza parada especial")) {  
 274: return false;  
 275: }  
 276: }  
 277:  
 278: var ckb\_municipios\_parada = $('#municipios\_parada\_nao\_sei\_carga').is(':checked');  
 279: if (!ckb\_municipios\_parada) {  
 280: // SUGESTAO DE MUNICIPIOS DE PARADA  
 281: var uma\_sugestao = false;  
 282: if (util.validaSelect("municipios\_parada\_uf1\_carga") && util.validaInputText("municipios\_parada\_mun1\_carga")) {  
 283: uma\_sugestao = true;  
 284: }  
 285: if (util.validaSelect("municipios\_parada\_uf2\_carga") && util.validaInputText("municipios\_parada\_mun2\_carga")) {  
 286: uma\_sugestao = true;  
 287: }  
 288: if (!uma\_sugestao) {  
 289: alert("Preencha pelo menos um dos campos de sugestão de parada obrigatória para descanso",  
 290: "Erro no preenchimento!", null, 'error');  
 291: }  
 292: return uma\_sugestao;  
 293: } else {  
 294: return true;  
 295: }  
 296:  
 297: return false;  
 298: },  
 299:  
 300: limitePesoCargaInputText : function(){  
 301: var unidade = $('input[name="unidade\_peso\_carga"]:checked').val();  
 302: if(unidade=="ton"){  
 303: $("#peso\_carga").attr("maxlength",4);  
 304: var peso = $.trim($("#peso\_carga").val());  
 305: if(peso.length>3){  
 306: $("#peso\_carga").val("");  
 307: app.setAtributo("pesoDaCarga",null);  
 308: }  
 309: }else{  
 310: $("#peso\_carga").attr("maxlength",7);  
 311: }  
 312: },  
 313:  
 314: limitePrecoInputText : function(nome\_campo,limite){  
 315: //Numero máximo de pontos por conta da mascara  
 316: var numero\_pts\_msk=Math.floor(Math.max(limite-2,0)/3.0);  
 317: if(limite>2 && (limite-2)%3==0)  
 318: numero\_pts\_msk=numero\_pts\_msk-1;  
 319:  
 320: $("#"+nome\_campo).attr("maxlength",4+limite+numero\_pts\_msk);  
 321: }  
 322:};

## .\mobile\www\controllers\carga\caracterizacao\textunderscore viagem\textunderscore carga.js

1:controllers.caracterizacao\_viagem\_carga = {  
 2: config : function() {  
 3: var me = controllers.caracterizacao\_viagem\_carga;  
 4: me.inicializaElementos();  
 5: me.progressoTela();  
 6: me.buttons();  
 7: },  
 8:  
 9: buttons : function() {  
 10: $("#caracterizacao\_viagem\_carga\_avancar").click(function() {  
 11: var ok = controllers.caracterizacao\_viagem\_carga.validar\_componentes();  
 12: if (ok) {  
 13: app.trocaPagina('views/carga/caracterizacao\_viagem2\_carga.html', controllers.caracterizacao\_viagem2\_carga);  
 14: }  
 15: })  
 16: },  
 17:  
 18: // Inicializa os elementos da tela  
 19: inicializaElementos : function() {  
 20:  
 21: // INICIO PAÍSES  
 22: util.inicializaSelectPais("idOrigemPais", "origem\_pais\_carga", true, "País");  
 23: util.inicializaSelectCustomValueAsIndex("origem\_uf\_carga", lista\_estados, "UF");  
 24:  
 25: util.inicializaSelectPais("idDestinoPais", "destino\_pais\_carga", true, "País");  
 26: util.inicializaSelectCustomValueAsIndex("destino\_uf\_carga", lista\_estados, "UF");  
 27: // FIM PAÍSES  
 28:  
 29: util.inicializaSelectFrequencia('carga');  
 30:  
 31: // var lista\_motivos\_rota = [ 'Asfalto/Sinalização', 'Caminho mais curto', 'Caminho mais rápido', 'Ordens da empresa',  
 32: // 'Ausência de pedágio', 'Proximidade hotéis/postos', 'Segurança', 'Outros' ];  
 33: // util.inicializaSelect("motivo\_rota\_carga", lista\_motivos\_rota);  
 34: util.inicializaTabelaAuxiliar("motivo\_rota\_carga", "Selecione", lista\_motivo\_escolha\_rota, "carga");  
 35: util.limitaTamanhoCampoPorValor('frequencia\_num\_carga','999');  
 36: },  
 37:  
 38: // Controla o show e hide dos elementos da tela  
 39: progressoTela : function() {  
 40:  
 41: // Origem  
 42: util.progressoSelectPais("idOrigemPais", "idOrigemMunicipio", "origem\_pais\_carga", "origem\_uf", "origem\_municipio",  
 43: "grupo\_destino\_carga", "carga");  
 44: util.progressoSelect("origem\_uf", "origem\_uf\_carga", "grupo\_origem\_municipio\_carga");  
 45: $('#origem\_uf\_carga').change(  
 46: function() {  
 47: var estado = $(this).val();  
 48: if (estado != '-1') {  
 49: $('#origem\_municipio\_carga').off("click").click(  
 50: function() {  
 51: util.autocomplete("origem\_municipio\_carga", lista\_municipios[estado], "Município de origem",  
 52: "Entre com o município de origem.");  
 53: }).trigger('click');  
 54: }  
 55: });  
 56: util.progressoInputText("idOrigemMunicipio", "origem\_municipio\_carga", "grupo\_destino\_carga", true);  
 57:  
 58: // Destino  
 59: util.progressoSelectPais("idDestinoPais", "idDestinoMunicipio", "destino\_pais\_carga", "destino\_uf", "destino\_municipio",  
 60: "grupo\_frequencia\_carga", "carga");  
 61: util.progressoSelect("destino\_uf", "destino\_uf\_carga", "grupo\_destino\_municipio\_carga");  
 62: $('#destino\_uf\_carga').change(  
 63: function() {  
 64: var estado = $(this).val();  
 65: if (estado != '-1') {  
 66: $('#destino\_municipio\_carga').off("click").click(  
 67: function() {  
 68: util.autocomplete("destino\_municipio\_carga", lista\_municipios[estado],  
 69: "Município de destino", "Entre com o município de destino.");  
 70: }).trigger('click');  
 71: }  
 72: });  
 73: util.progressoInputText("idDestinoMunicipio", "destino\_municipio\_carga", "grupo\_frequencia\_carga", true);  
 74:  
 75: // Frequencia  
 76: $('#frequencia\_num\_carga').change(function() {  
 77: app.setAtributo('frequenciaQtd', $(this).inputmask('unmaskedvalue'));  
 78: });  
 79: util.progressoSelect("frequenciaPeriodo", "frequencia\_sel\_carga", "grupo\_motivo\_rota\_carga");  
 80:  
 81: // Motivo rota  
 82: util.progressoSelect("idMotivoDeEscolhaDaRota", "motivo\_rota\_carga", "grupo\_caracterizacao\_viagem\_carga\_avancar");  
 83: },  
 84:  
 85: // Controla as validações dos componentes de tela após clicar em AVANÇAR  
 86: validar\_componentes : function(id\_avancar) {  
 87: if (util.validaSelect("origem\_pais\_carga", "Origem da viagem")  
 88: && util.validaSelect("destino\_pais\_carga", "Destino da viagem")  
 89: && util.validaInputText("frequencia\_num\_carga", "Frequência da viagem")  
 90: && util.validaSelect("frequencia\_sel\_carga", "Frequência da viagem")  
 91: && util.validaSelect("motivo\_rota\_carga", "Motivo da escolha da rota")) {  
 92:  
 93: var validacoes = true;  
 94: if ((Number($("#origem\_pais\_carga").val())) == 1) { // Brasil  
 95: validacoes = validacoes  
 96: && (util.validaSelect("origem\_uf\_carga", "Origem da viagem - estado") && util.validaInputText(  
 97: "origem\_municipio\_carga", "Origem da viagem - município"));  
 98: }  
 99:  
 100: if ((Number($("#destino\_pais\_carga").val())) == 1) { // Brasil  
 101: validacoes = validacoes  
 102: && (util.validaSelect("destino\_uf\_carga", "Destino da viagem - estado") && util.validaInputText(  
 103: "destino\_municipio\_carga", "Destino da viagem - município"));  
 104: }  
 105:  
 106: if (Number($("#frequencia\_num\_carga").val()) < 1) {  
 107: validacoes = validacoes  
 108: && (util.validaInputText("frequencia\_num\_carga", "Frequência da viagem"));  
 109: }  
 110:  
 111: return validacoes;  
 112: }  
 113: return false;  
 114: }  
 115:  
 116:};

## .\mobile\www\controllers\carga\identificacao\textunderscore visual\textunderscore carga.js

1:controllers.identificacao\_visual\_carga = {  
 2: config : function() {  
 3: var me = controllers.identificacao\_visual\_carga;  
 4: me.inicializaElementos();  
 5: me.progressoTela();  
 6: me.buttons();  
 7: },  
 8:  
 9: buttons : function() {  
 10: $("#identificacao\_visual\_carga\_avancar").click(function() {  
 11: var ok = controllers.identificacao\_visual\_carga.validar\_componentes();  
 12: if (ok) {  
 13: app.trocaPagina('views/carga/caracterizacao\_carga.html', controllers.caracterizacao\_carga);  
 14: }  
 15: });  
 16:   
 17: $('#tipo\_carroceria\_carga').click(function() {  
 18: $("#grupo\_tipo\_carroceria\_imagens\_carga").show();  
 19: $("#grupo\_tipo\_carroceria\_imagens\_carga").siblings(":not(#grupo\_tipo\_carroceria\_carga)").hide();  
 20: });  
 21:   
 22: },  
 23:  
 24: // Inicializa os elementos da tela  
 25: inicializaElementos : function() {  
 26: util.inicializaSelectPais("idPaisPlacaEstrangeira", "pais\_carga", false);  
 27: util.inicializaPlacas("carga");  
 28:  
 29: var lista\_tipo\_conteiner = [ '1 Conteiner de 20 pés', '1 Conteiner de 20 pés e 1 Conteiner de 40 pés',  
 30: '1 Conteiner de 40 pés', '2 Conteineres de 20 pés' ];  
 31: util.inicializaSelect("tipo\_conteiner\_carga", lista\_tipo\_conteiner)  
 32:  
 33: var lista\_rntrc = [ 'TAC', 'ETC', 'CTC' ];  
 34: util.inicializaSelectCustomValueAsIndex("placa\_vermelha\_rntrc\_sel\_carga", lista\_rntrc, "Selecione");  
 35:   
 36: $("img:not([src\*='outros']).img\_carroceria").each(function(key, value) {  
 37: $(value).after("<span>" + $(value).attr('alt') + "</span>");  
 38: });  
 39: },  
 40:  
 41: // Controla o show e hide dos elementos da tela  
 42: progressoTela : function() {  
 43:  
 44: $(".img\_carroceria").click(function() {  
 45: var imgAlt = $(this).attr('alt');  
 46: var elemTipoCarroceria = $('#tipo\_carroceria\_carga');  
 47: elemTipoCarroceria.val(imgAlt);  
 48: $('label[for="' + elemTipoCarroceria.attr('id') + '"]').text(imgAlt);  
 49:  
 50: app.setAtributo('idTipoCarroceria', $(this).attr('id').split("\_")[1]);  
 51: $("#grupo\_tipo\_carroceria\_imagens\_carga").hide();  
 52: $("#grupo\_tipo\_carroceria\_imagens\_carga").next().show();  
 53: });  
 54:  
 55: $(".tipo\_porta\_conteiner").click(function() {  
 56: $("#grupo\_tipo\_conteiner\_carga").show();  
 57: });  
 58: $(".img\_carroceria:not(.tipo\_porta\_conteiner)").click(function() {  
 59: $("#grupo\_tipo\_conteiner\_carga").hide();  
 60: $("#grupo\_placa\_estrangeira\_carga").show();  
 61: app.setAtributo('idTipoDeContainer', null);  
 62: util.progressoRestartSelect("tipo\_conteiner\_carga", "Selecione");  
 63: });  
 64:  
 65: util.progressoSelect('idTipoDeContainer', "tipo\_conteiner\_carga", "grupo\_placa\_estrangeira\_carga");  
 66:  
 67: util.progressoRadioPlacaEstrangeira("carga");  
 68:  
 69: // País (somente placa estrangeira)  
 70: util.progressoSelect("idPaisPlacaEstrangeira", "pais\_carga", "grupo\_placa\_unica\_carga");  
 71:  
 72: // Placa Brasil  
 73: util.progressoPlacaNumeros("carga", "grupo\_placa\_vermelha\_carga");  
 74:  
 75: // Placa única  
 76: $('#placa\_unica\_carga').keyup(function() {  
 77: if (util.isEmpty($(this).val())) {  
 78: $("#grupo\_carga\_perigosa\_carga").hide();  
 79: $("#placa\_vermelha\_rntrc\_num\_carga").val(null);  
 80: } else {  
 81: $("#grupo\_carga\_perigosa\_carga").show();  
 82: }  
 83: });  
 84: $('#placa\_unica\_carga').change(function() {  
 85: app.setAtributo('placa\_unica', $(this).val());  
 86: app.setAtributo('placaVermelha', false);  
 87: app.setAtributo('placa\_vermelha\_rntrc\_sel', null);  
 88: app.setAtributo('placa\_vermelha\_rntrc\_num', null);  
 89: });  
 90:  
 91: // Placa Vermelha  
 92: $('#placa\_vermelha\_carga\_sim').click(function() {  
 93: $('#grupo\_qr\_code\_carga').show();  
 94: if ((registro.placa\_vermelha == undefined) || (!registro.placa\_vermelha)) {  
 95: $('#grupo\_carga\_perigosa\_carga').hide();  
 96: $('#grupo\_placa\_vermelha\_rntrc\_num\_carga').hide();  
 97: }  
 98: app.setAtributo('placaVermelha', true);  
 99: });  
 100: $('#placa\_vermelha\_carga\_nao').click(function() {  
 101: $('#grupo\_qr\_code\_carga').hide();  
 102: $('#grupo\_placa\_vermelha\_rntrc\_carga').hide();  
 103: $('#grupo\_carga\_perigosa\_carga').show();  
 104: app.setAtributo('placaVermelha', false);  
 105: app.setAtributo('placa\_vermelha\_rntrc\_sel', null);  
 106: app.setAtributo('placa\_vermelha\_rntrc\_num', null);  
 107: $("#placa\_vermelha\_rntrc\_num\_carga").val(null);  
 108: util.progressoRestartSelect("placa\_vermelha\_rntrc\_sel\_carga", "Selecione");  
 109: });  
 110:   
 111: //QR Code  
 112: $('#qr\_code\_carga\_sim').click(function() {  
 113: $('#grupo\_placa\_vermelha\_rntrc\_carga').show();  
 114: app.setAtributo('placa\_vermelha\_rntrc\_num', null);  
 115: $("#placa\_vermelha\_rntrc\_num\_carga").val(null);  
 116: $('#placa\_vermelha\_rntrc\_num\_carga').attr('maxlength','9');  
 117: $('#placa\_vermelha\_rntrc\_num\_carga').attr('minlength','9');  
 118: $('#grupo\_carga\_perigosa\_carga').hide();  
 119: util.progressoRestartSelect("placa\_vermelha\_rntrc\_sel\_carga", "Selecione");  
 120: });  
 121: $('#qr\_code\_carga\_nao').click(function() {  
 122: $('#grupo\_placa\_vermelha\_rntrc\_carga').show();  
 123: app.setAtributo('placa\_vermelha\_rntrc\_num', null);  
 124: $("#placa\_vermelha\_rntrc\_num\_carga").val(null);  
 125: $('#placa\_vermelha\_rntrc\_num\_carga').attr('maxlength','8');  
 126: $('#placa\_vermelha\_rntrc\_num\_carga').attr('minlength','8');  
 127: $('#grupo\_carga\_perigosa\_carga').hide();  
 128: util.progressoRestartSelect("placa\_vermelha\_rntrc\_sel\_carga", "Selecione");  
 129: });  
 130:  
 131: // RNTRC  
 132: $("#placa\_vermelha\_rntrc\_sel\_carga").change(function() {  
 133: var nome\_registro = "placa\_vermelha\_rntrc\_sel";  
 134: if (Number($(this).val()) != -1) {  
 135: app.setAtributo(nome\_registro, $(this).val());  
 136: $("#grupo\_placa\_vermelha\_rntrc\_num\_carga").show();  
 137: } else {  
 138: app.setAtributo(nome\_registro, null);  
 139: app.setAtributo('placa\_vermelha\_rntrc\_num', null);  
 140: $("#placa\_vermelha\_rntrc\_num\_carga").val(null);  
 141: $("#grupo\_placa\_vermelha\_rntrc\_num\_carga").hide();  
 142: $("#grupo\_carga\_perigosa\_carga").hide();  
 143: }  
 144: });  
 145: util.progressoInputTextLen("placa\_vermelha\_rntrc\_num", "placa\_vermelha\_rntrc\_num\_carga", "grupo\_carga\_perigosa\_carga");  
 146:  
 147: // Carga Perigosa  
 148: $('#carga\_perigosa\_carga\_sim').click(function() {  
 149: $('#grupo\_carga\_perigosa\_numeros\_carga').show();  
 150: if ((registro.carga\_perigosa == undefined) || (!registro.carga\_perigosa)) {  
 151: $('#grupo\_identificacao\_visual\_carga\_avancar').hide();  
 152: }  
 153: app.setAtributo('possuiCargaPerigosa', true);  
 154: });  
 155: $('#carga\_perigosa\_carga\_nao').click(function() {  
 156: $('#grupo\_carga\_perigosa\_numeros\_carga').hide();  
 157: $('#grupo\_identificacao\_visual\_carga\_avancar').show();  
 158: app.setAtributo('possuiCargaPerigosa', false);  
 159: app.setAtributo('idNumeroDeRisco', null);  
 160: app.setAtributo('idNumeroDaOnu', null);  
 161: $('#carga\_perigosa\_risco\_carga').val("");  
 162: $('#carga\_perigosa\_onu\_carga').val("");  
 163: });  
 164: util.progressoInputText("idNumeroDeRisco", "carga\_perigosa\_risco\_carga", "grupo\_carga\_perigosa\_onu\_carga");  
 165: util.progressoInputText("idNumeroDaOnu", "carga\_perigosa\_onu\_carga", "grupo\_identificacao\_visual\_carga\_avancar");  
 166: },  
 167:  
 168: // Controla as validações dos componentes de tela após clicar em AVANÇAR  
 169: validar\_componentes : function(id\_avancar) {  
 170:  
 171: if (util.validaRadioChecked("tipo\_carroceria\_carga", "Tipo de carroceria")  
 172: && util.validaRadioSimNao("placa\_estrangeira\_carga", "Placa estrangeira")  
 173: && util.validaRadioSimNao("carga\_perigosa\_carga", "Carga perigosa")) {  
 174:  
 175: var ok\_tipo\_conteiner = true;  
 176: var opt\_conteiner = $('input[name=tipo\_carroceria\_carga]').val();  
 177: if (opt\_conteiner == 'Porta-Conteiner') {  
 178: if (!util.validaSelect("tipo\_conteiner\_carga", "Porta-Conteiner")) {  
 179: ok\_tipo\_conteiner = false;  
 180: }  
 181: }  
 182:  
 183: var ok\_placa\_estrangeira = true;  
 184: var opt\_estrangeira = $('input[name=placa\_estrangeira\_carga]:checked').val();  
 185: if (opt\_estrangeira == 'sim' && util.validaSelect("pais\_carga", "País")) {  
 186: if ((Number($("#pais\_carga").val())) == 1) { // Brasil  
 187: alert("O país do veículo de placa estrangeira não pode ser Brasil");  
 188: ok\_placa\_estrangeira = false;  
 189: }  
 190: }  
 191:  
 192: var ok\_placa\_estrangeira = true;  
 193: var ok\_placa\_veiculo = true;  
 194: var option = $('input[name=placa\_estrangeira\_carga]:checked').val();  
 195: if (option == 'sim') {  
 196: if ((Number($("#pais\_carga").val())) == 1) { // Brasil  
 197: alert("O país do veículo de placa estrangeira não pode ser Brasil");  
 198: ok\_placa\_estrangeira = false;  
 199: }  
 200:  
 201: if (!util.validaSelect("pais\_carga", "País") || !util.validaInputText("placa\_unica\_carga", "Placa do veículo")) {  
 202: ok\_placa\_veiculo = false;  
 203: }  
 204: } else if (option == 'nao') {  
 205: if (!util.validaInputText("placa\_letras\_carga", "Placa do veículo (letras)")  
 206: || !util.validaInputText("placa\_numeros\_carga", "Placa do veículo (números)")  
 207: || !util.validaLenInputText("placa\_letras\_carga", "Placa do veículo (letras)")  
 208: || !util.validaLenInputText("placa\_numeros\_carga", "Placa do veículo (números)")) {  
 209: ok\_placa\_veiculo = false;  
 210: }  
 211: }  
 212:  
 213: var ok\_placa\_vermelha = true;  
 214: if (option == 'nao') {  
 215: ok\_placa\_vermelha = util.validaRadioSimNao("placa\_vermelha\_carga", "Placa vermelha");  
 216: var opt\_vermelha = $('input[name=placa\_vermelha\_carga]:checked').val();  
 217: if (opt\_vermelha == 'sim') {  
 218: if (!util.validaSelect("placa\_vermelha\_rntrc\_sel\_carga", "RNTRC")) {  
 219: ok\_placa\_vermelha = false;  
 220: }  
 221: if (!util.validaInputText("placa\_vermelha\_rntrc\_num\_carga", "Número do RNTRC")  
 222: || !util.validaLenInputText("placa\_vermelha\_rntrc\_num\_carga", "Número do RNTRC")) {  
 223: ok\_placa\_vermelha = false;  
 224: }  
 225: }  
 226: }  
 227:  
 228: var ok\_carga\_perigosa = true;  
 229: var opt\_perigosa = $('input[name=carga\_perigosa\_carga]:checked').val();  
 230: if (opt\_perigosa == 'sim') {  
 231: if (!util.validaInputText("carga\_perigosa\_risco\_carga", "Número de Risco")  
 232: || !util.validaInputText("carga\_perigosa\_onu\_carga", "Número da ONU")) {  
 233: ok\_carga\_perigosa = false;  
 234: }  
 235: if (!util.validaValueInList("carga\_perigosa\_risco\_carga", "Número de Risco", lista\_numero\_risco, "idNumeroDeRisco")  
 236: || !util.validaValueInList("carga\_perigosa\_onu\_carga", "Número da ONU", lista\_numero\_onu, "idNumeroDaOnu")) {  
 237: ok\_carga\_perigosa = false;  
 238: }  
 239: }  
 240:  
 241: return ok\_tipo\_conteiner && ok\_placa\_estrangeira && ok\_placa\_veiculo && ok\_placa\_vermelha && ok\_carga\_perigosa;  
 242: }  
 243:  
 244: return false;  
 245: }  
 246:};

## .\mobile\www\controllers\onibus\caracterizacao\textunderscore onibus.js

1:/\* global util, registro \*/  
 2:  
 3:controllers.caracterizacao\_onibus = {  
 4: config : function() {  
 5: var me = controllers.caracterizacao\_onibus;  
 6: me.inicializaElementos();  
 7: me.progressoTela();  
 8: me.buttons();  
 9: },  
 10:  
 11: buttons : function() {  
 12: $("#caracterizacao\_onibus\_avancar").click(function() {  
 13: var ok = controllers.caracterizacao\_onibus.validar\_componentes();  
 14: if (ok) {  
 15: app.trocaPagina('views/onibus/caracterizacao\_viagem\_onibus.html', controllers.caracterizacao\_viagem\_onibus);  
 16: }  
 17: })  
 18: },  
 19:  
 20: // Inicializa os elementos da tela  
 21: inicializaElementos : function() {  
 22:  
 23:// var lista\_anos = [];  
 24:// for (var ano = 2017; ano > 1899; ano--) {  
 25:// lista\_anos = lista\_anos.concat(ano);  
 26:// }  
 27:  
 28: var lista\_anos = util.geraListaAnos();  
 29:   
 30: util.inicializaSelectCustomValueAsIndex("ano\_onibus", lista\_anos, "Selecione");  
 31:  
 32: util.inicializaTabelaAuxiliar("propriedade\_onibus", "Selecione", lista\_propriedade\_veiculo, "onibus");  
 33:  
 34: },  
 35:  
 36: // Controla o show e hide dos elementos da tela  
 37: progressoTela : function() {  
 38:  
 39: util.progressoSelect("anoDeFabricacao", "ano\_onibus", "grupo\_propriedade\_onibus");  
 40: util.progressoSelect("idPropriedadesDoVeiculo", "propriedade\_onibus", "grupo\_caracterizacao\_onibus\_avancar");  
 41: },  
 42:  
 43: // Controla as validações dos componentes de tela após clicar em AVANÇAR  
 44: validar\_componentes : function(id\_avancar) {  
 45: if (util.validaSelect("ano\_onibus", "Ano do veículo") && util.validaSelect("propriedade\_onibus", "Propriedade")) {  
 46: return true;  
 47: }  
 48: return false;  
 49: }  
 50:  
 51:};

## .\mobile\www\controllers\onibus\caracterizacao\textunderscore viagem\textunderscore onibus.js

1:controllers.caracterizacao\_viagem\_onibus = {  
 2: config : function() {  
 3: var me = controllers.caracterizacao\_viagem\_onibus;  
 4: me.inicializaElementos();  
 5: me.progressoTela();  
 6: me.buttons();  
 7: },  
 8:  
 9: buttons : function() {  
 10: $("#caracterizacao\_viagem\_onibus\_avancar").click(function() {  
 11: var ok = controllers.caracterizacao\_viagem\_onibus.validar\_componentes();  
 12: if (ok) {  
 13: if (logins.user\_logado != null && logins.user\_logado.perguntaExtra) {  
 14: app.trocaPagina('views/pergunta\_extra.html', controllers.pergunta\_extra);  
 15: } else {  
 16: app.finalizaRegistro(function() {  
 17: app.trocaPagina('views/menu.html', controllers.menu);  
 18: });  
 19: }  
 20: }  
 21: })  
 22: },  
 23:  
 24: // Inicializa os elementos da tela  
 25: inicializaElementos : function() {  
 26:  
 27: // Verifica se possui Pergunta Extra  
 28: if (logins.user\_logado != null && logins.user\_logado.perguntaExtra) {  
 29: $("#caracterizacao\_viagem\_onibus\_avancar").html("Avançar");  
 30: }  
 31:  
 32: // INICIO PAÍSES  
 33: util.inicializaSelectPais("idOrigemPais", "origem\_pais\_onibus", true, "País");  
 34: util.inicializaSelectCustomValueAsIndex("origem\_uf\_onibus", lista\_estados, "UF");  
 35:  
 36: util.inicializaSelectPais("idDestinoPais", "destino\_pais\_onibus", true, "País");  
 37: util.inicializaSelectCustomValueAsIndex("destino\_uf\_onibus", lista\_estados, "UF");  
 38: // FIM PAÍSES  
 39:  
 40: util.inicializaSelectFrequencia('onibus');  
 41:  
 42: var lista\_tipo\_viagem = [ 'Regular', 'Turismo', 'Fretado/Particular', 'Outros' ];  
 43: util.inicializaSelect("tipo\_viagem\_onibus", lista\_tipo\_viagem);  
 44: util.limitaTamanhoCampoPorValor('frequencia\_num\_onibus','999');  
 45: },  
 46:  
 47: // Controla o show e hide dos elementos da tela  
 48: progressoTela : function() {  
 49:  
 50: // Origem  
 51: util.progressoSelectPais("idOrigemPais", "idOrigemMunicipio", "origem\_pais\_onibus", "origem\_uf", "origem\_municipio",  
 52: "grupo\_destino\_onibus", "onibus");  
 53: util.progressoSelect("origem\_uf", "origem\_uf\_onibus", "grupo\_origem\_municipio\_onibus");  
 54: $('#origem\_uf\_onibus').change(  
 55: function() {  
 56: var estado = $(this).val();  
 57: if (estado != '-1') {  
 58: $('#origem\_municipio\_onibus').off("click").click(  
 59: function() {  
 60: util.autocomplete("origem\_municipio\_onibus", lista\_municipios[estado], "Município de origem",  
 61: "Entre com o município de origem.");  
 62: }).trigger('click')  
 63: }  
 64: });  
 65: util.progressoInputText("idOrigemMunicipio", "origem\_municipio\_onibus", "grupo\_destino\_onibus", true);  
 66:  
 67: // Destino  
 68: util.progressoSelectPais("idDestinoPais", "idDestinoMunicipio", "destino\_pais\_onibus", "destino\_uf", "destino\_municipio",  
 69: "grupo\_frequencia\_onibus", "onibus");  
 70: util.progressoSelect("destino\_uf", "destino\_uf\_onibus", "grupo\_destino\_municipio\_onibus");  
 71: $('#destino\_uf\_onibus').change(  
 72: function() {  
 73: var estado = $(this).val();  
 74: if (estado != '-1') {  
 75: $('#destino\_municipio\_onibus').off("click").click(  
 76: function() {  
 77: util.autocomplete("destino\_municipio\_onibus", lista\_municipios[estado],  
 78: "Município de destino", "Entre com o município de destino.");  
 79: }).trigger('click');  
 80: }  
 81: });  
 82: util.progressoInputText("idDestinoMunicipio", "destino\_municipio\_onibus", "grupo\_frequencia\_onibus", true);  
 83:  
 84: // Frequencia  
 85: $('#frequencia\_num\_onibus').change(function() {  
 86: app.setAtributo('frequenciaQtd', $(this).inputmask('unmaskedvalue'));  
 87: });  
 88: util.progressoSelect("frequenciaPeriodo", "frequencia\_sel\_onibus", "grupo\_tipo\_viagem\_onibus");  
 89:  
 90: util.progressoSelect("idTipoDeViagemOuServico", "tipo\_viagem\_onibus", "grupo\_pessoas\_onibus");  
 91:   
 92: $('#tipo\_viagem\_onibus').change(function() {  
 93: if (Number($(this).val()) != -1) {  
 94: util.limitaTamanhoCampo('pessoas\_onibus','limitePessoas');  
 95: }  
 96: });  
 97:  
 98: util.progressoInputText("numeroDePessoasNoVeiculo", "pessoas\_onibus", "grupo\_caracterizacao\_viagem\_onibus\_avancar");  
 99: },  
 100:  
 101: // Controla as validações dos componentes de tela após clicar em AVANÇAR  
 102: validar\_componentes : function(id\_avancar) {  
 103:  
 104: if (util.validaSelect("origem\_pais\_onibus", "Origem da viagem")  
 105: && util.validaSelect("destino\_pais\_onibus", "Destino da viagem")  
 106: && util.validaInputText("frequencia\_num\_onibus", "Frequência da viagem")  
 107: && util.validaSelect("frequencia\_sel\_onibus", "Frequência da viagem")  
 108: && util.validaSelect("tipo\_viagem\_onibus", "Tipo de viagem/serviço")  
 109: && util.validaInputText("pessoas\_onibus", "Pessoas no veículo")) {  
 110:  
 111: var validacoes = true;  
 112: if (Number($("#origem\_pais\_onibus").val()) == 1) { // Brasil  
 113: validacoes = validacoes  
 114: && (util.validaSelect("origem\_uf\_onibus", "Origem da viagem - estado") && util.validaInputText(  
 115: "origem\_municipio\_onibus", "Origem da viagem - município"));  
 116: }  
 117:  
 118: if (Number($("#destino\_pais\_onibus").val()) == 1) { // Brasil  
 119: validacoes = validacoes  
 120: && (util.validaSelect("destino\_uf\_onibus", "Destino da viagem - estado") && util.validaInputText(  
 121: "destino\_municipio\_onibus", "Destino da viagem - município"));  
 122: }  
 123:  
 124: if (Number($("#frequencia\_num\_onibus").val()) < 1) {  
 125: validacoes = validacoes  
 126: && (util.validaInputText("frequencia\_num\_onibus", "Frequência da viagem"));  
 127: }  
 128:  
 129: validacoes = validacoes && util.validaTemPessoasVeiculo("pessoas\_onibus", "Pessoas no veículo");  
 130: validacoes = validacoes && util.validaLimitePessoas("pessoas\_onibus", "Pessoas no veículo");  
 131:  
 132: return validacoes;  
 133: }  
 134: return false;  
 135: }  
 136:  
 137:};

## .\mobile\www\controllers\onibus\identificacao\textunderscore visual\textunderscore onibus.js

1:controllers.identificacao\_visual\_onibus = {  
 2: config : function() {  
 3: var me = controllers.identificacao\_visual\_onibus;  
 4: me.inicializaElementos();  
 5: me.progressoTela();  
 6: me.buttons();  
 7: },  
 8:  
 9: buttons : function() {  
 10: $("#identificacao\_visual\_onibus\_avancar").click(function() {  
 11: var ok = controllers.identificacao\_visual\_onibus.validar\_componentes();  
 12: if (ok) {  
 13: app.trocaPagina('views/onibus/caracterizacao\_onibus.html', controllers.caracterizacao\_onibus);  
 14: }  
 15: })  
 16: },  
 17:  
 18: // Inicializa os elementos da tela  
 19: inicializaElementos : function() {  
 20: util.inicializaSelectPais("idPaisPlacaEstrangeira", "pais\_onibus", false);  
 21: util.inicializaPlacas("onibus");  
 22: },  
 23:  
 24: // Controla o show e hide dos elementos da tela  
 25: progressoTela : function() {  
 26:  
 27: util.progressoRadioPlacaEstrangeira("onibus");  
 28:  
 29: // País (somente placa estrangeira)  
 30: util.progressoSelect("idPaisPlacaEstrangeira", "pais\_onibus", "grupo\_placa\_unica\_onibus");  
 31:  
 32: // Placa Brasil  
 33: util.progressoPlacaNumeros("onibus", "grupo\_identificacao\_visual\_onibus\_avancar");  
 34:  
 35: // Placa única  
 36: util.progressoInputText("placa\_unica", "placa\_unica\_onibus", "grupo\_identificacao\_visual\_onibus\_avancar");  
 37: },  
 38:  
 39: // Controla as validações dos componentes de tela após clicar em AVANÇAR  
 40: validar\_componentes : function(id\_avancar) {  
 41:  
 42: if (util.validaRadioSimNao("placa\_estrangeira\_onibus", "Placa estrangeira")) {  
 43: var option = $('input[name=placa\_estrangeira\_onibus]:checked').val();  
 44: if (option == 'sim') {  
 45: var ok\_placa\_estrangeira = true;  
 46: if ((Number($("#pais\_onibus").val())) == 1) { // Brasil  
 47: alert("O país do veículo de placa estrangeira não pode ser Brasil");  
 48: ok\_placa\_estrangeira = false;  
 49: }  
 50:  
 51: return ok\_placa\_estrangeira && util.validaSelect("pais\_onibus", "País")  
 52: && util.validaInputText("placa\_unica\_onibus", "Placa do veículo");  
 53: } else if (option == 'nao') {  
 54: return util.validaInputText("placa\_letras\_onibus", "Placa do veículo (letras)")  
 55: && util.validaInputText("placa\_numeros\_onibus", "Placa do veículo (números)")  
 56: && util.validaLenInputText("placa\_letras\_onibus", "Placa do veículo (letras)")  
 57: && util.validaLenInputText("placa\_numeros\_onibus", "Placa do veículo (números)");  
 58: }  
 59:  
 60: return true;  
 61: }  
 62: return false;  
 63: }  
 64:};

## .\mobile\www\js\carga\textunderscore perigosa.js

1:var lista\_numero\_risco = [  
 2: {id : '1', numeroid : '20'},  
 3: {id : '2', numeroid : '22'},  
 4: {id : '3', numeroid : '223'},  
 5: {id : '4', numeroid : '225'},  
 6: {id : '5', numeroid : '23'},  
 7: {id : '6', numeroid : '236'},  
 8: {id : '7', numeroid : '239'},  
 9: {id : '8', numeroid : '25'},  
 10: {id : '9', numeroid : '26'},  
 11: {id : '10', numeroid : '263'},  
 12: {id : '11', numeroid : '265'},  
 13: {id : '12', numeroid : '266'},  
 14: {id : '13', numeroid : '268'},  
 15: {id : '14', numeroid : '286'},  
 16: {id : '15', numeroid : '30'},  
 17: {id : '16', numeroid : '323'},  
 18: {id : '17', numeroid : 'X323'},  
 19: {id : '18', numeroid : '33'},  
 20: {id : '19', numeroid : '333'},  
 21: {id : '20', numeroid : 'X333'},  
 22: {id : '21', numeroid : '336'},  
 23: {id : '22', numeroid : '338'},  
 24: {id : '23', numeroid : 'X338'},  
 25: {id : '24', numeroid : '339'},  
 26: {id : '25', numeroid : '36'},  
 27: {id : '26', numeroid : '362'},  
 28: {id : '27', numeroid : 'X362'},  
 29: {id : '28', numeroid : '368'},  
 30: {id : '29', numeroid : '38'},  
 31: {id : '30', numeroid : '382'},  
 32: {id : '31', numeroid : 'X382'},  
 33: {id : '32', numeroid : '39'},  
 34: {id : '33', numeroid : '40'},  
 35: {id : '34', numeroid : '423'},  
 36: {id : '35', numeroid : 'X423'},  
 37: {id : '36', numeroid : '43'},  
 38: {id : '37', numeroid : '44'},  
 39: {id : '38', numeroid : '446'},  
 40: {id : '39', numeroid : '46'},  
 41: {id : '40', numeroid : '462'},  
 42: {id : '41', numeroid : 'X462'},  
 43: {id : '42', numeroid : '48'},  
 44: {id : '43', numeroid : '482'},  
 45: {id : '44', numeroid : 'X482'},  
 46: {id : '45', numeroid : '50'},  
 47: {id : '46', numeroid : '539'},  
 48: {id : '47', numeroid : '55'},  
 49: {id : '48', numeroid : '556'},  
 50: {id : '49', numeroid : '558'},  
 51: {id : '50', numeroid : '559'},  
 52: {id : '51', numeroid : '56'},  
 53: {id : '52', numeroid : '568'},  
 54: {id : '53', numeroid : '58'},  
 55: {id : '54', numeroid : '59'},  
 56: {id : '55', numeroid : '60'},  
 57: {id : '56', numeroid : '606'},  
 58: {id : '57', numeroid : '623'},  
 59: {id : '58', numeroid : '63'},  
 60: {id : '59', numeroid : '638'},  
 61: {id : '60', numeroid : '639'},  
 62: {id : '61', numeroid : '66'},  
 63: {id : '62', numeroid : '642'},  
 64: {id : '63', numeroid : '65'},  
 65: {id : '64', numeroid : '66'},  
 66: {id : '65', numeroid : '663'},  
 67: {id : '66', numeroid : '664'},  
 68: {id : '67', numeroid : '665'},  
 69: {id : '68', numeroid : '668'},  
 70: {id : '69', numeroid : '669'},  
 71: {id : '70', numeroid : '68'},  
 72: {id : '71', numeroid : '69'},  
 73: {id : '72', numeroid : '70'},  
 74: {id : '73', numeroid : '72'},  
 75: {id : '74', numeroid : '723'},  
 76: {id : '75', numeroid : '73'},  
 77: {id : '76', numeroid : '74'},  
 78: {id : '77', numeroid : '75'},  
 79: {id : '78', numeroid : '76'},  
 80: {id : '79', numeroid : '78'},  
 81: {id : '80', numeroid : '80'},  
 82: {id : '81', numeroid : 'X80'},  
 83: {id : '82', numeroid : '823'},  
 84: {id : '83', numeroid : '83'},  
 85: {id : '84', numeroid : 'X83'},  
 86: {id : '85', numeroid : '839'},  
 87: {id : '86', numeroid : 'X839'},  
 88: {id : '87', numeroid : '84'},  
 89: {id : '88', numeroid : '842'},  
 90: {id : '89', numeroid : '85'},  
 91: {id : '90', numeroid : '856'},  
 92: {id : '91', numeroid : '86'},  
 93: {id : '92', numeroid : '88'},  
 94: {id : '93', numeroid : 'X88'},  
 95: {id : '94', numeroid : '883'},  
 96: {id : '95', numeroid : '884'},  
 97: {id : '96', numeroid : '885'},  
 98: {id : '97', numeroid : '886'},  
 99: {id : '98', numeroid : 'X886'},  
 100: {id : '99', numeroid : '89'},  
 101: {id : '100', numeroid : '90'},  
 102: {id : '101', numeroid : '99'}   
 103:];  
 104:  
 105:var lista\_numero\_onu = [  
 106: {id : '1', numeroid : '4'},  
 107: {id : '2', numeroid : '5'},  
 108: {id : '3', numeroid : '6'},  
 109: {id : '4', numeroid : '7'},  
 110: {id : '5', numeroid : '9'},  
 111: {id : '6', numeroid : '10'},  
 112: {id : '7', numeroid : '12'},  
 113: {id : '8', numeroid : '14'},  
 114: {id : '9', numeroid : '15'},  
 115: {id : '10', numeroid : '16'},  
 116: {id : '11', numeroid : '18'},  
 117: {id : '12', numeroid : '19'},  
 118: {id : '13', numeroid : '20'},  
 119: {id : '14', numeroid : '21'},  
 120: {id : '15', numeroid : '27'},  
 121: {id : '16', numeroid : '28'},  
 122: {id : '17', numeroid : '29'},  
 123: {id : '18', numeroid : '30'},  
 124: {id : '19', numeroid : '33'},  
 125: {id : '20', numeroid : '34'},  
 126: {id : '21', numeroid : '35'},  
 127: {id : '22', numeroid : '37'},  
 128: {id : '23', numeroid : '38'},  
 129: {id : '24', numeroid : '39'},  
 130: {id : '25', numeroid : '42'},  
 131: {id : '26', numeroid : '43'},  
 132: {id : '27', numeroid : '44'},  
 133: {id : '28', numeroid : '48'},  
 134: {id : '29', numeroid : '49'},  
 135: {id : '30', numeroid : '50'},  
 136: {id : '31', numeroid : '54'},  
 137: {id : '32', numeroid : '55'},  
 138: {id : '33', numeroid : '56'},  
 139: {id : '34', numeroid : '59'},  
 140: {id : '35', numeroid : '60'},  
 141: {id : '36', numeroid : '65'},  
 142: {id : '37', numeroid : '66'},  
 143: {id : '38', numeroid : '70'},  
 144: {id : '39', numeroid : '72'},  
 145: {id : '40', numeroid : '73'},  
 146: {id : '41', numeroid : '74'},  
 147: {id : '42', numeroid : '75'},  
 148: {id : '43', numeroid : '76'},  
 149: {id : '44', numeroid : '77'},  
 150: {id : '45', numeroid : '78'},  
 151: {id : '46', numeroid : '79'},  
 152: {id : '47', numeroid : '81'},  
 153: {id : '48', numeroid : '82'},  
 154: {id : '49', numeroid : '83'},  
 155: {id : '50', numeroid : '84'},  
 156: {id : '51', numeroid : '92'},  
 157: {id : '52', numeroid : '93'},  
 158: {id : '53', numeroid : '94'},  
 159: {id : '54', numeroid : '99'},  
 160: {id : '55', numeroid : '101'},  
 161: {id : '56', numeroid : '102'},  
 162: {id : '57', numeroid : '103'},  
 163: {id : '58', numeroid : '104'},  
 164: {id : '59', numeroid : '105'},  
 165: {id : '60', numeroid : '106'},  
 166: {id : '61', numeroid : '107'},  
 167: {id : '62', numeroid : '110'},  
 168: {id : '63', numeroid : '113'},  
 169: {id : '64', numeroid : '114'},  
 170: {id : '65', numeroid : '118'},  
 171: {id : '66', numeroid : '121'},  
 172: {id : '67', numeroid : '124'},  
 173: {id : '68', numeroid : '129'},  
 174: {id : '69', numeroid : '130'},  
 175: {id : '70', numeroid : '131'},  
 176: {id : '71', numeroid : '132'},  
 177: {id : '72', numeroid : '133'},  
 178: {id : '73', numeroid : '135'},  
 179: {id : '74', numeroid : '136'},  
 180: {id : '75', numeroid : '137'},  
 181: {id : '76', numeroid : '138'},  
 182: {id : '77', numeroid : '143'},  
 183: {id : '78', numeroid : '144'},  
 184: {id : '79', numeroid : '146'},  
 185: {id : '80', numeroid : '147'},  
 186: {id : '81', numeroid : '150'},  
 187: {id : '82', numeroid : '151'},  
 188: {id : '83', numeroid : '153'},  
 189: {id : '84', numeroid : '154'},  
 190: {id : '85', numeroid : '155'},  
 191: {id : '86', numeroid : '158'},  
 192: {id : '87', numeroid : '159'},  
 193: {id : '88', numeroid : '160'},  
 194: {id : '89', numeroid : '161'},  
 195: {id : '90', numeroid : '167'},  
 196: {id : '91', numeroid : '168'},  
 197: {id : '92', numeroid : '169'},  
 198: {id : '93', numeroid : '171'},  
 199: {id : '94', numeroid : '173'},  
 200: {id : '95', numeroid : '174'},  
 201: {id : '96', numeroid : '180'},  
 202: {id : '97', numeroid : '181'},  
 203: {id : '98', numeroid : '182'},  
 204: {id : '99', numeroid : '183'},  
 205: {id : '100', numeroid : '186'},  
 206: {id : '101', numeroid : '190'},  
 207: {id : '102', numeroid : '191'},  
 208: {id : '103', numeroid : '192'},  
 209: {id : '104', numeroid : '193'},  
 210: {id : '105', numeroid : '194'},  
 211: {id : '106', numeroid : '195'},  
 212: {id : '107', numeroid : '196'},  
 213: {id : '108', numeroid : '197'},  
 214: {id : '109', numeroid : '203'},  
 215: {id : '110', numeroid : '204'},  
 216: {id : '111', numeroid : '207'},  
 217: {id : '112', numeroid : '208'},  
 218: {id : '113', numeroid : '209'},  
 219: {id : '114', numeroid : '212'},  
 220: {id : '115', numeroid : '213'},  
 221: {id : '116', numeroid : '214'},  
 222: {id : '117', numeroid : '215'},  
 223: {id : '118', numeroid : '216'},  
 224: {id : '119', numeroid : '217'},  
 225: {id : '120', numeroid : '218'},  
 226: {id : '121', numeroid : '219'},  
 227: {id : '122', numeroid : '220'},  
 228: {id : '123', numeroid : '221'},  
 229: {id : '124', numeroid : '222'},  
 230: {id : '125', numeroid : '223'},  
 231: {id : '126', numeroid : '224'},  
 232: {id : '127', numeroid : '225'},  
 233: {id : '128', numeroid : '226'},  
 234: {id : '129', numeroid : '234'},  
 235: {id : '130', numeroid : '235'},  
 236: {id : '131', numeroid : '236'},  
 237: {id : '132', numeroid : '237'},  
 238: {id : '133', numeroid : '238'},  
 239: {id : '134', numeroid : '240'},  
 240: {id : '135', numeroid : '241'},  
 241: {id : '136', numeroid : '242'},  
 242: {id : '137', numeroid : '243'},  
 243: {id : '138', numeroid : '244'},  
 244: {id : '139', numeroid : '245'},  
 245: {id : '140', numeroid : '246'},  
 246: {id : '141', numeroid : '247'},  
 247: {id : '142', numeroid : '248'},  
 248: {id : '143', numeroid : '249'},  
 249: {id : '144', numeroid : '250'},  
 250: {id : '145', numeroid : '254'},  
 251: {id : '146', numeroid : '255'},  
 252: {id : '147', numeroid : '257'},  
 253: {id : '148', numeroid : '266'},  
 254: {id : '149', numeroid : '267'},  
 255: {id : '150', numeroid : '268'},  
 256: {id : '151', numeroid : '271'},  
 257: {id : '152', numeroid : '272'},  
 258: {id : '153', numeroid : '275'},  
 259: {id : '154', numeroid : '276'},  
 260: {id : '155', numeroid : '277'},  
 261: {id : '156', numeroid : '278'},  
 262: {id : '157', numeroid : '279'},  
 263: {id : '158', numeroid : '280'},  
 264: {id : '159', numeroid : '281'},  
 265: {id : '160', numeroid : '282'},  
 266: {id : '161', numeroid : '283'},  
 267: {id : '162', numeroid : '284'},  
 268: {id : '163', numeroid : '285'},  
 269: {id : '164', numeroid : '286'},  
 270: {id : '165', numeroid : '287'},  
 271: {id : '166', numeroid : '288'},  
 272: {id : '167', numeroid : '289'},  
 273: {id : '168', numeroid : '290'},  
 274: {id : '169', numeroid : '291'},  
 275: {id : '170', numeroid : '292'},  
 276: {id : '171', numeroid : '293'},  
 277: {id : '172', numeroid : '294'},  
 278: {id : '173', numeroid : '295'},  
 279: {id : '174', numeroid : '296'},  
 280: {id : '175', numeroid : '297'},  
 281: {id : '176', numeroid : '299'},  
 282: {id : '177', numeroid : '300'},  
 283: {id : '178', numeroid : '301'},  
 284: {id : '179', numeroid : '303'},  
 285: {id : '180', numeroid : '305'},  
 286: {id : '181', numeroid : '306'},  
 287: {id : '182', numeroid : '312'},  
 288: {id : '183', numeroid : '313'},  
 289: {id : '184', numeroid : '314'},  
 290: {id : '185', numeroid : '315'},  
 291: {id : '186', numeroid : '316'},  
 292: {id : '187', numeroid : '317'},  
 293: {id : '188', numeroid : '318'},  
 294: {id : '189', numeroid : '319'},  
 295: {id : '190', numeroid : '320'},  
 296: {id : '191', numeroid : '321'},  
 297: {id : '192', numeroid : '322'},  
 298: {id : '193', numeroid : '323'},  
 299: {id : '194', numeroid : '324'},  
 300: {id : '195', numeroid : '325'},  
 301: {id : '196', numeroid : '326'},  
 302: {id : '197', numeroid : '327'},  
 303: {id : '198', numeroid : '328'},  
 304: {id : '199', numeroid : '329'},  
 305: {id : '200', numeroid : '330'},  
 306: {id : '201', numeroid : '331'},  
 307: {id : '202', numeroid : '332'},  
 308: {id : '203', numeroid : '333'},  
 309: {id : '204', numeroid : '334'},  
 310: {id : '205', numeroid : '335'},  
 311: {id : '206', numeroid : '336'},  
 312: {id : '207', numeroid : '337'},  
 313: {id : '208', numeroid : '338'},  
 314: {id : '209', numeroid : '339'},  
 315: {id : '210', numeroid : '340'},  
 316: {id : '211', numeroid : '341'},  
 317: {id : '212', numeroid : '342'},  
 318: {id : '213', numeroid : '343'},  
 319: {id : '214', numeroid : '344'},  
 320: {id : '215', numeroid : '345'},  
 321: {id : '216', numeroid : '346'},  
 322: {id : '217', numeroid : '347'},  
 323: {id : '218', numeroid : '348'},  
 324: {id : '219', numeroid : '349'},  
 325: {id : '220', numeroid : '350'},  
 326: {id : '221', numeroid : '351'},  
 327: {id : '222', numeroid : '352'},  
 328: {id : '223', numeroid : '353'},  
 329: {id : '224', numeroid : '354'},  
 330: {id : '225', numeroid : '355'},  
 331: {id : '226', numeroid : '356'},  
 332: {id : '227', numeroid : '357'},  
 333: {id : '228', numeroid : '358'},  
 334: {id : '229', numeroid : '359'},  
 335: {id : '230', numeroid : '360'},  
 336: {id : '231', numeroid : '361'},  
 337: {id : '232', numeroid : '362'},  
 338: {id : '233', numeroid : '363'},  
 339: {id : '234', numeroid : '364'},  
 340: {id : '235', numeroid : '365'},  
 341: {id : '236', numeroid : '366'},  
 342: {id : '237', numeroid : '367'},  
 343: {id : '238', numeroid : '368'},  
 344: {id : '239', numeroid : '369'},  
 345: {id : '240', numeroid : '370'},  
 346: {id : '241', numeroid : '371'},  
 347: {id : '242', numeroid : '372'},  
 348: {id : '243', numeroid : '373'},  
 349: {id : '244', numeroid : '374'},  
 350: {id : '245', numeroid : '375'},  
 351: {id : '246', numeroid : '376'},  
 352: {id : '247', numeroid : '377'},  
 353: {id : '248', numeroid : '378'},  
 354: {id : '249', numeroid : '379'},  
 355: {id : '250', numeroid : '380'},  
 356: {id : '251', numeroid : '381'},  
 357: {id : '252', numeroid : '382'},  
 358: {id : '253', numeroid : '383'},  
 359: {id : '254', numeroid : '384'},  
 360: {id : '255', numeroid : '385'},  
 361: {id : '256', numeroid : '386'},  
 362: {id : '257', numeroid : '387'},  
 363: {id : '258', numeroid : '388'},  
 364: {id : '259', numeroid : '389'},  
 365: {id : '260', numeroid : '390'},  
 366: {id : '261', numeroid : '391'},  
 367: {id : '262', numeroid : '392'},  
 368: {id : '263', numeroid : '393'},  
 369: {id : '264', numeroid : '394'},  
 370: {id : '265', numeroid : '395'},  
 371: {id : '266', numeroid : '396'},  
 372: {id : '267', numeroid : '397'},  
 373: {id : '268', numeroid : '398'},  
 374: {id : '269', numeroid : '399'},  
 375: {id : '270', numeroid : '400'},  
 376: {id : '271', numeroid : '401'},  
 377: {id : '272', numeroid : '402'},  
 378: {id : '273', numeroid : '403'},  
 379: {id : '274', numeroid : '404'},  
 380: {id : '275', numeroid : '405'},  
 381: {id : '276', numeroid : '406'},  
 382: {id : '277', numeroid : '407'},  
 383: {id : '278', numeroid : '408'},  
 384: {id : '279', numeroid : '409'},  
 385: {id : '280', numeroid : '410'},  
 386: {id : '281', numeroid : '411'},  
 387: {id : '282', numeroid : '412'},  
 388: {id : '283', numeroid : '413'},  
 389: {id : '284', numeroid : '414'},  
 390: {id : '285', numeroid : '415'},  
 391: {id : '286', numeroid : '417'},  
 392: {id : '287', numeroid : '418'},  
 393: {id : '288', numeroid : '419'},  
 394: {id : '289', numeroid : '420'},  
 395: {id : '290', numeroid : '421'},  
 396: {id : '291', numeroid : '424'},  
 397: {id : '292', numeroid : '425'},  
 398: {id : '293', numeroid : '426'},  
 399: {id : '294', numeroid : '427'},  
 400: {id : '295', numeroid : '428'},  
 401: {id : '296', numeroid : '429'},  
 402: {id : '297', numeroid : '430'},  
 403: {id : '298', numeroid : '431'},  
 404: {id : '299', numeroid : '432'},  
 405: {id : '300', numeroid : '433'},  
 406: {id : '301', numeroid : '434'},  
 407: {id : '302', numeroid : '435'},  
 408: {id : '303', numeroid : '436'},  
 409: {id : '304', numeroid : '437'},  
 410: {id : '305', numeroid : '438'},  
 411: {id : '306', numeroid : '439'},  
 412: {id : '307', numeroid : '440'},  
 413: {id : '308', numeroid : '441'},  
 414: {id : '309', numeroid : '442'},  
 415: {id : '310', numeroid : '443'},  
 416: {id : '311', numeroid : '444'},  
 417: {id : '312', numeroid : '445'},  
 418: {id : '313', numeroid : '446'},  
 419: {id : '314', numeroid : '447'},  
 420: {id : '315', numeroid : '448'},  
 421: {id : '316', numeroid : '449'},  
 422: {id : '317', numeroid : '450'},  
 423: {id : '318', numeroid : '451'},  
 424: {id : '319', numeroid : '452'},  
 425: {id : '320', numeroid : '453'},  
 426: {id : '321', numeroid : '454'},  
 427: {id : '322', numeroid : '455'},  
 428: {id : '323', numeroid : '456'},  
 429: {id : '324', numeroid : '457'},  
 430: {id : '325', numeroid : '458'},  
 431: {id : '326', numeroid : '459'},  
 432: {id : '327', numeroid : '460'},  
 433: {id : '328', numeroid : '461'},  
 434: {id : '329', numeroid : '462'},  
 435: {id : '330', numeroid : '463'},  
 436: {id : '331', numeroid : '464'},  
 437: {id : '332', numeroid : '465'},  
 438: {id : '333', numeroid : '466'},  
 439: {id : '334', numeroid : '467'},  
 440: {id : '335', numeroid : '468'},  
 441: {id : '336', numeroid : '469'},  
 442: {id : '337', numeroid : '470'},  
 443: {id : '338', numeroid : '471'},  
 444: {id : '339', numeroid : '472'},  
 445: {id : '340', numeroid : '473'},  
 446: {id : '341', numeroid : '474'},  
 447: {id : '342', numeroid : '475'},  
 448: {id : '343', numeroid : '476'},  
 449: {id : '344', numeroid : '477'},  
 450: {id : '345', numeroid : '478'},  
 451: {id : '346', numeroid : '479'},  
 452: {id : '347', numeroid : '480'},  
 453: {id : '348', numeroid : '481'},  
 454: {id : '349', numeroid : '482'},  
 455: {id : '350', numeroid : '483'},  
 456: {id : '351', numeroid : '484'},  
 457: {id : '352', numeroid : '485'},  
 458: {id : '353', numeroid : '486'},  
 459: {id : '354', numeroid : '487'},  
 460: {id : '355', numeroid : '488'},  
 461: {id : '356', numeroid : '489'},  
 462: {id : '357', numeroid : '490'},  
 463: {id : '358', numeroid : '491'},  
 464: {id : '359', numeroid : '492'},  
 465: {id : '360', numeroid : '493'},  
 466: {id : '361', numeroid : '494'},  
 467: {id : '362', numeroid : '495'},  
 468: {id : '363', numeroid : '497'},  
 469: {id : '364', numeroid : '498'},  
 470: {id : '365', numeroid : '499'},  
 471: {id : '366', numeroid : '501'},  
 472: {id : '367', numeroid : '1001'},  
 473: {id : '368', numeroid : '1002'},  
 474: {id : '369', numeroid : '1003'},  
 475: {id : '370', numeroid : '1005'},  
 476: {id : '371', numeroid : '1006'},  
 477: {id : '372', numeroid : '1008'},  
 478: {id : '373', numeroid : '1009'},  
 479: {id : '374', numeroid : '1010'},  
 480: {id : '375', numeroid : '1011'},  
 481: {id : '376', numeroid : '1012'},  
 482: {id : '377', numeroid : '1013'},  
 483: {id : '378', numeroid : '1014'},  
 484: {id : '379', numeroid : '1015'},  
 485: {id : '380', numeroid : '1016'},  
 486: {id : '381', numeroid : '1017'},  
 487: {id : '382', numeroid : '1018'},  
 488: {id : '383', numeroid : '1020'},  
 489: {id : '384', numeroid : '1021'},  
 490: {id : '385', numeroid : '1022'},  
 491: {id : '386', numeroid : '1023'},  
 492: {id : '387', numeroid : '1026'},  
 493: {id : '388', numeroid : '1027'},  
 494: {id : '389', numeroid : '1028'},  
 495: {id : '390', numeroid : '1029'},  
 496: {id : '391', numeroid : '1030'},  
 497: {id : '392', numeroid : '1032'},  
 498: {id : '393', numeroid : '1033'},  
 499: {id : '394', numeroid : '1035'},  
 500: {id : '395', numeroid : '1036'},  
 501: {id : '396', numeroid : '1037'},  
 502: {id : '397', numeroid : '1038'},  
 503: {id : '398', numeroid : '1039'},  
 504: {id : '399', numeroid : '1040'},  
 505: {id : '400', numeroid : '1041'},  
 506: {id : '401', numeroid : '1043'},  
 507: {id : '402', numeroid : '1044'},  
 508: {id : '403', numeroid : '1045'},  
 509: {id : '404', numeroid : '1046'},  
 510: {id : '405', numeroid : '1048'},  
 511: {id : '406', numeroid : '1049'},  
 512: {id : '407', numeroid : '1050'},  
 513: {id : '408', numeroid : '1051'},  
 514: {id : '409', numeroid : '1052'},  
 515: {id : '410', numeroid : '1053'},  
 516: {id : '411', numeroid : '1055'},  
 517: {id : '412', numeroid : '1056'},  
 518: {id : '413', numeroid : '1057'},  
 519: {id : '414', numeroid : '1058'},  
 520: {id : '415', numeroid : '1060'},  
 521: {id : '416', numeroid : '1061'},  
 522: {id : '417', numeroid : '1062'},  
 523: {id : '418', numeroid : '1063'},  
 524: {id : '419', numeroid : '1064'},  
 525: {id : '420', numeroid : '1065'},  
 526: {id : '421', numeroid : '1066'},  
 527: {id : '422', numeroid : '1067'},  
 528: {id : '423', numeroid : '1069'},  
 529: {id : '424', numeroid : '1070'},  
 530: {id : '425', numeroid : '1071'},  
 531: {id : '426', numeroid : '1072'},  
 532: {id : '427', numeroid : '1073'},  
 533: {id : '428', numeroid : '1075'},  
 534: {id : '429', numeroid : '1076'},  
 535: {id : '430', numeroid : '1077'},  
 536: {id : '431', numeroid : '1078'},  
 537: {id : '432', numeroid : '1079'},  
 538: {id : '433', numeroid : '1080'},  
 539: {id : '434', numeroid : '1081'},  
 540: {id : '435', numeroid : '1082'},  
 541: {id : '436', numeroid : '1083'},  
 542: {id : '437', numeroid : '1085'},  
 543: {id : '438', numeroid : '1086'},  
 544: {id : '439', numeroid : '1087'},  
 545: {id : '440', numeroid : '1088'},  
 546: {id : '441', numeroid : '1089'},  
 547: {id : '442', numeroid : '1090'},  
 548: {id : '443', numeroid : '1091'},  
 549: {id : '444', numeroid : '1092'},  
 550: {id : '445', numeroid : '1093'},  
 551: {id : '446', numeroid : '1098'},  
 552: {id : '447', numeroid : '1099'},  
 553: {id : '448', numeroid : '1100'},  
 554: {id : '449', numeroid : '1104'},  
 555: {id : '450', numeroid : '1105'},  
 556: {id : '451', numeroid : '1106'},  
 557: {id : '452', numeroid : '1107'},  
 558: {id : '453', numeroid : '1108'},  
 559: {id : '454', numeroid : '1109'},  
 560: {id : '455', numeroid : '1110'},  
 561: {id : '456', numeroid : '1111'},  
 562: {id : '457', numeroid : '1112'},  
 563: {id : '458', numeroid : '1113'},  
 564: {id : '459', numeroid : '1114'},  
 565: {id : '460', numeroid : '1118'},  
 566: {id : '461', numeroid : '1120'},  
 567: {id : '462', numeroid : '1123'},  
 568: {id : '463', numeroid : '1125'},  
 569: {id : '464', numeroid : '1126'},  
 570: {id : '465', numeroid : '1127'},  
 571: {id : '466', numeroid : '1128'},  
 572: {id : '467', numeroid : '1129'},  
 573: {id : '468', numeroid : '1130'},  
 574: {id : '469', numeroid : '1131'},  
 575: {id : '470', numeroid : '1133'},  
 576: {id : '471', numeroid : '1134'},  
 577: {id : '472', numeroid : '1135'},  
 578: {id : '473', numeroid : '1136'},  
 579: {id : '474', numeroid : '1139'},  
 580: {id : '475', numeroid : '1143'},  
 581: {id : '476', numeroid : '1144'},  
 582: {id : '477', numeroid : '1145'},  
 583: {id : '478', numeroid : '1146'},  
 584: {id : '479', numeroid : '1147'},  
 585: {id : '480', numeroid : '1148'},  
 586: {id : '481', numeroid : '1149'},  
 587: {id : '482', numeroid : '1150'},  
 588: {id : '483', numeroid : '1152'},  
 589: {id : '484', numeroid : '1153'},  
 590: {id : '485', numeroid : '1154'},  
 591: {id : '486', numeroid : '1155'},  
 592: {id : '487', numeroid : '1156'},  
 593: {id : '488', numeroid : '1157'},  
 594: {id : '489', numeroid : '1158'},  
 595: {id : '490', numeroid : '1159'},  
 596: {id : '491', numeroid : '1160'},  
 597: {id : '492', numeroid : '1161'},  
 598: {id : '493', numeroid : '1162'},  
 599: {id : '494', numeroid : '1163'},  
 600: {id : '495', numeroid : '1164'},  
 601: {id : '496', numeroid : '1165'},  
 602: {id : '497', numeroid : '1166'},  
 603: {id : '498', numeroid : '1167'},  
 604: {id : '499', numeroid : '1169'},  
 605: {id : '500', numeroid : '1170'},  
 606: {id : '501', numeroid : '1171'},  
 607: {id : '502', numeroid : '1172'},  
 608: {id : '503', numeroid : '1173'},  
 609: {id : '504', numeroid : '1175'},  
 610: {id : '505', numeroid : '1176'},  
 611: {id : '506', numeroid : '1177'},  
 612: {id : '507', numeroid : '1178'},  
 613: {id : '508', numeroid : '1179'},  
 614: {id : '509', numeroid : '1180'},  
 615: {id : '510', numeroid : '1181'},  
 616: {id : '511', numeroid : '1182'},  
 617: {id : '512', numeroid : '1183'},  
 618: {id : '513', numeroid : '1184'},  
 619: {id : '514', numeroid : '1185'},  
 620: {id : '515', numeroid : '1188'},  
 621: {id : '516', numeroid : '1189'},  
 622: {id : '517', numeroid : '1190'},  
 623: {id : '518', numeroid : '1191'},  
 624: {id : '519', numeroid : '1192'},  
 625: {id : '520', numeroid : '1193'},  
 626: {id : '521', numeroid : '1194'},  
 627: {id : '522', numeroid : '1195'},  
 628: {id : '523', numeroid : '1196'},  
 629: {id : '524', numeroid : '1197'},  
 630: {id : '525', numeroid : '1198'},  
 631: {id : '526', numeroid : '1199'},  
 632: {id : '527', numeroid : '1201'},  
 633: {id : '528', numeroid : '1202'},  
 634: {id : '529', numeroid : '1203'},  
 635: {id : '530', numeroid : '1204'},  
 636: {id : '531', numeroid : '1206'},  
 637: {id : '532', numeroid : '1207'},  
 638: {id : '533', numeroid : '1208'},  
 639: {id : '534', numeroid : '1210'},  
 640: {id : '535', numeroid : '1212'},  
 641: {id : '536', numeroid : '1213'},  
 642: {id : '537', numeroid : '1214'},  
 643: {id : '538', numeroid : '1216'},  
 644: {id : '539', numeroid : '1218'},  
 645: {id : '540', numeroid : '1219'},  
 646: {id : '541', numeroid : '1220'},  
 647: {id : '542', numeroid : '1221'},  
 648: {id : '543', numeroid : '1222'},  
 649: {id : '544', numeroid : '1223'},  
 650: {id : '545', numeroid : '1224'},  
 651: {id : '546', numeroid : '1228'},  
 652: {id : '547', numeroid : '1229'},  
 653: {id : '548', numeroid : '1230'},  
 654: {id : '549', numeroid : '1231'},  
 655: {id : '550', numeroid : '1233'},  
 656: {id : '551', numeroid : '1234'},  
 657: {id : '552', numeroid : '1235'},  
 658: {id : '553', numeroid : '1237'},  
 659: {id : '554', numeroid : '1238'},  
 660: {id : '555', numeroid : '1239'},  
 661: {id : '556', numeroid : '1242'},  
 662: {id : '557', numeroid : '1243'},  
 663: {id : '558', numeroid : '1244'},  
 664: {id : '559', numeroid : '1245'},  
 665: {id : '560', numeroid : '1246'},  
 666: {id : '561', numeroid : '1247'},  
 667: {id : '562', numeroid : '1248'},  
 668: {id : '563', numeroid : '1249'},  
 669: {id : '564', numeroid : '1250'},  
 670: {id : '565', numeroid : '1251'},  
 671: {id : '566', numeroid : '1255'},  
 672: {id : '567', numeroid : '1256'},  
 673: {id : '568', numeroid : '1257'},  
 674: {id : '569', numeroid : '1259'},  
 675: {id : '570', numeroid : '1261'},  
 676: {id : '571', numeroid : '1262'},  
 677: {id : '572', numeroid : '1263'},  
 678: {id : '573', numeroid : '1264'},  
 679: {id : '574', numeroid : '1265'},  
 680: {id : '575', numeroid : '1266'},  
 681: {id : '576', numeroid : '1267'},  
 682: {id : '577', numeroid : '1268'},  
 683: {id : '578', numeroid : '1270'},  
 684: {id : '579', numeroid : '1271'},  
 685: {id : '580', numeroid : '1272'},  
 686: {id : '581', numeroid : '1274'},  
 687: {id : '582', numeroid : '1275'},  
 688: {id : '583', numeroid : '1276'},  
 689: {id : '584', numeroid : '1277'},  
 690: {id : '585', numeroid : '1278'},  
 691: {id : '586', numeroid : '1279'},  
 692: {id : '587', numeroid : '1280'},  
 693: {id : '588', numeroid : '1281'},  
 694: {id : '589', numeroid : '1282'},  
 695: {id : '590', numeroid : '1286'},  
 696: {id : '591', numeroid : '1287'},  
 697: {id : '592', numeroid : '1288'},  
 698: {id : '593', numeroid : '1289'},  
 699: {id : '594', numeroid : '1292'},  
 700: {id : '595', numeroid : '1293'},  
 701: {id : '596', numeroid : '1294'},  
 702: {id : '597', numeroid : '1295'},  
 703: {id : '598', numeroid : '1296'},  
 704: {id : '599', numeroid : '1297'},  
 705: {id : '600', numeroid : '1298'},  
 706: {id : '601', numeroid : '1299'},  
 707: {id : '602', numeroid : '1300'},  
 708: {id : '603', numeroid : '1301'},  
 709: {id : '604', numeroid : '1302'},  
 710: {id : '605', numeroid : '1303'},  
 711: {id : '606', numeroid : '1304'},  
 712: {id : '607', numeroid : '1305'},  
 713: {id : '608', numeroid : '1306'},  
 714: {id : '609', numeroid : '1307'},  
 715: {id : '610', numeroid : '1308'},  
 716: {id : '611', numeroid : '1309'},  
 717: {id : '612', numeroid : '1310'},  
 718: {id : '613', numeroid : '1312'},  
 719: {id : '614', numeroid : '1313'},  
 720: {id : '615', numeroid : '1314'},  
 721: {id : '616', numeroid : '1318'},  
 722: {id : '617', numeroid : '1320'},  
 723: {id : '618', numeroid : '1321'},  
 724: {id : '619', numeroid : '1322'},  
 725: {id : '620', numeroid : '1323'},  
 726: {id : '621', numeroid : '1324'},  
 727: {id : '622', numeroid : '1325'},  
 728: {id : '623', numeroid : '1326'},  
 729: {id : '624', numeroid : '1327'},  
 730: {id : '625', numeroid : '1328'},  
 731: {id : '626', numeroid : '1330'},  
 732: {id : '627', numeroid : '1331'},  
 733: {id : '628', numeroid : '1332'},  
 734: {id : '629', numeroid : '1333'},  
 735: {id : '630', numeroid : '1334'},  
 736: {id : '631', numeroid : '1336'},  
 737: {id : '632', numeroid : '1337'},  
 738: {id : '633', numeroid : '1338'},  
 739: {id : '634', numeroid : '1339'},  
 740: {id : '635', numeroid : '1340'},  
 741: {id : '636', numeroid : '1341'},  
 742: {id : '637', numeroid : '1343'},  
 743: {id : '638', numeroid : '1344'},  
 744: {id : '639', numeroid : '1345'},  
 745: {id : '640', numeroid : '1346'},  
 746: {id : '641', numeroid : '1347'},  
 747: {id : '642', numeroid : '1348'},  
 748: {id : '643', numeroid : '1349'},  
 749: {id : '644', numeroid : '1350'},  
 750: {id : '645', numeroid : '1352'},  
 751: {id : '646', numeroid : '1353'},  
 752: {id : '647', numeroid : '1354'},  
 753: {id : '648', numeroid : '1355'},  
 754: {id : '649', numeroid : '1356'},  
 755: {id : '650', numeroid : '1357'},  
 756: {id : '651', numeroid : '1358'},  
 757: {id : '652', numeroid : '1360'},  
 758: {id : '653', numeroid : '1361'},  
 759: {id : '654', numeroid : '1362'},  
 760: {id : '655', numeroid : '1363'},  
 761: {id : '656', numeroid : '1364'},  
 762: {id : '657', numeroid : '1365'},  
 763: {id : '658', numeroid : '1366'},  
 764: {id : '659', numeroid : '1369'},  
 765: {id : '660', numeroid : '1370'},  
 766: {id : '661', numeroid : '1373'},  
 767: {id : '662', numeroid : '1374'},  
 768: {id : '663', numeroid : '1376'},  
 769: {id : '664', numeroid : '1378'},  
 770: {id : '665', numeroid : '1379'},  
 771: {id : '666', numeroid : '1380'},  
 772: {id : '667', numeroid : '1381'},  
 773: {id : '668', numeroid : '1382'},  
 774: {id : '669', numeroid : '1383'},  
 775: {id : '670', numeroid : '1384'},  
 776: {id : '671', numeroid : '1385'},  
 777: {id : '672', numeroid : '1386'},  
 778: {id : '673', numeroid : '1389'},  
 779: {id : '674', numeroid : '1390'},  
 780: {id : '675', numeroid : '1391'},  
 781: {id : '676', numeroid : '1392'},  
 782: {id : '677', numeroid : '1393'},  
 783: {id : '678', numeroid : '1394'},  
 784: {id : '679', numeroid : '1395'},  
 785: {id : '680', numeroid : '1396'},  
 786: {id : '681', numeroid : '1397'},  
 787: {id : '682', numeroid : '1398'},  
 788: {id : '683', numeroid : '1400'},  
 789: {id : '684', numeroid : '1401'},  
 790: {id : '685', numeroid : '1402'},  
 791: {id : '686', numeroid : '1403'},  
 792: {id : '687', numeroid : '1404'},  
 793: {id : '688', numeroid : '1405'},  
 794: {id : '689', numeroid : '1407'},  
 795: {id : '690', numeroid : '1408'},  
 796: {id : '691', numeroid : '1409'},  
 797: {id : '692', numeroid : '1410'},  
 798: {id : '693', numeroid : '1411'},  
 799: {id : '694', numeroid : '1413'},  
 800: {id : '695', numeroid : '1414'},  
 801: {id : '696', numeroid : '1415'},  
 802: {id : '697', numeroid : '1417'},  
 803: {id : '698', numeroid : '1418'},  
 804: {id : '699', numeroid : '1419'},  
 805: {id : '700', numeroid : '1420'},  
 806: {id : '701', numeroid : '1421'},  
 807: {id : '702', numeroid : '1422'},  
 808: {id : '703', numeroid : '1423'},  
 809: {id : '704', numeroid : '1426'},  
 810: {id : '705', numeroid : '1427'},  
 811: {id : '706', numeroid : '1428'},  
 812: {id : '707', numeroid : '1431'},  
 813: {id : '708', numeroid : '1432'},  
 814: {id : '709', numeroid : '1433'},  
 815: {id : '710', numeroid : '1435'},  
 816: {id : '711', numeroid : '1436'},  
 817: {id : '712', numeroid : '1437'},  
 818: {id : '713', numeroid : '1438'},  
 819: {id : '714', numeroid : '1439'},  
 820: {id : '715', numeroid : '1442'},  
 821: {id : '716', numeroid : '1444'},  
 822: {id : '717', numeroid : '1445'},  
 823: {id : '718', numeroid : '1446'},  
 824: {id : '719', numeroid : '1447'},  
 825: {id : '720', numeroid : '1448'},  
 826: {id : '721', numeroid : '1449'},  
 827: {id : '722', numeroid : '1450'},  
 828: {id : '723', numeroid : '1451'},  
 829: {id : '724', numeroid : '1452'},  
 830: {id : '725', numeroid : '1453'},  
 831: {id : '726', numeroid : '1454'},  
 832: {id : '727', numeroid : '1455'},  
 833: {id : '728', numeroid : '1456'},  
 834: {id : '729', numeroid : '1457'},  
 835: {id : '730', numeroid : '1458'},  
 836: {id : '731', numeroid : '1459'},  
 837: {id : '732', numeroid : '1461'},  
 838: {id : '733', numeroid : '1462'},  
 839: {id : '734', numeroid : '1463'},  
 840: {id : '735', numeroid : '1465'},  
 841: {id : '736', numeroid : '1466'},  
 842: {id : '737', numeroid : '1467'},  
 843: {id : '738', numeroid : '1469'},  
 844: {id : '739', numeroid : '1470'},  
 845: {id : '740', numeroid : '1471'},  
 846: {id : '741', numeroid : '1472'},  
 847: {id : '742', numeroid : '1473'},  
 848: {id : '743', numeroid : '1474'},  
 849: {id : '744', numeroid : '1475'},  
 850: {id : '745', numeroid : '1476'},  
 851: {id : '746', numeroid : '1477'},  
 852: {id : '747', numeroid : '1479'},  
 853: {id : '748', numeroid : '1481'},  
 854: {id : '749', numeroid : '1482'},  
 855: {id : '750', numeroid : '1483'},  
 856: {id : '751', numeroid : '1484'},  
 857: {id : '752', numeroid : '1485'},  
 858: {id : '753', numeroid : '1486'},  
 859: {id : '754', numeroid : '1487'},  
 860: {id : '755', numeroid : '1488'},  
 861: {id : '756', numeroid : '1489'},  
 862: {id : '757', numeroid : '1490'},  
 863: {id : '758', numeroid : '1491'},  
 864: {id : '759', numeroid : '1492'},  
 865: {id : '760', numeroid : '1493'},  
 866: {id : '761', numeroid : '1494'},  
 867: {id : '762', numeroid : '1495'},  
 868: {id : '763', numeroid : '1496'},  
 869: {id : '764', numeroid : '1498'},  
 870: {id : '765', numeroid : '1499'},  
 871: {id : '766', numeroid : '1500'},  
 872: {id : '767', numeroid : '1502'},  
 873: {id : '768', numeroid : '1503'},  
 874: {id : '769', numeroid : '1504'},  
 875: {id : '770', numeroid : '1505'},  
 876: {id : '771', numeroid : '1506'},  
 877: {id : '772', numeroid : '1507'},  
 878: {id : '773', numeroid : '1508'},  
 879: {id : '774', numeroid : '1509'},  
 880: {id : '775', numeroid : '1510'},  
 881: {id : '776', numeroid : '1511'},  
 882: {id : '777', numeroid : '1512'},  
 883: {id : '778', numeroid : '1513'},  
 884: {id : '779', numeroid : '1514'},  
 885: {id : '780', numeroid : '1515'},  
 886: {id : '781', numeroid : '1516'},  
 887: {id : '782', numeroid : '1517'},  
 888: {id : '783', numeroid : '1541'},  
 889: {id : '784', numeroid : '1544'},  
 890: {id : '785', numeroid : '1545'},  
 891: {id : '786', numeroid : '1546'},  
 892: {id : '787', numeroid : '1547'},  
 893: {id : '788', numeroid : '1548'},  
 894: {id : '789', numeroid : '1549'},  
 895: {id : '790', numeroid : '1550'},  
 896: {id : '791', numeroid : '1551'},  
 897: {id : '792', numeroid : '1553'},  
 898: {id : '793', numeroid : '1554'},  
 899: {id : '794', numeroid : '1555'},  
 900: {id : '795', numeroid : '1556'},  
 901: {id : '796', numeroid : '1557'},  
 902: {id : '797', numeroid : '1558'},  
 903: {id : '798', numeroid : '1559'},  
 904: {id : '799', numeroid : '1560'},  
 905: {id : '800', numeroid : '1561'},  
 906: {id : '801', numeroid : '1562'},  
 907: {id : '802', numeroid : '1564'},  
 908: {id : '803', numeroid : '1565'},  
 909: {id : '804', numeroid : '1566'},  
 910: {id : '805', numeroid : '1567'},  
 911: {id : '806', numeroid : '1569'},  
 912: {id : '807', numeroid : '1570'},  
 913: {id : '808', numeroid : '1571'},  
 914: {id : '809', numeroid : '1572'},  
 915: {id : '810', numeroid : '1573'},  
 916: {id : '811', numeroid : '1574'},  
 917: {id : '812', numeroid : '1575'},  
 918: {id : '813', numeroid : '1577'},  
 919: {id : '814', numeroid : '1578'},  
 920: {id : '815', numeroid : '1579'},  
 921: {id : '816', numeroid : '1580'},  
 922: {id : '817', numeroid : '1581'},  
 923: {id : '818', numeroid : '1582'},  
 924: {id : '819', numeroid : '1583'},  
 925: {id : '820', numeroid : '1585'},  
 926: {id : '821', numeroid : '1586'},  
 927: {id : '822', numeroid : '1587'},  
 928: {id : '823', numeroid : '1588'},  
 929: {id : '824', numeroid : '1589'},  
 930: {id : '825', numeroid : '1590'},  
 931: {id : '826', numeroid : '1591'},  
 932: {id : '827', numeroid : '1593'},  
 933: {id : '828', numeroid : '1594'},  
 934: {id : '829', numeroid : '1595'},  
 935: {id : '830', numeroid : '1596'},  
 936: {id : '831', numeroid : '1597'},  
 937: {id : '832', numeroid : '1598'},  
 938: {id : '833', numeroid : '1599'},  
 939: {id : '834', numeroid : '1600'},  
 940: {id : '835', numeroid : '1601'},  
 941: {id : '836', numeroid : '1602'},  
 942: {id : '837', numeroid : '1603'},  
 943: {id : '838', numeroid : '1604'},  
 944: {id : '839', numeroid : '1605'},  
 945: {id : '840', numeroid : '1606'},  
 946: {id : '841', numeroid : '1607'},  
 947: {id : '842', numeroid : '1608'},  
 948: {id : '843', numeroid : '1610'},  
 949: {id : '844', numeroid : '1611'},  
 950: {id : '845', numeroid : '1612'},  
 951: {id : '846', numeroid : '1613'},  
 952: {id : '847', numeroid : '1614'},  
 953: {id : '848', numeroid : '1616'},  
 954: {id : '849', numeroid : '1617'},  
 955: {id : '850', numeroid : '1618'},  
 956: {id : '851', numeroid : '1620'},  
 957: {id : '852', numeroid : '1621'},  
 958: {id : '853', numeroid : '1622'},  
 959: {id : '854', numeroid : '1623'},  
 960: {id : '855', numeroid : '1624'},  
 961: {id : '856', numeroid : '1625'},  
 962: {id : '857', numeroid : '1626'},  
 963: {id : '858', numeroid : '1627'},  
 964: {id : '859', numeroid : '1629'},  
 965: {id : '860', numeroid : '1630'},  
 966: {id : '861', numeroid : '1631'},  
 967: {id : '862', numeroid : '1634'},  
 968: {id : '863', numeroid : '1636'},  
 969: {id : '864', numeroid : '1637'},  
 970: {id : '865', numeroid : '1638'},  
 971: {id : '866', numeroid : '1639'},  
 972: {id : '867', numeroid : '1640'},  
 973: {id : '868', numeroid : '1641'},  
 974: {id : '869', numeroid : '1642'},  
 975: {id : '870', numeroid : '1643'},  
 976: {id : '871', numeroid : '1644'},  
 977: {id : '872', numeroid : '1645'},  
 978: {id : '873', numeroid : '1646'},  
 979: {id : '874', numeroid : '1647'},  
 980: {id : '875', numeroid : '1648'},  
 981: {id : '876', numeroid : '1649'},  
 982: {id : '877', numeroid : '1650'},  
 983: {id : '878', numeroid : '1651'},  
 984: {id : '879', numeroid : '1652'},  
 985: {id : '880', numeroid : '1653'},  
 986: {id : '881', numeroid : '1654'},  
 987: {id : '882', numeroid : '1655'},  
 988: {id : '883', numeroid : '1656'},  
 989: {id : '884', numeroid : '1657'},  
 990: {id : '885', numeroid : '1658'},  
 991: {id : '886', numeroid : '1659'},  
 992: {id : '887', numeroid : '1660'},  
 993: {id : '888', numeroid : '1661'},  
 994: {id : '889', numeroid : '1662'},  
 995: {id : '890', numeroid : '1663'},  
 996: {id : '891', numeroid : '1664'},  
 997: {id : '892', numeroid : '1665'},  
 998: {id : '893', numeroid : '1669'},  
 999: {id : '894', numeroid : '1670'},  
1000: {id : '895', numeroid : '1671'},  
1001: {id : '896', numeroid : '1672'},  
1002: {id : '897', numeroid : '1673'},  
1003: {id : '898', numeroid : '1674'},  
1004: {id : '899', numeroid : '1677'},  
1005: {id : '900', numeroid : '1678'},  
1006: {id : '901', numeroid : '1679'},  
1007: {id : '902', numeroid : '1680'},  
1008: {id : '903', numeroid : '1683'},  
1009: {id : '904', numeroid : '1684'},  
1010: {id : '905', numeroid : '1685'},  
1011: {id : '906', numeroid : '1686'},  
1012: {id : '907', numeroid : '1687'},  
1013: {id : '908', numeroid : '1688'},  
1014: {id : '909', numeroid : '1689'},  
1015: {id : '910', numeroid : '1690'},  
1016: {id : '911', numeroid : '1691'},  
1017: {id : '912', numeroid : '1692'},  
1018: {id : '913', numeroid : '1693'},  
1019: {id : '914', numeroid : '1694'},  
1020: {id : '915', numeroid : '1695'},  
1021: {id : '916', numeroid : '1697'},  
1022: {id : '917', numeroid : '1698'},  
1023: {id : '918', numeroid : '1699'},  
1024: {id : '919', numeroid : '1700'},  
1025: {id : '920', numeroid : '1701'},  
1026: {id : '921', numeroid : '1702'},  
1027: {id : '922', numeroid : '1703'},  
1028: {id : '923', numeroid : '1704'},  
1029: {id : '924', numeroid : '1705'},  
1030: {id : '925', numeroid : '1707'},  
1031: {id : '926', numeroid : '1708'},  
1032: {id : '927', numeroid : '1709'},  
1033: {id : '928', numeroid : '1710'},  
1034: {id : '929', numeroid : '1711'},  
1035: {id : '930', numeroid : '1712'},  
1036: {id : '931', numeroid : '1713'},  
1037: {id : '932', numeroid : '1714'},  
1038: {id : '933', numeroid : '1715'},  
1039: {id : '934', numeroid : '1716'},  
1040: {id : '935', numeroid : '1717'},  
1041: {id : '936', numeroid : '1718'},  
1042: {id : '937', numeroid : '1719'},  
1043: {id : '938', numeroid : '1722'},  
1044: {id : '939', numeroid : '1723'},  
1045: {id : '940', numeroid : '1724'},  
1046: {id : '941', numeroid : '1725'},  
1047: {id : '942', numeroid : '1726'},  
1048: {id : '943', numeroid : '1727'},  
1049: {id : '944', numeroid : '1728'},  
1050: {id : '945', numeroid : '1729'},  
1051: {id : '946', numeroid : '1730'},  
1052: {id : '947', numeroid : '1731'},  
1053: {id : '948', numeroid : '1732'},  
1054: {id : '949', numeroid : '1733'},  
1055: {id : '950', numeroid : '1736'},  
1056: {id : '951', numeroid : '1737'},  
1057: {id : '952', numeroid : '1738'},  
1058: {id : '953', numeroid : '1739'},  
1059: {id : '954', numeroid : '1740'},  
1060: {id : '955', numeroid : '1741'},  
1061: {id : '956', numeroid : '1742'},  
1062: {id : '957', numeroid : '1743'},  
1063: {id : '958', numeroid : '1744'},  
1064: {id : '959', numeroid : '1745'},  
1065: {id : '960', numeroid : '1746'},  
1066: {id : '961', numeroid : '1747'},  
1067: {id : '962', numeroid : '1748'},  
1068: {id : '963', numeroid : '1749'},  
1069: {id : '964', numeroid : '1750'},  
1070: {id : '965', numeroid : '1751'},  
1071: {id : '966', numeroid : '1752'},  
1072: {id : '967', numeroid : '1753'},  
1073: {id : '968', numeroid : '1754'},  
1074: {id : '969', numeroid : '1755'},  
1075: {id : '970', numeroid : '1756'},  
1076: {id : '971', numeroid : '1757'},  
1077: {id : '972', numeroid : '1758'},  
1078: {id : '973', numeroid : '1759'},  
1079: {id : '974', numeroid : '1760'},  
1080: {id : '975', numeroid : '1761'},  
1081: {id : '976', numeroid : '1762'},  
1082: {id : '977', numeroid : '1763'},  
1083: {id : '978', numeroid : '1764'},  
1084: {id : '979', numeroid : '1765'},  
1085: {id : '980', numeroid : '1766'},  
1086: {id : '981', numeroid : '1767'},  
1087: {id : '982', numeroid : '1768'},  
1088: {id : '983', numeroid : '1769'},  
1089: {id : '984', numeroid : '1770'},  
1090: {id : '985', numeroid : '1771'},  
1091: {id : '986', numeroid : '1773'},  
1092: {id : '987', numeroid : '1774'},  
1093: {id : '988', numeroid : '1775'},  
1094: {id : '989', numeroid : '1776'},  
1095: {id : '990', numeroid : '1777'},  
1096: {id : '991', numeroid : '1778'},  
1097: {id : '992', numeroid : '1779'},  
1098: {id : '993', numeroid : '1780'},  
1099: {id : '994', numeroid : '1781'},  
1100: {id : '995', numeroid : '1782'},  
1101: {id : '996', numeroid : '1783'},  
1102: {id : '997', numeroid : '1784'},  
1103: {id : '998', numeroid : '1786'},  
1104: {id : '999', numeroid : '1787'},  
1105: {id : '1000', numeroid : '1788'},  
1106: {id : '1001', numeroid : '1789'},  
1107: {id : '1002', numeroid : '1790'},  
1108: {id : '1003', numeroid : '1791'},  
1109: {id : '1004', numeroid : '1792'},  
1110: {id : '1005', numeroid : '1793'},  
1111: {id : '1006', numeroid : '1794'},  
1112: {id : '1007', numeroid : '1796'},  
1113: {id : '1008', numeroid : '1798'},  
1114: {id : '1009', numeroid : '1799'},  
1115: {id : '1010', numeroid : '1800'},  
1116: {id : '1011', numeroid : '1801'},  
1117: {id : '1012', numeroid : '1802'},  
1118: {id : '1013', numeroid : '1803'},  
1119: {id : '1014', numeroid : '1804'},  
1120: {id : '1015', numeroid : '1805'},  
1121: {id : '1016', numeroid : '1806'},  
1122: {id : '1017', numeroid : '1807'},  
1123: {id : '1018', numeroid : '1808'},  
1124: {id : '1019', numeroid : '1809'},  
1125: {id : '1020', numeroid : '1810'},  
1126: {id : '1021', numeroid : '1811'},  
1127: {id : '1022', numeroid : '1812'},  
1128: {id : '1023', numeroid : '1813'},  
1129: {id : '1024', numeroid : '1814'},  
1130: {id : '1025', numeroid : '1815'},  
1131: {id : '1026', numeroid : '1816'},  
1132: {id : '1027', numeroid : '1817'},  
1133: {id : '1028', numeroid : '1818'},  
1134: {id : '1029', numeroid : '1819'},  
1135: {id : '1030', numeroid : '1823'},  
1136: {id : '1031', numeroid : '1824'},  
1137: {id : '1032', numeroid : '1825'},  
1138: {id : '1033', numeroid : '1826'},  
1139: {id : '1034', numeroid : '1827'},  
1140: {id : '1035', numeroid : '1828'},  
1141: {id : '1036', numeroid : '1829'},  
1142: {id : '1037', numeroid : '1830'},  
1143: {id : '1038', numeroid : '1831'},  
1144: {id : '1039', numeroid : '1832'},  
1145: {id : '1040', numeroid : '1833'},  
1146: {id : '1041', numeroid : '1834'},  
1147: {id : '1042', numeroid : '1835'},  
1148: {id : '1043', numeroid : '1836'},  
1149: {id : '1044', numeroid : '1837'},  
1150: {id : '1045', numeroid : '1838'},  
1151: {id : '1046', numeroid : '1839'},  
1152: {id : '1047', numeroid : '1840'},  
1153: {id : '1048', numeroid : '1841'},  
1154: {id : '1049', numeroid : '1843'},  
1155: {id : '1050', numeroid : '1845'},  
1156: {id : '1051', numeroid : '1846'},  
1157: {id : '1052', numeroid : '1847'},  
1158: {id : '1053', numeroid : '1848'},  
1159: {id : '1054', numeroid : '1849'},  
1160: {id : '1055', numeroid : '1851'},  
1161: {id : '1056', numeroid : '1854'},  
1162: {id : '1057', numeroid : '1855'},  
1163: {id : '1058', numeroid : '1858'},  
1164: {id : '1059', numeroid : '1859'},  
1165: {id : '1060', numeroid : '1860'},  
1166: {id : '1061', numeroid : '1862'},  
1167: {id : '1062', numeroid : '1863'},  
1168: {id : '1063', numeroid : '1864'},  
1169: {id : '1064', numeroid : '1865'},  
1170: {id : '1065', numeroid : '1866'},  
1171: {id : '1066', numeroid : '1868'},  
1172: {id : '1067', numeroid : '1869'},  
1173: {id : '1068', numeroid : '1870'},  
1174: {id : '1069', numeroid : '1871'},  
1175: {id : '1070', numeroid : '1872'},  
1176: {id : '1071', numeroid : '1873'},  
1177: {id : '1072', numeroid : '1884'},  
1178: {id : '1073', numeroid : '1885'},  
1179: {id : '1074', numeroid : '1886'},  
1180: {id : '1075', numeroid : '1887'},  
1181: {id : '1076', numeroid : '1888'},  
1182: {id : '1077', numeroid : '1889'},  
1183: {id : '1078', numeroid : '1891'},  
1184: {id : '1079', numeroid : '1892'},  
1185: {id : '1080', numeroid : '1894'},  
1186: {id : '1081', numeroid : '1895'},  
1187: {id : '1082', numeroid : '1897'},  
1188: {id : '1083', numeroid : '1898'},  
1189: {id : '1084', numeroid : '1902'},  
1190: {id : '1085', numeroid : '1903'},  
1191: {id : '1086', numeroid : '1905'},  
1192: {id : '1087', numeroid : '1906'},  
1193: {id : '1088', numeroid : '1907'},  
1194: {id : '1089', numeroid : '1908'},  
1195: {id : '1090', numeroid : '1910'},  
1196: {id : '1091', numeroid : '1911'},  
1197: {id : '1092', numeroid : '1912'},  
1198: {id : '1093', numeroid : '1913'},  
1199: {id : '1094', numeroid : '1914'},  
1200: {id : '1095', numeroid : '1915'},  
1201: {id : '1096', numeroid : '1916'},  
1202: {id : '1097', numeroid : '1917'},  
1203: {id : '1098', numeroid : '1918'},  
1204: {id : '1099', numeroid : '1919'},  
1205: {id : '1100', numeroid : '1920'},  
1206: {id : '1101', numeroid : '1921'},  
1207: {id : '1102', numeroid : '1922'},  
1208: {id : '1103', numeroid : '1923'},  
1209: {id : '1104', numeroid : '1928'},  
1210: {id : '1105', numeroid : '1929'},  
1211: {id : '1106', numeroid : '1931'},  
1212: {id : '1107', numeroid : '1932'},  
1213: {id : '1108', numeroid : '1935'},  
1214: {id : '1109', numeroid : '1938'},  
1215: {id : '1110', numeroid : '1939'},  
1216: {id : '1111', numeroid : '1940'},  
1217: {id : '1112', numeroid : '1941'},  
1218: {id : '1113', numeroid : '1942'},  
1219: {id : '1114', numeroid : '1944'},  
1220: {id : '1115', numeroid : '1945'},  
1221: {id : '1116', numeroid : '1950'},  
1222: {id : '1117', numeroid : '1951'},  
1223: {id : '1118', numeroid : '1952'},  
1224: {id : '1119', numeroid : '1953'},  
1225: {id : '1120', numeroid : '1954'},  
1226: {id : '1121', numeroid : '1955'},  
1227: {id : '1122', numeroid : '1956'},  
1228: {id : '1123', numeroid : '1957'},  
1229: {id : '1124', numeroid : '1958'},  
1230: {id : '1125', numeroid : '1959'},  
1231: {id : '1126', numeroid : '1960'},  
1232: {id : '1127', numeroid : '1961'},  
1233: {id : '1128', numeroid : '1962'},  
1234: {id : '1129', numeroid : '1963'},  
1235: {id : '1130', numeroid : '1964'},  
1236: {id : '1131', numeroid : '1965'},  
1237: {id : '1132', numeroid : '1966'},  
1238: {id : '1133', numeroid : '1967'},  
1239: {id : '1134', numeroid : '1968'},  
1240: {id : '1135', numeroid : '1969'},  
1241: {id : '1136', numeroid : '1970'},  
1242: {id : '1137', numeroid : '1971'},  
1243: {id : '1138', numeroid : '1972'},  
1244: {id : '1139', numeroid : '1973'},  
1245: {id : '1140', numeroid : '1974'},  
1246: {id : '1141', numeroid : '1975'},  
1247: {id : '1142', numeroid : '1976'},  
1248: {id : '1143', numeroid : '1977'},  
1249: {id : '1144', numeroid : '1978'},  
1250: {id : '1145', numeroid : '1979'},  
1251: {id : '1146', numeroid : '1980'},  
1252: {id : '1147', numeroid : '1981'},  
1253: {id : '1148', numeroid : '1982'},  
1254: {id : '1149', numeroid : '1983'},  
1255: {id : '1150', numeroid : '1984'},  
1256: {id : '1151', numeroid : '1986'},  
1257: {id : '1152', numeroid : '1987'},  
1258: {id : '1153', numeroid : '1988'},  
1259: {id : '1154', numeroid : '1989'},  
1260: {id : '1155', numeroid : '1991'},  
1261: {id : '1156', numeroid : '1992'},  
1262: {id : '1157', numeroid : '1993'},  
1263: {id : '1158', numeroid : '1994'},  
1264: {id : '1159', numeroid : '1999'},  
1265: {id : '1160', numeroid : '2000'},  
1266: {id : '1161', numeroid : '2001'},  
1267: {id : '1162', numeroid : '2002'},  
1268: {id : '1163', numeroid : '2003'},  
1269: {id : '1164', numeroid : '2004'},  
1270: {id : '1165', numeroid : '2005'},  
1271: {id : '1166', numeroid : '2006'},  
1272: {id : '1167', numeroid : '2008'},  
1273: {id : '1168', numeroid : '2009'},  
1274: {id : '1169', numeroid : '2010'},  
1275: {id : '1170', numeroid : '2011'},  
1276: {id : '1171', numeroid : '2012'},  
1277: {id : '1172', numeroid : '2013'},  
1278: {id : '1173', numeroid : '2014'},  
1279: {id : '1174', numeroid : '2015'},  
1280: {id : '1175', numeroid : '2016'},  
1281: {id : '1176', numeroid : '2017'},  
1282: {id : '1177', numeroid : '2018'},  
1283: {id : '1178', numeroid : '2019'},  
1284: {id : '1179', numeroid : '2020'},  
1285: {id : '1180', numeroid : '2021'},  
1286: {id : '1181', numeroid : '2022'},  
1287: {id : '1182', numeroid : '2023'},  
1288: {id : '1183', numeroid : '2024'},  
1289: {id : '1184', numeroid : '2025'},  
1290: {id : '1185', numeroid : '2026'},  
1291: {id : '1186', numeroid : '2027'},  
1292: {id : '1187', numeroid : '2028'},  
1293: {id : '1188', numeroid : '2029'},  
1294: {id : '1189', numeroid : '2030'},  
1295: {id : '1190', numeroid : '2031'},  
1296: {id : '1191', numeroid : '2032'},  
1297: {id : '1192', numeroid : '2033'},  
1298: {id : '1193', numeroid : '2034'},  
1299: {id : '1194', numeroid : '2035'},  
1300: {id : '1195', numeroid : '2036'},  
1301: {id : '1196', numeroid : '2037'},  
1302: {id : '1197', numeroid : '2038'},  
1303: {id : '1198', numeroid : '2044'},  
1304: {id : '1199', numeroid : '2045'},  
1305: {id : '1200', numeroid : '2046'},  
1306: {id : '1201', numeroid : '2047'},  
1307: {id : '1202', numeroid : '2048'},  
1308: {id : '1203', numeroid : '2049'},  
1309: {id : '1204', numeroid : '2050'},  
1310: {id : '1205', numeroid : '2051'},  
1311: {id : '1206', numeroid : '2052'},  
1312: {id : '1207', numeroid : '2053'},  
1313: {id : '1208', numeroid : '2054'},  
1314: {id : '1209', numeroid : '2055'},  
1315: {id : '1210', numeroid : '2056'},  
1316: {id : '1211', numeroid : '2057'},  
1317: {id : '1212', numeroid : '2058'},  
1318: {id : '1213', numeroid : '2059'},  
1319: {id : '1214', numeroid : '2067'},  
1320: {id : '1215', numeroid : '2068'},  
1321: {id : '1216', numeroid : '2069'},  
1322: {id : '1217', numeroid : '2070'},  
1323: {id : '1218', numeroid : '2071'},  
1324: {id : '1219', numeroid : '2072'},  
1325: {id : '1220', numeroid : '2073'},  
1326: {id : '1221', numeroid : '2074'},  
1327: {id : '1222', numeroid : '2075'},  
1328: {id : '1223', numeroid : '2076'},  
1329: {id : '1224', numeroid : '2077'},  
1330: {id : '1225', numeroid : '2078'},  
1331: {id : '1226', numeroid : '2079'},  
1332: {id : '1227', numeroid : '2186'},  
1333: {id : '1228', numeroid : '2187'},  
1334: {id : '1229', numeroid : '2188'},  
1335: {id : '1230', numeroid : '2189'},  
1336: {id : '1231', numeroid : '2190'},  
1337: {id : '1232', numeroid : '2191'},  
1338: {id : '1233', numeroid : '2192'},  
1339: {id : '1234', numeroid : '2193'},  
1340: {id : '1235', numeroid : '2194'},  
1341: {id : '1236', numeroid : '2195'},  
1342: {id : '1237', numeroid : '2196'},  
1343: {id : '1238', numeroid : '2197'},  
1344: {id : '1239', numeroid : '2198'},  
1345: {id : '1240', numeroid : '2199'},  
1346: {id : '1241', numeroid : '2200'},  
1347: {id : '1242', numeroid : '2201'},  
1348: {id : '1243', numeroid : '2202'},  
1349: {id : '1244', numeroid : '2203'},  
1350: {id : '1245', numeroid : '2204'},  
1351: {id : '1246', numeroid : '2205'},  
1352: {id : '1247', numeroid : '2206'},  
1353: {id : '1248', numeroid : '2207'},  
1354: {id : '1249', numeroid : '2208'},  
1355: {id : '1250', numeroid : '2209'},  
1356: {id : '1251', numeroid : '2210'},  
1357: {id : '1252', numeroid : '2211'},  
1358: {id : '1253', numeroid : '2212'},  
1359: {id : '1254', numeroid : '2213'},  
1360: {id : '1255', numeroid : '2214'},  
1361: {id : '1256', numeroid : '2215'},  
1362: {id : '1257', numeroid : '2216'},  
1363: {id : '1258', numeroid : '2217'},  
1364: {id : '1259', numeroid : '2218'},  
1365: {id : '1260', numeroid : '2219'},  
1366: {id : '1261', numeroid : '2222'},  
1367: {id : '1262', numeroid : '2224'},  
1368: {id : '1263', numeroid : '2225'},  
1369: {id : '1264', numeroid : '2226'},  
1370: {id : '1265', numeroid : '2227'},  
1371: {id : '1266', numeroid : '2228'},  
1372: {id : '1267', numeroid : '2229'},  
1373: {id : '1268', numeroid : '2232'},  
1374: {id : '1269', numeroid : '2233'},  
1375: {id : '1270', numeroid : '2234'},  
1376: {id : '1271', numeroid : '2235'},  
1377: {id : '1272', numeroid : '2236'},  
1378: {id : '1273', numeroid : '2237'},  
1379: {id : '1274', numeroid : '2238'},  
1380: {id : '1275', numeroid : '2239'},  
1381: {id : '1276', numeroid : '2240'},  
1382: {id : '1277', numeroid : '2241'},  
1383: {id : '1278', numeroid : '2242'},  
1384: {id : '1279', numeroid : '2243'},  
1385: {id : '1280', numeroid : '2244'},  
1386: {id : '1281', numeroid : '2245'},  
1387: {id : '1282', numeroid : '2246'},  
1388: {id : '1283', numeroid : '2247'},  
1389: {id : '1284', numeroid : '2248'},  
1390: {id : '1285', numeroid : '2249'},  
1391: {id : '1286', numeroid : '2250'},  
1392: {id : '1287', numeroid : '2251'},  
1393: {id : '1288', numeroid : '2252'},  
1394: {id : '1289', numeroid : '2253'},  
1395: {id : '1290', numeroid : '2254'},  
1396: {id : '1291', numeroid : '2256'},  
1397: {id : '1292', numeroid : '2257'},  
1398: {id : '1293', numeroid : '2258'},  
1399: {id : '1294', numeroid : '2259'},  
1400: {id : '1295', numeroid : '2260'},  
1401: {id : '1296', numeroid : '2261'},  
1402: {id : '1297', numeroid : '2262'},  
1403: {id : '1298', numeroid : '2263'},  
1404: {id : '1299', numeroid : '2264'},  
1405: {id : '1300', numeroid : '2265'},  
1406: {id : '1301', numeroid : '2266'},  
1407: {id : '1302', numeroid : '2267'},  
1408: {id : '1303', numeroid : '2269'},  
1409: {id : '1304', numeroid : '2270'},  
1410: {id : '1305', numeroid : '2271'},  
1411: {id : '1306', numeroid : '2272'},  
1412: {id : '1307', numeroid : '2273'},  
1413: {id : '1308', numeroid : '2274'},  
1414: {id : '1309', numeroid : '2275'},  
1415: {id : '1310', numeroid : '2276'},  
1416: {id : '1311', numeroid : '2277'},  
1417: {id : '1312', numeroid : '2278'},  
1418: {id : '1313', numeroid : '2279'},  
1419: {id : '1314', numeroid : '2280'},  
1420: {id : '1315', numeroid : '2281'},  
1421: {id : '1316', numeroid : '2282'},  
1422: {id : '1317', numeroid : '2283'},  
1423: {id : '1318', numeroid : '2284'},  
1424: {id : '1319', numeroid : '2285'},  
1425: {id : '1320', numeroid : '2286'},  
1426: {id : '1321', numeroid : '2287'},  
1427: {id : '1322', numeroid : '2288'},  
1428: {id : '1323', numeroid : '2289'},  
1429: {id : '1324', numeroid : '2290'},  
1430: {id : '1325', numeroid : '2291'},  
1431: {id : '1326', numeroid : '2293'},  
1432: {id : '1327', numeroid : '2294'},  
1433: {id : '1328', numeroid : '2295'},  
1434: {id : '1329', numeroid : '2296'},  
1435: {id : '1330', numeroid : '2297'},  
1436: {id : '1331', numeroid : '2298'},  
1437: {id : '1332', numeroid : '2299'},  
1438: {id : '1333', numeroid : '2300'},  
1439: {id : '1334', numeroid : '2301'},  
1440: {id : '1335', numeroid : '2302'},  
1441: {id : '1336', numeroid : '2303'},  
1442: {id : '1337', numeroid : '2304'},  
1443: {id : '1338', numeroid : '2305'},  
1444: {id : '1339', numeroid : '2306'},  
1445: {id : '1340', numeroid : '2307'},  
1446: {id : '1341', numeroid : '2308'},  
1447: {id : '1342', numeroid : '2309'},  
1448: {id : '1343', numeroid : '2310'},  
1449: {id : '1344', numeroid : '2311'},  
1450: {id : '1345', numeroid : '2312'},  
1451: {id : '1346', numeroid : '2313'},  
1452: {id : '1347', numeroid : '2315'},  
1453: {id : '1348', numeroid : '2316'},  
1454: {id : '1349', numeroid : '2317'},  
1455: {id : '1350', numeroid : '2318'},  
1456: {id : '1351', numeroid : '2319'},  
1457: {id : '1352', numeroid : '2320'},  
1458: {id : '1353', numeroid : '2321'},  
1459: {id : '1354', numeroid : '2322'},  
1460: {id : '1355', numeroid : '2323'},  
1461: {id : '1356', numeroid : '2324'},  
1462: {id : '1357', numeroid : '2325'},  
1463: {id : '1358', numeroid : '2326'},  
1464: {id : '1359', numeroid : '2327'},  
1465: {id : '1360', numeroid : '2328'},  
1466: {id : '1361', numeroid : '2329'},  
1467: {id : '1362', numeroid : '2330'},  
1468: {id : '1363', numeroid : '2331'},  
1469: {id : '1364', numeroid : '2332'},  
1470: {id : '1365', numeroid : '2333'},  
1471: {id : '1366', numeroid : '2334'},  
1472: {id : '1367', numeroid : '2335'},  
1473: {id : '1368', numeroid : '2336'},  
1474: {id : '1369', numeroid : '2337'},  
1475: {id : '1370', numeroid : '2338'},  
1476: {id : '1371', numeroid : '2339'},  
1477: {id : '1372', numeroid : '2340'},  
1478: {id : '1373', numeroid : '2341'},  
1479: {id : '1374', numeroid : '2342'},  
1480: {id : '1375', numeroid : '2343'},  
1481: {id : '1376', numeroid : '2344'},  
1482: {id : '1377', numeroid : '2345'},  
1483: {id : '1378', numeroid : '2346'},  
1484: {id : '1379', numeroid : '2347'},  
1485: {id : '1380', numeroid : '2348'},  
1486: {id : '1381', numeroid : '2350'},  
1487: {id : '1382', numeroid : '2351'},  
1488: {id : '1383', numeroid : '2352'},  
1489: {id : '1384', numeroid : '2353'},  
1490: {id : '1385', numeroid : '2354'},  
1491: {id : '1386', numeroid : '2356'},  
1492: {id : '1387', numeroid : '2357'},  
1493: {id : '1388', numeroid : '2358'},  
1494: {id : '1389', numeroid : '2359'},  
1495: {id : '1390', numeroid : '2360'},  
1496: {id : '1391', numeroid : '2361'},  
1497: {id : '1392', numeroid : '2362'},  
1498: {id : '1393', numeroid : '2363'},  
1499: {id : '1394', numeroid : '2364'},  
1500: {id : '1395', numeroid : '2366'},  
1501: {id : '1396', numeroid : '2367'},  
1502: {id : '1397', numeroid : '2368'},  
1503: {id : '1398', numeroid : '2369'},  
1504: {id : '1399', numeroid : '2370'},  
1505: {id : '1400', numeroid : '2371'},  
1506: {id : '1401', numeroid : '2372'},  
1507: {id : '1402', numeroid : '2373'},  
1508: {id : '1403', numeroid : '2374'},  
1509: {id : '1404', numeroid : '2375'},  
1510: {id : '1405', numeroid : '2376'},  
1511: {id : '1406', numeroid : '2377'},  
1512: {id : '1407', numeroid : '2378'},  
1513: {id : '1408', numeroid : '2379'},  
1514: {id : '1409', numeroid : '2380'},  
1515: {id : '1410', numeroid : '2381'},  
1516: {id : '1411', numeroid : '2382'},  
1517: {id : '1412', numeroid : '2383'},  
1518: {id : '1413', numeroid : '2384'},  
1519: {id : '1414', numeroid : '2385'},  
1520: {id : '1415', numeroid : '2386'},  
1521: {id : '1416', numeroid : '2387'},  
1522: {id : '1417', numeroid : '2388'},  
1523: {id : '1418', numeroid : '2389'},  
1524: {id : '1419', numeroid : '2390'},  
1525: {id : '1420', numeroid : '2391'},  
1526: {id : '1421', numeroid : '2392'},  
1527: {id : '1422', numeroid : '2393'},  
1528: {id : '1423', numeroid : '2394'},  
1529: {id : '1424', numeroid : '2395'},  
1530: {id : '1425', numeroid : '2396'},  
1531: {id : '1426', numeroid : '2397'},  
1532: {id : '1427', numeroid : '2398'},  
1533: {id : '1428', numeroid : '2399'},  
1534: {id : '1429', numeroid : '2400'},  
1535: {id : '1430', numeroid : '2401'},  
1536: {id : '1431', numeroid : '2402'},  
1537: {id : '1432', numeroid : '2403'},  
1538: {id : '1433', numeroid : '2404'},  
1539: {id : '1434', numeroid : '2405'},  
1540: {id : '1435', numeroid : '2406'},  
1541: {id : '1436', numeroid : '2407'},  
1542: {id : '1437', numeroid : '2409'},  
1543: {id : '1438', numeroid : '2410'},  
1544: {id : '1439', numeroid : '2411'},  
1545: {id : '1440', numeroid : '2412'},  
1546: {id : '1441', numeroid : '2413'},  
1547: {id : '1442', numeroid : '2414'},  
1548: {id : '1443', numeroid : '2416'},  
1549: {id : '1444', numeroid : '2417'},  
1550: {id : '1445', numeroid : '2418'},  
1551: {id : '1446', numeroid : '2419'},  
1552: {id : '1447', numeroid : '2420'},  
1553: {id : '1448', numeroid : '2421'},  
1554: {id : '1449', numeroid : '2422'},  
1555: {id : '1450', numeroid : '2424'},  
1556: {id : '1451', numeroid : '2426'},  
1557: {id : '1452', numeroid : '2427'},  
1558: {id : '1453', numeroid : '2428'},  
1559: {id : '1454', numeroid : '2429'},  
1560: {id : '1455', numeroid : '2430'},  
1561: {id : '1456', numeroid : '2431'},  
1562: {id : '1457', numeroid : '2432'},  
1563: {id : '1458', numeroid : '2433'},  
1564: {id : '1459', numeroid : '2434'},  
1565: {id : '1460', numeroid : '2435'},  
1566: {id : '1461', numeroid : '2436'},  
1567: {id : '1462', numeroid : '2437'},  
1568: {id : '1463', numeroid : '2438'},  
1569: {id : '1464', numeroid : '2439'},  
1570: {id : '1465', numeroid : '2440'},  
1571: {id : '1466', numeroid : '2441'},  
1572: {id : '1467', numeroid : '2442'},  
1573: {id : '1468', numeroid : '2443'},  
1574: {id : '1469', numeroid : '2444'},  
1575: {id : '1470', numeroid : '2445'},  
1576: {id : '1471', numeroid : '2446'},  
1577: {id : '1472', numeroid : '2447'},  
1578: {id : '1473', numeroid : '2448'},  
1579: {id : '1474', numeroid : '2451'},  
1580: {id : '1475', numeroid : '2452'},  
1581: {id : '1476', numeroid : '2453'},  
1582: {id : '1477', numeroid : '2454'},  
1583: {id : '1478', numeroid : '2455'},  
1584: {id : '1479', numeroid : '2456'},  
1585: {id : '1480', numeroid : '2457'},  
1586: {id : '1481', numeroid : '2458'},  
1587: {id : '1482', numeroid : '2459'},  
1588: {id : '1483', numeroid : '2460'},  
1589: {id : '1484', numeroid : '2461'},  
1590: {id : '1485', numeroid : '2463'},  
1591: {id : '1486', numeroid : '2464'},  
1592: {id : '1487', numeroid : '2465'},  
1593: {id : '1488', numeroid : '2466'},  
1594: {id : '1489', numeroid : '2467'},  
1595: {id : '1490', numeroid : '2468'},  
1596: {id : '1491', numeroid : '2469'},  
1597: {id : '1492', numeroid : '2470'},  
1598: {id : '1493', numeroid : '2471'},  
1599: {id : '1494', numeroid : '2473'},  
1600: {id : '1495', numeroid : '2474'},  
1601: {id : '1496', numeroid : '2475'},  
1602: {id : '1497', numeroid : '2477'},  
1603: {id : '1498', numeroid : '2478'},  
1604: {id : '1499', numeroid : '2480'},  
1605: {id : '1500', numeroid : '2481'},  
1606: {id : '1501', numeroid : '2482'},  
1607: {id : '1502', numeroid : '2483'},  
1608: {id : '1503', numeroid : '2484'},  
1609: {id : '1504', numeroid : '2485'},  
1610: {id : '1505', numeroid : '2486'},  
1611: {id : '1506', numeroid : '2487'},  
1612: {id : '1507', numeroid : '2488'},  
1613: {id : '1508', numeroid : '2489'},  
1614: {id : '1509', numeroid : '2490'},  
1615: {id : '1510', numeroid : '2491'},  
1616: {id : '1511', numeroid : '2493'},  
1617: {id : '1512', numeroid : '2495'},  
1618: {id : '1513', numeroid : '2496'},  
1619: {id : '1514', numeroid : '2497'},  
1620: {id : '1515', numeroid : '2498'},  
1621: {id : '1516', numeroid : '2501'},  
1622: {id : '1517', numeroid : '2502'},  
1623: {id : '1518', numeroid : '2503'},  
1624: {id : '1519', numeroid : '2504'},  
1625: {id : '1520', numeroid : '2505'},  
1626: {id : '1521', numeroid : '2506'},  
1627: {id : '1522', numeroid : '2507'},  
1628: {id : '1523', numeroid : '2508'},  
1629: {id : '1524', numeroid : '2509'},  
1630: {id : '1525', numeroid : '2511'},  
1631: {id : '1526', numeroid : '2512'},  
1632: {id : '1527', numeroid : '2513'},  
1633: {id : '1528', numeroid : '2514'},  
1634: {id : '1529', numeroid : '2515'},  
1635: {id : '1530', numeroid : '2516'},  
1636: {id : '1531', numeroid : '2517'},  
1637: {id : '1532', numeroid : '2518'},  
1638: {id : '1533', numeroid : '2520'},  
1639: {id : '1534', numeroid : '2521'},  
1640: {id : '1535', numeroid : '2522'},  
1641: {id : '1536', numeroid : '2524'},  
1642: {id : '1537', numeroid : '2525'},  
1643: {id : '1538', numeroid : '2526'},  
1644: {id : '1539', numeroid : '2527'},  
1645: {id : '1540', numeroid : '2528'},  
1646: {id : '1541', numeroid : '2529'},  
1647: {id : '1542', numeroid : '2530'},  
1648: {id : '1543', numeroid : '2531'},  
1649: {id : '1544', numeroid : '2533'},  
1650: {id : '1545', numeroid : '2534'},  
1651: {id : '1546', numeroid : '2535'},  
1652: {id : '1547', numeroid : '2536'},  
1653: {id : '1548', numeroid : '2538'},  
1654: {id : '1549', numeroid : '2541'},  
1655: {id : '1550', numeroid : '2542'},  
1656: {id : '1551', numeroid : '2545'},  
1657: {id : '1552', numeroid : '2546'},  
1658: {id : '1553', numeroid : '2547'},  
1659: {id : '1554', numeroid : '2548'},  
1660: {id : '1555', numeroid : '2552'},  
1661: {id : '1556', numeroid : '2553'},  
1662: {id : '1557', numeroid : '2554'},  
1663: {id : '1558', numeroid : '2555'},  
1664: {id : '1559', numeroid : '2556'},  
1665: {id : '1560', numeroid : '2557'},  
1666: {id : '1561', numeroid : '2558'},  
1667: {id : '1562', numeroid : '2560'},  
1668: {id : '1563', numeroid : '2561'},  
1669: {id : '1564', numeroid : '2564'},  
1670: {id : '1565', numeroid : '2565'},  
1671: {id : '1566', numeroid : '2567'},  
1672: {id : '1567', numeroid : '2570'},  
1673: {id : '1568', numeroid : '2571'},  
1674: {id : '1569', numeroid : '2572'},  
1675: {id : '1570', numeroid : '2573'},  
1676: {id : '1571', numeroid : '2574'},  
1677: {id : '1572', numeroid : '2576'},  
1678: {id : '1573', numeroid : '2577'},  
1679: {id : '1574', numeroid : '2578'},  
1680: {id : '1575', numeroid : '2579'},  
1681: {id : '1576', numeroid : '2580'},  
1682: {id : '1577', numeroid : '2581'},  
1683: {id : '1578', numeroid : '2582'},  
1684: {id : '1579', numeroid : '2583'},  
1685: {id : '1580', numeroid : '2584'},  
1686: {id : '1581', numeroid : '2585'},  
1687: {id : '1582', numeroid : '2586'},  
1688: {id : '1583', numeroid : '2587'},  
1689: {id : '1584', numeroid : '2588'},  
1690: {id : '1585', numeroid : '2589'},  
1691: {id : '1586', numeroid : '2590'},  
1692: {id : '1587', numeroid : '2591'},  
1693: {id : '1588', numeroid : '2599'},  
1694: {id : '1589', numeroid : '2600'},  
1695: {id : '1590', numeroid : '2601'},  
1696: {id : '1591', numeroid : '2602'},  
1697: {id : '1592', numeroid : '2603'},  
1698: {id : '1593', numeroid : '2604'},  
1699: {id : '1594', numeroid : '2605'},  
1700: {id : '1595', numeroid : '2606'},  
1701: {id : '1596', numeroid : '2607'},  
1702: {id : '1597', numeroid : '2608'},  
1703: {id : '1598', numeroid : '2609'},  
1704: {id : '1599', numeroid : '2610'},  
1705: {id : '1600', numeroid : '2611'},  
1706: {id : '1601', numeroid : '2612'},  
1707: {id : '1602', numeroid : '2614'},  
1708: {id : '1603', numeroid : '2615'},  
1709: {id : '1604', numeroid : '2616'},  
1710: {id : '1605', numeroid : '2617'},  
1711: {id : '1606', numeroid : '2618'},  
1712: {id : '1607', numeroid : '2619'},  
1713: {id : '1608', numeroid : '2620'},  
1714: {id : '1609', numeroid : '2621'},  
1715: {id : '1610', numeroid : '2622'},  
1716: {id : '1611', numeroid : '2623'},  
1717: {id : '1612', numeroid : '2624'},  
1718: {id : '1613', numeroid : '2626'},  
1719: {id : '1614', numeroid : '2627'},  
1720: {id : '1615', numeroid : '2628'},  
1721: {id : '1616', numeroid : '2629'},  
1722: {id : '1617', numeroid : '2630'},  
1723: {id : '1618', numeroid : '2642'},  
1724: {id : '1619', numeroid : '2643'},  
1725: {id : '1620', numeroid : '2644'},  
1726: {id : '1621', numeroid : '2645'},  
1727: {id : '1622', numeroid : '2646'},  
1728: {id : '1623', numeroid : '2647'},  
1729: {id : '1624', numeroid : '2648'},  
1730: {id : '1625', numeroid : '2649'},  
1731: {id : '1626', numeroid : '2650'},  
1732: {id : '1627', numeroid : '2651'},  
1733: {id : '1628', numeroid : '2653'},  
1734: {id : '1629', numeroid : '2655'},  
1735: {id : '1630', numeroid : '2656'},  
1736: {id : '1631', numeroid : '2657'},  
1737: {id : '1632', numeroid : '2658'},  
1738: {id : '1633', numeroid : '2659'},  
1739: {id : '1634', numeroid : '2660'},  
1740: {id : '1635', numeroid : '2661'},  
1741: {id : '1636', numeroid : '2662'},  
1742: {id : '1637', numeroid : '2664'},  
1743: {id : '1638', numeroid : '2666'},  
1744: {id : '1639', numeroid : '2667'},  
1745: {id : '1640', numeroid : '2668'},  
1746: {id : '1641', numeroid : '2669'},  
1747: {id : '1642', numeroid : '2670'},  
1748: {id : '1643', numeroid : '2671'},  
1749: {id : '1644', numeroid : '2672'},  
1750: {id : '1645', numeroid : '2673'},  
1751: {id : '1646', numeroid : '2674'},  
1752: {id : '1647', numeroid : '2676'},  
1753: {id : '1648', numeroid : '2677'},  
1754: {id : '1649', numeroid : '2678'},  
1755: {id : '1650', numeroid : '2679'},  
1756: {id : '1651', numeroid : '2680'},  
1757: {id : '1652', numeroid : '2681'},  
1758: {id : '1653', numeroid : '2682'},  
1759: {id : '1654', numeroid : '2683'},  
1760: {id : '1655', numeroid : '2684'},  
1761: {id : '1656', numeroid : '2685'},  
1762: {id : '1657', numeroid : '2686'},  
1763: {id : '1658', numeroid : '2687'},  
1764: {id : '1659', numeroid : '2688'},  
1765: {id : '1660', numeroid : '2689'},  
1766: {id : '1661', numeroid : '2690'},  
1767: {id : '1662', numeroid : '2691'},  
1768: {id : '1663', numeroid : '2692'},  
1769: {id : '1664', numeroid : '2693'},  
1770: {id : '1665', numeroid : '2698'},  
1771: {id : '1666', numeroid : '2699'},  
1772: {id : '1667', numeroid : '2705'},  
1773: {id : '1668', numeroid : '2707'},  
1774: {id : '1669', numeroid : '2708'},  
1775: {id : '1670', numeroid : '2709'},  
1776: {id : '1671', numeroid : '2710'},  
1777: {id : '1672', numeroid : '2711'},  
1778: {id : '1673', numeroid : '2713'},  
1779: {id : '1674', numeroid : '2714'},  
1780: {id : '1675', numeroid : '2715'},  
1781: {id : '1676', numeroid : '2716'},  
1782: {id : '1677', numeroid : '2717'},  
1783: {id : '1678', numeroid : '2719'},  
1784: {id : '1679', numeroid : '2720'},  
1785: {id : '1680', numeroid : '2721'},  
1786: {id : '1681', numeroid : '2722'},  
1787: {id : '1682', numeroid : '2723'},  
1788: {id : '1683', numeroid : '2724'},  
1789: {id : '1684', numeroid : '2725'},  
1790: {id : '1685', numeroid : '2726'},  
1791: {id : '1686', numeroid : '2727'},  
1792: {id : '1687', numeroid : '2728'},  
1793: {id : '1688', numeroid : '2729'},  
1794: {id : '1689', numeroid : '2730'},  
1795: {id : '1690', numeroid : '2732'},  
1796: {id : '1691', numeroid : '2733'},  
1797: {id : '1692', numeroid : '2734'},  
1798: {id : '1693', numeroid : '2735'},  
1799: {id : '1694', numeroid : '2738'},  
1800: {id : '1695', numeroid : '2739'},  
1801: {id : '1696', numeroid : '2740'},  
1802: {id : '1697', numeroid : '2741'},  
1803: {id : '1698', numeroid : '2742'},  
1804: {id : '1699', numeroid : '2743'},  
1805: {id : '1700', numeroid : '2744'},  
1806: {id : '1701', numeroid : '2745'},  
1807: {id : '1702', numeroid : '2746'},  
1808: {id : '1703', numeroid : '2747'},  
1809: {id : '1704', numeroid : '2748'},  
1810: {id : '1705', numeroid : '2749'},  
1811: {id : '1706', numeroid : '2750'},  
1812: {id : '1707', numeroid : '2751'},  
1813: {id : '1708', numeroid : '2752'},  
1814: {id : '1709', numeroid : '2753'},  
1815: {id : '1710', numeroid : '2754'},  
1816: {id : '1711', numeroid : '2757'},  
1817: {id : '1712', numeroid : '2758'},  
1818: {id : '1713', numeroid : '2759'},  
1819: {id : '1714', numeroid : '2760'},  
1820: {id : '1715', numeroid : '2761'},  
1821: {id : '1716', numeroid : '2762'},  
1822: {id : '1717', numeroid : '2763'},  
1823: {id : '1718', numeroid : '2764'},  
1824: {id : '1719', numeroid : '2765'},  
1825: {id : '1720', numeroid : '2766'},  
1826: {id : '1721', numeroid : '2767'},  
1827: {id : '1722', numeroid : '2768'},  
1828: {id : '1723', numeroid : '2769'},  
1829: {id : '1724', numeroid : '2770'},  
1830: {id : '1725', numeroid : '2771'},  
1831: {id : '1726', numeroid : '2772'},  
1832: {id : '1727', numeroid : '2773'},  
1833: {id : '1728', numeroid : '2774'},  
1834: {id : '1729', numeroid : '2775'},  
1835: {id : '1730', numeroid : '2776'},  
1836: {id : '1731', numeroid : '2777'},  
1837: {id : '1732', numeroid : '2778'},  
1838: {id : '1733', numeroid : '2779'},  
1839: {id : '1734', numeroid : '2780'},  
1840: {id : '1735', numeroid : '2781'},  
1841: {id : '1736', numeroid : '2782'},  
1842: {id : '1737', numeroid : '2783'},  
1843: {id : '1738', numeroid : '2784'},  
1844: {id : '1739', numeroid : '2785'},  
1845: {id : '1740', numeroid : '2786'},  
1846: {id : '1741', numeroid : '2787'},  
1847: {id : '1742', numeroid : '2788'},  
1848: {id : '1743', numeroid : '2789'},  
1849: {id : '1744', numeroid : '2790'},  
1850: {id : '1745', numeroid : '2793'},  
1851: {id : '1746', numeroid : '2794'},  
1852: {id : '1747', numeroid : '2795'},  
1853: {id : '1748', numeroid : '2796'},  
1854: {id : '1749', numeroid : '2797'},  
1855: {id : '1750', numeroid : '2798'},  
1856: {id : '1751', numeroid : '2799'},  
1857: {id : '1752', numeroid : '2800'},  
1858: {id : '1753', numeroid : '2801'},  
1859: {id : '1754', numeroid : '2802'},  
1860: {id : '1755', numeroid : '2803'},  
1861: {id : '1756', numeroid : '2805'},  
1862: {id : '1757', numeroid : '2806'},  
1863: {id : '1758', numeroid : '2807'},  
1864: {id : '1759', numeroid : '2809'},  
1865: {id : '1760', numeroid : '2810'},  
1866: {id : '1761', numeroid : '2811'},  
1867: {id : '1762', numeroid : '2812'},  
1868: {id : '1763', numeroid : '2813'},  
1869: {id : '1764', numeroid : '2814'},  
1870: {id : '1765', numeroid : '2815'},  
1871: {id : '1766', numeroid : '2817'},  
1872: {id : '1767', numeroid : '2818'},  
1873: {id : '1768', numeroid : '2819'},  
1874: {id : '1769', numeroid : '2820'},  
1875: {id : '1770', numeroid : '2821'},  
1876: {id : '1771', numeroid : '2822'},  
1877: {id : '1772', numeroid : '2823'},  
1878: {id : '1773', numeroid : '2826'},  
1879: {id : '1774', numeroid : '2829'},  
1880: {id : '1775', numeroid : '2830'},  
1881: {id : '1776', numeroid : '2831'},  
1882: {id : '1777', numeroid : '2834'},  
1883: {id : '1778', numeroid : '2835'},  
1884: {id : '1779', numeroid : '2837'},  
1885: {id : '1780', numeroid : '2838'},  
1886: {id : '1781', numeroid : '2839'},  
1887: {id : '1782', numeroid : '2840'},  
1888: {id : '1783', numeroid : '2841'},  
1889: {id : '1784', numeroid : '2842'},  
1890: {id : '1785', numeroid : '2844'},  
1891: {id : '1786', numeroid : '2845'},  
1892: {id : '1787', numeroid : '2846'},  
1893: {id : '1788', numeroid : '2849'},  
1894: {id : '1789', numeroid : '2850'},  
1895: {id : '1790', numeroid : '2851'},  
1896: {id : '1791', numeroid : '2852'},  
1897: {id : '1792', numeroid : '2853'},  
1898: {id : '1793', numeroid : '2854'},  
1899: {id : '1794', numeroid : '2855'},  
1900: {id : '1795', numeroid : '2856'},  
1901: {id : '1796', numeroid : '2857'},  
1902: {id : '1797', numeroid : '2858'},  
1903: {id : '1798', numeroid : '2859'},  
1904: {id : '1799', numeroid : '2860'},  
1905: {id : '1800', numeroid : '2861'},  
1906: {id : '1801', numeroid : '2862'},  
1907: {id : '1802', numeroid : '2863'},  
1908: {id : '1803', numeroid : '2864'},  
1909: {id : '1804', numeroid : '2865'},  
1910: {id : '1805', numeroid : '2869'},  
1911: {id : '1806', numeroid : '2870'},  
1912: {id : '1807', numeroid : '2871'},  
1913: {id : '1808', numeroid : '2872'},  
1914: {id : '1809', numeroid : '2873'},  
1915: {id : '1810', numeroid : '2874'},  
1916: {id : '1811', numeroid : '2875'},  
1917: {id : '1812', numeroid : '2876'},  
1918: {id : '1813', numeroid : '2878'},  
1919: {id : '1814', numeroid : '2879'},  
1920: {id : '1815', numeroid : '2880'},  
1921: {id : '1816', numeroid : '2881'},  
1922: {id : '1817', numeroid : '2900'},  
1923: {id : '1818', numeroid : '2901'},  
1924: {id : '1819', numeroid : '2902'},  
1925: {id : '1820', numeroid : '2903'},  
1926: {id : '1821', numeroid : '2904'},  
1927: {id : '1822', numeroid : '2905'},  
1928: {id : '1823', numeroid : '2906'},  
1929: {id : '1824', numeroid : '2907'},  
1930: {id : '1825', numeroid : '2908'},  
1931: {id : '1826', numeroid : '2909'},  
1932: {id : '1827', numeroid : '2910'},  
1933: {id : '1828', numeroid : '2911'},  
1934: {id : '1829', numeroid : '2912'},  
1935: {id : '1830', numeroid : '2913'},  
1936: {id : '1831', numeroid : '2915'},  
1937: {id : '1832', numeroid : '2916'},  
1938: {id : '1833', numeroid : '2917'},  
1939: {id : '1834', numeroid : '2918'},  
1940: {id : '1835', numeroid : '2919'},  
1941: {id : '1836', numeroid : '2920'},  
1942: {id : '1837', numeroid : '2921'},  
1943: {id : '1838', numeroid : '2922'},  
1944: {id : '1839', numeroid : '2923'},  
1945: {id : '1840', numeroid : '2924'},  
1946: {id : '1841', numeroid : '2925'},  
1947: {id : '1842', numeroid : '2926'},  
1948: {id : '1843', numeroid : '2927'},  
1949: {id : '1844', numeroid : '2928'},  
1950: {id : '1845', numeroid : '2929'},  
1951: {id : '1846', numeroid : '2930'},  
1952: {id : '1847', numeroid : '2931'},  
1953: {id : '1848', numeroid : '2933'},  
1954: {id : '1849', numeroid : '2934'},  
1955: {id : '1850', numeroid : '2935'},  
1956: {id : '1851', numeroid : '2936'},  
1957: {id : '1852', numeroid : '2937'},  
1958: {id : '1853', numeroid : '2938'},  
1959: {id : '1854', numeroid : '2940'},  
1960: {id : '1855', numeroid : '2941'},  
1961: {id : '1856', numeroid : '2942'},  
1962: {id : '1857', numeroid : '2943'},  
1963: {id : '1858', numeroid : '2945'},  
1964: {id : '1859', numeroid : '2946'},  
1965: {id : '1860', numeroid : '2947'},  
1966: {id : '1861', numeroid : '2948'},  
1967: {id : '1862', numeroid : '2949'},  
1968: {id : '1863', numeroid : '2950'},  
1969: {id : '1864', numeroid : '2956'},  
1970: {id : '1865', numeroid : '2965'},  
1971: {id : '1866', numeroid : '2966'},  
1972: {id : '1867', numeroid : '2967'},  
1973: {id : '1868', numeroid : '2968'},  
1974: {id : '1869', numeroid : '2969'},  
1975: {id : '1870', numeroid : '2974'},  
1976: {id : '1871', numeroid : '2975'},  
1977: {id : '1872', numeroid : '2976'},  
1978: {id : '1873', numeroid : '2977'},  
1979: {id : '1874', numeroid : '2978'},  
1980: {id : '1875', numeroid : '2979'},  
1981: {id : '1876', numeroid : '2980'},  
1982: {id : '1877', numeroid : '2981'},  
1983: {id : '1878', numeroid : '2982'},  
1984: {id : '1879', numeroid : '2983'},  
1985: {id : '1880', numeroid : '2984'},  
1986: {id : '1881', numeroid : '2985'},  
1987: {id : '1882', numeroid : '2986'},  
1988: {id : '1883', numeroid : '2987'},  
1989: {id : '1884', numeroid : '2988'},  
1990: {id : '1885', numeroid : '2989'},  
1991: {id : '1886', numeroid : '2990'},  
1992: {id : '1887', numeroid : '2991'},  
1993: {id : '1888', numeroid : '2992'},  
1994: {id : '1889', numeroid : '2993'},  
1995: {id : '1890', numeroid : '2994'},  
1996: {id : '1891', numeroid : '2995'},  
1997: {id : '1892', numeroid : '2996'},  
1998: {id : '1893', numeroid : '2997'},  
1999: {id : '1894', numeroid : '2998'},  
2000: {id : '1895', numeroid : '2999'},  
2001: {id : '1896', numeroid : '3000'},  
2002: {id : '1897', numeroid : '3001'},  
2003: {id : '1898', numeroid : '3002'},  
2004: {id : '1899', numeroid : '3003'},  
2005: {id : '1900', numeroid : '3004'},  
2006: {id : '1901', numeroid : '3005'},  
2007: {id : '1902', numeroid : '3006'},  
2008: {id : '1903', numeroid : '3007'},  
2009: {id : '1904', numeroid : '3008'},  
2010: {id : '1905', numeroid : '3009'},  
2011: {id : '1906', numeroid : '3010'},  
2012: {id : '1907', numeroid : '3011'},  
2013: {id : '1908', numeroid : '3012'},  
2014: {id : '1909', numeroid : '3013'},  
2015: {id : '1910', numeroid : '3014'},  
2016: {id : '1911', numeroid : '3015'},  
2017: {id : '1912', numeroid : '3016'},  
2018: {id : '1913', numeroid : '3017'},  
2019: {id : '1914', numeroid : '3018'},  
2020: {id : '1915', numeroid : '3019'},  
2021: {id : '1916', numeroid : '3020'},  
2022: {id : '1917', numeroid : '3021'},  
2023: {id : '1918', numeroid : '3022'},  
2024: {id : '1919', numeroid : '3023'},  
2025: {id : '1920', numeroid : '3024'},  
2026: {id : '1921', numeroid : '3025'},  
2027: {id : '1922', numeroid : '3026'},  
2028: {id : '1923', numeroid : '3027'},  
2029: {id : '1924', numeroid : '3028'},  
2030: {id : '1925', numeroid : '3048'},  
2031: {id : '1926', numeroid : '3049'},  
2032: {id : '1927', numeroid : '3050'},  
2033: {id : '1928', numeroid : '3051'},  
2034: {id : '1929', numeroid : '3052'},  
2035: {id : '1930', numeroid : '3053'},  
2036: {id : '1931', numeroid : '3054'},  
2037: {id : '1932', numeroid : '3055'},  
2038: {id : '1933', numeroid : '3056'},  
2039: {id : '1934', numeroid : '3057'},  
2040: {id : '1935', numeroid : '3064'},  
2041: {id : '1936', numeroid : '3065'},  
2042: {id : '1937', numeroid : '3066'},  
2043: {id : '1938', numeroid : '3070'},  
2044: {id : '1939', numeroid : '3071'},  
2045: {id : '1940', numeroid : '3072'},  
2046: {id : '1941', numeroid : '3073'},  
2047: {id : '1942', numeroid : '3076'},  
2048: {id : '1943', numeroid : '3077'},  
2049: {id : '1944', numeroid : '3078'},  
2050: {id : '1945', numeroid : '3079'},  
2051: {id : '1946', numeroid : '3080'},  
2052: {id : '1947', numeroid : '3082'},  
2053: {id : '1948', numeroid : '3083'},  
2054: {id : '1949', numeroid : '3084'},  
2055: {id : '1950', numeroid : '3085'},  
2056: {id : '1951', numeroid : '3086'},  
2057: {id : '1952', numeroid : '3087'},  
2058: {id : '1953', numeroid : '3088'},  
2059: {id : '1954', numeroid : '3089'},  
2060: {id : '1955', numeroid : '3090'},  
2061: {id : '1956', numeroid : '3091'},  
2062: {id : '1957', numeroid : '3092'},  
2063: {id : '1958', numeroid : '3093'},  
2064: {id : '1959', numeroid : '3094'},  
2065: {id : '1960', numeroid : '3095'},  
2066: {id : '1961', numeroid : '3096'},  
2067: {id : '1962', numeroid : '3097'},  
2068: {id : '1963', numeroid : '3098'},  
2069: {id : '1964', numeroid : '3099'},  
2070: {id : '1965', numeroid : '3100'},  
2071: {id : '1966', numeroid : '3101'},  
2072: {id : '1967', numeroid : '3102'},  
2073: {id : '1968', numeroid : '3103'},  
2074: {id : '1969', numeroid : '3104'},  
2075: {id : '1970', numeroid : '3105'},  
2076: {id : '1971', numeroid : '3106'},  
2077: {id : '1972', numeroid : '3107'},  
2078: {id : '1973', numeroid : '3108'},  
2079: {id : '1974', numeroid : '3109'},  
2080: {id : '1975', numeroid : '3110'},  
2081: {id : '1976', numeroid : '3111'},  
2082: {id : '1977', numeroid : '3112'},  
2083: {id : '1978', numeroid : '3113'},  
2084: {id : '1979', numeroid : '3114'},  
2085: {id : '1980', numeroid : '3115'},  
2086: {id : '1981', numeroid : '3116'},  
2087: {id : '1982', numeroid : '3117'},  
2088: {id : '1983', numeroid : '3118'},  
2089: {id : '1984', numeroid : '3119'},  
2090: {id : '1985', numeroid : '3120'},  
2091: {id : '1986', numeroid : '3121'},  
2092: {id : '1987', numeroid : '3122'},  
2093: {id : '1988', numeroid : '3123'},  
2094: {id : '1989', numeroid : '3124'},  
2095: {id : '1990', numeroid : '3125'},  
2096: {id : '1991', numeroid : '3126'},  
2097: {id : '1992', numeroid : '3127'},  
2098: {id : '1993', numeroid : '3128'},  
2099: {id : '1994', numeroid : '3129'},  
2100: {id : '1995', numeroid : '3130'},  
2101: {id : '1996', numeroid : '3131'},  
2102: {id : '1997', numeroid : '3132'},  
2103: {id : '1998', numeroid : '3133'},  
2104: {id : '1999', numeroid : '3134'},  
2105: {id : '2000', numeroid : '3135'},  
2106: {id : '2001', numeroid : '3136'},  
2107: {id : '2002', numeroid : '3137'},  
2108: {id : '2003', numeroid : '3138'},  
2109: {id : '2004', numeroid : '3139'},  
2110: {id : '2005', numeroid : '3140'},  
2111: {id : '2006', numeroid : '3141'},  
2112: {id : '2007', numeroid : '3142'},  
2113: {id : '2008', numeroid : '3143'},  
2114: {id : '2009', numeroid : '3144'},  
2115: {id : '2010', numeroid : '3145'},  
2116: {id : '2011', numeroid : '3146'},  
2117: {id : '2012', numeroid : '3147'},  
2118: {id : '2013', numeroid : '3148'},  
2119: {id : '2014', numeroid : '3149'},  
2120: {id : '2015', numeroid : '3150'},  
2121: {id : '2016', numeroid : '3151'},  
2122: {id : '2017', numeroid : '3152'},  
2123: {id : '2018', numeroid : '3153'},  
2124: {id : '2019', numeroid : '3154'},  
2125: {id : '2020', numeroid : '3155'},  
2126: {id : '2021', numeroid : '3156'},  
2127: {id : '2022', numeroid : '3157'},  
2128: {id : '2023', numeroid : '3158'},  
2129: {id : '2024', numeroid : '3159'},  
2130: {id : '2025', numeroid : '3160'},  
2131: {id : '2026', numeroid : '3161'},  
2132: {id : '2027', numeroid : '3162'},  
2133: {id : '2028', numeroid : '3163'},  
2134: {id : '2029', numeroid : '3164'},  
2135: {id : '2030', numeroid : '3165'},  
2136: {id : '2031', numeroid : '3166'},  
2137: {id : '2032', numeroid : '3167'},  
2138: {id : '2033', numeroid : '3168'},  
2139: {id : '2034', numeroid : '3169'},  
2140: {id : '2035', numeroid : '3170'},  
2141: {id : '2036', numeroid : '3171'},  
2142: {id : '2037', numeroid : '3172'},  
2143: {id : '2038', numeroid : '3174'},  
2144: {id : '2039', numeroid : '3175'},  
2145: {id : '2040', numeroid : '3176'},  
2146: {id : '2041', numeroid : '3178'},  
2147: {id : '2042', numeroid : '3179'},  
2148: {id : '2043', numeroid : '3180'},  
2149: {id : '2044', numeroid : '3181'},  
2150: {id : '2045', numeroid : '3182'},  
2151: {id : '2046', numeroid : '3183'},  
2152: {id : '2047', numeroid : '3184'},  
2153: {id : '2048', numeroid : '3185'},  
2154: {id : '2049', numeroid : '3186'},  
2155: {id : '2050', numeroid : '3187'},  
2156: {id : '2051', numeroid : '3188'},  
2157: {id : '2052', numeroid : '3189'},  
2158: {id : '2053', numeroid : '3190'},  
2159: {id : '2054', numeroid : '3191'},  
2160: {id : '2055', numeroid : '3192'},  
2161: {id : '2056', numeroid : '3194'},  
2162: {id : '2057', numeroid : '3200'},  
2163: {id : '2058', numeroid : '3203'},  
2164: {id : '2059', numeroid : '3205'},  
2165: {id : '2060', numeroid : '3206'},  
2166: {id : '2061', numeroid : '3207'},  
2167: {id : '2062', numeroid : '3208'},  
2168: {id : '2063', numeroid : '3209'},  
2169: {id : '2064', numeroid : '3210'},  
2170: {id : '2065', numeroid : '3211'},  
2171: {id : '2066', numeroid : '3212'},  
2172: {id : '2067', numeroid : '3213'},  
2173: {id : '2068', numeroid : '3214'},  
2174: {id : '2069', numeroid : '3215'},  
2175: {id : '2070', numeroid : '3216'},  
2176: {id : '2071', numeroid : '3217'},  
2177: {id : '2072', numeroid : '3218'},  
2178: {id : '2073', numeroid : '3219'},  
2179: {id : '2074', numeroid : '3220'},  
2180: {id : '2075', numeroid : '3221'},  
2181: {id : '2076', numeroid : '3222'},  
2182: {id : '2077', numeroid : '3223'},  
2183: {id : '2078', numeroid : '3224'},  
2184: {id : '2079', numeroid : '3225'},  
2185: {id : '2080', numeroid : '3226'},  
2186: {id : '2081', numeroid : '3227'},  
2187: {id : '2082', numeroid : '3228'},  
2188: {id : '2083', numeroid : '3229'},  
2189: {id : '2084', numeroid : '3230'},  
2190: {id : '2085', numeroid : '3231'},  
2191: {id : '2086', numeroid : '3232'},  
2192: {id : '2087', numeroid : '3233'},  
2193: {id : '2088', numeroid : '3234'},  
2194: {id : '2089', numeroid : '3235'},  
2195: {id : '2090', numeroid : '3236'},  
2196: {id : '2091', numeroid : '3237'},  
2197: {id : '2092', numeroid : '3238'},  
2198: {id : '2093', numeroid : '3239'},  
2199: {id : '2094', numeroid : '3240'},  
2200: {id : '2095', numeroid : '3241'},  
2201: {id : '2096', numeroid : '3242'},  
2202: {id : '2097', numeroid : '3243'},  
2203: {id : '2098', numeroid : '3244'},  
2204: {id : '2099', numeroid : '3245'},  
2205: {id : '2100', numeroid : '3246'},  
2206: {id : '2101', numeroid : '3247'},  
2207: {id : '2102', numeroid : '3248'},  
2208: {id : '2103', numeroid : '3249'},  
2209: {id : '2104', numeroid : '3250'},  
2210: {id : '2105', numeroid : '3256'},  
2211: {id : '2106', numeroid : '3257'},  
2212: {id : '2107', numeroid : '3258'},  
2213: {id : '2108', numeroid : '3259'},  
2214: {id : '2109', numeroid : '3260'},  
2215: {id : '2110', numeroid : '3261'},  
2216: {id : '2111', numeroid : '3262'},  
2217: {id : '2112', numeroid : '3263'},  
2218: {id : '2113', numeroid : '3264'},  
2219: {id : '2114', numeroid : '3265'},  
2220: {id : '2115', numeroid : '3266'},  
2221: {id : '2116', numeroid : '3267'},  
2222: {id : '2117', numeroid : '3271'},  
2223: {id : '2118', numeroid : '3272'},  
2224: {id : '2119', numeroid : '3273'},  
2225: {id : '2120', numeroid : '3274'},  
2226: {id : '2121', numeroid : '3275'},  
2227: {id : '2122', numeroid : '3276'},  
2228: {id : '2123', numeroid : '3277'},  
2229: {id : '2124', numeroid : '3278'},  
2230: {id : '2125', numeroid : '3279'},  
2231: {id : '2126', numeroid : '3280'},  
2232: {id : '2127', numeroid : '3281'},  
2233: {id : '2128', numeroid : '3282'},  
2234: {id : '2129', numeroid : '3283'},  
2235: {id : '2130', numeroid : '3284'},  
2236: {id : '2131', numeroid : '3285'},  
2237: {id : '2132', numeroid : '3286'},  
2238: {id : '2133', numeroid : '3287'},  
2239: {id : '2134', numeroid : '3288'},  
2240: {id : '2135', numeroid : '3289'},  
2241: {id : '2136', numeroid : '3290'},  
2242: {id : '2137', numeroid : '3291'},  
2243: {id : '2138', numeroid : '3295'},  
2244: {id : '2139', numeroid : '3301'},  
2245: {id : '2140', numeroid : '3303'},  
2246: {id : '2141', numeroid : '3304'},  
2247: {id : '2142', numeroid : '3305'},  
2248: {id : '2143', numeroid : '3306'},  
2249: {id : '2144', numeroid : '3307'},  
2250: {id : '2145', numeroid : '3308'},  
2251: {id : '2146', numeroid : '3309'},  
2252: {id : '2147', numeroid : '3310'},  
2253: {id : '2148', numeroid : '3311'},  
2254: {id : '2149', numeroid : '3312'},  
2255: {id : '2150', numeroid : '3313'},  
2256: {id : '2151', numeroid : '3315'},  
2257: {id : '2152', numeroid : '3319'},  
2258: {id : '2153', numeroid : '3321'},  
2259: {id : '2154', numeroid : '3322'},  
2260: {id : '2155', numeroid : '3323'},  
2261: {id : '2156', numeroid : '3324'},  
2262: {id : '2157', numeroid : '3325'},  
2263: {id : '2158', numeroid : '3326'},  
2264: {id : '2159', numeroid : '3327'},  
2265: {id : '2160', numeroid : '3328'},  
2266: {id : '2161', numeroid : '3329'},  
2267: {id : '2162', numeroid : '3330'},  
2268: {id : '2163', numeroid : '3331'},  
2269: {id : '2164', numeroid : '3332'},  
2270: {id : '2165', numeroid : '3333'},  
2271: {id : '2166', numeroid : '3334'},  
2272: {id : '2167', numeroid : '3335'},  
2273: {id : '2168', numeroid : '3336'},  
2274: {id : '2169', numeroid : '3342'},  
2275: {id : '2170', numeroid : '3343'},  
2276: {id : '2171', numeroid : '3344'},  
2277: {id : '2172', numeroid : '3345'},  
2278: {id : '2173', numeroid : '3346'},  
2279: {id : '2174', numeroid : '3347'},  
2280: {id : '2175', numeroid : '3348'},  
2281: {id : '2176', numeroid : '3349'},  
2282: {id : '2177', numeroid : '3350'},  
2283: {id : '2178', numeroid : '3351'},  
2284: {id : '2179', numeroid : '3352'},  
2285: {id : '2180', numeroid : '3354'},  
2286: {id : '2181', numeroid : '3355'},  
2287: {id : '2182', numeroid : '3357'},  
2288: {id : '2183', numeroid : '3361'},  
2289: {id : '2184', numeroid : '3362'},  
2290: {id : '2185', numeroid : '3372'}  
2291:  
2292:];

## .\mobile\www\js\combustivel.js

1:var lista\_combustivel = [  
 2: {id : '1', nome : "Álcool/Etanol", simples : true, moto : false},  
 3: {id : '2', nome : "Bi-combustível/Flex", simples : true, moto : true},  
 4: {id : '3', nome : "Diesel", simples : true, moto : false},  
 5: {id : '4', nome : "Gasolina", simples : true, moto : true},  
 6: {id : '5', nome : "GNV/Gás Natural", simples : true, moto : false},  
 7: {id : '6', nome : "Híbrido", simples : true, moto : false}  
 8:];

## .\mobile\www\js\controllers.js

1:var controllers = controllers || {};  
 2:  
 3://insert controller files  
 4:var insert\_controllers={  
 5: //files to be inserted  
 6: controller\_files : ['menu.js',   
 7: 'selecionar\_tipo.js',  
 8: 'confirmar\_veiculo.js',   
 9: 'simples/identificacao\_visual\_simples.js',  
 10: 'simples/caracterizacao\_simples.js',  
 11: 'simples/caracterizacao\_viagem\_simples.js',  
 12: 'onibus/identificacao\_visual\_onibus.js',  
 13: 'onibus/caracterizacao\_onibus.js',  
 14: 'onibus/caracterizacao\_viagem\_onibus.js',  
 15: 'carga/identificacao\_visual\_carga.js',  
 16: 'carga/caracterizacao\_carga.js',  
 17: 'carga/caracterizacao\_viagem\_carga.js',  
 18: 'carga/caracterizacao\_viagem2\_carga.js',  
 19: 'pergunta\_extra.js',  
 20: 'sumario.js'  
 21: ],  
 22:   
 23: controller\_files\_path : 'controllers/',  
 24:   
 25: insert : function(){  
 26: var me = this;   
 27: $.each(me.controller\_files,  
 28: function(index, item){  
 29: var s = document.createElement("script");  
 30: s.type = "text/javascript";  
 31: s.src = me.controller\_files\_path + item;  
 32: $("head").append(s);  
 33: //$(document).append(s);  
 34: });  
 35: }  
 36:};

## .\mobile\www\js\datas.js

1:var datas = datas || {};  
 2:datas.pesquisa = [  
 3: '18/11/2017', '19/11/2017', '20/11/2017', '21/11/2017', '22/11/2017', '23/11/2017', '24/11/2017', '25/11/2017'  
 4:];  
 5:  
 6:/\*\*  
 7: \* Verifica se a data atual é um dia de pesquisa  
 8: \* @param {String} dataTeste  
 9: \* @returns {Boolean}  
 10: \*/  
 11:datas.verificaData = function(){  
 12: var r = false;  
 13: var dataTeste = util.formateDateOnly(new Date());  
 14: $.each(datas.pesquisa, function(index, value){  
 15: if (dataTeste == value){  
 16: r = true;  
 17: }  
 18: })  
 19: return r;  
 20:}

## .\mobile\www\js\eventQueue.js

1:eventQueue = {  
 2: \_fila : [],  
 3:  
 4: \_ocupado : false,  
 5:  
 6: initEventQueue : function() {  
 7: app.logger.log("Iniciando fila de eventos.");  
 8: var me = this;  
 9: // inicia o monitor  
 10: setInterval(function() {  
 11: eventQueue.\_monitoraFila();  
 12: }, 300);  
 13: app.logger.log("Fila de eventos iniciada.");  
 14: },  
 15:  
 16: \_monitoraFila : function() {  
 17: if (eventQueue.\_fila.length > 0 && !eventQueue.\_ocupado) {  
 18: eventQueue.\_ocupado = true;  
 19: eventQueue.\_consume(eventQueue.\_fila.shift());  
 20: eventQueue.\_ocupado = false;  
 21: }  
 22: },  
 23:  
 24: appendRow : function(event) {  
 25: if (!eventQueue.\_ocupado) {  
 26: app.logger.log("Inserindo evento na fila: " + event);  
 27: eventQueue.\_fila.push(event);  
 28: } else {  
 29: app.logger.log("Evento descartado da fila: " + event);  
 30: }  
 31: },  
 32:  
 33: \_consumeQueue : function(row) {  
 34: app.logger.log("Consumindo evento da fila: " + event);  
 35: // TODO  
 36: app.logger.log("Evento consumido da fila: " + event);  
 37: }  
 38:  
 39:};

## .\mobile\www\js\index.js

1:/\* global util, ipadID \*/  
 2:  
 3:/// @file index.js  
 4:/// @namespace app  
 5:/// Módulo com fluxo principal da aplicação  
 6:var app = {  
 7:  
 8: versao : '2.6.1',  
 9:   
 10: debugMode : false,  
 11:   
 12: versaoBD : 1,  
 13:  
 14: /// @function app.login  
 15: /// Identifica e autoriza o usuário  
 16: /// @return {void} função sem retorno  
 17: login : function() {  
 18: var usuario = $("#usuario").val().trim();  
 19: var senha = $("#senha").val().trim();  
 20: if(app.sentido === null){  
 21: alert("Selecione o sentido!");  
 22: return;  
 23: }  
 24:  
 25: // configura identificador do ipad, para executar uma única vez(durante a instalação)  
 26: if (logins.autenticaMaster(usuario, senha)) {  
 27: ipadID.requestID(function(id) {  
 28: $("#ipadID").html(id);  
 29: });  
 30: } else if (logins.autentica(usuario, senha)) {  
 31: app.logger.log("Login efetuado pelo usuário: " + usuario);  
 32: // navega para página e executa o script de configuração depois do carregamento  
 33: app.trocaPagina("views/menu.html", controllers.menu)  
 34: // set: user\_login, senha\_login, posto e sentido  
 35: app.user\_login = usuario;  
 36: app.senha\_login = senha;  
 37: app.posto = String(usuario).substr(0, 3);  
 38: //app.sentido = String(usuario).substr(3, 2).toUpperCase();  
 39: if (isNaN(app.posto)) { // apenas para efeitos ao usuário de testes  
 40: app.posto = '000';  
 41: }  
 42: if ((app.sentido != 'AB') && (app.sentido != 'BA')) { // apenas para efeitos ao usuário de testes  
 43: app.sentido = 'AB';  
 44: }  
 45: // limpa o registro  
 46: app.limpaRegistro();  
 47: }  
 48: },  
 49:  
 50: /// @function app.logout  
 51: /// Termina a utilização do aplicativo e desconecta o usuário  
 52: /// @return {void} função sem retorno  
 53: logout : function() {  
 54: // limpa o registro  
 55: app.limpaRegistro();  
 56: // realiza logout  
 57: $("#usuario").val('').textinput("refresh");  
 58: $("#senha").val('').textinput("refresh");  
 59: $("#agregador input[type='radio']").prop('checked', false);  
 60: $("#entrar").prop('disabled', true);  
 61: $(":mobile-pagecontainer").pagecontainer("change", $("#page\_login"));  
 62: app.logger.log('Logout');  
 63: app.user\_login = null;  
 64: app.senha\_login = null;  
 65: app.posto = null;  
 66: app.sentido = null;  
 67: app.restart(); //problably the only line we need  
 68: },  
 69:  
 70: exportaDbToJson : function() {  
 71: if (app.filePaths) {  
 72: resolveLocalFileSystemURL(app.filePaths.externalFolder, function(dir) {  
 73: var newJsonName = ipadID.id + "\_" + app.jsonName;  
 74: app.removeFile(newJsonName, app.filePaths.externalFolder, function() {  
 75: realExporter(newJsonName, dir);  
 76: }, function() {  
 77: realExporter(newJsonName, dir);  
 78: });  
 79: }, function(err) {  
 80: app.logger.log('ERRO ao acessar o folder externo ' + app.filePaths.externalFolder + ' ' + JSON.stringify(err));  
 81: });  
 82: } else {  
 83: alert('Operação não realizada, o sistema de arquivos não foi definido');  
 84: }  
 85:  
 86: function realExporter(jsonName, dir) {  
 87: dir.getFile(jsonName, {  
 88: create : true  
 89: }, function(file) {  
 90: app.logger.log("arquivo JSON: ", file);  
 91: jsonWriter.setJsonFile(file);  
 92: file.createWriter(function(fileWriter) {  
 93: jsonWriter.setJsonWriter(fileWriter);  
 94: myDb.exportaDbToJson(jsonWriter);  
 95: }, function() {  
 96: app.logger.log('ERRO criando o escritor do JSON a ser exportado.');  
 97: });  
 98: }, function(e) {  
 99: app.logger.log('ERRO ao criar o arquivo JSON a ser exportado: ' + e.message);  
 100: });  
 101: }  
 102: },  
 103:  
 104: duplicaDb : function() {  
 105: if (app.filePaths) {  
 106: //opens first. And tests the db's integrity  
 107: app.openDB(function(){  
 108: //on success, trys to close the db and exports the data  
 109: app.database.close(function() {  
 110: app.copyFile(app.dbName, app.filePaths.dbFolder, app.filePaths.externalFolder, false, function(newName) {  
 111: alert('Banco de dados ' + newName + ' exportado com sucesso.');  
 112: });  
 113: }, function(err){  
 114: //if app.database.close fails, exports the corrupted database  
 115: app.exportaDBCorrompido(err);  
 116: })  
 117: },   
 118: //if app.openDB fails, exports the corrupted database (this is the errorCalback)   
 119: //and inform that this is an export operation  
 120: app.exportaDBCorrompido, true  
 121: );   
 122:   
 123: // Exports Log file anyway  
 124: app.duplicaLog();  
 125: } else {  
 126: alert('Operação não realizada, o sistema de arquivos não foi definido');  
 127: }  
 128: },  
 129:   
 130: exportaDBCorrompido : function(err){  
 131: app.logger.log('Exportando bando de dados corrompido');  
 132: app.copyFile(app.dbName, app.filePaths.dbFolder, app.filePaths.externalFolder, true,   
 133: function(newName) {  
 134: alert('Banco de dados ' + newName + ' exportado!\n\nEntre em contato com o suporte.',  
 135: "Banco da Dados Corrompido", null, "info");  
 136: //apaga banco atual (corrompido)  
 137: app.removeFile(app.dbName, app.filePaths.dbFolder,  
 138: function(){ //removeu com sucesso  
 139: app.logger.log("Banco de dados corrompido removido do sistema de arquivos do app.");  
 140: }, function(){//falhou quando removia  
 141: app.logger.log("Falha removendo banco de dados corrompido");  
 142: alert('Houve uma falha grava no sistema. \n Continue o processo de exportação, entre em contato com o suporte e NÃO CONTINUE A USAR ESSE IPAD.',  
 143: "Erro no Banco da Dados", null, "error");  
 144: });  
 145:  
 146: });  
 147: if(err){  
 148: app.logger.log(JSON.stringify(err));  
 149: }  
 150: },  
 151:  
 152: duplicaLog : function() {  
 153: app.copyFile(app.logFileName, cordova.file.dataDirectory, app.filePaths.externalFolder, false, function(newName) {  
 154: alert('Arquivo de log ' + newName + ' exportado com sucesso.');  
 155: });  
 156: },  
 157:  
 158: /\*\*  
 159: \* Validates interview's cancellation  
 160: \*   
 161: \* @param Function  
 162: \* cb receives a Boolean representing the success in validation  
 163: \*/  
 164: validaCancelamento : function(cb) {  
 165: prompt("Insira a senha para cancelar a entrevista.", function(result) {  
 166: // botão prosseguir cancelamento  
 167: if (result == app.senha\_login) {  
 168: cb(true);  
 169: } else {  
 170: confirm("Senha incorreta para o cancelamento.\nDeseja tentar novamente?",  
 171: // ok button pressed  
 172: function() {  
 173: app.validaCancelamento(cb);  
 174:  
 175: }, function() {  
 176: cb(false);  
 177: }, "Senha incorreta.");  
 178:  
 179: }  
 180: }, "Cancelamento de entrevista", "Cancelar entrevista", "Voltar", "Senha", 'password');  
 181: },  
 182:  
 183: validaOperacoes : function(operacao, txtPrompt, titlePrompt, txtConfirm, titleConfirm, btnOKPrompt, btnCancelPrompt) {  
 184: prompt(txtPrompt, function(result) {  
 185: if (result == app.senha\_login) {  
 186: operacao();  
 187: } else {  
 188: confirm(txtConfirm,  
 189: function() {  
 190: app.validaOperacoes(operacao, txtPrompt, titlePrompt, txtConfirm, titleConfirm, btnOKPrompt,  
 191: btnCancelPrompt); // try again  
 192: }, null, titleConfirm);  
 193: }  
 194: }, titlePrompt, btnOKPrompt, btnCancelPrompt, "Senha", 'password');  
 195: },  
 196:  
 197: /\*  
 198: \* Application constructor  
 199: \*/  
 200: initialize : function() {  
 201: this.bindEvents();  
 202: this.extraConfig();  
 203: this.baseUrl = window.location.href.replace("index.html", "");  
 204: // configuring dialogs  
 205: window.alert = myDialogs.alert;  
 206: window.prompt = myDialogs.prompt;  
 207: window.confirm = myDialogs.confirm;  
 208:  
 209: },  
 210:  
 211: /\*  
 212: \* bind any events that are required on startup to listeners:  
 213: \*/  
 214: bindEvents : function() {  
 215: document.addEventListener('deviceready', app.onDeviceReady);  
 216: console.log('bindindEvents');  
 217: },  
 218:  
 219: /\*  
 220: \* this runs when the device is ready for user interaction:  
 221: \*/  
 222: onDeviceReady : function() {  
 223: app.logger.log('device ready');  
 224: window.addEventListener('batterycritical',function(status){  
 225: alert("A bateria atingiu nível crítco! \nCarregue o ipad imediatamente.\nNível de bateria: " + status.level + "%",  
 226: 'Alerta de Bateria', null, 'info');  
 227: }, false);  
 228: window.addEventListener('batterylow', function(status){  
 229: alert("O nível da bateria está baixo. \nCarregue o ipad assim que possível.\nNível de bateria: " + status.level + "%",  
 230: 'Alerta de Bateria', null, 'info');  
 231: }, false);  
 232:   
 233: document.addEventListener('pause', function(){  
 234: console.log("[" + (new Date()) + "]" + 'Ocorrência de evento: PAUSE');  
 235: }, false);  
 236:   
 237: document.addEventListener('resume', function(){  
 238: console.log("[" + (new Date()) + "]" + 'Ocorrência de evento: RESUME');  
 239: }, false);  
 240:   
 241: document.addEventListener('resign', function(){  
 242: console.log("[" + (new Date()) + "]" + 'Ocorrência de evento: RESIGN');  
 243: }, false);  
 244:  
 245: // a plataforma Browser não permite o desenvolvimento das escritas em arquivo  
 246: if (device.platform == 'iOS' || device.platform == 'Android') {  
 247: app.filePaths = {}; // inicia a variável  
 248:  
 249: setTimeout(function() {  
 250: navigator.splashscreen.hide();  
 251: console.log("esperando " + device.platform);  
 252: }, 3000);  
 253:  
 254:  
 255: // durante o debug, inicia o sistema de arquivos  
 256: if (app.debugMode){  
 257: navigator.notification.alert("App em modo debug. \nConecte o debugger", app.onFileSystemReady, 'Alerta de desenvolvimento', 'OK');  
 258: }else{  
 259: app.onFileSystemReady();  
 260: }   
 261: }  
 262:  
 263: // nessa altura do desenvolvimento não faz sentido  
 264: /\*  
 265: \* if (device.platform == 'browser' && device.model != 'Firefox') { alert('ATENÇÃO!!! \n Use o Firefox para fazer a  
 266: \* simulação (cordova run browser --target=firefox)'); }  
 267: \*/  
 268:  
 269: if (device.platform == 'iOS') {  
 270: // configurando a statusBar  
 271: StatusBar.overlaysWebView(false);  
 272: StatusBar.backgroundColorByName("black");  
 273: // black, darkGray, lightGray, white, gray, red, green, blue, cyan, yellow, magenta, orange, purple, brown  
 274:  
 275: // ajusta os caminhos para o iOS  
 276: app.filePaths['externalFolder'] = cordova.file.documentsDirectory;  
 277: app.filePaths['dbFolder'] = 'cdvfile://localhost/library/LocalDatabase';  
 278:  
 279: } else if (device.platform == 'Android') {  
 280: StatusBar.hide();  
 281: StatusBar.backgroundColorByName("darkGray");  
 282:  
 283: // ajusta os caminhos para o Android  
 284: app.filePaths['externalFolder'] = cordova.file.externalDataDirectory;  
 285: app.filePaths['dbFolder'] = cordova.file.applicationStorageDirectory + "databases";  
 286: }  
 287:  
 288: // alert sem a página como título  
 289: /\*  
 290: \* window.alert = function(txt, cb) { navigator.notification.alert(txt, cb, "Aviso", "Fechar"); }  
 291: \*/  
 292:  
 293: // novo valor para variavel criada no extraConfig()  
 294: app.uuid\_device = device.uuid;  
 295:  
 296: },  
 297:  
 298: onFileSystemReady : function() {  
 299:  
 300: app.logger.log("folder dos dados: ", cordova.file.dataDirectory);  
 301:  
 302: // setting ipadID file and value  
 303: window.resolveLocalFileSystemURL(cordova.file.dataDirectory,  
 304: // success resolving dir  
 305: function(dir) {  
 306: app.logger.log('file system ready: ', dir);  
 307: // modifica o atributo ipadId.id que havia sido configurado no extraConfig() como 'browser'  
 308: // reading ipadID file  
 309: dir.getFile(ipadID.ipadIDFileName, {  
 310: create : false  
 311: }, function(file) {  
 312: ipadID.readId(file, function(id) {  
 313: $("#ipadID").html(id);  
 314: afterIpadID(id, dir);  
 315: });  
 316: }, function(e) {  
 317: ipadID.requestID(function(id) {  
 318: $("#ipadID").html(id);  
 319: afterIpadID(id, dir);  
 320: });  
 321: });  
 322: },  
 323: // error resolving dir  
 324: function(err) {  
 325: app.logger.log("erro no sistema de arquivos: " + err.name + " -> " + err.message);  
 326: alert("erro no sistema de arquivos: " + err.name + " -> " + err.message);  
 327: });  
 328:  
 329: // internal function  
 330: function afterIpadID(id, dir) {  
 331:  
 332: dir.getFile(app.logFileName, {  
 333: create : true  
 334: },  
 335: // success getting file  
 336: function(file) {  
 337: app.logger.log("arquivo de log: ", file);  
 338: myLogger.setLogFile(file);  
 339: file.createWriter(function(fileWriter) {  
 340: myLogger.setLogWriter(fileWriter);  
 341: // novo valor para variável criada no extraConfig  
 342: app.logger = myLogger;  
 343: app.openDB(function(){// if successfully create fileWriter, open database for testing  
 344: app.logger.log("Conexão inicial com o banco para verificação");  
 345: app.database.close(function(){  
 346: app.logger.log("Conexão com o banco fechada");  
 347: });  
 348: });   
 349:   
 350: },  
 351: // error creating fileWriter  
 352: function() {  
 353: app.logger.log('erro criando o escritor do log');  
 354: });  
 355: });  
 356: }  
 357: ;  
 358:  
 359: },  
 360: /\*\*  
 361: \* Creates and open database connection  
 362: \* @param cb - option callback function  
 363: \* @param errorCB - optional callback function for when an error occurs  
 364: \* @param isExporting - optional boolean to identify data exporting operations   
 365: \*/  
 366: openDB : function(cb, errorCB, isExporting) {  
 367: app.database = sqlitePlugin.openDatabase({  
 368: name : app.dbName,  
 369: iosDatabaseLocation : 'default'  
 370: },  
 371: // sucsess  
 372: function() {  
 373: app.logger.log('Conexão com o banco de dados criada com sucesso.');  
 374: myDb.cretateTblDados(function(){  
 375: myDb.sanitize(function(){  
 376: myDb.updateSchema(function(){  
 377: if(util.isFunction(cb)){  
 378: cb();  
 379: }   
 380: });  
 381: });  
 382: });   
 383: },  
 384: // fail  
 385: function(err) {  
 386: app.logger.log('ERRO ao tentar conectar com o banco de dados.');  
 387: app.logger.log(JSON.stringify(err));  
 388: if (!isExporting){  
 389: alert("O Banco da dados foi corrompido. \n Exporte todos os dados e entre em contato com o suporte.",  
 390: "Erro no Banco da Dados", null, "error");  
 391: }  
 392: if(util.isFunction(errorCB)){  
 393: errorCB();  
 394: }  
 395: });  
 396: },  
 397:   
 398: extraConfig : function() {  
 399: // initialize panel  
 400: $(function() {  
 401: $("[data-role=panel]").panel().enhanceWithin();  
 402: });  
 403: $('#dataHoraLogin').html(util.formatDateTimeToDisplay());  
 404: var updater\_dataTimeToDisplay = setInterval(function(){  
 405: $('#dataHoraLogin').html(util.formatDateTimeToDisplay());  
 406: },10000);  
 407:   
 408: $("#versao").html(app.versao);  
 409: $("#entrar").click(function(){  
 410: clearInterval(updater\_dataTimeToDisplay);  
 411: app.login();  
 412: });  
 413: $("#entrar").prop('disabled', true);  
 414: $("#agregador input[type='radio']").prop('checked', false).change(function(event, ui){  
 415: $("#entrar").prop('disabled', false);  
 416: app.sentido = $(this).val();  
 417: });  
 418:  
 419: // valores iniciais (vão ficar assim se estiver usando o browser)  
 420: app.uuid\_device = "browser";  
 421:   
 422: ipadID.id = 'browser';  
 423:   
 424: // botões do menu  
 425: $("#btn\_sair").click(  
 426: function() {  
 427: app.validaOperacoes(app.logout, "Insira a senha para realizar o logout.", "Logout"  
 428:, "Senha incorreta.\nDeseja tentar novamente?", "Senha Incorreta", "Logout", "Voltar");  
 429: });  
 430:   
 431: /\*  
 432: \* A partir da Fase 2 o log passou a ser sempre exportado junto ao DB  
 433: $("#duplica\_log").click(  
 434: function() {  
 435: app.validaOperacoes(app.duplicaLog, "Insira a senha para exportar o Log de operações.",  
 436: "Exportar log de operações", "Senha incorreta para a exportação.\nDeseja tentar novamente?",  
 437: "Senha Incorreta", "Exportar", "Voltar");  
 438: });  
 439: \*/  
 440: $("#duplica\_db").click(  
 441: function() {  
 442: app.validaOperacoes(app.duplicaDb, "Insira a senha para exportar o banco de dados.",  
 443: "Exportar banco de dados", "Senha incorreta para a exportação.\nDeseja tentar novamente?",  
 444: "Senha Incorreta", "Exportar", "Voltar");  
 445: });  
 446:   
 447: /\*  
 448: \* Funçao retirada a partir da fase 2 do projeto  
 449: $("#exporta\_db\_to\_json").click(  
 450: function() {  
 451: app.validaOperacoes(app.exportaDbToJson, "Insira a senha para exportar os dados no formato JSON.",  
 452: "Exportar JSON", "Senha incorreta para a exportação.\nDeseja tentar novamente?", "Senha Incorreta",  
 453: "Exportar", "Voltar");  
 454: });  
 455: \*/  
 456: // remove o filtro original do autocomplete para poder filtrar acentos  
 457: $.mobile.filterable.prototype.options.filterCallback = function(index, value) {  
 458: return false  
 459: };  
 460:   
 461: //configurando o spinner  
 462: $.mobile.loader.prototype.options.text = "Aguarde";  
 463: $.mobile.loader.prototype.options.textVisible = true;  
 464: $.mobile.loader.prototype.options.theme = "b";  
 465: $.mobile.loader.prototype.options.html = "";  
 466: },  
 467: /\*\*  
 468: \* Change pages within app  
 469: \* @param view -> new view  
 470: \* @param controller -> controller that will run on view change  
 471: \* @param String changeFunction -> ['old', 'new'] -- default new  
 472: \*   
 473: \*/  
 474: trocaPagina : function(view, controller, changeFunction) {  
 475:   
 476: var changeF = (util.isEmpty(changeFunction) || changeFunction != 'old')? newChange : oldChange;  
 477: var abortOperation = false;  
 478:   
 479: if (view == null) {  
 480: app.logger.log("[ERRO] trocaPagina: view null");  
 481: abortOperation = true;  
 482: }   
 483:   
 484: if (controller == null){  
 485: app.logger.log("[ERRO] trocaPagina: controller null");  
 486: abortOperation = true;  
 487: }  
 488:   
 489: if (abortOperation){  
 490: myDialogs.alert("Ocorreu um erro interno na aplicação. A entrevista será cancelada.",  
 491: "Erro interno da aplicação.",  
 492: function(){  
 493: app.cancelaRegistro();  
 494: app.trocaPagina('views/menu.html', controllers.menu);  
 495: }, 'error');  
 496: } else {  
 497: try {  
 498: app.onChangeHandler.controller = controller.config;  
 499: $(":mobile-pagecontainer").off("pagecontainershow", app.onChangeHandler.handler).on("pagecontainershow",  
 500: app.onChangeHandler.handler);  
 501: } catch(exc) {  
 502: app.logger.log("[ERRO] trocaPagina: excecao no changeHandler. Detalhes: ")  
 503: app.logger.log(exc.message)  
 504: }  
 505:  
 506: try {  
 507: changeF(view);  
 508: } catch(exc) {  
 509: app.logger.log("[ERRO] trocaPagina: excecao ao chamar pagecontainer. Detalhes: ");  
 510: app.logger.log(exc.message);  
 511: }  
 512:   
 513: app.logger.log(view);  
 514: app.logger.log('número de mudanças de página: ' + ++app.changesCounter);  
 515:   
 516: }  
 517:   
 518: function oldChange(v){  
 519: $(":mobile-pagecontainer").pagecontainer("change", app.baseUrl + v);  
 520: app.logger.log('oldTrocaPagina');  
 521: }  
 522:   
 523: function newChange(v){  
 524: $(":mobile-pagecontainer").pagecontainer("change", app.baseUrl + v, {reload : true, changeHash : false, transition : 'none'});  
 525:// app.logger.log('newTrocaPagina');  
 526: }  
 527: },  
 528:  
 529: // para apendar coisas aos controllers  
 530: onChangeHandler : {  
 531: handler : function() {  
 532: try {  
 533:// $("section[data-role=page]").append(  
 534:// "<footer data-role='footer' data-position='fixed'> \  
 535:// <span class='versao'>Versão: <span id='versao'></span></span> \  
 536:// <span class='data\_atual'>Data: <span id='data\_atual'></span></span> \  
 537:// <span class='posto'>Posto: <span id='posto'></span></span> \  
 538:// <span class='sentido'>Sentido: <span id='sentido'></span></span> \  
 539:// <span class='ipad'>id-iPad: <span id='ipadID'></span></span> \  
 540:// </footer>"  
 541:// );  
 542:// $("footer[data-role='footer']").toolbar({  
 543:// defaults : true,  
 544:// position: "fixed"  
 545:// });  
 546:   
 547: app.onChangeHandler.controller();  
 548:  
 549: // Botão "Cancelar"  
 550: $("#btn\_cancelar").click(app.cancelar);  
 551:  
 552: // Input radio  
 553: $('input[type="radio"]').click(function() {  
 554: $(this).focus();  
 555: });  
 556:  
 557: // Input integer  
 558: $('input[typeMask="integer"]').each(function(key, input) {  
 559: var minValue = 0;  
 560: var maxValue = 99999999;  
 561: if (!util.isEmpty($(input).attr('min'))) {  
 562: minValue = Number($(input).attr('min'));  
 563: }  
 564: if (!util.isEmpty($(input).attr('max'))) {  
 565: maxValue = Number($(input).attr('max'));  
 566: }  
 567: $(input).inputmask('integer', {  
 568: min : minValue,  
 569: max : maxValue  
 570: });  
 571: });  
 572:  
 573: // Input money  
 574: $('input[typeMask="money-BRL"]').each(function(key, input) {  
 575: $(input).maskMoney({  
 576: prefix : "R$ ",  
 577: affixesStay : "true",  
 578: thousands : ".",  
 579: decimal : ","  
 580: });  
 581: });  
 582:  
 583: $('input[mask]').each(function(key, input) {  
 584: var decimalSeparator = $(input).attr('mask-decimal-separator');  
 585: if (util.isEmpty(decimalSeparator)) {  
 586: $(input).inputmask($(input).attr('mask'), {  
 587: 'autoUnmask' : true  
 588: });  
 589: } else {  
 590: $(input).inputmask($(input).attr('mask'), {  
 591: 'autoUnmask' : true,  
 592: numericInput : true,  
 593: radixPoint : decimalSeparator  
 594: });  
 595: }  
 596: });  
 597:   
 598:   
 599: $(".versao #versao").html(app.versao);  
 600: $(".data\_atual #data\_atual").html(util.formateDateOnly(new Date()));  
 601: $(".ipad #ipadID").html(ipadID.id);  
 602: $(".posto #posto").html(app.posto);  
 603: $(".sentido #sentido").html(app.sentido);  
 604:  
 605: if (typeof device != 'undefined' && device.platform == "Android") {  
 606: StatusBar.hide();  
 607: }  
 608: //removendo outras páginas do DOM  
 609: $('div[data-role=page]:hidden').remove();  
 610: } catch (e) {  
 611: app.logger.log(e);  
 612: alert(e.message);  
 613: }  
 614: },  
 615:  
 616: controller : null,  
 617: },  
 618:  
 619: getAtributo : function(nome) {  
 620: return registro[nome];  
 621: },  
 622:  
 623: setAtributo : function(nome, valor) {  
 624: registro[nome] = valor;  
 625: app.logger.log(JSON.stringify(registro));  
 626: },  
 627:  
 628: splitAtributo : function(nome) {  
 629: var valor = registro[nome];  
 630: if (!util.isEmpty(valor)) {  
 631: var splited = String(valor).split("|");  
 632: if (splited.length > 1) {  
 633: valor = splited[splited.length - 1].trim();  
 634: if (!util.isEmpty(valor)) {  
 635: if (isNaN(valor)) {  
 636: app.setAtributo(nome, valor);  
 637: } else {  
 638: app.setAtributo(nome, parseInt(valor));  
 639: }  
 640: }  
 641: }  
 642: }  
 643: },  
 644:  
 645: cancelar : function() {  
 646: app.validaCancelamento(function(result) {  
 647: if (result) {  
 648: app.cancelaRegistro(function() {  
 649: app.trocaPagina('views/menu.html', controllers.menu);  
 650: });  
 651: }  
 652: });  
 653: },  
 654:  
 655: limpaRegistro : function() {  
 656: app.logger.log('Limpando registro', true);  
 657: registro = {};  
 658: },  
 659:  
 660: iniciaRegistro : function() {  
 661: try {  
 662: var now = new Date();  
 663: app.logger.log('Iniciando registro');  
 664: app.limpaRegistro();  
 665: app.setAtributo('id', ipadID.id + util.getTimeUnixTimestamp(now));  
 666: app.setAtributo('login', app.user\_login);  
 667: app.setAtributo('idPosto', app.posto);  
 668: app.setAtributo('sentido', app.sentido);  
 669: app.setAtributo('uuid', app.uuid\_device);  
 670: app.setAtributo('dataIniPesq', util.getTimeDefaultFormated(now));  
 671: app.setAtributo('dataIniPesqAbsolute', util.getTimeUnixTimestamp(now));  
 672: app.setAtributo('idIpad', ipadID.id);  
 673: app.logger.log(JSON.stringify(registro));  
 674: app.logger.log('Registro iniciado: ' + registro.id);  
 675: } catch (e) {  
 676: app.logger.log(e.message);  
 677: }  
 678: },  
 679:  
 680: setCamposDerivados : function() {  
 681: try {  
 682: //Treinamento  
 683: if (app.isTreinamento){  
 684: app.setAtributo('treinamento', 1);  
 685: }else{  
 686: app.setAtributo('treinamento', 0);   
 687: }  
 688:   
 689:   
 690: // TIPO VEICULO  
 691: if (!util.isEmpty(app.getAtributo('tipo'))) {  
 692: var tipoReal = app.getAtributo('tipo').split("\_")[0];  
 693: app.setAtributo('tipo', tipoReal.toUpperCase());  
 694: } else if (registro.cancelado != 1) {  
 695: app.setAtributo('erro', "ERRO (tipo vazio)");  
 696: app.logger.log(registro.erro + " no registro: ", registro.id);  
 697: }  
 698:   
 699: // PLACA  
 700: if ((!util.isEmpty(registro.placaEstrangeira)) && (!registro.placaEstrangeira)) {  
 701: if ((!util.isEmpty(registro.placa\_letras)) && (!util.isEmpty(registro.placa\_numeros))) {  
 702: app.setAtributo('placa', registro.placa\_letras.toUpperCase() + "-" + registro.placa\_numeros);  
 703: }  
 704: } else if ((!util.isEmpty(registro.placaEstrangeira)) && registro.placaEstrangeira) {  
 705: if (!util.isEmpty(registro.placa\_unica)) {  
 706: app.setAtributo('placa', registro.placa\_unica.toUpperCase());  
 707: }  
 708: }  
 709: if (util.isEmpty(registro.placa)) {  
 710: if (registro.cancelado != 1) {  
 711: app.setAtributo('erro', "ERRO (placa vazio)");  
 712: app.logger.log(registro.erro + " no registro: ", registro.id);  
 713: }  
 714: }  
 715:   
 716: // TIPO DE COMBUSTÍVEL  
 717: if (util.isEmpty(registro.idCombustivel) && (!util.isEmpty(registro.categoria))  
 718: && (registro.categoria == 'Onibus' || registro.categoria == 'Pesado')) {  
 719: app.setAtributo('idCombustivel', util.getIdFromTabelaAuxiliar('Diesel', lista\_combustivel));  
 720: }  
 721: if (registro.cancelado != 1) {  
 722: if (util.isEmpty(registro.categoria)) {  
 723: app.setAtributo('erro', "ERRO (categoria vazio)");  
 724: app.logger.log(registro.erro + " no registro: ", registro.id);  
 725: }  
 726: if (util.isEmpty(registro.idCombustivel)) {  
 727: app.setAtributo('erro', "ERRO (idCombustivel vazio)");  
 728: app.logger.log(registro.erro + " no registro: ", registro.id);  
 729: }  
 730: }  
 731:   
 732: // FRENQUENCIA  
 733: if (!util.isEmpty(registro.frequenciaPeriodo)) {  
 734: app.setAtributo('frequenciaPeriodo', util.getListaFrequencias()[Number(registro.frequenciaPeriodo) - 1]);  
 735: if (registro.frequenciaPeriodo == 'Eventualmente') {  
 736: app.setAtributo('frequenciaQtd', 1);  
 737: } else if (util.isEmpty(registro.frequenciaQtd) || (Number(registro.frequenciaQtd) <= 0)) {  
 738: if (registro.cancelado != 1) {  
 739: app.setAtributo('erro', "ERRO (frequenciaQtd vazio)");  
 740: app.logger.log(registro.erro + " no registro: ", registro.id);  
 741: }  
 742: }  
 743: } else if (registro.cancelado != 1) {  
 744: app.setAtributo('erro', "ERRO (frequenciaPeriodo vazio)");  
 745: app.logger.log(registro.erro + " no registro: ", registro.id);  
 746: }  
 747:   
 748: // ORIGEM: MUNICÍPIO  
 749: app.splitAtributo('idOrigemMunicipio');  
 750: if (util.isEmpty(registro.idOrigemMunicipio) && (registro.idOrigemPais == 1)) { // Brasil  
 751: if (registro.cancelado != 1) {  
 752: app.setAtributo('erro', "ERRO (idOrigemMunicipio vazio)");  
 753: app.logger.log(registro.erro + " no registro: ", registro.id);  
 754: }  
 755: }  
 756:   
 757: // DESTINO: MUNICÍPIO  
 758: app.splitAtributo('idDestinoMunicipio');  
 759: if (util.isEmpty(registro.idDestinoMunicipio) && (registro.idDestinoPais == 1)) { // Brasil  
 760: if (registro.cancelado != 1) {  
 761: app.setAtributo('erro', "ERRO (idDestinoMunicipio vazio)");  
 762: app.logger.log(registro.erro + " no registro: ", registro.id);  
 763: }  
 764: }  
 765:   
 766: // ESPECÍFICOS PARA FLUXO DE CARGA  
 767: if (app.getAtributo('classeVeiculo') == 'carga') {  
 768:   
 769: // RNTRC  
 770: if ((!util.isEmpty(registro.placaVermelha)) && registro.placaVermelha) {  
 771: if (!util.isEmpty(registro.placa\_vermelha\_rntrc\_sel) && !util.isEmpty(registro.placa\_vermelha\_rntrc\_num)) {  
 772: app.setAtributo('rntrc', String(registro.placa\_vermelha\_rntrc\_sel)  
 773: + String(registro.placa\_vermelha\_rntrc\_num));  
 774: } else {  
 775: if (registro.cancelado != 1) {  
 776: app.setAtributo('erro', "ERRO (rntrc vazio)");  
 777: app.logger.log(registro.erro + " no registro: ", registro.id);  
 778: }  
 779: }  
 780: }  
 781:   
 782: // RISCO E ONU  
 783: if (registro.possuiCargaPerigosa) {  
 784: var idNumRisco = util.findValueInList(registro.idNumeroDeRisco, lista\_numero\_risco);  
 785: if (!util.isEmpty(idNumRisco)) {  
 786: app.setAtributo('idNumeroDeRisco',idNumRisco);  
 787: } else {  
 788: if (registro.cancelado != 1) {  
 789: app.setAtributo('erro', "ERRO (idNumeroDeRisco não encontrado)");  
 790: app.logger.log(registro.erro + " no registro: ", registro.id);  
 791: }  
 792: }  
 793:  
 794: var idNumOnu = util.findValueInList(registro.idNumeroDaOnu, lista\_numero\_onu);  
 795: if (!util.isEmpty(idNumOnu)) {  
 796: app.setAtributo('idNumeroDaOnu',idNumOnu);  
 797: } else {  
 798: if (registro.cancelado != 1) {  
 799: app.setAtributo('erro', "ERRO (idNumeroDaOnu não encontrado)");  
 800: app.logger.log(registro.erro + " no registro: ", registro.id);  
 801: }  
 802: }  
 803: }  
 804:   
 805: //POSSUI CARGA  
 806: if (registro.possui\_carga) {  
 807: // TIPO DE PRODUTO (CARGA)  
 808: app.splitAtributo('idProduto');  
 809: if (util.isEmpty(registro.idProduto) && (registro.cancelado != 1)) {  
 810: app.setAtributo('erro', "ERRO (idProduto vazio)");  
 811: app.logger.log(registro.erro + " no registro: ", registro.id);  
 812: }  
 813:  
 814: // PESO DA CARGA ('ton' -> 'kg')  
 815: if (!util.isEmpty(registro.pesoDaCarga) && ((typeof registro.pesoDaCarga) != 'number')) {  
 816: var peso = Number(registro.pesoDaCarga);  
 817: if (registro.unidadePesoDaCarga == 'kg') {  
 818: peso = peso / 10;  
 819: } else {  
 820: peso = peso \* 100;  
 821: }  
 822: app.setAtributo('pesoDaCarga', peso);  
 823: }  
 824: if (util.isEmpty(registro.pesoDaCarga) && (registro.cancelado != 1)) {  
 825: app.setAtributo('erro', "ERRO (pesoDaCarga vazio)");  
 826: app.logger.log(registro.erro + " no registro: ", registro.id);  
 827: }  
 828:   
 829: // MUNICÍPIO EMBARQUE DA CARGA  
 830: app.splitAtributo('municipioEmbarqueCarga');  
 831: if (util.isEmpty(registro.municipioEmbarqueCarga) && registro.sabe\_embarque && (registro.cancelado != 1)) {  
 832: app.setAtributo('erro', "ERRO (municipioEmbarqueCarga vazio)");  
 833: app.logger.log(registro.erro + " no registro: ", registro.id);  
 834: }  
 835:   
 836: // MUNICÍPIO DESEMBARQUE DA CARGA  
 837: app.splitAtributo('municipioDesembarqueCarga');  
 838: if (util.isEmpty(registro.municipioDesembarqueCarga) && registro.sabe\_desembarque && (registro.cancelado != 1)) {  
 839: app.setAtributo('erro', "ERRO (municipioDesembarqueCarga vazio)");  
 840: app.logger.log(registro.erro + " no registro: ", registro.id);  
 841: }  
 842: } else {  
 843: // NAO POSSUI CARGA  
 844: app.setAtributo('idProduto', 3000); // Produto VAZIO  
 845: app.setAtributo('pesoDaCarga', null);  
 846: app.setAtributo('unidadePesoDaCarga', null);  
 847: app.setAtributo('valorDoFrete', null);  
 848: app.setAtributo('valorDaCarga', null);  
 849: app.setAtributo('embarqueCargaNaoSabe', null);  
 850: app.setAtributo('embarque\_uf', null);  
 851: app.setAtributo('municipioEmbarqueCarga', null);  
 852: app.setAtributo('idLocalEmbarqueCarga', null);  
 853: app.setAtributo('desembarqueCargaNaoSabe', null);  
 854: app.setAtributo('desembarque\_uf', null);  
 855: app.setAtributo('municipioDesembarqueCarga', null);  
 856: app.setAtributo('idLocalDesembarqueCarga', null);  
 857:  
 858: // POSSUI CARGA ANTERIOR  
 859: if (registro.carga\_anterior) {  
 860: app.splitAtributo('idCargaAnterior');  
 861: if (util.isEmpty(registro.idCargaAnterior) && registro.carga\_anterior && (registro.cancelado != 1)) {  
 862: app.setAtributo('erro', "ERRO (idCargaAnterior vazio)");  
 863: app.logger.log(registro.erro + " no registro: ", registro.id);  
 864: }  
 865: } else {  
 866: app.setAtributo('idCargaAnterior', 3000); // Produto VAZIO  
 867: }  
 868: }  
 869:   
 870: // CARGA SUGESTÃO PARADA OBRIGATÓRIA MUNICÍPIOS  
 871: app.splitAtributo('paradaObrigatoriaMunicipio1');  
 872: app.splitAtributo('paradaObrigatoriaMunicipio2');  
 873:   
 874: if (util.isEmpty(registro.paradaObrigatoriaMunicipio1) && util.isEmpty(registro.paradaObrigatoriaMunicipio1)) {  
 875: if ((!registro.municipiosParadaNaoSabe) && (registro.cancelado != 1)) {  
 876: app.setAtributo('erro', "ERRO (paradaObrigatoriaMunicipio1 ou paradaObrigatoriaMunicipio2 vazio)");  
 877: app.logger.log(registro.erro + " no registro: ", registro.id);  
 878: }  
 879: }  
 880: }  
 881: }  
 882: catch(exc) {  
 883: app.logger.log("[ERRO] Excecao em setCamposDerivados; Detalhes: ");  
 884: app.logger.log(exc);  
 885: }  
 886:   
 887: var now = new Date();  
 888: app.setAtributo('dataFimPesq', util.getTimeDefaultFormated(now));  
 889: var duracao = util.getTimeUnixTimestamp(now) - app.getAtributo('dataIniPesqAbsolute');  
 890: app.setAtributo('duracaoPesq', duracao);  
 891: },  
 892:  
 893: finalizaRegistro : function(cb) {  
 894: app.logger.log('Finalizando registro: ' + registro.id);  
 895: app.setAtributo('cancelado', 0);  
 896: app.setCamposDerivados();  
 897: app.setAtributo('finalizado', 1); //para facilitar a recuperação do registro final pelo log  
 898: $.mobile.loading("show");  
 899: app.openDB(function(){  
 900: app.inserirRegistro(function(){  
 901: $.mobile.loading('hide');  
 902: app.database.close(function(){  
 903: app.logger.log("Conexão com o banco fechada");  
 904: cb();  
 905: });//fechando a conexão com o banco depois de inserir o registro;  
 906:   
 907: });  
 908: });  
 909: },  
 910:   
 911: inserirRegistro : function(cb){ //função é chamada por finalizaRegistro  
 912: myDb.insertRegistro(registro,  
 913: // erro  
 914: function(error) {  
 915: app.logger.log('Erro ao inserir registro: ' + error.message);  
 916: // confirma se tenta outra vez  
 917: confirm("Houve uma falha ao inserir o registro.\nDeseja tentar novamente?",  
 918: // button ok  
 919: function() {  
 920: app.inserirRegistro(cb);  
 921: },  
 922: // button cancel  
 923: function() {  
 924: app.logger.log("Descartando registro: " + registro.id);  
 925: if (cb != null) {  
 926: cb();  
 927: }  
 928: }, "Falha na gravação.", // título  
 929: "Sim", "Não (irá descartar o registro)");  
 930: },  
 931: // ok  
 932: function() {  
 933: app.logger.log('Registro finalizado: ' + registro.id);  
 934: app.limpaRegistro();  
 935:   
 936: alert("Entrevista registrada.");  
 937: if (util.isFunction(cb)) {  
 938: cb();  
 939: }  
 940: });  
 941: },  
 942:   
 943: buscaDuracoesRegistros : function(cb) {  
 944: app.openDB(function(){  
 945: myDb.selectDuracoesDiaRegistro(  
 946: // erro  
 947: function(error) {  
 948: app.logger.log('Erro ao buscar duracoes: ' + error.message);  
 949: // confirma se tenta outra vez  
 950: confirm("Houve uma falha ao buscar os registros.\nDeseja tentar novamente?",  
 951: // button ok  
 952: function() {  
 953: app.buscaDuracoesRegistros();  
 954: },  
 955: // button cancel  
 956: function() {  
 957: app.logger.log("buscaDuracoesRegistros cancelado");  
 958: app.database.close(function(){  
 959: app.logger.log('conexão com o banco fechada');  
 960: });  
 961: }, "Falha na busca.", // título  
 962: "Sim", "Não");  
 963: },  
 964: // ok  
 965: function() {  
 966: app.logger.log('buscaDuracoesRegistros executado com sucesso');  
 967: app.database.close(function(){  
 968: app.logger.log('conexão com o banco fechada');  
 969: if(util.isFunction(cb)){  
 970: cb();  
 971: }  
 972: });  
 973:   
 974: });  
 975: });  
 976:   
 977: },  
 978:  
 979: buscaUltimaPesquisa : function(cb) {  
 980: app.openDB(function(){  
 981: myDb.selectUltimaPesquisaValida(  
 982: // erro  
 983: function(error) {  
 984: app.logger.log('Erro ao buscar última pesquisa: ' + error.message);  
 985: // confirma se tenta outra vez  
 986: confirm("Houve uma falha ao buscar a última pesquisa realizada.\nDeseja tentar novamente?",  
 987: // button ok  
 988: function() {  
 989: app.buscaUltimaPesquisa();  
 990: },  
 991: // button cancel  
 992: function() {  
 993: app.logger.log("buscaUltimaPesquisa cancelado");  
 994: app.database.close(function(){  
 995: app.logger.log('conexão com o banco fechada');  
 996: });  
 997: }, "Falha na busca.", // título  
 998: "Sim", "Não");  
 999: },  
1000: // ok  
1001: function() {  
1002: app.logger.log('buscaUltimaPesquisa executado com sucesso');  
1003: app.database.close(function(){  
1004: app.logger.log('Conexão com o banco fechada.');  
1005: if(util.isFunction(cb)){  
1006: cb();  
1007: }  
1008: });  
1009: });  
1010: });  
1011:   
1012: },  
1013:  
1014: buscaRegistrosCancelados : function(cb) {  
1015: app.openDB(function(){  
1016: myDb.selectRegistrosCancelados(  
1017: // erro  
1018: function(error) {  
1019: app.logger.log('Erro ao buscar cancelados: ' + error.message);  
1020: // confirma se tenta outra vez  
1021: confirm("Houve uma falha ao buscar os registros cancelados.\nDeseja tentar novamente?",  
1022: // button ok  
1023: function() {  
1024: app.buscaRegistrosCancelados();  
1025: },  
1026: // button cancel  
1027: function() {  
1028: app.logger.log("buscaRegistrosCancelados cancelado");  
1029: app.database.close(function(){  
1030: app.logger.log('conexão com o banco fechada');  
1031: });  
1032: }, "Falha na busca.", // título  
1033: "Sim", "Não");  
1034: },  
1035: // ok  
1036: function() {  
1037: app.logger.log('buscaRegistrosCancelados executado com sucesso');  
1038: app.database.close(function(){  
1039: app.logger.log('Conexão com o banco fechada');  
1040: if(util.isFunction(cb)){  
1041: cb();  
1042: }  
1043: });  
1044:   
1045: });   
1046: });  
1047:   
1048: },  
1049:  
1050: cancelaRegistro : function(cb) {  
1051: app.logger.log('Cancelando registro: ' + registro.id);  
1052: app.setAtributo('cancelado', 1);  
1053: app.setCamposDerivados();  
1054: app.openDB(function(){  
1055: myDb.insertRegistro(registro, function(error) {  
1056: app.logger.log(error.message);  
1057: // confirma se tenta outra vez  
1058: confirm("Houve uma falha ao registrar o cancelamento.\nDeseja tentar novamente?",  
1059: // button ok  
1060: function() {  
1061: app.cancelaRegistro(cb);  
1062: },  
1063: // button cancel  
1064: function() {  
1065: app.logger.log("Descartando registro: " + registro.id);  
1066: if (cb != null) {  
1067: cb();  
1068: }  
1069:  
1070: }, "Falha na gravação.", // título  
1071: "Sim", "Não (irá descartar o registro)");  
1072: }, function() {  
1073: app.logger.log('Registro cancelado: ' + registro.id);  
1074: app.limpaRegistro();  
1075: app.database.close(function(){  
1076: app.logger.log("Conexão com o banco fechada.");  
1077: });  
1078: alert("Entrevista cancelada.");  
1079: if (cb != null) {  
1080: cb();  
1081: }  
1082: });  
1083: });  
1084:   
1085: },  
1086:  
1087: /\*\*  
1088: \* Remove a file from fs, if it exists  
1089: \*   
1090: \* @param String  
1091: \* fileName  
1092: \* @param String  
1093: \* dir -> prefers cordova.file.{dir}  
1094: \* @param function  
1095: \* cbSuccess success callback  
1096: \* @param function  
1097: \* cbFail error callback  
1098: \*/  
1099: removeFile : function(fileName, dirURI, cbSuccess, cbFail) {  
1100: window.resolveLocalFileSystemURL(dirURI,  
1101: // success  
1102: function(folder) {  
1103: app.logger.log('Removendo arquivo ' + fileName + ' do folder ' + JSON.stringify(folder));  
1104: folder.getFile(fileName, {  
1105: create : false  
1106: }, function(arquivo) {  
1107: arquivo.remove(function() {  
1108: app.logger.log('Arquivo ' + fileName + ' removido');  
1109: if (util.isFunction(cbSuccess)) {  
1110: cbSuccess();  
1111: }  
1112: });  
1113: }, function(err) {  
1114: app.logger.log('Arquivo ' + fileName + ' não existe no folder ' + JSON.stringify(folder));  
1115: if (util.isFunction(cbFail)) {  
1116: cbFail();  
1117: }  
1118:  
1119: });  
1120: },  
1121: // fail  
1122: function(err) {  
1123: app.logger.log('Erro resolvendo o diretório ' + dir + JSON.stringify(err));  
1124: });  
1125: },  
1126:  
1127: /\*\*  
1128: \* Copy a file overwriting the destination file, if it exists  
1129: \*   
1130: \* @param String  
1131: \* fileName  
1132: \* @param String  
1133: \* originDir  
1134: \* @param String  
1135: \* destDir  
1136: \* @param Boolean  
1137: \* isCorrupted  
1138: \* @param Function  
1139: \* cb success callback function  
1140: \*/  
1141: copyFile : function(fileName, originDirURI, destDirURI, isCorrupted, cb) {  
1142: var now = new Date();  
1143: var newName = (isCorrupted)? "CORROMPIDO" + ipadID.id + "\_" + fileName : ipadID.id + "\_" + fileName;  
1144: var extension = "";  
1145: if(newName.lastIndexOf(".txt") > -1) {  
1146: newName = newName.replace(".txt","");  
1147: extension = ".txt";  
1148: }  
1149: else if (newName.lastIndexOf(".db") > -1) {  
1150: newName = newName.replace(".db","");  
1151: extension = ".db";  
1152: }  
1153: newName = newName + "\_" + util.getTimeToExportFormated(now) + extension;  
1154: // get original dir  
1155: resolveLocalFileSystemURL(originDirURI,  
1156: // success  
1157: function(dir) {  
1158: // get original file  
1159: dir.getFile(fileName, {},  
1160: // success getting original file  
1161: function(file) {  
1162: // resolving destination  
1163: resolveLocalFileSystemURL(destDirURI,  
1164: // success resolving destination  
1165: function(destDir) {  
1166: app.logger.log('Copiando o arquivo ' + newName + ' do folder ' + dir.nativeURL + ' para o folder '  
1167: + destDir.nativeURL);  
1168: //copying new file  
1169: realCopier(file, newName, destDir);  
1170:   
1171: // Deixou de sobrescrever arquivo a partir da Fase 2  
1172: /\*  
1173: // removing destination file  
1174: app.removeFile(newName, destDirURI, function() {  
1175: realCopier(file, newName, destDir);  
1176: },  
1177: // mesmo se não conseguir remover  
1178: function() {   
1179: realCopier(file, newName, destDir);  
1180: });  
1181: \*/  
1182: }, function(err) {  
1183: app.logger.log('Erro acessando o folder de destino ' + destDir + ' ' + JSON.stringify(err));  
1184: });  
1185:  
1186: }, function(err) {  
1187: app.logger.log('Erro acessando arquivo original ' + fileName + ' ' + JSON.stringify(err));  
1188: });  
1189:  
1190: }, function(err) {  
1191: app.logger.log('Erro acessando o folder original ' + originDirURI + ' ' + JSON.stringify(err));  
1192: });  
1193: // internal function  
1194: function realCopier(f, newName, d) {  
1195: f.copyTo(d, newName, function() {  
1196: // success  
1197: app.logger.log('Arquivo ' + newName + ' copiado.');  
1198: if (util.isFunction(cb)) {  
1199: cb(newName);  
1200: }  
1201: }, function(err) {  
1202: app.logger.log('Erro copiando arquivo ' + newName + ' ' + JSON.stringify(err));  
1203:  
1204: });  
1205: }  
1206: ;  
1207: },  
1208:  
1209: restart : function(){  
1210: window.location.href = app.baseUrl + 'index.html'; //href para a página inicial  
1211: window.setTimeout(function(){  
1212: window.location.reload();  
1213: },1000);  
1214: },  
1215:   
1216: baseUrl : null,  
1217:  
1218: logFileName : "logOD.txt",  
1219:  
1220: dbName : "dadosOD.db",  
1221:  
1222: jsonName : "dadosOD.json",  
1223:  
1224: user\_login : null,  
1225:  
1226: senha\_login : null,  
1227:  
1228: posto : null,  
1229:  
1230: sentido : null,  
1231:   
1232: isTreinamento : false,  
1233:  
1234: filePaths : null, // { externalFolder : null, dbFolder : null, }  
1235:   
1236: changesCounter : 0,  
1237:   
1238: sumario\_lista : [],  
1239:   
1240: sumario\_lista\_cancelados : [],  
1241:   
1242: ultima\_pesquisa : [],  
1243:   
1244: logger : window.console //initial value, changed to use logger.js on file system initialization  
1245: ,  
1246:  
1247:}; // end of app  
1248:  
1249:// Registro do momento  
1250:var registro;  
1251:  
1252:$(document).ready(function() {  
1253:   
1254: //captura todos os erros   
1255: window.onerror = function(msg, url, line, col, error) {  
1256: // Note that col & error are new to the HTML 5 spec and may not be   
1257: // supported in every browser. It worked for me in Chrome.  
1258: var extra = !col ? '' : '\ncolumn: ' + col;  
1259: extra += !error ? '' : '\nerror: ' + error;  
1260:   
1261: var mesageString = "Error: " + msg + "\nurl: " + url + "\nline: " + line + extra;  
1262:  
1263: if (app == null){  
1264: console.log(mesageString);  
1265: } else {  
1266: app.logger.log(mesageString);  
1267: }  
1268:  
1269: // You can view the information in an alert to see things working like this:  
1270: if (msg.indexOf("TypeError: null is not an object (evaluating 'input.val().replace')")>-1){  
1271: //ignores  
1272: } else if (msg.indexOf("SecurityError: DOM Exception 18")>-1){ //app freezes  
1273:   
1274: if (app != null){  
1275: alert("Ocorreu uma condição que impede \no prosseguimento do programa.\nO aplicativo será reiniciado",  
1276: 'Erro fatal.',  
1277: app.restart,  
1278: 'error');  
1279: } else {  
1280: alert("Ocorreu uma condição que impede \no prosseguimento do programa.\nO aplicativo será reiniciado");  
1281: window.location.reload()  
1282: }  
1283:   
1284: } else {  
1285: alert("Ocorreu uma condição inesperada anote \n a mensagem abaixo\n" + mesageString);  
1286: }  
1287:   
1288:   
1289: var suppressErrorAlert = true;  
1290: // If you return true, then error alerts (like in older versions of   
1291: // Internet Explorer) will be suppressed.  
1292: return suppressErrorAlert;  
1293: };  
1294: insert\_controllers.insert();  
1295: app.initialize();  
1296:});

## .\mobile\www\js\ipadID.js

1:var ipadID = {  
 2:   
 3: storeID : function(id, cb){  
 4: window.resolveLocalFileSystemURL(cordova.file.dataDirectory,  
 5: function(dir){  
 6: app.removeFile(ipadID.ipadIDFileName,  
 7: cordova.file.dataDirectory,  
 8: function(){\_realStore(dir);}, //store even if remove old file fail  
 9: function(){\_realStore(dir);}  
 10: );  
 11: });  
 12:   
 13: function \_realStore(dir){  
 14: dir.getFile(ipadID.ipadIDFileName, {create:true},  
 15: function(file){  
 16: app.logger.log("Creating ipadIDFile " + file.nativeURL);  
 17: file.createWriter(function(idWriter){  
 18: idWriter.onerror = function(e){  
 19: app.logger.log("Erro escrevendo arquivo " + ipadID.ipadIDFileName);  
 20: }  
 21: var blob = new Blob([ id ], {  
 22: type : 'text/plain'  
 23: });  
 24: try {  
 25: idWriter.seek(idWriter.length);  
 26: idWriter.write(blob);  
 27: app.logger.log("Arquivo " + ipadID.ipadIDFileName + " criado.");  
 28: if (util.isFunction(cb)){  
 29: cb();  
 30: }  
 31: } catch (e) {  
 32: if (util.isFunction(cb)){  
 33: cb(e);  
 34: }  
 35: app.logger.log(e.message);  
 36: }  
 37: });  
 38: });  
 39: }  
 40: },  
 41:   
 42:  
 43: /\*\*  
 44: \* Read ipadIDFile  
 45: \* @param Function success callback  
 46: \* @param Function fail callback  
 47: \*/  
 48: readId : function(fileEntry, success, fail){  
 49: fileEntry.file(function(file){  
 50: var reader = new FileReader();  
 51: reader.readAsText(file);  
 52: reader.onloadend = function(e){  
 53: ipadID.id = reader.result.toString().trim();  
 54: app.logger.log("Lendo ipadID " + ipadID.id);  
 55: if (util.isFunction(success)){  
 56: success(ipadID.id);  
 57: }  
 58: };  
 59: reader.onerror = function(e){  
 60: app.logger.log("Erro lendo arquivo " + ipadID.ipadIDFileName + " ERRO: " + JSON.stringify(e));  
 61: if(util.isFunction(fail)){  
 62: fail();  
 63: }  
 64: };  
 65:   
 66: });  
 67: },  
 68:   
 69: requestID : function(success){  
 70: navigator.notification.prompt("Entre o ID desse ipad",  
 71: function(results){  
 72: if (results.buttonIndex == 1 || !ipadID.\_isValid(results.input1.trim())){  
 73: navigator.notification.alert(  
 74: "O ipadID informado deve estar no formato 000000 (seis dígitos), sendo 3 deles para o posto e 3 para o número de ordem [posto][ordem]",  
 75: ipadID.requestID(success),  
 76: "Erro entrando ipadID",  
 77: "OK"  
 78: );  
 79: } else if (results.buttonIndex ==2){  
 80: ipadID.storeID(results.input1.trim(),   
 81: function(e){  
 82: if (e == null){  
 83: ipadID.id = results.input1.trim();  
 84: alert("Id " + ipadID.id + " gravado com sucesso");  
 85: if(util.isFunction(success)){  
 86: success(ipadID.id);  
 87: }  
 88: } else {  
 89: alert(e.message);  
 90: }  
 91: });  
 92: }  
 93:   
 94: },  
 95: "Identificação do iPad",  
 96: ["Cancelar", "Cadastrar"]  
 97: );  
 98: },  
 99:   
 100: \_isValid : function (id){  
 101: return (id.length === 6   
 102: && $.isNumeric(id)   
 103: && id.substring(0,3) <= 303  
 104: && id.substring(0,3) > 0  
 105: && id.substring(3) > 0  
 106: && id.substring(3) <= 15)? true : false;  
 107: },  
 108:   
 109: id : null,  
 110:   
 111: ipadIDFileName : "ipID.txt",  
 112: }  
 113:;  
 114:

## .\mobile\www\js\jsonWriter.js

1:jsonWriter = {  
 2: \_fila : [],  
 3:  
 4: \_ocupado : false,  
 5:  
 6: \_logWriter : null,  
 7:  
 8: setJsonFile : function(arquivo) {  
 9: jsonWriter.\_logFile = arquivo;  
 10: },  
 11:  
 12: setJsonWriter : function(writer) {  
 13: var me = this;  
 14:  
 15: // inicia o monitor  
 16: setInterval(function() {  
 17: jsonWriter.\_monitoraFila();  
 18: }, 300);  
 19:  
 20: jsonWriter.\_logWriter = writer;  
 21: jsonWriter.\_logWriter.onwriteend = function(e) {  
 22: me.\_ocupado = false;  
 23: };  
 24: jsonWriter.\_logWriter.onerror = function(e) {  
 25: console.log('Erro de escrita: ' + e.message);  
 26: };  
 27: },  
 28:  
 29: \_monitoraFila : function() {  
 30: if (jsonWriter.\_fila.length > 0 && !jsonWriter.\_ocupado) {  
 31: jsonWriter.\_ocupado = true;  
 32: jsonWriter.\_internalWrite(jsonWriter.\_fila.shift());  
 33: }  
 34: },  
 35:  
 36: appendRow : function(str) {  
 37: console.log(str);  
 38: jsonWriter.\_fila.push(str);  
 39: },  
 40:  
 41: /\*\*  
 42: \* @param {fileEntry}  
 43: \* arquivo para leitura  
 44: \* @param {function}  
 45: \* cb callback que recebe o resultado da leitura do arquivo json  
 46: \*/  
 47: read : function(arquivo, cb) {  
 48: arquivo.file(function(file) {  
 49: var reader = new FileReader();  
 50: reader.onloadend = function(e) {  
 51: cb(this.result);  
 52: };  
 53: reader.readAsText(file);  
 54: });  
 55: },  
 56:  
 57: \_internalWrite : function(row) {  
 58: var blob = new Blob([ row + '\n' ], {  
 59: type : 'text/plain'  
 60: });  
 61: var me = this;  
 62: try {  
 63: me.\_logWriter.seek(me.\_logWriter.length);  
 64: me.\_logWriter.write(blob);  
 65: } catch (e) {  
 66: console.log(e.message);  
 67: }  
 68: }  
 69:  
 70:};

## .\mobile\www\js\logger.js

1:myLogger = {  
 2: \_fila : [],  
 3:  
 4: \_ocupado : false,  
 5:  
 6: \_logWriter : null,  
 7:  
 8: setLogFile : function(arquivo) {  
 9: myLogger.\_logFile = arquivo;  
 10: },  
 11:  
 12: setLogWriter : function(writer) {  
 13: var me = this;  
 14:  
 15: // inicia o monitor  
 16: setInterval(function() {  
 17: myLogger.\_monitoraFila();  
 18: }, 300);  
 19:  
 20: myLogger.\_logWriter = writer;  
 21: myLogger.\_logWriter.onwriteend = function(e) {  
 22: me.\_ocupado = false;  
 23: };  
 24: myLogger.\_logWriter.onerror = function(e) {  
 25: console.log('Erro de escrita do log: ' + e.message);  
 26: };  
 27: },  
 28:  
 29: \_monitoraFila : function() {  
 30: if (myLogger.\_fila.length > 0 && !myLogger.\_ocupado) {  
 31: myLogger.\_ocupado = true;  
 32: var strVO = myLogger.\_fila.slice();  
 33: myLogger.\_fila = [];  
 34:// while (myLogger.\_fila.length > 0){  
 35:// str.push(myLogger.\_fila.shift());  
 36:// if (myLogger.\_fila.length > 1){  
 37:// str += "\n";  
 38:// }  
 39:// }  
 40: myLogger.\_internalWrite(strVO);  
 41: }  
 42: },  
 43:  
 44: log : function(str) {  
 45: myLogger.\_fila.push("[" + device.uuid + "] [" + (new Date()) + "] " + str + "\n"); //log moment  
 46: },  
 47:  
 48: /\*\*  
 49: \* @param {fileEntry}  
 50: \* arquivo para leitura  
 51: \* @param {function}  
 52: \* cb callback que recebe o resultado da leitura do arquivo de log  
 53: \*/  
 54: read : function(arquivo, cb) {  
 55: arquivo.file(function(file) {  
 56: var reader = new FileReader();  
 57: reader.onloadend = function(e) {  
 58: cb(this.result);  
 59: };  
 60: reader.readAsText(file);  
 61: });  
 62: },  
 63:  
 64: \_internalWrite : function(strVO) {  
 65: //var log = "[" + device.uuid + "] [" + (new Date()) + "] " + str + "\n";  
 66: str = "";  
 67: while(strVO.length > 0){  
 68: str += "[" + (new Date()) + "]" + strVO.shift(); //write moment  
 69: if (strVO.length > 1){  
 70: str += "\n";  
 71: }  
 72: }  
 73:   
 74: var blob = new Blob([ str ], {  
 75: type : 'text/plain'  
 76: });  
 77: var me = this;  
 78: try {  
 79: me.\_logWriter.seek(me.\_logWriter.length);  
 80: me.\_logWriter.write(blob);  
 81: console.log(str);  
 82: } catch (e) {  
 83: console.log(e.message);  
 84: }  
 85: }  
 86:  
 87:};

## .\mobile\www\js\logins.js

1:var logins = {  
 2:  
 3: autenticaMaster : function(usuario, senha) {  
 4: if ((usuario == logins.user\_master.usr) && (senha == logins.user\_master.pwd)) {  
 5: return true;  
 6: }  
 7: return false;  
 8: },  
 9:  
 10: autentica : function(usuario, senha) {  
 11: var posto = null;  
 12: var msg = null;  
 13: if ((logins.user\_tester != undefined) && (logins.user\_tester.usr != undefined) && (usuario == logins.user\_tester.usr)  
 14: && (senha == logins.user\_tester.pwd)) {  
 15: logins.user\_logado = logins.user\_tester;  
 16: return true;  
 17: } else if (String(usuario).length == 3 && logins.verificaPostoSenha(String(usuario), senha)) {  
 18: if(datas.verificaData()){  
 19: app.isTreinamento = false;  
 20: app.logger.log("Sistema em modo de produção.");  
 21: return true;  
 22: }else{  
 23: msg = "Logins de produção não funcionam fora das datas da pesquisa real!";  
 24: alert(msg ,"Erro!",null,'error');  
 25: app.logger.log(msg);  
 26: return false;  
 27: }  
 28:   
 29: //treinamento  
 30: } else if (String(usuario).length == 4 && String(usuario).substr(3).toUpperCase() == 'T'){  
 31: if (datas.verificaData()){  
 32: msg = "Logins de treinamento não funcionam em datas da pesquisa real!";  
 33: alert(msg ,"Erro!",null,'error');  
 34: app.logger.log(msg);  
 35: return false;  
 36: }  
 37: posto = String(usuario).substr(0,3);  
 38: if (logins.verificaPostoSenha(posto, senha)){  
 39: app.isTreinamento = true;  
 40: msg = "O sistema funcionará em modo de treinamento!";  
 41: alert(msg,null,null,'error');  
 42: app.logger.log(msg);  
 43: return true;  
 44: }  
 45: }  
 46: msg = "Usuário e/ou Senha informados não estão cadastrados no sistema!";  
 47: alert(msg);  
 48: app.logger.log(msg);  
 49: return false;  
 50: },  
 51:   
 52: verificaPostoSenha : function(posto, senha){  
 53: if (posto.length == 3 && Number(posto) > 0) {  
 54: var i = Number(posto) - 1;  
 55: // for (i = 0; i < logins.users.length; i++) {  
 56: /\* comentar o 'for' se o sequencial do posto for igual ao posicionamento dele na lista. \*/  
 57: var login = logins.users[i];  
 58: if ((posto == login.usr) && (senha == login.pwd)) {  
 59: logins.user\_logado = login;  
 60: return true;  
 61: }  
 62: // }  
 63: }  
 64: return false;  
 65: },  
 66:   
 67: user\_master : { // Usuário exclusivo para configurar o idIpad na instalação, ele não faz login no aplicativo.  
 68: usr : 'Master', // usuário mestre  
 69: pwd : "434723" // senha mestra (ver teclado numérico para a string 'idIPad')  
 70: },  
 71:  
 72: user\_tester : { // Usuário destinado aos testes do aplicativo.  
 73: // Antes de ir para produção, basta comentar as duas linhas abaixo se quiser inativar para esse ambiente.  
 74: usr : 'Teste', // usuário de teste  
 75: pwd : "837837", // senha de teste (ver teclado numérico para a string 'tester')  
 76: perguntaExtra : true  
 77: },  
 78:   
 79: // Postos com pergunta extra na 1ª fase: 211  
 80: // Postos com pergunta extra na 2ª fase: 020, 184, 185   
 81: users : [  
 82: // usr = número do posto  
 83: // pwd = 4 últimos números do SNV  
 84: {  
 85: usr : '001',  
 86: pwd : '1590',  
 87: fase : '3',  
 88: ladoA : '',  
 89: ladoB : '',  
 90: trecho : 'ENTR AC-463/465 (BOM FUTURO) - ENTR BR-317'  
 91: }, {  
 92: usr : '002',  
 93: pwd : '0808',  
 94: fase : '1',  
 95: ladoA : 'LUCAS DO RIO VERDE',  
 96: ladoB : 'INÍCIO DA TRAVESSIA URBANA DE SORRISO',  
 97: trecho : 'LUCAS DO RIO VERDE – INÍCIO DA TRAVESSIA URBANA DE SORRISO'  
 98: }, {  
 99: usr : '003',  
 100: pwd : '0630',  
 101: fase : '2',  
 102: ladoA : '',  
 103: ladoB : '',  
 104: trecho : 'ENTR BR-104(A) - ENTR BR-104(B)'  
 105: }, {  
 106: usr : '004',  
 107: pwd : '0890',  
 108: fase : '2',  
 109: ladoA : '',  
 110: ladoB : '',  
 111: trecho : 'ENTR AL-225(A) - ENTR AL-225(B) (P/PORTO REAL DO COLÉGIO)'  
 112: }, {  
 113: usr : '005',  
 114: pwd : '0950',  
 115: fase : '3',  
 116: ladoA : '',  
 117: ladoB : '',  
 118: trecho : 'ENTR AL-115(A) - ENTR AL-115(B) (PALMEIRA DOS ÍNDIOS)'  
 119: }, {  
 120: usr : '006',  
 121: pwd : '0750',  
 122: fase : '4',  
 123: ladoA : '',  
 124: ladoB : '',  
 125: trecho : 'ENTR BR-101(A) - ENTR BR-101(B) (P/ALAGOINHAS)'  
 126: }, {  
 127: usr : '007',  
 128: pwd : '1810',  
 129: fase : '3',  
 130: ladoA : '',  
 131: ladoB : '',  
 132: trecho : 'ENTR BA-120(A)/262(B) (P/ITAJUÍPE) - ENTR BR-415/BA-120(B) (ITABUNA)'  
 133: }, {  
 134: usr : '008',  
 135: pwd : '0330',  
 136: fase : '2',  
 137: ladoA : '',  
 138: ladoB : '',  
 139: trecho : 'ENTR BR-324(A) - ENTR BR-324(B)/BA-502/503 (FEIRA DE SANTANA)'  
 140: }, {  
 141: usr : '009',  
 142: pwd : '0934',  
 143: fase : '3',  
 144: ladoA : '',  
 145: ladoB : '',  
 146: trecho : 'ENTR BA-643 (PLANALTO) - ENTR BA-641'  
 147: }, {  
 148: usr : '010',  
 149: pwd : '0365',  
 150: fase : '3',  
 151: ladoA : '',  
 152: ladoB : '',  
 153: trecho : 'ENTR BR-242(A) - ENTR BR-330/BA-148 (SEABRA)'  
 154: }, {  
 155: usr : '011',  
 156: pwd : '0300',  
 157: fase : '2',  
 158: ladoA : '',  
 159: ladoB : '',  
 160: trecho : 'JAGUARARI - ENTR BA-220 (SENHOR DO BONFIM)'  
 161: }, {  
 162: usr : '012',  
 163: pwd : '0530',  
 164: fase : '3',  
 165: ladoA : '',  
 166: ladoB : '',  
 167: trecho : 'ENTR CE-187 (P/MARRECOS) - ENTR BR-404/CE-176/187/363 (P/TAUÁ)'  
 168: }, {  
 169: usr : '013',  
 170: pwd : '0090',  
 171: fase : '3',  
 172: ladoA : '',  
 173: ladoB : '',  
 174: trecho : 'ENTR CE-253 (PACAJÚS) - ENTR BR-122(A)/CE-354 (CHORÓZINHO)'  
 175: }, {  
 176: usr : '014',  
 177: pwd : '0260',  
 178: fase : '2',  
 179: ladoA : '',  
 180: ladoB : '',  
 181: trecho : 'ENTR BR-226(B)/CE-275(B) (P/PEREIRO) - ENTR BR-404/434 (ICÓ)'  
 182: }, {  
 183: usr : '015',  
 184: pwd : '2195',  
 185: fase : '2',  
 186: ladoA : '',  
 187: ladoB : '',  
 188: trecho : 'ENTR ES-245(B) - ENTR ES-440'  
 189: }, {  
 190: usr : '016',  
 191: pwd : '2470',  
 192: fase : '3',  
 193: ladoA : '',  
 194: ladoB : '',  
 195: trecho : 'ENTR ES-375(A) (P/PIÚMA) - ENTR ES-375(B) (ICONHA)'  
 196: }, {  
 197: usr : '017',  
 198: pwd : '0070',  
 199: fase : '3',  
 200: ladoA : '',  
 201: ladoB : '',  
 202: trecho : 'ENTR BR-101(B) - ENTR ES-465 (P/DOMINGOS MARTINS)'  
 203: }, {  
 204: usr : '018',  
 205: pwd : '0015',  
 206: fase : '2',  
 207: ladoA : '',  
 208: ladoB : '',  
 209: trecho : 'KM 15,3 - ENTR BR-484 (P/PONTE SOBRE RIO DOCE)'  
 210: }, {  
 211: usr : '019',  
 212: pwd : '0110',  
 213: fase : '2',  
 214: ladoA : '',  
 215: ladoB : '',  
 216: trecho : 'ENTR GO-116 (P/FORMOSA) - ENTR GO-346 (P/CABEÇEIRAS)',  
 217: perguntaExtra : true // Possui pergunta extra na 4ª fase  
 218: }, {  
 219: usr : '020',  
 220: pwd : '0090',  
 221: fase : '2',  
 222: ladoA : '',  
 223: ladoB : '',  
 224: trecho : 'ENTR BR-457(B)/GO-219 - ENTR GO-020(A)'  
 225: // perguntaExtra : true // Possuiu pergunta extra na 2ª fase  
 226: }, {  
 227: usr : '021',  
 228: pwd : '0112',  
 229: fase : '4',  
 230: ladoA : '',  
 231: ladoB : '',  
 232: trecho : 'ENTR GO-338 (ABADIÂNIA - FIM TRAVESSIA URBANA) - ENTR BR-153(A) (P/ANÁPOLIS)'  
 233: }, {  
 234: usr : '022',  
 235: pwd : '0340',  
 236: fase : '3',  
 237: ladoA : '',  
 238: ladoB : '',  
 239: trecho : 'ENTR GO-353(A) (LINDA VISTA) - ENTR BR-414(A)/GO-151/244/353(B) (PORANGATU)'  
 240: }, {  
 241: usr : '023',  
 242: pwd : '0770',  
 243: fase : '2',  
 244: ladoA : '',  
 245: ladoB : '',  
 246: trecho : 'ENTR BR-154/452(A)/483 - ENTR BR-452(B) (DIV GO/MG) (ITUMBIARA)'  
 247: }, {  
 248: usr : '024',  
 249: pwd : '0510',  
 250: fase : '2',  
 251: ladoA : '',  
 252: ladoB : '',  
 253: trecho : 'ENTR BR-060(B)/158(B)/GO-184(B) (P/ESTREITO) - ENTR GO-050(B)'  
 254: }, {  
 255: usr : '025',  
 256: pwd : '0050',  
 257: fase : '2',  
 258: ladoA : '',  
 259: ladoB : '',  
 260: trecho : 'FIM DA DUPLICAÇÃO - ENTR BR-402/MA-110 (BACABEIRA)'  
 261: }, {  
 262: usr : '026',  
 263: pwd : '0150',  
 264: fase : '3',  
 265: ladoA : '',  
 266: ladoB : '',  
 267: trecho : 'ENTR BR-316(A) (CACHUCHA) - ENTR BR-316(B) (PERITORÓ)'  
 268: }, {  
 269: usr : '027',  
 270: pwd : '0680',  
 271: fase : '3',  
 272: ladoA : '',  
 273: ladoB : '',  
 274: trecho : 'RIO PINDARÉ - ENTR BR-010(A) (AÇAILÂNDIA)'  
 275: }, {  
 276: usr : '028',  
 277: pwd : '0670',  
 278: fase : '2',  
 279: ladoA : '',  
 280: ladoB : '',  
 281: trecho : 'ENTR BR-251(A) (CANACÍ) - ENTR BR-135/251(B)/308/365 (MONTES CLAROS)'  
 282: }, {  
 283: usr : '029',  
 284: pwd : '0370',  
 285: fase : '3',  
 286: ladoA : '',  
 287: ladoB : '',  
 288: trecho : 'ENTR MA-026 (DEZESSETE) - ENTR MA-034(A)/127/349 (CAXIAS)'  
 289: }, {  
 290: usr : '030',  
 291: pwd : '0130',  
 292: fase : '2',  
 293: ladoA : '',  
 294: ladoB : '',  
 295: trecho : 'ENTR MG-410 (P/PONTO DIAMANTE) - ENTR MG-181 (JOÃO PINHEIRO)'  
 296: }, {  
 297: usr : '031',  
 298: pwd : '0330',  
 299: fase : '3',  
 300: ladoA : '',  
 301: ladoB : '',  
 302: trecho : 'ENTR MG-238 (P/SETE LAGOAS) - ENTR MG-432 (P/ESMERALDAS)'  
 303: }, {  
 304: usr : '032',  
 305: pwd : '0040',  
 306: fase : '2',  
 307: ladoA : '',  
 308: ladoB : '',  
 309: trecho : 'ENTR BR-040(A) (ANEL RODOVIÁRIO DE BELO HORIZONTE) - ENTR BR-040(B)'  
 310: }, {  
 311: usr : '033',  
 312: pwd : '0570',  
 313: fase : '3',  
 314: ladoA : '',  
 315: ladoB : '',  
 316: trecho : 'ENTR BR-499 (SANTOS DUMONT) - ENTR ANT UNIÃO E INDÚSTRIA (B. TRIUNFO)'  
 317: }, {  
 318: usr : '034',  
 319: pwd : '0250',  
 320: fase : '4',  
 321: ladoA : '',  
 322: ladoB : '',  
 323: trecho : 'ENTR BR-365(B)/452(B) - RIO TIJUCO'  
 324: }, {  
 325: usr : '035',  
 326: pwd : '1130',  
 327: fase : '2',  
 328: ladoA : '',  
 329: ladoB : '',  
 330: trecho : 'ENTR BR-342(B)/418/MG-217 (RIB STO ANTÔNIO) (TEÓFILO OTONI) - ACESSO ITAMBACURI'  
 331: }, {  
 332: usr : '036',  
 333: pwd : '1280',  
 334: fase : '4',  
 335: ladoA : '',  
 336: ladoB : '',  
 337: trecho : 'FIM PISTA DUPLA - DOM CORRÊA'  
 338: }, {  
 339: usr : '037',  
 340: pwd : '1430',  
 341: fase : '3',  
 342: ladoA : '',  
 343: ladoB : '',  
 344: trecho : 'ENTR BR-116(A)/120 (LEOPOLDINA) - ENTR BR-116(B)'  
 345: }, {  
 346: usr : '038',  
 347: pwd : '0850',  
 348: fase : '3',  
 349: ladoA : '',  
 350: ladoB : '',  
 351: trecho : 'ENTR BR-259(B) (CURVELO) - ENTR BR-040(A)'  
 352: }, {  
 353: usr : '039',  
 354: pwd : '0290',  
 355: fase : '2',  
 356: ladoA : '',  
 357: ladoB : '',  
 358: trecho : 'ENTR BR-364(A)/262(A) (P/COMENDADOR GOMES) - ENTR BR-364(B)/262(B) (P/FRUTAL)'  
 359: }, {  
 360: usr : '040',  
 361: pwd : '0890',  
 362: fase : '4',  
 363: ladoA : '',  
 364: ladoB : '',  
 365: trecho : 'ENTR MG-187 (P/IBIÁ) - ENTR BR-146 (P/ARAXÁ)',  
 366: perguntaExtra : true // Possui pergunta extra na 4ª fase  
 367: }, {  
 368: usr : '041',  
 369: pwd : '0230',  
 370: fase : '3',  
 371: ladoA : '',  
 372: ladoB : '',  
 373: trecho : 'ENTR MG-451 (ITUTINGA) - ENTR BR-354 (LAVRAS)'  
 374: }, {  
 375: usr : '042',  
 376: pwd : '0215',  
 377: fase : '4',  
 378: ladoA : '',  
 379: ladoB : '',  
 380: trecho : 'ACESSO CARMO DO PARANAIBA - ENTR MG-230'  
 381: }, {  
 382: usr : '043',  
 383: pwd : '0100',  
 384: fase : '3',  
 385: ladoA : '',  
 386: ladoB : '',  
 387: trecho : 'PRESIDENTE OLEGÁRIO - ENTR BR-146(A)/365 (PATOS DE MINAS)'  
 388: }, {  
 389: usr : '044',  
 390: pwd : '0370',  
 391: fase : '3',  
 392: ladoA : '',  
 393: ladoB : '',  
 394: trecho : 'ENTR BR-153 - ENTR BR-154(A)'  
 395: }, {  
 396: usr : '045',  
 397: pwd : '0230',  
 398: fase : '2',  
 399: ladoA : '',  
 400: ladoB : '',  
 401: trecho : 'ENTR MG-425 (CORONEL FABRICIANO) - ENTR MG-320 (P/JAGUARAÇU)'  
 402: }, {  
 403: usr : '046',  
 404: pwd : '0490',  
 405: fase : '2',  
 406: ladoA : '',  
 407: ladoB : '',  
 408: trecho : 'ENTR BR-262(B) (BETIM) - ENTR MG-155'  
 409: }, {  
 410: usr : '047',  
 411: pwd : '0750',  
 412: fase : '3',  
 413: ladoA : '',  
 414: ladoB : '',  
 415: trecho : 'ENTR MG-458 (CAREAÇU) - ENTR BR-459 (P/POUSO ALEGRE)'  
 416: }, {  
 417: usr : '048',  
 418: pwd : '0324',  
 419: fase : '1',  
 420: ladoA : 'VILA SÃO PEDRO',  
 421: ladoB : 'ACESSO COLÔNIA AGRÍCOLA SEXTA LINHA',  
 422: trecho : 'ACESSO COLÂNIA AGRICOLA SEXTA LINHA - FINAL PISTA DUPLA (VILA VARGAS)'  
 423: }, {  
 424: usr : '049',  
 425: pwd : '0450',  
 426: fase : '1',  
 427: ladoA : 'ENTR MS-441(B) (P/BANDEIRANTES)',  
 428: ladoB : 'ENTR MS-244 (P/BONFIM)',  
 429: trecho : 'ENTR MS-441(B) (P/BANDEIRANTES) - ENTR MS-244 (P/BONFIM)'  
 430: }, {  
 431: usr : '050',  
 432: pwd : '1290',  
 433: fase : '2',  
 434: ladoA : '',  
 435: ladoB : '',  
 436: trecho : 'ENTR BR-158(B)/MS-395 (P/BRASILÂNDIA) - ENTR MS-453/459 (P/ARAPUÁ)'  
 437: }, {  
 438: usr : '051',  
 439: pwd : '0880',  
 440: fase : '2',  
 441: ladoA : '',  
 442: ladoB : '',  
 443: trecho : 'FIM TRAV RIO PARANÁ (PONTE M. JOPPERT) - INCÍO DA PISTA DUPLA'  
 444: }, {  
 445: usr : '052',  
 446: pwd : '0030',  
 447: fase : '1',  
 448: ladoA : 'ENTR BR-070(B)',  
 449: ladoB : 'ENTR MT-170 (CARAMÚJO)',  
 450: trecho : 'ENTR BR-070(B) - ENTR MT-170 (CARAMÚJO)'  
 451: }, {  
 452: usr : '053',  
 453: pwd : '0592',  
 454: fase : '1',  
 455: ladoA : 'ENTR MT-270(B)',  
 456: ladoB : 'ENTR MT-469(A)',  
 457: trecho : 'ENTR MT-270(B) - ENTR MT-469(A)'  
 458: }, {  
 459: usr : '054',  
 460: pwd : '0840',  
 461: fase : '3',  
 462: ladoA : '',  
 463: ladoB : '',  
 464: trecho : 'JANGADA - ENTR MT-246(B)'  
 465: }, {  
 466: usr : '055',  
 467: pwd : '0510',  
 468: fase : '3',  
 469: ladoA : '',  
 470: ladoB : '',  
 471: trecho : 'ENTR BR-222(B)/PA-332 (DOM ELISEU) - ENTR PA-125/263 (GURUPIZINHO)'  
 472: }, {  
 473: usr : '056',  
 474: pwd : '0335',  
 475: fase : '3',  
 476: ladoA : '',  
 477: ladoB : '',  
 478: trecho : 'ENTR PB-018 (P/CONDE) - ENTR PB-034'  
 479: }, {  
 480: usr : '057',  
 481: pwd : '0010',  
 482: fase : '4',  
 483: ladoA : '',  
 484: ladoB : '',  
 485: trecho : 'ENTR PB-100 (P/GALANTE) - ENTR BR-104(A)/408(B)/PB-095 (CAMPINA GRANDE)'  
 486: }, {  
 487: usr : '058',  
 488: pwd : '0510',  
 489: fase : '3',  
 490: ladoA : '',  
 491: ladoB : '',  
 492: trecho : 'ENTR PE-063 (P/AMARAJI) - ENTR PE-064/085 (RIBEIRÃO)'  
 493: }, {  
 494: usr : '059',  
 495: pwd : '0460',  
 496: fase : '3',  
 497: ladoA : '',  
 498: ladoB : '',  
 499: trecho : 'ENTR PE-460 - ENTR BR-316/428 (P/CABROBÓ)'  
 500: }, {  
 501: usr : '060',  
 502: pwd : '0140',  
 503: fase : '3',  
 504: ladoA : '',  
 505: ladoB : '',  
 506: trecho : 'ENTR PE-103 (P/BONITO) - ENTR BR-104/423(A) (CARUARÚ)'  
 507: }, {  
 508: usr : '061',  
 509: pwd : '0420',  
 510: fase : '4',  
 511: ladoA : '',  
 512: ladoB : '',  
 513: trecho : 'ENTR PE-450 (P/VERDEJANTE) - ENTR BR-116/361 (SALGUEIRO)'  
 514: }, {  
 515: usr : '062',  
 516: pwd : '0370',  
 517: fase : '3',  
 518: ladoA : '',  
 519: ladoB : '',  
 520: trecho : 'ENTR BR-428(A) (LAGOA GRANDE) - ENTR BR-235/407/423/428(B) (DIV PE/BA) (PETROLINA/JUAZEIRO)'  
 521: }, {  
 522: usr : '063',  
 523: pwd : '0830',  
 524: fase : '4',  
 525: ladoA : '',  
 526: ladoB : '',  
 527: trecho : 'ENTR PI-242 - ENTR BR-407(A)/PI-238/245(A) (PICOS)'  
 528: }, {  
 529: usr : '064',  
 530: pwd : '0130',  
 531: fase : '3',  
 532: ladoA : '',  
 533: ladoB : '',  
 534: trecho : 'ENTR BR-404(B)/407(B) - INÍCIO PISTA DUPLA (CAPITÃO CAMPOS)'  
 535: }, {  
 536: usr : '065',  
 537: pwd : '0420',  
 538: fase : '3',  
 539: ladoA : '',  
 540: ladoB : '',  
 541: trecho : 'DEMERVAL LOBÃO - ENTR PI-223 (MONSENHOR GIL)'  
 542: }, {  
 543: usr : '066',  
 544: pwd : '2710',  
 545: fase : '1',  
 546: ladoA : 'DIV SP/PR (CAB NORTE PONTE S/ RIO PARDINHO)',  
 547: ladoB : 'INÍCIO VARIANTE DO ALPINO(PISTA DIREITA)',  
 548: trecho : 'DIV SP/PR (CAB NORTE PONTE S/ RIO PARDINHO) - INÍCIO VARIANTE DO ALPINO(PISTA DIREITA)'  
 549: }, {  
 550: usr : '067',  
 551: pwd : '2810',  
 552: fase : '1',  
 553: ladoA : 'ENTR PR-427 (CAMPO DO TENENTE)',  
 554: ladoB : 'DIV PR/SC (RIO NEGRO/MAFRA)',  
 555: trecho : 'ENTR PR-427 (CAMPO DO TENENTE) - DIV PR/SC (RIO NEGRO/MAFRA)'  
 556: }, {  
 557: usr : '068',  
 558: pwd : '0205',  
 559: fase : '1',  
 560: ladoA : 'ACESSO GUARAPUAVA',  
 561: ladoB : 'ENTR BR-466 (P/GUARAPUAVA)',  
 562: trecho : 'ACESSO GUARAPUAVA - ENTR BR-466 (P/GUARAPUAVA)'  
 563: }, {  
 564: usr : '069',  
 565: pwd : '0310',  
 566: fase : '1',  
 567: ladoA : 'ENTR PR-182 (P/CAPITÃO LEÔNIDAS MARQUES)',  
 568: ladoB : 'ENTR BR-163',  
 569: trecho : 'ACESSO OESTE CASCAVEL - ENTR BR-163/PR-182 (P/CAPITÃO LEÔNIDAS MARQUES)'  
 570: }, {  
 571: usr : '070',  
 572: pwd : '0480',  
 573: fase : '2',  
 574: ladoA : '',  
 575: ladoB : '',  
 576: trecho : 'FIM PISTA DUPLA - ENTR PR-431 (P/JACAREZINHO)'  
 577: }, {  
 578: usr : '071',  
 579: pwd : '0490',  
 580: fase : '1',  
 581: ladoA : 'ENTR PR-281 (P/TIJUCAS DO SUL)',  
 582: ladoB : 'DIV PR/SC (ENTR BR-101)',  
 583: trecho : 'ENTR PR-281 (P/TIJUCAS DO SUL) - DIV PR/SC (ENTR BR-101)'  
 584: }, {  
 585: usr : '072',  
 586: pwd : '0095',  
 587: fase : '1',  
 588: ladoA : 'ENTR PR-510/511 (P/CONTENDA)',  
 589: ladoB : 'ENTR PR-428 (LAPA)',  
 590: trecho : 'ENTR PR-510/511 (P/CONTENDA) - ENTR PR-428 (LAPA)'  
 591: }, {  
 592: usr : '073',  
 593: pwd : '0730',  
 594: fase : '2',  
 595: ladoA : '',  
 596: ladoB : '',  
 597: trecho : 'ENTR BR-393(B) (TREVO MOURA BRASIL) - ENTR BR-492(A) (TREVO P/ AREAL)'  
 598: }, {  
 599: usr : '074',  
 600: pwd : '2930',  
 601: fase : '3',  
 602: ladoA : '',  
 603: ladoB : '',  
 604: trecho : 'ENTR RJ-162 (P/RIO DOURADO) - ENTR BR-120 (CASIMIRO DE ABREU)'  
 605: }, {  
 606: usr : '075',  
 607: pwd : '2130',  
 608: fase : '2',  
 609: ladoA : '',  
 610: ladoB : '',  
 611: trecho : 'ENTR RJ-159 (FLORIANO) - ENTR RJ-161 (RESENDE)'  
 612: }, {  
 613: usr : '076',  
 614: pwd : '0310',  
 615: fase : '2',  
 616: ladoA : '',  
 617: ladoB : '',  
 618: trecho : 'ENTR BR-356(A)/393(B) - ENTR BR-356(B) (ITAPERUNA)'  
 619: }, {  
 620: usr : '077',  
 621: pwd : '0330',  
 622: fase : '3',  
 623: ladoA : '',  
 624: ladoB : '',  
 625: trecho : 'SAPUCAIA - ENTR BR-040(A)'  
 626: }, {  
 627: usr : '078',  
 628: pwd : '0167',  
 629: fase : '3',  
 630: ladoA : '',  
 631: ladoB : '',  
 632: trecho : 'ACESSO À LAGOA DO BONFIM - ENTR RN-315/002(A) (SÃO JOSÉ DE MIPIBÚ/NÍSIA FLORESTA)'  
 633: }, {  
 634: usr : '079',  
 635: pwd : '0070',  
 636: fase : '3',  
 637: ladoA : '',  
 638: ladoB : '',  
 639: trecho : 'ENTR RN-203 (P/SÃO PEDRO) - ACESSO BOM JESUS'  
 640: }, {  
 641: usr : '080',  
 642: pwd : '0270',  
 643: fase : '3',  
 644: ladoA : '',  
 645: ladoB : '',  
 646: trecho : 'ENTR BR-104(B) (LAJES) - ENTR RN-023 (CAIÇARA DO RIO DO VENTO)'  
 647: }, {  
 648: usr : '081',  
 649: pwd : '0070',  
 650: fase : '3',  
 651: ladoA : '',  
 652: ladoB : '',  
 653: trecho : 'ENTR RN-233 (P/CARAÚBAS) - ENTR RN-031/177(A)'  
 654: }, {  
 655: usr : '082',  
 656: pwd : '1210',  
 657: fase : '3',  
 658: ladoA : '',  
 659: ladoB : '',  
 660: trecho : 'ENTR BR-429(B) (JI-PARANÁ) - ENTR RO-473 (P/URUPÁ)'  
 661: }, {  
 662: usr : '083',  
 663: pwd : '3275',  
 664: fase : '3',  
 665: ladoA : '',  
 666: ladoB : '',  
 667: trecho : 'ENTR ERS-703 (GUAIBA) - ENTR ERS-709 (P/ BARRA DO RIBEIRO)'  
 668: }, {  
 669: usr : '084',  
 670: pwd : '1150',  
 671: fase : '1',  
 672: ladoA : 'ENTR BR-472 (P/PALMITINHO)',  
 673: ladoB : 'ENTR RS-585/587 (SEBERI)',  
 674: trecho : 'ENTR RSC-472 (P/ PALMITINHO) - ENTR ERS-585/587 (SEBERI)'  
 675: }, {  
 676: usr : '085',  
 677: pwd : '1310',  
 678: fase : '1',  
 679: ladoA : 'ENTR RS-348 (VAL DE SERRA)',  
 680: ladoB : 'ENTR RS-509 (P/SANTA MARIA)',  
 681: trecho : 'ENTR ERS-348 (VAL DE SERRA) - ENTR ERS-509 (P/ SANTA MARIA)'  
 682: }, {  
 683: usr : '086',  
 684: pwd : '0180',  
 685: fase : '1',  
 686: ladoA : 'ACESSO A MINAS DO LEÃO',  
 687: ladoB : 'ENTR BR-471 (PANTANO GRANDE)',  
 688: trecho : 'ACESSO A MINAS DO LEAO - ENTR BRS-471 (PANTANO GRANDE)'  
 689: }, {  
 690: usr : '087',  
 691: pwd : '1840',  
 692: fase : '1',  
 693: ladoA : 'ENTR RS-705 (P/GERIBA)',  
 694: ladoB : 'ENTR BR-290(B)',  
 695: trecho : 'ENTR ERS-705 (GERIBA) - ENTR BRS-153(B) (P/ BAGE)'  
 696: }, {  
 697: usr : '088',  
 698: pwd : '0410',  
 699: fase : '1',  
 700: ladoA : 'ENTR BR-293(A)',  
 701: ladoB : 'ENTR BR-472(A)',  
 702: trecho : 'ENTR BRS-293(A) (P/ QUARAI) - ENTR BRS-472(A) (P/ BARRA DO QUARAI)'  
 703: }, {  
 704: usr : '089',  
 705: pwd : '0130',  
 706: fase : '1',  
 707: ladoA : 'ENTR BR-473 (P/BAGÉ)',  
 708: ladoB : 'ENTR RS-630 (P/DOM PEDRITO)',  
 709: trecho : 'ENTR RSC-473 (P/ BAGE) - ENTR ERS-630/634 (P/ DOM PEDRITO)'  
 710: }, {  
 711: usr : '090',  
 712: pwd : '0070',  
 713: fase : '1',  
 714: ladoA : 'ENTR BR-471(A) (QUINTA)',  
 715: ladoB : 'ACESSO PELOTAS',  
 716: trecho : 'ENTR BRS-471(A) (QUINTA) - ACESSO SUL A PELOTAS'  
 717: }, {  
 718: usr : '091',  
 719: pwd : '0210',  
 720: fase : '1',  
 721: ladoA : 'ACESSO LESTE A ITAQUI',  
 722: ladoB : 'ENTR BR-290(A)/293(A) (URUGUAIANA)',  
 723: trecho : 'ACESSO LESTE A ITAQUI - ENTR BR-290(A)/293(A) (URUGUAIANA)'  
 724: }, {  
 725: usr : '092',  
 726: pwd : '3830',  
 727: fase : '1',  
 728: ladoA : 'GARUVA',  
 729: ladoB : 'ENTR SC-430 (PIRABEIRABA)',  
 730: trecho : 'GARUVA - ENTR SC-430 (PIRABEIRABA)'  
 731: }, {  
 732: usr : '093',  
 733: pwd : '4010',  
 734: fase : '1',  
 735: ladoA : 'BALNEÁRIO DE CAMBORIÚ',  
 736: ladoB : 'P/PORTO BELO',  
 737: trecho : 'BALNEÁRIO DE CAMBORIÚ - P/PORTO BELO'  
 738: }, {  
 739: usr : '094',  
 740: pwd : '4125',  
 741: fase : '1',  
 742: ladoA : 'PONTE SOBRE O RIO DA MADRE',  
 743: ladoB : 'PONTE SOBRE O RIO ARAÇATUBA',  
 744: trecho : 'PONTE SOBRE O RIO DA MADRE - PONTE SOBRE O RIO ARAÇATUBA'  
 745: }, {  
 746: usr : '095',  
 747: pwd : '4240',  
 748: fase : '1',  
 749: ladoA : 'ENTR BR-285(A) (ARARANGUÁ)',  
 750: ladoB : 'ENTR BR-285(B)/SC-285 (SANGA DA TOCA)',  
 751: trecho : 'ENTR BR-285(A)/SC-449 (ARARANGUÁ) - ENTR BR-285(B)/SC-285 (P/ERMO)'  
 752: }, {  
 753: usr : '096',  
 754: pwd : '2910',  
 755: fase : '2',  
 756: ladoA : '',  
 757: ladoB : '',  
 758: trecho : 'ENTR SC-352(B) (P/TAIÓ) - SÃO CRISTOVÃO DO SUL'  
 759: }, {  
 760: usr : '097',  
 761: pwd : '1590',  
 762: fase : '3',  
 763: ladoA : '',  
 764: ladoB : '',  
 765: trecho : 'ENTR SC-463 (P/JABORÁ) - ENTR BR-283 (P/CONCÓRDIA)'  
 766: }, {  
 767: usr : '098',  
 768: pwd : '0020',  
 769: fase : '3',  
 770: ladoA : '',  
 771: ladoB : '',  
 772: trecho : 'ENTR SC-114(A) (ÍNDIOS) - ENTR SC-114(B) (P/PAINEL)'  
 773: }, {  
 774: usr : '099',  
 775: pwd : '0377',  
 776: fase : '1',  
 777: ladoA : 'ENTR SC-157(B) (P/MODELO)',  
 778: ladoB : 'ENTR BR-158(A) (P/CUNHA PORÃ)',  
 779: trecho : 'ENTR SC-157(B) (P/MODELO) - ENTR BR-158(A) (P/CUNHA PORÃ)'  
 780: }, {  
 781: usr : '100',  
 782: pwd : '0190',  
 783: fase : '4',  
 784: ladoA : '',  
 785: ladoB : '',  
 786: trecho : 'ENTR SC-114(B) (P/OTACÍLIO COSTA) - ENTR BR-116'  
 787: }, {  
 788: usr : '101',  
 789: pwd : '0040',  
 790: fase : '3',  
 791: ladoA : '',  
 792: ladoB : '',  
 793: trecho : 'ENTR SE-464 (P/SÃO CRISTOVÃO) - ENTR BR-349(B)/SE-265 (ITAPORANGA D`AJUDA)'  
 794: }, {  
 795: usr : '102',  
 796: pwd : '0060',  
 797: fase : '3',  
 798: ladoA : '',  
 799: ladoB : '',  
 800: trecho : 'ENTR SE-160 - ENTR SE-170(A)'  
 801: }, {  
 802: usr : '103',  
 803: pwd : '2410',  
 804: fase : '3',  
 805: ladoA : '',  
 806: ladoB : '',  
 807: trecho : 'ENTR SP-070 - ENTR SP-103 (CAÇAPAVA)'  
 808: }, {  
 809: usr : '104',  
 810: pwd : '2570',  
 811: fase : '3',  
 812: ladoA : '',  
 813: ladoB : '',  
 814: trecho : 'ENTR SP-228 - ENTR SP-057 (P/SIDERÚRGICA)'  
 815: }, {  
 816: usr : '105',  
 817: pwd : '1043',  
 818: fase : '3',  
 819: ladoA : '',  
 820: ladoB : '',  
 821: trecho : 'ACESSO GUAIÇARA - ENTR BR-154/267 (LINS)'  
 822: }, {  
 823: usr : '106',  
 824: pwd : '0810',  
 825: fase : '3',  
 826: ladoA : '',  
 827: ladoB : '',  
 828: trecho : 'DIV MG/SP - ENTR SP-063 (P/BRAGANÇA PAULISTA)'  
 829: }, {  
 830: usr : '107',  
 831: pwd : '0105',  
 832: fase : '2',  
 833: ladoA : '',  
 834: ladoB : '',  
 835: trecho : 'ENTR TO-420 - INICIO PISTA DUPLA ARAGUAINA'  
 836: }, {  
 837: usr : '108',  
 838: pwd : '0205',  
 839: fase : '3',  
 840: ladoA : '',  
 841: ladoB : '',  
 842: trecho : 'ENTR TO-342(B) (MIRANORTE) - ENTR TO-348 (BARROLÂNDIA)'  
 843: }, {  
 844: usr : '109',  
 845: pwd : '0300',  
 846: fase : '3',  
 847: ladoA : '',  
 848: ladoB : '',  
 849: trecho : 'ENTR TO-483 (FIGUEIRÓPOLIS) - ENTR TO-296(A)/373 (ALVORADA)'  
 850: }, {  
 851: usr : '110',  
 852: pwd : '0495',  
 853: fase : '3',  
 854: ladoA : '',  
 855: ladoB : '',  
 856: trecho : 'ENTR PI-260 (P/BARREIRAS DO PIAUÍ) - ENTR PI-255 (CORRENTE)'  
 857: }, {  
 858: usr : '111',  
 859: pwd : '0150',  
 860: fase : '3',  
 861: ladoA : '',  
 862: ladoB : '',  
 863: trecho : 'FIM PISTA DUPLA - ENTR GO-156(B)'  
 864: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
 865: }, {  
 866: usr : '112',  
 867: pwd : '0025',  
 868: fase : '1',  
 869: ladoA : 'ENTR PR-508 (P/MATINHOS)',  
 870: ladoB : 'ENTR BR-101/PR-408 (P/MORRETES)',  
 871: trecho : 'ENTR PR-508 (P/MATINHOS) - ENTR BR-101/PR-408 (P/MORRETES)'  
 872: }, {  
 873: usr : '113',  
 874: pwd : '0075',  
 875: fase : '1',  
 876: ladoA : 'ACESSO SANTA',  
 877: ladoB : 'ENTR BR-376(B)/PR-428 (SÃO LUIS PURUNÃ)',  
 878: trecho : 'ACESSO SANTA - ENTR BR-376(B)/PR-428 (SÃO LUIS PURUNÃ)'  
 879: }, {  
 880: usr : '114',  
 881: pwd : '0950',  
 882: fase : '1',  
 883: ladoA : 'ENTR MS-375 (ZUZU)',  
 884: ladoB : 'ENTR BR-163(A) (NOVA ALVORADA DO SUL)',  
 885: trecho : 'ENTR MS-375 (ZUZU) - ENTR BR-163(A) (NOVA ALVORADA)'  
 886: }, {  
 887: usr : '115',  
 888: pwd : '0440',  
 889: fase : '1',  
 890: ladoA : 'ENTR MS-434 (RAIMUNDO)',  
 891: ladoB : 'ENTR BR-483/497 (INÍCIO DA PISTA DUPLA)',  
 892: trecho : 'ENTR MS-434 (RAIMUNDO) - ENTR BR-483/497 (INÍCIO DA PISTA DUPLA)'  
 893: }, {  
 894: usr : '116',  
 895: pwd : '0810',  
 896: fase : '3',  
 897: ladoA : '',  
 898: ladoB : '',  
 899: trecho : 'ENTR PISTA INVERSA (B) - ENTR BR-101'  
 900: }, {  
 901: usr : '117',  
 902: pwd : '0070',  
 903: fase : '3',  
 904: ladoA : '',  
 905: ladoB : '',  
 906: trecho : 'ENTR BR-493(A) (P/MAGÉ) - ENTR RJ-107 (IMBARIÊ)'  
 907: }, {  
 908: usr : '118',  
 909: pwd : '0040',  
 910: fase : '1',  
 911: ladoA : 'ENTR BR-101',  
 912: ladoB : 'ENTR SC-108(A) (NEUDOR)',  
 913: trecho : 'ENTR BR-101 - ENTR SC-108(A) (NEUDOR)'  
 914: }, {  
 915: usr : '119',  
 916: pwd : '1490',  
 917: fase : '1',  
 918: ladoA : 'ENTR BR-280(A)/PR-446',  
 919: ladoB : 'ENTR PR-170',  
 920: trecho : 'ENTR BR-153(A)/PR-446 - ENTR PR-170'  
 921: }, {  
 922: usr : '120',  
 923: pwd : '0060',  
 924: fase : '2',  
 925: ladoA : '',  
 926: ladoB : '',  
 927: trecho : 'ENTR BR-262(B) - ENTR BR-050(B)'  
 928: }, {  
 929: usr : '121',  
 930: pwd : '0834',  
 931: fase : '1',  
 932: ladoA : 'ENTR MT-423',  
 933: ladoB : 'ITAUBA',  
 934: trecho : 'ENTR MT-423 - ITAUBA'  
 935: }, {  
 936: usr : '122',  
 937: pwd : '0852',  
 938: fase : '3',  
 939: ladoA : '',  
 940: ladoB : '',  
 941: trecho : 'MATUPÁ - GUARANTÃ DO NORTE'  
 942: }, {  
 943: usr : '123',  
 944: pwd : '0790',  
 945: fase : '3',  
 946: ladoA : '',  
 947: ladoB : '',  
 948: trecho : 'ENTR AL-215(B) - ENTR AL-220(A) (P/BARRA DE SÃO MIGUEL)'  
 949: }, {  
 950: usr : '124',  
 951: pwd : '0260',  
 952: fase : '4',  
 953: ladoA : '',  
 954: ladoB : '',  
 955: trecho : 'ENTR PE-270 (P/BUÍQUE) - ENTR BR-110 (CRUZEIRO DO NORDESTE)'  
 956: }, {  
 957: usr : '125',  
 958: pwd : '0090',  
 959: fase : '4',  
 960: ladoA : '',  
 961: ladoB : '',  
 962: trecho : 'ENTR PE-218 (P/BOM CONSELHO) - ENTR PE-187 (P/PALMEIRINA)'  
 963: }, {  
 964: usr : '126',  
 965: pwd : '1695',  
 966: fase : '3',  
 967: ladoA : '',  
 968: ladoB : '',  
 969: trecho : 'ENTR BR-420(B) (P/LAJE) - ENTR BA-542 (P/GUERÉM)'  
 970: }, {  
 971: usr : '127',  
 972: pwd : '0790',  
 973: fase : '4',  
 974: ladoA : '',  
 975: ladoB : '',  
 976: trecho : 'ENTR BA-493 (ITATIM) - ENTR BA-245'  
 977: }, {  
 978: usr : '128',  
 979: pwd : '0334',  
 980: fase : '2',  
 981: ladoA : '',  
 982: ladoB : '',  
 983: trecho : 'MAIRI - ENTR BR-349'  
 984: }, {  
 985: usr : '129',  
 986: pwd : '0270',  
 987: fase : '2',  
 988: ladoA : '',  
 989: ladoB : '',  
 990: trecho : 'GAVIÃO - ENTR BR-349 (NOVA FÁTIMA)'  
 991: }, {  
 992: usr : '130',  
 993: pwd : '1010',  
 994: fase : '2',  
 995: ladoA : '',  
 996: ladoB : '',  
 997: trecho : 'DIV BA/MG - ENTR BR-251(A) (P/SALINAS)'  
 998: }, {  
 999: usr : '131',  
1000: pwd : '0530',  
1001: fase : '2',  
1002: ladoA : '',  
1003: ladoB : '',  
1004: trecho : 'ENTR BA-937 (P/PAJEÚ DO VENTO) - ENTR BR-122(B)/430/BA-569 (CAETITÉ)'  
1005: }, {  
1006: usr : '132',  
1007: pwd : '0245',  
1008: fase : '3',  
1009: ladoA : '',  
1010: ladoB : '',  
1011: trecho : 'ENTR BA-131 (P/CAÉM) - ENTR BA-417'  
1012: }, {  
1013: usr : '133',  
1014: pwd : '0550',  
1015: fase : '2',  
1016: ladoA : '',  
1017: ladoB : '',  
1018: trecho : 'ENTR BR-235 - ENTR BA-220 (EUCLIDES DA CUNHA)'  
1019: }, {  
1020: usr : '134',  
1021: pwd : '0690',  
1022: fase : '3',  
1023: ladoA : '',  
1024: ladoB : '',  
1025: trecho : 'ENTR CE-373 (CARMELÓPOLIS) - ENTR CE-187/292 (CAMPOS SALES)'  
1026: }, {  
1027: usr : '135',  
1028: pwd : '0190',  
1029: fase : '2',  
1030: ladoA : '',  
1031: ladoB : '',  
1032: trecho : 'ENTR CE-266 (SUCESSO) - ENTR BR-226/404 (CRATEÚS)'  
1033: }, {  
1034: usr : '136',  
1035: pwd : '0610',  
1036: fase : '4',  
1037: ladoA : '',  
1038: ladoB : '',  
1039: trecho : 'ENTR CE-060(B) (MINEIROLÂNDIA) - ENTR CE-168 (PEDRA BRANCA)'  
1040: }, {  
1041: usr : '137',  
1042: pwd : '0050',  
1043: fase : '4',  
1044: ladoA : '',  
1045: ladoB : '',  
1046: trecho : 'ENTR CE-154 (PIRANGI) - ENTR CE-060(A)/265 (QUIXADÁ)'  
1047: }, {  
1048: usr : '138',  
1049: pwd : '0590',  
1050: fase : '2',  
1051: ladoA : '',  
1052: ladoB : '',  
1053: trecho : 'ENTR CE-456 (P/TARGINOS) - ENTR CE-257 (CANINDÉ)'  
1054: }, {  
1055: usr : '139',  
1056: pwd : '0070',  
1057: fase : '4',  
1058: ladoA : '',  
1059: ladoB : '',  
1060: trecho : 'ENTR CE-341 (CROATÁ) - ENTR CE-162 (SÃO LUÍS DO CURU)'  
1061: }, {  
1062: usr : '140',  
1063: pwd : '0160',  
1064: fase : '3',  
1065: ladoA : '',  
1066: ladoB : '',  
1067: trecho : 'ENTR CE-263 (P/JAGUARUANA) - RUSSAS'  
1068: }, {  
1069: usr : '141',  
1070: pwd : '2130',  
1071: fase : '2',  
1072: ladoA : '',  
1073: ladoB : '',  
1074: trecho : 'ENTR BR-381 (SÃO MATEUS) - ENTR ES-430 (P/JAGUARÉ)'  
1075: }, {  
1076: usr : '142',  
1077: pwd : '0300',  
1078: fase : '4',  
1079: ladoA : '',  
1080: ladoB : '',  
1081: trecho : 'ENTR ES-137 - ENTR ES-130 (P/NOVA VENÉCIA)'  
1082: }, {  
1083: usr : '143',  
1084: pwd : '1990',  
1085: fase : '3',  
1086: ladoA : '',  
1087: ladoB : '',  
1088: trecho : 'ENTR BA-290 (TEIXEIRA DE FREITAS) - ENTR BR-418 (P/POSTO DA MATA)'  
1089: }, {  
1090: usr : '144',  
1091: pwd : '0030',  
1092: fase : '2',  
1093: ladoA : '',  
1094: ladoB : '',  
1095: trecho : 'MORRO GRANDE (ACESSO IV C. ITAPEMIRIM) - ENTR ES-166 (COUTINHO)'  
1096: }, {  
1097: usr : '145',  
1098: pwd : '0130',  
1099: fase : '2',  
1100: ladoA : '',  
1101: ladoB : '',  
1102: trecho : 'ENTR MG-422 (RESPLENDOR) - ENTR BR-458 (CONSELHEIRO PENA)'  
1103: }, {  
1104: usr : '146',  
1105: pwd : '0125',  
1106: fase : '3',  
1107: ladoA : '',  
1108: ladoB : '',  
1109: trecho : 'ENTR GO-230 - ENTR GO-237'  
1110: }, {  
1111: usr : '147',  
1112: pwd : '0160',  
1113: fase : '2',  
1114: ladoA : '',  
1115: ladoB : '',  
1116: trecho : 'ENTR GO-114 (P/FLORES DE GOIÁS) - SANTA MARIA'  
1117: }, {  
1118: usr : '148',  
1119: pwd : '0470',  
1120: fase : '3',  
1121: ladoA : '',  
1122: ladoB : '',  
1123: trecho : 'DIV MG/GO - ENTR GO-010(A)'  
1124: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
1125: }, {  
1126: usr : '149',  
1127: pwd : '0050',  
1128: fase : '4',  
1129: ladoA : '',  
1130: ladoB : '',  
1131: trecho : 'ENTR GO-010 (P/LUZIÂNIA) - ENTR BR-050(B)/354/457/GO-309 (CRISTALINA)',  
1132: perguntaExtra : true // Possui pergunta extra na 4ª fase  
1133: }, {  
1134: usr : '150',  
1135: pwd : '0080',  
1136: fase : '3',  
1137: ladoA : '',  
1138: ladoB : '',  
1139: trecho : 'URUTAÍ - ENTR BR-490/GO-213/307 (IPAMERI)'  
1140: }, {  
1141: usr : '151',  
1142: pwd : '0550',  
1143: fase : '2',  
1144: ladoA : '',  
1145: ladoB : '',  
1146: trecho : 'ENTR GO-080(B) (P/SAO FRANCISCO DE GOIÁS) - ENTR GO-431 (P/PIRENOPOLIS)'  
1147: }, {  
1148: usr : '152',  
1149: pwd : '0190',  
1150: fase : '2',  
1151: ladoA : '',  
1152: ladoB : '',  
1153: trecho : 'ENTR BR-153(A)/GO-342(B) - ENTR BR-153(B)/GO-237 (URUAÇÚ)'  
1154: }, {  
1155: usr : '153',  
1156: pwd : '0750',  
1157: fase : '2',  
1158: ladoA : '',  
1159: ladoB : '',  
1160: trecho : 'ENTR GO-210(B) (P/PANAMÁ) - ENTR BR-154/452(A)/483'  
1161: }, {  
1162: usr : '154',  
1163: pwd : '0050',  
1164: fase : '2',  
1165: ladoA : '',  
1166: ladoB : '',  
1167: trecho : 'ENTR GO-040 - ENTR BR-154(A)/483(A)/GO-206(A) (P/CACHOEIRA DOURADA)'  
1168: }, {  
1169: usr : '155',  
1170: pwd : '0570',  
1171: fase : '3',  
1172: ladoA : '',  
1173: ladoB : '',  
1174: trecho : 'ENTR GO-194 (P/PORTELÂNDIA) - INÍCIO PISTA DUPLA'  
1175: }, {  
1176: usr : '156',  
1177: pwd : '0440',  
1178: fase : '3',  
1179: ladoA : '',  
1180: ladoB : '',  
1181: trecho : 'ENTR GO-174(B) - ENTR GO-206 (P/CAÇU)'  
1182: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
1183: }, {  
1184: usr : '157',  
1185: pwd : '0280',  
1186: fase : '2',  
1187: ladoA : '',  
1188: ladoB : '',  
1189: trecho : 'ENTR MA-006(A) (COCALINHO) - ENTR MA-216 (BOM JARDIM)'  
1190: }, {  
1191: usr : '158',  
1192: pwd : '0860',  
1193: fase : '3',  
1194: ladoA : '',  
1195: ladoB : '',  
1196: trecho : 'ENTR MA-259 (P/TUNTUM) - ENTR MA-012/272 (BARRA DO CORDA)'  
1197: }, {  
1198: usr : '159',  
1199: pwd : '0300',  
1200: fase : '2',  
1201: ladoA : '',  
1202: ladoB : '',  
1203: trecho : 'ENTR BR-222(B) (ESTACA ZERO) - ENTR MA-240 (PIO XII)'  
1204: }, {  
1205: usr : '160',  
1206: pwd : '0730',  
1207: fase : '3',  
1208: ladoA : '',  
1209: ladoB : '',  
1210: trecho : 'BOM JESUS DO TOCANTINS - RIO JACUNDÁ (R MÃE MARIA)'  
1211: }, {  
1212: usr : '161',  
1213: pwd : '0750',  
1214: fase : '3',  
1215: ladoA : '',  
1216: ladoB : '',  
1217: trecho : 'ENTR BR-122/251/365 (MONTES CLAROS) - ENTR BR-451 (BOCAIÚVA)'  
1218: }, {  
1219: usr : '162',  
1220: pwd : '0240',  
1221: fase : '4',  
1222: ladoA : '',  
1223: ladoB : '',  
1224: trecho : 'ENTR BR-342/MG-404 (SALINAS) - RIO VACARIA'  
1225: }, {  
1226: usr : '163',  
1227: pwd : '0290',  
1228: fase : '2',  
1229: ladoA : '',  
1230: ladoB : '',  
1231: trecho : 'ENTR MA-132/270 (COLINAS) - ENTR MA-134 (PEIXE)'  
1232: }, {  
1233: usr : '164',  
1234: pwd : '0030',  
1235: fase : '3',  
1236: ladoA : '',  
1237: ladoB : '',  
1238: trecho : 'ENTR BR-040(B) - ENTR MG-155 (SÃO BRÁS DO SUAÇUÍ)'  
1239: }, {  
1240: usr : '165',  
1241: pwd : '0335',  
1242: fase : '2',  
1243: ladoA : '',  
1244: ladoB : '',  
1245: trecho : 'ENTR MG-434 - ENTR MG-435 (P/CAETÉ)'  
1246: }, {  
1247: usr : '166',  
1248: pwd : '0510',  
1249: fase : '2',  
1250: ladoA : '',  
1251: ladoB : '',  
1252: trecho : 'ENTR MG-275 (P/CARANDAÍ) - ACESSO ALTO DOCE (INÍCIO PISTA DUPLA)'  
1253: }, {  
1254: usr : '167',  
1255: pwd : '0130',  
1256: fase : '3',  
1257: ladoA : '',  
1258: ladoB : '',  
1259: trecho : 'ENTR BR-040(B) - ENTR MG-135'  
1260: }, {  
1261: usr : '168',  
1262: pwd : '0250',  
1263: fase : '3',  
1264: ladoA : '',  
1265: ladoB : '',  
1266: trecho : 'ENTR MG-190 (RIO BAGAGEM) - ACESSO INDIANÓPOLIS'  
1267: }, {  
1268: usr : '169',  
1269: pwd : '0152',  
1270: fase : '3',  
1271: ladoA : '',  
1272: ladoB : '',  
1273: trecho : 'ENTR BR-352(B)/GO-210(B)/330 (CATALÃO) - ENTR GO-402'  
1274: }, {  
1275: usr : '170',  
1276: pwd : '0010',  
1277: fase : '3',  
1278: ladoA : '',  
1279: ladoB : '',  
1280: trecho : 'ENTR BR-050/365/452/455 (UBERLÂNDIA) - ENTR BR-153/464 (PRATA)'  
1281: }, {  
1282: usr : '171',  
1283: pwd : '0310',  
1284: fase : '4',  
1285: ladoA : '',  
1286: ladoB : '',  
1287: trecho : 'ENTR BR-050(B)/455/497 (UBERLÂNDIA) - ENTR BR-452(B)'  
1288: }, {  
1289: usr : '172',  
1290: pwd : '0230',  
1291: fase : '4',  
1292: ladoA : '',  
1293: ladoB : '',  
1294: trecho : 'ENTR MG-190 (P/NOVA PONTE) - ACESSO SANTA JULIANA'  
1295: }, {  
1296: usr : '173',  
1297: pwd : '0110',  
1298: fase : '3',  
1299: ladoA : '',  
1300: ladoB : '',  
1301: trecho : 'ACESSO CARLOS CHAGAS - ENTR MG-412'  
1302: }, {  
1303: usr : '174',  
1304: pwd : '0110',  
1305: fase : '2',  
1306: ladoA : '',  
1307: ladoB : '',  
1308: trecho : 'ACESSO SÃO JOSÉ DO JACURÍ - ENTR MG-117 (SÃO JOÃO EVANGELISTA)'  
1309: }, {  
1310: usr : '175',  
1311: pwd : '1310',  
1312: fase : '4',  
1313: ladoA : '',  
1314: ladoB : '',  
1315: trecho : 'TREVO DE ACESSO SÃO JOÃO DO MANHUAÇU - ENTR MG-265 (P/DIVINO)'  
1316: }, {  
1317: usr : '176',  
1318: pwd : '0210',  
1319: fase : '2',  
1320: ladoA : '',  
1321: ladoB : '',  
1322: trecho : 'ENTR ES-185(B) (P/IÚNA) - DIV ES/MG'  
1323: }, {  
1324: usr : '177',  
1325: pwd : '1195',  
1326: fase : '3',  
1327: ladoA : '',  
1328: ladoB : '',  
1329: trecho : 'ACESSO ITANHOMI - ENTR BR-458(A) (TURUAÇÚ)'  
1330: }, {  
1331: usr : '178',  
1332: pwd : '0030',  
1333: fase : '2',  
1334: ladoA : '',  
1335: ladoB : '',  
1336: trecho : 'ENTR BR-116(B) - ENTR MG-126 (BICAS)'  
1337: }, {  
1338: usr : '179',  
1339: pwd : '1370',  
1340: fase : '2',  
1341: ladoA : '',  
1342: ladoB : '',  
1343: trecho : 'ENTR BR-265/356 (MURIAÉ) - ENTR MG-285 (LARANJAL)'  
1344: }, {  
1345: usr : '180',  
1346: pwd : '0310',  
1347: fase : '3',  
1348: ladoA : '',  
1349: ladoB : '',  
1350: trecho : 'PONTE S/ RIO PARÁ - ENTR MG-423(A) (PITANGUI)'  
1351: }, {  
1352: usr : '181',  
1353: pwd : '0195',  
1354: fase : '2',  
1355: ladoA : '',  
1356: ladoB : '',  
1357: trecho : 'ACESSO MORADA NOVA DE MINAS - ACESSO SÃO JOSÉ DO BURITI'  
1358: }, {  
1359: usr : '182',  
1360: pwd : '0330',  
1361: fase : '2',  
1362: ladoA : '',  
1363: ladoB : '',  
1364: trecho : 'ENTR BR-367(B) (GOUVEIA) - ACESSO NOSSA SENHORA DA GLÓRIA'  
1365: }, {  
1366: usr : '183',  
1367: pwd : '0910',  
1368: fase : '2',  
1369: ladoA : '',  
1370: ladoB : '',  
1371: trecho : 'ENTR MG-255 - ENTR BR-262(B) (DIV MG/SP)'  
1372: }, {  
1373: usr : '184',  
1374: pwd : '0250',  
1375: fase : '3',  
1376: ladoA : '',  
1377: ladoB : '',  
1378: trecho : 'ENTR BR-462(B) - ENTR BR-262'  
1379: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
1380: }, {  
1381: usr : '185',  
1382: pwd : '0930',  
1383: fase : '4',  
1384: ladoA : '',  
1385: ladoB : '',  
1386: trecho : 'ENTR BR-452 (P/UBERLÂNDIA) - ENTR BR-462 (PERDIZES)'  
1387: // perguntaExtra : true // Possuiu pergunta extra na 3ª fase  
1388: }, {  
1389: usr : '186',  
1390: pwd : '0070',  
1391: fase : '2',  
1392: ladoA : '',  
1393: ladoB : '',  
1394: trecho : 'ENTR BR-265(B) - CAMPOS GERAIS'  
1395: }, {  
1396: usr : '187',  
1397: pwd : '0030',  
1398: fase : '2',  
1399: ladoA : '',  
1400: ladoB : '',  
1401: trecho : 'ENTR MG-354(B) - ENTR BR-265(A)'  
1402: }, {  
1403: usr : '188',  
1404: pwd : '0190',  
1405: fase : '2',  
1406: ladoA : '',  
1407: ladoB : '',  
1408: trecho : 'ENTR MG-135/338 (BARBACENA) - ACESSO TIRADENTES'  
1409: }, {  
1410: usr : '189',  
1411: pwd : '0670',  
1412: fase : '4',  
1413: ladoA : '',  
1414: ladoB : '',  
1415: trecho : 'ENTR BR-265 (LAVRAS) - ENTR MG-167(A) (P/TRÊS CORAÇÕES)'  
1416: }, {  
1417: usr : '190',  
1418: pwd : '0350',  
1419: fase : '2',  
1420: ladoA : '',  
1421: ladoB : '',  
1422: trecho : 'ACESSO FORMIGA - ENTR MG-164 (CANDEIAS)'  
1423: }, {  
1424: usr : '191',  
1425: pwd : '0590',  
1426: fase : '2',  
1427: ladoA : '',  
1428: ladoB : '',  
1429: trecho : 'ENTR MG-270 (CARMÓPOLIS DE MINAS) - ENTR BR-494(A) (P/OLIVEIRA)'  
1430: }, {  
1431: usr : '192',  
1432: pwd : '0230',  
1433: fase : '3',  
1434: ladoA : '',  
1435: ladoB : '',  
1436: trecho : 'ENTR BR-354(A) (P/BAMBUÍ) - ENTR BR-354(B)'  
1437: }, {  
1438: usr : '193',  
1439: pwd : '0205',  
1440: fase : '2',  
1441: ladoA : '',  
1442: ladoB : '',  
1443: trecho : 'ACESSO GUIMARÂNIA - ENTR MG-188'  
1444: }, {  
1445: usr : '194',  
1446: pwd : '0090',  
1447: fase : '3',  
1448: ladoA : '',  
1449: ladoB : '',  
1450: trecho : 'ENTR MG-410 - PRESIDENTE OLEGÁRIO'  
1451: }, {  
1452: usr : '195',  
1453: pwd : '0150',  
1454: fase : '4',  
1455: ladoA : '',  
1456: ladoB : '',  
1457: trecho : 'ENTR MG-060 (P/SÃO GONÇALO DO ABAETÉ) - ACESSO GALENA'  
1458: }, {  
1459: usr : '196',  
1460: pwd : '0410',  
1461: fase : '2',  
1462: ladoA : '',  
1463: ladoB : '',  
1464: trecho : 'ENTR BR-154(B)/461/464 (ITUIUTABA) - ACESSO GURINHATÃ'  
1465: }, {  
1466: usr : '197',  
1467: pwd : '0830',  
1468: fase : '2',  
1469: ladoA : '',  
1470: ladoB : '',  
1471: trecho : 'ENTR BR-365 (P/MONTE ALEGRE DE MINAS) - ENTR BR-464/497 (P/PRATA)'  
1472: }, {  
1473: usr : '198',  
1474: pwd : '0170',  
1475: fase : '3',  
1476: ladoA : '',  
1477: ladoB : '',  
1478: trecho : 'ENTR MG-229 (P/SENHORA DO PORTO) - ENTR MG-232 (P/CARMESIA)'  
1479: }, {  
1480: usr : '199',  
1481: pwd : '0370',  
1482: fase : '2',  
1483: ladoA : '',  
1484: ladoB : '',  
1485: trecho : 'ENTR MG-320 (P/SÃO JOSÉ DO GOIABAL) - ENTR BR-120 (VARGEM LINDA)'  
1486: }, {  
1487: usr : '200',  
1488: pwd : '0170',  
1489: fase : '2',  
1490: ladoA : '',  
1491: ladoB : '',  
1492: trecho : 'ACESSO À GOV. VALADARES - PERIQUITO'  
1493: }, {  
1494: usr : '201',  
1495: pwd : '0685',  
1496: fase : '3',  
1497: ladoA : '',  
1498: ladoB : '',  
1499: trecho : 'ACESSO FLORESTAL - ENTR BR-352 (PARÁ DE MINAS)'  
1500: }, {  
1501: usr : '202',  
1502: pwd : '0050',  
1503: fase : '2',  
1504: ladoA : '',  
1505: ladoB : '',  
1506: trecho : 'ENTR MG-050(B) (P/DIVINOPÓLIS) - ENTR MG-260 (P/CLÁUDIO)'  
1507: }, {  
1508: usr : '203',  
1509: pwd : '0270',  
1510: fase : '2',  
1511: ladoA : '',  
1512: ladoB : '',  
1513: trecho : 'ACESSO CONCEIÇÃO DO RIO VERDE - ENTR BR-460'  
1514: }, {  
1515: usr : '204',  
1516: pwd : '0023',  
1517: fase : '3',  
1518: ladoA : '',  
1519: ladoB : '',  
1520: trecho : 'ACESSO SANTA RITA DAS CALDAS - ENTR MG-179 (POUSO ALEGRE)'  
1521: }, {  
1522: usr : '205',  
1523: pwd : '0270',  
1524: fase : '3',  
1525: ladoA : '',  
1526: ladoB : '',  
1527: trecho : 'ENTR BR-369 (BOTELHOS) - ENTR BR-267(A) (P/BANDEIRA DO SUL)'  
1528: }, {  
1529: usr : '206',  
1530: pwd : '0212',  
1531: fase : '3',  
1532: ladoA : '',  
1533: ladoB : '',  
1534: trecho : 'ENTR BR-487(A) - ACESSO NAVIRAÍ I'  
1535: }, {  
1536: usr : '207',  
1537: pwd : '0612',  
1538: fase : '1',  
1539: ladoA : 'ENTR BR-267(A)',  
1540: ladoB : 'INÍCIO DUPLICAÇÃO',  
1541: trecho : 'ENTR BR-267(A) - INÍCIO DUPLICAÇÃO'  
1542: }, {  
1543: usr : '208',  
1544: pwd : '0090',  
1545: fase : '3',  
1546: ladoA : '',  
1547: ladoB : '',  
1548: trecho : 'ENTR MS-141 (IVINHEMA) - VILA AMANDINA'  
1549: }, {  
1550: usr : '209',  
1551: pwd : '1390',  
1552: fase : '1',  
1553: ladoA : 'ENTR MS-339/446/448 (MIRANDA)',  
1554: ladoB : 'ENTR MS-185/243 (GUAICURUS)',  
1555: trecho : 'ENTR MS-339/446/448 (MIRANDA) - ENTR MS-185/243 (GUAICURUS)'  
1556: }, {  
1557: usr : '210',  
1558: pwd : '0058',  
1559: fase : '1',  
1560: ladoA : 'BURITIZINHO',  
1561: ladoB : 'ENTR MS-223(A) (SILVOLÂNDIA)',  
1562: trecho : 'BURITIZINHO - ENTR MS-223(A) (SILVOLÂNDIA)'  
1563: }, {  
1564: usr : '211',  
1565: pwd : '0390',  
1566: fase : '1',  
1567: ladoA : 'ENTR MS-316 (PARAÍSO DAS ÁGUAS)',  
1568: ladoB : 'ENTR MS-324(A)',  
1569: trecho : 'ENTR MS-316 (PARAÍSO) - ENTR MS-324(A)'  
1570: // perguntaExtra : true // Possuiu pergunta extra na 1ª fase  
1571: }, {  
1572: usr : '212',  
1573: pwd : '0180',  
1574: fase : '3',  
1575: ladoA : '',  
1576: ladoB : '',  
1577: trecho : 'ENTR BR-158(B) (P/TAMBOARA) - ACESSO ALTO PARANÁ/MARISTELA'  
1578: }, {  
1579: usr : '213',  
1580: pwd : '0050',  
1581: fase : '2',  
1582: ladoA : '',  
1583: ladoB : '',  
1584: trecho : 'ENTR SP-425 (PRESIDENTE PRUDENTE) - ENTR SP-425'  
1585: }, {  
1586: usr : '214',  
1587: pwd : '0330',  
1588: fase : '1',  
1589: ladoA : 'ENTR MT-373',  
1590: ladoB : 'ENTR MT-130(A) (PRIMAVERA DO LESTE)',  
1591: trecho : 'ENTR MT-373 - ENTR MT-130(A) (PRIMAVERA DO LESTE)'  
1592: }, {  
1593: usr : '215',  
1594: pwd : '0765',  
1595: fase : '2',  
1596: ladoA : '',  
1597: ladoB : '',  
1598: trecho : 'FIM VARIANTE II SERRA DE SÃO VICENTE - ACESSO DISTRITO INDUSTRIAL'  
1599: }, {  
1600: usr : '216',  
1601: pwd : '0150',  
1602: fase : '3',  
1603: ladoA : '',  
1604: ladoB : '',  
1605: trecho : 'ENTR BR-010(B) (P/SANTA MARIA DO PARÁ) - ENTR PA-324 (P/SALINÓPOLIS)'  
1606: }, {  
1607: usr : '217',  
1608: pwd : '0270',  
1609: fase : '3',  
1610: ladoA : '',  
1611: ladoB : '',  
1612: trecho : 'ENTR PB-177 (SOLEDADE) - ENTR PB-195 (JUAZEIRINHO)'  
1613: }, {  
1614: usr : '218',  
1615: pwd : '0322',  
1616: fase : '2',  
1617: ladoA : '',  
1618: ladoB : '',  
1619: trecho : 'BARRA DE SANTANA - ENTR PB-196 (P/RIACHO DE SANTO ANTÔNIO)'  
1620: }, {  
1621: usr : '219',  
1622: pwd : '0280',  
1623: fase : '2',  
1624: ladoA : '',  
1625: ladoB : '',  
1626: trecho : 'ENTR PE-263 (P/ITAPETIM) - ENTR PE-320 (SÃO JOSÉ DO EGITO)'  
1627: }, {  
1628: usr : '220',  
1629: pwd : '0410',  
1630: fase : '3',  
1631: ladoA : '',  
1632: ladoB : '',  
1633: trecho : 'ENTR PE-015 (P/PAULISTA - VIADUTO SOBRE PE-015) - VIADUTO SOBRE AVENIDA CAXANGÁ'  
1634: }, {  
1635: usr : '221',  
1636: pwd : '0530',  
1637: fase : '2',  
1638: ladoA : '',  
1639: ladoB : '',  
1640: trecho : 'DIV PE/AL - ENTR BR-423(A)'  
1641: }, {  
1642: usr : '222',  
1643: pwd : '0630',  
1644: fase : '2',  
1645: ladoA : '',  
1646: ladoB : '',  
1647: trecho : 'ENTR PE-590 (P/IPUBÍ) - ENTR BR-122 (OURICURÍ)'  
1648: }, {  
1649: usr : '223',  
1650: pwd : '0250',  
1651: fase : '3',  
1652: ladoA : '',  
1653: ladoB : '',  
1654: trecho : 'ENTR CE-292(B) - DIV CE/PE'  
1655: }, {  
1656: usr : '224',  
1657: pwd : '0300',  
1658: fase : '3',  
1659: ladoA : '',  
1660: ladoB : '',  
1661: trecho : 'ENTR PE-310/312 (CUSTÓDIA) - ENTR BR-426/PE-340 (SÍTIO DOS NUNES)'  
1662: }, {  
1663: usr : '225',  
1664: pwd : '0250',  
1665: fase : '2',  
1666: ladoA : '',  
1667: ladoB : '',  
1668: trecho : 'RAJADA - ENTR BR-235(A)'  
1669: }, {  
1670: usr : '226',  
1671: pwd : '0115',  
1672: fase : '2',  
1673: ladoA : '',  
1674: ladoB : '',  
1675: trecho : 'ENTR PI-394 - ENTR PI-140 (CANTO DO BURITI)'  
1676: }, {  
1677: usr : '227',  
1678: pwd : '0070',  
1679: fase : '3',  
1680: ladoA : '',  
1681: ladoB : '',  
1682: trecho : 'ENTR PI-211 (CAXINGÓ) - ENTR PI-213(A)'  
1683: }, {  
1684: usr : '228',  
1685: pwd : '0190',  
1686: fase : '2',  
1687: ladoA : '',  
1688: ladoB : '',  
1689: trecho : 'ACESSO OESTE SOBRAL - ENTR BR-403(B)/CE-183'  
1690: }, {  
1691: usr : '229',  
1692: pwd : '0330',  
1693: fase : '2',  
1694: ladoA : '',  
1695: ladoB : '',  
1696: trecho : 'ENTR MA-034 (SÃO JOÃO DOS PATOS) - ENTR BR-230(B)/MA-364(A) (DOIS IRMÃOS)'  
1697: }, {  
1698: usr : '230',  
1699: pwd : '0290',  
1700: fase : '2',  
1701: ladoA : '',  
1702: ladoB : '',  
1703: trecho : 'ENTR PI-130 (P/AMARANTE) - ENTR PI-120'  
1704: }, {  
1705: usr : '231',  
1706: pwd : '0210',  
1707: fase : '3',  
1708: ladoA : '',  
1709: ladoB : '',  
1710: trecho : 'ENTR SP-252 (GUAPIARA) - ENTR SP-165/249 (APIAÍ)'  
1711: }, {  
1712: usr : '232',  
1713: pwd : '0180',  
1714: fase : '1',  
1715: ladoA : 'ENTR PR-549 (P/LUIZIANA)',  
1716: ladoB : 'ENTR PR-462 (P/IRETAMA)',  
1717: trecho : 'ENTR PR-549 (P/LUIZIANA) - ENTR PR-462 (P/IRETAMA)'  
1718: }, {  
1719: usr : '233',  
1720: pwd : '0080',  
1721: fase : '3',  
1722: ladoA : '',  
1723: ladoB : '',  
1724: trecho : 'ACESSO IVAIPORÃ - ENTR BR-487 (P/MANOEL RIBAS)'  
1725: }, {  
1726: usr : '234',  
1727: pwd : '0250',  
1728: fase : '1',  
1729: ladoA : 'ENTR BR-158(B) (LARANJEIRAS DO SUL)',  
1730: ladoB : 'ENTR PR-473',  
1731: trecho : 'ENTR BR-158(B) (LARANJEIRAS DO SUL) - ENTR PR-473'  
1732: }, {  
1733: usr : '235',  
1734: pwd : '0455',  
1735: fase : '1',  
1736: ladoA : 'ACESSO ÁGUA MINERAL SANTA CLARA',  
1737: ladoB : 'ENTR PR-281(A) (P/CHOPINZINHO)',  
1738: trecho : 'ACESSO ÁGUA MINERAL SANTA CLARA - ENTR PR-281(A) (P/CHOPINZINHO)'  
1739: }, {  
1740: usr : '236',  
1741: pwd : '0500',  
1742: fase : '1',  
1743: ladoA : 'ENTR PR-468(B)',  
1744: ladoB : 'ENTR PR-180 (GOIO ERÊ)',  
1745: trecho : 'ENTR PR-468(B) - ENTR PR-180 (GOIO ERÊ)'  
1746: }, {  
1747: usr : '237',  
1748: pwd : '1230',  
1749: fase : '1',  
1750: ladoA : 'ENTR PR-431(B)',  
1751: ladoB : 'ENTR PR-092(A) (P/BARRA DO JACARÉ)',  
1752: trecho : 'ENTR PR-431(B) - ENTR PR-092(A) (P/BARRA DO JACARÉ)'  
1753: }, {  
1754: usr : '238',  
1755: pwd : '1140',  
1756: fase : '2',  
1757: ladoA : '',  
1758: ladoB : '',  
1759: trecho : 'ENTR SP-331(A) - ENTR SP-331(B)'  
1760: }, {  
1761: usr : '239',  
1762: pwd : '0230',  
1763: fase : '3',  
1764: ladoA : '',  
1765: ladoB : '',  
1766: trecho : 'ENTR BR-369 - ENTR SP-261 (P/ÁGUAS DE SANTA BÁRBARA)'  
1767: }, {  
1768: usr : '240',  
1769: pwd : '0710',  
1770: fase : '1',  
1771: ladoA : 'ACESSO PIRAPÓ',  
1772: ladoB : 'ACESSO CAMBIRA',  
1773: trecho : 'ACESSO PIRAPÓ - ACESSO CAMBIRA'  
1774: }, {  
1775: usr : '241',  
1776: pwd : '0410',  
1777: fase : '3',  
1778: ladoA : '',  
1779: ladoB : '',  
1780: trecho : 'ACESSO PARAÍBA DO SUL - ENTR RJ-135 (ANDRADE PINTO)'  
1781: }, {  
1782: usr : '242',  
1783: pwd : '2850',  
1784: fase : '4',  
1785: ladoA : '',  
1786: ladoB : '',  
1787: trecho : 'ENTR RJ-196 - ENTR RJ-182'  
1788: }, {  
1789: usr : '243',  
1790: pwd : '0590',  
1791: fase : '3',  
1792: ladoA : '',  
1793: ladoB : '',  
1794: trecho : 'ITAMONTE - ENTR BR-485 (GARGANTA DO REGISTRO)'  
1795: }, {  
1796: usr : '244',  
1797: pwd : '3350',  
1798: fase : '3',  
1799: ladoA : '',  
1800: ladoB : '',  
1801: trecho : 'ENTR RJ-149 (P/RIO CLARO) - ENTR BR-494(A) (ANGRA DOS REIS)'  
1802: }, {  
1803: usr : '245',  
1804: pwd : '0530',  
1805: fase : '3',  
1806: ladoA : '',  
1807: ladoB : '',  
1808: trecho : 'ENTR RJ-145 (P/BARRA DO PIRAÍ) - ENTR RJ-141 (DORANDIA)'  
1809: }, {  
1810: usr : '246',  
1811: pwd : '0170',  
1812: fase : '4',  
1813: ladoA : '',  
1814: ladoB : '',  
1815: trecho : 'ENTR BR-494 (ARANTINA) - AIURUOCA'  
1816: }, {  
1817: usr : '247',  
1818: pwd : '0170',  
1819: fase : '2',  
1820: ladoA : '',  
1821: ladoB : '',  
1822: trecho : 'ENTR RJ-198(B) - ENTR RJ-200 (MONTE ALEGRE)'  
1823: }, {  
1824: usr : '248',  
1825: pwd : '0350',  
1826: fase : '4',  
1827: ladoA : '',  
1828: ladoB : '',  
1829: trecho : 'ENTR RJ-202 - ENTR RJ-234 (P/PENEDO)'  
1830: }, {  
1831: usr : '249',  
1832: pwd : '0150',  
1833: fase : '3',  
1834: ladoA : '',  
1835: ladoB : '',  
1836: trecho : 'ENTR BR-482(B) - ENTR ES-181'  
1837: }, {  
1838: usr : '250',  
1839: pwd : '0210',  
1840: fase : '3',  
1841: ladoA : '',  
1842: ladoB : '',  
1843: trecho : 'ENTR BR-116/265 (MURIAÉ) - DIV MG/RJ'  
1844: }, {  
1845: usr : '251',  
1846: pwd : '1530',  
1847: fase : '2',  
1848: ladoA : '',  
1849: ladoB : '',  
1850: trecho : 'ENTR RJ-156 (P/VOLTA DO PIÃO) - ENTR BR-492(A)'  
1851: }, {  
1852: usr : '252',  
1853: pwd : '0065',  
1854: fase : '3',  
1855: ladoA : '',  
1856: ladoB : '',  
1857: trecho : 'ACESSO À JENIPABU - ENTR RN-160(B) (P/EXTREMOZ)'  
1858: }, {  
1859: usr : '253',  
1860: pwd : '0110',  
1861: fase : '3',  
1862: ladoA : '',  
1863: ladoB : '',  
1864: trecho : 'TAIPÚ - ENTR RN-064 (P/CEARÁ MIRIM)'  
1865: }, {  
1866: usr : '254',  
1867: pwd : '0055',  
1868: fase : '3',  
1869: ladoA : '',  
1870: ladoB : '',  
1871: trecho : 'ENTR CE-261 - DIV CE/RN'  
1872: }, {  
1873: usr : '255',  
1874: pwd : '0210',  
1875: fase : '3',  
1876: ladoA : '',  
1877: ladoB : '',  
1878: trecho : 'ENTR BR-230(A) (SANTA GERTRUDES) - ENTR PB-275(A) (P/SÃO JOSÉ DE ESPINHARA)'  
1879: }, {  
1880: usr : '256',  
1881: pwd : '1375',  
1882: fase : '3',  
1883: ladoA : '',  
1884: ladoB : '',  
1885: trecho : 'ACESSO USINA DE SAMUEL - INICIO PISTA DUPLA'  
1886: }, {  
1887: usr : '257',  
1888: pwd : '1510',  
1889: fase : '3',  
1890: ladoA : '',  
1891: ladoB : '',  
1892: trecho : 'VISTA ALEGRE DO ABUANÃ - VILA EXTREMA'  
1893: }, {  
1894: usr : '258',  
1895: pwd : '1300',  
1896: fase : '2',  
1897: ladoA : '',  
1898: ladoB : '',  
1899: trecho : 'ENTR BR-421 (ARIQUEMES) - ENTR RO-459(A) (ALTO PARAISO)'  
1900: }, {  
1901: usr : '259',  
1902: pwd : '0031',  
1903: fase : '2',  
1904: ladoA : '',  
1905: ladoB : '',  
1906: trecho : 'ENTR RO-135 (P/NOVA LONDRINA) - FIM DO TRECHO PAVIMENTADO'  
1907: }, {  
1908: usr : '260',  
1909: pwd : '1148',  
1910: fase : '2',  
1911: ladoA : '',  
1912: ladoB : '',  
1913: trecho : 'ENTR RO-471 (P/MINISTRO ANDREAZZA) - ENTR RO-479(A) (P/ROLIM DE MOURA)'  
1914: }, {  
1915: usr : '261',  
1916: pwd : '1090',  
1917: fase : '2',  
1918: ladoA : '',  
1919: ladoB : '',  
1920: trecho : 'ENTR RO-391 (POSTO GUAPORÉ) - MARCO RONDON'  
1921: }, {  
1922: usr : '262',  
1923: pwd : '0132',  
1924: fase : '1',  
1925: ladoA : 'PADRONAL',  
1926: ladoB : 'RIO 12 DE OUTUBRO',  
1927: trecho : 'PADRONAL - RIO 12 DE OUTUBRO'  
1928: }, {  
1929: usr : '263',  
1930: pwd : '4425',  
1931: fase : '1',  
1932: ladoA : 'ENTR ACESSO SUL DE CAPÃO DA CANOA',  
1933: ladoB : 'ENTR BR-290(A) (OSÓRIO)',  
1934: trecho : 'ENTR ACESSO SUL DE CAPÃO DA CANOA - ENTR BR-290(A) (OSÓRIO)'  
1935: }, {  
1936: usr : '264',  
1937: pwd : '0330',  
1938: fase : '1',  
1939: ladoA : 'ENTR BR-470',  
1940: ladoB : 'ENTR BR-470/116(A) (CANOAS)',  
1941: trecho : 'ENTR RSC-470 (P/ MONTENEGRO) - ENTR ERS-124 (P/ POLO PETROQUIMICO)'  
1942: }, {  
1943: usr : '265',  
1944: pwd : '0210',  
1945: fase : '1',  
1946: ladoA : 'ENTR BR-153(B)/RS-324 (P/PONTÃO)',  
1947: ladoB : 'ENTR BR-377(A)/386 (CARAZINHO)',  
1948: trecho : 'ENTR BRS-153(B)/ERS-324 (P/ PONTAO) - ENTR BRS-377(A)/386 (P/ CARAZINHO)'  
1949: }, {  
1950: usr : '266',  
1951: pwd : '0470',  
1952: fase : '1',  
1953: ladoA : 'ENTR RS-168 (P/SÃO PAULO DAS MISSÕES)',  
1954: ladoB : 'ENTR BR-472 (FRONT BRASIL/ARGENTINA) (PORTO XAVIER)',  
1955: trecho : 'ENTR RS-168 (P/SÃO PAULO DAS MISSÕES) - ENTR BR-472 (FRONT BRASIL/ARGENTINA) (PORTO XAVIER)'  
1956: }, {  
1957: usr : '267',  
1958: pwd : '0140',  
1959: fase : '1',  
1960: ladoA : 'A. BELTRÃO',  
1961: ladoB : 'ENTR BR-287 (SANTIAGO)',  
1962: trecho : 'ENTR ERS-533 (P/ CAPAO CIPO) - ENTR BRS-287 (CONTORNO DE SANTIAGO)'  
1963: }, {  
1964: usr : '268',  
1965: pwd : '1850',  
1966: fase : '3',  
1967: ladoA : 'ENTR BR-290(B)',  
1968: ladoB : 'ENTR BR-392',  
1969: trecho : 'ENTR BRS-290(B) (P/ SAO GABRIEL) - ENTR BRS-392 (P/ CACAPAVA DO SUL)'  
1970: }, {  
1971: usr : '269',  
1972: pwd : '1350',  
1973: fase : '1',  
1974: ladoA : 'ENTR BR-290(A) (P/ROSÁRIO DO SUL)',  
1975: ladoB : 'ENTR RS-640 (P/CACEQUI)',  
1976: trecho : 'ENTR BRS-158(A) (P/ AZEVEDO SODRE) - ENTR ERS-640 (P/ CACEQUI)'  
1977: }, {  
1978: usr : '270',  
1979: pwd : '0230',  
1980: fase : '1',  
1981: ladoA : 'ENTR BR-473 (SARANDI)',  
1982: ladoB : 'CURRAL ALTO',  
1983: trecho : 'ENTR RSC-473 (VILA SARANDI) - CURRAL ALTO'  
1984: }, {  
1985: usr : '271',  
1986: pwd : '0310',  
1987: fase : '1',  
1988: ladoA : 'ENTR RS-476 (LAJEADO GRANDE)',  
1989: ladoB : 'ENTR RS-110 (VÁRZEA DO CEDRO)',  
1990: trecho : 'ENTR ERS-476 (LAJEADO GRANDE) - ENTR ERS-110 (VARZEA DO CEDRO)'  
1991: }, {  
1992: usr : '272',  
1993: pwd : '0015',  
1994: fase : '1',  
1995: ladoA : 'ENTR BR-282(B)/470(A)',  
1996: ladoB : 'ENTR SC-455 (P/IBICUÍ)',  
1997: trecho : 'ENTR BR-282(B)/470(A) - ENTR SC-455 (P/IBICUÍ)'  
1998: }, {  
1999: usr : '273',  
2000: pwd : '3010',  
2001: fase : '1',  
2002: ladoA : 'DIV SC/RS (RIO PELOTAS)',  
2003: ladoB : 'ENTR BR-285(A) (P/VACARIA)',  
2004: trecho : 'DIVISA SC/RS (RIO PELOTAS) - ENTR BRS-285(A) (P/ VACARIA)'  
2005: }, {  
2006: usr : '274',  
2007: pwd : '0110',  
2008: fase : '1',  
2009: ladoA : 'ENTR BR-486 (BOM RETIRO)',  
2010: ladoB : 'ENTR SC-345(B) (P/URUBICI)',  
2011: trecho : 'ENTR BR-486 (BOM RETIRO) - ENTR SC-345(B) (P/URUBICI)'  
2012: }, {  
2013: usr : '275',  
2014: pwd : '0027',  
2015: fase : '1',  
2016: ladoA : 'ENTR SC-473 (P/ANCHIETA)',  
2017: ladoB : 'SÃO JOSÉ DO CEDRO',  
2018: trecho : 'ENTR SC-473 (P/ANCHIETA) - SÃO JOSÉ DO CEDRO'  
2019: }, {  
2020: usr : '276',  
2021: pwd : '0290',  
2022: fase : '3',  
2023: ladoA : '',  
2024: ladoB : '',  
2025: trecho : 'ENTR BR-280(A) - ENTR BR-280(B) (VITORINO)'  
2026: }, {  
2027: usr : '277',  
2028: pwd : '0612',  
2029: fase : '2',  
2030: ladoA : '',  
2031: ladoB : '',  
2032: trecho : 'ENTR BA-392 (ANTAS) - ENTR BA-220 (CÍCERO DANTAS)'  
2033: }, {  
2034: usr : '278',  
2035: pwd : '0190',  
2036: fase : '3',  
2037: ladoA : '',  
2038: ladoB : '',  
2039: trecho : 'ENTR SP-183 - ENTR BR-116(A) (P/LORENA)'  
2040: }, {  
2041: usr : '279',  
2042: pwd : '2360',  
2043: fase : '3',  
2044: ladoA : '',  
2045: ladoB : '',  
2046: trecho : 'ENTR SP-062 (P/ROSEIRA) - ENTR BR-383(A)/SP-132 (PINDAMONHANGABA)'  
2047: }, {  
2048: usr : '280',  
2049: pwd : '2470',  
2050: fase : '3',  
2051: ladoA : '',  
2052: ladoB : '',  
2053: trecho : 'ENTR SP-065 (P/IGARATÁ) - ENTR SP-056 (ARUJÁ)'  
2054: }, {  
2055: usr : '281',  
2056: pwd : '2630',  
2057: fase : '3',  
2058: ladoA : '',  
2059: ladoB : '',  
2060: trecho : 'ENTR BR-116(A) (JUQUIÁ) - ENTR SP-139 (REGISTRO)'  
2061: }, {  
2062: usr : '282',  
2063: pwd : '0130',  
2064: fase : '3',  
2065: ladoA : '',  
2066: ladoB : '',  
2067: trecho : 'ENTR BR-373(A) (ITAPETININGA) - ENTR BR-373(B)'  
2068: }, {  
2069: usr : '283',  
2070: pwd : '1077',  
2071: fase : '3',  
2072: ladoA : '',  
2073: ladoB : '',  
2074: trecho : 'ACESSO GUAIMBÉ - ENTR SP-333(A)'  
2075: }, {  
2076: usr : '284',  
2077: pwd : '1010',  
2078: fase : '2',  
2079: ladoA : '',  
2080: ladoB : '',  
2081: trecho : 'ENTR SP-355 - ENTR SP-425'  
2082: }, {  
2083: usr : '285',  
2084: pwd : '0110',  
2085: fase : '3',  
2086: ladoA : '',  
2087: ladoB : '',  
2088: trecho : 'ENTR SP-255 (ARARAQUARA) - ENTR BR-364(B)'  
2089: }, {  
2090: usr : '286',  
2091: pwd : '0783',  
2092: fase : '3',  
2093: ladoA : '',  
2094: ladoB : '',  
2095: trecho : 'ACESSO RINÓPOLIS - ENTR SP-294 (PARAPUÃ)'  
2096: }, {  
2097: usr : '287',  
2098: pwd : '0850',  
2099: fase : '3',  
2100: ladoA : '',  
2101: ladoB : '',  
2102: trecho : 'ENTR SP-065 (ATIBAIA) - ENTR SP-023 (MARIPORÃ)'  
2103: }, {  
2104: usr : '288',  
2105: pwd : '0010',  
2106: fase : '3',  
2107: ladoA : '',  
2108: ladoB : '',  
2109: trecho : 'ENTR BR-222/230(A)/PA-150 (MARABÁ) - ENTR PA-405'  
2110: }, {  
2111: usr : '289',  
2112: pwd : '0390',  
2113: fase : '2',  
2114: ladoA : '',  
2115: ladoB : '',  
2116: trecho : 'ENTR BR-010(A) (PORTO FRANCO) - ENTR BR-010(B)/230(A)/MA-138 (ESTREITO)'  
2117: }, {  
2118: usr : '290',  
2119: pwd : '1010',  
2120: fase : '3',  
2121: ladoA : '',  
2122: ladoB : '',  
2123: trecho : 'ENTR MA-012/375 (SÃO RAIMUNDO DAS MANGABEIRAS) - ENTR MA-006(A)'  
2124: }, {  
2125: usr : '291',  
2126: pwd : '0290',  
2127: fase : '3',  
2128: ladoA : '',  
2129: ladoB : '',  
2130: trecho : 'FIM PISTA DUPLA - GURUPÍ - ENTR BR-242(B)/TO-280'  
2131: }, {  
2132: usr : '292',  
2133: pwd : '0330',  
2134: fase : '4',  
2135: ladoA : '',  
2136: ladoB : '',  
2137: trecho : 'ENTR BR-020(A)/135(B)/BA-020 - ENTR BR-020(B)'  
2138: }, {  
2139: usr : '293',  
2140: pwd : '0272',  
2141: fase : '3',  
2142: ladoA : '',  
2143: ladoB : '',  
2144: trecho : 'INDIANÓPOLIS - VALE DO SONHO'  
2145: }, {  
2146: usr : '294',  
2147: pwd : '0210',  
2148: fase : '4',  
2149: ladoA : '',  
2150: ladoB : '',  
2151: trecho : 'ENTR GO-217 (P/MAIRIPOTABA) - ENTR GO-320 (INDIARA)'  
2152: }, {  
2153: usr : '295',  
2154: pwd : '0370',  
2155: fase : '1',  
2156: ladoA : 'ENTR BR-153 (P/TIBAGI/IPIRANGA)',  
2157: ladoB : 'ENTR BR-373(A)/487(A) (CAETANO)',  
2158: trecho : 'ENTR BR-153 (P/TIBAGI/IPIRANGA) - ENTR BR-373(A)/487(A) (CAETANO)'  
2159: }, {  
2160: usr : '296',  
2161: pwd : '0390',  
2162: fase : '1',  
2163: ladoA : 'RIO ANHANDUÍ',  
2164: ladoB : 'ENTR BR-262(A) (CAMPO GRANDE)',  
2165: trecho : 'RIO ANHANDUÍ - ENTR BR-262(A) (CAMPO GRANDE)'  
2166: }, {  
2167: usr : '297',  
2168: pwd : '0434',  
2169: fase : '3',  
2170: ladoA : '',  
2171: ladoB : '',  
2172: trecho : 'ACESSO ITAJÁ (GO) - ENTR MS-431 (P/SÃO JOÃO DO APORÉ)'  
2173: }, {  
2174: usr : '298',  
2175: pwd : '0930',  
2176: fase : '3',  
2177: ladoA : '',  
2178: ladoB : '',  
2179: trecho : 'FNM (ENTR PISTA INVERSA) - ENTR BR-116(A)/493/RJ-109'  
2180: }, {  
2181: usr : '299',  
2182: pwd : '0110',  
2183: fase : '1',  
2184: ladoA : 'ENTR SC-418 (P/POMERODE)',  
2185: ladoB : 'ENTR BR-477(B) (P/TIMBÓ)',  
2186: trecho : 'ENTR SC-418 (P/POMERODE) - ENTR BR-477(B) (P/TIMBÓ)'  
2187: }, {  
2188: usr : '300',  
2189: pwd : '0110',  
2190: fase : '1',  
2191: ladoA : 'ENTR BR-116(B)',  
2192: ladoB : 'P/TRES BARRAS',  
2193: trecho : 'ENTR BR-116(B) - P/TRES BARRAS'  
2194: }, {  
2195: usr : '301',  
2196: pwd : '0150',  
2197: fase : '2',  
2198: ladoA : '',  
2199: ladoB : '',  
2200: trecho : 'DIV PI/BA - ENTR BR-235/BA-161 (REMANSO)'  
2201: }, {  
2202: usr : '302',  
2203: pwd : '0770',  
2204: fase : '2',  
2205: ladoA : '',  
2206: ladoB : '',  
2207: trecho : 'ENTR BR-369(A) (BARBOSA FERRAZ) - ENTR BR-158(A)/369(B) (ANEL VIÁRIO CAMPO MOURÃO)'  
2208: } ],  
2209:  
2210: user\_logado : {}  
2211:};

## .\mobile\www\js\motivoEscolhaRota.js

1:var lista\_motivo\_escolha\_rota = [  
 2: {id : '1', nome : "Asfalto/Sinalização", simples : true, carga : true},  
 3: {id : '2', nome : "Caminho mais curto", simples : true, carga : true},  
 4: {id : '3', nome : "Ordens da Empresa", simples : false, carga : true},  
 5: {id : '4', nome : "Proximidade hotéis/postos", simples : true, carga : true},  
 6: {id : '5', nome : "Segurança", simples : true, carga : true},  
 7: {id : '6', nome : "Turismo/Paisagem", simples : true, carga : false},  
 8: {id : '7', nome : "Outros", simples : true, carga : true},  
 9: {id : '8', nome : "Caminho mais rápido", simples : true, carga : true},  
 10: {id : '9', nome : "Ausência de pedágio", simples : true, carga : true},  
 11: {id : '10', nome : "Ponto obrigatório de passagem", simples : true, carga : false}  
 12:];

## .\mobile\www\js\motivoViagem.js

1:var lista\_motivo\_viagem = [  
 2: {id : '1', nome : "Compras", simples : true},  
 3: {id : '2', nome : "Estudo", simples : true},  
 4: {id : '3', nome : "Lazer", simples : true},  
 5: {id : '4', nome : "Pessoais", simples : false},  
 6: {id : '5', nome : "Saúde", simples : true},  
 7: {id : '6', nome : "Trabalho", simples : true},  
 8: {id : '7', nome : "Outros", simples : true},  
 9: {id : '8', nome : "Negócios", simples : true}  
 10:];

## .\mobile\www\js\myDb.js

1:myDb = {  
 2:  
 3: camposNaoExportaveisJson : [ 'id', 'uuid', 'login', 'erro', 'duracaoPesq' ],  
 4:  
 5: tabelaOD : [  
 6: {field : 'id', type : 'text primary key'},  
 7: {field : 'estaNoNote', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 8: {field : 'cancelado', type : 'integer'},  
 9: {field : 'idPosto', type : 'integer', notNull: true},  
 10: {field : 'sentido', type : 'text', notNull: true},  
 11: {field : 'idIpad', type : 'text', notNull: true},  
 12: // {field : 'uuid', type : 'text'},  
 13: {field : 'login', type : 'text'}, // idPost + sentido (redundante?)  
 14: {field : 'dataIniPesq', type : 'text'},  
 15: {field : 'dataFimPesq', type : 'text'},  
 16: {field : 'placa', type : 'text'},  
 17: {field : 'anoDeFabricacao', type : 'integer', notNull: true},  
 18: {field : 'tipo', type : 'text', notNull: true},  
 19: {field : 'idOrigemPais', type : 'integer', notNull: true},  
 20: {field : 'idOrigemMunicipio', type : 'integer'},  
 21: {field : 'idDestinoPais', type : 'integer', notNull: true},  
 22: {field : 'idDestinoMunicipio', type : 'integer'},  
 23: {field : 'idMotivoDeEscolhaDaRota', type : 'integer'},  
 24: {field : 'frequenciaQtd', type : 'integer'},  
 25: {field : 'frequenciaPeriodo', type : 'text', notNull: true},  
 26: {field : 'idPropriedadesDoVeiculo', type : 'integer', notNull: true},  
 27: {field : 'placaEstrangeira', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 28: {field : 'idPaisPlacaEstrangeira', type : 'integer'},  
 29: {field : 'idCombustivel', type : 'integer', notNull: true},  
 30: {field : 'categoria', type : 'text', notNull: true},  
 31: {field : 'possuiReboque', type : 'integer'},  
 32: {field : 'numeroDePessoasNoVeiculo', type : 'integer'},  
 33: {field : 'numeroDePessoasATrabalho', type : 'integer'},  
 34: {field : 'idRendaMedia', type : 'integer'},  
 35: {field : 'idMotivoDaViagem', type : 'integer'},  
 36: {field : 'tipoCaminhao', type : 'text'},  
 37: {field : 'idTipoDeContainer', type : 'integer'},  
 38: {field : 'idTipoCarroceria', type : 'integer'},  
 39: {field : 'rntrc', type : 'text'},  
 40: {field : 'possuiCargaPerigosa', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 41: {field : 'idNumeroDeRisco', type : 'integer'},  
 42: {field : 'idNumeroDaOnu', type : 'integer'},  
 43: {field : 'idAgregado', type : 'integer'},  
 44: {field : 'placaVermelha', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 45: {field : 'idTipoDeViagemOuServico', type : 'integer'},  
 46: {field : 'pesoDaCarga', type : 'real'},  
 47: {field : 'valorDoFrete', type : 'real'},  
 48: {field : 'utilizaParadaEspecial', type : 'integer'}, // Boolean 1->true || false, otherwise;  
 49: {field : 'idProduto', type : 'integer'},  
 50: {field : 'idCargaAnterior', type : 'integer'},  
 51: {field : 'valorDaCarga', type : 'real'},  
 52: {field : 'municipioEmbarqueCarga', type : 'integer'},  
 53: {field : 'idLocalEmbarqueCarga', type : 'integer'},  
 54: {field : 'municipioDesembarqueCarga', type : 'integer'},  
 55: {field : 'idLocalDesembarqueCarga', type : 'integer'},  
 56: {field : 'indoPegarCarga', type : 'integer'},  
 57: {field : 'paradaObrigatoriaMunicipio1', type : 'integer'},  
 58: {field : 'paradaObrigatoriaMunicipio2', type : 'integer'},  
 59: {field : 'idPerguntaExtra', type : 'integer'},  
 60: {field : 'idPerguntaExtra2', type : 'integer'},  
 61: {field : 'erro', type : 'text'},  
 62: {field : 'duracaoPesq', type : 'integer'},  
 63: {field : 'treinamento', type : 'integer', notNull: true}// Boolean 1->true || false, otherwise;  
 64: ],  
 65:  
 66: fieldExists : function(str) {  
 67: var found = false;  
 68: $.each(myDb.tabelaOD, function(index, item) {  
 69: if (item.field == str) {  
 70: found = true;  
 71: return false; // breaks $.each  
 72: }  
 73: });  
 74: return found;  
 75: },  
 76:  
 77: cretateTblDados : function(cb) {  
 78: app.logger.log("criando tabela: tblDados");  
 79: app.database.transaction(function(tx) {  
 80: var sql = "CREATE TABLE IF NOT EXISTS tblDados ( ";  
 81: $.each(myDb.tabelaOD, function(index, item) {  
 82: sql += item.field + " " + item.type;  
 83: if (index < myDb.tabelaOD.length - 1)  
 84: sql += ", ";  
 85: });  
 86: sql += ");";  
 87: tx.executeSql(sql);  
 88: }, function(e) {  
 89: app.logger.log('ERRO: ' + e.message);  
 90: }, function() {  
 91: app.logger.log("tabela criada: tblDados");  
 92: if(util.isFunction(cb)){  
 93: cb();  
 94: }  
 95: });  
 96:  
 97: },  
 98:  
 99: createTblSchema : function(cb) {  
 100: app.logger.log("criando tabela: tblSchema");  
 101: app.database.transaction(function(tx) {  
 102: var sql = "CREATE TABLE IF NOT EXISTS tblSchema (versao integer);";  
 103: tx.executeSql(sql);  
 104: }, function(e) {  
 105: app.logger.log('ERRO: ' + e.message);  
 106: }, function() {  
 107: app.logger.log("tabela criada: tblSchema");   
 108: if(util.isFunction(cb)){  
 109: cb();  
 110: }  
 111: });  
 112:  
 113: },  
 114:   
 115: updateSchema : function(cb){  
 116: app.logger.log("update schema");  
 117: myDb.createTblSchema(function(){  
 118: var versaoAtual = 0;  
 119: app.database.transaction(function(tx) {  
 120: tx.executeSql("SELECT versao from tblSchema LIMIT 1;", [], function(tx, res) {  
 121: if( res.rows.length > 0) {  
 122: versaoAtual = res.rows.item(0).versao;  
 123: }  
 124:   
 125: if(versaoAtual == 0){  
 126: tx.executeSql("INSERT INTO tblSchema VALUES(" + app.versaoBD + ");");  
 127: } else if (versaoAtual != app.versaoBD){  
 128: tx.executeSql("UPDATE tblSchema SET versao = " + app.versaoBD + ");");  
 129:   
 130: if (versaoAtual < 1 ){  
 131: tx.executeSql("ALTER TABLE tblDados ADD COLUMN idPerguntaExtra2 integer;");  
 132: }  
 133: }  
 134:   
 135:   
 136: });  
 137:   
 138: }, function(e){  
 139: app.logger.log('ERRO: ' + e.message);  
 140: }, function(){  
 141: app.logger.log("BD schema anterior: " + versaoAtual);  
 142: app.logger.log("BD schema atual: " + app.versaoBD);  
 143: if(util.isFunction(cb)){  
 144: cb();  
 145: }  
 146: });  
 147: });  
 148: },  
 149:  
 150: /\*\*  
 151: \* Inserts a registro variable into database  
 152: \* @param Registro reg registro to be inserted into database  
 153: \* @param Function fail error callback  
 154: \* @param Function success callback  
 155: \*/  
 156: insertRegistro : function(reg, fail, success) {  
 157: if (reg.id != null){  
 158: app.logger.log("inserindo registro: " + reg.id);  
 159: try {  
 160: app.database.transaction(function(tx) {  
 161:  
 162: var fields = "(";  
 163: var places = "(";  
 164: var values = [];  
 165: $.each(myDb.tabelaOD, function(index, item) {  
 166: fields += item.field;  
 167: places += "?";  
 168: var value = reg[item.field];  
 169: values.push((value == undefined ? null : value));  
 170: if (index < myDb.tabelaOD.length - 1) {  
 171: fields += ", ";  
 172: places += ", ";  
 173: }  
 174: });  
 175: fields += ")";  
 176: places += ");";  
 177:  
 178: var sql = "INSERT INTO tblDados " + fields + " VALUES " + places;  
 179:  
 180: tx.executeSql(sql, values, function(tx, res) {  
 181: app.logger.log('id inserido no banco de dados: ' + res.insertId);  
 182: });  
 183: },  
 184: // transaction fail  
 185: function(e) {  
 186: // inseriu = false;  
 187: app.logger.log('ERRO ao inserir registro (' + reg.id + '): ' + e.message);  
 188: fail(e);  
 189: },  
 190: // transaction success  
 191: function() {  
 192: // inseriu = true;  
 193: app.logger.log("registro inserido: " + reg.id);  
 194: success();  
 195: });  
 196: } catch (e) {  
 197: fail(e);  
 198: }  
 199: } else {  
 200: app.logger.log("Tentativa de inserção de registro sem id");  
 201: }  
 202:   
 203: },  
 204:   
 205: sanitize: function(cb){  
 206: app.database.transaction(function(tx) {  
 207: var sql = "DELETE from tblDados WHERE id is null;";  
 208: tx.executeSql(sql);  
 209: },  
 210: //fail  
 211: function(e){  
 212: app.logger.log('ERRO ao limpar a base de dados: ' + e.message);  
 213: },  
 214: //success  
 215: function(){  
 216: app.logger.log('Base de dados limpa');  
 217: if (util.isFunction(cb)){  
 218: cb();  
 219: }  
 220: })  
 221: },  
 222:  
 223:/\*  
 224:select date(dataIniPesq) as 'diaPesq',AVG(duracaoPesq) as 'mediaDia',SUM(duracaoPesq) as 'somaDia',COUNT(id) as 'qtdDia', MAX(duracaoPesq) as 'maxTempoDia', MIN(duracaoPesq) as 'minTempoDia'  
 225:from tblDados  
 226:WHERE cancelado = 0  
 227:GROUP by date(dataIniPesq)  
 228:ORDER BY diaPesq DESC;  
 229:\*/  
 230: selectDuracoesDiaRegistro : function(fail, success) {  
 231: app.logger.log("(selectDuracoesDiaRegistro) buscando no registro");  
 232: var linhas = [];  
 233: var treinamento = (app.isTreinamento)? 1 : 0;  
 234: try {  
 235: app.database.transaction(function(tx) {  
 236:  
 237: var sql = "SELECT DATE(dataIniPesq) as 'diaPesq'," +  
 238: " AVG(duracaoPesq) as 'mediaDia'," +  
 239: " SUM(duracaoPesq) as 'somaDia',"+  
 240: " COUNT(id) as 'qtdDia'," +  
 241: " MAX(duracaoPesq) as 'maxTempoDia'," +  
 242: " MIN(duracaoPesq) as 'minTempoDia'" +  
 243: " FROM tblDados WHERE cancelado = 0" +  
 244: " AND treinamento = " + treinamento +  
 245: " AND idPosto = " + app.posto +  
 246: " GROUP by DATE(dataIniPesq) ORDER BY diaPesq DESC";  
 247:  
 248: tx.executeSql(sql, [], function(tx, res) {  
 249: app.logger.log('qtd linhas select: ' + res.rows.length);  
 250: for (i = 0; i < res.rows.length; i++) {  
 251: var elem = res.rows.item(i);  
 252: linhas.push(elem);  
 253: app.logger.log('item ' + i + ':' + elem.diaPesq + '; ' + elem.mediaDia + '; ' + elem.somaDia + '; ' + elem.qtdDia + '; ' + elem.maxTempoDia + '; ' + elem.minTempoDia);  
 254: }  
 255: });  
 256: },  
 257: // transaction fail  
 258: function(e) {  
 259: app.logger.log('ERRO ao buscar tempos das pesquisas: ' + e.message);  
 260: fail(e);  
 261: },  
 262: // transaction success  
 263: function() {  
 264: app.logger.log("Busca dos tempos das pesquisas realizado com sucesso");  
 265: app.sumario\_lista = linhas;  
 266: success();  
 267: });  
 268: } catch (e) {  
 269: fail(e);  
 270: }  
 271: },  
 272:  
 273: selectUltimaPesquisaValida : function(fail, success) {  
 274: app.logger.log("(selectUltimaPesquisaValida) buscando no registro");  
 275: var ultimoRegistro = [];  
 276: var treinamento = (app.isTreinamento)? 1 : 0;  
 277: try {  
 278: app.database.transaction(function(tx) {  
 279:  
 280: var sql = "SELECT dataIniPesq," +  
 281: " duracaoPesq" +  
 282: " FROM tblDados WHERE cancelado = 0" +  
 283: " AND treinamento = " + treinamento +  
 284: " AND idPosto = " + app.posto +  
 285: " ORDER BY dataIniPesq DESC" +  
 286: " LIMIT 1";  
 287:  
 288: tx.executeSql(sql, [], function(tx, res) {  
 289: if( res.rows.length > 0) {  
 290: ultimoRegistro.push(res.rows.item(0).dataIniPesq);  
 291: ultimoRegistro.push(res.rows.item(0).duracaoPesq);  
 292: app.logger.log('Ultimo registro: dataIniPesq = ' + ultimoRegistro[0] + '; duracaoPesq = ' + ultimoRegistro[1]);  
 293: }  
 294: });  
 295: },  
 296: // transaction fail  
 297: function(e) {  
 298: app.logger.log('ERRO ao buscar última pesquisa: ' + e.message);  
 299: fail(e);  
 300: },  
 301: // transaction success  
 302: function() {  
 303: app.logger.log("Busca da última pesquisa realizada com sucesso");  
 304: app.ultima\_pesquisa = ultimoRegistro;  
 305: success();  
 306: });  
 307: } catch (e) {  
 308: fail(e);  
 309: }  
 310: },  
 311:  
 312:/\*  
 313:select date(dataIniPesq) as 'diaPesq',AVG(duracaoPesq) as 'mediaDia',SUM(duracaoPesq) as 'somaDia',COUNT(id) as 'qtdDia'  
 314:from tblDados  
 315:WHERE cancelado = 1  
 316:GROUP by date(dataIniPesq)  
 317:ORDER BY diaPesq DESC;  
 318:\*/  
 319: selectRegistrosCancelados : function(fail, success) {  
 320: app.logger.log("(selectRegistrosCancelados) buscando no registro");  
 321: var linhas = [];  
 322: var treinamento = (app.isTreinamento)? 1 : 0;  
 323: try {  
 324: app.database.transaction(function(tx) {  
 325:  
 326: var sql = "SELECT DATE(dataIniPesq) as 'diaPesq'," +  
 327: " AVG(duracaoPesq) as 'mediaDia'," +  
 328: " SUM(duracaoPesq) as 'somaDia',"+  
 329: " COUNT(id) as 'qtdDia'" +  
 330: " FROM tblDados WHERE cancelado = 1" +  
 331: " AND treinamento = " + treinamento +  
 332: " AND idPosto = " + app.posto +  
 333: " GROUP by DATE(dataIniPesq) ORDER BY diaPesq DESC";  
 334:  
 335: tx.executeSql(sql, [], function(tx, res) {  
 336: app.logger.log('qtd linhas select: ' + res.rows.length);  
 337: for (i = 0; i < res.rows.length; i++) {  
 338: var elem = res.rows.item(i);  
 339: linhas.push(elem);  
 340: app.logger.log('item ' + i + ':' + elem.diaPesq + '; ' + elem.mediaDia + '; ' + elem.somaDia + '; ' + elem.qtdDia);  
 341: }  
 342: });  
 343: },  
 344: // transaction fail  
 345: function(e) {  
 346: app.logger.log('ERRO ao buscar pesquisas canceladas: ' + e.message);  
 347: fail(e);  
 348: },  
 349: // transaction success  
 350: function() {  
 351: app.logger.log("Busca das pesquisas canceladas realizada com sucesso");  
 352: app.sumario\_lista\_cancelados = linhas;  
 353: success();  
 354: });  
 355: } catch (e) {  
 356: fail(e);  
 357: }  
 358: },  
 359:  
 360: exportaDbToJson : function(writer, fail, success) {  
 361: app.logger.log("exportando Json: ");  
 362: var treinamento = (app.isTreinamento)? 1 : 0;  
 363: try {  
 364: app.database.transaction(function(tx) {  
 365: var fields = "";  
 366: $.each(myDb.tabelaOD, function(index, item) {  
 367: fields += item.field;  
 368: if (index < myDb.tabelaOD.length - 1) {  
 369: fields += ", ";  
 370: }  
 371: });  
 372:  
 373: var sql = "SELECT " + fields + " FROM tblDados WHERE cancelado = 0 AND treinamento = " + treinamento + " ;";  
 374:  
 375: tx.executeSql(sql, [], function(tx, res) {  
 376: for (var rowIndex = 0; rowIndex < res.rows.length; rowIndex++) {  
 377: var rowDB = res.rows.item(rowIndex);  
 378: var rowJson = "{";  
 379: if (rowIndex == 0) {  
 380: rowJson = "[{";  
 381: }  
 382: $.each(myDb.tabelaOD, function(index, item) {  
 383: if (!util.contains(item.field, myDb.camposNaoExportaveisJson)) {  
 384: var value = rowDB[item.field];  
 385: if (value != null) {  
 386: // app.logger.log('FIELD (' + item.type + '): ' + item.field + '\tVALUE (' + typeof value  
 387: // + '): ' + value);  
 388: var typeValue = typeof value;  
 389: if (item.type == 'text') {  
 390: value = '"' + value + '"';  
 391: } else if (item.type == 'integer' && (typeValue != 'number')) {  
 392: if (typeValue == 'boolean') {  
 393: value = Number(value);  
 394: } else if (typeValue == 'string') {  
 395: value = ((value == 'true') ? 1 : ((value == 'false') ? 0 : (isNaN(value) ? ('"'  
 396: + value + '"') : Number(value))));  
 397: }  
 398: }  
 399: } else if (item.notNull) {  
 400: if (item.type == 'text') {  
 401: value = '""';  
 402: } else {  
 403: value = 0;  
 404: }  
 405: }  
 406: rowJson += '"' + item.field + '":' + value;  
 407: if (index < myDb.tabelaOD.length - 2) { // (length - 2) pq o último campo ('erro') tb não será exportado  
 408: rowJson += ",";  
 409: }  
 410: }  
 411: });  
 412: rowJson += "}";  
 413: if (rowIndex < res.rows.length - 1) {  
 414: rowJson += ",";  
 415: } else {  
 416: rowJson += "]";  
 417: }  
 418: writer.appendRow(rowJson);  
 419: }  
 420: });  
 421:  
 422: },  
 423: // transaction fail  
 424: function(e) {  
 425: app.logger.log('ERRO ao exportar os dados para o formato JSON: ' + e.message);  
 426: if (util.isFunction(fail)) {  
 427: fail(e);  
 428: }  
 429: },  
 430: // transaction success  
 431: function() {  
 432: app.logger.log("Dados exportados para o formato JSON.");  
 433: if (util.isFunction(success)) {  
 434: success();  
 435: }  
 436: });  
 437: } catch (e) {  
 438: if (util.isFunction(fail)) {  
 439: fail(e);  
 440: }  
 441: }  
 442: }  
 443:  
 444:};

## .\mobile\www\js\myDialogs.js

1:/\* global util \*/  
 2:  
 3:var myDialogs = {  
 4:  
 5: /\*\*  
 6: \* Shows an alert dialog  
 7: \* @param String txt Message  
 8: \* @param Strint title Dialog Title  
 9: \* @param Function cb callback function  
 10: \* @param String type optional dialog type ['info', 'error']  
 11: \*/  
 12: alert : function (txt, title, cb, type){  
 13: title = (title == null)? "Aviso!" : title;  
 14: var icon;  
 15: var btn;  
 16: switch (type){  
 17: case 'info' :  
 18: icon = 'fa fa-info-circle';  
 19: btn = 'btn-warning';  
 20: break;  
 21: case 'error' :  
 22: icon = 'fa fa-exclamation-triangle';  
 23: btn = 'btn-danger';  
 24: break;  
 25: default:  
 26: icon = null;  
 27: btn = 'btn-info';  
 28:   
 29: }  
 30: var d = $.alert({  
 31: title : title,  
 32: content : txt,  
 33: icon : icon,  
 34: confirmButton : "Ok",  
 35: confirmButtonClass : btn,  
 36: animation : 'none',  
 37: closeAnimation: 'none',  
 38: animationBounce : 1, //1 is none  
 39: confirm : function(){  
 40: if(util.isFunction(cb)){  
 41: cb();  
 42: }  
 43: },  
 44:   
 45: });  
 46: d.$body.addClass("ui-page-theme-a").css('margin-top', '0px');  
 47:   
 48: },  
 49:   
 50: /\*\*  
 51: \* Shows a confirmation dialog  
 52: \* @param String txt Message  
 53: \* @param Function cbOK Ok button click callback  
 54: \* @param Function cbCancel Cancel button click callback  
 55: \* @param String title Dialog title  
 56: \* @param String btnOK Ok button text  
 57: \* @param String btnCancel Cancel button text  
 58: \*/  
 59: confirm : function(txt, cbOK, cbCancel, title, btnOK, btnCancel){  
 60: title = (title == null)? "Confirmação." : title;  
 61: btnOK = (btnOK == null)? "Confirmar" : btnOK;  
 62: btnCancel = (btnCancel == null)? "Cancelar" : btnCancel;  
 63: $.confirm({  
 64: title : title,  
 65: content: txt,  
 66: confirmButton : btnOK,  
 67: confirmButtonClass : "btn-primary",//"ui-btn ui-corner-all ui-btn-b",  
 68: cancelButton : btnCancel,  
 69: cancelButtonClass : "btn-info",//"ui-btn ui-corner-all ui-btn-b",  
 70: animation : 'none',  
 71: closeAnimation: 'none',  
 72: animationBounce : 1, //1 is none  
 73: confirm : function(){  
 74: if(util.isFunction(cbOK)){  
 75: cbOK();  
 76: }  
 77: },  
 78: cancel: function(){  
 79: if(util.isFunction(cbCancel)){  
 80: cbCancel();  
 81: }  
 82: }  
 83: }).$body.addClass("ui-page-theme-a")/\*8.children(".buttons").css("width","100%")\*/;  
 84: },  
 85:   
 86: /\*\*  
 87: \* Shows a prompt dialog  
 88: \* @param String txt Message  
 89: \* @param Function cb Callback function that receives the entered data  
 90: \* @param String title Dialog title  
 91: \* @param String btnOK Ok button text  
 92: \* @param String btnCancel Cancel button text  
 93: \* @param String placeholder Placeholder text for the input box  
 94: \* @param String inputType Type of the input -> ['text', 'password', 'number']  
 95: \*/  
 96: prompt : function(txt, cb, title, btnOK, btnCancel, placeholder, inputType){  
 97: title = (title == null)? "Entrada de dados." : title;  
 98: btnOK = (btnOK == null)? "Entrar" : btnOK;  
 99: btnCancel = (btnCancel == null)? "Cancelar" : btnCancel;  
 100: placeholder = (placeholder == null)? "Entre com os dados" : placeholder;  
 101: if (inputType != 'text' && inputType != 'password' && inputType != 'number'){   
 102: inputType = 'text';  
 103: }  
 104: var dialog = $.confirm({  
 105: title : title,  
 106: content : txt,  
 107: confirmButton : btnOK,  
 108: confirmButtonClass : "btn-primary",//"ui-btn ui-corner-all ui-btn-b",  
 109: cancelButton : btnCancel,  
 110: cancelButtonClass : "btn-info",//"ui-btn ui-corner-all ui-btn-b",  
 111: animation : 'none',  
 112: closeAnimation: 'none',  
 113: animationBounce : 1, //1 is none  
 114: confirm : function(){  
 115: var enteredVal = this.$content.find('input').val(); // get the input value.  
 116: if (enteredVal.trim() == '') { // validate it.  
 117: return false; // dont close the dialog. (and maybe show some error.)  
 118: }  
 119: if(util.isFunction(cb)){  
 120: cb(enteredVal);  
 121: }  
 122: },  
 123: });  
 124: dialog.$content.append('<p><input style="width:100%;" id="prompt\_input" autofocus type="'+   
 125: inputType +'" placeholder="' + placeholder + '" data-theme="a"></p>');  
 126: dialog.$content.find("#prompt\_input").textinput();//textinput({theme: "a"});  
 127: dialog.$body.addClass("ui-page-theme-a");  
 128: //dialog.$body.children(".buttons").css("width","100%");  
 129:   
 130: },  
 131:   
 132:};  
 133:  
 134:

## .\mobile\www\js\paises.js

1:var paises = {  
 2:   
 3: listados : function(){  
 4: // return paises.america\_sul.concat(paises.america\_norte\_central\_caribe,  
 5: // paises.africa, paises.asia, paises.europa, paises.oceania);  
 6:   
 7: return paises.america\_sul;  
 8: },  
 9:  
 10: america\_sul : ['Brasil', 'Argentina', 'Bolívia', 'Chile', 'Colômbia',   
 11: 'Equador', 'Guiana', 'Guiana Francesa', 'Paraguai', 'Peru', 'Suriname', 'Uruguai', 'Venezuela'],  
 12:  
 13: america\_norte\_central\_caribe : ['Antígua e Barbuda', 'Bahamas', 'Barbados',  
 14: 'Belize', 'Canadá', 'Costa Rica', 'Cuba', 'Domínica', 'República Dominicana',  
 15: 'El Salvador', 'Granada', 'Guatemala', 'Haiti', 'Honduras', 'Jamaica',   
 16: 'México', 'Nicarágua', 'Panamá', 'São Cristóvão e Neves', 'Santa Lúcia',   
 17: 'São Vicente e Granadinas', 'Trinidad e Tobago', 'Estados Unidos'],  
 18:  
 19: africa : ['Argélia', 'Angola', 'Benim', 'Botsuana', 'Burkina Faso',  
 20: 'Burundi', 'Camarões', 'Cabo Verde', 'República Centro-Africana',  
 21: 'Chade', 'Comores', 'República Democrática do Congo', 'Djibouti',  
 22: 'Egito', 'Guiné Equatorial', 'Eritreia', 'Etiópia', 'Gabão', 'Gâmbia',  
 23: 'Gana', 'Guiné', 'Guiné-Bissau', 'Costa do Marfim', 'Quénia', 'Lesoto',  
 24: 'Libéria', 'Líbia', 'Madagascar', 'Malawi', 'Mali', 'Mauritânia',  
 25: 'Mauritius', 'Marrocos', 'Moçambique', 'Namíbia', 'Níger', 'Nigéria',  
 26: 'Ruanda', 'São Tomé e Príncipe', 'Senegal', 'Seicheles', 'Serra Leoa',  
 27: 'Somália', 'África do Sul', 'Sudão do Sul', 'Sudão', 'Suazilândia',  
 28: 'Tanzânia', 'Togo', 'Tunísia', 'Uganda', 'Zâmbia', 'Zimbabwe'],  
 29:  
 30: europa : ['Albânia', 'Andorra', 'Arménia', 'Áustria', 'Azerbaijão',  
 31: 'Bielorrússia', 'Bélgica', 'Bósnia e Herzegovina', 'Bulgária',  
 32: 'Croácia', 'Chipre', 'República Tcheca', 'Dinamarca', 'Estónia',  
 33: 'Finlândia', 'França', 'Geórgia', 'Alemanha', 'Grécia', 'Hungria',  
 34: 'Islândia', 'Irlanda', 'Itália', 'Letónia', 'Liechtenstein ',  
 35: 'Lituânia', 'Luxemburgo', 'Macedónia', 'Malta', 'Moldávia', 'Mónaco',  
 36: 'Montenegro', 'Países Baixos', 'Noruega', 'Polónia', 'Portugal',  
 37: 'Roménia', 'São Marino', 'Sérvia', 'Eslováquia', 'Eslovénia',  
 38: 'Espanha', 'Suécia', 'Suíça', 'Ucrânia', 'Reino Unido', 'Vaticano'],  
 39:  
 40: oceania : ['Austrália', 'Fiji', 'Kiribati', 'Ilhas Marshall', 'Micronésia',  
 41: 'Nauru', 'Nova Zelândia', 'Palau', 'Papua-Nova Guiné', 'Samoa',  
 42: 'Ilhas Salomão', 'Tonga', 'Tuvalu', 'Vanuatu'],  
 43:  
 44: asia : ['Afeganistão', 'Bahrein', 'Bangladeche', 'Butão', 'Brunei', 'Burma',   
 45: 'Camboja', 'China', 'Timor Leste', 'India', 'Indonésia', 'Irã', 'Iraque',   
 46: 'Israel', 'Japão', 'Jordânia', 'Cazaquistão', 'Coréia do Norte',   
 47: 'Coréia do Sul', 'Kuwait', 'Quirguistão', 'Laos', 'Líbano', 'Malásia',   
 48: 'Maldivas', 'Mongólia', 'Nepal', 'Omã', 'Paquistão', 'Filipinas', 'Qatar',   
 49: 'Rússia', 'Arábia Saudita', 'Singapura', 'Sri Lanka', 'Síria', 'Tajaquistão',   
 50: 'Tailândia', 'Turquia', 'Turcomenistão', 'Emirados Arábes Unidos',   
 51: 'Uzbequistão', 'Vietname', 'Iémen']  
 52:};  
 53:  
 54:var lista\_estados = ['AC', 'AL', 'AM', 'AP', 'BA', 'CE', 'DF', 'ES', 'GO', 'MA', 'MG', 'MS', 'MT', 'PA', 'PB', 'PE', 'PI',  
 55: 'PR', 'RJ', 'RN', 'RO', 'RR', 'RS', 'SC', 'SE', 'SP', 'TO'];  
 56:  
 57:var lista\_municipios = {  
 58:   
 59: AC : [  
 60: // UF = AC  
 61: {id : '1309', geocod : '1200013', nome : "Acrelândia"},  
 62: {id : '1298', geocod : '1200054', nome : "Assis Brasil"},  
 63: {id : '1302', geocod : '1200104', nome : "Brasiléia"},  
 64: {id : '1293', geocod : '1200138', nome : "Bujari"},  
 65: {id : '1304', geocod : '1200179', nome : "Capixaba"},  
 66: {id : '359', geocod : '1200203', nome : "Cruzeiro do Sul"},  
 67: {id : '1301', geocod : '1200252', nome : "Epitaciolândia"},  
 68: {id : '1296', geocod : '1200302', nome : "Feijó"},  
 69: {id : '1297', geocod : '1200328', nome : "Jordão"},  
 70: {id : '357', geocod : '1200336', nome : "Mâncio Lima"},  
 71: {id : '1300', geocod : '1200344', nome : "Manoel Urbano"},  
 72: {id : '1291', geocod : '1200351', nome : "Marechal Thaumaturgo"},  
 73: {id : '1294', geocod : '1200385', nome : "Plácido de Castro"},  
 74: {id : '1307', geocod : '1200807', nome : "Porto Acre"},  
 75: {id : '1292', geocod : '1200393', nome : "Porto Walter"},  
 76: {id : '5477', geocod : '1200401', nome : "Rio Branco"},  
 77: {id : '358', geocod : '1200427', nome : "Rodrigues Alves"},  
 78: {id : '1299', geocod : '1200435', nome : "Santa Rosa do Purus"},  
 79: {id : '1305', geocod : '1200500', nome : "Sena Madureira"},  
 80: {id : '1306', geocod : '1200450', nome : "Senador Guiomard"},  
 81: {id : '1295', geocod : '1200609', nome : "Tarauacá"},  
 82: {id : '1303', geocod : '1200708', nome : "Xapuri"}  
 83: ],  
 84: AL : [  
 85: // UF = AL  
 86: {id : '1678', geocod : '2700102', nome : "Água Branca"},  
 87: {id : '1806', geocod : '2700201', nome : "Anadia"},  
 88: {id : '5691', geocod : '2709400', nome : "Anel"},  
 89: {id : '1796', geocod : '2700300', nome : "Arapiraca"},  
 90: {id : '1814', geocod : '2700409', nome : "Atalaia"},  
 91: {id : '1902', geocod : '2700508', nome : "Barra de Santo Antônio"},  
 92: {id : '1883', geocod : '2700607', nome : "Barra de São Miguel"},  
 93: {id : '5880', geocod : '2704500', nome : "Barra Grande"},  
 94: {id : '1757', geocod : '2700706', nome : "Batalha"},  
 95: {id : '1803', geocod : '2700805', nome : "Belém"},  
 96: {id : '1729', geocod : '2700904', nome : "Belo Monte"},  
 97: {id : '1810', geocod : '2701001', nome : "Boca da Mata"},  
 98: {id : '1559', geocod : '2701100', nome : "Branquinha"},  
 99: {id : '1776', geocod : '2701209', nome : "Cacimbinhas"},  
 100: {id : '1557', geocod : '2701308', nome : "Cajueiro"},  
 101: {id : '6210', geocod : '2706307', nome : "Caldeirões de Cima"},  
 102: {id : '1912', geocod : '2701357', nome : "Campestre"},  
 103: {id : '1805', geocod : '2701407', nome : "Campo Alegre"},  
 104: {id : '1762', geocod : '2701506', nome : "Campo Grande"},  
 105: {id : '6272', geocod : '2706307', nome : "Canafístula"},  
 106: {id : '1767', geocod : '2701605', nome : "Canapi"},  
 107: {id : '1558', geocod : '2701704', nome : "Capela"},  
 108: {id : '1732', geocod : '2701803', nome : "Carneiros"},  
 109: {id : '1825', geocod : '2701902', nome : "Chã Preta"},  
 110: {id : '1800', geocod : '2702009', nome : "Coité do Nóia"},  
 111: {id : '1875', geocod : '2702108', nome : "Colônia Leopoldina"},  
 112: {id : '1885', geocod : '2702207', nome : "Coqueiro Seco"},  
 113: {id : '1812', geocod : '2702306', nome : "Coruripe"},  
 114: {id : '1764', geocod : '2702355', nome : "Craíbas"},  
 115: {id : '1676', geocod : '2702405', nome : "Delmiro Gouveia"},  
 116: {id : '1775', geocod : '2702504', nome : "Dois Riachos"},  
 117: {id : '6742', geocod : '2707107', nome : "Entremontes"},  
 118: {id : '1779', geocod : '2702553', nome : "Estrela de Alagoas"},  
 119: {id : '1795', geocod : '2702603', nome : "Feira Grande"},  
 120: {id : '1794', geocod : '2702702', nome : "Feliz Deserto"},  
 121: {id : '6818', geocod : '2704302', nome : "Fernão Velho"},  
 122: {id : '1873', geocod : '2702801', nome : "Flexeiras"},  
 123: {id : '6843', geocod : '2704302', nome : "Floriano Peixoto"},  
 124: {id : '1761', geocod : '2702900', nome : "Girau do Ponciano"},  
 125: {id : '1871', geocod : '2703007', nome : "Ibateguara"},  
 126: {id : '1801', geocod : '2703106', nome : "Igaci"},  
 127: {id : '1792', geocod : '2703205', nome : "Igreja Nova"},  
 128: {id : '1679', geocod : '2703304', nome : "Inhapi"},  
 129: {id : '1756', geocod : '2703403', nome : "Jacaré dos Homens"},  
 130: {id : '1915', geocod : '2703502', nome : "Jacuípe"},  
 131: {id : '1909', geocod : '2703601', nome : "Japaratinga"},  
 132: {id : '1759', geocod : '2703700', nome : "Jaramataia"},  
 133: {id : '5558', geocod : '2703759', nome : "Jequiá da Praia"},  
 134: {id : '1874', geocod : '2703809', nome : "Joaquim Gomes"},  
 135: {id : '1905', geocod : '2703908', nome : "Jundiá"},  
 136: {id : '1798', geocod : '2704005', nome : "Junqueiro"},  
 137: {id : '1763', geocod : '2704104', nome : "Lagoa da Canoa"},  
 138: {id : '7456', geocod : '2706406', nome : "Limoeiro"},  
 139: {id : '1799', geocod : '2704203', nome : "Limoeiro de Anadia"},  
 140: {id : '5470', geocod : '2704302', nome : "Maceió"},  
 141: {id : '1760', geocod : '2704401', nome : "Major Isidoro"},  
 142: {id : '1808', geocod : '2704906', nome : "Mar Vermelho"},  
 143: {id : '1910', geocod : '2704500', nome : "Maragogi"},  
 144: {id : '1770', geocod : '2704609', nome : "Maravilha"},  
 145: {id : '1884', geocod : '2704708', nome : "Marechal Deodoro"},  
 146: {id : '1809', geocod : '2704807', nome : "Maribondo"},  
 147: {id : '1765', geocod : '2705002', nome : "Mata Grande"},  
 148: {id : '1904', geocod : '2705101', nome : "Matriz de Camaragibe"},  
 149: {id : '1872', geocod : '2705200', nome : "Messias"},  
 150: {id : '1778', geocod : '2705309', nome : "Minador do Negrão"},  
 151: {id : '1734', geocod : '2705408', nome : "Monteirópolis"},  
 152: {id : '7747', geocod : '2708105', nome : "Munguba"},  
 153: {id : '1870', geocod : '2705507', nome : "Murici"},  
 154: {id : '1876', geocod : '2705606', nome : "Novo Lino"},  
 155: {id : '1735', geocod : '2705705', nome : "Olho d'Água das Flores"},  
 156: {id : '1667', geocod : '2705804', nome : "Olho d'Água do Casado"},  
 157: {id : '1755', geocod : '2705903', nome : "Olho d'Água Grande"},  
 158: {id : '1758', geocod : '2706000', nome : "Olivença"},  
 159: {id : '1771', geocod : '2706109', nome : "Ouro Branco"},  
 160: {id : '1733', geocod : '2706208', nome : "Palestina"},  
 161: {id : '1815', geocod : '2706307', nome : "Palmeira dos Índios"},  
 162: {id : '1727', geocod : '2706406', nome : "Pão de Açúcar"},  
 163: {id : '1677', geocod : '2706422', nome : "Pariconha"},  
 164: {id : '1900', geocod : '2706448', nome : "Paripueira"},  
 165: {id : '1903', geocod : '2706505', nome : "Passo de Camaragibe"},  
 166: {id : '1823', geocod : '2706604', nome : "Paulo Jacinto"},  
 167: {id : '1789', geocod : '2706703', nome : "Penedo"},  
 168: {id : '1793', geocod : '2706802', nome : "Piaçabuçu"},  
 169: {id : '1882', geocod : '2706901', nome : "Pilar"},  
 170: {id : '1811', geocod : '2707008', nome : "Pindoba"},  
 171: {id : '1666', geocod : '2707107', nome : "Piranhas"},  
 172: {id : '1768', geocod : '2707206', nome : "Poço das Trincheiras"},  
 173: {id : '1906', geocod : '2707305', nome : "Porto Calvo"},  
 174: {id : '1908', geocod : '2707404', nome : "Porto de Pedras"},  
 175: {id : '1753', geocod : '2707503', nome : "Porto Real do Colégio"},  
 176: {id : '8247', geocod : '2702306', nome : "Poxim"},  
 177: {id : '1816', geocod : '2707602', nome : "Quebrangulo"},  
 178: {id : '8370', geocod : '2704401', nome : "Riacho do Sertão"},  
 179: {id : '1888', geocod : '2707701', nome : "Rio Largo"},  
 180: {id : '8443', geocod : '2709301', nome : "Rocha Cavalcante"},  
 181: {id : '1881', geocod : '2707800', nome : "Roteiro"},  
 182: {id : '8515', geocod : '2701704', nome : "Santa Efigênia"},  
 183: {id : '1886', geocod : '2707909', nome : "Santa Luzia do Norte"},  
 184: {id : '1769', geocod : '2708006', nome : "Santana do Ipanema"},  
 185: {id : '1828', geocod : '2708105', nome : "Santana do Mundaú"},  
 186: {id : '1754', geocod : '2708204', nome : "São Brás"},  
 187: {id : '1560', geocod : '2708303', nome : "São José da Laje"},  
 188: {id : '1731', geocod : '2708402', nome : "São José da Tapera"},  
 189: {id : '1901', geocod : '2708501', nome : "São Luís do Quitunde"},  
 190: {id : '1813', geocod : '2708600', nome : "São Miguel dos Campos"},  
 191: {id : '1907', geocod : '2708709', nome : "São Miguel dos Milagres"},  
 192: {id : '1797', geocod : '2708808', nome : "São Sebastião"},  
 193: {id : '8689', geocod : '2700409', nome : "Sapucaia"},  
 194: {id : '1887', geocod : '2708907', nome : "Satuba"},  
 195: {id : '1730', geocod : '2708956', nome : "Senador Rui Palmeira"},  
 196: {id : '1807', geocod : '2709004', nome : "Tanque d'Arca"},  
 197: {id : '1802', geocod : '2709103', nome : "Taquarana"},  
 198: {id : '9174', geocod : '2707404', nome : "Tatuamunha"},  
 199: {id : '1804', geocod : '2709152', nome : "Teotônio Vilela"},  
 200: {id : '1744', geocod : '2709202', nome : "Traipu"},  
 201: {id : '1561', geocod : '2709301', nome : "União dos Palmares"},  
 202: {id : '1824', geocod : '2709400', nome : "Viçosa"}  
 203: ],  
 204: AM : [  
 205: // UF = AM  
 206: {id : '37', geocod : '1300029', nome : "Alvarães"},  
 207: {id : '5677', geocod : '1301902', nome : "Amatari"},  
 208: {id : '26', geocod : '1300060', nome : "Amaturá"},  
 209: {id : '44', geocod : '1300086', nome : "Anamã"},  
 210: {id : '42', geocod : '1300102', nome : "Anori"},  
 211: {id : '377', geocod : '1300144', nome : "Apuí"},  
 212: {id : '5774', geocod : '1300508', nome : "Ariaú"},  
 213: {id : '364', geocod : '1300201', nome : "Atalaia do Norte"},  
 214: {id : '5811', geocod : '1304401', nome : "Augusto Montenegro"},  
 215: {id : '54', geocod : '1300300', nome : "Autazes"},  
 216: {id : '5833', geocod : '1301308', nome : "Badajos"},  
 217: {id : '5851', geocod : '1303536', nome : "Balbina"},  
 218: {id : '40', geocod : '1300409', nome : "Barcelos"},  
 219: {id : '64', geocod : '1300508', nome : "Barreirinha"},  
 220: {id : '365', geocod : '1300607', nome : "Benjamin Constant"},  
 221: {id : '43', geocod : '1300631', nome : "Beruri"},  
 222: {id : '63', geocod : '1300680', nome : "Boa Vista do Ramos"},  
 223: {id : '1308', geocod : '1300706', nome : "Boca do Acre"},  
 224: {id : '378', geocod : '1300805', nome : "Borba"},  
 225: {id : '45', geocod : '1300839', nome : "Caapiranga"},  
 226: {id : '6225', geocod : '1300508', nome : "Cametá"},  
 227: {id : '371', geocod : '1300904', nome : "Canutama"},  
 228: {id : '369', geocod : '1301001', nome : "Carauari"},  
 229: {id : '47', geocod : '1301100', nome : "Careiro"},  
 230: {id : '52', geocod : '1301159', nome : "Careiro da Várzea"},  
 231: {id : '374', geocod : '1301209', nome : "Coari"},  
 232: {id : '41', geocod : '1301308', nome : "Codajás"},  
 233: {id : '6600', geocod : '1303809', nome : "Cucui"},  
 234: {id : '363', geocod : '1301407', nome : "Eirunepé"},  
 235: {id : '361', geocod : '1301506', nome : "Envira"},  
 236: {id : '6842', geocod : '1300706', nome : "Floriano Peixoto"},  
 237: {id : '33', geocod : '1301605', nome : "Fonte Boa"},  
 238: {id : '6860', geocod : '1300508', nome : "Freguesia do Andirá"},  
 239: {id : '360', geocod : '1301654', nome : "Guajará"},  
 240: {id : '372', geocod : '1301704', nome : "Humaitá"},  
 241: {id : '7243', geocod : '1303809', nome : "Içana"},  
 242: {id : '362', geocod : '1301803', nome : "Ipixuna"},  
 243: {id : '50', geocod : '1301852', nome : "Iranduba"},  
 244: {id : '57', geocod : '1301902', nome : "Itacoatiara"},  
 245: {id : '367', geocod : '1301951', nome : "Itamarati"},  
 246: {id : '60', geocod : '1302009', nome : "Itapiranga"},  
 247: {id : '34', geocod : '1302108', nome : "Japurá"},  
 248: {id : '31', geocod : '1302207', nome : "Juruá"},  
 249: {id : '32', geocod : '1302306', nome : "Jutaí"},  
 250: {id : '370', geocod : '1302405', nome : "Lábrea"},  
 251: {id : '7388', geocod : '1300680', nome : "Lago Preto"},  
 252: {id : '48', geocod : '1302504', nome : "Manacapuru"},  
 253: {id : '49', geocod : '1302553', nome : "Manaquiri"},  
 254: {id : '5475', geocod : '1302603', nome : "Manaus"},  
 255: {id : '375', geocod : '1302702', nome : "Manicoré"},  
 256: {id : '36', geocod : '1302801', nome : "Maraã"},  
 257: {id : '7592', geocod : '1300680', nome : "Massauari"},  
 258: {id : '56', geocod : '1302900', nome : "Maués"},  
 259: {id : '7660', geocod : '1303403', nome : "Mocambo"},  
 260: {id : '7754', geocod : '1300300', nome : "Murutinga"},  
 261: {id : '67', geocod : '1303007', nome : "Nhamundá"},  
 262: {id : '53', geocod : '1303106', nome : "Nova Olinda do Norte"},  
 263: {id : '46', geocod : '1303205', nome : "Novo Airão"},  
 264: {id : '376', geocod : '1303304', nome : "Novo Aripuanã"},  
 265: {id : '7891', geocod : '1302900', nome : "Osório da Fonseca"},  
 266: {id : '65', geocod : '1303403', nome : "Parintins"},  
 267: {id : '368', geocod : '1303502', nome : "Pauini"},  
 268: {id : '8052', geocod : '1300508', nome : "Pedras"},  
 269: {id : '51', geocod : '1303536', nome : "Presidente Figueiredo"},  
 270: {id : '8353', geocod : '1302900', nome : "Repartimento"},  
 271: {id : '55', geocod : '1303569', nome : "Rio Preto da Eva"},  
 272: {id : '39', geocod : '1303601', nome : "Santa Isabel do Rio Negro"},  
 273: {id : '8568', geocod : '1303908', nome : "Santa Rita"},  
 274: {id : '28', geocod : '1303700', nome : "Santo Antônio do Içá"},  
 275: {id : '8842', geocod : '1303809', nome : "São Felipe"},  
 276: {id : '30', geocod : '1303809', nome : "São Gabriel da Cachoeira"},  
 277: {id : '27', geocod : '1303908', nome : "São Paulo de Olivença"},  
 278: {id : '61', geocod : '1303957', nome : "São Sebastião do Uatumã"},  
 279: {id : '58', geocod : '1304005', nome : "Silves"},  
 280: {id : '366', geocod : '1304062', nome : "Tabatinga"},  
 281: {id : '373', geocod : '1304104', nome : "Tapauá"},  
 282: {id : '38', geocod : '1304203', nome : "Tefé"},  
 283: {id : '29', geocod : '1304237', nome : "Tonantins"},  
 284: {id : '35', geocod : '1304260', nome : "Uarini"},  
 285: {id : '62', geocod : '1304302', nome : "Urucará"},  
 286: {id : '59', geocod : '1304401', nome : "Urucurituba"}  
 287: ],  
 288: AP : [  
 289: // UF = AP  
 290: {id : '14', geocod : '1600105', nome : "Amapá"},  
 291: {id : '5838', geocod : '1600303', nome : "Bailique"},  
 292: {id : '23', geocod : '1600204', nome : "Calçoene"},  
 293: {id : '6375', geocod : '1600402', nome : "Carvão"},  
 294: {id : '6466', geocod : '1600501', nome : "Clevelândia do Norte"},  
 295: {id : '6606', geocod : '1600204', nome : "Cunani"},  
 296: {id : '20', geocod : '1600212', nome : "Cutias"},  
 297: {id : '6805', geocod : '1600303', nome : "Fazendinha"},  
 298: {id : '18', geocod : '1600238', nome : "Ferreira Gomes"},  
 299: {id : '7071', geocod : '1600600', nome : "Igarapé do Lago"},  
 300: {id : '7089', geocod : '1600600', nome : "Ilha de Santana"},  
 301: {id : '19', geocod : '1600253', nome : "Itaubal"},  
 302: {id : '90', geocod : '1600279', nome : "Laranjal do Jari"},  
 303: {id : '7472', geocod : '1600204', nome : "Lourenço"},  
 304: {id : '5476', geocod : '1600303', nome : "Macapá"},  
 305: {id : '86', geocod : '1600402', nome : "Mazagão"},  
 306: {id : '7610', geocod : '1600402', nome : "Mazagão Velho"},  
 307: {id : '22', geocod : '1600501', nome : "Oiapoque"},  
 308: {id : '15', geocod : '1600154', nome : "Pedra Branca do Amaparí"},  
 309: {id : '17', geocod : '1600535', nome : "Porto Grande"},  
 310: {id : '8239', geocod : '1600303', nome : "Portuário do Igarapé"},  
 311: {id : '13', geocod : '1600550', nome : "Pracuúba"},  
 312: {id : '87', geocod : '1600600', nome : "Santana"},  
 313: {id : '8878', geocod : '1600212', nome : "São Joaquim do Pacuí"},  
 314: {id : '16', geocod : '1600055', nome : "Serra do Navio"},  
 315: {id : '8803', geocod : '1600105', nome : "Sucuriju"},  
 316: {id : '21', geocod : '1600709', nome : "Tartarugalzinho"},  
 317: {id : '9407', geocod : '1600501', nome : "Vila Velha"},  
 318: {id : '81', geocod : '1600808', nome : "Vitória do Jari"}  
 319: ],  
 320: BA : [  
 321: // UF = BA  
 322: {id : '5566', geocod : '2917904', nome : "Abadia"},  
 323: {id : '2448', geocod : '2900108', nome : "Abaíra"},  
 324: {id : '1655', geocod : '2900207', nome : "Abaré"},  
 325: {id : '5575', geocod : '2918407', nome : "Abóbora"},  
 326: {id : '5571', geocod : '2905701', nome : "Abrantes"},  
 327: {id : '1626', geocod : '2900306', nome : "Acajutiba"},  
 328: {id : '5821', geocod : '2921005', nome : "Açu da Torre"},  
 329: {id : '5823', geocod : '2906105', nome : "Açudina"},  
 330: {id : '5577', geocod : '2928604', nome : "Acupe"},  
 331: {id : '1648', geocod : '2900355', nome : "Adustina"},  
 332: {id : '5582', geocod : '2929305', nome : "Afligidos"},  
 333: {id : '5585', geocod : '2919306', nome : "Afrânio Peixoto"},  
 334: {id : '9467', geocod : '2912301', nome : "Água Doce"},  
 335: {id : '1621', geocod : '2900405', nome : "Água Fria"},  
 336: {id : '9480', geocod : '2923704', nome : "Águas do Paulista"},  
 337: {id : '2319', geocod : '2900603', nome : "Aiquara"},  
 338: {id : '2443', geocod : '2900702', nome : "Alagoinhas"},  
 339: {id : '3065', geocod : '2900801', nome : "Alcobaça"},  
 340: {id : '5612', geocod : '2908705', nome : "Alegre"},  
 341: {id : '5621', geocod : '2912905', nome : "Algodão"},  
 342: {id : '5624', geocod : '2925907', nome : "Algodões"},  
 343: {id : '2344', geocod : '2900900', nome : "Almadina"},  
 344: {id : '5634', geocod : '2922102', nome : "Alto Bonito"},  
 345: {id : '5665', geocod : '2921005', nome : "Amado Bahia"},  
 346: {id : '5673', geocod : '2930204', nome : "Amaniú"},  
 347: {id : '2372', geocod : '2901007', nome : "Amargosa"},  
 348: {id : '2432', geocod : '2901106', nome : "Amélia Rodrigues"},  
 349: {id : '1544', geocod : '2901155', nome : "América Dourada"},  
 350: {id : '5685', geocod : '2930204', nome : "Américo Alves"},  
 351: {id : '2293', geocod : '2901205', nome : "Anagé"},  
 352: {id : '2306', geocod : '2901304', nome : "Andaraí"},  
 353: {id : '1553', geocod : '2901353', nome : "Andorinha"},  
 354: {id : '2233', geocod : '2901403', nome : "Angical"},  
 355: {id : '2426', geocod : '2901502', nome : "Anguera"},  
 356: {id : '1658', geocod : '2901601', nome : "Antas"},  
 357: {id : '2424', geocod : '2901700', nome : "Antônio Cardoso"},  
 358: {id : '1583', geocod : '2901809', nome : "Antônio Gonçalves"},  
 359: {id : '1625', geocod : '2901908', nome : "Aporá"},  
 360: {id : '2246', geocod : '2901957', nome : "Apuarema"},  
 361: {id : '2446', geocod : '2902054', nome : "Araças"},  
 362: {id : '2289', geocod : '2902005', nome : "Aracatu"},  
 363: {id : '1618', geocod : '2902104', nome : "Araci"},  
 364: {id : '2442', geocod : '2902203', nome : "Aramari"},  
 365: {id : '5744', geocod : '2926707', nome : "Arapiranga"},  
 366: {id : '2339', geocod : '2902252', nome : "Arataca"},  
 367: {id : '2391', geocod : '2902302', nome : "Aratuípe"},  
 368: {id : '5771', geocod : '2925956', nome : "Argoim"},  
 369: {id : '5773', geocod : '2923001', nome : "Argolo"},  
 370: {id : '5775', geocod : '2910701', nome : "Aribice"},  
 371: {id : '5780', geocod : '2913606', nome : "Aritaguá"},  
 372: {id : '2241', geocod : '2902401', nome : "Aurelino Leal"},  
 373: {id : '2234', geocod : '2902500', nome : "Baianópolis"},  
 374: {id : '5843', geocod : '2929602', nome : "Baixa do Palmeira"},  
 375: {id : '1565', geocod : '2902609', nome : "Baixa Grande"},  
 376: {id : '5848', geocod : '2918001', nome : "Baixão"},  
 377: {id : '5844', geocod : '2932101', nome : "Baixinha"},  
 378: {id : '5855', geocod : '2905909', nome : "Baluarte"},  
 379: {id : '5857', geocod : '2913606', nome : "Banco Central"},  
 380: {id : '5859', geocod : '2915502', nome : "Bandeira do Almada"},  
 381: {id : '5860', geocod : '2916401', nome : "Bandeira do Colnia"},  
 382: {id : '1636', geocod : '2902658', nome : "Banzaê"},  
 383: {id : '5867', geocod : '2929909', nome : "Baraúnas"},  
 384: {id : '5870', geocod : '2905800', nome : "Barcelos do Sul"},  
 385: {id : '1451', geocod : '2902708', nome : "Barra"},  
 386: {id : '2302', geocod : '2902807', nome : "Barra da Estiva"},  
 387: {id : '2276', geocod : '2902906', nome : "Barra do Choça"},  
 388: {id : '1529', geocod : '2903003', nome : "Barra do Mendes"},  
 389: {id : '2355', geocod : '2903102', nome : "Barra do Rocha"},  
 390: {id : '2231', geocod : '2903201', nome : "Barreiras"},  
 391: {id : '1534', geocod : '2903235', nome : "Barro Alto"},  
 392: {id : '5929', geocod : '2909901', nome : "Barro Vermelho"},  
 393: {id : '5552', geocod : '2903276', nome : "Barrocas"},  
 394: {id : '5942', geocod : '2904001', nome : "Bastião"},  
 395: {id : '5947', geocod : '2933307', nome : "Bate Pé"},  
 396: {id : '5950', geocod : '2916005', nome : "Batinga"},  
 397: {id : '5964', geocod : '2906824', nome : "Bedengó"},  
 398: {id : '5967', geocod : '2907509', nome : "Bela Flor"},  
 399: {id : '5989', geocod : '2904902', nome : "Belém da Cachoeira"},  
 400: {id : '2335', geocod : '2903409', nome : "Belmonte"},  
 401: {id : '2266', geocod : '2903508', nome : "Belo Campo"},  
 402: {id : '5984', geocod : '2901155', nome : "Belo Campo - América Dourada"},  
 403: {id : '5990', geocod : '2907202', nome : "Bem bom"},  
 404: {id : '5995', geocod : '2908507', nome : "Bento Simões"},  
 405: {id : '1607', geocod : '2903607', nome : "Biritinga"},  
 406: {id : '6010', geocod : '2928307', nome : "Boa Espera"},  
 407: {id : '2314', geocod : '2903706', nome : "Boa Nova"},  
 408: {id : '6020', geocod : '2900702', nome : "Boa União"},  
 409: {id : '6032', geocod : '2913200', nome : "Boa Vista do Lagamar"},  
 410: {id : '2324', geocod : '2903805', nome : "Boa Vista do Tupim"},  
 411: {id : '6039', geocod : '2918001', nome : "Boaçu"},  
 412: {id : '6040', geocod : '2903409', nome : "Boca do Córrego"},  
 413: {id : '2206', geocod : '2903904', nome : "Bom Jesus da Lapa"},  
 414: {id : '2312', geocod : '2903953', nome : "Bom Jesus da Serra"},  
 415: {id : '6068', geocod : '2923209', nome : "Bom Sossego"},  
 416: {id : '6078', geocod : '2910800', nome : "Bonfim da Feira"},  
 417: {id : '2301', geocod : '2904001', nome : "Boninal"},  
 418: {id : '1538', geocod : '2904050', nome : "Bonito"},  
 419: {id : '2221', geocod : '2904100', nome : "Boquira"},  
 420: {id : '2182', geocod : '2904209', nome : "Botuporã"},  
 421: {id : '6099', geocod : '2926400', nome : "Botuquara"},  
 422: {id : '6111', geocod : '2905206', nome : "Brejinho das Ametistas"},  
 423: {id : '6116', geocod : '2924405', nome : "Brejo da Serra"},  
 424: {id : '6114', geocod : '2923035', nome : "Brejo Luiza de Brito"},  
 425: {id : '6115', geocod : '2903805', nome : "Brejo Novo"},  
 426: {id : '2371', geocod : '2904308', nome : "Brejões"},  
 427: {id : '2140', geocod : '2904407', nome : "Brejolândia"},  
 428: {id : '1452', geocod : '2904506', nome : "Brotas de Macaúbas"},  
 429: {id : '2290', geocod : '2904605', nome : "Brumado"},  
 430: {id : '6126', geocod : '2904100', nome : "Bucuituba"},  
 431: {id : '2351', geocod : '2904704', nome : "Buerarema"},  
 432: {id : '6132', geocod : '2900702', nome : "Buracica"},  
 433: {id : '6133', geocod : '2911808', nome : "Buranhém"},  
 434: {id : '6135', geocod : '2909604', nome : "Buril"},  
 435: {id : '1450', geocod : '2904753', nome : "Buritirama"},  
 436: {id : '2275', geocod : '2904803', nome : "Caatiba"},  
 437: {id : '2423', geocod : '2904852', nome : "Cabaceiras do Paraguaçu"},  
 438: {id : '6148', geocod : '2933307', nome : "Cabeceira da Jibóia"},  
 439: {id : '6151', geocod : '2924306', nome : "Cabrália"},  
 440: {id : '6154', geocod : '2933208', nome : "Cacha Pregos"},  
 441: {id : '2396', geocod : '2904902', nome : "Cachoeira"},  
 442: {id : '6170', geocod : '2931350', nome : "Cachoeira do Mato"},  
 443: {id : '2216', geocod : '2905008', nome : "Caculé"},  
 444: {id : '1571', geocod : '2905107', nome : "Caém"},  
 445: {id : '2294', geocod : '2905156', nome : "Caetanos"},  
 446: {id : '6181', geocod : '2923506', nome : "Caeté Açu"},  
 447: {id : '2214', geocod : '2905206', nome : "Caetité"},  
 448: {id : '1537', geocod : '2905305', nome : "Cafarnaum"},  
 449: {id : '6188', geocod : '2910701', nome : "Caimbe"},  
 450: {id : '2382', geocod : '2905404', nome : "Cairu"},  
 451: {id : '6195', geocod : '2916302', nome : "Caiubi"},  
 452: {id : '6202', geocod : '2930204', nome : "Cajuí"},  
 453: {id : '6208', geocod : '2932002', nome : "Caldeirão"},  
 454: {id : '1577', geocod : '2905503', nome : "Caldeirão Grande"},  
 455: {id : '6206', geocod : '2905206', nome : "Caldeiras"},  
 456: {id : '2337', geocod : '2905602', nome : "Camacan"},  
 457: {id : '2407', geocod : '2905701', nome : "Camaçari"},  
 458: {id : '2359', geocod : '2905800', nome : "Camamu"},  
 459: {id : '6219', geocod : '2917805', nome : "Camassandi"},  
 460: {id : '6227', geocod : '2929255', nome : "Camirim"},  
 461: {id : '6242', geocod : '2928604', nome : "Campinhos"},  
 462: {id : '1503', geocod : '2905909', nome : "Campo Alegre de Lourdes"},  
 463: {id : '1582', geocod : '2906006', nome : "Campo Formoso"},  
 464: {id : '6249', geocod : '2925600', nome : "Campo Formoso - Presidente D"},  
 465: {id : '6263', geocod : '2931202', nome : "Camurugi"},  
 466: {id : '6270', geocod : '2923605', nome : "Canabravinha"},  
 467: {id : '2175', geocod : '2906105', nome : "Canápolis"},  
 468: {id : '1535', geocod : '2906204', nome : "Canarana"},  
 469: {id : '6277', geocod : '2919801', nome : "Canatiba"},  
 470: {id : '2350', geocod : '2906303', nome : "Canavieiras"},  
 471: {id : '6279', geocod : '2918100', nome : "Canché"},  
 472: {id : '1601', geocod : '2906402', nome : "Candeal"},  
 473: {id : '2404', geocod : '2906501', nome : "Candeias"},  
 474: {id : '2210', geocod : '2906600', nome : "Candiba"},  
 475: {id : '2262', geocod : '2906709', nome : "Cândido Sales"},  
 476: {id : '6296', geocod : '2913101', nome : "Canoão"},  
 477: {id : '1616', geocod : '2906808', nome : "Cansanção"},  
 478: {id : '1650', geocod : '2906824', nome : "Canudos"},  
 479: {id : '6297', geocod : '2922052', nome : "Canudos - Mulungu do Morro"},  
 480: {id : '6314', geocod : '2918803', nome : "Capão"},  
 481: {id : '1597', geocod : '2906857', nome : "Capela do Alto Alegre"},  
 482: {id : '1578', geocod : '2906873', nome : "Capim Grosso"},  
 483: {id : '6328', geocod : '2918605', nome : "Caraguataí"},  
 484: {id : '2267', geocod : '2906899', nome : "Caraíbas"},  
 485: {id : '6330', geocod : '2908804', nome : "Caraibuna"},  
 486: {id : '6331', geocod : '2929107', nome : "Caraipé"},  
 487: {id : '6332', geocod : '2925303', nome : "Caraiva"},  
 488: {id : '3064', geocod : '2906907', nome : "Caravelas"},  
 489: {id : '1627', geocod : '2907004', nome : "Cardeal da Silva"},  
 490: {id : '2200', geocod : '2907103', nome : "Carinhanha"},  
 491: {id : '6354', geocod : '2926202', nome : "Cariparé"},  
 492: {id : '6361', geocod : '2918407', nome : "Carnaíba do Sertão"},  
 493: {id : '6367', geocod : '2930105', nome : "Carrapichel"},  
 494: {id : '1588', geocod : '2907202', nome : "Casa Nova"},  
 495: {id : '6383', geocod : '2913606', nome : "Castelo Novo"},  
 496: {id : '2378', geocod : '2907301', nome : "Castro Alves"},  
 497: {id : '6392', geocod : '2917508', nome : "Catinga do Moura"},  
 498: {id : '6393', geocod : '2920403', nome : "Catingal"},  
 499: {id : '2232', geocod : '2907400', nome : "Catolândia"},  
 500: {id : '6397', geocod : '2900108', nome : "Catolés"},  
 501: {id : '6394', geocod : '2915809', nome : "Catolezinho"},  
 502: {id : '2440', geocod : '2907509', nome : "Catu"},  
 503: {id : '2225', geocod : '2907558', nome : "Caturama"},  
 504: {id : '6409', geocod : '2913804', nome : "Cavunge"},  
 505: {id : '1459', geocod : '2907608', nome : "Central"},  
 506: {id : '6425', geocod : '2911709', nome : "Ceraíma"},  
 507: {id : '6426', geocod : '2933307', nome : "Cercadinho"},  
 508: {id : '1653', geocod : '2907707', nome : "Chorrochó"},  
 509: {id : '1638', geocod : '2907806', nome : "Cícero Dantas"},  
 510: {id : '6451', geocod : '2929503', nome : "Cinco Rios"},  
 511: {id : '1632', geocod : '2907905', nome : "Cipó"},  
 512: {id : '2356', geocod : '2908002', nome : "Coaraci"},  
 513: {id : '2167', geocod : '2908101', nome : "Cocos"},  
 514: {id : '6495', geocod : '2908309', nome : "Comércio"},  
 515: {id : '2427', geocod : '2908200', nome : "Conceição da Feira"},  
 516: {id : '2387', geocod : '2908309', nome : "Conceição do Almeida"},  
 517: {id : '1602', geocod : '2908408', nome : "Conceição do Coité"},  
 518: {id : '2433', geocod : '2908507', nome : "Conceição do Jacuípe"},  
 519: {id : '1681', geocod : '2908606', nome : "Conde"},  
 520: {id : '2237', geocod : '2908705', nome : "Condeúba"},  
 521: {id : '2297', geocod : '2908804', nome : "Contendas do Sincorá"},  
 522: {id : '6529', geocod : '2933604', nome : "Copixaba"},  
 523: {id : '6531', geocod : '2904902', nome : "Coqueiros"},  
 524: {id : '6532', geocod : '2901205', nome : "Coquinhos"},  
 525: {id : '2435', geocod : '2908903', nome : "Coração de Maria"},  
 526: {id : '2258', geocod : '2909000', nome : "Cordeiros"},  
 527: {id : '2170', geocod : '2909109', nome : "Coribe"},  
 528: {id : '1662', geocod : '2909208', nome : "Coronel João Sá"},  
 529: {id : '2171', geocod : '2909307', nome : "Correntina"},  
 530: {id : '6549', geocod : '2901007', nome : "Corta Mão"},  
 531: {id : '2236', geocod : '2909406', nome : "Cotegipe"},  
 532: {id : '6560', geocod : '2913606', nome : "Coutos"},  
 533: {id : '2363', geocod : '2909505', nome : "Cravolândia"},  
 534: {id : '1639', geocod : '2909604', nome : "Crisópolis"},  
 535: {id : '6576', geocod : '2904605', nome : "Cristalândia"},  
 536: {id : '2235', geocod : '2909703', nome : "Cristópolis"},  
 537: {id : '6589', geocod : '2907301', nome : "Crussai"},  
 538: {id : '2390', geocod : '2909802', nome : "Cruz das Almas"},  
 539: {id : '6605', geocod : '2925501', nome : "Cumuruxatiba"},  
 540: {id : '6609', geocod : '2917805', nome : "Cunhangi"},  
 541: {id : '1550', geocod : '2909901', nome : "Curaçá"},  
 542: {id : '6645', geocod : '2933307', nome : "Dantilândia"},  
 543: {id : '2315', geocod : '2910008', nome : "Dário Meira"},  
 544: {id : '6649', geocod : '2932457', nome : "Delfino"},  
 545: {id : '6655', geocod : '2909109', nome : "Descoberto"},  
 546: {id : '6663', geocod : '2921708', nome : "Dias Coelho"},  
 547: {id : '2449', geocod : '2910057', nome : "Dias d'Ávila"},  
 548: {id : '6669', geocod : '2901007', nome : "Diógenes Sampaio"},  
 549: {id : '2292', geocod : '2910107', nome : "Dom Basílio"},  
 550: {id : '2385', geocod : '2910206', nome : "Dom Macedo Costa"},  
 551: {id : '6707', geocod : '2921708', nome : "Duas Barras do Morro"},  
 552: {id : '2375', geocod : '2910305', nome : "Elísio Medrado"},  
 553: {id : '2264', geocod : '2910404', nome : "Encruzilhada"},  
 554: {id : '6724', geocod : '2932101', nome : "Engenheiro Franca"},  
 555: {id : '6728', geocod : '2918803', nome : "Engenheiro Pontes"},  
 556: {id : '1624', geocod : '2910503', nome : "Entre Rios"},  
 557: {id : '2227', geocod : '2900504', nome : "Érico Cardoso"},  
 558: {id : '1628', geocod : '2910602', nome : "Esplanada"},  
 559: {id : '1620', geocod : '2910701', nome : "Euclides da Cunha"},  
 560: {id : '3087', geocod : '2910727', nome : "Eunápolis"},  
 561: {id : '1647', geocod : '2910750', nome : "Fátima"},  
 562: {id : '2168', geocod : '2910776', nome : "Feira da Mata"},  
 563: {id : '2429', geocod : '2910800', nome : "Feira de Santana"},  
 564: {id : '6807', geocod : '2913101', nome : "Feira Nova"},  
 565: {id : '6819', geocod : '2914802', nome : "Ferradas"},  
 566: {id : '1584', geocod : '2910859', nome : "Filadélfia"},  
 567: {id : '6829', geocod : '2901957', nome : "Filanesia"},  
 568: {id : '2283', geocod : '2910909', nome : "Firmino Alves"},  
 569: {id : '2342', geocod : '2911006', nome : "Floresta Azul"},  
 570: {id : '1481', geocod : '2911105', nome : "Formosa do Rio Preto"},  
 571: {id : '6858', geocod : '2924801', nome : "França"},  
 572: {id : '6883', geocod : '2905404', nome : "Galeão"},  
 573: {id : '6885', geocod : '2905404', nome : "Gamboa"},  
 574: {id : '6888', geocod : '2930758', nome : "Gameleira da Lapa"},  
 575: {id : '6890', geocod : '2911303', nome : "Gameleira do Assuruá"},  
 576: {id : '2249', geocod : '2911204', nome : "Gandu"},  
 577: {id : '1608', geocod : '2911253', nome : "Gavião"},  
 578: {id : '1455', geocod : '2911303', nome : "Gentio do Ouro"},  
 579: {id : '6904', geocod : '2904852', nome : "Geolândia"},  
 580: {id : '1673', geocod : '2911402', nome : "Glória"},  
 581: {id : '2239', geocod : '2911501', nome : "Gongogi"},  
 582: {id : '6926', geocod : '2910800', nome : "Governador Doutor João Durva"},  
 583: {id : '5557', geocod : '2903300', nome : "Governador Lomanto Júnior"},  
 584: {id : '2393', geocod : '2911600', nome : "Governador Mangabeira"},  
 585: {id : '6988', geocod : '2920601', nome : "Guaí"},  
 586: {id : '6944', geocod : '2932903', nome : "Guaibim"},  
 587: {id : '2286', geocod : '2911659', nome : "Guajeru"},  
 588: {id : '2211', geocod : '2911709', nome : "Guanambi"},  
 589: {id : '6957', geocod : '2920601', nome : "Guapira"},  
 590: {id : '3085', geocod : '2911808', nome : "Guaratinga"},  
 591: {id : '6989', geocod : '2932903', nome : "Guerém"},  
 592: {id : '6994', geocod : '2921906', nome : "Guiné"},  
 593: {id : '6995', geocod : '2924504', nome : "Guirapá"},  
 594: {id : '6997', geocod : '2925402', nome : "Gurupá Mirim"},  
 595: {id : '1637', geocod : '2911857', nome : "Heliópolis"},  
 596: {id : '7001', geocod : '2923001', nome : "Helvécia"},  
 597: {id : '7009', geocod : '2932408', nome : "Hidrolândia"},  
 598: {id : '7015', geocod : '2910800', nome : "Humildes"},  
 599: {id : '2331', geocod : '2911907', nome : "Iaçu"},  
 600: {id : '7020', geocod : '2910503', nome : "Ibatuí"},  
 601: {id : '7026', geocod : '2920700', nome : "Ibiaçu"},  
 602: {id : '7023', geocod : '2926905', nome : "Ibiajara"},  
 603: {id : '7025', geocod : '2922102', nome : "Ibiaporã"},  
 604: {id : '2217', geocod : '2912004', nome : "Ibiassucê"},  
 605: {id : '2343', geocod : '2912103', nome : "Ibicaraí"},  
 606: {id : '2303', geocod : '2912202', nome : "Ibicoara"},  
 607: {id : '2284', geocod : '2912301', nome : "Ibicuí"},  
 608: {id : '1454', geocod : '2912400', nome : "Ibipeba"},  
 609: {id : '7032', geocod : '2914109', nome : "Ibipetum"},  
 610: {id : '2223', geocod : '2912509', nome : "Ibipitanga"},  
 611: {id : '2309', geocod : '2912608', nome : "Ibiquera"},  
 612: {id : '7035', geocod : '2902708', nome : "Ibiraba"},  
 613: {id : '7036', geocod : '2916005', nome : "Ibirajá"},  
 614: {id : '2243', geocod : '2912707', nome : "Ibirapitanga"},  
 615: {id : '3054', geocod : '2912806', nome : "Ibirapuã"},  
 616: {id : '2245', geocod : '2912905', nome : "Ibirataia"},  
 617: {id : '2228', geocod : '2913002', nome : "Ibitiara"},  
 618: {id : '7041', geocod : '2924900', nome : "Ibitiguira"},  
 619: {id : '7043', geocod : '2926806', nome : "Ibitira"},  
 620: {id : '1530', geocod : '2913101', nome : "Ibititá"},  
 621: {id : '7051', geocod : '2911303', nome : "Ibitunane"},  
 622: {id : '7053', geocod : '2912301', nome : "Ibitupã"},  
 623: {id : '7058', geocod : '2900207', nome : "Ibó"},  
 624: {id : '2143', geocod : '2913200', nome : "Ibotirama"},  
 625: {id : '1603', geocod : '2913309', nome : "Ichu"},  
 626: {id : '7064', geocod : '2921708', nome : "Icó"},  
 627: {id : '2215', geocod : '2913408', nome : "Igaporã"},  
 628: {id : '7068', geocod : '2930105', nome : "Igara"},  
 629: {id : '7073', geocod : '2902708', nome : "Igarité"},  
 630: {id : '7075', geocod : '2901304', nome : "Igatu"},  
 631: {id : '2360', geocod : '2913457', nome : "Igrapiúna"},  
 632: {id : '7086', geocod : '2933307', nome : "Iguá"},  
 633: {id : '2285', geocod : '2913507', nome : "Iguaí"},  
 634: {id : '7079', geocod : '2913507', nome : "Iguaibi"},  
 635: {id : '7080', geocod : '2919504', nome : "Iguatemi"},  
 636: {id : '7084', geocod : '2933604', nome : "Iguira"},  
 637: {id : '7085', geocod : '2912400', nome : "Iguitu"},  
 638: {id : '2353', geocod : '2913606', nome : "Ilhéus"},  
 639: {id : '7098', geocod : '2922102', nome : "Indaí"},  
 640: {id : '7101', geocod : '2913606', nome : "Inema"},  
 641: {id : '1623', geocod : '2913705', nome : "Inhambupe"},  
 642: {id : '7106', geocod : '2901106', nome : "Inhatá"},  
 643: {id : '7108', geocod : '2928109', nome : "Inhaúmas"},  
 644: {id : '7109', geocod : '2933307', nome : "Inhobim"},  
 645: {id : '7117', geocod : '2924306', nome : "Inúbia"},  
 646: {id : '2425', geocod : '2913804', nome : "Ipecaetá"},  
 647: {id : '2244', geocod : '2913903', nome : "Ipiaú"},  
 648: {id : '2419', geocod : '2914000', nome : "Ipirá"},  
 649: {id : '7126', geocod : '2917607', nome : "Ipiúna"},  
 650: {id : '7134', geocod : '2923209', nome : "Ipuçaba"},  
 651: {id : '1453', geocod : '2914109', nome : "Ipupiara"},  
 652: {id : '2330', geocod : '2914208', nome : "Irajuba"},  
 653: {id : '2304', geocod : '2914307', nome : "Iramaia"},  
 654: {id : '7139', geocod : '2914406', nome : "Iraporanga"},  
 655: {id : '2410', geocod : '2914406', nome : "Iraquara"},  
 656: {id : '2444', geocod : '2914505', nome : "Irará"},  
 657: {id : '1541', geocod : '2914604', nome : "Irecê"},  
 658: {id : '7149', geocod : '2917409', nome : "Irundiara"},  
 659: {id : '3086', geocod : '2914653', nome : "Itabela"},  
 660: {id : '2417', geocod : '2914703', nome : "Itaberaba"},  
 661: {id : '2352', geocod : '2914802', nome : "Itabuna"},  
 662: {id : '2358', geocod : '2914901', nome : "Itacaré"},  
 663: {id : '7156', geocod : '2908903', nome : "Itacava"},  
 664: {id : '7157', geocod : '2901007', nome : "Itachama"},  
 665: {id : '2307', geocod : '2915007', nome : "Itaeté"},  
 666: {id : '2316', geocod : '2915106', nome : "Itagi"},  
 667: {id : '2354', geocod : '2915205', nome : "Itagibá"},  
 668: {id : '3088', geocod : '2915304', nome : "Itagimirim"},  
 669: {id : '1457', geocod : '2915353', nome : "Itaguaçu da Bahia"},  
 670: {id : '7178', geocod : '2910909', nome : "Itaiá"},  
 671: {id : '7166', geocod : '2913903', nome : "Itaibó"},  
 672: {id : '7176', geocod : '2917508', nome : "Itaitu"},  
 673: {id : '7181', geocod : '2922706', nome : "Itajaí"},  
 674: {id : '2340', geocod : '2915403', nome : "Itaju do Colônia"},  
 675: {id : '7182', geocod : '2911303', nome : "Itajubaquara"},  
 676: {id : '2357', geocod : '2915502', nome : "Itajuípe"},  
 677: {id : '7185', geocod : '2918001', nome : "Itajuru"},  
 678: {id : '3063', geocod : '2915601', nome : "Itamaraju"},  
 679: {id : '2247', geocod : '2915700', nome : "Itamari"},  
 680: {id : '2273', geocod : '2915809', nome : "Itambé"},  
 681: {id : '7190', geocod : '2901908', nome : "Itamira"},  
 682: {id : '7193', geocod : '2918407', nome : "Itamotinga"},  
 683: {id : '7195', geocod : '2919504', nome : "Itanagé"},  
 684: {id : '2447', geocod : '2915908', nome : "Itanagra"},  
 685: {id : '3056', geocod : '2916005', nome : "Itanhém"},  
 686: {id : '2399', geocod : '2916104', nome : "Itaparica"},  
 687: {id : '2347', geocod : '2916203', nome : "Itapé"},  
 688: {id : '2334', geocod : '2916302', nome : "Itapebi"},  
 689: {id : '7205', geocod : '2917508', nome : "Itapeipu"},  
 690: {id : '2278', geocod : '2916401', nome : "Itapetinga"},  
 691: {id : '1640', geocod : '2916500', nome : "Itapicuru"},  
 692: {id : '2238', geocod : '2916609', nome : "Itapitanga"},  
 693: {id : '7214', geocod : '2922300', nome : "Itaporã"},  
 694: {id : '7216', geocod : '2921203', nome : "Itapura"},  
 695: {id : '2329', geocod : '2916708', nome : "Itaquara"},  
 696: {id : '7218', geocod : '2904605', nome : "Itaquaraí"},  
 697: {id : '2279', geocod : '2916807', nome : "Itarantim"},  
 698: {id : '7222', geocod : '2917102', nome : "Itati"},  
 699: {id : '2373', geocod : '2916856', nome : "Itatim"},  
 700: {id : '7223', geocod : '2902252', nome : "Itatingui"},  
 701: {id : '2325', geocod : '2916906', nome : "Itiruçu"},  
 702: {id : '1614', geocod : '2917003', nome : "Itiúba"},  
 703: {id : '2282', geocod : '2917102', nome : "Itororó"},  
 704: {id : '2296', geocod : '2917201', nome : "Ituaçu"},  
 705: {id : '2361', geocod : '2917300', nome : "Ituberá"},  
 706: {id : '7238', geocod : '2921104', nome : "Itupeva"},  
 707: {id : '2201', geocod : '2917334', nome : "Iuiú"},  
 708: {id : '2172', geocod : '2917359', nome : "Jaborandi"},  
 709: {id : '2198', geocod : '2917409', nome : "Jacaraci"},  
 710: {id : '1570', geocod : '2917508', nome : "Jacobina"},  
 711: {id : '7263', geocod : '2931707', nome : "Jacu"},  
 712: {id : '7271', geocod : '2929503', nome : "Jacuípe"},  
 713: {id : '7268', geocod : '2917805', nome : "Jacuruna"},  
 714: {id : '2328', geocod : '2917607', nome : "Jaguaquara"},  
 715: {id : '7273', geocod : '2910800', nome : "Jaguara"},  
 716: {id : '1593', geocod : '2917706', nome : "Jaguarari"},  
 717: {id : '2383', geocod : '2917805', nome : "Jaguaripe"},  
 718: {id : '7308', geocod : '2910800', nome : "Jaíba"},  
 719: {id : '1629', geocod : '2917904', nome : "Jandaíra"},  
 720: {id : '7290', geocod : '2915205', nome : "Japomirim"},  
 721: {id : '7291', geocod : '2913606', nome : "Japu"},  
 722: {id : '2321', geocod : '2918001', nome : "Jequié"},  
 723: {id : '1660', geocod : '2918100', nome : "Jeremoabo"},  
 724: {id : '2368', geocod : '2918209', nome : "Jiquiriçá"},  
 725: {id : '7314', geocod : '2933208', nome : "Jiribatuba"},  
 726: {id : '2320', geocod : '2918308', nome : "Jitaúna"},  
 727: {id : '7335', geocod : '2911907', nome : "João Amaro"},  
 728: {id : '7338', geocod : '2921906', nome : "João Correia"},  
 729: {id : '1543', geocod : '2918357', nome : "João Dourado"},  
 730: {id : '7328', geocod : '2933307', nome : "José Gonçalves"},  
 731: {id : '7341', geocod : '2917706', nome : "Juacema"},  
 732: {id : '1594', geocod : '2918407', nome : "Juazeiro"},  
 733: {id : '3057', geocod : '2918456', nome : "Jucuruçu"},  
 734: {id : '7348', geocod : '2906907', nome : "Juerana"},  
 735: {id : '7351', geocod : '2917508', nome : "Junco - Jacobina"},  
 736: {id : '7352', geocod : '2918407', nome : "Junco - Juazeiro"},  
 737: {id : '7358', geocod : '2909406', nome : "Jupaguá"},  
 738: {id : '7360', geocod : '2920809', nome : "Juraci"},  
 739: {id : '7362', geocod : '2918407', nome : "Juremal"},  
 740: {id : '1532', geocod : '2918506', nome : "Jussara"},  
 741: {id : '2345', geocod : '2918555', nome : "Jussari"},  
 742: {id : '2300', geocod : '2918605', nome : "Jussiape"},  
 743: {id : '2318', geocod : '2918704', nome : "Lafaiete Coutinho"},  
 744: {id : '7395', geocod : '2919801', nome : "Lagoa Clara"},  
 745: {id : '7410', geocod : '2903235', nome : "Lagoa do Boi"},  
 746: {id : '7397', geocod : '2906709', nome : "Lagoa Grande - Cândido Sales"},  
 747: {id : '7398', geocod : '2912400', nome : "Lagoa Grande - Ibipeba"},  
 748: {id : '7402', geocod : '2931806', nome : "Lagoa Preta"},  
 749: {id : '2219', geocod : '2918753', nome : "Lagoa Real"},  
 750: {id : '2370', geocod : '2918803', nome : "Laje"},  
 751: {id : '7422', geocod : '2902401', nome : "Laje do Banco"},  
 752: {id : '3053', geocod : '2918902', nome : "Lajedão"},  
 753: {id : '2414', geocod : '2919009', nome : "Lajedinho"},  
 754: {id : '7430', geocod : '2911907', nome : "Lajedo Alto"},  
 755: {id : '2326', geocod : '2919058', nome : "Lajedo do Tabocal"},  
 756: {id : '1606', geocod : '2919108', nome : "Lamarão"},  
 757: {id : '7433', geocod : '2929503', nome : "Lamarão do Passé"},  
 758: {id : '1540', geocod : '2919157', nome : "Lapão"},  
 759: {id : '7444', geocod : '2924801', nome : "Largo"},  
 760: {id : '2405', geocod : '2919207', nome : "Lauro de Freitas"},  
 761: {id : '2412', geocod : '2919306', nome : "Lençóis"},  
 762: {id : '2193', geocod : '2919405', nome : "Licínio de Almeida"},  
 763: {id : '2291', geocod : '2919504', nome : "Livramento de Nossa Senhora"},  
 764: {id : '7478', geocod : '2925006', nome : "Lucaia"},  
 765: {id : '5506', geocod : '2919553', nome : "Luís Eduardo Magalhães"},  
 766: {id : '7488', geocod : '2907202', nome : "Luís Viana"},  
 767: {id : '7487', geocod : '2931400', nome : "Lustosa"},  
 768: {id : '2418', geocod : '2919603', nome : "Macajuba"},  
 769: {id : '2272', geocod : '2919702', nome : "Macarani"},  
 770: {id : '2222', geocod : '2919801', nome : "Macaúbas"},  
 771: {id : '1654', geocod : '2919900', nome : "Macururé"},  
 772: {id : '2403', geocod : '2919926', nome : "Madre de Deus"},  
 773: {id : '2288', geocod : '2919959', nome : "Maetinga"},  
 774: {id : '2277', geocod : '2920007', nome : "Maiquinique"},  
 775: {id : '1566', geocod : '2920106', nome : "Mairi"},  
 776: {id : '2199', geocod : '2920205', nome : "Malhada"},  
 777: {id : '2287', geocod : '2920304', nome : "Malhada de Pedras"},  
 778: {id : '7517', geocod : '2930006', nome : "Mandiroba"},  
 779: {id : '7521', geocod : '2917904', nome : "Mangue Seco"},  
 780: {id : '7524', geocod : '2905206', nome : "Maniaçu"},  
 781: {id : '2317', geocod : '2920403', nome : "Manoel Vitorino"},  
 782: {id : '1486', geocod : '2920452', nome : "Mansidão"},  
 783: {id : '7537', geocod : '2933208', nome : "Mar Grande"},  
 784: {id : '2322', geocod : '2920502', nome : "Maracás"},  
 785: {id : '2397', geocod : '2920601', nome : "Maragogipe"},  
 786: {id : '7539', geocod : '2902302', nome : "Maragogipinho"},  
 787: {id : '2362', geocod : '2920700', nome : "Maraú"},  
 788: {id : '2323', geocod : '2920809', nome : "Marcionílio Souza"},  
 789: {id : '7556', geocod : '2926707', nome : "Marcolino Moura"},  
 790: {id : '7563', geocod : '2910800', nome : "Maria Quitéria"},  
 791: {id : '7566', geocod : '2932903', nome : "Maricoabo"},  
 792: {id : '7573', geocod : '2930907', nome : "Mariquita"},  
 793: {id : '2348', geocod : '2920908', nome : "Mascote"},  
 794: {id : '7590', geocod : '2910701', nome : "Massacará"},  
 795: {id : '7591', geocod : '2918407', nome : "Massaroca"},  
 796: {id : '7597', geocod : '2901106', nome : "Mata da Aliança"},  
 797: {id : '2439', geocod : '2921005', nome : "Mata de São João"},  
 798: {id : '7601', geocod : '2929206', nome : "Mataripe"},  
 799: {id : '2213', geocod : '2921054', nome : "Matina"},  
 800: {id : '3055', geocod : '2921104', nome : "Medeiros Neto"},  
 801: {id : '1569', geocod : '2921203', nome : "Miguel Calmon"},  
 802: {id : '2333', geocod : '2921302', nome : "Milagres"},  
 803: {id : '7632', geocod : '2903003', nome : "Minas do Espírito Santo"},  
 804: {id : '7633', geocod : '2930204', nome : "Minas do Mimoso"},  
 805: {id : '7648', geocod : '2925204', nome : "Miranga"},  
 806: {id : '1572', geocod : '2921401', nome : "Mirangaba"},  
 807: {id : '2310', geocod : '2921450', nome : "Mirante"},  
 808: {id : '7662', geocod : '2913002', nome : "Mocambo"},  
 809: {id : '7663', geocod : '2903409', nome : "Mogiquiçaba"},  
 810: {id : '7685', geocod : '2918001', nome : "Monte Branco"},  
 811: {id : '7691', geocod : '2910305', nome : "Monte Cruzeiro"},  
 812: {id : '7693', geocod : '2905701', nome : "Monte Gordo"},  
 813: {id : '7698', geocod : '2929206', nome : "Monte Recôncavo"},  
 814: {id : '1554', geocod : '2921500', nome : "Monte Santo"},  
 815: {id : '1449', geocod : '2921609', nome : "Morpará"},  
 816: {id : '7723', geocod : '2927200', nome : "Morro das Flores"},  
 817: {id : '1539', geocod : '2921708', nome : "Morro do Chapéu"},  
 818: {id : '2197', geocod : '2921807', nome : "Mortugaba"},  
 819: {id : '2305', geocod : '2921906', nome : "Mucugê"},  
 820: {id : '3060', geocod : '2922003', nome : "Mucuri"},  
 821: {id : '1536', geocod : '2922052', nome : "Mulungu do Morro"},  
 822: {id : '1563', geocod : '2922102', nome : "Mundo Novo"},  
 823: {id : '2386', geocod : '2922201', nome : "Muniz Ferreira"},  
 824: {id : '2142', geocod : '2922250', nome : "Muquém de S. Francisco"},  
 825: {id : '2394', geocod : '2922300', nome : "Muritiba"},  
 826: {id : '7758', geocod : '2911709', nome : "Mutãs"},  
 827: {id : '2369', geocod : '2922409', nome : "Mutuípe"},  
 828: {id : '7760', geocod : '2920601', nome : "Nagé"},  
 829: {id : '2392', geocod : '2922508', nome : "Nazaré"},  
 830: {id : '2379', geocod : '2922607', nome : "Nilo Peçanha"},  
 831: {id : '1615', geocod : '2922656', nome : "Nordestina"},  
 832: {id : '7813', geocod : '2926657', nome : "Nova Brasilia"},  
 833: {id : '2281', geocod : '2922706', nome : "Nova Canaã"},  
 834: {id : '1598', geocod : '2922730', nome : "Nova Fátima"},  
 835: {id : '2248', geocod : '2922755', nome : "Nova Ibiá"},  
 836: {id : '7829', geocod : '2924900', nome : "Nova Itaípe"},  
 837: {id : '2332', geocod : '2922805', nome : "Nova Itarana"},  
 838: {id : '7832', geocod : '2921104', nome : "Nova Lídice"},  
 839: {id : '2308', geocod : '2922854', nome : "Nova Redenção"},  
 840: {id : '1631', geocod : '2922904', nome : "Nova Soure"},  
 841: {id : '3061', geocod : '2923001', nome : "Nova Viçosa"},  
 842: {id : '7848', geocod : '2914307', nome : "Novo Acre"},  
 843: {id : '2229', geocod : '2923035', nome : "Novo Horizonte"},  
 844: {id : '1659', geocod : '2923050', nome : "Novo Triunfo"},  
 845: {id : '7864', geocod : '2921401', nome : "Nuguaçu"},  
 846: {id : '7874', geocod : '2913002', nome : "Olhos d'Água do Seco"},  
 847: {id : '1630', geocod : '2923100', nome : "Olindina"},  
 848: {id : '2144', geocod : '2923209', nome : "Oliveira dos Brejinhos"},  
 849: {id : '7880', geocod : '2913606', nome : "Olivença"},  
 850: {id : '7883', geocod : '2922201', nome : "Onha"},  
 851: {id : '7887', geocod : '2918001', nome : "Oriente Novo"},  
 852: {id : '7893', geocod : '2906303', nome : "Ouricana"},  
 853: {id : '2445', geocod : '2923308', nome : "Ouriçangas"},  
 854: {id : '7894', geocod : '2904506', nome : "Ouricuri do Ouro"},  
 855: {id : '1547', geocod : '2923357', nome : "Ourolândia"},  
 856: {id : '7903', geocod : '2929008', nome : "Outeiro Redondo"},  
 857: {id : '7927', geocod : '2917409', nome : "Paiol"},  
 858: {id : '7931', geocod : '2913408', nome : "Pajeú do Vento"},  
 859: {id : '7934', geocod : '2908606', nome : "Palame"},  
 860: {id : '2202', geocod : '2923407', nome : "Palmas de Monte Alto"},  
 861: {id : '2409', geocod : '2923506', nome : "Palmeiras"},  
 862: {id : '2224', geocod : '2923605', nome : "Paramirim"},  
 863: {id : '7977', geocod : '2920205', nome : "Parateca"},  
 864: {id : '2208', geocod : '2923704', nome : "Paratinga"},  
 865: {id : '1551', geocod : '2923803', nome : "Paripiranga"},  
 866: {id : '8026', geocod : '2903607', nome : "Pataíba"},  
 867: {id : '8025', geocod : '2909901', nome : "Patamuté"},  
 868: {id : '8034', geocod : '2907202', nome : "Pau a Pique"},  
 869: {id : '2336', geocod : '2923902', nome : "Pau Brasil"},  
 870: {id : '1672', geocod : '2924009', nome : "Paulo Afonso"},  
 871: {id : '8042', geocod : '2903235', nome : "Paz de Salobro"},  
 872: {id : '1596', geocod : '2924058', nome : "Pé de Serra"},  
 873: {id : '2437', geocod : '2924108', nome : "Pedrão"},  
 874: {id : '1663', geocod : '2924207', nome : "Pedro Alexandre"},  
 875: {id : '8068', geocod : '2905909', nome : "Peixe"},  
 876: {id : '8093', geocod : '2907301', nome : "Petim"},  
 877: {id : '8095', geocod : '2920700', nome : "Piabanha"},  
 878: {id : '2299', geocod : '2924306', nome : "Piatã"},  
 879: {id : '8175', geocod : '2930204', nome : "Piçarrão"},  
 880: {id : '1510', geocod : '2924405', nome : "Pilão Arcado"},  
 881: {id : '8106', geocod : '2906600', nome : "Pilões"},  
 882: {id : '8107', geocod : '2920908', nome : "Pimenta"},  
 883: {id : '8108', geocod : '2913606', nome : "Pimenteira"},  
 884: {id : '2212', geocod : '2924504', nome : "Pindaí"},  
 885: {id : '1581', geocod : '2924603', nome : "Pindobaçu"},  
 886: {id : '8136', geocod : '2918407', nome : "Pinhões"},  
 887: {id : '1568', geocod : '2924652', nome : "Pintadas"},  
 888: {id : '2251', geocod : '2924678', nome : "Piraí do Norte"},  
 889: {id : '8147', geocod : '2917805', nome : "Pirajuia"},  
 890: {id : '8160', geocod : '2907202', nome : "Piri"},  
 891: {id : '2259', geocod : '2924702', nome : "Piripá"},  
 892: {id : '1564', geocod : '2924801', nome : "Piritiba"},  
 893: {id : '8172', geocod : '2911303', nome : "Pituba"},  
 894: {id : '2327', geocod : '2924900', nome : "Planaltino"},  
 895: {id : '2311', geocod : '2925006', nome : "Planalto"},  
 896: {id : '8250', geocod : '2902401', nome : "Poço Central"},  
 897: {id : '8259', geocod : '2909901', nome : "Poço de Fora"},  
 898: {id : '2313', geocod : '2925105', nome : "Poções"},  
 899: {id : '8261', geocod : '2926004', nome : "Poços"},  
 900: {id : '2441', geocod : '2925204', nome : "Pojuca"},  
 901: {id : '8190', geocod : '2906907', nome : "Ponta da Areia"},  
 902: {id : '1580', geocod : '2925253', nome : "Ponto Novo"},  
 903: {id : '8222', geocod : '2928208', nome : "Porto Novo"},  
 904: {id : '3089', geocod : '2925303', nome : "Porto Seguro"},  
 905: {id : '2280', geocod : '2925402', nome : "Potiraguá"},  
 906: {id : '8248', geocod : '2906303', nome : "Poxim do Sul"},  
 907: {id : '3066', geocod : '2925501', nome : "Prado"},  
 908: {id : '8265', geocod : '2933307', nome : "Pradoso"},  
 909: {id : '1531', geocod : '2925600', nome : "Presidente Dutra"},  
 910: {id : '2260', geocod : '2925709', nome : "Presidente Jânio Quadros"},  
 911: {id : '2367', geocod : '2925758', nome : "Presidente Tancredo Neves"},  
 912: {id : '8283', geocod : '2901155', nome : "Prevenido"},  
 913: {id : '8307', geocod : '2906709', nome : "Quaraçu"},  
 914: {id : '1613', geocod : '2925808', nome : "Queimadas"},  
 915: {id : '1619', geocod : '2925907', nome : "Quijingue"},  
 916: {id : '8332', geocod : '2921609', nome : "Quixaba"},  
 917: {id : '1576', geocod : '2925931', nome : "Quixabeira"},  
 918: {id : '2420', geocod : '2925956', nome : "Rafael Jambeiro"},  
 919: {id : '8345', geocod : '2918506', nome : "Recife"},  
 920: {id : '1512', geocod : '2926004', nome : "Remanso"},  
 921: {id : '8350', geocod : '2923035', nome : "Remédios"},  
 922: {id : '1611', geocod : '2926103', nome : "Retirolândia"},  
 923: {id : '1483', geocod : '2926202', nome : "Riachão das Neves"},  
 924: {id : '1599', geocod : '2926301', nome : "Riachão do Jacuípe"},  
 925: {id : '8371', geocod : '2932804', nome : "Riachão do Utinga"},  
 926: {id : '8367', geocod : '2900702', nome : "Riacho da Guia"},  
 927: {id : '2220', geocod : '2926400', nome : "Riacho de Santana"},  
 928: {id : '8364', geocod : '2909901', nome : "Riacho Seco"},  
 929: {id : '1633', geocod : '2926509', nome : "Ribeira do Amparo"},  
 930: {id : '1635', geocod : '2926608', nome : "Ribeira do Pombal"},  
 931: {id : '2271', geocod : '2926657', nome : "Ribeirão do Largo"},  
 932: {id : '8385', geocod : '2916807', nome : "Ribeirão do Salto"},  
 933: {id : '8427', geocod : '2933174', nome : "Rio da Dona"},  
 934: {id : '2298', geocod : '2926707', nome : "Rio de Contas"},  
 935: {id : '2218', geocod : '2926806', nome : "Rio do Antônio"},  
 936: {id : '8433', geocod : '2913606', nome : "Rio do Braço"},  
 937: {id : '8434', geocod : '2917102', nome : "Rio do Meio"},  
 938: {id : '2226', geocod : '2926905', nome : "Rio do Pires"},  
 939: {id : '8410', geocod : '2931707', nome : "Rio Fundo"},  
 940: {id : '1641', geocod : '2927002', nome : "Rio Real"},  
 941: {id : '1669', geocod : '2927101', nome : "Rodelas"},  
 942: {id : '2416', geocod : '2927200', nome : "Ruy Barbosa"},  
 943: {id : '8473', geocod : '2924405', nome : "Saldanha"},  
 944: {id : '8475', geocod : '2908408', nome : "Salgadália"},  
 945: {id : '2398', geocod : '2927309', nome : "Salinas da Margarida"},  
 946: {id : '5471', geocod : '2927408', nome : "Salvador"},  
 947: {id : '8489', geocod : '2916500', nome : "Sambaíba"},  
 948: {id : '1604', geocod : '2927507', nome : "Santa Bárbara"},  
 949: {id : '1664', geocod : '2927606', nome : "Santa Brígida"},  
 950: {id : '3090', geocod : '2927705', nome : "Santa Cruz Cabrália"},  
 951: {id : '2341', geocod : '2927804', nome : "Santa Cruz da Vitória"},  
 952: {id : '2364', geocod : '2927903', nome : "Santa Inês"},  
 953: {id : '2338', geocod : '2928059', nome : "Santa Luzia"},  
 954: {id : '2173', geocod : '2928109', nome : "Santa Maria da Vitória"},  
 955: {id : '1485', geocod : '2928406', nome : "Santa Rita de Cássia"},  
 956: {id : '2377', geocod : '2928505', nome : "Santa Teresinha"},  
 957: {id : '1612', geocod : '2928000', nome : "Santaluz"},  
 958: {id : '2176', geocod : '2928208', nome : "Santana"},  
 959: {id : '2430', geocod : '2928307', nome : "Santanópolis"},  
 960: {id : '8620', geocod : '2904902', nome : "Santiago do Iguape"},  
 961: {id : '2431', geocod : '2928604', nome : "Santo Amaro"},  
 962: {id : '8647', geocod : '2906907', nome : "Santo Antônio de Barcelona"},  
 963: {id : '2384', geocod : '2928703', nome : "Santo Antônio de Jesus"},  
 964: {id : '2422', geocod : '2928802', nome : "Santo Estêvão"},  
 965: {id : '8678', geocod : '2911303', nome : "Santo Inácio"},  
 966: {id : '2230', geocod : '2928901', nome : "São Desidério"},  
 967: {id : '1609', geocod : '2928950', nome : "São Domingos"},  
 968: {id : '2389', geocod : '2929107', nome : "São Felipe"},  
 969: {id : '2395', geocod : '2929008', nome : "São Félix"},  
 970: {id : '2174', geocod : '2929057', nome : "São Félix do Coribe"},  
 971: {id : '2402', geocod : '2929206', nome : "São Francisco do Conde"},  
 972: {id : '1542', geocod : '2929255', nome : "São Gabriel"},  
 973: {id : '2428', geocod : '2929305', nome : "São Gonçalo dos Campos"},  
 974: {id : '8949', geocod : '2907806', nome : "São João da Fortaleza"},  
 975: {id : '8965', geocod : '2920908', nome : "São João do Paraíso"},  
 976: {id : '2346', geocod : '2929354', nome : "São José da Vitória"},  
 977: {id : '8916', geocod : '2916401', nome : "São José do Colônia"},  
 978: {id : '1579', geocod : '2929370', nome : "São José do Jacuípe"},  
 979: {id : '8931', geocod : '2926202', nome : "São José do Rio Grande"},  
 980: {id : '8989', geocod : '2909307', nome : "São Manoel do Norte"},  
 981: {id : '2374', geocod : '2929404', nome : "São Miguel das Matas"},  
 982: {id : '9053', geocod : '2920601', nome : "São Roque do Paraguaçu"},  
 983: {id : '9055', geocod : '2933307', nome : "São Sebastião"},  
 984: {id : '2438', geocod : '2929503', nome : "São Sebastião do Passé"},  
 985: {id : '9096', geocod : '2919504', nome : "São Timóteo"},  
 986: {id : '2388', geocod : '2929602', nome : "Sapeaçu"},  
 987: {id : '1622', geocod : '2929701', nome : "Sátiro Dias"},  
 988: {id : '2401', geocod : '2929750', nome : "Saubara"},  
 989: {id : '8703', geocod : '2904506', nome : "Saudável"},  
 990: {id : '1573', geocod : '2929800', nome : "Saúde"},  
 991: {id : '2408', geocod : '2929909', nome : "Seabra"},  
 992: {id : '2209', geocod : '2930006', nome : "Sebastião Laranjeiras"},  
 993: {id : '1592', geocod : '2930105', nome : "Senhor do Bonfim"},  
 994: {id : '1585', geocod : '2930204', nome : "Sento Sé"},  
 995: {id : '8726', geocod : '2929305', nome : "Sergi"},  
 996: {id : '8736', geocod : '2932002', nome : "Serra da Canabrava"},  
 997: {id : '2203', geocod : '2930154', nome : "Serra do Ramalho"},  
 998: {id : '2204', geocod : '2930303', nome : "Serra Dourada"},  
 999: {id : '8732', geocod : '2932903', nome : "Serra Grande"},  
1000: {id : '2421', geocod : '2930402', nome : "Serra Preta"},  
1001: {id : '1605', geocod : '2930501', nome : "Serrinha"},  
1002: {id : '1575', geocod : '2930600', nome : "Serrolândia"},  
1003: {id : '2406', geocod : '2930709', nome : "Simões Filho"},  
1004: {id : '9118', geocod : '2901353', nome : "Sítio da Baraúna"},  
1005: {id : '2207', geocod : '2930758', nome : "Sítio do Mato"},  
1006: {id : '9119', geocod : '2907301', nome : "Sítio do Meio"},  
1007: {id : '1661', geocod : '2930766', nome : "Sítio do Quinto"},  
1008: {id : '9115', geocod : '2928901', nome : "Sítio Grande"},  
1009: {id : '9116', geocod : '2907509', nome : "Sítio Novo"},  
1010: {id : '8777', geocod : '2901155', nome : "Soares"},  
1011: {id : '1591', geocod : '2930774', nome : "Sobradinho"},  
1012: {id : '8778', geocod : '2907202', nome : "Sobrado"},  
1013: {id : '2411', geocod : '2930808', nome : "Souto Soares"},  
1014: {id : '8799', geocod : '2910503', nome : "Subaúma"},  
1015: {id : '8810', geocod : '2931004', nome : "Sussuarana"},  
1016: {id : '2205', geocod : '2930907', nome : "Tabocas do Brejo Velho"},  
1017: {id : '9130', geocod : '2933174', nome : "Taboleiro do Castro"},  
1018: {id : '9131', geocod : '2914901', nome : "Taboquinhas"},  
1019: {id : '9134', geocod : '2909406', nome : "Taguá"},  
1020: {id : '9144', geocod : '2921708', nome : "Tamboril"},  
1021: {id : '2295', geocod : '2931004', nome : "Tanhaçu"},  
1022: {id : '2181', geocod : '2931053', nome : "Tanque Novo"},  
1023: {id : '1600', geocod : '2931103', nome : "Tanquinho"},  
1024: {id : '9149', geocod : '2901353', nome : "Tanquinho do Poço"},  
1025: {id : '2380', geocod : '2931202', nome : "Taperoá"},  
1026: {id : '9158', geocod : '2927200', nome : "Tapiraípe"},  
1027: {id : '9155', geocod : '2911501', nome : "Tapirama"},  
1028: {id : '1562', geocod : '2931301', nome : "Tapiramutá"},  
1029: {id : '9156', geocod : '2921203', nome : "Tapiranga"},  
1030: {id : '9159', geocod : '2905800', nome : "Tapuia"},  
1031: {id : '9164', geocod : '2921401', nome : "Taquarandi"},  
1032: {id : '9168', geocod : '2922003', nome : "Taquarinha"},  
1033: {id : '9172', geocod : '2921302', nome : "Tartaruga"},  
1034: {id : '9176', geocod : '2919405', nome : "Tauape"},  
1035: {id : '3062', geocod : '2931350', nome : "Teixeira de Freitas"},  
1036: {id : '9183', geocod : '2920908', nome : "Teixeira do Progresso"},  
1037: {id : '2436', geocod : '2931400', nome : "Teodoro Sampaio"},  
1038: {id : '1617', geocod : '2931509', nome : "Teofilândia"},  
1039: {id : '2366', geocod : '2931608', nome : "Teolândia"},  
1040: {id : '2434', geocod : '2931707', nome : "Terra Nova"},  
1041: {id : '9200', geocod : '2930105', nome : "Tijuaçu"},  
1042: {id : '9209', geocod : '2910800', nome : "Tiquaruçu"},  
1043: {id : '2265', geocod : '2931806', nome : "Tremedal"},  
1044: {id : '9239', geocod : '2902807', nome : "Triunfo do Sincorá"},  
1045: {id : '1634', geocod : '2931905', nome : "Tucano"},  
1046: {id : '1649', geocod : '2932002', nome : "Uauá"},  
1047: {id : '2365', geocod : '2932101', nome : "Ubaíra"},  
1048: {id : '2242', geocod : '2932200', nome : "Ubaitaba"},  
1049: {id : '2240', geocod : '2932309', nome : "Ubatã"},  
1050: {id : '9279', geocod : '2904605', nome : "Ubiraçaba"},  
1051: {id : '9278', geocod : '2901304', nome : "Ubiraitá"},  
1052: {id : '1458', geocod : '2932408', nome : "Uibaí"},  
1053: {id : '1546', geocod : '2932457', nome : "Umburanas"},  
1054: {id : '2349', geocod : '2932507', nome : "Una"},  
1055: {id : '2192', geocod : '2932606', nome : "Urandi"},  
1056: {id : '2252', geocod : '2932705', nome : "Uruçuca"},  
1057: {id : '2415', geocod : '2932804', nome : "Utinga"},  
1058: {id : '9316', geocod : '2925303', nome : "Vale Verde"},  
1059: {id : '2381', geocod : '2932903', nome : "Valença"},  
1060: {id : '1610', geocod : '2933000', nome : "Valente"},  
1061: {id : '1567', geocod : '2933059', nome : "Várzea da Roça"},  
1062: {id : '9444', geocod : '2929909', nome : "Várzea do Caldas"},  
1063: {id : '9445', geocod : '2922052', nome : "Várzea do Cerco"},  
1064: {id : '1574', geocod : '2933109', nome : "Várzea do Poço"},  
1065: {id : '1545', geocod : '2933158', nome : "Várzea Nova"},  
1066: {id : '9448', geocod : '2902500', nome : "Várzeas"},  
1067: {id : '2376', geocod : '2933174', nome : "Varzedo"},  
1068: {id : '9344', geocod : '2905404', nome : "Velha Boipeba"},  
1069: {id : '9352', geocod : '2921708', nome : "Ventura"},  
1070: {id : '2400', geocod : '2933208', nome : "Vera Cruz"},  
1071: {id : '3058', geocod : '2933257', nome : "Vereda"},  
1072: {id : '9356', geocod : '2933307', nome : "Veredinha"},  
1073: {id : '2274', geocod : '2933307', nome : "Vitória da Conquista"},  
1074: {id : '9435', geocod : '2931301', nome : "Volta Grande"},  
1075: {id : '2413', geocod : '2933406', nome : "Wagner"},  
1076: {id : '2141', geocod : '2933455', nome : "Wanderley"},  
1077: {id : '2250', geocod : '2933505', nome : "Wenceslau Guimarães"},  
1078: {id : '1456', geocod : '2933604', nome : "Xique-Xique"}  
1079: ],  
1080: CE : [  
1081: // UF = CE  
1082: {id : '782', geocod : '2300101', nome : "Abaiara"},  
1083: {id : '924', geocod : '2300150', nome : "Acarapé"},  
1084: {id : '333', geocod : '2300200', nome : "Acaraú"},  
1085: {id : '858', geocod : '2300309', nome : "Acopiara"},  
1086: {id : '5822', geocod : '2305332', nome : "Açude dos Pinheiros"},  
1087: {id : '5580', geocod : '2304707', nome : "Adrianópolis"},  
1088: {id : '9468', geocod : '2311504', nome : "Água Fria"},  
1089: {id : '9472', geocod : '2304954', nome : "Água Verde"},  
1090: {id : '5590', geocod : '2306306', nome : "Aguaí"},  
1091: {id : '5597', geocod : '2308005', nome : "Aiuá"},  
1092: {id : '720', geocod : '2300408', nome : "Aiuaba"},  
1093: {id : '5599', geocod : '2301307', nome : "Alagoinha"},  
1094: {id : '317', geocod : '2300507', nome : "Alcântaras"},  
1095: {id : '5615', geocod : '2302503', nome : "Aleixo"},  
1096: {id : '5622', geocod : '2311264', nome : "Algodões"},  
1097: {id : '5626', geocod : '2306553', nome : "Almofala"},  
1098: {id : '785', geocod : '2300606', nome : "Altaneira"},  
1099: {id : '5641', geocod : '2304004', nome : "Alto Lindo"},  
1100: {id : '893', geocod : '2300705', nome : "Alto Santo"},  
1101: {id : '5666', geocod : '2311702', nome : "Amanaiara"},  
1102: {id : '5667', geocod : '2307700', nome : "Amanari"},  
1103: {id : '5672', geocod : '2307502', nome : "Amaniutuba"},  
1104: {id : '5675', geocod : '2302602', nome : "Amarelas"},  
1105: {id : '5676', geocod : '2301604', nome : "Amaro"},  
1106: {id : '5684', geocod : '2305902', nome : "América"},  
1107: {id : '340', geocod : '2300754', nome : "Amontada"},  
1108: {id : '5687', geocod : '2308104', nome : "Anauá"},  
1109: {id : '5698', geocod : '2305233', nome : "Aningas"},  
1110: {id : '5699', geocod : '2312106', nome : "Anjinhos"},  
1111: {id : '5702', geocod : '2311355', nome : "Antonico"},  
1112: {id : '722', geocod : '2300804', nome : "Antonina do Norte"},  
1113: {id : '5705', geocod : '2304400', nome : "Antônio Bezerra"},  
1114: {id : '5707', geocod : '2311603', nome : "Antônio Diogo"},  
1115: {id : '5710', geocod : '2307700', nome : "Antônio Marques"},  
1116: {id : '338', geocod : '2300903', nome : "Apuiarés"},  
1117: {id : '5730', geocod : '2306900', nome : "Aquinópolis"},  
1118: {id : '356', geocod : '2301000', nome : "Aquiraz"},  
1119: {id : '933', geocod : '2301109', nome : "Aracati"},  
1120: {id : '5734', geocod : '2313609', nome : "Aracatiaçu"},  
1121: {id : '5733', geocod : '2300754', nome : "Aracatiara"},  
1122: {id : '879', geocod : '2301208', nome : "Aracoiaba"},  
1123: {id : '5739', geocod : '2301901', nome : "Arajara"},  
1124: {id : '5741', geocod : '2300200', nome : "Aranaú"},  
1125: {id : '5748', geocod : '2313401', nome : "Arapá"},  
1126: {id : '5743', geocod : '2306405', nome : "Arapari"},  
1127: {id : '5745', geocod : '2312106', nome : "Araporanga"},  
1128: {id : '5749', geocod : '2304004', nome : "Araquém"},  
1129: {id : '5752', geocod : '2302057', nome : "Araras"},  
1130: {id : '750', geocod : '2301257', nome : "Ararendá"},  
1131: {id : '716', geocod : '2301307', nome : "Araripe"},  
1132: {id : '5754', geocod : '2303105', nome : "Arariús"},  
1133: {id : '5755', geocod : '2301604', nome : "Aratama"},  
1134: {id : '5756', geocod : '2313609', nome : "Araticum"},  
1135: {id : '874', geocod : '2301406', nome : "Aratuba"},  
1136: {id : '5778', geocod : '2309458', nome : "Arisco dos Marianos"},  
1137: {id : '721', geocod : '2301505', nome : "Arneiroz"},  
1138: {id : '5784', geocod : '2304004', nome : "Aroeiras"},  
1139: {id : '5792', geocod : '2307502', nome : "Arrojado"},  
1140: {id : '5795', geocod : '2308708', nome : "Aruaru"},  
1141: {id : '719', geocod : '2301604', nome : "Assaré"},  
1142: {id : '5802', geocod : '2303808', nome : "Assunção - Cedro"},  
1143: {id : '5803', geocod : '2306405', nome : "Assunção - Itapipoca"},  
1144: {id : '5804', geocod : '2313005', nome : "Assunção - Solonópole"},  
1145: {id : '786', geocod : '2301703', nome : "Aurora"},  
1146: {id : '5836', geocod : '2312007', nome : "Bahia"},  
1147: {id : '5841', geocod : '2312007', nome : "Baixa Fria"},  
1148: {id : '5842', geocod : '2306306', nome : "Baixa Grande"},  
1149: {id : '822', geocod : '2301802', nome : "Baixio"},  
1150: {id : '5845', geocod : '2304269', nome : "Baixio - Deputado Irapuan Pi"},  
1151: {id : '5846', geocod : '2307403', nome : "Baixio da Donana"},  
1152: {id : '5847', geocod : '2304202', nome : "Baixio das Palmeiras"},  
1153: {id : '5852', geocod : '2306405', nome : "Baleia"},  
1154: {id : '868', geocod : '2301851', nome : "Banabuiú"},  
1155: {id : '5858', geocod : '2306603', nome : "Bandeira"},  
1156: {id : '5936', geocod : '2302701', nome : "Barão de Aquiraz"},  
1157: {id : '5868', geocod : '2303931', nome : "Barbada"},  
1158: {id : '652', geocod : '2301901', nome : "Barbalha"},  
1159: {id : '5871', geocod : '2300408', nome : "Barra - Aiuaba"},  
1160: {id : '5872', geocod : '2304459', nome : "Barra - Fortin"},  
1161: {id : '5900', geocod : '2300309', nome : "Barra do Ingá"},  
1162: {id : '5906', geocod : '2304236', nome : "Barra do Sotero"},  
1163: {id : '5884', geocod : '2313302', nome : "Barra Nova"},  
1164: {id : '925', geocod : '2301950', nome : "Barreira"},  
1165: {id : '5910', geocod : '2301109', nome : "Barreira dos Vianas"},  
1166: {id : '5911', geocod : '2305506', nome : "Barreiras - Iguatu"},  
1167: {id : '5919', geocod : '2305506', nome : "Barreiros - Iguatu"},  
1168: {id : '5920', geocod : '2311207', nome : "Barreiros - Potengi"},  
1169: {id : '5921', geocod : '2312304', nome : "Barreiros - São Benedito"},  
1170: {id : '5922', geocod : '2306405', nome : "Barrento"},  
1171: {id : '5924', geocod : '2311900', nome : "Barrinha"},  
1172: {id : '809', geocod : '2302008', nome : "Barro"},  
1173: {id : '5928', geocod : '2312007', nome : "Barro Preto"},  
1174: {id : '307', geocod : '2302057', nome : "Barroquinha"},  
1175: {id : '878', geocod : '2302107', nome : "Baturité"},  
1176: {id : '5961', geocod : '2305506', nome : "Baú"},  
1177: {id : '986', geocod : '2302206', nome : "Beberibe"},  
1178: {id : '331', geocod : '2302305', nome : "Bela Cruz"},  
1179: {id : '5975', geocod : '2302602', nome : "Bela Vista - Camocim"},  
1180: {id : '5976', geocod : '2304202', nome : "Bela Vista - Crato"},  
1181: {id : '5988', geocod : '2311405', nome : "Belém"},  
1182: {id : '5983', geocod : '2304202', nome : "Belmonte"},  
1183: {id : '5999', geocod : '2304236', nome : "Betânia - Croatá"},  
1184: {id : '6000', geocod : '2304269', nome : "Betânia - Dep. Irapuan Pinhe"},  
1185: {id : '6001', geocod : '2305209', nome : "Betânia - Hidrolândia"},  
1186: {id : '6006', geocod : '2302057', nome : "Bitupitá"},  
1187: {id : '6008', geocod : '2307601', nome : "Bixopá"},  
1188: {id : '6035', geocod : '2308708', nome : "Boa Água"},  
1189: {id : '6015', geocod : '2308807', nome : "Boa Esperança - Moraújo"},  
1190: {id : '6016', geocod : '2313203', nome : "Boa Esperança - Tamboril"},  
1191: {id : '869', geocod : '2302404', nome : "Boa Viagem"},  
1192: {id : '6022', geocod : '2308500', nome : "Boa Vista"},  
1193: {id : '6028', geocod : '2302107', nome : "Boa Vista - Baturité"},  
1194: {id : '6031', geocod : '2313351', nome : "Boa Vista do Caxitor"},  
1195: {id : '6062', geocod : '2303709', nome : "Bom Princípio"},  
1196: {id : '6074', geocod : '2312700', nome : "Bonfim - Senador Pompeu"},  
1197: {id : '6075', geocod : '2312908', nome : "Bonfim - Sobral"},  
1198: {id : '6080', geocod : '2311801', nome : "Bonhu"},  
1199: {id : '6081', geocod : '2302800', nome : "Bonito"},  
1200: {id : '6090', geocod : '2307007', nome : "Borges"},  
1201: {id : '6109', geocod : '2301307', nome : "Brejinho - Araripe"},  
1202: {id : '6110', geocod : '2302008', nome : "Brejinho - Barro"},  
1203: {id : '6113', geocod : '2312106', nome : "Brejo Grande"},  
1204: {id : '781', geocod : '2302503', nome : "Brejo Santo"},  
1205: {id : '6121', geocod : '2308377', nome : "Brotas"},  
1206: {id : '6141', geocod : '2311009', nome : "Buritizal"},  
1207: {id : '6142', geocod : '2308104', nome : "Buritizinho"},  
1208: {id : '6150', geocod : '2301109', nome : "Cabreiro"},  
1209: {id : '6155', geocod : '2307700', nome : "Cachoeira"},  
1210: {id : '6159', geocod : '2311009', nome : "Cachoeira Grande"},  
1211: {id : '6196', geocod : '2304251', nome : "Caiçara"},  
1212: {id : '6198', geocod : '2303931', nome : "Caiçarinha"},  
1213: {id : '6189', geocod : '2306504', nome : "Caio Prado"},  
1214: {id : '6190', geocod : '2312908', nome : "Caioca"},  
1215: {id : '6191', geocod : '2303303', nome : "Caipu"},  
1216: {id : '6203', geocod : '2314003', nome : "Calabaça"},  
1217: {id : '6207', geocod : '2301307', nome : "Caldeirão"},  
1218: {id : '6212', geocod : '2311306', nome : "Califórnia"},  
1219: {id : '6217', geocod : '2301000', nome : "Camará"},  
1220: {id : '6226', geocod : '2308203', nome : "Camilos"},  
1221: {id : '328', geocod : '2302602', nome : "Camocim"},  
1222: {id : '6262', geocod : '2313906', nome : "Campánario"},  
1223: {id : '6231', geocod : '2303956', nome : "Campestre - Chorozinho"},  
1224: {id : '6232', geocod : '2304459', nome : "Campestre - Fortin"},  
1225: {id : '6244', geocod : '2304202', nome : "Campo Alegre"},  
1226: {id : '6252', geocod : '2311702', nome : "Campo Lindo"},  
1227: {id : '717', geocod : '2302701', nome : "Campos Sales"},  
1228: {id : '6267', geocod : '2313500', nome : "Canaan"},  
1229: {id : '6273', geocod : '2300903', nome : "Canafístula - Apuiarés"},  
1230: {id : '6274', geocod : '2307403', nome : "Canafístula - Jucás"},  
1231: {id : '6281', geocod : '2303808', nome : "Candeias"},  
1232: {id : '6285', geocod : '2313005', nome : "Cangati"},  
1233: {id : '6286', geocod : '2308500', nome : "Cangatí - Mombaça"},  
1234: {id : '775', geocod : '2302800', nome : "Canindé"},  
1235: {id : '6289', geocod : '2305332', nome : "Canindezinho - Ibicuitinga"},  
1236: {id : '6290', geocod : '2309300', nome : "Canindezinho - Nova Russas"},  
1237: {id : '6291', geocod : '2311231', nome : "Canindezinho - Potiretama"},  
1238: {id : '6292', geocod : '2314003', nome : "Canindezinho - Várzea Alegre"},  
1239: {id : '877', geocod : '2302909', nome : "Capistrano"},  
1240: {id : '6306', geocod : '2302800', nome : "Capitão Pedro Sampai"},  
1241: {id : '6310', geocod : '2303501', nome : "Caponga"},  
1242: {id : '6311', geocod : '2301000', nome : "Caponga da Bernarda"},  
1243: {id : '6327', geocod : '2312908', nome : "Caracará"},  
1244: {id : '872', geocod : '2303006', nome : "Caridade"},  
1245: {id : '312', geocod : '2303105', nome : "Cariré"},  
1246: {id : '655', geocod : '2303204', nome : "Caririaçu"},  
1247: {id : '6356', geocod : '2304202', nome : "Caririmirim"},  
1248: {id : '651', geocod : '2303303', nome : "Cariús"},  
1249: {id : '6357', geocod : '2304301', nome : "Cariutaba"},  
1250: {id : '6359', geocod : '2302701', nome : "Carmelópolis"},  
1251: {id : '936', geocod : '2303402', nome : "Carnaubal"},  
1252: {id : '6362', geocod : '2308500', nome : "Carnaúbas"},  
1253: {id : '6360', geocod : '2308351', nome : "Carnaubinha"},  
1254: {id : '6365', geocod : '2309003', nome : "Carqueijo"},  
1255: {id : '6366', geocod : '2313302', nome : "Carrapateiras"},  
1256: {id : '6370', geocod : '2313401', nome : "Caruataí"},  
1257: {id : '6372', geocod : '2313203', nome : "Carvalho"},  
1258: {id : '6374', geocod : '2306553', nome : "Carvoeiro"},  
1259: {id : '985', geocod : '2303501', nome : "Cascavel"},  
1260: {id : '6381', geocod : '2300705', nome : "Castanhão"},  
1261: {id : '747', geocod : '2303600', nome : "Catarina"},  
1262: {id : '6395', geocod : '2308500', nome : "Catolé"},  
1263: {id : '6398', geocod : '2303709', nome : "Catuana"},  
1264: {id : '756', geocod : '2303659', nome : "Catunda"},  
1265: {id : '353', geocod : '2303709', nome : "Caucaia"},  
1266: {id : '6410', geocod : '2313351', nome : "Caxitoré - Tejuçuoca"},  
1267: {id : '6411', geocod : '2313757', nome : "Caxitoré - Umirim"},  
1268: {id : '788', geocod : '2303808', nome : "Cedro"},  
1269: {id : '6415', geocod : '2303956', nome : "Cedro - Chorozinho"},  
1270: {id : '6419', geocod : '2313559', nome : "Cemoaba"},  
1271: {id : '306', geocod : '2303907', nome : "Chaval"},  
1272: {id : '6441', geocod : '2305332', nome : "Chile"},  
1273: {id : '773', geocod : '2303931', nome : "Choró"},  
1274: {id : '926', geocod : '2303956', nome : "Chorozinho"},  
1275: {id : '6457', geocod : '2311306', nome : "Cipó dos Anjos"},  
1276: {id : '6471', geocod : '2310308', nome : "Cococi"},  
1277: {id : '6472', geocod : '2312700', nome : "Codia"},  
1278: {id : '6477', geocod : '2308104', nome : "Coité"},  
1279: {id : '6478', geocod : '2304954', nome : "Colina"},  
1280: {id : '6496', geocod : '2305209', nome : "Conceição - Hidrolândia"},  
1281: {id : '6497', geocod : '2313559', nome : "Conceição - Tururu"},  
1282: {id : '314', geocod : '2304004', nome : "Coreaú"},  
1283: {id : '6628', geocod : '2301950', nome : "Córrego"},  
1284: {id : '6636', geocod : '2304459', nome : "Córrego dos Fernandes"},  
1285: {id : '6546', geocod : '2307106', nome : "Corrente"},  
1286: {id : '749', geocod : '2304103', nome : "Crateús"},  
1287: {id : '647', geocod : '2304202', nome : "Crato"},  
1288: {id : '6571', geocod : '2310803', nome : "Crioulos"},  
1289: {id : '6573', geocod : '2303501', nome : "Cristais"},  
1290: {id : '742', geocod : '2304236', nome : "Croatá"},  
1291: {id : '6586', geocod : '2312403', nome : "Croatá - São Gonçalo do Amar"},  
1292: {id : '6585', geocod : '2313955', nome : "Croata - Varjota"},  
1293: {id : '332', geocod : '2304251', nome : "Cruz"},  
1294: {id : '6590', geocod : '2306306', nome : "Cruz - Itapagé"},  
1295: {id : '6595', geocod : '2305407', nome : "Cruzeirinho"},  
1296: {id : '6607', geocod : '2302008', nome : "Cuncas"},  
1297: {id : '6611', geocod : '2313203', nome : "Curatis"},  
1298: {id : '6619', geocod : '2309458', nome : "Curupira"},  
1299: {id : '6624', geocod : '2311306', nome : "Custódio"},  
1300: {id : '6643', geocod : '2311405', nome : "Damião Carneiro"},  
1301: {id : '6644', geocod : '2311306', nome : "Daniel de Queiróz"},  
1302: {id : '864', geocod : '2304269', nome : "Dep. Irapuan Pinheiro"},  
1303: {id : '6659', geocod : '2306405', nome : "Deserto"},  
1304: {id : '6678', geocod : '2312106', nome : "Dom Leme"},  
1305: {id : '6679', geocod : '2311306', nome : "Dom Maurício"},  
1306: {id : '6681', geocod : '2304202', nome : "Dom Quintino"},  
1307: {id : '6683', geocod : '2302404', nome : "Domingos da Costa"},  
1308: {id : '6687', geocod : '2310951', nome : "Donato"},  
1309: {id : '6696', geocod : '2305233', nome : "Dourados"},  
1310: {id : '6709', geocod : '2300309', nome : "Ebron"},  
1311: {id : '6714', geocod : '2306009', nome : "Ema"},  
1312: {id : '6715', geocod : '2305605', nome : "Ematuba"},  
1313: {id : '6718', geocod : '2311405', nome : "Encantado"},  
1314: {id : '6726', geocod : '2305902', nome : "Engenheiro João Tomé"},  
1315: {id : '6725', geocod : '2312700', nome : "Engenheiro José Lope"},  
1316: {id : '6735', geocod : '2302008', nome : "Engenho Velho"},  
1317: {id : '889', geocod : '2304277', nome : "Ererê"},  
1318: {id : '6751', geocod : '2311702', nome : "Espacinha"},  
1319: {id : '6752', geocod : '2302800', nome : "Esperança"},  
1320: {id : '6773', geocod : '2301901', nome : "Estrela"},  
1321: {id : '355', geocod : '2304285', nome : "Eusébio"},  
1322: {id : '648', geocod : '2304301', nome : "Farias Brito"},  
1323: {id : '6873', geocod : '2309805', nome : "Fátima"},  
1324: {id : '6808', geocod : '2306900', nome : "Feiticeiro"},  
1325: {id : '6810', geocod : '2303204', nome : "Feitosa"},  
1326: {id : '6812', geocod : '2311900', nome : "Felipe"},  
1327: {id : '6814', geocod : '2305704', nome : "Felizardo"},  
1328: {id : '6831', geocod : '2311900', nome : "Flamengo"},  
1329: {id : '6833', geocod : '2305803', nome : "Flores - Ipu"},  
1330: {id : '6834', geocod : '2311801', nome : "Flores - Russas"},  
1331: {id : '322', geocod : '2304350', nome : "Forquilha"},  
1332: {id : '5472', geocod : '2304400', nome : "Fortaleza"},  
1333: {id : '934', geocod : '2304459', nome : "Fortim"},  
1334: {id : '310', geocod : '2304509', nome : "Frecheirinha"},  
1335: {id : '6878', geocod : '2309805', nome : "Gado"},  
1336: {id : '6879', geocod : '2310100', nome : "Gados dos Rodrigues"},  
1337: {id : '6889', geocod : '2308401', nome : "Gameleira de São Sebastião"},  
1338: {id : '6897', geocod : '2300754', nome : "Garças"},  
1339: {id : '6898', geocod : '2310308', nome : "Gavião"},  
1340: {id : '6998', geocod : '2305902', nome : "Gázea"},  
1341: {id : '942', geocod : '2304608', nome : "General Sampaio"},  
1342: {id : '6903', geocod : '2314102', nome : "General Tibúrcio"},  
1343: {id : '6906', geocod : '2306256', nome : "Gererau"},  
1344: {id : '6909', geocod : '2307007', nome : "Giqui"},  
1345: {id : '939', geocod : '2304657', nome : "Graça"},  
1346: {id : '327', geocod : '2304707', nome : "Granja"},  
1347: {id : '656', geocod : '2304806', nome : "Granjeiro"},  
1348: {id : '313', geocod : '2304905', nome : "Groaíras"},  
1349: {id : '981', geocod : '2304954', nome : "Guaiúba"},  
1350: {id : '6951', geocod : '2304459', nome : "Guajiru"},  
1351: {id : '6955', geocod : '2303501', nome : "Guanacés"},  
1352: {id : '938', geocod : '2305001', nome : "Guaraciaba do Norte"},  
1353: {id : '876', geocod : '2305100', nome : "Guaramiranga"},  
1354: {id : '6969', geocod : '2303709', nome : "Guararu"},  
1355: {id : '6981', geocod : '2311603', nome : "Guassi"},  
1356: {id : '6983', geocod : '2309508', nome : "Guassussê"},  
1357: {id : '6990', geocod : '2302404', nome : "Guia"},  
1358: {id : '6996', geocod : '2302602', nome : "Guriú"},  
1359: {id : '646', geocod : '2305209', nome : "Hidrolândia"},  
1360: {id : '7011', geocod : '2313203', nome : "Holanda"},  
1361: {id : '982', geocod : '2305233', nome : "Horizonte"},  
1362: {id : '7016', geocod : '2305605', nome : "Iapi"},  
1363: {id : '7017', geocod : '2302008', nome : "Iara"},  
1364: {id : '919', geocod : '2305266', nome : "Ibaretama"},  
1365: {id : '7024', geocod : '2304103', nome : "Ibiapaba"},  
1366: {id : '309', geocod : '2305308', nome : "Ibiapina"},  
1367: {id : '7028', geocod : '2314003', nome : "Ibicatu"},  
1368: {id : '7030', geocod : '2310902', nome : "Ibicuã"},  
1369: {id : '7029', geocod : '2305357', nome : "Ibicuitaba"},  
1370: {id : '920', geocod : '2305332', nome : "Ibicuitinga"},  
1371: {id : '7054', geocod : '2307502', nome : "Iborepi"},  
1372: {id : '7055', geocod : '2302404', nome : "Ibuaçu"},  
1373: {id : '7056', geocod : '2304707', nome : "Ibuguaçu"},  
1374: {id : '1086', geocod : '2305357', nome : "Icapuí"},  
1375: {id : '7060', geocod : '2300754', nome : "Icaraí"},  
1376: {id : '791', geocod : '2305407', nome : "Icó"},  
1377: {id : '7063', geocod : '2305407', nome : "Icozinho"},  
1378: {id : '7066', geocod : '2301208', nome : "Ideal"},  
1379: {id : '7074', geocod : '2309508', nome : "Igarói"},  
1380: {id : '7082', geocod : '2302800', nome : "Iguaçu"},  
1381: {id : '789', geocod : '2305506', nome : "Iguatu"},  
1382: {id : '748', geocod : '2305605', nome : "Independência"},  
1383: {id : '7102', geocod : '2301703', nome : "Ingazeiras"},  
1384: {id : '7105', geocod : '2313302', nome : "Inhamuns"},  
1385: {id : '7113', geocod : '2312304', nome : "Inhuçu"},  
1386: {id : '7111', geocod : '2312106', nome : "Inhumas"},  
1387: {id : '7112', geocod : '2304657', nome : "Inhuporanga"},  
1388: {id : '7119', geocod : '2308005', nome : "Ipaguaçu"},  
1389: {id : '751', geocod : '2305654', nome : "Ipaporanga"},  
1390: {id : '821', geocod : '2305704', nome : "Ipaumirim"},  
1391: {id : '753', geocod : '2305803', nome : "Ipu"},  
1392: {id : '7129', geocod : '2306405', nome : "Ipu Mazagão"},  
1393: {id : '752', geocod : '2305902', nome : "Ipueiras"},  
1394: {id : '7132', geocod : '2305506', nome : "Ipueiras dos Gomes"},  
1395: {id : '890', geocod : '2306009', nome : "Iracema"},  
1396: {id : '7137', geocod : '2305209', nome : "Irajá"},  
1397: {id : '7142', geocod : '2304103', nome : "Irapuá"},  
1398: {id : '7145', geocod : '2306306', nome : "Iratinga"},  
1399: {id : '335', geocod : '2306108', nome : "Irauçuba"},  
1400: {id : '7152', geocod : '2300309', nome : "Isidoro"},  
1401: {id : '7159', geocod : '2304954', nome : "Itacima"},  
1402: {id : '7164', geocod : '2302701', nome : "Itaguá"},  
1403: {id : '932', geocod : '2306207', nome : "Itaiçaba"},  
1404: {id : '7172', geocod : '2309607', nome : "Itaipaba"},  
1405: {id : '354', geocod : '2306256', nome : "Itaitinga"},  
1406: {id : '7187', geocod : '2304905', nome : "Itamaracá"},  
1407: {id : '7198', geocod : '2306504', nome : "Itans"},  
1408: {id : '336', geocod : '2306306', nome : "Itapagé"},  
1409: {id : '7202', geocod : '2307700', nome : "Itapebussu"},  
1410: {id : '7204', geocod : '2302206', nome : "Itapeim"},  
1411: {id : '342', geocod : '2306405', nome : "Itapipoca"},  
1412: {id : '873', geocod : '2306504', nome : "Itapiúna"},  
1413: {id : '334', geocod : '2306553', nome : "Itarema"},  
1414: {id : '871', geocod : '2306603', nome : "Itatira"},  
1415: {id : '7249', geocod : '2313609', nome : "Jaburuna"},  
1416: {id : '7251', geocod : '2302404', nome : "Jacampari"},  
1417: {id : '7253', geocod : '2303501', nome : "Jacarecoara"},  
1418: {id : '7256', geocod : '2304400', nome : "Jacaúna"},  
1419: {id : '7278', geocod : '2301208', nome : "Jaguarão"},  
1420: {id : '891', geocod : '2306702', nome : "Jaguaretama"},  
1421: {id : '892', geocod : '2306801', nome : "Jaguaribara"},  
1422: {id : '881', geocod : '2306900', nome : "Jaguaribe"},  
1423: {id : '931', geocod : '2307007', nome : "Jaguaruana"},  
1424: {id : '7281', geocod : '2312908', nome : "Jaibaras"},  
1425: {id : '7282', geocod : '2308401', nome : "Jamacaru"},  
1426: {id : '7285', geocod : '2305605', nome : "Jandrangoeira"},  
1427: {id : '778', geocod : '2307106', nome : "Jardim"},  
1428: {id : '7295', geocod : '2310209', nome : "Jardim - Paracuru"},  
1429: {id : '7302', geocod : '2307106', nome : "Jardimirim"},  
1430: {id : '780', geocod : '2307205', nome : "Jati"},  
1431: {id : '7309', geocod : '2301208', nome : "Jenipapeiro"},  
1432: {id : '7311', geocod : '2307254', nome : "Jericoacoara"},  
1433: {id : '329', geocod : '2307254', nome : "Jijoca de Jericoacoara"},  
1434: {id : '7313', geocod : '2301109', nome : "Jirau"},  
1435: {id : '7337', geocod : '2312007', nome : "João Cordeiro"},  
1436: {id : '7324', geocod : '2312908', nome : "Jordão"},  
1437: {id : '7333', geocod : '2305506', nome : "José de Alencar"},  
1438: {id : '7378', geocod : '2306108', nome : "Juá - Irauçuba"},  
1439: {id : '7379', geocod : '2311306', nome : "Juá - Quixadá"},  
1440: {id : '7342', geocod : '2311306', nome : "Juatama"},  
1441: {id : '7343', geocod : '2308708', nome : "Juazeiro de Baixo"},  
1442: {id : '653', geocod : '2307304', nome : "Juazeiro do Norte"},  
1443: {id : '7344', geocod : '2307700', nome : "Jubaia"},  
1444: {id : '650', geocod : '2307403', nome : "Jucás"},  
1445: {id : '7361', geocod : '2304400', nome : "Jurema"},  
1446: {id : '7363', geocod : '2300200', nome : "Juritianha"},  
1447: {id : '7373', geocod : '2301000', nome : "Justiniano de Serpa"},  
1448: {id : '7382', geocod : '2311405', nome : "Lacerda"},  
1449: {id : '7384', geocod : '2307700', nome : "Ladeira Grande"},  
1450: {id : '7386', geocod : '2303808', nome : "Lagedo"},  
1451: {id : '7387', geocod : '2307700', nome : "Lages"},  
1452: {id : '7407', geocod : '2301208', nome : "Lagoa de São João"},  
1453: {id : '7408', geocod : '2301950', nome : "Lagoa do Barro"},  
1454: {id : '7411', geocod : '2300200', nome : "Lagoa do Carneiro"},  
1455: {id : '7412', geocod : '2307700', nome : "Lagoa do Juvenal"},  
1456: {id : '7413', geocod : '2306603', nome : "Lagoa do Mato"},  
1457: {id : '7415', geocod : '2311959', nome : "Lagoa dos Crioulos"},  
1458: {id : '7399', geocod : '2301950', nome : "Lagoa Grande - Barreira"},  
1459: {id : '7400', geocod : '2306405', nome : "Lagoa Grande - Itapipoca"},  
1460: {id : '7401', geocod : '2311801', nome : "Lagoa Grande - Russas"},  
1461: {id : '7417', geocod : '2311504', nome : "Lagoinha"},  
1462: {id : '7435', geocod : '2314102', nome : "Lambedouro"},  
1463: {id : '7437', geocod : '2304657', nome : "Lapa"},  
1464: {id : '7441', geocod : '2301851', nome : "Laranjeiras"},  
1465: {id : '790', geocod : '2307502', nome : "Lavras da Mangabeira"},  
1466: {id : '7453', geocod : '2305407', nome : "Lima Campos"},  
1467: {id : '927', geocod : '2307601', nome : "Limoeiro do Norte"},  
1468: {id : '7463', geocod : '2312205', nome : "Lisieux"},  
1469: {id : '7464', geocod : '2305902', nome : "Livramento"},  
1470: {id : '7468', geocod : '2312205', nome : "Logradouro"},  
1471: {id : '7493', geocod : '2311009', nome : "Macambira"},  
1472: {id : '7494', geocod : '2307635', nome : "Macaoca"},  
1473: {id : '7495', geocod : '2312205', nome : "Macaraú"},  
1474: {id : '7496', geocod : '2304459', nome : "Maceió"},  
1475: {id : '870', geocod : '2307635', nome : "Madalena"},  
1476: {id : '7509', geocod : '2309300', nome : "Major Simplício"},  
1477: {id : '7510', geocod : '2311900', nome : "Malhada"},  
1478: {id : '7511', geocod : '2312205', nome : "Malhada Grande"},  
1479: {id : '7518', geocod : '2307502', nome : "Mangabeira"},  
1480: {id : '7523', geocod : '2314102', nome : "Manhoso"},  
1481: {id : '7525', geocod : '2305357', nome : "Manibu"},  
1482: {id : '7527', geocod : '2311405', nome : "Manituba"},  
1483: {id : '7530', geocod : '2308500', nome : "Manoel Correia"},  
1484: {id : '7531', geocod : '2307700', nome : "Manoel Guedes"},  
1485: {id : '7536', geocod : '2306900', nome : "Mapuá"},  
1486: {id : '352', geocod : '2307650', nome : "Maracanaú"},  
1487: {id : '7540', geocod : '2308104', nome : "Maraguá"},  
1488: {id : '351', geocod : '2307700', nome : "Maranguape"},  
1489: {id : '7546', geocod : '2308104', nome : "Mararupá"},  
1490: {id : '7548', geocod : '2303931', nome : "Maravilha"},  
1491: {id : '330', geocod : '2307809', nome : "Marco"},  
1492: {id : '7571', geocod : '2306405', nome : "Marinheiros"},  
1493: {id : '7581', geocod : '2313302', nome : "Marrecas"},  
1494: {id : '7582', geocod : '2307304', nome : "Marrocos"},  
1495: {id : '7583', geocod : '2313302', nome : "Marruás"},  
1496: {id : '316', geocod : '2307908', nome : "Martinópole"},  
1497: {id : '7587', geocod : '2305001', nome : "Martinslândia"},  
1498: {id : '323', geocod : '2308005', nome : "Massapê"},  
1499: {id : '7594', geocod : '2301109', nome : "Mata Fresca"},  
1500: {id : '7602', geocod : '2310704', nome : "Matias"},  
1501: {id : '7607', geocod : '2305902', nome : "Matriz"},  
1502: {id : '768', geocod : '2308104', nome : "Mauriti"},  
1503: {id : '7613', geocod : '2307403', nome : "Mel"},  
1504: {id : '318', geocod : '2308203', nome : "Meruoca"},  
1505: {id : '7621', geocod : '2304400', nome : "Messejana"},  
1506: {id : '7625', geocod : '2303204', nome : "Miguel Xavier"},  
1507: {id : '657', geocod : '2308302', nome : "Milagres"},  
1508: {id : '866', geocod : '2308351', nome : "Milhã"},  
1509: {id : '7628', geocod : '2301208', nome : "Milton Belo"},  
1510: {id : '7634', geocod : '2310506', nome : "Mineirolândia"},  
1511: {id : '7639', geocod : '2303204', nome : "Miragem"},  
1512: {id : '326', geocod : '2308377', nome : "Miraíma"},  
1513: {id : '7644', geocod : '2303709', nome : "Mirambé"},  
1514: {id : '7645', geocod : '2310308', nome : "Miranda"},  
1515: {id : '7657', geocod : '2308401', nome : "Missão Nova"},  
1516: {id : '654', geocod : '2308401', nome : "Missão Velha"},  
1517: {id : '7655', geocod : '2306108', nome : "Missi"},  
1518: {id : '7661', geocod : '2307809', nome : "Mocambo"},  
1519: {id : '7665', geocod : '2300754', nome : "Moitas"},  
1520: {id : '860', geocod : '2308500', nome : "Mombaça"},  
1521: {id : '7666', geocod : '2304400', nome : "Mondubim"},  
1522: {id : '7668', geocod : '2309706', nome : "Monguba"},  
1523: {id : '764', geocod : '2308609', nome : "Monsenhor Tabosa"},  
1524: {id : '7678', geocod : '2302800', nome : "Monte Alegre"},  
1525: {id : '7677', geocod : '2302008', nome : "Monte Alegre - Barro"},  
1526: {id : '7681', geocod : '2304202', nome : "Monte Alverne"},  
1527: {id : '7687', geocod : '2300408', nome : "Monte Castelo"},  
1528: {id : '7688', geocod : '2303931', nome : "Monte Castelo - Choró"},  
1529: {id : '7694', geocod : '2308351', nome : "Monte Grave"},  
1530: {id : '7700', geocod : '2305605', nome : "Monte Sinai"},  
1531: {id : '7702', geocod : '2310308', nome : "Monte Sion"},  
1532: {id : '7706', geocod : '2304103', nome : "Montenebo"},  
1533: {id : '921', geocod : '2308708', nome : "Morada Nova"},  
1534: {id : '315', geocod : '2308807', nome : "Moraújo"},  
1535: {id : '325', geocod : '2308906', nome : "Morrinhos"},  
1536: {id : '7716', geocod : '2305001', nome : "Morrinhos Novos"},  
1537: {id : '7718', geocod : '2306603', nome : "Morro Branco"},  
1538: {id : '311', geocod : '2309003', nome : "Mucambo"},  
1539: {id : '7736', geocod : '2305001', nome : "Mucambo - Guaraciba do Norte"},  
1540: {id : '875', geocod : '2309102', nome : "Mulungu"},  
1541: {id : '7740', geocod : '2310902', nome : "Mulungu - Piquet Carneiro"},  
1542: {id : '7742', geocod : '2308005', nome : "Mumbaba"},  
1543: {id : '7743', geocod : '2313500', nome : "Mundaú"},  
1544: {id : '7751', geocod : '2312205', nome : "Muribeca"},  
1545: {id : '7756', geocod : '2312007', nome : "Mutambeiras"},  
1546: {id : '7762', geocod : '2314003', nome : "Naraniú"},  
1547: {id : '7763', geocod : '2300754', nome : "Nascente"},  
1548: {id : '7773', geocod : '2311405', nome : "Nenelândia"},  
1549: {id : '7800', geocod : '2308609', nome : "Nossa Senhora do Livramento"},  
1550: {id : '7811', geocod : '2309300', nome : "Nova Betânia"},  
1551: {id : '7810', geocod : '2304301', nome : "Nova Betânia - Farias Brito"},  
1552: {id : '7825', geocod : '2305902', nome : "Nova Fátima"},  
1553: {id : '7824', geocod : '2306900', nome : "Nova Floresta"},  
1554: {id : '784', geocod : '2309201', nome : "Nova Olinda"},  
1555: {id : '755', geocod : '2309300', nome : "Nova Russas"},  
1556: {id : '7843', geocod : '2313609', nome : "Nova Veneza"},  
1557: {id : '7846', geocod : '2305266', nome : "Nova Vida"},  
1558: {id : '7849', geocod : '2310308', nome : "Novo Assis"},  
1559: {id : '7853', geocod : '2309458', nome : "Novo Horizonte"},  
1560: {id : '745', geocod : '2309409', nome : "Novo Oriente"},  
1561: {id : '922', geocod : '2309458', nome : "Ocara"},  
1562: {id : '7867', geocod : '2304103', nome : "Oiticica - Crateús"},  
1563: {id : '7868', geocod : '2305266', nome : "Oiticica - Ibaretama"},  
1564: {id : '7869', geocod : '2310308', nome : "Oiticica - Parambu"},  
1565: {id : '7873', geocod : '2313104', nome : "Olho d-gua da Bica"},  
1566: {id : '7879', geocod : '2313203', nome : "Oliveiras"},  
1567: {id : '865', geocod : '2309508', nome : "Orós"},  
1568: {id : '983', geocod : '2309607', nome : "Pacajus"},  
1569: {id : '350', geocod : '2309706', nome : "Pacatuba"},  
1570: {id : '880', geocod : '2309805', nome : "Pacoti"},  
1571: {id : '308', geocod : '2309904', nome : "Pacujá"},  
1572: {id : '7912', geocod : '2307304', nome : "Padre Cícero"},  
1573: {id : '7918', geocod : '2308005', nome : "Padre Linhares"},  
1574: {id : '7923', geocod : '2314102', nome : "Padre Vieira"},  
1575: {id : '7932', geocod : '2301307', nome : "Pajeú"},  
1576: {id : '7933', geocod : '2307650', nome : "Pajuçara"},  
1577: {id : '7936', geocod : '2309508', nome : "Palestina"},  
1578: {id : '7938', geocod : '2308203', nome : "Palestina do Norte"},  
1579: {id : '930', geocod : '2310001', nome : "Palhano"},  
1580: {id : '944', geocod : '2310100', nome : "Palmácia"},  
1581: {id : '7942', geocod : '2306504', nome : "Palmatória"},  
1582: {id : '7955', geocod : '2307809', nome : "Panacuí"},  
1583: {id : '7962', geocod : '2307700', nome : "Papara"},  
1584: {id : '7964', geocod : '2313906', nome : "Paracuá"},  
1585: {id : '349', geocod : '2310209', nome : "Paracuru"},  
1586: {id : '346', geocod : '2310258', nome : "Paraipaba"},  
1587: {id : '7981', geocod : '2303659', nome : "Paraíso"},  
1588: {id : '7968', geocod : '2302206', nome : "Parajuru"},  
1589: {id : '743', geocod : '2310308', nome : "Parambu"},  
1590: {id : '943', geocod : '2310407', nome : "Paramoti"},  
1591: {id : '7972', geocod : '2304400', nome : "Parangaba"},  
1592: {id : '7976', geocod : '2312007', nome : "Parapuí"},  
1593: {id : '7980', geocod : '2304707', nome : "Parazinho"},  
1594: {id : '7989', geocod : '2302206', nome : "Paripueira"},  
1595: {id : '7996', geocod : '2309607', nome : "Pascoal"},  
1596: {id : '7999', geocod : '2303907', nome : "Passagem - Chaval"},  
1597: {id : '8000', geocod : '2311405', nome : "Passagem - Quixeramobim"},  
1598: {id : '8002', geocod : '2314102', nome : "Passagem da Onça"},  
1599: {id : '8023', geocod : '2313005', nome : "Pasta"},  
1600: {id : '8024', geocod : '2301000', nome : "Patacas"},  
1601: {id : '8028', geocod : '2303956', nome : "Patos dos Liberatos"},  
1602: {id : '8029', geocod : '2312908', nome : "Patriarca"},  
1603: {id : '8041', geocod : '2309706', nome : "Pavuna"},  
1604: {id : '8043', geocod : '2312403', nome : "Pecém"},  
1605: {id : '861', geocod : '2310506', nome : "Pedra Branca"},  
1606: {id : '8044', geocod : '2301208', nome : "Pedra Branca - Aracoiaba"},  
1607: {id : '8053', geocod : '2308708', nome : "Pedras"},  
1608: {id : '8055', geocod : '2301851', nome : "Pedras Brancas"},  
1609: {id : '8059', geocod : '2305407', nome : "Pedrinhas"},  
1610: {id : '8067', geocod : '2311801', nome : "Peixe"},  
1611: {id : '8069', geocod : '2313104', nome : "Peixe Gordo"},  
1612: {id : '777', geocod : '2310605', nome : "Penaforte"},  
1613: {id : '8073', geocod : '2307700', nome : "Penedo"},  
1614: {id : '344', geocod : '2310704', nome : "Pentecoste"},  
1615: {id : '884', geocod : '2310803', nome : "Pereiro"},  
1616: {id : '8085', geocod : '2305100', nome : "Pernambuquinho"},  
1617: {id : '8092', geocod : '2304707', nome : "Pessoa Anta"},  
1618: {id : '8112', geocod : '2313401', nome : "Pindoguaba"},  
1619: {id : '984', geocod : '2310852', nome : "Pindoretama"},  
1620: {id : '8139', geocod : '2313708', nome : "Pio X"},  
1621: {id : '859', geocod : '2310902', nome : "Piquet Carneiro"},  
1622: {id : '8151', geocod : '2305266', nome : "Piranji"},  
1623: {id : '754', geocod : '2310951', nome : "Pires Ferreira"},  
1624: {id : '8169', geocod : '2303501', nome : "Pitombeira"},  
1625: {id : '8171', geocod : '2303501', nome : "Pitombeiras"},  
1626: {id : '8182', geocod : '2301208', nome : "Plácido Martins"},  
1627: {id : '8249', geocod : '2302503', nome : "Poço"},  
1628: {id : '8252', geocod : '2300754', nome : "Poço Comprido - Amontada"},  
1629: {id : '8253', geocod : '2306801', nome : "Poço Comprido - Jaguaribara"},  
1630: {id : '8254', geocod : '2310209', nome : "Poço Doce"},  
1631: {id : '8257', geocod : '2307403', nome : "Poço Grande"},  
1632: {id : '8185', geocod : '2308302', nome : "Podimirim"},  
1633: {id : '8191', geocod : '2304202', nome : "Ponta da Serra"},  
1634: {id : '8194', geocod : '2312106', nome : "Pontal da Santa Cruz"},  
1635: {id : '739', geocod : '2311009', nome : "Poranga"},  
1636: {id : '8212', geocod : '2310704', nome : "Porfírio Sampaio"},  
1637: {id : '779', geocod : '2311108', nome : "Porteiras"},  
1638: {id : '718', geocod : '2311207', nome : "Potengi"},  
1639: {id : '8242', geocod : '2304103', nome : "Poti"},  
1640: {id : '915', geocod : '2311231', nome : "Potiretama"},  
1641: {id : '8273', geocod : '2302305', nome : "Prata"},  
1642: {id : '8277', geocod : '2313005', nome : "Prefeito Suely Pinheiro"},  
1643: {id : '8310', geocod : '2314102', nome : "Quatiguaba"},  
1644: {id : '8315', geocod : '2305233', nome : "Queimados"},  
1645: {id : '8321', geocod : '2308401', nome : "Quimami"},  
1646: {id : '8322', geocod : '2300309', nome : "Quincoê"},  
1647: {id : '8323', geocod : '2304301', nome : "Quincuncá"},  
1648: {id : '8329', geocod : '2307502', nome : "Quitaiús"},  
1649: {id : '744', geocod : '2311264', nome : "Quiterianópolis"},  
1650: {id : '774', geocod : '2311306', nome : "Quixadá"},  
1651: {id : '8333', geocod : '2302701', nome : "Quixariú"},  
1652: {id : '863', geocod : '2311355', nome : "Quixelô"},  
1653: {id : '772', geocod : '2311405', nome : "Quixeramobim"},  
1654: {id : '928', geocod : '2311504', nome : "Quixeré"},  
1655: {id : '8334', geocod : '2305506', nome : "Quixoá"},  
1656: {id : '8335', geocod : '2312908', nome : "Rafael Arruda"},  
1657: {id : '8336', geocod : '2312205', nome : "Raimundo Martins"},  
1658: {id : '923', geocod : '2311603', nome : "Redenção"},  
1659: {id : '940', geocod : '2311702', nome : "Reriutaba"},  
1660: {id : '8361', geocod : '2301307', nome : "Riacho Grande"},  
1661: {id : '8365', geocod : '2314003', nome : "Riacho Verde"},  
1662: {id : '8387', geocod : '2311405', nome : "Rinaré"},  
1663: {id : '8449', geocod : '2308708', nome : "Roldão"},  
1664: {id : '929', geocod : '2311801', nome : "Russas"},  
1665: {id : '8467', geocod : '2300754', nome : "Sabiaguaba"},  
1666: {id : '723', geocod : '2311900', nome : "Saboeiro"},  
1667: {id : '8469', geocod : '2305654', nome : "Sacramento"},  
1668: {id : '8485', geocod : '2312809', nome : "Salão"},  
1669: {id : '714', geocod : '2311959', nome : "Salitre"},  
1670: {id : '8488', geocod : '2304707', nome : "Sambaíba"},  
1671: {id : '8495', geocod : '2309805', nome : "Santa Ana"},  
1672: {id : '8511', geocod : '2310506', nome : "Santa Cruz do Banabu"},  
1673: {id : '8526', geocod : '2304202', nome : "Santa Fé"},  
1674: {id : '8523', geocod : '2300309', nome : "Santa Felícia"},  
1675: {id : '8543', geocod : '2313807', nome : "Santa Luzia"},  
1676: {id : '757', geocod : '2312205', nome : "Santa Quitéria"},  
1677: {id : '8581', geocod : '2304202', nome : "Santa Rosa"},  
1678: {id : '8591', geocod : '2301109', nome : "Santa Tereza - Aracati"},  
1679: {id : '8592', geocod : '2304236', nome : "Santa Tereza - Croatá"},  
1680: {id : '8593', geocod : '2309409', nome : "Santa Tereza - Novo Oriente"},  
1681: {id : '324', geocod : '2312007', nome : "Santana do Acaraú"},  
1682: {id : '783', geocod : '2312106', nome : "Santana do Cariri"},  
1683: {id : '8617', geocod : '2309508', nome : "Santarém"},  
1684: {id : '8635', geocod : '2300309', nome : "Santo Antônio - Acopiara"},  
1685: {id : '8626', geocod : '2301257', nome : "Santo Antonio - Ararendá"},  
1686: {id : '8636', geocod : '2302008', nome : "Santo Antônio - Barro"},  
1687: {id : '8637', geocod : '2303808', nome : "Santo Antônio - Cedro"},  
1688: {id : '8638', geocod : '2304103', nome : "Santo Antônio - Crateús"},  
1689: {id : '8644', geocod : '2305308', nome : "Santo Antônio da Pindoba"},  
1690: {id : '8672', geocod : '2308203', nome : "Santo Antônio dos Fernandes"},  
1691: {id : '8682', geocod : '2310951', nome : "Santo Izidro"},  
1692: {id : '8813', geocod : '2303303', nome : "São Bartolomeu"},  
1693: {id : '937', geocod : '2312304', nome : "São Benedito"},  
1694: {id : '8821', geocod : '2311306', nome : "São Bernardo"},  
1695: {id : '8836', geocod : '2303006', nome : "São Domingos"},  
1696: {id : '8843', geocod : '2302503', nome : "São Filipe"},  
1697: {id : '8846', geocod : '2308203', nome : "São Francisco - Meruoca"},  
1698: {id : '8847', geocod : '2311264', nome : "São Francisco - Quiterianópo"},  
1699: {id : '8864', geocod : '2311603', nome : "São Gerardo"},  
1700: {id : '348', geocod : '2312403', nome : "São Gonçalo do Amarante"},  
1701: {id : '8869', geocod : '2308500', nome : "São Gonçalo do Umari"},  
1702: {id : '8956', geocod : '2311801', nome : "São João de Deus"},  
1703: {id : '8960', geocod : '2307700', nome : "São João do Amanari"},  
1704: {id : '916', geocod : '2312502', nome : "São João do Jaguaribe"},  
1705: {id : '8971', geocod : '2311306', nome : "São João dos Queiroz"},  
1706: {id : '8872', geocod : '2312403', nome : "São Joaquim"},  
1707: {id : '8880', geocod : '2312700', nome : "São Joaquim do Salgado"},  
1708: {id : '8893', geocod : '2300101', nome : "São José - Abaiara"},  
1709: {id : '8894', geocod : '2306009', nome : "São José - Iracema"},  
1710: {id : '8895', geocod : '2307007', nome : "São José - Jaguaruana"},  
1711: {id : '8896', geocod : '2310001', nome : "São José - Palhano"},  
1712: {id : '8897', geocod : '2311900', nome : "São José - Saboeiro"},  
1713: {id : '8904', geocod : '2305902', nome : "São José das Lontras"},  
1714: {id : '8910', geocod : '2313005', nome : "São José de Solonópo"},  
1715: {id : '8934', geocod : '2312908', nome : "São José do Torto"},  
1716: {id : '345', geocod : '2312601', nome : "São Luís do Curu"},  
1717: {id : '9004', geocod : '2303808', nome : "São Miguel - Cedro"},  
1718: {id : '9005', geocod : '2308104', nome : "São Miguel - Mauriti"},  
1719: {id : '9006', geocod : '2311405', nome : "São Miguel - Quixeramobim"},  
1720: {id : '9011', geocod : '2309300', nome : "São Pedro"},  
1721: {id : '9036', geocod : '2307403', nome : "São Pedro do Norte"},  
1722: {id : '9043', geocod : '2300606', nome : "São Romão"},  
1723: {id : '9045', geocod : '2304236', nome : "São Roque"},  
1724: {id : '9059', geocod : '2302107', nome : "São Sebastião - Baturité"},  
1725: {id : '9060', geocod : '2303303', nome : "São Sebastião - Cariús"},  
1726: {id : '9103', geocod : '2308500', nome : "São Vicente - Mombaça"},  
1727: {id : '9104', geocod : '2312908', nome : "São Vicente - Sobral"},  
1728: {id : '8698', geocod : '2312007', nome : "Sapó"},  
1729: {id : '8696', geocod : '2307700', nome : "Sapupara"},  
1730: {id : '8706', geocod : '2310704', nome : "Sebastião de Abreu"},  
1731: {id : '8716', geocod : '2309706', nome : "Senador Carlos Jereissate"},  
1732: {id : '862', geocod : '2312700', nome : "Senador Pompeu"},  
1733: {id : '320', geocod : '2312809', nome : "Senador Sá"},  
1734: {id : '8725', geocod : '2309458', nome : "Sereno de Cima"},  
1735: {id : '8741', geocod : '2302206', nome : "Serra do Félix"},  
1736: {id : '8746', geocod : '2309458', nome : "Serragem"},  
1737: {id : '8752', geocod : '2302008', nome : "Serrota - Barro"},  
1738: {id : '8753', geocod : '2312809', nome : "Serrota - Senador Sá"},  
1739: {id : '8754', geocod : '2312403', nome : "Serrote"},  
1740: {id : '8775', geocod : '2301851', nome : "Sitiá"},  
1741: {id : '9112', geocod : '2308906', nome : "Sítio Alegre"},  
1742: {id : '9117', geocod : '2305407', nome : "Sítio Velho"},  
1743: {id : '9121', geocod : '2303709', nome : "Sítios Novos"},  
1744: {id : '8776', geocod : '2312403', nome : "Siupé"},  
1745: {id : '321', geocod : '2312908', nome : "Sobral"},  
1746: {id : '8785', geocod : '2306405', nome : "Soledade"},  
1747: {id : '867', geocod : '2313005', nome : "Solonópole"},  
1748: {id : '8797', geocod : '2305506', nome : "Suassurana"},  
1749: {id : '8801', geocod : '2302206', nome : "Sucatinga"},  
1750: {id : '8802', geocod : '2313203', nome : "Sucesso"},  
1751: {id : '8809', geocod : '2305001', nome : "Sussuanha"},  
1752: {id : '9122', geocod : '2313401', nome : "Tabainha"},  
1753: {id : '9129', geocod : '2311900', nome : "Taboleiro"},  
1754: {id : '917', geocod : '2313104', nome : "Tabuleiro do Norte"},  
1755: {id : '9136', geocod : '2310209', nome : "Taiba"},  
1756: {id : '763', geocod : '2313203', nome : "Tamboril"},  
1757: {id : '9146', geocod : '2308005', nome : "Tangente"},  
1758: {id : '9148', geocod : '2307700', nome : "Tanques"},  
1759: {id : '9150', geocod : '2301000', nome : "Tapera"},  
1760: {id : '9152', geocod : '2312908', nome : "Taperuaba"},  
1761: {id : '9160', geocod : '2311306', nome : "Tapuiará"},  
1762: {id : '9170', geocod : '2302800', nome : "Targinos"},  
1763: {id : '649', geocod : '2313252', nome : "Tarrafas"},  
1764: {id : '746', geocod : '2313302', nome : "Tauá"},  
1765: {id : '337', geocod : '2313351', nome : "Tejuçuoca"},  
1766: {id : '289', geocod : '2313401', nome : "Tianguá"},  
1767: {id : '9202', geocod : '2303956', nome : "Timbaúba dos Marinheiros"},  
1768: {id : '9206', geocod : '2304707', nome : "Timonha"},  
1769: {id : '9208', geocod : '2301703', nome : "Tipi"},  
1770: {id : '9217', geocod : '2307601', nome : "Tomé"},  
1771: {id : '347', geocod : '2313500', nome : "Trairi"},  
1772: {id : '9228', geocod : '2305605', nome : "Tranqueiras"},  
1773: {id : '9230', geocod : '2304350', nome : "Trapiá - Forquilha"},  
1774: {id : '9231', geocod : '2312205', nome : "Trapiá - Santa Quitéria"},  
1775: {id : '9240', geocod : '2303956', nome : "Triângulo"},  
1776: {id : '9235', geocod : '2313302', nome : "Trici"},  
1777: {id : '9255', geocod : '2310506', nome : "Tróia"},  
1778: {id : '9244', geocod : '2300309', nome : "Trussu"},  
1779: {id : '9256', geocod : '2303709', nome : "Tucunduba"},  
1780: {id : '9257', geocod : '2304103', nome : "Tucuns"},  
1781: {id : '9274', geocod : '2308005', nome : "Tuína"},  
1782: {id : '343', geocod : '2313559', nome : "Tururu"},  
1783: {id : '301', geocod : '2313609', nome : "Ubajara"},  
1784: {id : '9277', geocod : '2304004', nome : "Ubaúna"},  
1785: {id : '9280', geocod : '2302800', nome : "Ubiraçu"},  
1786: {id : '9281', geocod : '2306801', nome : "Uiraponga"},  
1787: {id : '823', geocod : '2313708', nome : "Umari"},  
1788: {id : '9283', geocod : '2312403', nome : "Umarituba"},  
1789: {id : '9284', geocod : '2307700', nome : "Umarizeiras"},  
1790: {id : '9287', geocod : '2308104', nome : "Umburanas"},  
1791: {id : '339', geocod : '2313757', nome : "Umirim"},  
1792: {id : '341', geocod : '2313807', nome : "Uruburetama"},  
1793: {id : '319', geocod : '2313906', nome : "Uruoca"},  
1794: {id : '9303', geocod : '2311405', nome : "Uruquê"},  
1795: {id : '941', geocod : '2313955', nome : "Varjota"},  
1796: {id : '787', geocod : '2314003', nome : "Várzea Alegre"},  
1797: {id : '9442', geocod : '2303808', nome : "Várzea da Conceição"},  
1798: {id : '9443', geocod : '2308807', nome : "Várzea da Volta"},  
1799: {id : '9446', geocod : '2305803', nome : "Várzea do Giló"},  
1800: {id : '9447', geocod : '2305001', nome : "Várzea dos Espinhos"},  
1801: {id : '9342', geocod : '2301208', nome : "Vazantes"},  
1802: {id : '9351', geocod : '2300507', nome : "Ventura"},  
1803: {id : '9360', geocod : '2307700', nome : "Vertentes do Lagedo"},  
1804: {id : '9431', geocod : '2304459', nome : "Viçosa - Fortin"},  
1805: {id : '9432', geocod : '2305332', nome : "Viçosa - Ibicuitinga"},  
1806: {id : '290', geocod : '2314102', nome : "Viçosa do Ceará"},  
1807: {id : '9364', geocod : '2313203', nome : "Video"},  
1808: {id : '9401', geocod : '2300903', nome : "Vila Soares"}  
1809: ],  
1810: DF : [  
1811: // UF = DF  
1812: {id : '9486', geocod : '5300108', nome : "Alexandre Gusmão"},  
1813: {id : '9493', geocod : '5300108', nome : "Brasilândia"},  
1814: {id : '5478', geocod : '5300108', nome : "Brasília"},  
1815: {id : '9485', geocod : '5300108', nome : "Ceilândia"},  
1816: {id : '9491', geocod : '5300108', nome : "Cruzeiro"},  
1817: {id : '9484', geocod : '5300108', nome : "Gama"},  
1818: {id : '9490', geocod : '5300108', nome : "Guará"},  
1819: {id : '9488', geocod : '5300108', nome : "Núcleo Bandeirante"},  
1820: {id : '9492', geocod : '5300108', nome : "Paranoá"},  
1821: {id : '9496', geocod : '5300108', nome : "Planaltina"},  
1822: {id : '9487', geocod : '5300108', nome : "Samambaia"},  
1823: {id : '9494', geocod : '5300108', nome : "Sobradinho"},  
1824: {id : '9489', geocod : '5300108', nome : "Taguatinga"},  
1825: {id : '9495', geocod : '5300108', nome : "Vale do Amanhecer"}  
1826: ],  
1827: ES : [  
1828: // UF = ES  
1829: {id : '5576', geocod : '3203130', nome : "Acioli"},  
1830: {id : '4440', geocod : '3200102', nome : "Afonso Cláudio"},  
1831: {id : '5586', geocod : '3204203', nome : "Aghá"},  
1832: {id : '3020', geocod : '3200169', nome : "Água Doce do Norte"},  
1833: {id : '3011', geocod : '3200136', nome : "Águia Branca"},  
1834: {id : '5595', geocod : '3204807', nome : "Airituba"},  
1835: {id : '4391', geocod : '3200201', nome : "Alegre"},  
1836: {id : '4444', geocod : '3200300', nome : "Alfredo Chaves"},  
1837: {id : '5636', geocod : '3204807', nome : "Alto Calçado"},  
1838: {id : '5635', geocod : '3204609', nome : "Alto Caldeirão"},  
1839: {id : '5643', geocod : '3200805', nome : "Alto Mutum Preto"},  
1840: {id : '5645', geocod : '3200409', nome : "Alto Pongal"},  
1841: {id : '3000', geocod : '3200359', nome : "Alto Rio Novo"},  
1842: {id : '5650', geocod : '3204609', nome : "Alto Santa Maria"},  
1843: {id : '4443', geocod : '3200409', nome : "Anchieta"},  
1844: {id : '9482', geocod : '3201506', nome : "Ângelo Frechiani"},  
1845: {id : '5715', geocod : '3200201', nome : "Anutiba"},  
1846: {id : '4414', geocod : '3200508', nome : "Apiacá"},  
1847: {id : '5760', geocod : '3205101', nome : "Araçatiba"},  
1848: {id : '5737', geocod : '3201902', nome : "Aracê"},  
1849: {id : '2960', geocod : '3200607', nome : "Aracruz"},  
1850: {id : '5736', geocod : '3201407', nome : "Aracuí"},  
1851: {id : '5738', geocod : '3203346', nome : "Araguaia"},  
1852: {id : '5753', geocod : '3200201', nome : "Araraí"},  
1853: {id : '5772', geocod : '3205200', nome : "Argolas"},  
1854: {id : '4431', geocod : '3200706', nome : "Atilio Vivacqua"},  
1855: {id : '2994', geocod : '3200805', nome : "Baixo Guandu"},  
1856: {id : '3018', geocod : '3200904', nome : "Barra de São Francisco"},  
1857: {id : '5885', geocod : '3204906', nome : "Barra Nova"},  
1858: {id : '5887', geocod : '3203056', nome : "Barra Seca"},  
1859: {id : '5954', geocod : '3201506', nome : "Baunilha"},  
1860: {id : '5963', geocod : '3203205', nome : "Bebedouro"},  
1861: {id : '3041', geocod : '3201001', nome : "Boa Esperança"},  
1862: {id : '6036', geocod : '3201506', nome : "Boapaba"},  
1863: {id : '4413', geocod : '3201100', nome : "Bom Jesus do Norte"},  
1864: {id : '6106', geocod : '3201605', nome : "Braço do Rio"},  
1865: {id : '4438', geocod : '3201159', nome : "Brejetuba"},  
1866: {id : '6134', geocod : '3201209', nome : "Burarama"},  
1867: {id : '6176', geocod : '3200904', nome : "Cachoeirinha de Itaúnas"},  
1868: {id : '4434', geocod : '3201209', nome : "Cachoeiro de Itapemirim"},  
1869: {id : '6186', geocod : '3200201', nome : "Café"},  
1870: {id : '6211', geocod : '3204807', nome : "Calheiros"},  
1871: {id : '6214', geocod : '3205002', nome : "Calogi"},  
1872: {id : '6218', geocod : '3203809', nome : "Camará"},  
1873: {id : '6338', geocod : '3205002', nome : "Carapina"},  
1874: {id : '4450', geocod : '3201308', nome : "Cariacica"},  
1875: {id : '4433', geocod : '3201407', nome : "Castelo"},  
1876: {id : '6417', geocod : '3200201', nome : "Celina"},  
1877: {id : '3008', geocod : '3201506', nome : "Colatina"},  
1878: {id : '3059', geocod : '3201605', nome : "Conceição da Barra"},  
1879: {id : '4437', geocod : '3201704', nome : "Conceição do Castelo"},  
1880: {id : '6510', geocod : '3203403', nome : "Conceição do Muqui"},  
1881: {id : '6516', geocod : '3201209', nome : "Conduru"},  
1882: {id : '6558', geocod : '3202108', nome : "Cotaxé"},  
1883: {id : '6574', geocod : '3204054', nome : "Cristal do Norte"},  
1884: {id : '6588', geocod : '3200300', nome : "Crubixá"},  
1885: {id : '6658', geocod : '3203205', nome : "Desengano"},  
1886: {id : '4427', geocod : '3201803', nome : "Divino de Lourenço"},  
1887: {id : '6670', geocod : '3204500', nome : "Djalma Coutinho"},  
1888: {id : '4447', geocod : '3201902', nome : "Domingos Martins"},  
1889: {id : '6685', geocod : '3203403', nome : "Dona América"},  
1890: {id : '4425', geocod : '3202009', nome : "Dores do Rio Preto"},  
1891: {id : '6706', geocod : '3202603', nome : "Duas Barras"},  
1892: {id : '3036', geocod : '3202108', nome : "Ecoporanga"},  
1893: {id : '6778', geocod : '3201407', nome : "Estrela do Norte"},  
1894: {id : '6793', geocod : '3204708', nome : "Fartura"},  
1895: {id : '6799', geocod : '3200102', nome : "Fazenda Gandu"},  
1896: {id : '3002', geocod : '3202207', nome : "Fundão"},  
1897: {id : '6896', geocod : '3204559', nome : "Garrafão"},  
1898: {id : '6918', geocod : '3205309', nome : "Goiabeiras"},  
1899: {id : '6927', geocod : '3200169', nome : "Governador Lacerda de Aguiar"},  
1900: {id : '5550', geocod : '3202256', nome : "Governador Lindenberg"},  
1901: {id : '6936', geocod : '3201506', nome : "Graça Aranha"},  
1902: {id : '4390', geocod : '3202306', nome : "Guaçuí"},  
1903: {id : '6965', geocod : '3200607', nome : "Guaraná"},  
1904: {id : '4445', geocod : '3202405', nome : "Guarapari"},  
1905: {id : '6968', geocod : '3203908', nome : "Guararema"},  
1906: {id : '4401', geocod : '3202454', nome : "Ibatiba"},  
1907: {id : '7021', geocod : '3205200', nome : "Ibes"},  
1908: {id : '7027', geocod : '3200102', nome : "Ibicaba"},  
1909: {id : '3006', geocod : '3202504', nome : "Ibiraçu"},  
1910: {id : '4392', geocod : '3202553', nome : "Ibitirama"},  
1911: {id : '7047', geocod : '3200300', nome : "Ibitiruí"},  
1912: {id : '7050', geocod : '3200805', nome : "Ibituba"},  
1913: {id : '4441', geocod : '3202603', nome : "Iconha"},  
1914: {id : '7094', geocod : '3202108', nome : "Imburana"},  
1915: {id : '7148', geocod : '3202207', nome : "Irundi"},  
1916: {id : '4399', geocod : '3202652', nome : "Irupi"},  
1917: {id : '7151', geocod : '3201902', nome : "Isabel"},  
1918: {id : '7153', geocod : '3203601', nome : "Itabaiana"},  
1919: {id : '7229', geocod : '3202702', nome : "Itaçu"},  
1920: {id : '3004', geocod : '3202702', nome : "Itaguaçu"},  
1921: {id : '7167', geocod : '3203700', nome : "Itaici"},  
1922: {id : '7171', geocod : '3202702', nome : "Itaimbé"},  
1923: {id : '7173', geocod : '3202801', nome : "Itaipava"},  
1924: {id : '7191', geocod : '3204252', nome : "Itamira"},  
1925: {id : '7199', geocod : '3201209', nome : "Itaoca"},  
1926: {id : '7203', geocod : '3202801', nome : "Itapecoá"},  
1927: {id : '4421', geocod : '3202801', nome : "Itapemirim"},  
1928: {id : '7207', geocod : '3200904', nome : "Itaperuna"},  
1929: {id : '7209', geocod : '3201506', nome : "Itapina"},  
1930: {id : '7219', geocod : '3201308', nome : "Itaquari"},  
1931: {id : '3003', geocod : '3202900', nome : "Itarana"},  
1932: {id : '7228', geocod : '3201605', nome : "Itaúnas"},  
1933: {id : '7227', geocod : '3204906', nome : "Itauninhas"},  
1934: {id : '4400', geocod : '3203007', nome : "Iúna"},  
1935: {id : '7246', geocod : '3200409', nome : "Jabaquara"},  
1936: {id : '7259', geocod : '3205036', nome : "Jaciguá"},  
1937: {id : '7267', geocod : '3200607', nome : "Jacupemba"},  
1938: {id : '2964', geocod : '3203056', nome : "Jaguaré"},  
1939: {id : '4430', geocod : '3203106', nome : "Jerônimo Monteiro"},  
1940: {id : '7321', geocod : '3202108', nome : "Joaçuba"},  
1941: {id : '3007', geocod : '3203130', nome : "João Neiva"},  
1942: {id : '7320', geocod : '3203163', nome : "Joatuba"},  
1943: {id : '7327', geocod : '3200508', nome : "José Carlos"},  
1944: {id : '7347', geocod : '3205200', nome : "Jucu"},  
1945: {id : '7366', geocod : '3205176', nome : "Juruma"},  
1946: {id : '7432', geocod : '3204005', nome : "Lajinha"},  
1947: {id : '2992', geocod : '3203163', nome : "Laranja da Terra"},  
1948: {id : '2962', geocod : '3203205', nome : "Linhares"},  
1949: {id : '7520', geocod : '3204500', nome : "Mangaraí"},  
1950: {id : '3016', geocod : '3203304', nome : "Mantenópolis"},  
1951: {id : '4420', geocod : '3203320', nome : "Marataízes"},  
1952: {id : '4446', geocod : '3203346', nome : "Marechal Floriano"},  
1953: {id : '3009', geocod : '3203353', nome : "Marilândia"},  
1954: {id : '7603', geocod : '3200300', nome : "Matilde"},  
1955: {id : '7614', geocod : '3201902', nome : "Melgaço"},  
1956: {id : '7619', geocod : '3203700', nome : "Menino Jesus"},  
1957: {id : '4418', geocod : '3203403', nome : "Mimoso do Sul"},  
1958: {id : '3044', geocod : '3203502', nome : "Montanha"},  
1959: {id : '7686', geocod : '3200359', nome : "Monte Carmelo do Rio Novo"},  
1960: {id : '7696', geocod : '3201407', nome : "Monte Pio"},  
1961: {id : '7701', geocod : '3200904', nome : "Monte Sinai"},  
1962: {id : '3040', geocod : '3203601', nome : "Mucurici"},  
1963: {id : '7744', geocod : '3202009', nome : "Mundo Novo"},  
1964: {id : '4432', geocod : '3203700', nome : "Muniz Freire"},  
1965: {id : '4429', geocod : '3203809', nome : "Muqui"},  
1966: {id : '7775', geocod : '3204906', nome : "Nestor Gomes"},  
1967: {id : '7792', geocod : '3203056', nome : "Nossa Senhora de Fátima"},  
1968: {id : '7803', geocod : '3205002', nome : "Nova Almeida"},  
1969: {id : '3037', geocod : '3203908', nome : "Nova Venécia"},  
1970: {id : '7845', geocod : '3204906', nome : "Nova Verona"},  
1971: {id : '7850', geocod : '3202256', nome : "Novo Brasil"},  
1972: {id : '7908', geocod : '3201209', nome : "Pacotuba"},  
1973: {id : '7950', geocod : '3200359', nome : "Palmerino"},  
1974: {id : '3010', geocod : '3204005', nome : "Pancas"},  
1975: {id : '7967', geocod : '3201902', nome : "Paraju"},  
1976: {id : '8037', geocod : '3200904', nome : "Paulista"},  
1977: {id : '3046', geocod : '3204054', nome : "Pedro Canário"},  
1978: {id : '8071', geocod : '3202504', nome : "Pendanga"},  
1979: {id : '8080', geocod : '3203007', nome : "Pequiá"},  
1980: {id : '8082', geocod : '3203007', nome : "Perdição"},  
1981: {id : '8097', geocod : '3203700', nome : "Piaçu"},  
1982: {id : '3042', geocod : '3204104', nome : "Pinheiros"},  
1983: {id : '8146', geocod : '3200102', nome : "Piracema"},  
1984: {id : '4442', geocod : '3204203', nome : "Piúma"},  
1985: {id : '8203', geocod : '3203403', nome : "Ponte de Itabapoana"},  
1986: {id : '3039', geocod : '3204252', nome : "Ponto Belo"},  
1987: {id : '8210', geocod : '3200102', nome : "Pontões"},  
1988: {id : '8211', geocod : '3200904', nome : "Poranga"},  
1989: {id : '8267', geocod : '3202207', nome : "Praia Grande"},  
1990: {id : '4419', geocod : '3204302', nome : "Presidente Kennedy"},  
1991: {id : '8285', geocod : '3204401', nome : "Princesa"},  
1992: {id : '8314', geocod : '3205002', nome : "Queimado"},  
1993: {id : '8320', geocod : '3200805', nome : "Quilômetro 14 do Mutum"},  
1994: {id : '8348', geocod : '3203205', nome : "Regência"},  
1995: {id : '8359', geocod : '3200607', nome : "Riacho"},  
1996: {id : '8384', geocod : '3203346', nome : "Ribeirão do Cristo"},  
1997: {id : '2961', geocod : '3204351', nome : "Rio Bananal"},  
1998: {id : '8408', geocod : '3202405', nome : "Rio Calçado"},  
1999: {id : '8413', geocod : '3202801', nome : "Rio Muqui"},  
2000: {id : '4435', geocod : '3204401', nome : "Rio Novo do Sul"},  
2001: {id : '8418', geocod : '3203908', nome : "Rio Preto"},  
2002: {id : '8439', geocod : '3200201', nome : "Rive"},  
2003: {id : '8470', geocod : '3200300', nome : "Sagrada Família"},  
2004: {id : '8497', geocod : '3200201', nome : "Santa Angélica"},  
2005: {id : '8504', geocod : '3200607', nome : "Santa Cruz"},  
2006: {id : '8510', geocod : '3202652', nome : "Santa Cruz de Irupi"},  
2007: {id : '8539', geocod : '3204955', nome : "Santa Júlia"},  
2008: {id : '4451', geocod : '3204500', nome : "Santa Leopoldina"},  
2009: {id : '8547', geocod : '3203304', nome : "Santa Luzia de Mantenópolis"},  
2010: {id : '8548', geocod : '3200169', nome : "Santa Luzia do Azul"},  
2011: {id : '8550', geocod : '3202108', nome : "Santa Luzia do Norte"},  
2012: {id : '4448', geocod : '3204559', nome : "Santa Maria de Jetibá"},  
2013: {id : '8562', geocod : '3203346', nome : "Santa Maria de Marechal"},  
2014: {id : '8566', geocod : '3202553', nome : "Santa Marta"},  
2015: {id : '3001', geocod : '3204609', nome : "Santa Teresa"},  
2016: {id : '8595', geocod : '3202108', nome : "Santa Terezinha"},  
2017: {id : '8685', geocod : '3203007', nome : "Santíssima Trindade"},  
2018: {id : '8622', geocod : '3200169', nome : "Santo Agostinho"},  
2019: {id : '8629', geocod : '3201001', nome : "Santo Antônio"},  
2020: {id : '8652', geocod : '3204609', nome : "Santo Antônio do Canaã"},  
2021: {id : '8659', geocod : '3203403', nome : "Santo Antônio do Muqui"},  
2022: {id : '8665', geocod : '3201001', nome : "Santo Antônio do Pouso Alegr"},  
2023: {id : '8666', geocod : '3203908', nome : "Santo Antônio do Quinze"},  
2024: {id : '8667', geocod : '3200805', nome : "Santo Antônio do Rio Doce"},  
2025: {id : '3012', geocod : '3204658', nome : "São Domingos do Norte"},  
2026: {id : '8848', geocod : '3200102', nome : "São Francisco Xavier"},  
2027: {id : '2958', geocod : '3204708', nome : "São Gabriel da Palha"},  
2028: {id : '8860', geocod : '3203304', nome : "São Geraldo"},  
2029: {id : '8870', geocod : '3204955', nome : "São Jacinto"},  
2030: {id : '8958', geocod : '3204609', nome : "São João de Petrópolis"},  
2031: {id : '8959', geocod : '3205069', nome : "São João de Viçosa"},  
2032: {id : '8968', geocod : '3203007', nome : "São João do Príncipe"},  
2033: {id : '8970', geocod : '3204104', nome : "São João do Sobrado"},  
2034: {id : '8883', geocod : '3205176', nome : "São Jorge da Barra Seca"},  
2035: {id : '8884', geocod : '3201159', nome : "São Jorge de Oliveira"},  
2036: {id : '8885', geocod : '3204351', nome : "São Jorge de Tiradentes"},  
2037: {id : '8905', geocod : '3203403', nome : "São José das Torres"},  
2038: {id : '8907', geocod : '3203304', nome : "São José de Mantenópolis"},  
2039: {id : '4415', geocod : '3204807', nome : "São José do Calçado"},  
2040: {id : '8932', geocod : '3201001', nome : "São José do Sobradinho"},  
2041: {id : '3043', geocod : '3204906', nome : "São Mateus"},  
2042: {id : '9027', geocod : '3203403', nome : "São Pedro de Itabapoana"},  
2043: {id : '9028', geocod : '3202306', nome : "São Pedro de Rates"},  
2044: {id : '9037', geocod : '3203205', nome : "São Rafael"},  
2045: {id : '3005', geocod : '3204955', nome : "São Roque do Canaã"},  
2046: {id : '9095', geocod : '3202306', nome : "São Tiago"},  
2047: {id : '9097', geocod : '3205200', nome : "São Torquato"},  
2048: {id : '8691', geocod : '3203353', nome : "Sapucaia"},  
2049: {id : '4453', geocod : '3205002', nome : "Serra"},  
2050: {id : '8735', geocod : '3200102', nome : "Serra Pelada"},  
2051: {id : '8780', geocod : '3203163', nome : "Sobreiro"},  
2052: {id : '2963', geocod : '3205010', nome : "Sooretama"},  
2053: {id : '9204', geocod : '3202207', nome : "Timbuí"},  
2054: {id : '9215', geocod : '3202405', nome : "Todos os Santos"},  
2055: {id : '9307', geocod : '3200300', nome : "Urânia"},  
2056: {id : '4436', geocod : '3205036', nome : "Vargem Alta"},  
2057: {id : '9331', geocod : '3201209', nome : "Vargem Grande do Soturno"},  
2058: {id : '4439', geocod : '3205069', nome : "Venda Nova do Imigrante"},  
2059: {id : '4449', geocod : '3205101', nome : "Viana"},  
2060: {id : '9366', geocod : '3203700', nome : "Vieira Machado"},  
2061: {id : '9380', geocod : '3200169', nome : "Vila Nelita"},  
2062: {id : '9386', geocod : '3200805', nome : "Vila Nova de Bananal"},  
2063: {id : '3038', geocod : '3205150', nome : "Vila Pavão"},  
2064: {id : '2959', geocod : '3205176', nome : "Vila Valério"},  
2065: {id : '4452', geocod : '3205200', nome : "Vila Velha"},  
2066: {id : '9408', geocod : '3204005', nome : "Vila Verde"},  
2067: {id : '9415', geocod : '3203502', nome : "Vinhático"},  
2068: {id : '9416', geocod : '3204609', nome : "Vinte e Cinco de Julho"},  
2069: {id : '5473', geocod : '3205309', nome : "Vitória"}  
2070: ],  
2071: GO : [  
2072: // UF = GO  
2073: {id : '2626', geocod : '5200050', nome : "Abadia de Goiás"},  
2074: {id : '2650', geocod : '5200100', nome : "Abadiânia"},  
2075: {id : '2515', geocod : '5200134', nome : "Acreúna"},  
2076: {id : '2615', geocod : '5200159', nome : "Adelândia"},  
2077: {id : '2127', geocod : '5200175', nome : "Água Fria de Goiás"},  
2078: {id : '2595', geocod : '5200209', nome : "Água Limpa"},  
2079: {id : '2066', geocod : '5200258', nome : "Águas Lindas de Goiás"},  
2080: {id : '2651', geocod : '5200308', nome : "Alexânia"},  
2081: {id : '2581', geocod : '5200506', nome : "Aloândia"},  
2082: {id : '5632', geocod : '5215306', nome : "Alto Alvorada"},  
2083: {id : '2080', geocod : '5200555', nome : "Alto Horizonte"},  
2084: {id : '2136', geocod : '5200605', nome : "Alto Paraíso de Goiás"},  
2085: {id : '2106', geocod : '5200803', nome : "Alvorada do Norte"},  
2086: {id : '2081', geocod : '5200829', nome : "Amaralina"},  
2087: {id : '2620', geocod : '5200852', nome : "Americano do Brasil"},  
2088: {id : '2533', geocod : '5200902', nome : "Amorinópolis"},  
2089: {id : '2645', geocod : '5201108', nome : "Anápolis"},  
2090: {id : '2608', geocod : '5201207', nome : "Anhanguera"},  
2091: {id : '2617', geocod : '5201306', nome : "Anicuns"},  
2092: {id : '2605', geocod : '5201405', nome : "Aparecida de Goiânia"},  
2093: {id : '5718', geocod : '5210901', nome : "Aparecida de Goiás"},  
2094: {id : '5724', geocod : '5213707', nome : "Aparecida do Rio Claro"},  
2095: {id : '2508', geocod : '5201454', nome : "Aparecida do Rio Doce"},  
2096: {id : '2490', geocod : '5201504', nome : "Aporé"},  
2097: {id : '2624', geocod : '5201603', nome : "Araçu"},  
2098: {id : '2024', geocod : '5201702', nome : "Aragarças"},  
2099: {id : '2587', geocod : '5201801', nome : "Aragoiânia"},  
2100: {id : '2015', geocod : '5202155', nome : "Araguapaz"},  
2101: {id : '2531', geocod : '5202353', nome : "Arenópolis"},  
2102: {id : '2012', geocod : '5202502', nome : "Aruanã"},  
2103: {id : '2541', geocod : '5202601', nome : "Aurilândia"},  
2104: {id : '2623', geocod : '5202809', nome : "Avelinópolis"},  
2105: {id : '5830', geocod : '5207600', nome : "Bacilândia"},  
2106: {id : '2526', geocod : '5203104', nome : "Baliza"},  
2107: {id : '5863', geocod : '5214838', nome : "Bandeirantes"},  
2108: {id : '2070', geocod : '5203203', nome : "Barro Alto"},  
2109: {id : '2606', geocod : '5203302', nome : "Bela Vista de Goiás"},  
2110: {id : '2528', geocod : '5203401', nome : "Bom Jardim de Goiás"},  
2111: {id : '2569', geocod : '5203500', nome : "Bom Jesus de Goiás"},  
2112: {id : '2644', geocod : '5203559', nome : "Bonfinópolis"},  
2113: {id : '2076', geocod : '5203575', nome : "Bonópolis"},  
2114: {id : '2630', geocod : '5203609', nome : "Brazabrantes"},  
2115: {id : '2011', geocod : '5203807', nome : "Britânia"},  
2116: {id : '6129', geocod : '5208905', nome : "Buenolândia"},  
2117: {id : '2594', geocod : '5203906', nome : "Buriti Alegre"},  
2118: {id : '2520', geocod : '5203939', nome : "Buriti de Goiás"},  
2119: {id : '2108', geocod : '5203962', nome : "Buritinópolis"},  
2120: {id : '2131', geocod : '5204003', nome : "Cabeceiras"},  
2121: {id : '2507', geocod : '5204102', nome : "Cachoeira Alta"},  
2122: {id : '2540', geocod : '5204201', nome : "Cachoeira de Goiás"},  
2123: {id : '2568', geocod : '5204250', nome : "Cachoeira Dourada"},  
2124: {id : '2506', geocod : '5204300', nome : "Caçu"},  
2125: {id : '2504', geocod : '5204409', nome : "Caiapônia"},  
2126: {id : '6197', geocod : '5207535', nome : "Caiçara"},  
2127: {id : '6205', geocod : '5208905', nome : "Calcilândia"},  
2128: {id : '2478', geocod : '5204508', nome : "Caldas Novas"},  
2129: {id : '2643', geocod : '5204557', nome : "Caldazinha"},  
2130: {id : '2621', geocod : '5204607', nome : "Campestre de Goiás"},  
2131: {id : '2093', geocod : '5204656', nome : "Campinaçu"},  
2132: {id : '2086', geocod : '5204706', nome : "Campinorte"},  
2133: {id : '2686', geocod : '5204805', nome : "Campo Alegre de Goiás"},  
2134: {id : '5513', geocod : '5204854', nome : "Campo Limpo de Goiás"},  
2135: {id : '6259', geocod : '5211602', nome : "Campolândia"},  
2136: {id : '2115', geocod : '5204904', nome : "Campos Belos"},  
2137: {id : '2078', geocod : '5204953', nome : "Campos Verdes"},  
2138: {id : '6266', geocod : '5213087', nome : "Cana Brava"},  
2139: {id : '6271', geocod : '5212204', nome : "Canadá"},  
2140: {id : '6304', geocod : '5201306', nome : "Capelinha"},  
2141: {id : '6344', geocod : '5222005', nome : "Caraíba"},  
2142: {id : '2049', geocod : '5205000', nome : "Carmo do Rio Verde"},  
2143: {id : '2564', geocod : '5205059', nome : "Castelândia"},  
2144: {id : '6384', geocod : '5218607', nome : "Castrinópolis"},  
2145: {id : '2678', geocod : '5205109', nome : "Catalão"},  
2146: {id : '2627', geocod : '5205208', nome : "Caturaí"},  
2147: {id : '2137', geocod : '5205307', nome : "Cavalcante"},  
2148: {id : '6407', geocod : '5210109', nome : "Cavalheiro"},  
2149: {id : '2050', geocod : '5205406', nome : "Ceres"},  
2150: {id : '2584', geocod : '5205455', nome : "Cezarina"},  
2151: {id : '2471', geocod : '5205471', nome : "Chapadão do Céu"},  
2152: {id : '6445', geocod : '5201306', nome : "Choupana"},  
2153: {id : '6448', geocod : '5211206', nome : "Cibele"},  
2154: {id : '2921', geocod : '5205497', nome : "Cidade Ocidental"},  
2155: {id : '2064', geocod : '5205513', nome : "Cocalzinho de Goiás"},  
2156: {id : '2099', geocod : '5205521', nome : "Colinas do Sul"},  
2157: {id : '2543', geocod : '5205703', nome : "Córrego do Ouro"},  
2158: {id : '6630', geocod : '5217609', nome : "Córrego Rico"},  
2159: {id : '2063', geocod : '5205802', nome : "Corumbá de Goiás"},  
2160: {id : '2597', geocod : '5205901', nome : "Corumbaíba"},  
2161: {id : '2922', geocod : '5206206', nome : "Cristalina"},  
2162: {id : '2482', geocod : '5206305', nome : "Cristianópolis"},  
2163: {id : '2072', geocod : '5206404', nome : "Crixás"},  
2164: {id : '2586', geocod : '5206503', nome : "Cromínia"},  
2165: {id : '2609', geocod : '5206602', nome : "Cumari"},  
2166: {id : '2109', geocod : '5206701', nome : "Damianópolis"},  
2167: {id : '2632', geocod : '5206800', nome : "Damolândia"},  
2168: {id : '6646', geocod : '5208905', nome : "Davidópolis"},  
2169: {id : '2685', geocod : '5206909', nome : "Davinópolis"},  
2170: {id : '6664', geocod : '5211206', nome : "Diolândia"},  
2171: {id : '2536', geocod : '5207105', nome : "Diorama"},  
2172: {id : '2116', geocod : '5208301', nome : "Divinópolis de Goiás"},  
2173: {id : '6682', geocod : '5210109', nome : "Domiciano Ribeiro"},  
2174: {id : '2527', geocod : '5207253', nome : "Doverlândia"},  
2175: {id : '2580', geocod : '5207352', nome : "Edealina"},  
2176: {id : '2573', geocod : '5207402', nome : "Edéia"},  
2177: {id : '2089', geocod : '5207501', nome : "Estrela do Norte"},  
2178: {id : '2014', geocod : '5207535', nome : "Faina"},  
2179: {id : '2518', geocod : '5207600', nome : "Fazenda Nova"},  
2180: {id : '2613', geocod : '5207808', nome : "Firminópolis"},  
2181: {id : '2104', geocod : '5207907', nome : "Flores de Goiás"},  
2182: {id : '2129', geocod : '5208004', nome : "Formosa"},  
2183: {id : '2091', geocod : '5208103', nome : "Formoso"},  
2184: {id : '6853', geocod : '5220009', nome : "Forte"},  
2185: {id : '5514', geocod : '5208152', nome : "Gameleira de Goiás"},  
2186: {id : '6907', geocod : '5221601', nome : "Geriaçu"},  
2187: {id : '6919', geocod : '5201108', nome : "Goialândia"},  
2188: {id : '2641', geocod : '5208400', nome : "Goianápolis"},  
2189: {id : '2610', geocod : '5208509', nome : "Goiandira"},  
2190: {id : '2067', geocod : '5208608', nome : "Goianésia"},  
2191: {id : '5474', geocod : '5208707', nome : "Goiânia"},  
2192: {id : '2628', geocod : '5208806', nome : "Goianira"},  
2193: {id : '2044', geocod : '5208905', nome : "Goiás"},  
2194: {id : '2571', geocod : '5209101', nome : "Goiatuba"},  
2195: {id : '2561', geocod : '5209150', nome : "Gouvelândia"},  
2196: {id : '2588', geocod : '5209200', nome : "Guapó"},  
2197: {id : '2035', geocod : '5209291', nome : "Guaraíta"},  
2198: {id : '2111', geocod : '5209408', nome : "Guarani de Goiás"},  
2199: {id : '2057', geocod : '5209457', nome : "Guarinos"},  
2200: {id : '2037', geocod : '5209606', nome : "Heitoraí"},  
2201: {id : '2604', geocod : '5209705', nome : "Hidrolândia"},  
2202: {id : '2059', geocod : '5209804', nome : "Hidrolina"},  
2203: {id : '2110', geocod : '5209903', nome : "Iaciara"},  
2204: {id : '2563', geocod : '5209937', nome : "Inaciolândia"},  
2205: {id : '2576', geocod : '5209952', nome : "Indiara"},  
2206: {id : '2629', geocod : '5210000', nome : "Inhumas"},  
2207: {id : '7115', geocod : '5201108', nome : "Interlândia"},  
2208: {id : '2611', geocod : '5210109', nome : "Ipameri"},  
2209: {id : '5515', geocod : '5210158', nome : "Ipiranga de Goiás"},  
2210: {id : '2534', geocod : '5210208', nome : "Iporá"},  
2211: {id : '2535', geocod : '5210307', nome : "Israelândia"},  
2212: {id : '2634', geocod : '5210406', nome : "Itaberaí"},  
2213: {id : '7163', geocod : '5220405', nome : "Itaguaçu"},  
2214: {id : '2045', geocod : '5210562', nome : "Itaguari"},  
2215: {id : '2046', geocod : '5210604', nome : "Itaguaru"},  
2216: {id : '2492', geocod : '5210802', nome : "Itajá"},  
2217: {id : '2056', geocod : '5210901', nome : "Itapaci"},  
2218: {id : '2010', geocod : '5211008', nome : "Itapirapuã"},  
2219: {id : '2036', geocod : '5211206', nome : "Itapuranga"},  
2220: {id : '2505', geocod : '5211305', nome : "Itarumã"},  
2221: {id : '2635', geocod : '5211404', nome : "Itauçu"},  
2222: {id : '2592', geocod : '5211503', nome : "Itumbiara"},  
2223: {id : '2538', geocod : '5211602', nome : "Ivolândia"},  
2224: {id : '7260', geocod : '5211008', nome : "Jacilândia"},  
2225: {id : '2574', geocod : '5211701', nome : "Jandaia"},  
2226: {id : '2034', geocod : '5211800', nome : "Jaraguá"},  
2227: {id : '2502', geocod : '5211909', nome : "Jataí"},  
2228: {id : '2537', geocod : '5212006', nome : "Jaupaci"},  
2229: {id : '7312', geocod : '5207535', nome : "Jeroaquara"},  
2230: {id : '2047', geocod : '5212055', nome : "Jesúpolis"},  
2231: {id : '7318', geocod : '5201108', nome : "Joanápolis"},  
2232: {id : '2579', geocod : '5212105', nome : "Joviânia"},  
2233: {id : '7372', geocod : '5212204', nome : "Juscelândia"},  
2234: {id : '2028', geocod : '5212204', nome : "Jussara"},  
2235: {id : '5516', geocod : '5212253', nome : "Lagoa Santa"},  
2236: {id : '7419', geocod : '5217302', nome : "Lagolândia"},  
2237: {id : '2647', geocod : '5212303', nome : "Leopoldo de Bulhões"},  
2238: {id : '7475', geocod : '5212956', nome : "Lua Nova"},  
2239: {id : '7480', geocod : '5213707', nome : "Lucilândia"},  
2240: {id : '2920', geocod : '5212501', nome : "Luziânia"},  
2241: {id : '2585', geocod : '5212600', nome : "Mairipotaba"},  
2242: {id : '2165', geocod : '5212709', nome : "Mambaí"},  
2243: {id : '2087', geocod : '5212808', nome : "Mara Rosa"},  
2244: {id : '7553', geocod : '5209101', nome : "Marcianópolis"},  
2245: {id : '2598', geocod : '5212907', nome : "Marzagão"},  
2246: {id : '2013', geocod : '5212956', nome : "Matrinchã"},  
2247: {id : '2510', geocod : '5213004', nome : "Maurilândia"},  
2248: {id : '7622', geocod : '5213400', nome : "Messianópolis"},  
2249: {id : '2098', geocod : '5213053', nome : "Mimoso de Goiás"},  
2250: {id : '2100', geocod : '5213087', nome : "Minaçu"},  
2251: {id : '2476', geocod : '5213103', nome : "Mineiros"},  
2252: {id : '2539', geocod : '5213400', nome : "Moiporá"},  
2253: {id : '2113', geocod : '5213509', nome : "Monte Alegre de Goiás"},  
2254: {id : '2532', geocod : '5213707', nome : "Montes Claros de Goiás"},  
2255: {id : '2511', geocod : '5213756', nome : "Montividiu"},  
2256: {id : '2043', geocod : '5213772', nome : "Montividiu do Norte"},  
2257: {id : '2600', geocod : '5213806', nome : "Morrinhos"},  
2258: {id : '2038', geocod : '5213855', nome : "Morro Agudo de Goiás"},  
2259: {id : '2619', geocod : '5213905', nome : "Mossâmedes"},  
2260: {id : '2016', geocod : '5214002', nome : "Mozarlândia"},  
2261: {id : '2075', geocod : '5214051', nome : "Mundo Novo"},  
2262: {id : '2088', geocod : '5214101', nome : "Mutunópolis"},  
2263: {id : '7767', geocod : '5208608', nome : "Natinópolis"},  
2264: {id : '2616', geocod : '5214408', nome : "Nazário"},  
2265: {id : '2640', geocod : '5214507', nome : "Nerópolis"},  
2266: {id : '2092', geocod : '5214606', nome : "Niquelândia"},  
2267: {id : '2040', geocod : '5214705', nome : "Nova América"},  
2268: {id : '2599', geocod : '5214804', nome : "Nova Aurora"},  
2269: {id : '7814', geocod : '5201405', nome : "Nova Brasília"},  
2270: {id : '2074', geocod : '5214838', nome : "Nova Crixás"},  
2271: {id : '2055', geocod : '5214861', nome : "Nova Glória"},  
2272: {id : '2079', geocod : '5214879', nome : "Nova Iguaçu de Goiás"},  
2273: {id : '2105', geocod : '5214903', nome : "Nova Roma"},  
2274: {id : '2633', geocod : '5215009', nome : "Nova Veneza"},  
2275: {id : '2519', geocod : '5215207', nome : "Novo Brasil"},  
2276: {id : '5512', geocod : '5215231', nome : "Novo Gama"},  
2277: {id : '2084', geocod : '5215256', nome : "Novo Planalto"},  
2278: {id : '7870', geocod : '5210802', nome : "Olaria do Angico"},  
2279: {id : '2484', geocod : '5215306', nome : "Orizona"},  
2280: {id : '2646', geocod : '5215405', nome : "Ouro Verde de Goiás"},  
2281: {id : '7900', geocod : '5218805', nome : "Ouroana"},  
2282: {id : '2679', geocod : '5215504', nome : "Ouvidor"},  
2283: {id : '2071', geocod : '5215603', nome : "Padre Bernardo"},  
2284: {id : '2530', geocod : '5215652', nome : "Palestina de Goiás"},  
2285: {id : '2577', geocod : '5215702', nome : "Palmeiras de Goiás"},  
2286: {id : '2480', geocod : '5215801', nome : "Palmelo"},  
2287: {id : '2575', geocod : '5215900', nome : "Palminópolis"},  
2288: {id : '2570', geocod : '5216007', nome : "Panamá"},  
2289: {id : '2499', geocod : '5216304', nome : "Paranaiguara"},  
2290: {id : '2516', geocod : '5216403', nome : "Paraúna"},  
2291: {id : '8033', geocod : '5204706', nome : "Pau Terra"},  
2292: {id : '2503', geocod : '5216452', nome : "Perolândia"},  
2293: {id : '2638', geocod : '5216809', nome : "Petrolina de Goiás"},  
2294: {id : '2058', geocod : '5216908', nome : "Pilar de Goiás"},  
2295: {id : '8105', geocod : '5210307', nome : "Piloândia"},  
2296: {id : '2602', geocod : '5217104', nome : "Piracanjuba"},  
2297: {id : '2529', geocod : '5217203', nome : "Piranhas"},  
2298: {id : '2062', geocod : '5217302', nome : "Pirenópolis"},  
2299: {id : '2481', geocod : '5217401', nome : "Pires do Rio"},  
2300: {id : '2128', geocod : '5217609', nome : "Planaltina"},  
2301: {id : '2582', geocod : '5217708', nome : "Pontalina"},  
2302: {id : '2094', geocod : '5218003', nome : "Porangatu"},  
2303: {id : '2572', geocod : '5218052', nome : "Porteirão"},  
2304: {id : '2477', geocod : '5218102', nome : "Portelândia"},  
2305: {id : '2112', geocod : '5218300', nome : "Posse"},  
2306: {id : '8241', geocod : '5200100', nome : "Posse d'Abadia"},  
2307: {id : '2603', geocod : '5218391', nome : "Prof. Jamil"},  
2308: {id : '2509', geocod : '5218508', nome : "Quirinópolis"},  
2309: {id : '8347', geocod : '5213707', nome : "Registro do Araguaia"},  
2310: {id : '2051', geocod : '5218607', nome : "Rialma"},  
2311: {id : '2052', geocod : '5218706', nome : "Rianápolis"},  
2312: {id : '2601', geocod : '5218789', nome : "Rio Quente"},  
2313: {id : '2512', geocod : '5218805', nome : "Rio Verde"},  
2314: {id : '8440', geocod : '5218805', nome : "Riverlândia"},  
2315: {id : '8447', geocod : '5204854', nome : "Rodrigues Nascimento"},  
2316: {id : '8453', geocod : '5220108', nome : "Rosalândia"},  
2317: {id : '2054', geocod : '5218904', nome : "Rubiataba"},  
2318: {id : '2618', geocod : '5219001', nome : "Sanclerlândia"},  
2319: {id : '2622', geocod : '5219100', nome : "Santa Bárbara de Goiás"},  
2320: {id : '2479', geocod : '5219209', nome : "Santa Cruz de Goiás"},  
2321: {id : '2027', geocod : '5219258', nome : "Santa Fé de Goiás"},  
2322: {id : '2513', geocod : '5219308', nome : "Santa Helena de Goiás"},  
2323: {id : '2053', geocod : '5219357', nome : "Santa Isabel"},  
2324: {id : '2474', geocod : '5219407', nome : "Santa Rita do Araguaia"},  
2325: {id : '2069', geocod : '5219456', nome : "Santa Rita do Novo Destino"},  
2326: {id : '8582', geocod : '5208004', nome : "Santa Rosa"},  
2327: {id : '2637', geocod : '5219506', nome : "Santa Rosa de Goiás"},  
2328: {id : '2090', geocod : '5219605', nome : "Santa Tereza de Goiás"},  
2329: {id : '2077', geocod : '5219704', nome : "Santa Terezinha de Goiás"},  
2330: {id : '2514', geocod : '5219712', nome : "Santo Antônio da Barra"},  
2331: {id : '2631', geocod : '5219738', nome : "Santo Antônio de Goiás"},  
2332: {id : '2065', geocod : '5219753', nome : "Santo Antônio do Descoberto"},  
2333: {id : '8668', geocod : '5205109', nome : "Santo Antônio do Rio"},  
2334: {id : '2117', geocod : '5219803', nome : "São Domingos"},  
2335: {id : '2061', geocod : '5219902', nome : "São Francisco de Goiás"},  
2336: {id : '8857', geocod : '5217609', nome : "São Gabriel de Goiás"},  
2337: {id : '8943', geocod : '5208905', nome : "São João"},  
2338: {id : '2517', geocod : '5220058', nome : "São João da Paraúna"},  
2339: {id : '2130', geocod : '5220009', nome : "São João d'Aliança"},  
2340: {id : '2542', geocod : '5220108', nome : "São Luís de Montes Belos"},  
2341: {id : '2060', geocod : '5220157', nome : "São Luíz do Norte"},  
2342: {id : '8985', geocod : '5214606', nome : "São Luiz do Tocantin"},  
2343: {id : '2082', geocod : '5220207', nome : "São Miguel do Araguaia"},  
2344: {id : '2483', geocod : '5220264', nome : "São Miguel do Passa Quatro"},  
2345: {id : '2039', geocod : '5220280', nome : "São Patrício"},  
2346: {id : '9082', geocod : '5212204', nome : "São Sebastião do Rio"},  
2347: {id : '2500', geocod : '5220405', nome : "São Simão"},  
2348: {id : '2639', geocod : '5220454', nome : "Senador Canedo"},  
2349: {id : '8731', geocod : '5207600', nome : "Serra Dourada"},  
2350: {id : '2501', geocod : '5220504', nome : "Serranópolis"},  
2351: {id : '2648', geocod : '5220603', nome : "Silvânia"},  
2352: {id : '2107', geocod : '5220686', nome : "Simolândia"},  
2353: {id : '2135', geocod : '5220702', nome : "Sítio d'Abadia"},  
2354: {id : '8792', geocod : '5201108', nome : "Sousânia"},  
2355: {id : '2636', geocod : '5221007', nome : "Taquaral de Goiás"},  
2356: {id : '9181', geocod : '5214606', nome : "Taveira"},  
2357: {id : '2138', geocod : '5221080', nome : "Teresina de Goiás"},  
2358: {id : '2642', geocod : '5221197', nome : "Terezópolis de Goiás"},  
2359: {id : '9189', geocod : '5212253', nome : "Termas de Itajá"},  
2360: {id : '2677', geocod : '5221304', nome : "Três Ranchos"},  
2361: {id : '2625', geocod : '5221403', nome : "Trindade"},  
2362: {id : '2042', geocod : '5221452', nome : "Trombas"},  
2363: {id : '9269', geocod : '5214606', nome : "Tupiraçaba"},  
2364: {id : '2614', geocod : '5221502', nome : "Turvânia"},  
2365: {id : '2565', geocod : '5221551', nome : "Turvelândia"},  
2366: {id : '2073', geocod : '5221577', nome : "Uirapuru"},  
2367: {id : '2085', geocod : '5221601', nome : "Uruaçu"},  
2368: {id : '2048', geocod : '5221700', nome : "Uruana"},  
2369: {id : '9298', geocod : '5221700', nome : "Uruceres"},  
2370: {id : '9306', geocod : '5221700', nome : "Uruíta"},  
2371: {id : '2612', geocod : '5221809', nome : "Urutaí"},  
2372: {id : '9309', geocod : '5208905', nome : "Uvá"},  
2373: {id : '9313', geocod : '5218904', nome : "Valdelândia"},  
2374: {id : '2919', geocod : '5221858', nome : "Valparaíso de Goiás"},  
2375: {id : '2583', geocod : '5221908', nome : "Varjão"},  
2376: {id : '2649', geocod : '5222005', nome : "Vianópolis"},  
2377: {id : '2578', geocod : '5222054', nome : "Vicentinópolis"},  
2378: {id : '2132', geocod : '5222203', nome : "Vila Boa"},  
2379: {id : '2068', geocod : '5222302', nome : "Vila Propício"}  
2380: ],  
2381: MA : [  
2382: // UF = MA  
2383: {id : '513', geocod : '2100055', nome : "Açailândia"},  
2384: {id : '637', geocod : '2100105', nome : "Afonso Cunha"},  
2385: {id : '148', geocod : '2100154', nome : "Água Doce do Maranhão"},  
2386: {id : '253', geocod : '2100204', nome : "Alcântara"},  
2387: {id : '599', geocod : '2100303', nome : "Aldeias Altas"},  
2388: {id : '632', geocod : '2100402', nome : "Altamira do Maranhão"},  
2389: {id : '546', geocod : '2100436', nome : "Alto Alegre do Maranhão"},  
2390: {id : '197', geocod : '2100477', nome : "Alto Alegre do Pindaré"},  
2391: {id : '1493', geocod : '2100501', nome : "Alto Parnaíba"},  
2392: {id : '193', geocod : '2100550', nome : "Amapá do Maranhão"},  
2393: {id : '521', geocod : '2100600', nome : "Amarante do Maranhão"},  
2394: {id : '228', geocod : '2100709', nome : "Anajatuba"},  
2395: {id : '272', geocod : '2100808', nome : "Anapurus"},  
2396: {id : '262', geocod : '2100832', nome : "Apicum-Açu"},  
2397: {id : '209', geocod : '2100873', nome : "Araguanã"},  
2398: {id : '291', geocod : '2100907', nome : "Araioses"},  
2399: {id : '543', geocod : '2100956', nome : "Arame"},  
2400: {id : '226', geocod : '2101004', nome : "Arari"},  
2401: {id : '5812', geocod : '2104305', nome : "Aurizona"},  
2402: {id : '251', geocod : '2101103', nome : "Axixá"},  
2403: {id : '459', geocod : '2101202', nome : "Bacabal"},  
2404: {id : '247', geocod : '2101251', nome : "Bacabeira"},  
2405: {id : '5828', geocod : '2102309', nome : "Bacatuba"},  
2406: {id : '261', geocod : '2101301', nome : "Bacuri"},  
2407: {id : '239', geocod : '2101350', nome : "Bacurituba"},  
2408: {id : '525', geocod : '2101400', nome : "Balsas"},  
2409: {id : '562', geocod : '2101509', nome : "Barão de Grajaú"},  
2410: {id : '5941', geocod : '2102606', nome : "Barão de Tromaí"},  
2411: {id : '542', geocod : '2101608', nome : "Barra do Corda"},  
2412: {id : '144', geocod : '2101707', nome : "Barreirinhas"},  
2413: {id : '5927', geocod : '2112506', nome : "Barro Duro"},  
2414: {id : '201', geocod : '2101772', nome : "Bela Vista do Maranhão"},  
2415: {id : '273', geocod : '2101731', nome : "Belágua"},  
2416: {id : '534', geocod : '2101806', nome : "Benedito Leite"},  
2417: {id : '243', geocod : '2101905', nome : "Bequimão"},  
2418: {id : '456', geocod : '2101939', nome : "Bernardo do Mearim"},  
2419: {id : '184', geocod : '2101970', nome : "Boa Vista do Gurupi"},  
2420: {id : '6033', geocod : '2112902', nome : "Boa Vista do Pindaré"},  
2421: {id : '203', geocod : '2102002', nome : "Bom Jardim"},  
2422: {id : '523', geocod : '2102036', nome : "Bom Jesus das Selvas"},  
2423: {id : '454', geocod : '2102077', nome : "Bom Lugar"},  
2424: {id : '6079', geocod : '2101004', nome : "Bonfim do Arari"},  
2425: {id : '281', geocod : '2102101', nome : "Brejo"},  
2426: {id : '631', geocod : '2102150', nome : "Brejo de Areia"},  
2427: {id : '6117', geocod : '2107803', nome : "Brejo de São Félix"},  
2428: {id : '280', geocod : '2102200', nome : "Buriti"},  
2429: {id : '589', geocod : '2102309', nome : "Buriti Bravo"},  
2430: {id : '6139', geocod : '2112209', nome : "Buriti Cortado"},  
2431: {id : '524', geocod : '2102325', nome : "Buriticupu"},  
2432: {id : '520', geocod : '2102358', nome : "Buritirana"},  
2433: {id : '250', geocod : '2102374', nome : "Cachoeira Grande"},  
2434: {id : '240', geocod : '2102408', nome : "Cajapió"},  
2435: {id : '224', geocod : '2102507', nome : "Cajari"},  
2436: {id : '503', geocod : '2102556', nome : "Campestre do Maranhão"},  
2437: {id : '257', geocod : '2102606', nome : "Cândido Mendes"},  
2438: {id : '231', geocod : '2102705', nome : "Cantanhede"},  
2439: {id : '463', geocod : '2102754', nome : "Capinzal do Norte"},  
2440: {id : '6345', geocod : '2110906', nome : "Caraíba do Norte"},  
2441: {id : '480', geocod : '2102804', nome : "Carolina"},  
2442: {id : '194', geocod : '2102903', nome : "Carutapera"},  
2443: {id : '598', geocod : '2103000', nome : "Caxias"},  
2444: {id : '266', geocod : '2103109', nome : "Cedral"},  
2445: {id : '244', geocod : '2103125', nome : "Central do Maranhão"},  
2446: {id : '211', geocod : '2103158', nome : "Centro do Guilherme"},  
2447: {id : '212', geocod : '2103174', nome : "Centro Novo do Maranhão"},  
2448: {id : '271', geocod : '2103208', nome : "Chapadinha"},  
2449: {id : '512', geocod : '2103257', nome : "Cidelândia"},  
2450: {id : '597', geocod : '2103307', nome : "Codó"},  
2451: {id : '6473', geocod : '2103307', nome : "Codozinho"},  
2452: {id : '600', geocod : '2103406', nome : "Coelho Neto"},  
2453: {id : '442', geocod : '2103505', nome : "Colinas"},  
2454: {id : '221', geocod : '2103554', nome : "Conceição do Lago-Açu"},  
2455: {id : '636', geocod : '2103604', nome : "Coroatá"},  
2456: {id : '6620', geocod : '2100501', nome : "Curupá"},  
2457: {id : '263', geocod : '2103703', nome : "Cururupu"},  
2458: {id : '6623', geocod : '2109809', nome : "Curva Grande"},  
2459: {id : '6625', geocod : '2106300', nome : "Custódio Lima"},  
2460: {id : '509', geocod : '2103752', nome : "Davinópolis"},  
2461: {id : '461', geocod : '2103802', nome : "Dom Pedro"},  
2462: {id : '638', geocod : '2103901', nome : "Duque Bacelar"},  
2463: {id : '468', geocod : '2104008', nome : "Esperantinópolis"},  
2464: {id : '6768', geocod : '2102606', nome : "Estandarte"},  
2465: {id : '484', geocod : '2104057', nome : "Estreito"},  
2466: {id : '489', geocod : '2104073', nome : "Feira Nova do Maranhão"},  
2467: {id : '440', geocod : '2104081', nome : "Fernando Falcão"},  
2468: {id : '494', geocod : '2104099', nome : "Formosa da Serra Negra"},  
2469: {id : '493', geocod : '2104107', nome : "Fortaleza dos Nogueiras"},  
2470: {id : '448', geocod : '2104206', nome : "Fortuna"},  
2471: {id : '256', geocod : '2104305', nome : "Godofredo Viana"},  
2472: {id : '464', geocod : '2104404', nome : "Gonçalves Dias"},  
2473: {id : '544', geocod : '2104503', nome : "Governador Acher"},  
2474: {id : '508', geocod : '2104552', nome : "Governador Edison Lobão"},  
2475: {id : '449', geocod : '2104602', nome : "Governador Eugênio Barros"},  
2476: {id : '450', geocod : '2104628', nome : "Governador Luiz Rocha"},  
2477: {id : '198', geocod : '2104651', nome : "Governador Newton Bello"},  
2478: {id : '213', geocod : '2104677', nome : "Governador Nunes Freire"},  
2479: {id : '447', geocod : '2104701', nome : "Graça Aranha"},  
2480: {id : '522', geocod : '2104800', nome : "Grajaú"},  
2481: {id : '245', geocod : '2104909', nome : "Guimarães"},  
2482: {id : '277', geocod : '2105005', nome : "Humberto de Campos"},  
2483: {id : '7033', geocod : '2106706', nome : "Ibipira"},  
2484: {id : '252', geocod : '2105104', nome : "Icatu"},  
2485: {id : '207', geocod : '2105153', nome : "Igarapé do Meio"},  
2486: {id : '471', geocod : '2105203', nome : "Igarapé Grande"},  
2487: {id : '507', geocod : '2105302', nome : "Imperatriz"},  
2488: {id : '435', geocod : '2105351', nome : "Itaipava do Grajaú"},  
2489: {id : '232', geocod : '2105401', nome : "Itapecuru Mirim"},  
2490: {id : '7206', geocod : '2105104', nome : "Itapera"},  
2491: {id : '515', geocod : '2105427', nome : "Itinga do Maranhão"},  
2492: {id : '443', geocod : '2105450', nome : "Jatobá"},  
2493: {id : '541', geocod : '2105476', nome : "Jenipapo dos Vieiras"},  
2494: {id : '510', geocod : '2105500', nome : "João Lisboa"},  
2495: {id : '466', geocod : '2105609', nome : "Joselândia"},  
2496: {id : '191', geocod : '2105658', nome : "Junco do Maranhão"},  
2497: {id : '438', geocod : '2105708', nome : "Lago da Pedra"},  
2498: {id : '455', geocod : '2105807', nome : "Lago do Junco"},  
2499: {id : '469', geocod : '2105948', nome : "Lago dos Rodrigues"},  
2500: {id : '222', geocod : '2105906', nome : "Lago Verde"},  
2501: {id : '591', geocod : '2105922', nome : "Lagoa do Mato"},  
2502: {id : '436', geocod : '2105963', nome : "Lagoa Grande do Maranhão"},  
2503: {id : '517', geocod : '2105989', nome : "Lajeado Novo"},  
2504: {id : '545', geocod : '2106003', nome : "Lima Campos"},  
2505: {id : '530', geocod : '2106102', nome : "Loreto"},  
2506: {id : '258', geocod : '2106201', nome : "Luís Domingues"},  
2507: {id : '141', geocod : '2106300', nome : "Magalhães de Almeida"},  
2508: {id : '192', geocod : '2106326', nome : "Maracaçumé"},  
2509: {id : '437', geocod : '2106359', nome : "Marajá do Sena"},  
2510: {id : '216', geocod : '2106375', nome : "Maranhãozinho"},  
2511: {id : '7565', geocod : '2110302', nome : "Marianópolis"},  
2512: {id : '7593', geocod : '2111201', nome : "Mata"},  
2513: {id : '275', geocod : '2106409', nome : "Mata Roma"},  
2514: {id : '233', geocod : '2106508', nome : "Matinha"},  
2515: {id : '593', geocod : '2106607', nome : "Matões"},  
2516: {id : '230', geocod : '2106631', nome : "Matões do Norte"},  
2517: {id : '286', geocod : '2106672', nome : "Milagres do Maranhão"},  
2518: {id : '537', geocod : '2106706', nome : "Mirador"},  
2519: {id : '227', geocod : '2106755', nome : "Miranda do Norte"},  
2520: {id : '264', geocod : '2106805', nome : "Mirinzal"},  
2521: {id : '206', geocod : '2106904', nome : "Monção"},  
2522: {id : '518', geocod : '2107001', nome : "Montes Altos"},  
2523: {id : '276', geocod : '2107100', nome : "Morros"},  
2524: {id : '268', geocod : '2107209', nome : "Nina Rodrigues"},  
2525: {id : '492', geocod : '2107258', nome : "Nova Colinas"},  
2526: {id : '538', geocod : '2107308', nome : "Nova Iorque"},  
2527: {id : '210', geocod : '2107357', nome : "Nova Olinda do Maranhão"},  
2528: {id : '634', geocod : '2107407', nome : "Olho d'Água das Cunhãs"},  
2529: {id : '234', geocod : '2107456', nome : "Olinda Nova do Maranhão"},  
2530: {id : '254', geocod : '2107506', nome : "Paço do Lumiar"},  
2531: {id : '236', geocod : '2107605', nome : "Palmeirândia"},  
2532: {id : '7959', geocod : '2101608', nome : "Papagaio"},  
2533: {id : '556', geocod : '2107704', nome : "Paraibano"},  
2534: {id : '594', geocod : '2107803', nome : "Parnarama"},  
2535: {id : '590', geocod : '2107902', nome : "Passagem Franca"},  
2536: {id : '539', geocod : '2108009', nome : "Pastos Bons"},  
2537: {id : '145', geocod : '2108058', nome : "Paulino Neves"},  
2538: {id : '439', geocod : '2108108', nome : "Paulo Ramos"},  
2539: {id : '458', geocod : '2108207', nome : "Pedreiras"},  
2540: {id : '217', geocod : '2108256', nome : "Pedro do Rosário"},  
2541: {id : '208', geocod : '2108306', nome : "Penalva"},  
2542: {id : '242', geocod : '2108405', nome : "Peri Mirim"},  
2543: {id : '547', geocod : '2108454', nome : "Peritoró"},  
2544: {id : '8109', geocod : '2112274', nome : "Pimentel"},  
2545: {id : '205', geocod : '2108504', nome : "Pindaré-Mirim"},  
2546: {id : '241', geocod : '2108603', nome : "Pinheiro"},  
2547: {id : '202', geocod : '2108702', nome : "Pio XII"},  
2548: {id : '229', geocod : '2108801', nome : "Pirapemas"},  
2549: {id : '470', geocod : '2108900', nome : "Poção de Pedras"},  
2550: {id : '8236', geocod : '2100709', nome : "Porto das Gabarras"},  
2551: {id : '486', geocod : '2109007', nome : "Porto Franco"},  
2552: {id : '265', geocod : '2109056', nome : "Porto Rico do Maranhão"},  
2553: {id : '446', geocod : '2109106', nome : "Presidente Dutra"},  
2554: {id : '249', geocod : '2109205', nome : "Presidente Juscelino"},  
2555: {id : '215', geocod : '2109239', nome : "Presidente Médici"},  
2556: {id : '218', geocod : '2109270', nome : "Presidente Sarney"},  
2557: {id : '269', geocod : '2109304', nome : "Presidente Vargas"},  
2558: {id : '278', geocod : '2109403', nome : "Primeira Cruz"},  
2559: {id : '190', geocod : '2109452', nome : "Raposa"},  
2560: {id : '488', geocod : '2109502', nome : "Riachão"},  
2561: {id : '504', geocod : '2109551', nome : "Ribamar Fiquene"},  
2562: {id : '8378', geocod : '2110906', nome : "Ribeirão Azul"},  
2563: {id : '8461', geocod : '2108009', nome : "Roçado"},  
2564: {id : '8451', geocod : '2108603', nome : "Roque"},  
2565: {id : '248', geocod : '2109601', nome : "Rosário"},  
2566: {id : '528', geocod : '2109700', nome : "Sambaíba"},  
2567: {id : '445', geocod : '2109759', nome : "Santa Filomena do Maranhão"},  
2568: {id : '219', geocod : '2109809', nome : "Santa Helena"},  
2569: {id : '204', geocod : '2109908', nome : "Santa Inês"},  
2570: {id : '195', geocod : '2110005', nome : "Santa Luzia"},  
2571: {id : '214', geocod : '2110039', nome : "Santa Luzia do Paruá"},  
2572: {id : '287', geocod : '2110104', nome : "Santa Quitéria do Maranhão"},  
2573: {id : '246', geocod : '2110203', nome : "Santa Rita"},  
2574: {id : '146', geocod : '2110237', nome : "Santana do Maranhão"},  
2575: {id : '279', geocod : '2110278', nome : "Santo Amaro do Maranhão"},  
2576: {id : '462', geocod : '2110302', nome : "Santo Antônio dos Lopes"},  
2577: {id : '270', geocod : '2110401', nome : "São Benedito do Rio Preto"},  
2578: {id : '237', geocod : '2110500', nome : "São Bento"},  
2579: {id : '140', geocod : '2110609', nome : "São Bernardo"},  
2580: {id : '532', geocod : '2110658', nome : "São Domingos do Azeitão"},  
2581: {id : '444', geocod : '2110708', nome : "São Domingos do Maranhão"},  
2582: {id : '531', geocod : '2110807', nome : "São Félix de Balsas"},  
2583: {id : '514', geocod : '2110856', nome : "São Francisco do Brejão"},  
2584: {id : '601', geocod : '2110906', nome : "São Francisco do Maranhão"},  
2585: {id : '238', geocod : '2111003', nome : "São João Batista"},  
2586: {id : '8955', geocod : '2100204', nome : "São João de Cortes"},  
2587: {id : '150', geocod : '2111029', nome : "São João do Carú"},  
2588: {id : '490', geocod : '2111052', nome : "São João do Paraíso"},  
2589: {id : '595', geocod : '2111078', nome : "São João do Soter"},  
2590: {id : '558', geocod : '2111102', nome : "São João dos Patos"},  
2591: {id : '8881', geocod : '2112308', nome : "São Joaquim dos Melo"},  
2592: {id : '255', geocod : '2111201', nome : "São José de Ribamar"},  
2593: {id : '467', geocod : '2111250', nome : "São José dos Basílios"},  
2594: {id : '5479', geocod : '2111300', nome : "São Luis"},  
2595: {id : '460', geocod : '2111409', nome : "São Luís Gonzaga do Maranhão"},  
2596: {id : '635', geocod : '2111508', nome : "São Mateus do Maranhão"},  
2597: {id : '422', geocod : '2111532', nome : "São Pedro da Água Branca"},  
2598: {id : '491', geocod : '2111573', nome : "São Pedro dos Crentes"},  
2599: {id : '529', geocod : '2111607', nome : "São Raimundo das Mangabeiras"},  
2600: {id : '9039', geocod : '2103307', nome : "São Raimundo de Codó"},  
2601: {id : '452', geocod : '2111631', nome : "São Raimundo do Doca Bezerra"},  
2602: {id : '453', geocod : '2111672', nome : "São Roberto"},  
2603: {id : '235', geocod : '2111706', nome : "São Vicente Ferrer"},  
2604: {id : '200', geocod : '2111722', nome : "Satubinha"},  
2605: {id : '451', geocod : '2111748', nome : "Senador Alexandre Costa"},  
2606: {id : '511', geocod : '2111763', nome : "Senador La Rocque"},  
2607: {id : '260', geocod : '2111789', nome : "Serrano do Maranhão"},  
2608: {id : '519', geocod : '2111805', nome : "Sítio Novo"},  
2609: {id : '540', geocod : '2111904', nome : "Sucupira do Norte"},  
2610: {id : '559', geocod : '2111953', nome : "Sucupira do Riachão"},  
2611: {id : '1494', geocod : '2112001', nome : "Tasso Fragoso"},  
2612: {id : '596', geocod : '2112100', nome : "Timbiras"},  
2613: {id : '619', geocod : '2112209', nome : "Timon"},  
2614: {id : '457', geocod : '2112233', nome : "Trizidela do Vale"},  
2615: {id : '196', geocod : '2112274', nome : "Tufilândia"},  
2616: {id : '441', geocod : '2112308', nome : "Tuntum"},  
2617: {id : '259', geocod : '2112407', nome : "Turiaçu"},  
2618: {id : '220', geocod : '2112456', nome : "Turilândia"},  
2619: {id : '147', geocod : '2112506', nome : "Tutóia"},  
2620: {id : '274', geocod : '2112605', nome : "Urbano Santos"},  
2621: {id : '267', geocod : '2112704', nome : "Vargem Grande"},  
2622: {id : '225', geocod : '2112803', nome : "Viana"},  
2623: {id : '432', geocod : '2112852', nome : "Vila Nova dos Martírios"},  
2624: {id : '223', geocod : '2112902', nome : "Vitória do Mearim"},  
2625: {id : '633', geocod : '2113009', nome : "Vitorino Freire"},  
2626: {id : '199', geocod : '2114007', nome : "Zé Doca"}  
2627: ],  
2628: MG : [  
2629: // UF = MG  
2630: {id : '2683', geocod : '3100104', nome : "Abadia dos Dourados"},  
2631: {id : '2739', geocod : '3100203', nome : "Abaeté"},  
2632: {id : '5567', geocod : '3155504', nome : "Abaeté dos Mendes"},  
2633: {id : '5570', geocod : '3138401', nome : "Abaíba"},  
2634: {id : '4378', geocod : '3100302', nome : "Abre Campo"},  
2635: {id : '5573', geocod : '3102100', nome : "Abreus"},  
2636: {id : '4351', geocod : '3100401', nome : "Acaiaca"},  
2637: {id : '2716', geocod : '3100500', nome : "Açucena"},  
2638: {id : '5578', geocod : '3131901', nome : "Acuruí"},  
2639: {id : '5581', geocod : '3108503', nome : "Adão Colares"},  
2640: {id : '2720', geocod : '3100609', nome : "Água Boa"},  
2641: {id : '9465', geocod : '3117009', nome : "Água Branca de Minas"},  
2642: {id : '3687', geocod : '3100708', nome : "Água Comprida"},  
2643: {id : '9471', geocod : '3169208', nome : "Água Santa de Minas"},  
2644: {id : '9474', geocod : '3124609', nome : "Água Viva"},  
2645: {id : '4260', geocod : '3100807', nome : "Aguanil"},  
2646: {id : '9478', geocod : '3117702', nome : "Águas de Contendas"},  
2647: {id : '9477', geocod : '3164001', nome : "Águas Férreas"},  
2648: {id : '3034', geocod : '3100906', nome : "Águas Formosas"},  
2649: {id : '2255', geocod : '3101003', nome : "Águas Vermelhas"},  
2650: {id : '2993', geocod : '3101102', nome : "Aimorés"},  
2651: {id : '4096', geocod : '3101201', nome : "Aiuruoca"},  
2652: {id : '4074', geocod : '3101300', nome : "Alagoa"},  
2653: {id : '3888', geocod : '3101409', nome : "Albertina"},  
2654: {id : '5601', geocod : '3140506', nome : "Alberto Isaacson"},  
2655: {id : '5603', geocod : '3126109', nome : "Albertos"},  
2656: {id : '5607', geocod : '3120839', nome : "Aldeia"},  
2657: {id : '5613', geocod : '3119302', nome : "Alegre"},  
2658: {id : '5614', geocod : '3167608', nome : "Alegria"},  
2659: {id : '4214', geocod : '3101508', nome : "Além Paraíba"},  
2660: {id : '5618', geocod : '3134400', nome : "Alexandrita"},  
2661: {id : '4036', geocod : '3101607', nome : "Alfenas"},  
2662: {id : '4151', geocod : '3101631', nome : "Alfredo Vasconcelos"},  
2663: {id : '5625', geocod : '3134608', nome : "Almeida"},  
2664: {id : '3083', geocod : '3101706', nome : "Almenara"},  
2665: {id : '2978', geocod : '3101805', nome : "Alpercata"},  
2666: {id : '3958', geocod : '3101904', nome : "Alpinópolis"},  
2667: {id : '3932', geocod : '3102001', nome : "Alterosa"},  
2668: {id : '4393', geocod : '3102050', nome : "Alto Caparaó"},  
2669: {id : '5637', geocod : '3101102', nome : "Alto Capim"},  
2670: {id : '5656', geocod : '3127701', nome : "Alto de Santa Helena"},  
2671: {id : '4387', geocod : '3153509', nome : "Alto Jequitibá"},  
2672: {id : '5642', geocod : '3118007', nome : "Alto Maranhão"},  
2673: {id : '4175', geocod : '3102100', nome : "Alto Rio Doce"},  
2674: {id : '5659', geocod : '3168200', nome : "Altolândia"},  
2675: {id : '5661', geocod : '3118809', nome : "Alvação"},  
2676: {id : '2982', geocod : '3102209', nome : "Alvarenga"},  
2677: {id : '4353', geocod : '3102308', nome : "Alvinópolis"},  
2678: {id : '5662', geocod : '3113305', nome : "Alvorada"},  
2679: {id : '2862', geocod : '3102407', nome : "Alvorada de Minas"},  
2680: {id : '5668', geocod : '3105004', nome : "Amanda"},  
2681: {id : '5671', geocod : '3103504', nome : "Amanhece"},  
2682: {id : '5674', geocod : '3146107', nome : "Amarantina"},  
2683: {id : '4360', geocod : '3102506', nome : "Amparo do Serra"},  
2684: {id : '5688', geocod : '3124104', nome : "Andiroba"},  
2685: {id : '3889', geocod : '3102605', nome : "Andradas"},  
2686: {id : '3703', geocod : '3102803', nome : "Andrelândia"},  
2687: {id : '5690', geocod : '3169356', nome : "Andrequicé"},  
2688: {id : '5692', geocod : '3154101', nome : "Angaturama"},  
2689: {id : '2911', geocod : '3102852', nome : "Angelândia"},  
2690: {id : '5693', geocod : '3108602', nome : "Angicos de Minas"},  
2691: {id : '5694', geocod : '3120904', nome : "Angueretá"},  
2692: {id : '5695', geocod : '3101508', nome : "Angustura"},  
2693: {id : '4147', geocod : '3102902', nome : "Antônio Carlos"},  
2694: {id : '2807', geocod : '3103009', nome : "Antônio Dias"},  
2695: {id : '5713', geocod : '3110004', nome : "Antônio dos Santos"},  
2696: {id : '5708', geocod : '3126752', nome : "Antônio Ferreira"},  
2697: {id : '5711', geocod : '3146107', nome : "Antônio Pereira"},  
2698: {id : '4249', geocod : '3103108', nome : "Antônio Prado de Minas"},  
2699: {id : '5704', geocod : '3130200', nome : "Antunes"},  
2700: {id : '5719', geocod : '3127107', nome : "Aparecida de Minas"},  
2701: {id : '5723', geocod : '3143302', nome : "Aparecida do Mundo Novo"},  
2702: {id : '2712', geocod : '3103207', nome : "Araçaí"},  
2703: {id : '5759', geocod : '3148707', nome : "Araçaji de Minas"},  
2704: {id : '5732', geocod : '3115300', nome : "Aracati de Minas"},  
2705: {id : '4174', geocod : '3103306', nome : "Aracitaba"},  
2706: {id : '2917', geocod : '3103405', nome : "Araçuaí"},  
2707: {id : '2607', geocod : '3103504', nome : "Araguari"},  
2708: {id : '5740', geocod : '3100500', nome : "Aramirim"},  
2709: {id : '5742', geocod : '3109006', nome : "Aranha"},  
2710: {id : '3704', geocod : '3103603', nome : "Arantina"},  
2711: {id : '4366', geocod : '3103702', nome : "Araponga"},  
2712: {id : '2591', geocod : '3103751', nome : "Araporã"},  
2713: {id : '2674', geocod : '3103801', nome : "Arapuá"},  
2714: {id : '2724', geocod : '3103900', nome : "Araújos"},  
2715: {id : '5758', geocod : '3128105', nome : "Araúna"},  
2716: {id : '2664', geocod : '3104007', nome : "Araxá"},  
2717: {id : '5765', geocod : '3162500', nome : "Arcângelo"},  
2718: {id : '3909', geocod : '3104106', nome : "Arceburgo"},  
2719: {id : '4268', geocod : '3104205', nome : "Arcos"},  
2720: {id : '3931', geocod : '3104304', nome : "Areado"},  
2721: {id : '5770', geocod : '3129509', nome : "Argenita"},  
2722: {id : '4212', geocod : '3104403', nome : "Argirita"},  
2723: {id : '2897', geocod : '3104452', nome : "Aricanduva"},  
2724: {id : '2149', geocod : '3104502', nome : "Arinos"},  
2725: {id : '5779', geocod : '3118809', nome : "Aristides Batista"},  
2726: {id : '5798', geocod : '3147105', nome : "Ascenção"},  
2727: {id : '5799', geocod : '3151909', nome : "Assaraí"},  
2728: {id : '4220', geocod : '3104601', nome : "Astolfo Dutra"},  
2729: {id : '3022', geocod : '3104700', nome : "Ataléia"},  
2730: {id : '2771', geocod : '3104809', nome : "Augusto de Lima"},  
2731: {id : '5814', geocod : '3134707', nome : "Avaí do Jacinto"},  
2732: {id : '5820', geocod : '3140704', nome : "Azurita"},  
2733: {id : '5825', geocod : '3121209', nome : "Babilônia"},  
2734: {id : '5959', geocod : '3131901', nome : "Bação"},  
2735: {id : '4091', geocod : '3104908', nome : "Baependi"},  
2736: {id : '5834', geocod : '3127701', nome : "Baguari"},  
2737: {id : '5849', geocod : '3126109', nome : "Baiões"},  
2738: {id : '5840', geocod : '3170107', nome : "Baixa"},  
2739: {id : '2830', geocod : '3105004', nome : "Baldim"},  
2740: {id : '4256', geocod : '3105103', nome : "Bambuí"},  
2741: {id : '2268', geocod : '3105202', nome : "Bandeira"},  
2742: {id : '3920', geocod : '3105301', nome : "Bandeira do Sul"},  
2743: {id : '5862', geocod : '3140001', nome : "Bandeirantes"},  
2744: {id : '2713', geocod : '3105400', nome : "Barão de Cocais"},  
2745: {id : '4243', geocod : '3105509', nome : "Barão de Monte Alto"},  
2746: {id : '4150', geocod : '3105608', nome : "Barbacena"},  
2747: {id : '5873', geocod : '3131307', nome : "Barra Alegre"},  
2748: {id : '5889', geocod : '3151909', nome : "Barra da Figueira"},  
2749: {id : '5895', geocod : '3139607', nome : "Barra do Ariranha"},  
2750: {id : '5898', geocod : '3118403', nome : "Barra do Cuieté"},  
2751: {id : '5904', geocod : '3119500', nome : "Barra do Salinas"},  
2752: {id : '5878', geocod : '3157203', nome : "Barra Feliz"},  
2753: {id : '4352', geocod : '3105707', nome : "Barra Longa"},  
2754: {id : '5909', geocod : '3101607', nome : "Barranco Alto"},  
2755: {id : '5913', geocod : '3106200', nome : "Barreiro"},  
2756: {id : '5916', geocod : '3115474', nome : "Barreiro Branco"},  
2757: {id : '5917', geocod : '3135100', nome : "Barreiro da Raiz"},  
2758: {id : '5923', geocod : '3102308', nome : "Barretos de Alvinópo"},  
2759: {id : '5925', geocod : '3139250', nome : "Barrinha"},  
2760: {id : '5933', geocod : '3127800', nome : "Barrocão"},  
2761: {id : '4146', geocod : '3105905', nome : "Barroso"},  
2762: {id : '5956', geocod : '3124708', nome : "Baú"},  
2763: {id : '2837', geocod : '3106002', nome : "Bela Vista de Minas"},  
2764: {id : '5982', geocod : '3143906', nome : "Belisário"},  
2765: {id : '4155', geocod : '3106101', nome : "Belmiro Braga"},  
2766: {id : '5480', geocod : '3106200', nome : "Belo Horizonte"},  
2767: {id : '2714', geocod : '3106309', nome : "Belo Oriente"},  
2768: {id : '4328', geocod : '3106408', nome : "Belo Vale"},  
2769: {id : '5986', geocod : '3119104', nome : "Beltrão"},  
2770: {id : '5997', geocod : '3170008', nome : "Bentópolis de Minas"},  
2771: {id : '2914', geocod : '3106507', nome : "Berilo"},  
2772: {id : '2256', geocod : '3106655', nome : "Berizal"},  
2773: {id : '3051', geocod : '3106606', nome : "Bertópolis"},  
2774: {id : '2786', geocod : '3106705', nome : "Betim"},  
2775: {id : '4143', geocod : '3106804', nome : "Bias Fortes"},  
2776: {id : '4169', geocod : '3106903', nome : "Bicas"},  
2777: {id : '6004', geocod : '3154002', nome : "Bicuíba"},  
2778: {id : '2759', geocod : '3107000', nome : "Biquinhas"},  
2779: {id : '6007', geocod : '3135407', nome : "Bituri"},  
2780: {id : '4052', geocod : '3107109', nome : "Boa Esperança"},  
2781: {id : '6018', geocod : '3143906', nome : "Boa Família"},  
2782: {id : '6021', geocod : '3131802', nome : "Boa União de Itabirinha"},  
2783: {id : '6030', geocod : '3145208', nome : "Boa Vista de Minas"},  
2784: {id : '4082', geocod : '3107208', nome : "Bocaina de Minas"},  
2785: {id : '2875', geocod : '3107307', nome : "Bocaiúva"},  
2786: {id : '2737', geocod : '3107406', nome : "Bom Despacho"},  
2787: {id : '3705', geocod : '3107505', nome : "Bom Jardim de Minas"},  
2788: {id : '6052', geocod : '3143906', nome : "Bom Jesus da Cachoeira"},  
2789: {id : '3928', geocod : '3107604', nome : "Bom Jesus da Penha"},  
2790: {id : '6053', geocod : '3129608', nome : "Bom Jesus da Vereda"},  
2791: {id : '6054', geocod : '3170503', nome : "Bom Jesus de Cardoso"},  
2792: {id : '2833', geocod : '3107703', nome : "Bom Jesus do Amparo"},  
2793: {id : '6055', geocod : '3106309', nome : "Bom Jesus do Bagre"},  
2794: {id : '6056', geocod : '3122009', nome : "Bom Jesus do Divino"},  
2795: {id : '2788', geocod : '3107802', nome : "Bom Jesus do Galho"},  
2796: {id : '6057', geocod : '3125952', nome : "Bom Jesus do Madeira"},  
2797: {id : '6059', geocod : '3159407', nome : "Bom Jesus do Vermelho"},  
2798: {id : '6061', geocod : '3154309', nome : "Bom Pastor"},  
2799: {id : '3893', geocod : '3107901', nome : "Bom Repouso"},  
2800: {id : '6063', geocod : '3157302', nome : "Bom Retiro"},  
2801: {id : '4106', geocod : '3108008', nome : "Bom Sucesso"},  
2802: {id : '6071', geocod : '3148004', nome : "Bom Sucesso de Patos"},  
2803: {id : '6072', geocod : '3129657', nome : "Bonança"},  
2804: {id : '4305', geocod : '3108107', nome : "Bonfim"},  
2805: {id : '6077', geocod : '3105707', nome : "Bonfim da Barra"},  
2806: {id : '2928', geocod : '3108206', nome : "Bonfinópolis de Minas"},  
2807: {id : '2157', geocod : '3108255', nome : "Bonito de Minas"},  
2808: {id : '6087', geocod : '3125903', nome : "Borba Gato"},  
2809: {id : '3896', geocod : '3108305', nome : "Borda da Mata"},  
2810: {id : '3921', geocod : '3108404', nome : "Botelhos"},  
2811: {id : '2881', geocod : '3108503', nome : "Botumirim"},  
2812: {id : '4337', geocod : '3108701', nome : "Brás Pires"},  
2813: {id : '2762', geocod : '3108552', nome : "Brasilândia de Minas"},  
2814: {id : '2942', geocod : '3108602', nome : "Brasília de Minas"},  
2815: {id : '4010', geocod : '3108909', nome : "Brasópolis"},  
2816: {id : '2715', geocod : '3108800', nome : "Braúnas"},  
2817: {id : '6108', geocod : '3117504', nome : "Breja·ba"},  
2818: {id : '6107', geocod : '3127701', nome : "Brejaubinha"},  
2819: {id : '6112', geocod : '3148004', nome : "Brejo Bonito"},  
2820: {id : '6118', geocod : '3135209', nome : "Brejo do Amparo"},  
2821: {id : '6119', geocod : '3170909', nome : "Brejo do Mutambal"},  
2822: {id : '4308', geocod : '3109006', nome : "Brumadinho"},  
2823: {id : '6123', geocod : '3157203', nome : "Brumal"},  
2824: {id : '6125', geocod : '3118304', nome : "Buarque de Macedo"},  
2825: {id : '6127', geocod : '3118403', nome : "Bueno"},  
2826: {id : '3891', geocod : '3109105', nome : "Bueno Brandão"},  
2827: {id : '2773', geocod : '3109204', nome : "Buenópolis"},  
2828: {id : '2792', geocod : '3109253', nome : "Bugre"},  
2829: {id : '2133', geocod : '3109303', nome : "Buritis"},  
2830: {id : '2774', geocod : '3109402', nome : "Buritizeiro"},  
2831: {id : '6145', geocod : '3136306', nome : "Caatinga"},  
2832: {id : '2924', geocod : '3109451', nome : "Cabeceira Grande"},  
2833: {id : '3929', geocod : '3109501', nome : "Cabo Verde"},  
2834: {id : '6413', geocod : '3169703', nome : "Caçaratiba"},  
2835: {id : '6414', geocod : '3112703', nome : "Caçarema"},  
2836: {id : '6158', geocod : '3105509', nome : "Cachoeira Alegre"},  
2837: {id : '2701', geocod : '3109600', nome : "Cachoeira da Prata"},  
2838: {id : '3995', geocod : '3109709', nome : "Cachoeira de Minas"},  
2839: {id : '2254', geocod : '3102704', nome : "Cachoeira de Pajeú"},  
2840: {id : '6163', geocod : '3171303', nome : "Cachoeira de Santa Clara"},  
2841: {id : '6166', geocod : '3140001', nome : "Cachoeira do Brumado"},  
2842: {id : '6167', geocod : '3146107', nome : "Cachoeira do Campo"},  
2843: {id : '6169', geocod : '3109402', nome : "Cachoeira do Manteiga"},  
2844: {id : '6171', geocod : '3116100', nome : "Cachoeira do Norte"},  
2845: {id : '6173', geocod : '3168705', nome : "Cachoeira do Vale"},  
2846: {id : '2567', geocod : '3109808', nome : "Cachoeira Dourada"},  
2847: {id : '6175', geocod : '3119807', nome : "Cachoeirinha"},  
2848: {id : '6179', geocod : '3135407', nome : "Caetano Lopes"},  
2849: {id : '2709', geocod : '3109907', nome : "Caetanópolis"},  
2850: {id : '2826', geocod : '3110004', nome : "Caeté"},  
2851: {id : '6183', geocod : '3168408', nome : "Cafemirim"},  
2852: {id : '4423', geocod : '3110103', nome : "Caiana"},  
2853: {id : '6187', geocod : '3151107', nome : "Caiapó"},  
2854: {id : '4358', geocod : '3110202', nome : "Cajuri"},  
2855: {id : '3919', geocod : '3110301', nome : "Caldas"},  
2856: {id : '6213', geocod : '3154309', nome : "Calixto"},  
2857: {id : '4266', geocod : '3110400', nome : "Camacho"},  
2858: {id : '3882', geocod : '3110509', nome : "Camanducaia"},  
2859: {id : '6216', geocod : '3140001', nome : "Camargos"},  
2860: {id : '3725', geocod : '3110608', nome : "Cambuí"},  
2861: {id : '4048', geocod : '3110707', nome : "Cambuquira"},  
2862: {id : '2972', geocod : '3110806', nome : "Campanário"},  
2863: {id : '4042', geocod : '3110905', nome : "Campanha"},  
2864: {id : '3922', geocod : '3111002', nome : "Campestre"},  
2865: {id : '6233', geocod : '3102605', nome : "Campestrinho"},  
2866: {id : '2551', geocod : '3111101', nome : "Campina Verde"},  
2867: {id : '6245', geocod : '3129657', nome : "Campo Alegre de Minas - Ibiraca"},  
2868: {id : '6246', geocod : '3154309', nome : "Campo Alegre de Minas - Resplen"},  
2869: {id : '2935', geocod : '3111150', nome : "Campo Azul"},  
2870: {id : '4262', geocod : '3111200', nome : "Campo Belo"},  
2871: {id : '4040', geocod : '3111309', nome : "Campo do Meio"},  
2872: {id : '2556', geocod : '3111408', nome : "Campo Florido"},  
2873: {id : '6254', geocod : '3170909', nome : "Campo Redondo"},  
2874: {id : '2671', geocod : '3111507', nome : "Campos Altos"},  
2875: {id : '4039', geocod : '3111606', nome : "Campos Gerais"},  
2876: {id : '4056', geocod : '3111903', nome : "Cana Verde"},  
2877: {id : '4364', geocod : '3111705', nome : "Canaã"},  
2878: {id : '6268', geocod : '3126703', nome : "Canabrava - Francisco Sá"},  
2879: {id : '6269', geocod : '3136306', nome : "Canabrava - João Pinheiro"},  
2880: {id : '2589', geocod : '3111804', nome : "Canápolis"},  
2881: {id : '6275', geocod : '3168903', nome : "Canastrão"},  
2882: {id : '4263', geocod : '3112000', nome : "Candeias"},  
2883: {id : '6293', geocod : '3154408', nome : "Canjamba"},  
2884: {id : '6295', geocod : '3161700', nome : "Canoeiros"},  
2885: {id : '2888', geocod : '3112059', nome : "Cantagalo"},  
2886: {id : '4426', geocod : '3112109', nome : "Caparaó"},  
2887: {id : '4325', geocod : '3112208', nome : "Capela Nova"},  
2888: {id : '2905', geocod : '3112307', nome : "Capelinha"},  
2889: {id : '3721', geocod : '3112406', nome : "Capetinga"},  
2890: {id : '2706', geocod : '3112505', nome : "Capim Branco"},  
2891: {id : '2566', geocod : '3112604', nome : "Capinópolis"},  
2892: {id : '6305', geocod : '3142700', nome : "Capitania"},  
2893: {id : '2987', geocod : '3112653', nome : "Capitão Andrade"},  
2894: {id : '2949', geocod : '3112703', nome : "Capitão Enéas"},  
2895: {id : '4251', geocod : '3112802', nome : "Capitólio"},  
2896: {id : '4345', geocod : '3112901', nome : "Caputira"},  
2897: {id : '3028', geocod : '3113008', nome : "Caraí"},  
2898: {id : '4323', geocod : '3113107', nome : "Caranaíba"},  
2899: {id : '4296', geocod : '3113206', nome : "Carandaí"},  
2900: {id : '4317', geocod : '3113305', nome : "Carangola"},  
2901: {id : '2795', geocod : '3113404', nome : "Caratinga"},  
2902: {id : '2879', geocod : '3113503', nome : "Carbonita"},  
2903: {id : '6349', geocod : '3159001', nome : "Cardeal Mota"},  
2904: {id : '3999', geocod : '3113602', nome : "Careaçu"},  
2905: {id : '6353', geocod : '3147105', nome : "Carioca"},  
2906: {id : '6358', geocod : '3162906', nome : "Carlos Alves"},  
2907: {id : '3047', geocod : '3113701', nome : "Carlos Chagas"},  
2908: {id : '2848', geocod : '3113800', nome : "Carmésia"},  
2909: {id : '4054', geocod : '3113909', nome : "Carmo da Cachoeira"},  
2910: {id : '4276', geocod : '3114006', nome : "Carmo da Mata"},  
2911: {id : '4027', geocod : '3114105', nome : "Carmo de Minas"},  
2912: {id : '4281', geocod : '3114204', nome : "Carmo do Cajuru"},  
2913: {id : '2673', geocod : '3114303', nome : "Carmo do Paranaíba"},  
2914: {id : '4037', geocod : '3114402', nome : "Carmo do Rio Claro"},  
2915: {id : '4277', geocod : '3114501', nome : "Carmópolis de Minas"},  
2916: {id : '2496', geocod : '3114550', nome : "Carneirinho"},  
2917: {id : '4103', geocod : '3114600', nome : "Carrancas"},  
2918: {id : '6373', geocod : '3156700', nome : "Carvalho de Brito"},  
2919: {id : '4032', geocod : '3114709', nome : "Carvalhópolis"},  
2920: {id : '4109', geocod : '3114808', nome : "Carvalhos"},  
2921: {id : '4295', geocod : '3114907', nome : "Casa Grande"},  
2922: {id : '2675', geocod : '3115003', nome : "Cascalho Rico"},  
2923: {id : '3954', geocod : '3115102', nome : "Cássia"},  
2924: {id : '6385', geocod : '3115300', nome : "Cataguarino"},  
2925: {id : '4226', geocod : '3115300', nome : "Cataguases"},  
2926: {id : '6386', geocod : '3160306', nome : "Catajás"},  
2927: {id : '4264', geocod : '3115359', nome : "Catas Altas"},  
2928: {id : '4335', geocod : '3115409', nome : "Catas Altas da Noruega"},  
2929: {id : '6390', geocod : '3166808', nome : "Catiara"},  
2930: {id : '3027', geocod : '3115458', nome : "Catuji"},  
2931: {id : '6400', geocod : '3169208', nome : "Catuné"},  
2932: {id : '6399', geocod : '3126703', nome : "Catuni"},  
2933: {id : '2184', geocod : '3115474', nome : "Catuti"},  
2934: {id : '4090', geocod : '3115508', nome : "Caxambu"},  
2935: {id : '2734', geocod : '3115607', nome : "Cedro do Abaeté"},  
2936: {id : '6420', geocod : '3151909', nome : "Centenário"},  
2937: {id : '2976', geocod : '3115706', nome : "Central de Minas"},  
2938: {id : '6421', geocod : '3122108', nome : "Central de Santa Helena"},  
2939: {id : '2590', geocod : '3115805', nome : "Centralina"},  
2940: {id : '6435', geocod : '3108305', nome : "Cervo"},  
2941: {id : '4165', geocod : '3115904', nome : "Chácara"},  
2942: {id : '4398', geocod : '3116001', nome : "Chalé"},  
2943: {id : '6438', geocod : '3124807', nome : "Chapada de Minas"},  
2944: {id : '2909', geocod : '3116100', nome : "Chapada do Norte"},  
2945: {id : '2150', geocod : '3116159', nome : "Chapada Gaúcha"},  
2946: {id : '6439', geocod : '3159803', nome : "Chaveslândia"},  
2947: {id : '4162', geocod : '3116209', nome : "Chiador"},  
2948: {id : '6442', geocod : '3127701', nome : "Chonim"},  
2949: {id : '6443', geocod : '3127701', nome : "Chonim de Baixo"},  
2950: {id : '6446', geocod : '3148004', nome : "Chumbo"},  
2951: {id : '4333', geocod : '3116308', nome : "Cipotânea"},  
2952: {id : '6458', geocod : '3146701', nome : "Cisneiros"},  
2953: {id : '3951', geocod : '3116407', nome : "Claraval"},  
2954: {id : '6462', geocod : '3171006', nome : "Claro de Minas"},  
2955: {id : '2783', geocod : '3116506', nome : "Claro dos Poções"},  
2956: {id : '4280', geocod : '3116605', nome : "Cláudio"},  
2957: {id : '6464', geocod : '3140001', nome : "Cláudio Manoel"},  
2958: {id : '6468', geocod : '3105400', nome : "Cocais"},  
2959: {id : '6470', geocod : '3142304', nome : "Coco"},  
2960: {id : '4357', geocod : '3116704', nome : "Coimbra"},  
2961: {id : '6488', geocod : '3105608', nome : "Colônia Rodrigo Silva"},  
2962: {id : '2891', geocod : '3116803', nome : "Coluna"},  
2963: {id : '2553', geocod : '3116902', nome : "Comendador Gomes"},  
2964: {id : '3068', geocod : '3117009', nome : "Comercinho"},  
2965: {id : '3933', geocod : '3117108', nome : "Conceição da Aparecida"},  
2966: {id : '3708', geocod : '3115201', nome : "Conceição da Barra de Minas"},  
2967: {id : '6498', geocod : '3154101', nome : "Conceição da Boa Vista"},  
2968: {id : '6499', geocod : '3127503', nome : "Conceição da Brejaúba"},  
2969: {id : '6500', geocod : '3138609', nome : "Conceição da Ibitipo"},  
2970: {id : '2557', geocod : '3117306', nome : "Conceição das Alagoas"},  
2971: {id : '4020', geocod : '3117207', nome : "Conceição das Pedras"},  
2972: {id : '2980', geocod : '3117405', nome : "Conceição de Ipanema"},  
2973: {id : '6502', geocod : '3109006', nome : "Conceição de Itaguá"},  
2974: {id : '6504', geocod : '3121803', nome : "Conceição de Minas"},  
2975: {id : '6506', geocod : '3155702', nome : "Conceição de Piracic"},  
2976: {id : '6507', geocod : '3119203', nome : "Conceição de Tronque"},  
2977: {id : '6508', geocod : '3101102', nome : "Conceição do Capim"},  
2978: {id : '6509', geocod : '3160702', nome : "Conceição do Formoso"},  
2979: {id : '2843', geocod : '3117504', nome : "Conceição do Mato Dentro"},  
2980: {id : '2747', geocod : '3117603', nome : "Conceição do Pará"},  
2981: {id : '6511', geocod : '3157203', nome : "Conceição do Rio Acima"},  
2982: {id : '4089', geocod : '3117702', nome : "Conceição do Rio Verde"},  
2983: {id : '3994', geocod : '3117801', nome : "Conceição dos Ouros"},  
2984: {id : '6513', geocod : '3137007', nome : "Concórdia do Mucuri"},  
2985: {id : '6514', geocod : '3162401', nome : "Condado do Norte"},  
2986: {id : '6638', geocod : '3161007', nome : "Cônego João Pio"},  
2987: {id : '2160', geocod : '3117836', nome : "Cônego Marinho"},  
2988: {id : '2824', geocod : '3117876', nome : "Confins"},  
2989: {id : '3991', geocod : '3117900', nome : "Congonhal"},  
2990: {id : '4302', geocod : '3118007', nome : "Congonhas"},  
2991: {id : '2857', geocod : '3118106', nome : "Congonhas do Norte"},  
2992: {id : '2659', geocod : '3118205', nome : "Conquista"},  
2993: {id : '4301', geocod : '3118304', nome : "Conselheiro Lafaiete"},  
2994: {id : '6520', geocod : '3121605', nome : "Conselheiro Mata"},  
2995: {id : '2988', geocod : '3118403', nome : "Conselheiro Pena"},  
2996: {id : '3982', geocod : '3118502', nome : "Consolação"},  
2997: {id : '2703', geocod : '3118601', nome : "Contagem"},  
2998: {id : '6526', geocod : '3132503', nome : "Contrato"},  
2999: {id : '6527', geocod : '3119104', nome : "Contria"},  
3000: {id : '4053', geocod : '3118700', nome : "Coqueiral"},  
3001: {id : '2940', geocod : '3118809', nome : "Coração de Jesus"},  
3002: {id : '6534', geocod : '3113404', nome : "Cordeiro de Minas"},  
3003: {id : '2711', geocod : '3118908', nome : "Cordisburgo"},  
3004: {id : '4034', geocod : '3119005', nome : "Cordislândia"},  
3005: {id : '6536', geocod : '3130309', nome : "Corguinhos"},  
3006: {id : '2768', geocod : '3119104', nome : "Corinto"},  
3007: {id : '2901', geocod : '3119203', nome : "Coroaci"},  
3008: {id : '2684', geocod : '3119302', nome : "Coromandel"},  
3009: {id : '2811', geocod : '3119401', nome : "Coronel Fabriciano"},  
3010: {id : '2953', geocod : '3119500', nome : "Coronel Murta"},  
3011: {id : '4166', geocod : '3119609', nome : "Coronel Pacheco"},  
3012: {id : '3714', geocod : '3119708', nome : "Coronel Xavier Chaves"},  
3013: {id : '6543', geocod : '3105608', nome : "Correa de Almeida"},  
3014: {id : '2726', geocod : '3119807', nome : "Córrego Danta"},  
3015: {id : '6633', geocod : '3147105', nome : "Córrego do Barro"},  
3016: {id : '3981', geocod : '3119906', nome : "Córrego do Bom Jesus"},  
3017: {id : '6634', geocod : '3111606', nome : "Córrego do Ouro"},  
3018: {id : '4382', geocod : '3119955', nome : "Córrego Fundo"},  
3019: {id : '2717', geocod : '3120003', nome : "Córrego Novo"},  
3020: {id : '6637', geocod : '3117504', nome : "Córregos"},  
3021: {id : '6548', geocod : '3128006', nome : "Correntinho"},  
3022: {id : '6551', geocod : '3117504', nome : "Costa Sena"},  
3023: {id : '6554', geocod : '3147303', nome : "Costas"},  
3024: {id : '6555', geocod : '3157302', nome : "Costas da Mantiqueira"},  
3025: {id : '2868', geocod : '3120102', nome : "Couto de Magalhães de Minas"},  
3026: {id : '6583', geocod : '3146008', nome : "Crisólia"},  
3027: {id : '3033', geocod : '3120151', nome : "Crisólita"},  
3028: {id : '6572', geocod : '3168606', nome : "Crispim Jaques"},  
3029: {id : '4259', geocod : '3120201', nome : "Cristais"},  
3030: {id : '2944', geocod : '3120300', nome : "Cristália"},  
3031: {id : '4298', geocod : '3120409', nome : "Cristiano Otoni"},  
3032: {id : '4024', geocod : '3120508', nome : "Cristina"},  
3033: {id : '4292', geocod : '3120607', nome : "Crucilândia"},  
3034: {id : '2668', geocod : '3120706', nome : "Cruzeiro da Fortaleza"},  
3035: {id : '6597', geocod : '3170206', nome : "Cruzeiro dos Peixoto"},  
3036: {id : '4095', geocod : '3120805', nome : "Cruzília"},  
3037: {id : '6599', geocod : '3125903', nome : "Cubas"},  
3038: {id : '6603', geocod : '3118403', nome : "Cuieté Velho"},  
3039: {id : '2999', geocod : '3120839', nome : "Cuparaque"},  
3040: {id : '6612', geocod : '3109204', nome : "Curimataí"},  
3041: {id : '2253', geocod : '3120870', nome : "Curral de Dentro"},  
3042: {id : '2766', geocod : '3120904', nome : "Curvelo"},  
3043: {id : '2860', geocod : '3121001', nome : "Datas"},  
3044: {id : '4021', geocod : '3121100', nome : "Delfim Moreira"},  
3045: {id : '3955', geocod : '3121209', nome : "Delfinópolis"},  
3046: {id : '2658', geocod : '3121258', nome : "Delta"},  
3047: {id : '6651', geocod : '3167103', nome : "Deputado Augusto Clementino"},  
3048: {id : '6654', geocod : '3127701', nome : "Derribadinha"},  
3049: {id : '4218', geocod : '3121308', nome : "Descoberto"},  
3050: {id : '6656', geocod : '3121605', nome : "Desembargador Otoni"},  
3051: {id : '6657', geocod : '3156908', nome : "Desemboque"},  
3052: {id : '4286', geocod : '3121407', nome : "Desterro de Entre Rios"},  
3053: {id : '4153', geocod : '3121506', nome : "Desterro do Melo"},  
3054: {id : '6661', geocod : '3169901', nome : "Diamante de Uba"},  
3055: {id : '2861', geocod : '3121605', nome : "Diamantina"},  
3056: {id : '6662', geocod : '3108909', nome : "Dias"},  
3057: {id : '4340', geocod : '3121704', nome : "Diogo de Vasconcelos"},  
3058: {id : '2805', geocod : '3121803', nome : "Dionísio"},  
3059: {id : '4185', geocod : '3121902', nome : "Divinésia"},  
3060: {id : '4318', geocod : '3122009', nome : "Divino"},  
3061: {id : '2968', geocod : '3122108', nome : "Divino das Laranjeiras"},  
3062: {id : '6667', geocod : '3171907', nome : "Divino de Virgolândia"},  
3063: {id : '6668', geocod : '3123700', nome : "Divino do Traíra"},  
3064: {id : '6666', geocod : '3102001', nome : "Divino Espírito Santo"},  
3065: {id : '2886', geocod : '3122207', nome : "Divinolândia de Minas"},  
3066: {id : '4279', geocod : '3122306', nome : "Divinópolis"},  
3067: {id : '2261', geocod : '3122355', nome : "Divisa Alegre"},  
3068: {id : '3923', geocod : '3122405', nome : "Divisa Nova"},  
3069: {id : '2263', geocod : '3122454', nome : "Divisópolis"},  
3070: {id : '6675', geocod : '3146750', nome : "Dois de Abril"},  
3071: {id : '2926', geocod : '3122470', nome : "Dom Bosco"},  
3072: {id : '2852', geocod : '3122504', nome : "Dom Cavati"},  
3073: {id : '6676', geocod : '3139409', nome : "Dom Corrêa"},  
3074: {id : '2844', geocod : '3122603', nome : "Dom Joaquim"},  
3075: {id : '6677', geocod : '3113404', nome : "Dom Lara"},  
3076: {id : '6680', geocod : '3113404', nome : "Dom Modesto"},  
3077: {id : '4374', geocod : '3122702', nome : "Dom Silvério"},  
3078: {id : '4026', geocod : '3122801', nome : "Dom Viçoso"},  
3079: {id : '4221', geocod : '3122900', nome : "Dona Eusébia"},  
3080: {id : '6688', geocod : '3142205', nome : "Dores da Vitória"},  
3081: {id : '4145', geocod : '3123007', nome : "Dores de Campos"},  
3082: {id : '2813', geocod : '3123106', nome : "Dores de Guanhães"},  
3083: {id : '2736', geocod : '3123205', nome : "Dores do Indaiá"},  
3084: {id : '6690', geocod : '3160702', nome : "Dores do Paraibuna"},  
3085: {id : '4183', geocod : '3123304', nome : "Dores do Turvo"},  
3086: {id : '4255', geocod : '3123403', nome : "Doresópolis"},  
3087: {id : '6694', geocod : '3139003', nome : "Douradinho"},  
3088: {id : '2682', geocod : '3123502', nome : "Douradoquara"},  
3089: {id : '6699', geocod : '3135704', nome : "Doutor Campolina"},  
3090: {id : '6703', geocod : '3149309', nome : "Doutor Lund"},  
3091: {id : '6705', geocod : '3102902', nome : "Doutor Sá Fortes"},  
3092: {id : '4395', geocod : '3123528', nome : "Durandé"},  
3093: {id : '6710', geocod : '3133204', nome : "Edgard Melo"},  
3094: {id : '4044', geocod : '3123601', nome : "Elói Mendes"},  
3095: {id : '6716', geocod : '3162500', nome : "Emboabas"},  
3096: {id : '2856', geocod : '3123700', nome : "Engenheiro Caldas"},  
3097: {id : '6723', geocod : '3146107', nome : "Engenheiro Correia"},  
3098: {id : '2874', geocod : '3123809', nome : "Engenheiro Navarro"},  
3099: {id : '6730', geocod : '3103405', nome : "Engenheiro Schnoor"},  
3100: {id : '6738', geocod : '3107406', nome : "Engenho do Ribeiro"},  
3101: {id : '6734', geocod : '3139805', nome : "Engenho Novo"},  
3102: {id : '2791', geocod : '3123858', nome : "Entre Folhas"},  
3103: {id : '4290', geocod : '3123908', nome : "Entre Rios de Minas"},  
3104: {id : '6745', geocod : '3113701', nome : "Epaminondas Otoni"},  
3105: {id : '6747', geocod : '3142007', nome : "Ermidinha"},  
3106: {id : '4361', geocod : '3124005', nome : "Ervália"},  
3107: {id : '2699', geocod : '3124104', nome : "Esmeraldas"},  
3108: {id : '6750', geocod : '3125903', nome : "Esmeraldas de Ferros"},  
3109: {id : '4424', geocod : '3124203', nome : "Espera Feliz"},  
3110: {id : '2190', geocod : '3124302', nome : "Espinosa"},  
3111: {id : '3993', geocod : '3124401', nome : "Espírito Santo do Dourado"},  
3112: {id : '6770', geocod : '3138807', nome : "Esteios"},  
3113: {id : '6781', geocod : '3103702', nome : "Estêvão de Araújo"},  
3114: {id : '3989', geocod : '3124500', nome : "Estiva"},  
3115: {id : '6775', geocod : '3114550', nome : "Estrela da Barra"},  
3116: {id : '4235', geocod : '3124609', nome : "Estrela Dalva"},  
3117: {id : '6776', geocod : '3136504', nome : "Estrela de Jordânia"},  
3118: {id : '2729', geocod : '3124708', nome : "Estrela do Indaiá"},  
3119: {id : '2680', geocod : '3124807', nome : "Estrela do Sul"},  
3120: {id : '4248', geocod : '3124906', nome : "Eugenópolis"},  
3121: {id : '6783', geocod : '3156809', nome : "Euxenita"},  
3122: {id : '4159', geocod : '3125002', nome : "Ewbank da Câmara"},  
3123: {id : '6786', geocod : '3101102', nome : "Expedicionário Alício"},  
3124: {id : '6787', geocod : '3121605', nome : "Extração"},  
3125: {id : '3878', geocod : '3125101', nome : "Extrema"},  
3126: {id : '4038', geocod : '3125200', nome : "Fama"},  
3127: {id : '6789', geocod : '3105608', nome : "Faria"},  
3128: {id : '4316', geocod : '3125309', nome : "Faria Lemos"},  
3129: {id : '6791', geocod : '3128006', nome : "Farias"},  
3130: {id : '6875', geocod : '3114550', nome : "Fátima"},  
3131: {id : '6806', geocod : '3158508', nome : "Fechados"},  
3132: {id : '6811', geocod : '3100500', nome : "Felicina"},  
3133: {id : '2871', geocod : '3125408', nome : "Felício dos Santos"},  
3134: {id : '3079', geocod : '3125606', nome : "Felisburgo"},  
3135: {id : '2763', geocod : '3125705', nome : "Felixlândia"},  
3136: {id : '2854', geocod : '3125804', nome : "Fernandes Tourinho"},  
3137: {id : '6816', geocod : '3108602', nome : "Fernão Dias"},  
3138: {id : '6822', geocod : '3162005', nome : "Ferreiras"},  
3139: {id : '6823', geocod : '3157005', nome : "Ferreirópolis"},  
3140: {id : '2847', geocod : '3125903', nome : "Ferros"},  
3141: {id : '6824', geocod : '3118403', nome : "Ferruginha"},  
3142: {id : '4383', geocod : '3125952', nome : "Fervedouro"},  
3143: {id : '6825', geocod : '3149309', nome : "Fidalgo"},  
3144: {id : '6826', geocod : '3104700', nome : "Fidelândia"},  
3145: {id : '6845', geocod : '3157203', nome : "Florália"},  
3146: {id : '6832', geocod : '3129103', nome : "FlordDe Minas"},  
3147: {id : '6835', geocod : '3115706', nome : "Floresta"},  
3148: {id : '2698', geocod : '3126000', nome : "Florestal"},  
3149: {id : '6839', geocod : '3103504', nome : "Florestina"},  
3150: {id : '6847', geocod : '3102308', nome : "Fonseca"},  
3151: {id : '4261', geocod : '3126109', nome : "Formiga"},  
3152: {id : '2134', geocod : '3126208', nome : "Formoso"},  
3153: {id : '3724', geocod : '3126307', nome : "Fortaleza de Minas"},  
3154: {id : '2700', geocod : '3126406', nome : "Fortuna de Minas"},  
3155: {id : '6856', geocod : '3114808', nome : "Franceses"},  
3156: {id : '2915', geocod : '3126505', nome : "Francisco Badaró"},  
3157: {id : '2780', geocod : '3126604', nome : "Francisco Dumont"},  
3158: {id : '2950', geocod : '3126703', nome : "Francisco Sá"},  
3159: {id : '2723', geocod : '3126752', nome : "Franciscópolis"},  
3160: {id : '6861', geocod : '3118700', nome : "Frei Eustáquio"},  
3161: {id : '2974', geocod : '3126802', nome : "Frei Gaspar"},  
3162: {id : '2966', geocod : '3126901', nome : "Frei Inocêncio"},  
3163: {id : '2892', geocod : '3126950', nome : "Frei Lagonegro"},  
3164: {id : '6862', geocod : '3143500', nome : "Frei Orlando"},  
3165: {id : '6864', geocod : '3132701', nome : "Frei Serafim"},  
3166: {id : '6866', geocod : '3119500', nome : "Freire Cardoso"},  
3167: {id : '3675', geocod : '3127008', nome : "Fronteira"},  
3168: {id : '3035', geocod : '3127057', nome : "Fronteira dos Vales"},  
3169: {id : '2947', geocod : '3127073', nome : "Fruta de Leite"},  
3170: {id : '3676', geocod : '3127107', nome : "Frutal"},  
3171: {id : '6870', geocod : '3162104', nome : "Funchal"},  
3172: {id : '2755', geocod : '3127206', nome : "Funilândia"},  
3173: {id : '6872', geocod : '3140001', nome : "Furquim"},  
3174: {id : '6881', geocod : '3105608', nome : "Galego"},  
3175: {id : '6882', geocod : '3170750', nome : "Galena"},  
3176: {id : '2990', geocod : '3127305', nome : "Galiléia"},  
3177: {id : '6884', geocod : '3100500', nome : "Gama"},  
3178: {id : '2155', geocod : '3127339', nome : "Gameleiras"},  
3179: {id : '6891', geocod : '3170404', nome : "Garapuava"},  
3180: {id : '6899', geocod : '3169208', nome : "Gavião"},  
3181: {id : '6910', geocod : '3136009', nome : "Giru"},  
3182: {id : '2877', geocod : '3127354', nome : "Glaucilândia"},  
3183: {id : '6911', geocod : '3146107', nome : "Glaura"},  
3184: {id : '6916', geocod : '3115300', nome : "Glória de Cataguases"},  
3185: {id : '6913', geocod : '3158201', nome : "Glucínio"},  
3186: {id : '6917', geocod : '3127701', nome : "Goiabal"},  
3187: {id : '2998', geocod : '3127370', nome : "Goiabeira"},  
3188: {id : '4167', geocod : '3127388', nome : "Goianá"},  
3189: {id : '6920', geocod : '3112406', nome : "Goianases"},  
3190: {id : '3985', geocod : '3127404', nome : "Gonçalves"},  
3191: {id : '2887', geocod : '3127503', nome : "Gonzaga"},  
3192: {id : '6924', geocod : '3122603', nome : "Gororós"},  
3193: {id : '6925', geocod : '3152204', nome : "Gorutuba"},  
3194: {id : '2859', geocod : '3127602', nome : "Gouveia"},  
3195: {id : '2979', geocod : '3127701', nome : "Governador Valadares"},  
3196: {id : '6932', geocod : '3102605', nome : "Gramínea"},  
3197: {id : '6933', geocod : '3100302', nome : "Granada"},  
3198: {id : '6934', geocod : '3116100', nome : "Granjas do Norte"},  
3199: {id : '2943', geocod : '3127800', nome : "Grão Mogol"},  
3200: {id : '6937', geocod : '3135506', nome : "Grota"},  
3201: {id : '2676', geocod : '3127909', nome : "Grupiara"},  
3202: {id : '6945', geocod : '3170800', nome : "Guaicuí"},  
3203: {id : '6946', geocod : '3147204', nome : "Guaipava"},  
3204: {id : '2882', geocod : '3128006', nome : "Guanhães"},  
3205: {id : '4250', geocod : '3128105', nome : "Guapé"},  
3206: {id : '4342', geocod : '3128204', nome : "Guaraciaba"},  
3207: {id : '2876', geocod : '3128253', nome : "Guaraciama"},  
3208: {id : '3913', geocod : '3128303', nome : "Guaranésia"},  
3209: {id : '4180', geocod : '3128402', nome : "Guarani"},  
3210: {id : '6964', geocod : '3135803', nome : "Guaranilândia"},  
3211: {id : '4168', geocod : '3128501', nome : "Guarará"},  
3212: {id : '6970', geocod : '3132701', nome : "Guarataia"},  
3213: {id : '6974', geocod : '3162104', nome : "Guarda dos Ferreiros"},  
3214: {id : '2694', geocod : '3128600', nome : "Guarda-Mor"},  
3215: {id : '6975', geocod : '3164704', nome : "Guardinha"},  
3216: {id : '6985', geocod : '3118205', nome : "Guaxima"},  
3217: {id : '3915', geocod : '3128709', nome : "Guaxupé"},  
3218: {id : '4224', geocod : '3128808', nome : "Guidoval"},  
3219: {id : '2687', geocod : '3128907', nome : "Guimarânia"},  
3220: {id : '6993', geocod : '3121605', nome : "Guinda"},  
3221: {id : '4230', geocod : '3129004', nome : "Guiricema"},  
3222: {id : '2552', geocod : '3129103', nome : "Gurinhatã"},  
3223: {id : '4018', geocod : '3129202', nome : "Heliodora"},  
3224: {id : '7002', geocod : '3103009', nome : "Hematita"},  
3225: {id : '7007', geocod : '3113206', nome : "Hermilo Alves"},  
3226: {id : '7012', geocod : '3111101', nome : "Honorópolis"},  
3227: {id : '2793', geocod : '3129301', nome : "Iapu"},  
3228: {id : '4144', geocod : '3129400', nome : "Ibertioga"},  
3229: {id : '2669', geocod : '3129509', nome : "Ibiá"},  
3230: {id : '2777', geocod : '3129608', nome : "Ibiaí"},  
3231: {id : '2161', geocod : '3129657', nome : "Ibiracatu"},  
3232: {id : '3722', geocod : '3129707', nome : "Ibiraci"},  
3233: {id : '7038', geocod : '3166709', nome : "Ibiranhém"},  
3234: {id : '4294', geocod : '3129806', nome : "Ibirité"},  
3235: {id : '7044', geocod : '3140506', nome : "Ibitira"},  
3236: {id : '3890', geocod : '3129905', nome : "Ibitiúra de Minas"},  
3237: {id : '4105', geocod : '3130002', nome : "Ibituruna"},  
3238: {id : '2937', geocod : '3130051', nome : "Icaraí de Minas"},  
3239: {id : '4307', geocod : '3130101', nome : "Igarapé"},  
3240: {id : '2744', geocod : '3130200', nome : "Igaratinga"},  
3241: {id : '4258', geocod : '3130309', nome : "Iguatama"},  
3242: {id : '4102', geocod : '3130408', nome : "Ijaci"},  
3243: {id : '7091', geocod : '3161007', nome : "Ilhéus do Prata"},  
3244: {id : '4253', geocod : '3130507', nome : "Ilicínea"},  
3245: {id : '2956', geocod : '3130556', nome : "Imbé de Minas"},  
3246: {id : '7093', geocod : '3144003', nome : "Imbiruçu"},  
3247: {id : '3892', geocod : '3130606', nome : "Inconfidentes"},  
3248: {id : '2195', geocod : '3130655', nome : "Indaiabira"},  
3249: {id : '7099', geocod : '3154309', nome : "Independência"},  
3250: {id : '2661', geocod : '3130705', nome : "Indianópolis"},  
3251: {id : '4099', geocod : '3130804', nome : "Ingaí"},  
3252: {id : '7107', geocod : '3121605', nome : "Inhaí"},  
3253: {id : '2798', geocod : '3130903', nome : "Inhapim"},  
3254: {id : '2702', geocod : '3131000', nome : "Inhaúma"},  
3255: {id : '2767', geocod : '3131109', nome : "Inimutaba"},  
3256: {id : '2789', geocod : '3131158', nome : "Ipaba"},  
3257: {id : '2981', geocod : '3131208', nome : "Ipanema"},  
3258: {id : '2812', geocod : '3131307', nome : "Ipatinga"},  
3259: {id : '2562', geocod : '3131406', nome : "Ipiaçu"},  
3260: {id : '7127', geocod : '3131703', nome : "Ipoema"},  
3261: {id : '3898', geocod : '3131505', nome : "Ipuiúna"},  
3262: {id : '2656', geocod : '3131604', nome : "Iraí de Minas"},  
3263: {id : '2839', geocod : '3131703', nome : "Itabira"},  
3264: {id : '3015', geocod : '3131802', nome : "Itabirinha de Mantena"},  
3265: {id : '4329', geocod : '3131901', nome : "Itabirito"},  
3266: {id : '7154', geocod : '3159308', nome : "Itaboca"},  
3267: {id : '2880', geocod : '3132008', nome : "Itacambira"},  
3268: {id : '2163', geocod : '3132107', nome : "Itacarambi"},  
3269: {id : '7158', geocod : '3114402', nome : "Itaci"},  
3270: {id : '7160', geocod : '3117504', nome : "Itacolomi"},  
3271: {id : '4291', geocod : '3132206', nome : "Itaguara"},  
3272: {id : '7168', geocod : '3109709', nome : "Itaim"},  
3273: {id : '3026', geocod : '3132305', nome : "Itaipé"},  
3274: {id : '4014', geocod : '3132404', nome : "Itajubá"},  
3275: {id : '7186', geocod : '3130903', nome : "Itajutiba"},  
3276: {id : '2893', geocod : '3132503', nome : "Itamarandiba"},  
3277: {id : '7188', geocod : '3101003', nome : "Itamarati"},  
3278: {id : '4219', geocod : '3132602', nome : "Itamarati de Minas"},  
3279: {id : '2973', geocod : '3132701', nome : "Itambacuri"},  
3280: {id : '2835', geocod : '3132800', nome : "Itambé do Mato Dentro"},  
3281: {id : '7192', geocod : '3124302', nome : "Itamirim"},  
3282: {id : '3912', geocod : '3132909', nome : "Itamogi"},  
3283: {id : '4067', geocod : '3133006', nome : "Itamonte"},  
3284: {id : '7194', geocod : '3143906', nome : "Itamuri"},  
3285: {id : '4066', geocod : '3133105', nome : "Itanhandu"},  
3286: {id : '2986', geocod : '3133204', nome : "Itanhomi"},  
3287: {id : '3070', geocod : '3133303', nome : "Itaobim"},  
3288: {id : '2550', geocod : '3133402', nome : "Itapagipe"},  
3289: {id : '7200', geocod : '3102407', nome : "Itapanhoacanga"},  
3290: {id : '4267', geocod : '3133501', nome : "Itapecerica"},  
3291: {id : '3881', geocod : '3133600', nome : "Itapeva"},  
3292: {id : '7210', geocod : '3156601', nome : "Itapiru"},  
3293: {id : '7211', geocod : '3146701', nome : "Itapiruçu"},  
3294: {id : '4303', geocod : '3133709', nome : "Itatiaiuçu"},  
3295: {id : '3953', geocod : '3133758', nome : "Itaú de Minas"},  
3296: {id : '4282', geocod : '3133808', nome : "Itaúna"},  
3297: {id : '7226', geocod : '3158003', nome : "Itauninha"},  
3298: {id : '4327', geocod : '3133907', nome : "Itaverava"},  
3299: {id : '7233', geocod : '3172202', nome : "Itererê"},  
3300: {id : '3067', geocod : '3134004', nome : "Itinga"},  
3301: {id : '7234', geocod : '3103405', nome : "Itira"},  
3302: {id : '2991', geocod : '3134103', nome : "Itueta"},  
3303: {id : '7239', geocod : '3162906', nome : "Ituí"},  
3304: {id : '2544', geocod : '3134202', nome : "Ituiutaba"},  
3305: {id : '4100', geocod : '3134301', nome : "Itumirim"},  
3306: {id : '2548', geocod : '3134400', nome : "Iturama"},  
3307: {id : '4104', geocod : '3134509', nome : "Itutinga"},  
3308: {id : '7245', geocod : '3153202', nome : "J K"},  
3309: {id : '2829', geocod : '3134608', nome : "Jaboticatubas"},  
3310: {id : '7252', geocod : '3154200', nome : "Jacarandira"},  
3311: {id : '7254', geocod : '3134004', nome : "Jacaré"},  
3312: {id : '3073', geocod : '3134707', nome : "Jacinto"},  
3313: {id : '3916', geocod : '3134806', nome : "Jacuí"},  
3314: {id : '3887', geocod : '3134905', nome : "Jacutinga"},  
3315: {id : '2808', geocod : '3135001', nome : "Jaguaraçu"},  
3316: {id : '7279', geocod : '3134707', nome : "Jaguarão"},  
3317: {id : '7277', geocod : '3139201', nome : "Jaguaritira"},  
3318: {id : '2126', geocod : '3135050', nome : "Jaíba"},  
3319: {id : '2967', geocod : '3135076', nome : "Jampruca"},  
3320: {id : '2178', geocod : '3135100', nome : "Janaúba"},  
3321: {id : '2159', geocod : '3135209', nome : "Januária"},  
3322: {id : '4269', geocod : '3135308', nome : "Japaraíba"},  
3323: {id : '2123', geocod : '3135357', nome : "Japonvar"},  
3324: {id : '7303', geocod : '3152808', nome : "Jardinésia"},  
3325: {id : '4300', geocod : '3135407', nome : "Jeceaba"},  
3326: {id : '2913', geocod : '3135456', nome : "Jenipapo de Minas"},  
3327: {id : '4365', geocod : '3135506', nome : "Jequeri"},  
3328: {id : '2781', geocod : '3135605', nome : "Jequitaí"},  
3329: {id : '2831', geocod : '3135704', nome : "Jequitibá"},  
3330: {id : '3077', geocod : '3135803', nome : "Jequitinhonha"},  
3331: {id : '4047', geocod : '3135902', nome : "Jesuânia"},  
3332: {id : '3076', geocod : '3136009', nome : "Joaíma"},  
3333: {id : '2814', geocod : '3136108', nome : "Joanésia"},  
3334: {id : '2836', geocod : '3136207', nome : "João Monlevade"},  
3335: {id : '2696', geocod : '3136306', nome : "João Pinheiro"},  
3336: {id : '2779', geocod : '3136405', nome : "Joaquim Felício"},  
3337: {id : '2269', geocod : '3136504', nome : "Jordânia"},  
3338: {id : '2910', geocod : '3136520', nome : "José Gonçalves de Minas"},  
3339: {id : '2895', geocod : '3136553', nome : "José Raydan"},  
3340: {id : '7325', geocod : '3159100', nome : "Joselândia"},  
3341: {id : '2945', geocod : '3136579', nome : "Josenópolis"},  
3342: {id : '2785', geocod : '3136652', nome : "Juatuba"},  
3343: {id : '7345', geocod : '3118205', nome : "Jubaí"},  
3344: {id : '7349', geocod : '3161007', nome : "Juiraçu"},  
3345: {id : '4157', geocod : '3136702', nome : "Juiz de Fora"},  
3346: {id : '7353', geocod : '3139201', nome : "Junco de Minas"},  
3347: {id : '2878', geocod : '3136801', nome : "Juramento"},  
3348: {id : '7370', geocod : '3143005', nome : "Juréia"},  
3349: {id : '3924', geocod : '3136900', nome : "Juruaia"},  
3350: {id : '7367', geocod : '3154903', nome : "Jurumirim"},  
3351: {id : '7374', geocod : '3154606', nome : "Justinópolis"},  
3352: {id : '2169', geocod : '3136959', nome : "Juvenília"},  
3353: {id : '7383', geocod : '3113305', nome : "Lacerdinha"},  
3354: {id : '3024', geocod : '3137007', nome : "Ladainha"},  
3355: {id : '2688', geocod : '3137106', nome : "Lagamar"},  
3356: {id : '7391', geocod : '3118908', nome : "Lagoa Bonita"},  
3357: {id : '4270', geocod : '3137205', nome : "Lagoa da Prata"},  
3358: {id : '2778', geocod : '3137304', nome : "Lagoa dos Patos"},  
3359: {id : '4288', geocod : '3137403', nome : "Lagoa Dourada"},  
3360: {id : '2690', geocod : '3137502', nome : "Lagoa Formosa"},  
3361: {id : '2693', geocod : '3137536', nome : "Lagoa Grande"},  
3362: {id : '2825', geocod : '3137601', nome : "Lagoa Santa"},  
3363: {id : '7418', geocod : '3137601', nome : "Lagoinha de Fora"},  
3364: {id : '4428', geocod : '3137700', nome : "Lajinha"},  
3365: {id : '4046', geocod : '3137809', nome : "Lambari"},  
3366: {id : '4332', geocod : '3137908', nome : "Lamim"},  
3367: {id : '7436', geocod : '3133501', nome : "Lamounier"},  
3368: {id : '7438', geocod : '3137601', nome : "Lapinha"},  
3369: {id : '4227', geocod : '3138005', nome : "Laranjal"},  
3370: {id : '7442', geocod : '3110301', nome : "Laranjeiras de Calda"},  
3371: {id : '2765', geocod : '3138104', nome : "Lassance"},  
3372: {id : '4098', geocod : '3138203', nome : "Lavras"},  
3373: {id : '2746', geocod : '3138302', nome : "Leandro Ferreira"},  
3374: {id : '7449', geocod : '3106507', nome : "Lelivéldia"},  
3375: {id : '2907', geocod : '3138351', nome : "Leme do Prado"},  
3376: {id : '4216', geocod : '3138401', nome : "Leopoldina"},  
3377: {id : '7451', geocod : '3135209', nome : "Levinópolis"},  
3378: {id : '4083', geocod : '3138500', nome : "Liberdade"},  
3379: {id : '4139', geocod : '3138609', nome : "Lima Duarte"},  
3380: {id : '7454', geocod : '3139607', nome : "Limeira de Mantena"},  
3381: {id : '7455', geocod : '3137502', nome : "Limeira de Minas"},  
3382: {id : '2497', geocod : '3138625', nome : "Limeira do Oeste"},  
3383: {id : '7467', geocod : '3118007', nome : "Lobo Leite"},  
3384: {id : '2122', geocod : '3138658', nome : "Lontra"},  
3385: {id : '7473', geocod : '3132404', nome : "Lourenço Velho"},  
3386: {id : '7481', geocod : '3145307', nome : "Lufa"},  
3387: {id : '4348', geocod : '3138674', nome : "Luisburgo"},  
3388: {id : '2938', geocod : '3138682', nome : "Luislândia"},  
3389: {id : '7482', geocod : '3118809', nome : "Luiz Pires de Minas"},  
3390: {id : '7483', geocod : '3136306', nome : "Luizlândia do Oeste"},  
3391: {id : '4094', geocod : '3138708', nome : "Luminárias"},  
3392: {id : '7485', geocod : '3108909', nome : "Luminosa"},  
3393: {id : '2728', geocod : '3138807', nome : "Luz"},  
3394: {id : '7492', geocod : '3108008', nome : "Macaia"},  
3395: {id : '3048', geocod : '3138906', nome : "Machacalis"},  
3396: {id : '4031', geocod : '3139003', nome : "Machado"},  
3397: {id : '7497', geocod : '3101003', nome : "Machado Mineiro"},  
3398: {id : '7499', geocod : '3134301', nome : "Macuco de Minas"},  
3399: {id : '3707', geocod : '3139102', nome : "Madre de Deus de Minas"},  
3400: {id : '7759', geocod : '3164506', nome : "Mãe dos Homens"},  
3401: {id : '7506', geocod : '3102308', nome : "Major Ezequiel"},  
3402: {id : '7508', geocod : '3148004', nome : "Major Porto"},  
3403: {id : '2722', geocod : '3139201', nome : "Malacacheta"},  
3404: {id : '2189', geocod : '3139250', nome : "Mamonas"},  
3405: {id : '2153', geocod : '3139300', nome : "Manga"},  
3406: {id : '4349', geocod : '3139409', nome : "Manhuaçu"},  
3407: {id : '4388', geocod : '3139508', nome : "Manhumirim"},  
3408: {id : '3017', geocod : '3139607', nome : "Mantena"},  
3409: {id : '7533', geocod : '3160702', nome : "Mantiqueira"},  
3410: {id : '7534', geocod : '3105608', nome : "Mantiqueira do Palmital"},  
3411: {id : '4164', geocod : '3139805', nome : "Mar de Espanha"},  
3412: {id : '7542', geocod : '3113008', nome : "Marambainha"},  
3413: {id : '7544', geocod : '3100906', nome : "Maranhão"},  
3414: {id : '2753', geocod : '3139706', nome : "Maravilhas"},  
3415: {id : '4016', geocod : '3139904', nome : "Maria da Fé"},  
3416: {id : '4344', geocod : '3140001', nome : "Mariana"},  
3417: {id : '2718', geocod : '3140100', nome : "Marilac"},  
3418: {id : '7570', geocod : '3133501', nome : "Marilândia"},  
3419: {id : '4309', geocod : '3140159', nome : "Mário Campos"},  
3420: {id : '4211', geocod : '3140209', nome : "Maripá de Minas"},  
3421: {id : '2806', geocod : '3140308', nome : "Marliéria"},  
3422: {id : '4022', geocod : '3140407', nome : "Marmelópolis"},  
3423: {id : '7588', geocod : '3170206', nome : "Martinésia"},  
3424: {id : '2741', geocod : '3140506', nome : "Martinho Campos"},  
3425: {id : '7586', geocod : '3137205', nome : "Martins Guimarães"},  
3426: {id : '4394', geocod : '3140530', nome : "Martins Soares"},  
3427: {id : '2270', geocod : '3140555', nome : "Mata Verde"},  
3428: {id : '2867', geocod : '3140605', nome : "Materlândia"},  
3429: {id : '2784', geocod : '3140704', nome : "Mateus Leme"},  
3430: {id : '2965', geocod : '3171501', nome : "Mathias Lobato"},  
3431: {id : '4158', geocod : '3140803', nome : "Matias Barbosa"},  
3432: {id : '2152', geocod : '3140852', nome : "Matias Cardoso"},  
3433: {id : '4320', geocod : '3140902', nome : "Matipó"},  
3434: {id : '2185', geocod : '3141009', nome : "Mato Verde"},  
3435: {id : '2707', geocod : '3141108', nome : "Matozinhos"},  
3436: {id : '2732', geocod : '3141207', nome : "Matutina"},  
3437: {id : '2663', geocod : '3141306', nome : "Medeiros"},  
3438: {id : '3075', geocod : '3141405', nome : "Medina"},  
3439: {id : '7616', geocod : '3124104', nome : "Melo Viana"},  
3440: {id : '7617', geocod : '3121605', nome : "Mendanha"},  
3441: {id : '2977', geocod : '3141504', nome : "Mendes Pimentel"},  
3442: {id : '7618', geocod : '3169703', nome : "Mendonça"},  
3443: {id : '4176', geocod : '3141603', nome : "Mercês"},  
3444: {id : '7620', geocod : '3165008', nome : "Mercês de Água Limpa"},  
3445: {id : '2815', geocod : '3141702', nome : "Mesquita"},  
3446: {id : '7623', geocod : '3156700', nome : "Mestre Caetano"},  
3447: {id : '7624', geocod : '3146107', nome : "Miguel Burnier"},  
3448: {id : '7626', geocod : '3143203', nome : "Milagre"},  
3449: {id : '7627', geocod : '3167103', nome : "Milho Verde"},  
3450: {id : '2908', geocod : '3141801', nome : "Minas Novas"},  
3451: {id : '4097', geocod : '3141900', nome : "Minduri"},  
3452: {id : '2941', geocod : '3142007', nome : "Mirabela"},  
3453: {id : '4380', geocod : '3142106', nome : "Miradouro"},  
3454: {id : '7638', geocod : '3169901', nome : "Miragaia"},  
3455: {id : '4229', geocod : '3142205', nome : "Miraí"},  
3456: {id : '7642', geocod : '3143302', nome : "Miralta"},  
3457: {id : '7650', geocod : '3107208', nome : "Mirantão"},  
3458: {id : '7651', geocod : '3170206', nome : "Miraporanga"},  
3459: {id : '2162', geocod : '3142254', nome : "Miravânia"},  
3460: {id : '7656', geocod : '3102100', nome : "Missionário"},  
3461: {id : '7658', geocod : '3141108', nome : "Mocambeiro"},  
3462: {id : '7659', geocod : '3152204', nome : "Mocambinho"},  
3463: {id : '4306', geocod : '3142304', nome : "Moeda"},  
3464: {id : '2735', geocod : '3142403', nome : "Moema"},  
3465: {id : '7670', geocod : '3137502', nome : "Monjolinho de Minas"},  
3466: {id : '2772', geocod : '3142502', nome : "Monjolos"},  
3467: {id : '7673', geocod : '3140001', nome : "Monsenhor Horta"},  
3468: {id : '7674', geocod : '3133907', nome : "Monsenhor Isidro"},  
3469: {id : '7675', geocod : '3116605', nome : "Monsenhor João Alexandre"},  
3470: {id : '4043', geocod : '3142601', nome : "Monsenhor Paulo"},  
3471: {id : '2166', geocod : '3142700', nome : "Montalvânia"},  
3472: {id : '2593', geocod : '3142809', nome : "Monte Alegre de Minas"},  
3473: {id : '2188', geocod : '3142908', nome : "Monte Azul"},  
3474: {id : '3930', geocod : '3143005', nome : "Monte Belo"},  
3475: {id : '2681', geocod : '3143104', nome : "Monte Carmelo"},  
3476: {id : '7690', geocod : '3161502', nome : "Monte Celeste"},  
3477: {id : '3032', geocod : '3143153', nome : "Monte Formoso"},  
3478: {id : '7699', geocod : '3136959', nome : "Monte Rei"},  
3479: {id : '3911', geocod : '3143203', nome : "Monte Santo de Minas"},  
3480: {id : '3886', geocod : '3143401', nome : "Monte Sião"},  
3481: {id : '7703', geocod : '3110509', nome : "Monte Verde"},  
3482: {id : '2948', geocod : '3143302', nome : "Montes Claros"},  
3483: {id : '2191', geocod : '3143450', nome : "Montezuma"},  
3484: {id : '2760', geocod : '3143500', nome : "Morada Nova de Minas"},  
3485: {id : '7717', geocod : '3161106', nome : "Morro"},  
3486: {id : '2764', geocod : '3143609', nome : "Morro da Garça"},  
3487: {id : '7728', geocod : '3145604', nome : "Morro do Ferro"},  
3488: {id : '2841', geocod : '3143708', nome : "Morro do Pilar"},  
3489: {id : '7722', geocod : '3110004', nome : "Morro Vermelho"},  
3490: {id : '7737', geocod : '3168606', nome : "Mucuri"},  
3491: {id : '7741', geocod : '3152204', nome : "Mulungu de Minas"},  
3492: {id : '7746', geocod : '3101102', nome : "Mundo Novo de Minas"},  
3493: {id : '3880', geocod : '3143807', nome : "Munhoz"},  
3494: {id : '7749', geocod : '3142007', nome : "Muquém"},  
3495: {id : '4244', geocod : '3143906', nome : "Muriaé"},  
3496: {id : '2984', geocod : '3144003', nome : "Mutum"},  
3497: {id : '3925', geocod : '3144102', nome : "Muzambinho"},  
3498: {id : '2903', geocod : '3144201', nome : "Nacip Raydan"},  
3499: {id : '3045', geocod : '3144300', nome : "Nanuque"},  
3500: {id : '2800', geocod : '3144359', nome : "Naque"},  
3501: {id : '7761', geocod : '3100500', nome : "Naque Nanuque"},  
3502: {id : '2925', geocod : '3144375', nome : "Natalândia"},  
3503: {id : '4017', geocod : '3144409', nome : "Natércia"},  
3504: {id : '7770', geocod : '3144607', nome : "Nazaré de Minas"},  
3505: {id : '4107', geocod : '3144508', nome : "Nazareno"},  
3506: {id : '7772', geocod : '3162807', nome : "Nelson de Sena"},  
3507: {id : '7774', geocod : '3133501', nome : "Neolândia"},  
3508: {id : '4055', geocod : '3144607', nome : "Nepomuceno"},  
3509: {id : '7777', geocod : '3139300', nome : "Nhandutiba"},  
3510: {id : '7779', geocod : '3154309', nome : "Nicolândia"},  
3511: {id : '2257', geocod : '3144656', nome : "Ninheira"},  
3512: {id : '3019', geocod : '3144672', nome : "Nova Belém"},  
3513: {id : '2838', geocod : '3144706', nome : "Nova Era"},  
3514: {id : '7821', geocod : '3143302', nome : "Nova Esperança"},  
3515: {id : '2801', geocod : '3144805', nome : "Nova Lima"},  
3516: {id : '7833', geocod : '3157005', nome : "Nova Matrona"},  
3517: {id : '7835', geocod : '3135357', nome : "Nova Minda"},  
3518: {id : '2971', geocod : '3144904', nome : "Nova Módica"},  
3519: {id : '2652', geocod : '3145000', nome : "Nova Ponte"},  
3520: {id : '2179', geocod : '3145059', nome : "Nova Porteirinha"},  
3521: {id : '3927', geocod : '3145109', nome : "Nova Resende"},  
3522: {id : '2745', geocod : '3145208', nome : "Nova Serrana"},  
3523: {id : '2828', geocod : '3136603', nome : "Nova União"},  
3524: {id : '7847', geocod : '3145307', nome : "Novilhona"},  
3525: {id : '3025', geocod : '3145307', nome : "Novo Cruzeiro"},  
3526: {id : '7854', geocod : '3104700', nome : "Novo Horizonte - Ataléia"},  
3527: {id : '7855', geocod : '3135050', nome : "Novo Horizonte - Jaíba"},  
3528: {id : '3030', geocod : '3145356', nome : "Novo Oriente de Minas"},  
3529: {id : '2954', geocod : '3145372', nome : "Novorizonte"},  
3530: {id : '7865', geocod : '3144003', nome : "Ocidente"},  
3531: {id : '4138', geocod : '3145406', nome : "Olaria"},  
3532: {id : '7871', geocod : '3151008', nome : "Olegário Maciel"},  
3533: {id : '7875', geocod : '3136306', nome : "Olhos d'Água do Oeste"},  
3534: {id : '2873', geocod : '3145455', nome : "Olhos-d'Água"},  
3535: {id : '7881', geocod : '3162401', nome : "Olímpio Campos"},  
3536: {id : '4025', geocod : '3145505', nome : "Olímpio Noronha"},  
3537: {id : '4275', geocod : '3145604', nome : "Oliveira"},  
3538: {id : '4172', geocod : '3145703', nome : "Oliveira Fortes"},  
3539: {id : '2749', geocod : '3145802', nome : "Onça de Pitangui"},  
3540: {id : '4370', geocod : '3145851', nome : "Oratórios"},  
3541: {id : '4319', geocod : '3145877', nome : "Orizânia"},  
3542: {id : '4326', geocod : '3145901', nome : "Ouro Branco"},  
3543: {id : '3894', geocod : '3146008', nome : "Ouro Fino"},  
3544: {id : '4343', geocod : '3146107', nome : "Ouro Preto"},  
3545: {id : '3021', geocod : '3146206', nome : "Ouro Verde de Minas"},  
3546: {id : '7907', geocod : '3152204', nome : "Paciência"},  
3547: {id : '7911', geocod : '3105608', nome : "Padre Brito"},  
3548: {id : '2946', geocod : '3146255', nome : "Padre Carvalho"},  
3549: {id : '7913', geocod : '3102506', nome : "Padre Felisberto"},  
3550: {id : '7914', geocod : '3140902', nome : "Padre Fialho"},  
3551: {id : '7916', geocod : '3132503', nome : "Padre João Afonso"},  
3552: {id : '7917', geocod : '3153509', nome : "Padre Júlio Maria"},  
3553: {id : '3029', geocod : '3146305', nome : "Padre Paraíso"},  
3554: {id : '7920', geocod : '3155702', nome : "Padre Pinto"},  
3555: {id : '7922', geocod : '3140001', nome : "Padre Viegas"},  
3556: {id : '2154', geocod : '3146552', nome : "Pai Pedro"},  
3557: {id : '2740', geocod : '3146404', nome : "Paineiras"},  
3558: {id : '4257', geocod : '3146503', nome : "Pains"},  
3559: {id : '7928', geocod : '3151701', nome : "Paiolinho"},  
3560: {id : '4173', geocod : '3146602', nome : "Paiva"},  
3561: {id : '4242', geocod : '3146701', nome : "Palma"},  
3562: {id : '7944', geocod : '3108404', nome : "Palmeiral"},  
3563: {id : '7945', geocod : '3139409', nome : "Palmeiras"},  
3564: {id : '7948', geocod : '3170404', nome : "Palmeirinha"},  
3565: {id : '7952', geocod : '3113206', nome : "Palmital dos Carvalhos"},  
3566: {id : '3081', geocod : '3146750', nome : "Palmópolis"},  
3567: {id : '7958', geocod : '3143302', nome : "Panorâmica"},  
3568: {id : '8301', geocod : '3124500', nome : "Pântano"},  
3569: {id : '2751', geocod : '3146909', nome : "Papagaios"},  
3570: {id : '2750', geocod : '3147105', nome : "Pará de Minas"},  
3571: {id : '2695', geocod : '3147006', nome : "Paracatu"},  
3572: {id : '4035', geocod : '3147204', nome : "Paraguaçu"},  
3573: {id : '7965', geocod : '3110202', nome : "Paraguai"},  
3574: {id : '7984', geocod : '3150307', nome : "Paraíso da Piedade"},  
3575: {id : '7983', geocod : '3159407', nome : "Paraíso Garcia"},  
3576: {id : '3986', geocod : '3147303', nome : "Paraisópolis"},  
3577: {id : '2710', geocod : '3147402', nome : "Paraopeba"},  
3578: {id : '7975', geocod : '3155900', nome : "Parapeúna"},  
3579: {id : '7987', geocod : '3109402', nome : "Paredão de Minas"},  
3580: {id : '7991', geocod : '3129806', nome : "Parque Durval de Barros"},  
3581: {id : '7992', geocod : '3118601', nome : "Parque Industrial"},  
3582: {id : '7997', geocod : '3107802', nome : "Passa Dez"},  
3583: {id : '4065', geocod : '3147600', nome : "Passa Quatro"},  
3584: {id : '4284', geocod : '3147709', nome : "Passa Tempo"},  
3585: {id : '2845', geocod : '3147501', nome : "Passabém"},  
3586: {id : '8003', geocod : '3140001', nome : "Passagem de Mariana"},  
3587: {id : '4087', geocod : '3147808', nome : "Passa-Vinte"},  
3588: {id : '3956', geocod : '3147907', nome : "Passos"},  
3589: {id : '2918', geocod : '3147956', nome : "Patis"},  
3590: {id : '2691', geocod : '3148004', nome : "Patos de Minas"},  
3591: {id : '8030', geocod : '3152808', nome : "Patrimônio"},  
3592: {id : '2666', geocod : '3148103', nome : "Patrocínio"},  
3593: {id : '8031', geocod : '3113404', nome : "Patrocínio de Caratinga"},  
3594: {id : '4245', geocod : '3148202', nome : "Patrocínio do Muriaé"},  
3595: {id : '4354', geocod : '3148301', nome : "Paula Cândido"},  
3596: {id : '2883', geocod : '3148400', nome : "Paulistas"},  
3597: {id : '3031', geocod : '3148509', nome : "Pavão"},  
3598: {id : '8304', geocod : '3147600', nome : "Pé do Morro"},  
3599: {id : '2889', geocod : '3148608', nome : "Peçanha"},  
3600: {id : '3078', geocod : '3148707', nome : "Pedra Azul"},  
3601: {id : '4312', geocod : '3148756', nome : "Pedra Bonita"},  
3602: {id : '8047', geocod : '3149952', nome : "Pedra Corrida"},  
3603: {id : '4363', geocod : '3148806', nome : "Pedra do Anta"},  
3604: {id : '4271', geocod : '3148905', nome : "Pedra do Indaiá"},  
3605: {id : '8051', geocod : '3113206', nome : "Pedra do Sino"},  
3606: {id : '4313', geocod : '3149002', nome : "Pedra Dourada"},  
3607: {id : '8048', geocod : '3101706', nome : "Pedra Grande"},  
3608: {id : '8049', geocod : '3156007', nome : "Pedra Menina"},  
3609: {id : '4019', geocod : '3149101', nome : "Pedralva"},  
3610: {id : '2158', geocod : '3149150', nome : "Pedras de Maria da Cruz"},  
3611: {id : '2654', geocod : '3149200', nome : "Pedrinópolis"},  
3612: {id : '2823', geocod : '3149309', nome : "Pedro Leopoldo"},  
3613: {id : '8062', geocod : '3167103', nome : "Pedro Lessa"},  
3614: {id : '4142', geocod : '3149408', nome : "Pedro Teixeira"},  
3615: {id : '8064', geocod : '3168606', nome : "Pedro Versiani"},  
3616: {id : '8072', geocod : '3110004', nome : "Penedia"},  
3617: {id : '8075', geocod : '3132503', nome : "Penha de França"},  
3618: {id : '8076', geocod : '3101102', nome : "Penha do Capim"},  
3619: {id : '8077', geocod : '3127701', nome : "Penha do Cassiano"},  
3620: {id : '8078', geocod : '3118403', nome : "Penha do Norte"},  
3621: {id : '8074', geocod : '3116209', nome : "Penha Longa"},  
3622: {id : '4163', geocod : '3149507', nome : "Pequeri"},  
3623: {id : '2752', geocod : '3149606', nome : "Pequi"},  
3624: {id : '2742', geocod : '3149705', nome : "Perdigão"},  
3625: {id : '8081', geocod : '3159803', nome : "Perdilândia"},  
3626: {id : '2657', geocod : '3149804', nome : "Perdizes"},  
3627: {id : '4057', geocod : '3149903', nome : "Perdões"},  
3628: {id : '8083', geocod : '3121407', nome : "Pereirinhas"},  
3629: {id : '2850', geocod : '3149952', nome : "Periquito"},  
3630: {id : '8087', geocod : '3106309', nome : "Perpétuo Socorro"},  
3631: {id : '2969', geocod : '3150000', nome : "Pescador"},  
3632: {id : '8094', geocod : '3145109', nome : "Petúnia"},  
3633: {id : '8096', geocod : '3138401', nome : "Piacatuba"},  
3634: {id : '8173', geocod : '3159209', nome : "Pião"},  
3635: {id : '4160', geocod : '3150109', nome : "Piau"},  
3636: {id : '8098', geocod : '3104502', nome : "Pic Sagarana"},  
3637: {id : '2796', geocod : '3150158', nome : "Piedade de Caratinga"},  
3638: {id : '4373', geocod : '3150208', nome : "Piedade de Ponte Nova"},  
3639: {id : '8103', geocod : '3109006', nome : "Piedade do Paraopeba"},  
3640: {id : '3710', geocod : '3150307', nome : "Piedade do Rio Grande"},  
3641: {id : '4289', geocod : '3150406', nome : "Piedade dos Gerais"},  
3642: {id : '8104', geocod : '3148004', nome : "Pilar"},  
3643: {id : '4254', geocod : '3150505', nome : "Pimenta"},  
3644: {id : '8111', geocod : '3148004', nome : "Pindaíbas"},  
3645: {id : '2787', geocod : '3150539', nome : "Pingo-d'Água"},  
3646: {id : '8124', geocod : '3147600', nome : "Pinheirinhos"},  
3647: {id : '8127', geocod : '3105608', nome : "Pinheiro Grosso"},  
3648: {id : '8132', geocod : '3150802', nome : "Pinheiros Altos"},  
3649: {id : '8134', geocod : '3124906', nome : "Pinhotiba"},  
3650: {id : '2932', geocod : '3150570', nome : "Pintópolis"},  
3651: {id : '8138', geocod : '3139904', nome : "Pintos Negreiros"},  
3652: {id : '8145', geocod : '3103504', nome : "Piracaíba"},  
3653: {id : '4285', geocod : '3150604', nome : "Piracema"},  
3654: {id : '2555', geocod : '3150703', nome : "Pirajuba"},  
3655: {id : '4336', geocod : '3150802', nome : "Piranga"},  
3656: {id : '4011', geocod : '3150901', nome : "Piranguçu"},  
3657: {id : '4012', geocod : '3151008', nome : "Piranguinho"},  
3658: {id : '8150', geocod : '3155207', nome : "Piranguita"},  
3659: {id : '8152', geocod : '3143906', nome : "Pirapanema"},  
3660: {id : '4236', geocod : '3151107', nome : "Pirapetinga"},  
3661: {id : '2775', geocod : '3151206', nome : "Pirapora"},  
3662: {id : '4181', geocod : '3151305', nome : "Piraúba"},  
3663: {id : '8159', geocod : '3107307', nome : "Pires e Albuquerque"},  
3664: {id : '8165', geocod : '3135506', nome : "Piscamba"},  
3665: {id : '2748', geocod : '3151404', nome : "Pitangui"},  
3666: {id : '8168', geocod : '3142700', nome : "Pitarana"},  
3667: {id : '4252', geocod : '3151503', nome : "Piumhi"},  
3668: {id : '8178', geocod : '3121605', nome : "Planalto de Minas"},  
3669: {id : '3684', geocod : '3151602', nome : "Planura"},  
3670: {id : '8181', geocod : '3167707', nome : "Plautino Soares"},  
3671: {id : '8183', geocod : '3158201', nome : "Poaia"},  
3672: {id : '4029', geocod : '3151701', nome : "Poço Fundo"},  
3673: {id : '8263', geocod : '3146404', nome : "Poções de Paineiras"},  
3674: {id : '3917', geocod : '3151800', nome : "Poços de Caldas"},  
3675: {id : '2985', geocod : '3151909', nome : "Pocrane"},  
3676: {id : '2754', geocod : '3152006', nome : "Pompéu"},  
3677: {id : '8189', geocod : '3117306', nome : "Poncianos"},  
3678: {id : '8195', geocod : '3169406', nome : "Pontalete"},  
3679: {id : '8196', geocod : '3170107', nome : "Ponte Alta"},  
3680: {id : '8197', geocod : '3113305', nome : "Ponte Alta de Minas"},  
3681: {id : '8198', geocod : '3105608', nome : "Ponte Chique do Martelo"},  
3682: {id : '8199', geocod : '3105608', nome : "Ponte Cosme"},  
3683: {id : '8205', geocod : '3118809', nome : "Ponte dos Ciganos"},  
3684: {id : '8200', geocod : '3153400', nome : "Ponte Firme"},  
3685: {id : '4368', geocod : '3152105', nome : "Ponte Nova"},  
3686: {id : '8201', geocod : '3165578', nome : "Ponte Segura"},  
3687: {id : '8206', geocod : '3126109', nome : "Pontevila"},  
3688: {id : '2933', geocod : '3152131', nome : "Ponto Chique"},  
3689: {id : '8209', geocod : '3113008', nome : "Ponto do Marambaia"},  
3690: {id : '3069', geocod : '3152170', nome : "Ponto dos Volantes"},  
3691: {id : '2180', geocod : '3152204', nome : "Porteirinha"},  
3692: {id : '8216', geocod : '3136959', nome : "Porto Agrário"},  
3693: {id : '8235', geocod : '3106101', nome : "Porto das Flores"},  
3694: {id : '8238', geocod : '3111200', nome : "Porto dos Mendes"},  
3695: {id : '4341', geocod : '3152303', nome : "Porto Firme"},  
3696: {id : '3023', geocod : '3152402', nome : "Poté"},  
3697: {id : '3992', geocod : '3152501', nome : "Pouso Alegre"},  
3698: {id : '8244', geocod : '3135506', nome : "Pouso Alegre da Mata"},  
3699: {id : '4069', geocod : '3152600', nome : "Pouso Alto"},  
3700: {id : '3715', geocod : '3152709', nome : "Prados"},  
3701: {id : '2554', geocod : '3152808', nome : "Prata"},  
3702: {id : '8275', geocod : '3137700', nome : "Prata de Lajinha"},  
3703: {id : '3723', geocod : '3152907', nome : "Pratápolis"},  
3704: {id : '2670', geocod : '3153004', nome : "Pratinha"},  
3705: {id : '4338', geocod : '3153103', nome : "Presidente Bernardes"},  
3706: {id : '2769', geocod : '3153202', nome : "Presidente Juscelino"},  
3707: {id : '2858', geocod : '3153301', nome : "Presidente Kubitschek"},  
3708: {id : '2692', geocod : '3153400', nome : "Presidente Olegário"},  
3709: {id : '8282', geocod : '3113701', nome : "Presidente Pena"},  
3710: {id : '8286', geocod : '3116001', nome : "Professor Sperber"},  
3711: {id : '8291', geocod : '3138401', nome : "Providência"},  
3712: {id : '2708', geocod : '3153608', nome : "Prudente de Morais"},  
3713: {id : '8308', geocod : '3153707', nome : "Quartel de São João"},  
3714: {id : '8309', geocod : '3107802', nome : "Quartel do Sacramento"},  
3715: {id : '2738', geocod : '3153707', nome : "Quartel Geral"},  
3716: {id : '8311', geocod : '3134103', nome : "Quatituba"},  
3717: {id : '8316', geocod : '3145307', nome : "Queixada"},  
3718: {id : '4297', geocod : '3153806', nome : "Queluzito"},  
3719: {id : '8317', geocod : '3135100', nome : "Quem Quem"},  
3720: {id : '8318', geocod : '3156809', nome : "Quilombo"},  
3721: {id : '8325', geocod : '3114303', nome : "Quintinos"},  
3722: {id : '2802', geocod : '3153905', nome : "Raposos"},  
3723: {id : '4322', geocod : '3154002', nome : "Raul Soares"},  
3724: {id : '8342', geocod : '3156700', nome : "Ravena"},  
3725: {id : '8343', geocod : '3139409', nome : "Realeza"},  
3726: {id : '4217', geocod : '3154101', nome : "Recreio"},  
3727: {id : '4389', geocod : '3154150', nome : "Reduto"},  
3728: {id : '4287', geocod : '3154200', nome : "Resende Costa"},  
3729: {id : '2996', geocod : '3154309', nome : "Resplendor"},  
3730: {id : '4152', geocod : '3154408', nome : "Ressaquinha"},  
3731: {id : '8358', geocod : '3107802', nome : "Revés do Belém"},  
3732: {id : '2929', geocod : '3154457', nome : "Riachinho"},  
3733: {id : '8366', geocod : '3135209', nome : "Riacho da Cruz"},  
3734: {id : '2951', geocod : '3154507', nome : "Riacho dos Machados"},  
3735: {id : '8373', geocod : '3164209', nome : "Ribanceiras"},  
3736: {id : '2704', geocod : '3154606', nome : "Ribeirão das Neves"},  
3737: {id : '8382', geocod : '3157906', nome : "Ribeirão de São Domingos"},  
3738: {id : '4101', geocod : '3154705', nome : "Ribeirão Vermelho"},  
3739: {id : '8374', geocod : '3138401', nome : "Ribeiro Junqueira"},  
3740: {id : '8377', geocod : '3162005', nome : "Ribeiros"},  
3741: {id : '4330', geocod : '3154804', nome : "Rio Acima"},  
3742: {id : '4377', geocod : '3154903', nome : "Rio Casca"},  
3743: {id : '8432', geocod : '3162500', nome : "Rio das Mortes"},  
3744: {id : '3080', geocod : '3155108', nome : "Rio do Prado"},  
3745: {id : '4369', geocod : '3155009', nome : "Rio Doce"},  
3746: {id : '4331', geocod : '3155207', nome : "Rio Espera"},  
3747: {id : '4304', geocod : '3155306', nome : "Rio Manso"},  
3748: {id : '8412', geocod : '3155207', nome : "Rio Melo"},  
3749: {id : '4179', geocod : '3155405', nome : "Rio Novo"},  
3750: {id : '2672', geocod : '3155504', nome : "Rio Paranaíba"},  
3751: {id : '2186', geocod : '3155603', nome : "Rio Pardo de Minas"},  
3752: {id : '2818', geocod : '3155702', nome : "Rio Piracicaba"},  
3753: {id : '4178', geocod : '3155801', nome : "Rio Pomba"},  
3754: {id : '8417', geocod : '3168606', nome : "Rio Pretinho"},  
3755: {id : '4117', geocod : '3155900', nome : "Rio Preto"},  
3756: {id : '8424', geocod : '3140852', nome : "Rio Verde de Minas"},  
3757: {id : '2870', geocod : '3156007', nome : "Rio Vermelho"},  
3758: {id : '3709', geocod : '3156106', nome : "Ritápolis"},  
3759: {id : '8460', geocod : '3162906', nome : "Roça Grande"},  
3760: {id : '8462', geocod : '3110004', nome : "Roças Novas"},  
3761: {id : '4170', geocod : '3156205', nome : "Rochedo de Minas"},  
3762: {id : '8445', geocod : '3142502', nome : "Rodeador"},  
3763: {id : '4223', geocod : '3156304', nome : "Rodeiro"},  
3764: {id : '8446', geocod : '3146107', nome : "Rodrigo Silva"},  
3765: {id : '2655', geocod : '3156403', nome : "Romaria"},  
3766: {id : '4232', geocod : '3156452', nome : "Rosário da Limeira"},  
3767: {id : '8458', geocod : '3136702', nome : "Rosário de Minas"},  
3768: {id : '8459', geocod : '3152105', nome : "Rosário do Pontal"},  
3769: {id : '8454', geocod : '3144003', nome : "Roseiral"},  
3770: {id : '2952', geocod : '3156502', nome : "Rubelita"},  
3771: {id : '3082', geocod : '3156601', nome : "Rubim"},  
3772: {id : '2821', geocod : '3156700', nome : "Sabará"},  
3773: {id : '2865', geocod : '3156809', nome : "Sabinópolis"},  
3774: {id : '2662', geocod : '3156908', nome : "Sacramento"},  
3775: {id : '2955', geocod : '3157005', nome : "Salinas"},  
3776: {id : '8476', geocod : '3148103', nome : "Salitre de Minas"},  
3777: {id : '3084', geocod : '3157104', nome : "Salto da Divisa"},  
3778: {id : '2817', geocod : '3157203', nome : "Santa Bárbara"},  
3779: {id : '2803', geocod : '3157252', nome : "Santa Bárbara do Leste"},  
3780: {id : '4141', geocod : '3157278', nome : "Santa Bárbara do Monte Verde"},  
3781: {id : '4149', geocod : '3157302', nome : "Santa Bárbara do Tugúrio"},  
3782: {id : '8506', geocod : '3143005', nome : "Santa Cruz da Aparec"},  
3783: {id : '8509', geocod : '3108503', nome : "Santa Cruz de Botumirim"},  
3784: {id : '3712', geocod : '3157336', nome : "Santa Cruz de Minas"},  
3785: {id : '3071', geocod : '3157377', nome : "Santa Cruz de Salinas"},  
3786: {id : '4372', geocod : '3157401', nome : "Santa Cruz do Escalvado"},  
3787: {id : '8512', geocod : '3128709', nome : "Santa Cruz do Prata"},  
3788: {id : '8601', geocod : '3162658', nome : "Santa da Pedra"},  
3789: {id : '8516', geocod : '3113404', nome : "Santa Efigênia de Caratinga"},  
3790: {id : '2898', geocod : '3157500', nome : "Santa Efigênia de Minas"},  
3791: {id : '2931', geocod : '3157609', nome : "Santa Fé de Minas"},  
3792: {id : '8524', geocod : '3158904', nome : "Santa Filomena"},  
3793: {id : '3049', geocod : '3157658', nome : "Santa Helena de Minas"},  
3794: {id : '2653', geocod : '3157708', nome : "Santa Juliana"},  
3795: {id : '2822', geocod : '3157807', nome : "Santa Luzia"},  
3796: {id : '8545', geocod : '3136306', nome : "Santa Luzia da Serra"},  
3797: {id : '8546', geocod : '3113404', nome : "Santa Luzia de Caratinga"},  
3798: {id : '8551', geocod : '3148103', nome : "Santa Luzia dos Barros"},  
3799: {id : '4321', geocod : '3157906', nome : "Santa Margarida"},  
3800: {id : '2840', geocod : '3158003', nome : "Santa Maria de Itabira"},  
3801: {id : '8563', geocod : '3162609', nome : "Santa Maria do Baixio"},  
3802: {id : '3074', geocod : '3158102', nome : "Santa Maria do Salto"},  
3803: {id : '2719', geocod : '3158201', nome : "Santa Maria do Suaçuí"},  
3804: {id : '8573', geocod : '3124807', nome : "Santa Rita da Estrela"},  
3805: {id : '3895', geocod : '3159209', nome : "Santa Rita de Caldas"},  
3806: {id : '4140', geocod : '3159407', nome : "Santa Rita de Ibitipoca"},  
3807: {id : '4088', geocod : '3159308', nome : "Santa Rita de Jacutinga"},  
3808: {id : '2794', geocod : '3159357', nome : "Santa Rita de Minas"},  
3809: {id : '8575', geocod : '3146107', nome : "Santa Rita de Ouro Preto"},  
3810: {id : '8576', geocod : '3116100', nome : "Santa Rita do Araçua"},  
3811: {id : '8577', geocod : '3120904', nome : "Santa Rita do Cedro"},  
3812: {id : '2995', geocod : '3159506', nome : "Santa Rita do Itueto"},  
3813: {id : '8579', geocod : '3125903', nome : "Santa Rita do Rio do Peixe"},  
3814: {id : '3998', geocod : '3159605', nome : "Santa Rita do Sapucaí"},  
3815: {id : '8571', geocod : '3140001', nome : "Santa Rita Durão"},  
3816: {id : '2727', geocod : '3159704', nome : "Santa Rosa da Serra"},  
3817: {id : '8583', geocod : '3143302', nome : "Santa Rosa de Lima"},  
3818: {id : '8584', geocod : '3119302', nome : "Santa Rosa dos Dourados"},  
3819: {id : '8586', geocod : '3148608', nome : "Santa Teresa do Bonito"},  
3820: {id : '8599', geocod : '3133709', nome : "Santa Terezinha de Minas"},  
3821: {id : '2560', geocod : '3159803', nome : "Santa Vitória"},  
3822: {id : '4051', geocod : '3158300', nome : "Santana da Vargem"},  
3823: {id : '8606', geocod : '3110301', nome : "Santana de Caldas"},  
3824: {id : '4228', geocod : '3158409', nome : "Santana de Cataguases"},  
3825: {id : '8607', geocod : '3148004', nome : "Santana de Patos"},  
3826: {id : '2756', geocod : '3158508', nome : "Santana de Pirapama"},  
3827: {id : '8609', geocod : '3161007', nome : "Santana do Alfié"},  
3828: {id : '8610', geocod : '3152170', nome : "Santana do Araçuaí"},  
3829: {id : '8611', geocod : '3104601', nome : "Santana do Campestre"},  
3830: {id : '8612', geocod : '3152600', nome : "Santana do Capivari"},  
3831: {id : '4161', geocod : '3158607', nome : "Santana do Deserto"},  
3832: {id : '3706', geocod : '3158706', nome : "Santana do Garambéu"},  
3833: {id : '4265', geocod : '3158805', nome : "Santana do Jacaré"},  
3834: {id : '4396', geocod : '3158904', nome : "Santana do Manhuaçu"},  
3835: {id : '2816', geocod : '3158953', nome : "Santana do Paraíso"},  
3836: {id : '8614', geocod : '3106408', nome : "Santana do Paraopeba"},  
3837: {id : '2832', geocod : '3159001', nome : "Santana do Riacho"},  
3838: {id : '8615', geocod : '3154002', nome : "Santana do Tabuleiro"},  
3839: {id : '4324', geocod : '3159100', nome : "Santana dos Montes"},  
3840: {id : '8619', geocod : '3153400', nome : "Santiago de Minas"},  
3841: {id : '8641', geocod : '3162401', nome : "Santo Antônio da Boa"},  
3842: {id : '8643', geocod : '3125903', nome : "Santo Antônio da Fortaleza"},  
3843: {id : '8645', geocod : '3108107', nome : "Santo Antônio de Vargem Alegre"},  
3844: {id : '4273', geocod : '3159902', nome : "Santo Antônio do Amparo"},  
3845: {id : '4210', geocod : '3160009', nome : "Santo Antônio do Aventureiro"},  
3846: {id : '8651', geocod : '3170404', nome : "Santo Antônio do Boqueirão"},  
3847: {id : '8653', geocod : '3144607', nome : "Santo Antônio do Cruzeiro"},  
3848: {id : '8654', geocod : '3171402', nome : "Santo Antônio do Glória"},  
3849: {id : '4376', geocod : '3160108', nome : "Santo Antônio do Gramado"},  
3850: {id : '2864', geocod : '3160207', nome : "Santo Antônio do Itambé"},  
3851: {id : '3072', geocod : '3160306', nome : "Santo Antônio do Jacinto"},  
3852: {id : '8657', geocod : '3146107', nome : "Santo Antônio do Lei"},  
3853: {id : '4272', geocod : '3160405', nome : "Santo Antônio do Monte"},  
3854: {id : '8658', geocod : '3139201', nome : "Santo Antônio do Mucuri"},  
3855: {id : '8660', geocod : '3117504', nome : "Santo Antônio do Norte"},  
3856: {id : '8662', geocod : '3150802', nome : "Santo Antônio do Piranga"},  
3857: {id : '8663', geocod : '3127701', nome : "Santo Antônio do Pontal"},  
3858: {id : '8664', geocod : '3150307', nome : "Santo Antônio do Porto"},  
3859: {id : '2187', geocod : '3160454', nome : "Santo Antônio do Retiro"},  
3860: {id : '2842', geocod : '3160504', nome : "Santo Antônio do Rio Abaixo"},  
3861: {id : '8669', geocod : '3146107', nome : "Santo Antônio do Sal"},  
3862: {id : '8670', geocod : '3164506', nome : "Santo Antônio dos Araújos"},  
3863: {id : '8671', geocod : '3122306', nome : "Santo Antônio dos Campos"},  
3864: {id : '8639', geocod : '3113404', nome : "Santo Antônio Manhuaçu"},  
3865: {id : '8677', geocod : '3150505', nome : "Santo Hilário"},  
3866: {id : '2770', geocod : '3160603', nome : "Santo Hipólito"},  
3867: {id : '4148', geocod : '3160702', nome : "Santos Dumont"},  
3868: {id : '8814', geocod : '3146107', nome : "São Bartolomeu"},  
3869: {id : '8815', geocod : '3157807', nome : "São Benedito"},  
3870: {id : '4093', geocod : '3160801', nome : "São Bento Abade"},  
3871: {id : '8820', geocod : '3159209', nome : "São Bento de Caldas"},  
3872: {id : '8825', geocod : '3137106', nome : "São Brás de Minas"},  
3873: {id : '4299', geocod : '3160900', nome : "São Brás do Suaçuí"},  
3874: {id : '8826', geocod : '3120409', nome : "São Caetano"},  
3875: {id : '8833', geocod : '3113404', nome : "São Cândido"},  
3876: {id : '8838', geocod : '3160009', nome : "São Domingos"},  
3877: {id : '8841', geocod : '3138609', nome : "São Domingos da Boca"},  
3878: {id : '2799', geocod : '3160959', nome : "São Domingos das Dores"},  
3879: {id : '2804', geocod : '3161007', nome : "São Domingos do Prata"},  
3880: {id : '2970', geocod : '3161056', nome : "São Félix de Minas"},  
3881: {id : '2151', geocod : '3161106', nome : "São Francisco"},  
3882: {id : '4274', geocod : '3161205', nome : "São Francisco de Paula"},  
3883: {id : '2549', geocod : '3161304', nome : "São Francisco de Sales"},  
3884: {id : '4311', geocod : '3161403', nome : "São Francisco do Glória"},  
3885: {id : '8854', geocod : '3144003', nome : "São Francisco do Humaitá"},  
3886: {id : '8855', geocod : '3159506', nome : "São Francisco do Jataí"},  
3887: {id : '4356', geocod : '3161502', nome : "São Geraldo"},  
3888: {id : '8859', geocod : '3125705', nome : "São Geraldo - Felixlândia"},  
3889: {id : '2900', geocod : '3161601', nome : "São Geraldo da Piedade"},  
3890: {id : '8861', geocod : '3169505', nome : "São Geraldo de Tumiritinga"},  
3891: {id : '8862', geocod : '3162807', nome : "São Geraldo do Bagua"},  
3892: {id : '2997', geocod : '3161650', nome : "São Geraldo do Baixio"},  
3893: {id : '8863', geocod : '3125705', nome : "São Geraldo do Salto"},  
3894: {id : '8865', geocod : '3108404', nome : "São Gonçalo de Botel"},  
3895: {id : '2758', geocod : '3161700', nome : "São Gonçalo do Abaeté"},  
3896: {id : '8866', geocod : '3162500', nome : "São Gonçalo do Amara"},  
3897: {id : '8867', geocod : '3105400', nome : "São Gonçalo do Monte"},  
3898: {id : '2743', geocod : '3161809', nome : "São Gonçalo do Pará"},  
3899: {id : '2834', geocod : '3161908', nome : "São Gonçalo do Rio Abaixo"},  
3900: {id : '8868', geocod : '3167103', nome : "São Gonçalo do Rio das Pedra"},  
3901: {id : '2869', geocod : '3125507', nome : "São Gonçalo do Rio Preto"},  
3902: {id : '4041', geocod : '3162005', nome : "São Gonçalo do Sapucaí"},  
3903: {id : '2731', geocod : '3162104', nome : "São Gotardo"},  
3904: {id : '8871', geocod : '3111507', nome : "São Jerônimo dos Poços"},  
3905: {id : '3957', geocod : '3162203', nome : "São João Batista do Glória"},  
3906: {id : '8948', geocod : '3121605', nome : "São João da Chapada"},  
3907: {id : '2782', geocod : '3162252', nome : "São João da Lagoa"},  
3908: {id : '4028', geocod : '3162302', nome : "São João da Mata"},  
3909: {id : '2177', geocod : '3162401', nome : "São João da Ponte"},  
3910: {id : '8950', geocod : '3138005', nome : "São João da Sapucaia"},  
3911: {id : '8951', geocod : '3160702', nome : "São João da Serra"},  
3912: {id : '8952', geocod : '3148103', nome : "São João da Serra Negra"},  
3913: {id : '8953', geocod : '3143302', nome : "São João da Vereda"},  
3914: {id : '2164', geocod : '3162450', nome : "São João das Missões"},  
3915: {id : '3711', geocod : '3162500', nome : "São João del Rei"},  
3916: {id : '8962', geocod : '3141009', nome : "São João do Bonito"},  
3917: {id : '8963', geocod : '3113404', nome : "São João do Jacutinga"},  
3918: {id : '4347', geocod : '3162559', nome : "São João do Manhuaçu"},  
3919: {id : '3013', geocod : '3162575', nome : "São João do Manteninha"},  
3920: {id : '2851', geocod : '3162609', nome : "São João do Oriente"},  
3921: {id : '2939', geocod : '3162658', nome : "São João do Pacuí"},  
3922: {id : '2196', geocod : '3162708', nome : "São João do Paraíso"},  
3923: {id : '2884', geocod : '3162807', nome : "São João Evangelista"},  
3924: {id : '4171', geocod : '3162906', nome : "São João Nepomuceno"},  
3925: {id : '8873', geocod : '3118809', nome : "São Joaquim"},  
3926: {id : '8874', geocod : '3135209', nome : "São Joaquim - Januária"},  
3927: {id : '4310', geocod : '3162922', nome : "São Joaquim de Bicas"},  
3928: {id : '3959', geocod : '3162948', nome : "São José da Barra"},  
3929: {id : '8898', geocod : '3162104', nome : "São José da Bela Vista"},  
3930: {id : '2819', geocod : '3162955', nome : "São José da Lapa"},  
3931: {id : '8902', geocod : '3124203', nome : "São José da Pedra Me"},  
3932: {id : '2721', geocod : '3163003', nome : "São José da Safira"},  
3933: {id : '2697', geocod : '3163102', nome : "São José da Varginha"},  
3934: {id : '8909', geocod : '3101631', nome : "São José de Pouso Alegre"},  
3935: {id : '8911', geocod : '3123700', nome : "São José do Acácio"},  
3936: {id : '4013', geocod : '3163201', nome : "São José do Alegre"},  
3937: {id : '8912', geocod : '3164308', nome : "São José do Barreiro"},  
3938: {id : '8913', geocod : '3170057', nome : "São José do Batatal"},  
3939: {id : '8914', geocod : '3125705', nome : "São José do Buriti"},  
3940: {id : '3014', geocod : '3163300', nome : "São José do Divino"},  
3941: {id : '2809', geocod : '3163409', nome : "São José do Goiabal"},  
3942: {id : '8920', geocod : '3127701', nome : "São José do Itapinoã"},  
3943: {id : '8922', geocod : '3159506', nome : "São José do Itueto"},  
3944: {id : '2894', geocod : '3163508', nome : "São José do Jacuri"},  
3945: {id : '4397', geocod : '3163607', nome : "São José do Mantimento"},  
3946: {id : '8925', geocod : '3146008', nome : "São José do Mato Dentro"},  
3947: {id : '8929', geocod : '3152501', nome : "São José do Pântano"},  
3948: {id : '8926', geocod : '3109006', nome : "São José do Paraopebas"},  
3949: {id : '8935', geocod : '3171303', nome : "São José do Triunfo"},  
3950: {id : '8938', geocod : '3138609', nome : "São José dos Lopes"},  
3951: {id : '8939', geocod : '3114204', nome : "São José dos Salgado"},  
3952: {id : '4070', geocod : '3163706', nome : "São Lourenço"},  
3953: {id : '8988', geocod : '3122900', nome : "São Manoel do Guaiaçu"},  
3954: {id : '8999', geocod : '3110509', nome : "São Mateus de Minas"},  
3955: {id : '4362', geocod : '3163805', nome : "São Miguel do Anta"},  
3956: {id : '9022', geocod : '3143302', nome : "São Pedro da Garça"},  
3957: {id : '3926', geocod : '3163904', nome : "São Pedro da União"},  
3958: {id : '9024', geocod : '3149150', nome : "São Pedro das Taboca"},  
3959: {id : '9026', geocod : '3110301', nome : "São Pedro de Caldas"},  
3960: {id : '9030', geocod : '3139409', nome : "São Pedro do Avaí"},  
3961: {id : '9033', geocod : '3125952', nome : "São Pedro do Gloria"},  
3962: {id : '9035', geocod : '3135803', nome : "São Pedro do Jequitibá"},  
3963: {id : '2890', geocod : '3164100', nome : "São Pedro do Suaçuí"},  
3964: {id : '4379', geocod : '3164001', nome : "São Pedro dos Ferros"},  
3965: {id : '9042', geocod : '3162252', nome : "São Roberto de Minas"},  
3966: {id : '2934', geocod : '3164209', nome : "São Romão"},  
3967: {id : '3961', geocod : '3164308', nome : "São Roque de Minas"},  
3968: {id : '9061', geocod : '3129301', nome : "São Sebastião da Barra"},  
3969: {id : '3996', geocod : '3164407', nome : "São Sebastião da Bela Vista"},  
3970: {id : '9062', geocod : '3116100', nome : "São Sebastião da Boa Vista"},  
3971: {id : '9063', geocod : '3159902', nome : "São Sebastião da Estrela"},  
3972: {id : '9065', geocod : '3101102', nome : "São Sebastião da Vala"},  
3973: {id : '4231', geocod : '3164431', nome : "São Sebastião da Vargem Alegre"},  
3974: {id : '9066', geocod : '3162500', nome : "São Sebastião da Vitoria"},  
3975: {id : '9067', geocod : '3106309', nome : "São Sebastião de Braúnas"},  
3976: {id : '9068', geocod : '3102902', nome : "São Sebastião de Campolide"},  
3977: {id : '2957', geocod : '3164472', nome : "São Sebastião do Anta"},  
3978: {id : '9073', geocod : '3149952', nome : "São Sebastião do Baixio"},  
3979: {id : '9074', geocod : '3135076', nome : "São Sebastião do Barra"},  
3980: {id : '9075', geocod : '3157278', nome : "São Sebastião do Barreado"},  
3981: {id : '9076', geocod : '3170057', nome : "São Sebastião do Batatal"},  
3982: {id : '9077', geocod : '3117504', nome : "São Sebastião do Bonsucesso"},  
3983: {id : '9078', geocod : '3119203', nome : "São Sebastião do Bugre"},  
3984: {id : '9079', geocod : '3121407', nome : "São Sebastião do Gil"},  
3985: {id : '2896', geocod : '3164506', nome : "São Sebastião do Maranhão"},  
3986: {id : '9071', geocod : '3154002', nome : "São Sebastião do Óculo"},  
3987: {id : '4278', geocod : '3164605', nome : "São Sebastião do Oeste"},  
3988: {id : '3720', geocod : '3164704', nome : "São Sebastião do Paraíso"},  
3989: {id : '9081', geocod : '3114550', nome : "São Sebastião do Pontal"},  
3990: {id : '2846', geocod : '3164803', nome : "São Sebastião do Rio Preto"},  
3991: {id : '4068', geocod : '3164902', nome : "São Sebastião do Rio Verde"},  
3992: {id : '9083', geocod : '3139409', nome : "São Sebastião do Sacramento"},  
3993: {id : '9084', geocod : '3157401', nome : "São Sebastião do Soberbo"},  
3994: {id : '9087', geocod : '3142700', nome : "São Sebastião dos Poções"},  
3995: {id : '9088', geocod : '3134905', nome : "São Sebastião dos Robertos"},  
3996: {id : '9089', geocod : '3105608', nome : "São Sebastião dos Torres"},  
3997: {id : '9094', geocod : '3167608', nome : "São Simão do Rio Preto"},  
3998: {id : '4092', geocod : '3165206', nome : "São Thomé das Letras"},  
3999: {id : '4283', geocod : '3165008', nome : "São Tiago"},  
4000: {id : '3826', geocod : '3165107', nome : "São Tomás de Aquino"},  
4001: {id : '9101', geocod : '3105004', nome : "São Vicente"},  
4002: {id : '9105', geocod : '3154002', nome : "São Vicente da Estrela"},  
4003: {id : '4111', geocod : '3165305', nome : "São Vicente de Minas"},  
4004: {id : '9107', geocod : '3135506', nome : "São Vicente do Gramado"},  
4005: {id : '9108', geocod : '3168408', nome : "São Vicente do Rio Doce"},  
4006: {id : '9109', geocod : '3127701', nome : "São Vítor"},  
4007: {id : '8695', geocod : '3134905', nome : "Sapucaí"},  
4008: {id : '8690', geocod : '3113404', nome : "Sapucaia"},  
4009: {id : '8692', geocod : '3128006', nome : "Sapucaia de Guanhães"},  
4010: {id : '8693', geocod : '3127305', nome : "Sapucaia do Norte"},  
4011: {id : '3987', geocod : '3165404', nome : "Sapucaí-Mirim"},  
4012: {id : '8701', geocod : '3136702', nome : "Sarandira"},  
4013: {id : '2899', geocod : '3165503', nome : "Sardoá"},  
4014: {id : '4293', geocod : '3165537', nome : "Sarzedo"},  
4015: {id : '8702', geocod : '3139805', nome : "Saudade"},  
4016: {id : '4375', geocod : '3165560', nome : "Sem-Peixe"},  
4017: {id : '3883', geocod : '3165578', nome : "Senador Amaral"},  
4018: {id : '4209', geocod : '3165602', nome : "Senador Cortes"},  
4019: {id : '4339', geocod : '3165701', nome : "Senador Firmino"},  
4020: {id : '3897', geocod : '3165800', nome : "Senador José Bento"},  
4021: {id : '8717', geocod : '3119401', nome : "Senador Melo Viana"},  
4022: {id : '2872', geocod : '3165909', nome : "Senador Modestino Gonçalves"},  
4023: {id : '8718', geocod : '3121605', nome : "Senador Mourão"},  
4024: {id : '8720', geocod : '3160603', nome : "Senhora da Glória"},  
4025: {id : '8721', geocod : '3125804', nome : "Senhora da Penha"},  
4026: {id : '8722', geocod : '3105608', nome : "Senhora das Dores"},  
4027: {id : '4334', geocod : '3166006', nome : "Senhora de Oliveira"},  
4028: {id : '8723', geocod : '3131703', nome : "Senhora do Carmo"},  
4029: {id : '2849', geocod : '3166105', nome : "Senhora do Porto"},  
4030: {id : '4154', geocod : '3166204', nome : "Senhora dos Remédios"},  
4031: {id : '8724', geocod : '3115300', nome : "Sereno"},  
4032: {id : '4367', geocod : '3166303', nome : "Sericita"},  
4033: {id : '4108', geocod : '3166402', nome : "Seritinga"},  
4034: {id : '8728', geocod : '3140704', nome : "Serra Azul"},  
4035: {id : '2866', geocod : '3166501', nome : "Serra Azul de Minas"},  
4036: {id : '8729', geocod : '3109303', nome : "Serra Bonita"},  
4037: {id : '8737', geocod : '3164308', nome : "Serra da Canastra"},  
4038: {id : '2730', geocod : '3166600', nome : "Serra da Saudade"},  
4039: {id : '8739', geocod : '3116159', nome : "Serra das Araras"},  
4040: {id : '8740', geocod : '3123908', nome : "Serra do Camapuã"},  
4041: {id : '2667', geocod : '3166808', nome : "Serra do Salitre"},  
4042: {id : '3052', geocod : '3166709', nome : "Serra dos Aimorés"},  
4043: {id : '8745', geocod : '3109501', nome : "Serra dos Lemes"},  
4044: {id : '8734', geocod : '3155603', nome : "Serra Nova"},  
4045: {id : '4030', geocod : '3166907', nome : "Serrania"},  
4046: {id : '2156', geocod : '3166956', nome : "Serranópolis de Minas"},  
4047: {id : '4110', geocod : '3167004', nome : "Serranos"},  
4048: {id : '2863', geocod : '3167103', nome : "Serro"},  
4049: {id : '8755', geocod : '3108305', nome : "Sertãozinho"},  
4050: {id : '8759', geocod : '3125903', nome : "Sete Cachoeiras"},  
4051: {id : '2705', geocod : '3167202', nome : "Sete Lagoas"},  
4052: {id : '2912', geocod : '3165552', nome : "Setubinha"},  
4053: {id : '8764', geocod : '3152006', nome : "Silva Campos"},  
4054: {id : '8766', geocod : '3167202', nome : "Silva Xavier"},  
4055: {id : '8767', geocod : '3148103', nome : "Silvano"},  
4056: {id : '8768', geocod : '3105509', nome : "Silveira Carvalho"},  
4057: {id : '4182', geocod : '3167301', nome : "Silveirânia"},  
4058: {id : '8769', geocod : '3171303', nome : "Silvestre"},  
4059: {id : '3997', geocod : '3167400', nome : "Silvianópolis"},  
4060: {id : '8771', geocod : '3162401', nome : "Simão Campos"},  
4061: {id : '4156', geocod : '3167509', nome : "Simão Pereira"},  
4062: {id : '4350', geocod : '3167608', nome : "Simonésia"},  
4063: {id : '8779', geocod : '3104601', nome : "Sobral Pinto"},  
4064: {id : '2853', geocod : '3167707', nome : "Sobrália"},  
4065: {id : '4071', geocod : '3167806', nome : "Soledade de Minas"},  
4066: {id : '8788', geocod : '3121605', nome : "Sopa"},  
4067: {id : '8794', geocod : '3155306', nome : "Souza"},  
4068: {id : '8800', geocod : '3152402', nome : "Sucanga"},  
4069: {id : '9123', geocod : '3130903', nome : "Tabajara"},  
4070: {id : '9126', geocod : '3101102', nome : "Tabaúna"},  
4071: {id : '9133', geocod : '3107505', nome : "Tabuão"},  
4072: {id : '4177', geocod : '3167905', nome : "Tabuleiro"},  
4073: {id : '2183', geocod : '3168002', nome : "Taiobeiras"},  
4074: {id : '2983', geocod : '3168051', nome : "Taparuba"},  
4075: {id : '2665', geocod : '3168101', nome : "Tapira"},  
4076: {id : '2725', geocod : '3168200', nome : "Tapiraí"},  
4077: {id : '9161', geocod : '3170206', nome : "Tapuirama"},  
4078: {id : '2827', geocod : '3168309', nome : "Taquaraçu de Minas"},  
4079: {id : '9173', geocod : '3162906', nome : "Taruaçu"},  
4080: {id : '2855', geocod : '3168408', nome : "Tarumirim"},  
4081: {id : '9182', geocod : '3138401', nome : "Tebas"},  
4082: {id : '4359', geocod : '3168507', nome : "Teixeiras"},  
4083: {id : '9184', geocod : '3135209', nome : "Tejuco"},  
4084: {id : '2975', geocod : '3168606', nome : "Teófilo Otoni"},  
4085: {id : '9190', geocod : '3107307', nome : "Terra Branca"},  
4086: {id : '2810', geocod : '3168705', nome : "Timóteo"},  
4087: {id : '3713', geocod : '3168804', nome : "Tiradentes"},  
4088: {id : '2733', geocod : '3168903', nome : "Tiros"},  
4089: {id : '9211', geocod : '3129509', nome : "Tobati"},  
4090: {id : '9212', geocod : '3152204', nome : "Tocandira"},  
4091: {id : '4184', geocod : '3169000', nome : "Tocantins"},  
4092: {id : '9213', geocod : '3126505', nome : "Tocoiós de Minas"},  
4093: {id : '3990', geocod : '3169059', nome : "Tocos do Moji"},  
4094: {id : '3879', geocod : '3169109', nome : "Toledo"},  
4095: {id : '9216', geocod : '3120904', nome : "Tomás Gonzaga"},  
4096: {id : '4315', geocod : '3169208', nome : "Tombos"},  
4097: {id : '9218', geocod : '3168606', nome : "Topázio"},  
4098: {id : '9221', geocod : '3147105', nome : "Torneiros"},  
4099: {id : '9224', geocod : '3136702', nome : "Torreões"},  
4100: {id : '4049', geocod : '3169307', nome : "Três Corações"},  
4101: {id : '9248', geocod : '3106101', nome : "Três Ilhas"},  
4102: {id : '2761', geocod : '3169356', nome : "Três Marias"},  
4103: {id : '4050', geocod : '3169406', nome : "Três Pontas"},  
4104: {id : '9236', geocod : '3172103', nome : "Trimonte"},  
4105: {id : '9258', geocod : '3129004', nome : "Tuiutinga"},  
4106: {id : '2989', geocod : '3169505', nome : "Tumiritinga"},  
4107: {id : '2596', geocod : '3169604', nome : "Tupaciguara"},  
4108: {id : '9265', geocod : '3141405', nome : "Tuparecê"},  
4109: {id : '2906', geocod : '3169703', nome : "Turmalina"},  
4110: {id : '4033', geocod : '3169802', nome : "Turvolândia"},  
4111: {id : '4222', geocod : '3169901', nome : "Ubá"},  
4112: {id : '2936', geocod : '3170008', nome : "Ubaí"},  
4113: {id : '2797', geocod : '3170057', nome : "Ubaporanga"},  
4114: {id : '9275', geocod : '3169901', nome : "Ubari"},  
4115: {id : '2660', geocod : '3170107', nome : "Uberaba"},  
4116: {id : '2559', geocod : '3170206', nome : "Uberlândia"},  
4117: {id : '9288', geocod : '3106606', nome : "Umburaninha"},  
4118: {id : '3050', geocod : '3170305', nome : "Umburatiba"},  
4119: {id : '9290', geocod : '3138658', nome : "Umbuzeiro"},  
4120: {id : '2923', geocod : '3170404', nome : "Unaí"},  
4121: {id : '2498', geocod : '3170438', nome : "União de Minas"},  
4122: {id : '2927', geocod : '3170479', nome : "Uruana de Minas"},  
4123: {id : '4371', geocod : '3170503', nome : "Urucânia"},  
4124: {id : '2930', geocod : '3170529', nome : "Urucuia"},  
4125: {id : '9311', geocod : '3168408', nome : "Vai Volta"},  
4126: {id : '9324', geocod : '3152402', nome : "Valão"},  
4127: {id : '9325', geocod : '3151107', nome : "Valão Quente"},  
4128: {id : '9323', geocod : '3160603', nome : "Valo Fundo"},  
4129: {id : '2790', geocod : '3170578', nome : "Vargem Alegre"},  
4130: {id : '3960', geocod : '3170602', nome : "Vargem Bonita"},  
4131: {id : '9333', geocod : '3154408', nome : "Vargem do Amargoso"},  
4132: {id : '2194', geocod : '3170651', nome : "Vargem Grande do Rio Pardo"},  
4133: {id : '9332', geocod : '3161007', nome : "Vargem Linda"},  
4134: {id : '4045', geocod : '3170701', nome : "Varginha"},  
4135: {id : '2757', geocod : '3170750', nome : "Varjão de Minas"},  
4136: {id : '9440', geocod : '3135209', nome : "Várzea Bonita"},  
4137: {id : '2776', geocod : '3170800', nome : "Várzea da Palma"},  
4138: {id : '2124', geocod : '3170909', nome : "Varzelândia"},  
4139: {id : '9340', geocod : '3152105', nome : "Vau Açu"},  
4140: {id : '2689', geocod : '3171006', nome : "Vazante"},  
4141: {id : '9347', geocod : '3106200', nome : "Venda Nova"},  
4142: {id : '9353', geocod : '3149309', nome : "Vera Cruz de Minas"},  
4143: {id : '2125', geocod : '3171030', nome : "Verdelândia"},  
4144: {id : '9355', geocod : '3136306', nome : "Veredas"},  
4145: {id : '2904', geocod : '3171071', nome : "Veredinha"},  
4146: {id : '2558', geocod : '3171105', nome : "Veríssimo"},  
4147: {id : '9357', geocod : '3143906', nome : "Vermelho"},  
4148: {id : '4346', geocod : '3171154', nome : "Vermelho Novo"},  
4149: {id : '9358', geocod : '3154002', nome : "Vermelho Velho"},  
4150: {id : '2820', geocod : '3171204', nome : "Vespasiano"},  
4151: {id : '4355', geocod : '3171303', nome : "Viçosa"},  
4152: {id : '4381', geocod : '3171402', nome : "Vieiras"},  
4153: {id : '9367', geocod : '3104502', nome : "Vila Bom Jesus"},  
4154: {id : '9371', geocod : '3146503', nome : "Vila Costina"},  
4155: {id : '9389', geocod : '3143302', nome : "Vila Nova de Minas"},  
4156: {id : '9390', geocod : '3135100', nome : "Vila Nova dos Poções"},  
4157: {id : '9385', geocod : '3127701', nome : "Vila Nova Floresta"},  
4158: {id : '9394', geocod : '3144300', nome : "Vila Pereira"},  
4159: {id : '9414', geocod : '3129004', nome : "Vilas Boas"},  
4160: {id : '2916', geocod : '3171600', nome : "Virgem da Lapa"},  
4161: {id : '4023', geocod : '3171709', nome : "Virgínia"},  
4162: {id : '2885', geocod : '3171808', nome : "Virginópolis"},  
4163: {id : '2902', geocod : '3171907', nome : "Virgolândia"},  
4164: {id : '4225', geocod : '3172004', nome : "Visconde do Rio Branco"},  
4165: {id : '9420', geocod : '3138401', nome : "Vista Alegre"},  
4166: {id : '9429', geocod : '3102100', nome : "Vitorinos"},  
4167: {id : '4215', geocod : '3172103', nome : "Volta Grande"},  
4168: {id : '4015', geocod : '3172202', nome : "Wenceslau Braz"},  
4169: {id : '9458', geocod : '3157708', nome : "Zelândia"},  
4170: {id : '9459', geocod : '3157401', nome : "Zito Soares"}  
4171: ],  
4172: MS : [  
4173: // UF = MS  
4174: {id : '9464', geocod : '5007505', nome : "Água Boa"},  
4175: {id : '3390', geocod : '5000203', nome : "Água Clara"},  
4176: {id : '5604', geocod : '5003207', nome : "Albuquerque"},  
4177: {id : '2485', geocod : '5000252', nome : "Alcinópolis"},  
4178: {id : '5651', geocod : '5000203', nome : "Alto Sucuriú"},  
4179: {id : '3095', geocod : '5000609', nome : "Amambaí"},  
4180: {id : '5670', geocod : '5004700', nome : "Amandina"},  
4181: {id : '5681', geocod : '5003207', nome : "Amolar"},  
4182: {id : '3117', geocod : '5000708', nome : "Anastácio"},  
4183: {id : '3202', geocod : '5000807', nome : "Anaurilândia"},  
4184: {id : '3184', geocod : '5000856', nome : "Angélica"},  
4185: {id : '5696', geocod : '5002704', nome : "Anhanduí"},  
4186: {id : '3101', geocod : '5000906', nome : "Antônio João"},  
4187: {id : '3404', geocod : '5001003', nome : "Aparecida do Taboado"},  
4188: {id : '3118', geocod : '5001102', nome : "Aquidauana"},  
4189: {id : '3100', geocod : '5001243', nome : "Aral Moreira"},  
4190: {id : '5747', geocod : '5008305', nome : "Arapuá"},  
4191: {id : '5766', geocod : '5007695', nome : "Areado"},  
4192: {id : '9481', geocod : '5006309', nome : "Árvore Grande"},  
4193: {id : '5837', geocod : '5003108', nome : "Baianópolis"},  
4194: {id : '6144', geocod : '5007109', nome : "Bálsamo"},  
4195: {id : '2460', geocod : '5001508', nome : "Bandeirantes"},  
4196: {id : '3135', geocod : '5001904', nome : "Bataguassu"},  
4197: {id : '3194', geocod : '5002001', nome : "Batayporã"},  
4198: {id : '5957', geocod : '5003256', nome : "Baús"},  
4199: {id : '5966', geocod : '5000203', nome : "Bela Alvorada"},  
4200: {id : '3099', geocod : '5002100', nome : "Bela Vista"},  
4201: {id : '6042', geocod : '5003504', nome : "Bocajá"},  
4202: {id : '3115', geocod : '5002159', nome : "Bodoquena"},  
4203: {id : '6047', geocod : '5004908', nome : "Bom Fim"},  
4204: {id : '3103', geocod : '5002209', nome : "Bonito"},  
4205: {id : '6086', geocod : '5005004', nome : "Boqueirão"},  
4206: {id : '6105', geocod : '5002704', nome : "Braço Forte"},  
4207: {id : '3222', geocod : '5002308', nome : "Brasilândia"},  
4208: {id : '3108', geocod : '5002407', nome : "Caarapó"},  
4209: {id : '6149', geocod : '5006606', nome : "Cabeceira do Apa"},  
4210: {id : '6157', geocod : '5006309', nome : "Cachoeira"},  
4211: {id : '2461', geocod : '5002605', nome : "Camapuã"},  
4212: {id : '6228', geocod : '5001102', nome : "Camisão"},  
4213: {id : '6230', geocod : '5000906', nome : "Campestre"},  
4214: {id : '5481', geocod : '5002704', nome : "Campo Grande"},  
4215: {id : '6321', geocod : '5007901', nome : "Capão Seco"},  
4216: {id : '3092', geocod : '5002803', nome : "Caracol"},  
4217: {id : '6371', geocod : '5004502', nome : "Carumbé"},  
4218: {id : '2491', geocod : '5002902', nome : "Cassilândia"},  
4219: {id : '2488', geocod : '5002951', nome : "Chapadão do Sul"},  
4220: {id : '6455', geocod : '5001102', nome : "Cipolândia"},  
4221: {id : '6476', geocod : '5003207', nome : "Coimbra"},  
4222: {id : '6517', geocod : '5001508', nome : "Congonha"},  
4223: {id : '2453', geocod : '5003108', nome : "Corguinho"},  
4224: {id : '3094', geocod : '5003157', nome : "Coronel Sapucaia"},  
4225: {id : '2450', geocod : '5003207', nome : "Corumbá"},  
4226: {id : '2487', geocod : '5003256', nome : "Costa Rica"},  
4227: {id : '2457', geocod : '5003306', nome : "Coxim"},  
4228: {id : '6575', geocod : '5002407', nome : "Cristalina"},  
4229: {id : '6593', geocod : '5003504', nome : "Cruzaltina"},  
4230: {id : '6604', geocod : '5003801', nome : "Culturama"},  
4231: {id : '6610', geocod : '5001003', nome : "Cupins"},  
4232: {id : '6647', geocod : '5002308', nome : "Debrasa"},  
4233: {id : '3131', geocod : '5003454', nome : "Deodápolis"},  
4234: {id : '3119', geocod : '5003488', nome : "Dois Irmãos do Buriti"},  
4235: {id : '3112', geocod : '5003504', nome : "Douradina"},  
4236: {id : '3110', geocod : '5003702', nome : "Dourados"},  
4237: {id : '3124', geocod : '5003751', nome : "Eldorado"},  
4238: {id : '3127', geocod : '5003801', nome : "Fátima do Sul"},  
4239: {id : '5564', geocod : '5003900', nome : "Figueirão"},  
4240: {id : '6893', geocod : '5008305', nome : "Garcias"},  
4241: {id : '3130', geocod : '5004007', nome : "Glória de Dourados"},  
4242: {id : '6986', geocod : '5003702', nome : "Guaçu"},  
4243: {id : '6987', geocod : '5004007', nome : "Guaçulândia"},  
4244: {id : '6941', geocod : '5008305', nome : "Guadalupe do Alto Paraná"},  
4245: {id : '3105', geocod : '5004106', nome : "Guia Lopes da Laguna"},  
4246: {id : '3098', geocod : '5004304', nome : "Iguatemi"},  
4247: {id : '7088', geocod : '5008305', nome : "Ilha Comprida"},  
4248: {id : '7097', geocod : '5002902', nome : "Indaiá do Sul"},  
4249: {id : '7100', geocod : '5003702', nome : "Indápolis"},  
4250: {id : '2489', geocod : '5004403', nome : "Inocência"},  
4251: {id : '7120', geocod : '5000856', nome : "Ipezal"},  
4252: {id : '7197', geocod : '5003207', nome : "Itanorte"},  
4253: {id : '3111', geocod : '5004502', nome : "Itaporã"},  
4254: {id : '3125', geocod : '5004601', nome : "Itaquiraí"},  
4255: {id : '7225', geocod : '5003702', nome : "Itaum"},  
4256: {id : '3183', geocod : '5004700', nome : "Ivinhema"},  
4257: {id : '7250', geocod : '5002209', nome : "Jabuti"},  
4258: {id : '3122', geocod : '5004809', nome : "Japorã"},  
4259: {id : '3133', geocod : '5004908', nome : "Jaraguari"},  
4260: {id : '3104', geocod : '5005004', nome : "Jardim"},  
4261: {id : '3129', geocod : '5005103', nome : "Jateí"},  
4262: {id : '7306', geocod : '5003900', nome : "Jauru"},  
4263: {id : '7371', geocod : '5007406', nome : "Juscelândia"},  
4264: {id : '3109', geocod : '5005152', nome : "Juti"},  
4265: {id : '2451', geocod : '5005202', nome : "Ladário"},  
4266: {id : '7393', geocod : '5003454', nome : "Lagoa Bonita"},  
4267: {id : '3107', geocod : '5005251', nome : "Laguna Carapã"},  
4268: {id : '3113', geocod : '5005400', nome : "Maracaju"},  
4269: {id : '3116', geocod : '5005608', nome : "Miranda"},  
4270: {id : '7707', geocod : '5004502', nome : "Montese"},  
4271: {id : '7712', geocod : '5004403', nome : "Morangas"},  
4272: {id : '7714', geocod : '5002159', nome : "Morraria do Sul"},  
4273: {id : '7730', geocod : '5003751', nome : "Morumbi"},  
4274: {id : '3123', geocod : '5005681', nome : "Mundo Novo"},  
4275: {id : '3126', geocod : '5005707', nome : "Naviraí"},  
4276: {id : '7778', geocod : '5003207', nome : "Nhecolândia"},  
4277: {id : '3106', geocod : '5005806', nome : "Nioaque"},  
4278: {id : '7794', geocod : '5002100', nome : "Nossa Senhora de Fátima"},  
4279: {id : '3132', geocod : '5006002', nome : "Nova Alvorada do Sul"},  
4280: {id : '7808', geocod : '5002407', nome : "Nova América"},  
4281: {id : '3195', geocod : '5006200', nome : "Nova Andradina"},  
4282: {id : '7822', geocod : '5005103', nome : "Nova Esperança - Jateí"},  
4283: {id : '7823', geocod : '5007307', nome : "Nova Esperança - Rio Negro"},  
4284: {id : '7830', geocod : '5006309', nome : "Nova Jales"},  
4285: {id : '3182', geocod : '5006259', nome : "Novo Horizonte do Sul"},  
4286: {id : '7886', geocod : '5001003', nome : "Oriente"},  
4287: {id : '7925', geocod : '5003207', nome : "Paiaguás"},  
4288: {id : '7946', geocod : '5003488', nome : "Palmeiras"},  
4289: {id : '7956', geocod : '5003702', nome : "Panambi"},  
4290: {id : '7982', geocod : '5003256', nome : "Paraíso"},  
4291: {id : '2495', geocod : '5006309', nome : "Paranaíba"},  
4292: {id : '3093', geocod : '5006358', nome : "Paranhos"},  
4293: {id : '2458', geocod : '5006408', nome : "Pedro Gomes"},  
4294: {id : '8100', geocod : '5003702', nome : "Picadinha"},  
4295: {id : '8154', geocod : '5004502', nome : "Piraporã"},  
4296: {id : '8155', geocod : '5001102', nome : "Piraputanga"},  
4297: {id : '3102', geocod : '5006606', nome : "Ponta Porã"},  
4298: {id : '8202', geocod : '5007695', nome : "Ponte Vermelha"},  
4299: {id : '8208', geocod : '5002605', nome : "Pontinha do Cocho"},  
4300: {id : '8220', geocod : '5003207', nome : "Porto Esperança"},  
4301: {id : '3091', geocod : '5006903', nome : "Porto Murtinho"},  
4302: {id : '8232', geocod : '5003454', nome : "Porto Vilma"},  
4303: {id : '8233', geocod : '5001904', nome : "Porto XV de Novembro"},  
4304: {id : '8278', geocod : '5003454', nome : "Presidente Castelo"},  
4305: {id : '8292', geocod : '5007208', nome : "Prudêncio Thomaz"},  
4306: {id : '8312', geocod : '5007901', nome : "Quebra Côco"},  
4307: {id : '3389', geocod : '5007109', nome : "Ribas do Rio Pardo"},  
4308: {id : '3114', geocod : '5007208', nome : "Rio Brilhante"},  
4309: {id : '2454', geocod : '5007307', nome : "Rio Negro"},  
4310: {id : '2455', geocod : '5007406', nome : "Rio Verde de Mato Grosso"},  
4311: {id : '8444', geocod : '5004908', nome : "Rochedinho"},  
4312: {id : '2452', geocod : '5007505', nome : "Rochedo"},  
4313: {id : '8492', geocod : '5006606', nome : "Sanga Puitã"},  
4314: {id : '3203', geocod : '5007554', nome : "Santa Rita do Pardo"},  
4315: {id : '8598', geocod : '5004502', nome : "Santa Terezinha"},  
4316: {id : '2456', geocod : '5007695', nome : "São Gabriel do Oeste"},  
4317: {id : '8961', geocod : '5006309', nome : "São João do Aporé"},  
4318: {id : '8892', geocod : '5008404', nome : "São José"},  
4319: {id : '8933', geocod : '5004403', nome : "São José do Sucuriú"},  
4320: {id : '9019', geocod : '5003702', nome : "São Pedro"},  
4321: {id : '9020', geocod : '5004403', nome : "São Pedro - Inocência"},  
4322: {id : '9044', geocod : '5003306', nome : "São Romão"},  
4323: {id : '3394', geocod : '5007802', nome : "Selvíria"},  
4324: {id : '3096', geocod : '5007703', nome : "Sete Quedas"},  
4325: {id : '3120', geocod : '5007901', nome : "Sidrolândia"},  
4326: {id : '2459', geocod : '5007935', nome : "Sonora"},  
4327: {id : '3097', geocod : '5007950', nome : "Tacuru"},  
4328: {id : '9141', geocod : '5006309', nome : "Tamandaré"},  
4329: {id : '9166', geocod : '5003306', nome : "Taquari"},  
4330: {id : '3193', geocod : '5007976', nome : "Taquarussu"},  
4331: {id : '9180', geocod : '5001102', nome : "Taunay"},  
4332: {id : '3121', geocod : '5008008', nome : "Terenos"},  
4333: {id : '3391', geocod : '5008305', nome : "Três Lagoas"},  
4334: {id : '9345', geocod : '5006309', nome : "Velhacaria"},  
4335: {id : '3128', geocod : '5008404', nome : "Vicentina"},  
4336: {id : '9375', geocod : '5003702', nome : "Vila Formosa"},  
4337: {id : '9398', geocod : '5008404', nome : "Vila Rica"},  
4338: {id : '9405', geocod : '5003454', nome : "Vila União"},  
4339: {id : '9406', geocod : '5003702', nome : "Vila Vargas"},  
4340: {id : '9424', geocod : '5005400', nome : "Vista Alegre"}  
4341: ],  
4342: MT : [  
4343: // UF = MT  
4344: {id : '1990', geocod : '5100102', nome : "Acorizal"},  
4345: {id : '2018', geocod : '5100201', nome : "Água Boa"},  
4346: {id : '9470', geocod : '5103007', nome : "Água Fria"},  
4347: {id : '5588', geocod : '5105507', nome : "Aguapeí"},  
4348: {id : '5606', geocod : '5104203', nome : "Alcantilado"},  
4349: {id : '1374', geocod : '5100250', nome : "Alta Floresta"},  
4350: {id : '2473', geocod : '5100300', nome : "Alto Araguaia"},  
4351: {id : '1380', geocod : '5100359', nome : "Alto Boa Vista"},  
4352: {id : '5638', geocod : '5107008', nome : "Alto Coité"},  
4353: {id : '2486', geocod : '5100409', nome : "Alto Garças"},  
4354: {id : '5640', geocod : '5102686', nome : "Alto Juruena"},  
4355: {id : '1997', geocod : '5100508', nome : "Alto Paraguai"},  
4356: {id : '2472', geocod : '5100607', nome : "Alto Taquari"},  
4357: {id : '5697', geocod : '5107602', nome : "Anhumas"},  
4358: {id : '1365', geocod : '5100805', nome : "Apiacás"},  
4359: {id : '2026', geocod : '5101001', nome : "Araguaiana"},  
4360: {id : '2475', geocod : '5101209', nome : "Araguainha"},  
4361: {id : '1954', geocod : '5101258', nome : "Araputanga"},  
4362: {id : '1982', geocod : '5101308', nome : "Arenápolis"},  
4363: {id : '1352', geocod : '5101407', nome : "Aripuanã"},  
4364: {id : '5793', geocod : '5107701', nome : "Arruda"},  
4365: {id : '5800', geocod : '5101704', nome : "Assari"},  
4366: {id : '2467', geocod : '5101605', nome : "Barão de Melgaço"},  
4367: {id : '1975', geocod : '5101704', nome : "Barra do Bugres"},  
4368: {id : '2025', geocod : '5101803', nome : "Barra do Garças"},  
4369: {id : '5952', geocod : '5108105', nome : "Batovi"},  
4370: {id : '5958', geocod : '5107206', nome : "Baús"},  
4371: {id : '5955', geocod : '5107701', nome : "Bauxi"},  
4372: {id : '6003', geocod : '5102504', nome : "Bezerro Branco"},  
4373: {id : '6014', geocod : '5106240', nome : "Boa Esperança"},  
4374: {id : '6027', geocod : '5107602', nome : "Boa Vista"},  
4375: {id : '5503', geocod : '5101852', nome : "Bom Jesus do Araguaia"},  
4376: {id : '6070', geocod : '5108402', nome : "Bom Sucesso"},  
4377: {id : '2004', geocod : '5101902', nome : "Brasnorte"},  
4378: {id : '6137', geocod : '5100300', nome : "Buriti"},  
4379: {id : '6138', geocod : '5104203', nome : "Buriti - Guiratinga"},  
4380: {id : '2465', geocod : '5102504', nome : "Cáceres"},  
4381: {id : '6194', geocod : '5107800', nome : "Caité"},  
4382: {id : '2021', geocod : '5102603', nome : "Campinápolis"},  
4383: {id : '1978', geocod : '5102637', nome : "Campo Novo do Parecis"},  
4384: {id : '1996', geocod : '5102678', nome : "Campo Verde"},  
4385: {id : '1967', geocod : '5102686', nome : "Campos de Júlio"},  
4386: {id : '1381', geocod : '5102694', nome : "Canabrava do Norte"},  
4387: {id : '2029', geocod : '5102702', nome : "Canarana"},  
4388: {id : '6284', geocod : '5106505', nome : "Cangas"},  
4389: {id : '6319', geocod : '5108402', nome : "Capão Grande"},  
4390: {id : '6322', geocod : '5100508', nome : "Capão Verde"},  
4391: {id : '6335', geocod : '5102504', nome : "Caramujo"},  
4392: {id : '6342', geocod : '5107925', nome : "Caravágio"},  
4393: {id : '1375', geocod : '5102793', nome : "Carlinda"},  
4394: {id : '6380', geocod : '5108105', nome : "Cassununga"},  
4395: {id : '1356', geocod : '5102850', nome : "Castanheira"},  
4396: {id : '6418', geocod : '5107800', nome : "Celma"},  
4397: {id : '1993', geocod : '5103007', nome : "Chapada dos Guimarães"},  
4398: {id : '1372', geocod : '5103056', nome : "Cláudia"},  
4399: {id : '2032', geocod : '5103106', nome : "Cocalinho"},  
4400: {id : '1370', geocod : '5103205', nome : "Colíder"},  
4401: {id : '5553', geocod : '5103254', nome : "Colniza"},  
4402: {id : '6481', geocod : '5106216', nome : "Colorado do Norte"},  
4403: {id : '1965', geocod : '5103304', nome : "Comodoro"},  
4404: {id : '1383', geocod : '5103353', nome : "Confresa"},  
4405: {id : '5511', geocod : '5103361', nome : "Conquista D'oeste"},  
4406: {id : '6530', geocod : '5105903', nome : "Coqueiral"},  
4407: {id : '6541', geocod : '5102678', nome : "Coronel Ponce"},  
4408: {id : '1362', geocod : '5103379', nome : "Cotriguaçu"},  
4409: {id : '6566', geocod : '5103403', nome : "Coxipó da Ponte"},  
4410: {id : '6567', geocod : '5103403', nome : "Coxipó do Ouro"},  
4411: {id : '6577', geocod : '5107750', nome : "Cristinópolis"},  
4412: {id : '5482', geocod : '5103403', nome : "Cuiabá"},  
4413: {id : '5510', geocod : '5103437', nome : "Curvelândia"},  
4414: {id : '1977', geocod : '5103452', nome : "Denise"},  
4415: {id : '1998', geocod : '5103502', nome : "Diamantino"},  
4416: {id : '1995', geocod : '5103601', nome : "Dom Aquino"},  
4417: {id : '6732', geocod : '5100102', nome : "Engenho"},  
4418: {id : '6737', geocod : '5107800', nome : "Engenho Velho"},  
4419: {id : '6741', geocod : '5103601', nome : "Entre Rios"},  
4420: {id : '6777', geocod : '5104203', nome : "Estrela do Leste"},  
4421: {id : '6804', geocod : '5106505', nome : "Fazenda de Cima"},  
4422: {id : '2007', geocod : '5103700', nome : "Feliz Natal"},  
4423: {id : '1964', geocod : '5103809', nome : "Figueirópolis D'Oeste"},  
4424: {id : '6848', geocod : '5105150', nome : "Fontanillas"},  
4425: {id : '2022', geocod : '5103858', nome : "Gaúcha do Norte"},  
4426: {id : '2020', geocod : '5103908', nome : "General Carneiro"},  
4427: {id : '1971', geocod : '5103957', nome : "Glória D'Oeste"},  
4428: {id : '1355', geocod : '5104104', nome : "Guarantã do Norte"},  
4429: {id : '6991', geocod : '5103403', nome : "Guia"},  
4430: {id : '2521', geocod : '5104203', nome : "Guiratinga"},  
4431: {id : '7014', geocod : '5102504', nome : "Horizonte do Oeste"},  
4432: {id : '1972', geocod : '5104500', nome : "Indiavaí"},  
4433: {id : '5562', geocod : '5104526', nome : "Ipiranga do Norte"},  
4434: {id : '7146', geocod : '5105200', nome : "Irenópolis"},  
4435: {id : '5563', geocod : '5104542', nome : "Itanhangá"},  
4436: {id : '1369', geocod : '5104559', nome : "Itaúba"},  
4437: {id : '2462', geocod : '5104609', nome : "Itiquira"},  
4438: {id : '1994', geocod : '5104807', nome : "Jaciara"},  
4439: {id : '1988', geocod : '5104906', nome : "Jangada"},  
4440: {id : '7294', geocod : '5100201', nome : "Jaraguá"},  
4441: {id : '7304', geocod : '5107008', nome : "Jarudore"},  
4442: {id : '1963', geocod : '5105002', nome : "Jauru"},  
4443: {id : '7326', geocod : '5101605', nome : "Joselândia"},  
4444: {id : '1358', geocod : '5105101', nome : "Juara"},  
4445: {id : '1351', geocod : '5105150', nome : "Juína"},  
4446: {id : '1361', geocod : '5105176', nome : "Juruena"},  
4447: {id : '2469', geocod : '5105200', nome : "Juscimeira"},  
4448: {id : '1959', geocod : '5105234', nome : "Lambari D'Oeste"},  
4449: {id : '7447', geocod : '5100508', nome : "Lavouras"},  
4450: {id : '2001', geocod : '5105259', nome : "Lucas do Rio Verde"},  
4451: {id : '7479', geocod : '5105002', nome : "Lucialva"},  
4452: {id : '1385', geocod : '5105309', nome : "Luciára"},  
4453: {id : '1378', geocod : '5105580', nome : "Marcelândia"},  
4454: {id : '1354', geocod : '5105606', nome : "Matupá"},  
4455: {id : '7611', geocod : '5107701', nome : "Mazargão"},  
4456: {id : '7631', geocod : '5107800', nome : "Mimoso"},  
4457: {id : '7641', geocod : '5108055', nome : "Miraguaí do Norte"},  
4458: {id : '1957', geocod : '5105622', nome : "Mirassol d'Oeste"},  
4459: {id : '1992', geocod : '5105903', nome : "Nobres"},  
4460: {id : '7782', geocod : '5108055', nome : "Nona Agrovila"},  
4461: {id : '7783', geocod : '5108808', nome : "Nonaí do Norte"},  
4462: {id : '1983', geocod : '5106000', nome : "Nortelândia"},  
4463: {id : '1986', geocod : '5106109', nome : "Nossa Senhora do Livramento"},  
4464: {id : '7805', geocod : '5103304', nome : "Nova Alvorada"},  
4465: {id : '1363', geocod : '5106158', nome : "Nova Bandeirantes"},  
4466: {id : '1960', geocod : '5106208', nome : "Nova Brasilândia"},  
4467: {id : '1366', geocod : '5106216', nome : "Nova Canaã do Norte"},  
4468: {id : '7817', geocod : '5107297', nome : "Nova Catanduva"},  
4469: {id : '7826', geocod : '5107602', nome : "Nova Galiléia"},  
4470: {id : '1376', geocod : '5108808', nome : "Nova Guarita"},  
4471: {id : '1966', geocod : '5106182', nome : "Nova Lacerda"},  
4472: {id : '1981', geocod : '5108857', nome : "Nova Marilândia"},  
4473: {id : '1985', geocod : '5108907', nome : "Nova Maringá"},  
4474: {id : '1364', geocod : '5108956', nome : "Nova Monte Verde"},  
4475: {id : '1999', geocod : '5106224', nome : "Nova Mutum"},  
4476: {id : '5551', geocod : '5106174', nome : "Nova Nazaré"},  
4477: {id : '1976', geocod : '5106232', nome : "Nova Olímpia"},  
4478: {id : '5501', geocod : '5106190', nome : "Nova Santa Helena"},  
4479: {id : '2003', geocod : '5106240', nome : "Nova Ubiratã"},  
4480: {id : '2017', geocod : '5106257', nome : "Nova Xavantina"},  
4481: {id : '7852', geocod : '5108006', nome : "Novo Eldorado"},  
4482: {id : '1359', geocod : '5106273', nome : "Novo Horizonte do Norte"},  
4483: {id : '1350', geocod : '5106265', nome : "Novo Mundo"},  
4484: {id : '5504', geocod : '5106315', nome : "Novo Santo Antônio"},  
4485: {id : '2019', geocod : '5106281', nome : "Novo São Joaquim"},  
4486: {id : '7924', geocod : '5103304', nome : "Padronal"},  
4487: {id : '7985', geocod : '5107008', nome : "Paraíso do Leste"},  
4488: {id : '1373', geocod : '5106299', nome : "Paranaíta"},  
4489: {id : '2009', geocod : '5106307', nome : "Paranatinga"},  
4490: {id : '8001', geocod : '5108402', nome : "Passagem da Conceição"},  
4491: {id : '2463', geocod : '5106372', nome : "Pedra Preta"},  
4492: {id : '1353', geocod : '5106422', nome : "Peixoto de Azevedo"},  
4493: {id : '8163', geocod : '5106109', nome : "Pirizal"},  
4494: {id : '2002', geocod : '5106455', nome : "Planalto da Serra"},  
4495: {id : '2466', geocod : '5106505', nome : "Poconé"},  
4496: {id : '2023', geocod : '5106653', nome : "Pontal do Araguaia"},  
4497: {id : '2523', geocod : '5106703', nome : "Ponte Branca"},  
4498: {id : '1962', geocod : '5106752', nome : "Pontes e Lacerda"},  
4499: {id : '1382', geocod : '5106778', nome : "Porto Alegre do Norte"},  
4500: {id : '1357', geocod : '5106802', nome : "Porto dos Gaúchos"},  
4501: {id : '1970', geocod : '5106828', nome : "Porto Esperidião"},  
4502: {id : '1974', geocod : '5106851', nome : "Porto Estrela"},  
4503: {id : '8230', geocod : '5108402', nome : "Porto Velho"},  
4504: {id : '1969', geocod : '5107008', nome : "Poxoréo"},  
4505: {id : '8268', geocod : '5106208', nome : "Praia Rica"},  
4506: {id : '8284', geocod : '5107925', nome : "Primavera - Sorriso"},  
4507: {id : '2008', geocod : '5107040', nome : "Primavera do Leste"},  
4508: {id : '8290', geocod : '5107958', nome : "Progresso"},  
4509: {id : '2030', geocod : '5107065', nome : "Querência"},  
4510: {id : '1955', geocod : '5107156', nome : "Reserva do Cabaçal"},  
4511: {id : '2031', geocod : '5107180', nome : "Ribeirão Cascalheira"},  
4512: {id : '8386', geocod : '5106109', nome : "Ribeirão dos Cocais"},  
4513: {id : '2524', geocod : '5107198', nome : "Ribeirãozinho"},  
4514: {id : '1958', geocod : '5107206', nome : "Rio Branco"},  
4515: {id : '8437', geocod : '5106208', nome : "Riolândia"},  
4516: {id : '5554', geocod : '5107578', nome : "Rondolândia"},  
4517: {id : '2468', geocod : '5107602', nome : "Rondonópolis"},  
4518: {id : '1991', geocod : '5107701', nome : "Rosário Oeste"},  
4519: {id : '1973', geocod : '5107750', nome : "Salto do Céu"},  
4520: {id : '1368', geocod : '5107248', nome : "Santa Carmem"},  
4521: {id : '5502', geocod : '5107743', nome : "Santa Cruz do Xingu"},  
4522: {id : '8518', geocod : '5105200', nome : "Santa Elvira"},  
4523: {id : '8528', geocod : '5107107', nome : "Santa Fé"},  
4524: {id : '8533', geocod : '5106190', nome : "Santa Helena"},  
4525: {id : '5508', geocod : '5107768', nome : "Santa Rita do Trivelato"},  
4526: {id : '1387', geocod : '5107776', nome : "Santa Terezinha"},  
4527: {id : '1980', geocod : '5107263', nome : "Santo Afonso"},  
4528: {id : '5509', geocod : '5107792', nome : "Santo Antônio do Leste"},  
4529: {id : '1987', geocod : '5107800', nome : "Santo Antônio do Leverger"},  
4530: {id : '8840', geocod : '5108352', nome : "São Domingos"},  
4531: {id : '1384', geocod : '5107859', nome : "São Félix do Araguaia"},  
4532: {id : '8876', geocod : '5107958', nome : "São Joaquim"},  
4533: {id : '8882', geocod : '5107958', nome : "São Jorge"},  
4534: {id : '8917', geocod : '5102603', nome : "São José do Couto"},  
4535: {id : '8928', geocod : '5106372', nome : "São José do Planalto"},  
4536: {id : '2464', geocod : '5107297', nome : "São José do Povo"},  
4537: {id : '1984', geocod : '5107305', nome : "São José do Rio Claro"},  
4538: {id : '1379', geocod : '5107354', nome : "São José do Xingu"},  
4539: {id : '1956', geocod : '5107107', nome : "São José dos Quatro Marcos"},  
4540: {id : '8976', geocod : '5105200', nome : "São Lourenço de Fátima"},  
4541: {id : '2470', geocod : '5107404', nome : "São Pedro da Cipa"},  
4542: {id : '1968', geocod : '5107875', nome : "Sapezal"},  
4543: {id : '8709', geocod : '5106109', nome : "Seco"},  
4544: {id : '5505', geocod : '5107883', nome : "Serra Nova Dourada"},  
4545: {id : '8749', geocod : '5100201', nome : "Serrinha"},  
4546: {id : '1367', geocod : '5107909', nome : "Sinop"},  
4547: {id : '8787', geocod : '5105622', nome : "Sonho Azul"},  
4548: {id : '2005', geocod : '5107925', nome : "Sorriso"},  
4549: {id : '1360', geocod : '5107941', nome : "Tabaporã"},  
4550: {id : '1979', geocod : '5107958', nome : "Tangará da Serra"},  
4551: {id : '9157', geocod : '5107958', nome : "Tapirapuã"},  
4552: {id : '2000', geocod : '5108006', nome : "Tapurah"},  
4553: {id : '1371', geocod : '5108055', nome : "Terra Nova do Norte"},  
4554: {id : '2522', geocod : '5108105', nome : "Tesouro"},  
4555: {id : '9219', geocod : '5101803', nome : "Toricueije"},  
4556: {id : '9220', geocod : '5104203', nome : "Toriparu"},  
4557: {id : '2525', geocod : '5108204', nome : "Torixoréu"},  
4558: {id : '1377', geocod : '5108303', nome : "União do Sul"},  
4559: {id : '5507', geocod : '5108352', nome : "Vale de São Domingos"},  
4560: {id : '9321', geocod : '5101803', nome : "Vale dos Sonhos"},  
4561: {id : '9335', geocod : '5106109', nome : "Varginha"},  
4562: {id : '1989', geocod : '5108402', nome : "Várzea Grande"},  
4563: {id : '2006', geocod : '5108501', nome : "Vera"},  
4564: {id : '1961', geocod : '5105507', nome : "Vila Bela da Santíssima Trin"},  
4565: {id : '9391', geocod : '5107602', nome : "Vila Operária"},  
4566: {id : '9393', geocod : '5103908', nome : "Vila Paredão"},  
4567: {id : '9395', geocod : '5107958', nome : "Vila Progresso"},  
4568: {id : '1386', geocod : '5108600', nome : "Vila Rica"}  
4569: ],  
4570: PA : [  
4571: // UF = PA  
4572: {id : '115', geocod : '1500107', nome : "Abaetetuba"},  
4573: {id : '423', geocod : '1500131', nome : "Abel Figueiredo"},  
4574: {id : '129', geocod : '1500206', nome : "Acará"},  
4575: {id : '88', geocod : '1500305', nome : "Afuá"},  
4576: {id : '389', geocod : '1500347', nome : "Água Azul do Norte"},  
4577: {id : '76', geocod : '1500404', nome : "Alenquer"},  
4578: {id : '89', geocod : '1500503', nome : "Almeirim"},  
4579: {id : '5627', geocod : '1501709', nome : "Almoço"},  
4580: {id : '95', geocod : '1500602', nome : "Altamira"},  
4581: {id : '5628', geocod : '1506807', nome : "Alter do Chão"},  
4582: {id : '5680', geocod : '1506500', nome : "Americano"},  
4583: {id : '111', geocod : '1500701', nome : "Anajás"},  
4584: {id : '119', geocod : '1500800', nome : "Ananindeua"},  
4585: {id : '98', geocod : '1500859', nome : "Anapu"},  
4586: {id : '5703', geocod : '1501808', nome : "Antonio Lemos"},  
4587: {id : '5725', geocod : '1502400', nome : "Apeú"},  
4588: {id : '5728', geocod : '1507508', nome : "Apinagés"},  
4589: {id : '5769', geocod : '1502103', nome : "Areião"},  
4590: {id : '5768', geocod : '1504505', nome : "Areias"},  
4591: {id : '5796', geocod : '1500503', nome : "Arumanduba"},  
4592: {id : '5810', geocod : '1500909', nome : "Aturiaí"},  
4593: {id : '189', geocod : '1500909', nome : "Augusto Corrêa"},  
4594: {id : '149', geocod : '1500958', nome : "Aurora do Pará"},  
4595: {id : '70', geocod : '1501006', nome : "Aveiro"},  
4596: {id : '106', geocod : '1501105', nome : "Bagre"},  
4597: {id : '103', geocod : '1501204', nome : "Baião"},  
4598: {id : '386', geocod : '1501253', nome : "Bannach"},  
4599: {id : '118', geocod : '1501303', nome : "Barcarena"},  
4600: {id : '5965', geocod : '1500107', nome : "Beja"},  
4601: {id : '5483', geocod : '1501402', nome : "Belém"},  
4602: {id : '71', geocod : '1501451', nome : "Belterra"},  
4603: {id : '121', geocod : '1501501', nome : "Benevides"},  
4604: {id : '5992', geocod : '1501501', nome : "Benfica"},  
4605: {id : '6011', geocod : '1504307', nome : "Boa Esperança"},  
4606: {id : '6044', geocod : '1506807', nome : "Boim"},  
4607: {id : '421', geocod : '1501576', nome : "Bom Jesus do Tocantins"},  
4608: {id : '165', geocod : '1501600', nome : "Bonito"},  
4609: {id : '182', geocod : '1501709', nome : "Bragança"},  
4610: {id : '93', geocod : '1501725', nome : "Brasil Novo"},  
4611: {id : '6103', geocod : '1501006', nome : "Brasília Legal"},  
4612: {id : '418', geocod : '1501758', nome : "Brejo Grande do Araguaia"},  
4613: {id : '102', geocod : '1501782', nome : "Breu Branco"},  
4614: {id : '85', geocod : '1501808', nome : "Breves"},  
4615: {id : '130', geocod : '1501907', nome : "Bujaru"},  
4616: {id : '124', geocod : '1502004', nome : "Cachoeira do Arari"},  
4617: {id : '183', geocod : '1501956', nome : "Cachoeira do Piriá"},  
4618: {id : '6185', geocod : '1504109', nome : "Cafezal"},  
4619: {id : '6192', geocod : '1504703', nome : "Cairari"},  
4620: {id : '6200', geocod : '1507607', nome : "Caju"},  
4621: {id : '99', geocod : '1502103', nome : "Cametá"},  
4622: {id : '5497', geocod : '1502152', nome : "Canaã dos Carajas"},  
4623: {id : '179', geocod : '1502202', nome : "Capanema"},  
4624: {id : '177', geocod : '1502301', nome : "Capitão Poço"},  
4625: {id : '6336', geocod : '1502103', nome : "Carapajó"},  
4626: {id : '6337', geocod : '1506500', nome : "Caraparu"},  
4627: {id : '6340', geocod : '1501709', nome : "Caratateua"},  
4628: {id : '6355', geocod : '1503200', nome : "Caripi"},  
4629: {id : '6368', geocod : '1504505', nome : "Carrazedo"},  
4630: {id : '158', geocod : '1502400', nome : "Castanhal"},  
4631: {id : '112', geocod : '1502509', nome : "Chaves"},  
4632: {id : '127', geocod : '1502608', nome : "Colares"},  
4633: {id : '1440', geocod : '1502707', nome : "Conceição do Araguaia"},  
4634: {id : '155', geocod : '1502756', nome : "Concórdia do Pará"},  
4635: {id : '6515', geocod : '1506302', nome : "Condeixa"},  
4636: {id : '385', geocod : '1502764', nome : "Cumaru do Norte"},  
4637: {id : '410', geocod : '1502772', nome : "Curionópolis"},  
4638: {id : '108', geocod : '1502806', nome : "Curralinho"},  
4639: {id : '75', geocod : '1502855', nome : "Curuá"},  
4640: {id : '6616', geocod : '1506807', nome : "Curuai"},  
4641: {id : '170', geocod : '1502905', nome : "Curuçá"},  
4642: {id : '6622', geocod : '1502103', nome : "Curuçambaba"},  
4643: {id : '6618', geocod : '1501808', nome : "Curumu"},  
4644: {id : '516', geocod : '1502939', nome : "Dom Eliseu"},  
4645: {id : '411', geocod : '1502954', nome : "Eldorado dos Carajás"},  
4646: {id : '6717', geocod : '1500909', nome : "Emboraí"},  
4647: {id : '6763', geocod : '1507003', nome : "Espírito Santo do Tauá"},  
4648: {id : '66', geocod : '1503002', nome : "Faro"},  
4649: {id : '6815', geocod : '1508308', nome : "Fernandes Belo"},  
4650: {id : '391', geocod : '1503044', nome : "Floresta do Araguaia"},  
4651: {id : '176', geocod : '1503077', nome : "Garrafão do Norte"},  
4652: {id : '100', geocod : '1503093', nome : "Goianésia do Pará"},  
4653: {id : '6949', geocod : '1501907', nome : "Guajará Açu"},  
4654: {id : '6950', geocod : '1500206', nome : "Guajará Mirim"},  
4655: {id : '82', geocod : '1503101', nome : "Gurupá"},  
4656: {id : '7062', geocod : '1501402', nome : "Icoraci"},  
4657: {id : '164', geocod : '1503200', nome : "Igarapé-Açu"},  
4658: {id : '113', geocod : '1503309', nome : "Igarapé-Miri"},  
4659: {id : '157', geocod : '1503408', nome : "Inhangapi"},  
4660: {id : '154', geocod : '1503457', nome : "Ipixuna do Pará"},  
4661: {id : '162', geocod : '1503507', nome : "Irituia"},  
4662: {id : '382', geocod : '1503606', nome : "Itaituba"},  
4663: {id : '7212', geocod : '1508308', nome : "Itapixuna"},  
4664: {id : '7224', geocod : '1503101', nome : "Itatupã"},  
4665: {id : '413', geocod : '1503705', nome : "Itupiranga"},  
4666: {id : '379', geocod : '1503754', nome : "Jacareacanga"},  
4667: {id : '420', geocod : '1503804', nome : "Jacundá"},  
4668: {id : '7274', geocod : '1500206', nome : "Jaguarari"},  
4669: {id : '7289', geocod : '1507474', nome : "Japerica"},  
4670: {id : '7315', geocod : '1502103', nome : "Joana Coeli"},  
4671: {id : '7316', geocod : '1501204', nome : "Joana Peres"},  
4672: {id : '7317', geocod : '1506302', nome : "Joanes"},  
4673: {id : '7340', geocod : '1502103', nome : "Juaba"},  
4674: {id : '7346', geocod : '1506302', nome : "Jubim"},  
4675: {id : '69', geocod : '1503903', nome : "Juruti"},  
4676: {id : '7446', geocod : '1502905', nome : "Lauro Sodré"},  
4677: {id : '109', geocod : '1504000', nome : "Limoeiro do Ajuru"},  
4678: {id : '160', geocod : '1504059', nome : "Mãe do Rio"},  
4679: {id : '171', geocod : '1504109', nome : "Magalhães Barata"},  
4680: {id : '7503', geocod : '1503309', nome : "Maiauatá"},  
4681: {id : '414', geocod : '1504208', nome : "Marabá"},  
4682: {id : '173', geocod : '1504307', nome : "Maracanã"},  
4683: {id : '172', geocod : '1504406', nome : "Marapanim"},  
4684: {id : '120', geocod : '1504422', nome : "Marituba"},  
4685: {id : '7589', geocod : '1504406', nome : "Marudá"},  
4686: {id : '7599', geocod : '1504406', nome : "Matapiquara"},  
4687: {id : '92', geocod : '1504455', nome : "Medicilândia"},  
4688: {id : '84', geocod : '1504505', nome : "Melgaço"},  
4689: {id : '7652', geocod : '1502202', nome : "Mirasselvas"},  
4690: {id : '104', geocod : '1504604', nome : "Mocajuba"},  
4691: {id : '7664', geocod : '1502103', nome : "Moiraba"},  
4692: {id : '114', geocod : '1504703', nome : "Moju"},  
4693: {id : '7672', geocod : '1506302', nome : "Monsarás"},  
4694: {id : '78', geocod : '1504802', nome : "Monte Alegre"},  
4695: {id : '7680', geocod : '1504406', nome : "Monte Alegre do Maú"},  
4696: {id : '7692', geocod : '1500503', nome : "Monte Dourado"},  
4697: {id : '7732', geocod : '1501402', nome : "Mosqueiro"},  
4698: {id : '116', geocod : '1504901', nome : "Muaná"},  
4699: {id : '7738', geocod : '1506807', nome : "Mujuí dos Campos"},  
4700: {id : '7750', geocod : '1502905', nome : "Murajá"},  
4701: {id : '7752', geocod : '1501303', nome : "Murucupi"},  
4702: {id : '151', geocod : '1504950', nome : "Nova Esperança do Piriá"},  
4703: {id : '434', geocod : '1504976', nome : "Nova Ipixuna"},  
4704: {id : '7837', geocod : '1501709', nome : "Nova Mocajuba"},  
4705: {id : '166', geocod : '1505007', nome : "Nova Timboteua"},  
4706: {id : '380', geocod : '1505031', nome : "Novo Progresso"},  
4707: {id : '412', geocod : '1505064', nome : "Novo Repartimento"},  
4708: {id : '74', geocod : '1505106', nome : "Óbidos"},  
4709: {id : '107', geocod : '1505205', nome : "Oeiras do Pará"},  
4710: {id : '73', geocod : '1505304', nome : "Oriximiná"},  
4711: {id : '178', geocod : '1505403', nome : "Ourém"},  
4712: {id : '388', geocod : '1505437', nome : "Ourilândia do Norte"},  
4713: {id : '7902', geocod : '1501402', nome : "Outeiro"},  
4714: {id : '79', geocod : '1505486', nome : "Pacajá"},  
4715: {id : '7909', geocod : '1506005', nome : "Pacoval"},  
4716: {id : '419', geocod : '1505494', nome : "Palestina do Pará"},  
4717: {id : '153', geocod : '1505502', nome : "Paragominas"},  
4718: {id : '409', geocod : '1505536', nome : "Parauapebas"},  
4719: {id : '390', geocod : '1505551', nome : "Pau D'Arco"},  
4720: {id : '8057', geocod : '1501105', nome : "Pedreira"},  
4721: {id : '167', geocod : '1505601', nome : "Peixe-Boi"},  
4722: {id : '8088', geocod : '1507201', nome : "Perseverança"},  
4723: {id : '400', geocod : '1505635', nome : "Piçarra"},  
4724: {id : '8133', geocod : '1501006', nome : "Pinhel"},  
4725: {id : '8164', geocod : '1502806', nome : "Piriá"},  
4726: {id : '77', geocod : '1505650', nome : "Placas"},  
4727: {id : '117', geocod : '1505700', nome : "Ponta de Pedras"},  
4728: {id : '8193', geocod : '1502905', nome : "Ponta de Ramos"},  
4729: {id : '83', geocod : '1505809', nome : "Portel"},  
4730: {id : '80', geocod : '1505908', nome : "Porto de Moz"},  
4731: {id : '8225', geocod : '1503754', nome : "Porto Rico"},  
4732: {id : '8226', geocod : '1508209', nome : "Porto Salvo"},  
4733: {id : '94', geocod : '1506005', nome : "Prainha"},  
4734: {id : '186', geocod : '1506104', nome : "Primavera"},  
4735: {id : '187', geocod : '1506112', nome : "Quatipuru"},  
4736: {id : '1439', geocod : '1506138', nome : "Redenção"},  
4737: {id : '393', geocod : '1506161', nome : "Rio Maria"},  
4738: {id : '433', geocod : '1506187', nome : "Rondon do Pará"},  
4739: {id : '383', geocod : '1506195', nome : "Rurópolis"},  
4740: {id : '175', geocod : '1506203', nome : "Salinópolis"},  
4741: {id : '125', geocod : '1506302', nome : "Salvaterra"},  
4742: {id : '122', geocod : '1506351', nome : "Santa Bárbara do Pará"},  
4743: {id : '123', geocod : '1506401', nome : "Santa Cruz do Arari"},  
4744: {id : '131', geocod : '1506500', nome : "Santa Isabel do Pará"},  
4745: {id : '180', geocod : '1506559', nome : "Santa Luzia do Pará"},  
4746: {id : '1408', geocod : '1506583', nome : "Santa Maria das Barreiras"},  
4747: {id : '163', geocod : '1506609', nome : "Santa Maria do Pará"},  
4748: {id : '1388', geocod : '1506708', nome : "Santana do Araguaia"},  
4749: {id : '72', geocod : '1506807', nome : "Santarém"},  
4750: {id : '174', geocod : '1506906', nome : "Santarém Novo"},  
4751: {id : '132', geocod : '1507003', nome : "Santo Antônio do Tauá"},  
4752: {id : '134', geocod : '1507102', nome : "São Caetano de Odivelas"},  
4753: {id : '416', geocod : '1507151', nome : "São Domingos do Araguaia"},  
4754: {id : '156', geocod : '1507201', nome : "São Domingos do Capim"},  
4755: {id : '384', geocod : '1507300', nome : "São Félix do Xingu"},  
4756: {id : '8850', geocod : '1504901', nome : "São Francisco da Jararaca"},  
4757: {id : '159', geocod : '1507409', nome : "São Francisco do Pará"},  
4758: {id : '406', geocod : '1507458', nome : "São Geraldo do Araguaia"},  
4759: {id : '169', geocod : '1507466', nome : "São João da Ponta"},  
4760: {id : '188', geocod : '1507474', nome : "São João de Pirabas"},  
4761: {id : '415', geocod : '1507508', nome : "São João do Araguaia"},  
4762: {id : '8877', geocod : '1504703', nome : "São Joaquim do Ituquara"},  
4763: {id : '8918', geocod : '1508308', nome : "São José do Gurupi"},  
4764: {id : '8927', geocod : '1508308', nome : "São José do Piriá"},  
4765: {id : '161', geocod : '1507607', nome : "São Miguel do Guamá"},  
4766: {id : '9009', geocod : '1501808', nome : "São Miguel dos Macacos"},  
4767: {id : '9029', geocod : '1504604', nome : "São Pedro de Viseu"},  
4768: {id : '9038', geocod : '1507003', nome : "São Raimundo de Borralhos"},  
4769: {id : '9040', geocod : '1501758', nome : "São Raimundo do Araguaia"},  
4770: {id : '9041', geocod : '1504307', nome : "São Roberto"},  
4771: {id : '110', geocod : '1507706', nome : "São Sebastião da Boa Vista"},  
4772: {id : '9070', geocod : '1502509', nome : "São Sebastião de Viçosa"},  
4773: {id : '395', geocod : '1507755', nome : "Sapucaia"},  
4774: {id : '97', geocod : '1507805', nome : "Senador José Porfírio"},  
4775: {id : '126', geocod : '1507904', nome : "Soure"},  
4776: {id : '105', geocod : '1507953', nome : "Tailândia"},  
4777: {id : '9178', geocod : '1502202', nome : "Tauari"},  
4778: {id : '9179', geocod : '1505601', nome : "Tauarizinho"},  
4779: {id : '168', geocod : '1507961', nome : "Terra Alta"},  
4780: {id : '68', geocod : '1507979', nome : "Terra Santa"},  
4781: {id : '9199', geocod : '1500909', nome : "Tijoca"},  
4782: {id : '9203', geocod : '1505007', nome : "Timboteua"},  
4783: {id : '128', geocod : '1508001', nome : "Tomé-Açu"},  
4784: {id : '181', geocod : '1508035', nome : "Tracuateua"},  
4785: {id : '381', geocod : '1508050', nome : "Trairão"},  
4786: {id : '387', geocod : '1508084', nome : "Tucumã"},  
4787: {id : '101', geocod : '1508100', nome : "Tucuruí"},  
4788: {id : '152', geocod : '1508126', nome : "Ulianópolis"},  
4789: {id : '91', geocod : '1508159', nome : "Uruará"},  
4790: {id : '9300', geocod : '1507607', nome : "Urucuri"},  
4791: {id : '9301', geocod : '1507607', nome : "Urucuriteua"},  
4792: {id : '9312', geocod : '1501402', nome : "Val de Cães"},  
4793: {id : '9343', geocod : '1505908', nome : "Veiros"},  
4794: {id : '133', geocod : '1508209', nome : "Vigia"},  
4795: {id : '9412', geocod : '1502103', nome : "Vila do Carmo do Tocantins"},  
4796: {id : '9413', geocod : '1505908', nome : "Vilarinho do Monte"},  
4797: {id : '185', geocod : '1508308', nome : "Viseu"},  
4798: {id : '96', geocod : '1508357', nome : "Vitória do Xingu"},  
4799: {id : '394', geocod : '1508407', nome : "Xinguara"}  
4800: ],  
4801: PB : [  
4802: // UF = PB  
4803: {id : '990', geocod : '2500106', nome : "Água Branca"},  
4804: {id : '839', geocod : '2500205', nome : "Aguiar"},  
4805: {id : '1204', geocod : '2500304', nome : "Alagoa Grande"},  
4806: {id : '1153', geocod : '2500403', nome : "Alagoa Nova"},  
4807: {id : '1206', geocod : '2500502', nome : "Alagoinha"},  
4808: {id : '1029', geocod : '2500536', nome : "Alcantil"},  
4809: {id : '1055', geocod : '2500577', nome : "Algodão de Jandaíra"},  
4810: {id : '1122', geocod : '2500601', nome : "Alhandra"},  
4811: {id : '948', geocod : '2500734', nome : "Amparo"},  
4812: {id : '845', geocod : '2500775', nome : "Aparecida"},  
4813: {id : '1220', geocod : '2500809', nome : "Araçagi"},  
4814: {id : '1155', geocod : '2500908', nome : "Arara"},  
4815: {id : '1162', geocod : '2501005', nome : "Araruna"},  
4816: {id : '1154', geocod : '2501104', nome : "Areia"},  
4817: {id : '1000', geocod : '2501153', nome : "Areia de Baraúnas"},  
4818: {id : '1150', geocod : '2501203', nome : "Areial"},  
4819: {id : '1141', geocod : '2501302', nome : "Aroeiras"},  
4820: {id : '1004', geocod : '2501351', nome : "Assunção"},  
4821: {id : '1260', geocod : '2501401', nome : "Baía da Traição"},  
4822: {id : '5850', geocod : '2503308', nome : "Balanços"},  
4823: {id : '1227', geocod : '2501500', nome : "Bananeiras"},  
4824: {id : '1049', geocod : '2501534', nome : "Baraúna"},  
4825: {id : '5891', geocod : '2509305', nome : "Barra de Camaratuba"},  
4826: {id : '1056', geocod : '2501609', nome : "Barra de Santa Rosa"},  
4827: {id : '1131', geocod : '2501575', nome : "Barra de Santana"},  
4828: {id : '1025', geocod : '2501708', nome : "Barra de São Miguel"},  
4829: {id : '5902', geocod : '2512903', nome : "Barra do Mamanguape"},  
4830: {id : '1253', geocod : '2501807', nome : "Bayeux"},  
4831: {id : '1230', geocod : '2501906', nome : "Belém"},  
4832: {id : '1061', geocod : '2502003', nome : "Belém do Brejo do Cruz"},  
4833: {id : '826', geocod : '2502052', nome : "Bernardino Batista"},  
4834: {id : '799', geocod : '2502102', nome : "Boa Ventura"},  
4835: {id : '1038', geocod : '2502151', nome : "Boa Vista"},  
4836: {id : '812', geocod : '2502201', nome : "Bom Jesus"},  
4837: {id : '6051', geocod : '2514503', nome : "Bom Jesus - São José de Piranha"},  
4838: {id : '855', geocod : '2502300', nome : "Bom Sucesso"},  
4839: {id : '6069', geocod : '2516102', nome : "Bom Sucesso - Soledade"},  
4840: {id : '814', geocod : '2502409', nome : "Bonito de Santa Fé"},  
4841: {id : '1031', geocod : '2502508', nome : "Boqueirão"},  
4842: {id : '1225', geocod : '2502706', nome : "Borborema"},  
4843: {id : '970', geocod : '2502805', nome : "Brejo do Cruz"},  
4844: {id : '857', geocod : '2502904', nome : "Brejo dos Santos"},  
4845: {id : '1121', geocod : '2503001', nome : "Caaporã"},  
4846: {id : '1027', geocod : '2503100', nome : "Cabaceiras"},  
4847: {id : '1256', geocod : '2503209', nome : "Cabedelo"},  
4848: {id : '6156', geocod : '2506301', nome : "Cachoeira"},  
4849: {id : '811', geocod : '2503308', nome : "Cachoeira dos Índios"},  
4850: {id : '6174', geocod : '2506608', nome : "Cachoeirinha"},  
4851: {id : '975', geocod : '2503407', nome : "Cacimba de Areia"},  
4852: {id : '1160', geocod : '2503506', nome : "Cacimba de Dentro"},  
4853: {id : '977', geocod : '2503555', nome : "Cacimbas"},  
4854: {id : '1233', geocod : '2503605', nome : "Caiçara"},  
4855: {id : '813', geocod : '2503704', nome : "Cajazeiras"},  
4856: {id : '843', geocod : '2503753', nome : "Cajazeirinhas"},  
4857: {id : '1213', geocod : '2503803', nome : "Caldas Brandão"},  
4858: {id : '999', geocod : '2503902', nome : "Camalaú"},  
4859: {id : '1143', geocod : '2504009', nome : "Campina Grande"},  
4860: {id : '5555', geocod : '2516409', nome : "Campo de Santana"},  
4861: {id : '6250', geocod : '2506905', nome : "Campo Grande"},  
4862: {id : '6264', geocod : '2509057', nome : "Camurupim"},  
4863: {id : '1223', geocod : '2504033', nome : "Capim"},  
4864: {id : '1022', geocod : '2504074', nome : "Caraúbas"},  
4865: {id : '6350', geocod : '2504405', nome : "Cardoso"},  
4866: {id : '818', geocod : '2504108', nome : "Carrapateira"},  
4867: {id : '1159', geocod : '2504157', nome : "Casserengue"},  
4868: {id : '954', geocod : '2504207', nome : "Catingueira"},  
4869: {id : '6396', geocod : '2504009', nome : "Catolé"},  
4870: {id : '969', geocod : '2504306', nome : "Catolé do Rocha"},  
4871: {id : '1032', geocod : '2504355', nome : "Caturité"},  
4872: {id : '6424', geocod : '2500304', nome : "Cepilho"},  
4873: {id : '6447', geocod : '2509909', nome : "Chã do Rocha"},  
4874: {id : '769', geocod : '2504405', nome : "Conceição"},  
4875: {id : '956', geocod : '2504504', nome : "Condado"},  
4876: {id : '1251', geocod : '2504603', nome : "Conde"},  
4877: {id : '1020', geocod : '2504702', nome : "Congo"},  
4878: {id : '842', geocod : '2504801', nome : "Coremas"},  
4879: {id : '6537', geocod : '2504306', nome : "Coronel Maia"},  
4880: {id : '1021', geocod : '2504850', nome : "Coxixola"},  
4881: {id : '1250', geocod : '2504900', nome : "Cruz do Espírito Santo"},  
4882: {id : '1040', geocod : '2505006', nome : "Cubati"},  
4883: {id : '1057', geocod : '2505105', nome : "Cuité"},  
4884: {id : '1221', geocod : '2505238', nome : "Cuité de Mamanguape"},  
4885: {id : '1207', geocod : '2505204', nome : "Cuitegi"},  
4886: {id : '1241', geocod : '2505279', nome : "Curral de Cima"},  
4887: {id : '797', geocod : '2505303', nome : "Curral Velho"},  
4888: {id : '1157', geocod : '2505352', nome : "Damião"},  
4889: {id : '976', geocod : '2505402', nome : "Desterro"},  
4890: {id : '798', geocod : '2505600', nome : "Diamante"},  
4891: {id : '1228', geocod : '2505709', nome : "Dona Inês"},  
4892: {id : '1234', geocod : '2505808', nome : "Duas Estradas"},  
4893: {id : '953', geocod : '2505907', nome : "Emas"},  
4894: {id : '6731', geocod : '2503704', nome : "Engenheiro ávidos"},  
4895: {id : '1152', geocod : '2506004', nome : "Esperança"},  
4896: {id : '1140', geocod : '2506103', nome : "Fagundes"},  
4897: {id : '6874', geocod : '2503308', nome : "Fátima"},  
4898: {id : '1046', geocod : '2506202', nome : "Frei Martinho"},  
4899: {id : '1138', geocod : '2506251', nome : "Gado Bravo"},  
4900: {id : '6880', geocod : '2504009', nome : "Galante"},  
4901: {id : '6887', geocod : '2506806', nome : "Gameleira"},  
4902: {id : '1210', geocod : '2506301', nome : "Guarabira"},  
4903: {id : '6978', geocod : '2506905', nome : "Guarita"},  
4904: {id : '1128', geocod : '2506400', nome : "Gurinhém"},  
4905: {id : '1035', geocod : '2506509', nome : "Gurjão"},  
4906: {id : '770', geocod : '2506608', nome : "Ibiara"},  
4907: {id : '838', geocod : '2502607', nome : "Igaracy"},  
4908: {id : '992', geocod : '2506707', nome : "Imaculada"},  
4909: {id : '1125', geocod : '2506806', nome : "Ingá"},  
4910: {id : '1211', geocod : '2506905', nome : "Itabaiana"},  
4911: {id : '7183', geocod : '2504207', nome : "Itajubatiba"},  
4912: {id : '837', geocod : '2507002', nome : "Itaporanga"},  
4913: {id : '1222', geocod : '2507101', nome : "Itapororoca"},  
4914: {id : '1183', geocod : '2507200', nome : "Itatuba"},  
4915: {id : '1244', geocod : '2507309', nome : "Jacaraú"},  
4916: {id : '7266', geocod : '2504603', nome : "Jacumã"},  
4917: {id : '856', geocod : '2507408', nome : "Jericó"},  
4918: {id : '5484', geocod : '2507507', nome : "João Pessoa"},  
4919: {id : '1126', geocod : '2507606', nome : "Juarez Távora"},  
4920: {id : '1036', geocod : '2507705', nome : "Juazeirinho"},  
4921: {id : '1005', geocod : '2507804', nome : "Junco do Seridó"},  
4922: {id : '1197', geocod : '2507903', nome : "Juripiranga"},  
4923: {id : '808', geocod : '2508000', nome : "Juru"},  
4924: {id : '853', geocod : '2508109', nome : "Lagoa"},  
4925: {id : '1242', geocod : '2508208', nome : "Lagoa de Dentro"},  
4926: {id : '1144', geocod : '2508307', nome : "Lagoa Seca"},  
4927: {id : '848', geocod : '2508406', nome : "Lastro"},  
4928: {id : '950', geocod : '2508505', nome : "Livramento"},  
4929: {id : '1235', geocod : '2508554', nome : "Logradouro"},  
4930: {id : '1255', geocod : '2508604', nome : "Lucena"},  
4931: {id : '958', geocod : '2508703', nome : "Mãe d'Água"},  
4932: {id : '7502', geocod : '2502706', nome : "Maia"},  
4933: {id : '957', geocod : '2508802', nome : "Malta"},  
4934: {id : '1254', geocod : '2508901', nome : "Mamanguape"},  
4935: {id : '794', geocod : '2509008', nome : "Manaíra"},  
4936: {id : '1258', geocod : '2509057', nome : "Marcação"},  
4937: {id : '1219', geocod : '2509107', nome : "Mari"},  
4938: {id : '7572', geocod : '2502508', nome : "Marinho"},  
4939: {id : '819', geocod : '2509156', nome : "Marizópolis"},  
4940: {id : '1146', geocod : '2509206', nome : "Massaranduba"},  
4941: {id : '7595', geocod : '2501104', nome : "Mata Limpa"},  
4942: {id : '7596', geocod : '2517001', nome : "Mata Virgem"},  
4943: {id : '1259', geocod : '2509305', nome : "Mataraca"},  
4944: {id : '1148', geocod : '2509339', nome : "Matinhas"},  
4945: {id : '967', geocod : '2509370', nome : "Mato Grosso"},  
4946: {id : '960', geocod : '2509396', nome : "Maturéia"},  
4947: {id : '7615', geocod : '2505105', nome : "Melo"},  
4948: {id : '7629', geocod : '2510907', nome : "Mimoso"},  
4949: {id : '1127', geocod : '2509404', nome : "Mogeiro"},  
4950: {id : '1145', geocod : '2509503', nome : "Montadas"},  
4951: {id : '810', geocod : '2509602', nome : "Monte Horebe"},  
4952: {id : '998', geocod : '2509701', nome : "Monteiro"},  
4953: {id : '7708', geocod : '2504405', nome : "Montevidéo"},  
4954: {id : '1209', geocod : '2509800', nome : "Mulungu"},  
4955: {id : '7748', geocod : '2501104', nome : "Muquém"},  
4956: {id : '1181', geocod : '2509909', nome : "Natuba"},  
4957: {id : '7769', geocod : '2512002', nome : "Nazaré"},  
4958: {id : '820', geocod : '2510006', nome : "Nazarezinho"},  
4959: {id : '7799', geocod : '2513703', nome : "Nossa Senhora do Livramento"},  
4960: {id : '1051', geocod : '2510105', nome : "Nova Floresta"},  
4961: {id : '800', geocod : '2510204', nome : "Nova Olinda"},  
4962: {id : '1047', geocod : '2510303', nome : "Nova Palmeira"},  
4963: {id : '952', geocod : '2510402', nome : "Olho d'Água"},  
4964: {id : '1041', geocod : '2510501', nome : "Olivedos"},  
4965: {id : '945', geocod : '2510600', nome : "Ouro Velho"},  
4966: {id : '7937', geocod : '2515500', nome : "Palestina do Cariri"},  
4967: {id : '7943', geocod : '2506707', nome : "Palmeira"},  
4968: {id : '1033', geocod : '2510659', nome : "Parari"},  
4969: {id : '978', geocod : '2510709', nome : "Passagem"},  
4970: {id : '963', geocod : '2510808', nome : "Patos"},  
4971: {id : '965', geocod : '2510907', nome : "Paulista"},  
4972: {id : '801', geocod : '2511004', nome : "Pedra Branca"},  
4973: {id : '1043', geocod : '2511103', nome : "Pedra Lavrada"},  
4974: {id : '1119', geocod : '2511202', nome : "Pedras de Fogo"},  
4975: {id : '1243', geocod : '2512721', nome : "Pedro Régis"},  
4976: {id : '8070', geocod : '2509008', nome : "Pelo Sinal"},  
4977: {id : '841', geocod : '2511301', nome : "Piancó"},  
4978: {id : '8102', geocod : '2504306', nome : "Picos"},  
4979: {id : '1050', geocod : '2511400', nome : "Picuí"},  
4980: {id : '1214', geocod : '2511509', nome : "Pilar"},  
4981: {id : '1205', geocod : '2511608', nome : "Pilões"},  
4982: {id : '1208', geocod : '2511707', nome : "Pilõezinhos"},  
4983: {id : '8114', geocod : '2504702', nome : "Pindurão"},  
4984: {id : '8140', geocod : '2516300', nome : "Pio X"},  
4985: {id : '1229', geocod : '2511806', nome : "Pirpirituba"},  
4986: {id : '8167', geocod : '2508901', nome : "Pitanga da Estrada"},  
4987: {id : '1124', geocod : '2511905', nome : "Pitimbu"},  
4988: {id : '8170', geocod : '2513604', nome : "Pitombeira de Dentro"},  
4989: {id : '1054', geocod : '2512002', nome : "Pocinhos"},  
4990: {id : '831', geocod : '2512036', nome : "Poço Dantas"},  
4991: {id : '828', geocod : '2512077', nome : "Poço de José de Moura"},  
4992: {id : '854', geocod : '2512101', nome : "Pombal"},  
4993: {id : '8207', geocod : '2515807', nome : "Pontina"},  
4994: {id : '8271', geocod : '2508604', nome : "Praia de Fagundes"},  
4995: {id : '946', geocod : '2512200', nome : "Prata"},  
4996: {id : '803', geocod : '2512309', nome : "Princesa Isabel"},  
4997: {id : '1142', geocod : '2512408', nome : "Puxinanã"},  
4998: {id : '1139', geocod : '2512507', nome : "Queimadas"},  
4999: {id : '979', geocod : '2512606', nome : "Quixabá"},  
5000: {id : '1156', geocod : '2512705', nome : "Remígio"},  
5001: {id : '8351', geocod : '2515302', nome : "Renascença"},  
5002: {id : '1236', geocod : '2512747', nome : "Riachão"},  
5003: {id : '1147', geocod : '2512754', nome : "Riachão do Bacamarte"},  
5004: {id : '1216', geocod : '2512762', nome : "Riachão do Poço"},  
5005: {id : '1030', geocod : '2512788', nome : "Riacho de São Antônio"},  
5006: {id : '968', geocod : '2512804', nome : "Riacho dos Cavalos"},  
5007: {id : '1257', geocod : '2512903', nome : "Rio Tinto"},  
5008: {id : '8463', geocod : '2501906', nome : "Rua Nova"},  
5009: {id : '8474', geocod : '2508901', nome : "Salema"},  
5010: {id : '1001', geocod : '2513000', nome : "Salgadinho"},  
5011: {id : '1186', geocod : '2513109', nome : "Salgado de São Félix"},  
5012: {id : '1134', geocod : '2513158', nome : "Santa Cecília"},  
5013: {id : '850', geocod : '2513208', nome : "Santa Cruz"},  
5014: {id : '8531', geocod : '2510808', nome : "Santa Gertrudes"},  
5015: {id : '824', geocod : '2513307', nome : "Santa Helena"},  
5016: {id : '767', geocod : '2513356', nome : "Santa Inês"},  
5017: {id : '1003', geocod : '2513406', nome : "Santa Luzia"},  
5018: {id : '8549', geocod : '2515500', nome : "Santa Luzia do Cariri"},  
5019: {id : '8558', geocod : '2514107', nome : "Santa Maria"},  
5020: {id : '1252', geocod : '2513703', nome : "Santa Rita"},  
5021: {id : '959', geocod : '2513802', nome : "Santa Teresinha"},  
5022: {id : '771', geocod : '2513505', nome : "Santana de Mangueira"},  
5023: {id : '807', geocod : '2513604', nome : "Santana dos Garrotes"},  
5024: {id : '829', geocod : '2513653', nome : "Santarém"},  
5025: {id : '1034', geocod : '2513851', nome : "Santo André"},  
5026: {id : '955', geocod : '2513927', nome : "São Bentinho"},  
5027: {id : '972', geocod : '2513901', nome : "São Bento"},  
5028: {id : '852', geocod : '2513968', nome : "São Domingos de Pombal"},  
5029: {id : '1026', geocod : '2513943', nome : "São Domingos do Cariri"},  
5030: {id : '846', geocod : '2513984', nome : "São Francisco"},  
5031: {id : '1024', geocod : '2514008', nome : "São João do Cariri"},  
5032: {id : '827', geocod : '2500700', nome : "São João do Rio do Peixe"},  
5033: {id : '1862', geocod : '2514107', nome : "São João do Tigre"},  
5034: {id : '840', geocod : '2514206', nome : "São José da Lagoa Tapada"},  
5035: {id : '8900', geocod : '2504009', nome : "São José da Mata"},  
5036: {id : '817', geocod : '2514305', nome : "São José de Caiana"},  
5037: {id : '964', geocod : '2514404', nome : "São José de Espinharas"},  
5038: {id : '8908', geocod : '2503308', nome : "São José de Marimbas"},  
5039: {id : '815', geocod : '2514503', nome : "São José de Piranhas"},  
5040: {id : '796', geocod : '2514552', nome : "São José de Princesa"},  
5041: {id : '962', geocod : '2514602', nome : "São José do Bonfim"},  
5042: {id : '1065', geocod : '2514651', nome : "São José do Brejo do Cruz"},  
5043: {id : '1013', geocod : '2514701', nome : "São José do Sabugi"},  
5044: {id : '951', geocod : '2514800', nome : "São José dos Cordeiros"},  
5045: {id : '1212', geocod : '2514453', nome : "São José dos Ramos"},  
5046: {id : '980', geocod : '2514909', nome : "São Mamede"},  
5047: {id : '1215', geocod : '2515005', nome : "São Miguel de Taipu"},  
5048: {id : '1151', geocod : '2515104', nome : "São Sebastião de Lagoa de Ro"},  
5049: {id : '1859', geocod : '2515203', nome : "São Sebastião do Umbuzeiro"},  
5050: {id : '1218', geocod : '2515302', nome : "Sapé"},  
5051: {id : '1106', geocod : '2515401', nome : "Seridó"},  
5052: {id : '8727', geocod : '2511400', nome : "Seridó - Picuí"},  
5053: {id : '1023', geocod : '2515500', nome : "Serra Branca"},  
5054: {id : '1232', geocod : '2515609', nome : "Serra da Raiz"},  
5055: {id : '816', geocod : '2515708', nome : "Serra Grande"},  
5056: {id : '1149', geocod : '2515807', nome : "Serra Redonda"},  
5057: {id : '1224', geocod : '2515906', nome : "Serraria"},  
5058: {id : '1231', geocod : '2515930', nome : "Sertãozinho"},  
5059: {id : '1217', geocod : '2515971', nome : "Sobrado"},  
5060: {id : '8782', geocod : '2510402', nome : "Socorro"},  
5061: {id : '1226', geocod : '2516003', nome : "Solânea"},  
5062: {id : '1039', geocod : '2516102', nome : "Soledade"},  
5063: {id : '1048', geocod : '2516151', nome : "Sossêgo"},  
5064: {id : '844', geocod : '2516201', nome : "Sousa"},  
5065: {id : '8804', geocod : '2515500', nome : "Sucuru"},  
5066: {id : '949', geocod : '2516300', nome : "Sumé"},  
5067: {id : '9132', geocod : '2501500', nome : "Tabuleiro"},  
5068: {id : '1002', geocod : '2516508', nome : "Taperoá"},  
5069: {id : '806', geocod : '2516607', nome : "Tavares"},  
5070: {id : '961', geocod : '2516706', nome : "Teixeira"},  
5071: {id : '1037', geocod : '2516755', nome : "Tenório"},  
5072: {id : '9205', geocod : '2512721', nome : "Timbó"},  
5073: {id : '825', geocod : '2516805', nome : "Triunfo"},  
5074: {id : '830', geocod : '2516904', nome : "Uiraúna"},  
5075: {id : '9282', geocod : '2500700', nome : "Umari"},  
5076: {id : '9289', geocod : '2517001', nome : "Umburetama"},  
5077: {id : '1137', geocod : '2517001', nome : "Umbuzeiro"},  
5078: {id : '1009', geocod : '2517100', nome : "Várzea"},  
5079: {id : '9441', geocod : '2512101', nome : "Várzea Comprida"},  
5080: {id : '9341', geocod : '2505600', nome : "Vazante"},  
5081: {id : '847', geocod : '2517209', nome : "Vieirópolis"},  
5082: {id : '966', geocod : '2505501', nome : "Vista Serrana"},  
5083: {id : '1860', geocod : '2517407', nome : "Zabelê"}  
5084: ],  
5085: PE : [  
5086: // UF = PE  
5087: {id : '1199', geocod : '2600054', nome : "Abreu e Lima"},  
5088: {id : '987', geocod : '2600104', nome : "Afogados da Ingazeira"},  
5089: {id : '1590', geocod : '2600203', nome : "Afrânio"},  
5090: {id : '1891', geocod : '2600302', nome : "Agrestina"},  
5091: {id : '9469', geocod : '2601706', nome : "Água Fria"},  
5092: {id : '1914', geocod : '2600401', nome : "Água Preta"},  
5093: {id : '1777', geocod : '2600500', nome : "Águas Belas"},  
5094: {id : '5594', geocod : '2605707', nome : "Airi"},  
5095: {id : '1785', geocod : '2600609', nome : "Alagoinha"},  
5096: {id : '5605', geocod : '2614105', nome : "Albuquerque Né"},  
5097: {id : '5623', geocod : '2614105', nome : "Algodões"},  
5098: {id : '1193', geocod : '2600708', nome : "Aliança"},  
5099: {id : '1843', geocod : '2600807', nome : "Altinho"},  
5100: {id : '5633', geocod : '2602308', nome : "Alto Bonito"},  
5101: {id : '1878', geocod : '2600906', nome : "Amaraji"},  
5102: {id : '5678', geocod : '2604908', nome : "Ameixas"},  
5103: {id : '1838', geocod : '2601003', nome : "Angelim"},  
5104: {id : '5729', geocod : '2606101', nome : "Apoti"},  
5105: {id : '1200', geocod : '2601052', nome : "Araçoiaba"},  
5106: {id : '707', geocod : '2601102', nome : "Araripina"},  
5107: {id : '1782', geocod : '2601201', nome : "Arcoverde"},  
5108: {id : '5776', geocod : '2611804', nome : "Aripibu"},  
5109: {id : '5781', geocod : '2600203', nome : "Arizona"},  
5110: {id : '1899', geocod : '2601300', nome : "Barra de Guabiraba"},  
5111: {id : '5894', geocod : '2609907', nome : "Barra de São Pedro"},  
5112: {id : '5896', geocod : '2602100', nome : "Barra do Brejo"},  
5113: {id : '5897', geocod : '2600302', nome : "Barra do Chata"},  
5114: {id : '5899', geocod : '2602605', nome : "Barra do Farias"},  
5115: {id : '5901', geocod : '2600302', nome : "Barra do Jardim"},  
5116: {id : '5903', geocod : '2613305', nome : "Barra do Riachão"},  
5117: {id : '5905', geocod : '2614204', nome : "Barra do Sirinhaém"},  
5118: {id : '1919', geocod : '2601409', nome : "Barreiros"},  
5119: {id : '5945', geocod : '2601508', nome : "Batateira"},  
5120: {id : '1896', geocod : '2601508', nome : "Belém de Maria"},  
5121: {id : '1657', geocod : '2601607', nome : "Belém de São Francisco"},  
5122: {id : '1865', geocod : '2601706', nome : "Belo Jardim"},  
5123: {id : '5993', geocod : '2610509', nome : "Bengalas"},  
5124: {id : '5994', geocod : '2602308', nome : "Bentevi"},  
5125: {id : '5998', geocod : '2613909', nome : "Bernardo Vieira"},  
5126: {id : '1856', geocod : '2601805', nome : "Betânia"},  
5127: {id : '1928', geocod : '2601904', nome : "Bezerros"},  
5128: {id : '6009', geocod : '2602209', nome : "Bizarra"},  
5129: {id : '6037', geocod : '2601904', nome : "Boas Novas"},  
5130: {id : '643', geocod : '2602001', nome : "Bodocó"},  
5131: {id : '1817', geocod : '2602100', nome : "Bom Conselho"},  
5132: {id : '1180', geocod : '2602209', nome : "Bom Jardim"},  
5133: {id : '6060', geocod : '2613503', nome : "Bom Nome"},  
5134: {id : '6073', geocod : '2613602', nome : "Bonfim"},  
5135: {id : '1898', geocod : '2602308', nome : "Bonito"},  
5136: {id : '1822', geocod : '2602407', nome : "Brejão"},  
5137: {id : '997', geocod : '2602506', nome : "Brejinho"},  
5138: {id : '1866', geocod : '2602605', nome : "Brejo da Madre de Deus"},  
5139: {id : '1188', geocod : '2602704', nome : "Buenos Aires"},  
5140: {id : '1781', geocod : '2602803', nome : "Buíque"},  
5141: {id : '6146', geocod : '2605152', nome : "Caatinga Grande"},  
5142: {id : '6147', geocod : '2603108', nome : "Cabanas"},  
5143: {id : '1937', geocod : '2602902', nome : "Cabo de Santo Agostinho"},  
5144: {id : '1656', geocod : '2603009', nome : "Cabrobó"},  
5145: {id : '6172', geocod : '2600203', nome : "Cachoeira do Roberto"},  
5146: {id : '1556', geocod : '2603108', nome : "Cachoeirinha"},  
5147: {id : '1829', geocod : '2603207', nome : "Caetés"},  
5148: {id : '6199', geocod : '2613909', nome : "Caiçarinha da Penha"},  
5149: {id : '1836', geocod : '2603306', nome : "Calçado"},  
5150: {id : '6209', geocod : '2602100', nome : "Caldeirões"},  
5151: {id : '792', geocod : '2603405', nome : "Calumbi"},  
5152: {id : '1940', geocod : '2603454', nome : "Camaragibe"},  
5153: {id : '6224', geocod : '2614204', nome : "Camela"},  
5154: {id : '1929', geocod : '2603504', nome : "Camocim de São Félix"},  
5155: {id : '1195', geocod : '2603603', nome : "Camutanga"},  
5156: {id : '6278', geocod : '2615706', nome : "Canaã"},  
5157: {id : '1839', geocod : '2603702', nome : "Canhotinho"},  
5158: {id : '1830', geocod : '2603801', nome : "Capoeiras"},  
5159: {id : '6346', geocod : '2612604', nome : "Caraíbas"},  
5160: {id : '6329', geocod : '2614808', nome : "Caraibeiras"},  
5161: {id : '6339', geocod : '2604106', nome : "Carapotós"},  
5162: {id : '6351', geocod : '2606200', nome : "Caricé"},  
5163: {id : '6352', geocod : '2601409', nome : "Carimã"},  
5164: {id : '804', geocod : '2603900', nome : "Carnaíba"},  
5165: {id : '1854', geocod : '2603926', nome : "Carnaubeira da Penha"},  
5166: {id : '6363', geocod : '2602803', nome : "Carneiro"},  
5167: {id : '1190', geocod : '2604007', nome : "Carpina"},  
5168: {id : '1926', geocod : '2604106', nome : "Caruaru"},  
5169: {id : '1136', geocod : '2604155', nome : "Casinhas"},  
5170: {id : '1895', geocod : '2604205', nome : "Catende"},  
5171: {id : '6391', geocod : '2602803', nome : "Catimbau"},  
5172: {id : '6406', geocod : '2607901', nome : "Cavaleiro"},  
5173: {id : '776', geocod : '2604304', nome : "Cedro"},  
5174: {id : '1936', geocod : '2604403', nome : "Chã de Alegria"},  
5175: {id : '1879', geocod : '2604502', nome : "Chã Grande"},  
5176: {id : '6450', geocod : '2610905', nome : "Cimbres"},  
5177: {id : '6460', geocod : '2602001', nome : "Claranã"},  
5178: {id : '6469', geocod : '2611903', nome : "Cocaú"},  
5179: {id : '6501', geocod : '2612208', nome : "Conceição das Crioulas"},  
5180: {id : '1117', geocod : '2604601', nome : "Condado"},  
5181: {id : '1826', geocod : '2604700', nome : "Correntes"},  
5182: {id : '1917', geocod : '2604809', nome : "Cortês"},  
5183: {id : '6559', geocod : '2611705', nome : "Couro dAntas"},  
5184: {id : '6582', geocod : '2608750', nome : "Cristália"},  
5185: {id : '6587', geocod : '2615300', nome : "Cruanji"},  
5186: {id : '6598', geocod : '2610202', nome : "Cruzes"},  
5187: {id : '6602', geocod : '2605905', nome : "Cuiambuca"},  
5188: {id : '1931', geocod : '2604908', nome : "Cumaru"},  
5189: {id : '1890', geocod : '2605004', nome : "Cupira"},  
5190: {id : '6614', geocod : '2600500', nome : "Curral Novo"},  
5191: {id : '6615', geocod : '2611101', nome : "Curral Queimado"},  
5192: {id : '1857', geocod : '2605103', nome : "Custódia"},  
5193: {id : '6674', geocod : '2611309', nome : "Dois Leões"},  
5194: {id : '1548', geocod : '2605152', nome : "Dormentes"},  
5195: {id : '6744', geocod : '2608701', nome : "Entroncamento"},  
5196: {id : '1934', geocod : '2605202', nome : "Escada"},  
5197: {id : '6761', geocod : '2613008', nome : "Espírito Santo"},  
5198: {id : '759', geocod : '2605301', nome : "Exu"},  
5199: {id : '6800', geocod : '2602605', nome : "Fazenda Nova"},  
5200: {id : '1176', geocod : '2605400', nome : "Feira Nova"},  
5201: {id : '6809', geocod : '2602001', nome : "Feitoria"},  
5202: {id : '5556', geocod : '2605459', nome : "Fernando de Noronha"},  
5203: {id : '1196', geocod : '2605509', nome : "Ferreiros"},  
5204: {id : '802', geocod : '2605608', nome : "Flores"},  
5205: {id : '1671', geocod : '2605707', nome : "Floresta"},  
5206: {id : '1130', geocod : '2605806', nome : "Frei Miguelinho"},  
5207: {id : '6867', geocod : '2605202', nome : "Frexeiras"},  
5208: {id : '1921', geocod : '2605905', nome : "Gameleira"},  
5209: {id : '1831', geocod : '2606002', nome : "Garanhuns"},  
5210: {id : '1935', geocod : '2606101', nome : "Glória do Goitá"},  
5211: {id : '1120', geocod : '2606200', nome : "Goiana"},  
5212: {id : '6922', geocod : '2604106', nome : "Gonçalves Ferreira"},  
5213: {id : '758', geocod : '2606309', nome : "Granito"},  
5214: {id : '1877', geocod : '2606408', nome : "Gravatá"},  
5215: {id : '6935', geocod : '2615003', nome : "Gravatá do Ibiapina"},  
5216: {id : '6938', geocod : '2616001', nome : "Grotão"},  
5217: {id : '6956', geocod : '2602803', nome : "Guanumbi"},  
5218: {id : '7003', geocod : '2614105', nome : "Henrique Dias"},  
5219: {id : '7013', geocod : '2610806', nome : "Horizonte Alegre"},  
5220: {id : '7018', geocod : '2612307', nome : "Iatecá"},  
5221: {id : '1780', geocod : '2606507', nome : "Iati"},  
5222: {id : '1773', geocod : '2606606', nome : "Ibimirim"},  
5223: {id : '1555', geocod : '2606705', nome : "Ibirajuba"},  
5224: {id : '7037', geocod : '2607653', nome : "Ibiranga"},  
5225: {id : '7039', geocod : '2611804', nome : "Ibiratinga - Ribeirão"},  
5226: {id : '7045', geocod : '2603900', nome : "Ibitiranga - Carnaíba"},  
5227: {id : '7057', geocod : '2601607', nome : "Ibó"},  
5228: {id : '7059', geocod : '2610400', nome : "Icaiçara"},  
5229: {id : '7067', geocod : '2608602', nome : "Igapó"},  
5230: {id : '7069', geocod : '2608701', nome : "Igarapeassu"},  
5231: {id : '7070', geocod : '2612901', nome : "Igarapeba"},  
5232: {id : '1202', geocod : '2606804', nome : "Igarassu"},  
5233: {id : '988', geocod : '2606903', nome : "Iguaraci"},  
5234: {id : '1123', geocod : '2607604', nome : "Ilha de Itamaracá"},  
5235: {id : '1680', geocod : '2607000', nome : "Inajá"},  
5236: {id : '993', geocod : '2607109', nome : "Ingazeira"},  
5237: {id : '1925', geocod : '2607208', nome : "Ipojuca"},  
5238: {id : '710', geocod : '2607307', nome : "Ipubi"},  
5239: {id : '7133', geocod : '2614006', nome : "Ipuera"},  
5240: {id : '7135', geocod : '2615706', nome : "Iraguaçu"},  
5241: {id : '7136', geocod : '2606903', nome : "Irajaí"},  
5242: {id : '7144', geocod : '2606002', nome : "Iratama"},  
5243: {id : '1670', geocod : '2607406', nome : "Itacuruba"},  
5244: {id : '1772', geocod : '2607505', nome : "Itaíba"},  
5245: {id : '1118', geocod : '2607653', nome : "Itambé"},  
5246: {id : '947', geocod : '2607703', nome : "Itapetim"},  
5247: {id : '1203', geocod : '2607752', nome : "Itapissuma"},  
5248: {id : '1201', geocod : '2607802', nome : "Itaquitinga"},  
5249: {id : '7237', geocod : '2600807', nome : "Ituguaçu"},  
5250: {id : '7247', geocod : '2606903', nome : "Jabitacá"},  
5251: {id : '7248', geocod : '2607901', nome : "Jaboatão"},  
5252: {id : '1942', geocod : '2607901', nome : "Jaboatão dos Guararapes"},  
5253: {id : '7262', geocod : '2608008', nome : "Jacu"},  
5254: {id : '1894', geocod : '2607950', nome : "Jaqueira"},  
5255: {id : '1018', geocod : '2608008', nome : "Jataúba"},  
5256: {id : '1674', geocod : '2608057', nome : "Jatobá"},  
5257: {id : '7310', geocod : '2612406', nome : "Jenipapo"},  
5258: {id : '1116', geocod : '2608107', nome : "João Alfredo"},  
5259: {id : '1916', geocod : '2608206', nome : "Joaquim Nabuco"},  
5260: {id : '7332', geocod : '2611903', nome : "José da Costa"},  
5261: {id : '7330', geocod : '2611804', nome : "José Mariano"},  
5262: {id : '7380', geocod : '2602902', nome : "Juçaral"},  
5263: {id : '1832', geocod : '2608255', nome : "Jucati"},  
5264: {id : '7355', geocod : '2608008', nome : "Jundiá"},  
5265: {id : '1835', geocod : '2608305', nome : "Jupi"},  
5266: {id : '1840', geocod : '2608404', nome : "Jurema"},  
5267: {id : '7375', geocod : '2608750', nome : "Jutaí"},  
5268: {id : '7404', geocod : '2605152', nome : "Lagoa de Fora"},  
5269: {id : '7405', geocod : '2611705', nome : "Lagoa de João Carlos"},  
5270: {id : '7406', geocod : '2602100', nome : "Lagoa de São José"},  
5271: {id : '7409', geocod : '2601102', nome : "Lagoa do Barro"},  
5272: {id : '1187', geocod : '2608453', nome : "Lagoa do Carro"},  
5273: {id : '1177', geocod : '2608503', nome : "Lagoa do Itaenga"},  
5274: {id : '1821', geocod : '2608602', nome : "Lagoa do Ouro"},  
5275: {id : '7414', geocod : '2608701', nome : "Lagoa do Souza"},  
5276: {id : '1893', geocod : '2608701', nome : "Lagoa dos Gatos"},  
5277: {id : '1549', geocod : '2608750', nome : "Lagoa Grande"},  
5278: {id : '7416', geocod : '2605152', nome : "Lagoas"},  
5279: {id : '7421', geocod : '2605004', nome : "Laje de São José"},  
5280: {id : '7420', geocod : '2604205', nome : "Laje Grande"},  
5281: {id : '1837', geocod : '2608800', nome : "Lajedo"},  
5282: {id : '7431', geocod : '2604106', nome : "Lajedo do Cedro"},  
5283: {id : '1175', geocod : '2608909', nome : "Limoeiro"},  
5284: {id : '7465', geocod : '2615300', nome : "Livramento do Tiúma"},  
5285: {id : '7469', geocod : '2613909', nome : "Logradouro"},  
5286: {id : '7476', geocod : '2613909', nome : "Luanda"},  
5287: {id : '1185', geocod : '2609006', nome : "Macaparana"},  
5288: {id : '1182', geocod : '2609105', nome : "Machados"},  
5289: {id : '7501', geocod : '2600708', nome : "Macujê"},  
5290: {id : '7512', geocod : '2601607', nome : "Malhada Grande"},  
5291: {id : '1766', geocod : '2609154', nome : "Manari"},  
5292: {id : '7516', geocod : '2602605', nome : "Mandaçaia"},  
5293: {id : '7515', geocod : '2606408', nome : "Mandacaru"},  
5294: {id : '7529', geocod : '2604106', nome : "Maniçoba"},  
5295: {id : '1892', geocod : '2609204', nome : "Maraial"},  
5296: {id : '7549', geocod : '2605103', nome : "Maravilha"},  
5297: {id : '7630', geocod : '2614501', nome : "Mimoso"},  
5298: {id : '7636', geocod : '2606002', nome : "Miracica"},  
5299: {id : '1855', geocod : '2609303', nome : "Mirandiba"},  
5300: {id : '7710', geocod : '2601102', nome : "Morais"},  
5301: {id : '760', geocod : '2614303', nome : "Moreilândia"},  
5302: {id : '1938', geocod : '2609402', nome : "Moreno"},  
5303: {id : '7735', geocod : '2606606', nome : "Moxotó"},  
5304: {id : '7739', geocod : '2612406', nome : "Mulungu"},  
5305: {id : '7753', geocod : '2616308', nome : "Murupé"},  
5306: {id : '7757', geocod : '2610905', nome : "Mutuca"},  
5307: {id : '7764', geocod : '2601102', nome : "Nascente"},  
5308: {id : '7768', geocod : '2610707', nome : "Navarro"},  
5309: {id : '1192', geocod : '2609501', nome : "Nazaré da Mata"},  
5310: {id : '7771', geocod : '2605707', nome : "Nazaré do Pico"},  
5311: {id : '7789', geocod : '2613701', nome : "Nossa Senhora da Luz"},  
5312: {id : '7797', geocod : '2613909', nome : "Nossa Senhora do Carmo"},  
5313: {id : '7795', geocod : '2607208', nome : "Nossa Senhora do Ó"},  
5314: {id : '7819', geocod : '2610707', nome : "Nova Cruz"},  
5315: {id : '7872', geocod : '2603702', nome : "Olho Dagua de Dentro"},  
5316: {id : '1941', geocod : '2609600', nome : "Olinda"},  
5317: {id : '7885', geocod : '2614006', nome : "Ori"},  
5318: {id : '1179', geocod : '2609709', nome : "Orobó"},  
5319: {id : '1652', geocod : '2609808', nome : "Orocó"},  
5320: {id : '708', geocod : '2609907', nome : "Ouricuri"},  
5321: {id : '7930', geocod : '2614006', nome : "Pajeú"},  
5322: {id : '1913', geocod : '2610004', nome : "Palmares"},  
5323: {id : '1827', geocod : '2610103', nome : "Palmeirina"},  
5324: {id : '1842', geocod : '2610202', nome : "Panelas"},  
5325: {id : '8302', geocod : '2602605', nome : "Pão de Açucar"},  
5326: {id : '8303', geocod : '2611200', nome : "Pão de Açúcar de Poção"},  
5327: {id : '7960', geocod : '2610905', nome : "Papagaio"},  
5328: {id : '7963', geocod : '2603702', nome : "Paquevira"},  
5329: {id : '7995', geocod : '2612505', nome : "Pará"},  
5330: {id : '1820', geocod : '2610301', nome : "Paranatama"},  
5331: {id : '7979', geocod : '2610707', nome : "Paratibe"},  
5332: {id : '1852', geocod : '2610400', nome : "Parnamirim"},  
5333: {id : '8004', geocod : '2608008', nome : "Passagem do Tó"},  
5334: {id : '1114', geocod : '2610509', nome : "Passira"},  
5335: {id : '8032', geocod : '2611507', nome : "Pau Ferro"},  
5336: {id : '1178', geocod : '2610608', nome : "Paudalho"},  
5337: {id : '1198', geocod : '2610707', nome : "Paulista"},  
5338: {id : '1784', geocod : '2610806', nome : "Pedra"},  
5339: {id : '8086', geocod : '2600609', nome : "Perpétuo Socorro"},  
5340: {id : '1863', geocod : '2610905', nome : "Pesqueira"},  
5341: {id : '1668', geocod : '2611002', nome : "Petrolândia"},  
5342: {id : '1595', geocod : '2611101', nome : "Petrolina"},  
5343: {id : '8135', geocod : '2611705', nome : "Pinhões"},  
5344: {id : '8157', geocod : '2613800', nome : "Pirauá"},  
5345: {id : '8162', geocod : '2616407', nome : "Pirituba"},  
5346: {id : '1861', geocod : '2611200', nome : "Poção"},  
5347: {id : '8262', geocod : '2600203', nome : "Poção de Afrânio"},  
5348: {id : '8251', geocod : '2604700', nome : "Poço Comprido"},  
5349: {id : '8260', geocod : '2602803', nome : "Poço do Boi"},  
5350: {id : '8255', geocod : '2612505', nome : "Poço Fundo"},  
5351: {id : '1880', geocod : '2611309', nome : "Pombos"},  
5352: {id : '8192', geocod : '2606200', nome : "Ponta de Pedras"},  
5353: {id : '8204', geocod : '2602902', nome : "Ponte dos Carvalhos"},  
5354: {id : '8269', geocod : '2610707', nome : "Praia da Conceição"},  
5355: {id : '1932', geocod : '2611408', nome : "Primavera"},  
5356: {id : '1841', geocod : '2611507', nome : "Quipapá"},  
5357: {id : '8330', geocod : '2605103', nome : "Quitimbu"},  
5358: {id : '805', geocod : '2611533', nome : "Quixaba"},  
5359: {id : '8337', geocod : '2602100', nome : "Rainha Isabel"},  
5360: {id : '8338', geocod : '2611101', nome : "Rajada"},  
5361: {id : '8339', geocod : '2601102', nome : "Rancharia"},  
5362: {id : '5485', geocod : '2611606', nome : "Recife"},  
5363: {id : '1927', geocod : '2611705', nome : "Riacho das Almas"},  
5364: {id : '8368', geocod : '2608008', nome : "Riacho do Meio - Jataúba"},  
5365: {id : '8369', geocod : '2613602', nome : "Riacho do Meio - São José do"},  
5366: {id : '8360', geocod : '2613107', nome : "Riacho Fechado"},  
5367: {id : '8363', geocod : '2601607', nome : "Riacho Pequeno"},  
5368: {id : '1922', geocod : '2611804', nome : "Ribeirão"},  
5369: {id : '8426', geocod : '2614105', nome : "Rio da Barra"},  
5370: {id : '1920', geocod : '2611903', nome : "Rio Formoso"},  
5371: {id : '1930', geocod : '2612000', nome : "Sairé"},  
5372: {id : '1115', geocod : '2612109', nome : "Salgadinho"},  
5373: {id : '1853', geocod : '2612208', nome : "Salgueiro"},  
5374: {id : '1819', geocod : '2612307', nome : "Saloá"},  
5375: {id : '8478', geocod : '2610905', nome : "Salobro"},  
5376: {id : '1864', geocod : '2612406', nome : "Sanharó"},  
5377: {id : '1850', geocod : '2612455', nome : "Santa Cruz"},  
5378: {id : '793', geocod : '2612471', nome : "Santa Cruz da Baixa Verde"},  
5379: {id : '1019', geocod : '2612505', nome : "Santa Cruz do Capibaribe"},  
5380: {id : '1849', geocod : '2612554', nome : "Santa Filomena"},  
5381: {id : '1651', geocod : '2612604', nome : "Santa Maria da Boa Vista"},  
5382: {id : '1132', geocod : '2612703', nome : "Santa Maria do Cambucá"},  
5383: {id : '8569', geocod : '2606903', nome : "Santa Rita - Iguaraci"},  
5384: {id : '8570', geocod : '2613909', nome : "Santa Rita - Serra Talhada"},  
5385: {id : '994', geocod : '2612802', nome : "Santa Terezinha"},  
5386: {id : '8594', geocod : '2600401', nome : "Santa Terezinha - Água Preta"},  
5387: {id : '8608', geocod : '2605004', nome : "Santana de São Joaquim"},  
5388: {id : '8621', geocod : '2602902', nome : "Santo Agostinho"},  
5389: {id : '8628', geocod : '2616001', nome : "Santo Antônio"},  
5390: {id : '8646', geocod : '2606705', nome : "Santo Antônio das Queimadas"},  
5391: {id : '8673', geocod : '2610004', nome : "Santo Antônio dos Palmares"},  
5392: {id : '1889', geocod : '2612901', nome : "São Benedito do Sul"},  
5393: {id : '1833', geocod : '2613008', nome : "São Bento do Una"},  
5394: {id : '8827', geocod : '2601805', nome : "São Caetano do Navio"},  
5395: {id : '1868', geocod : '2613107', nome : "São Caitano"},  
5396: {id : '8837', geocod : '2610400', nome : "São Domingos"},  
5397: {id : '1834', geocod : '2613206', nome : "São João"},  
5398: {id : '1897', geocod : '2613305', nome : "São Joaquim do Monte"},  
5399: {id : '8886', geocod : '2610202', nome : "São José"},  
5400: {id : '1918', geocod : '2613404', nome : "São José da Coroa Grande"},  
5401: {id : '766', geocod : '2613503', nome : "São José do Belmonte"},  
5402: {id : '996', geocod : '2613602', nome : "São José do Egito"},  
5403: {id : '8987', geocod : '2610202', nome : "São Lázaro"},  
5404: {id : '1939', geocod : '2613701', nome : "São Lorenço da Mata"},  
5405: {id : '9012', geocod : '2606002', nome : "São Pedro"},  
5406: {id : '9031', geocod : '2610806', nome : "São Pedro do Cordeiro"},  
5407: {id : '9100', geocod : '2607703', nome : "São Vicente"},  
5408: {id : '1184', geocod : '2613800', nome : "São Vicente Ferrer"},  
5409: {id : '8694', geocod : '2601904', nome : "Sapucarana"},  
5410: {id : '8704', geocod : '2614857', nome : "Saué"},  
5411: {id : '8730', geocod : '2607307', nome : "Serra Branca"},  
5412: {id : '8742', geocod : '2601706', nome : "Serra do Vento"},  
5413: {id : '765', geocod : '2613909', nome : "Serra Talhada"},  
5414: {id : '761', geocod : '2614006', nome : "Serrita"},  
5415: {id : '8751', geocod : '2607307', nome : "Serrolândia"},  
5416: {id : '1858', geocod : '2614105', nome : "Sertânia"},  
5417: {id : '8758', geocod : '2609204', nome : "Sertãozinho de Baixo"},  
5418: {id : '8774', geocod : '2613800', nome : "Siriji"},  
5419: {id : '1924', geocod : '2614204', nome : "Sirinhaém"},  
5420: {id : '9120', geocod : '2605608', nome : "Sítio dos Nunes"},  
5421: {id : '989', geocod : '2614402', nome : "Solidão"},  
5422: {id : '1135', geocod : '2614501', nome : "Surubim"},  
5423: {id : '991', geocod : '2614600', nome : "Tabira"},  
5424: {id : '9128', geocod : '2605301', nome : "Tabocas"},  
5425: {id : '1867', geocod : '2614709', nome : "Tacaimbó"},  
5426: {id : '1675', geocod : '2614808', nome : "Tacaratu"},  
5427: {id : '1923', geocod : '2614857', nome : "Tamandaré"},  
5428: {id : '9143', geocod : '2612703', nome : "Tamboatá"},  
5429: {id : '9154', geocod : '2613107', nome : "Tapiraim"},  
5430: {id : '1028', geocod : '2615003', nome : "Taquaritinga do Norte"},  
5431: {id : '9177', geocod : '2613909', nome : "Tauapiranga"},  
5432: {id : '9185', geocod : '2606200', nome : "Tejucopapo"},  
5433: {id : '1818', geocod : '2615102', nome : "Terezinha"},  
5434: {id : '1851', geocod : '2615201', nome : "Terra Nova"},  
5435: {id : '1194', geocod : '2615300', nome : "Timbaúba"},  
5436: {id : '9207', geocod : '2605301', nome : "Timorante"},  
5437: {id : '1869', geocod : '2615409', nome : "Toritama"},  
5438: {id : '1191', geocod : '2615508', nome : "Tracunhaém"},  
5439: {id : '9229', geocod : '2605806', nome : "Trapiá"},  
5440: {id : '9250', geocod : '2606804', nome : "Três Ladeiras"},  
5441: {id : '709', geocod : '2615607', nome : "Trindade"},  
5442: {id : '795', geocod : '2615706', nome : "Triunfo"},  
5443: {id : '9261', geocod : '2609303', nome : "Tupanaci"},  
5444: {id : '1774', geocod : '2615805', nome : "Tupanatinga"},  
5445: {id : '9264', geocod : '2600708', nome : "Tupaoca"},  
5446: {id : '995', geocod : '2615904', nome : "Tuparetama"},  
5447: {id : '9291', geocod : '2612208', nome : "Umãs"},  
5448: {id : '9296', geocod : '2600708', nome : "Upatininga"},  
5449: {id : '9297', geocod : '2612604', nome : "Urimamã"},  
5450: {id : '9305', geocod : '2606408', nome : "Uruçu mirim"},  
5451: {id : '9299', geocod : '2608909', nome : "Urucuba"},  
5452: {id : '9337', geocod : '2613909', nome : "Varzinha"},  
5453: {id : '9338', geocod : '2612208', nome : "Vasques"},  
5454: {id : '9350', geocod : '2610400', nome : "Veneza"},  
5455: {id : '1783', geocod : '2616001', nome : "Venturosa"},  
5456: {id : '762', geocod : '2616100', nome : "Verdejante"},  
5457: {id : '1133', geocod : '2616183', nome : "Vertente do Lério"},  
5458: {id : '1129', geocod : '2616209', nome : "Vertentes"},  
5459: {id : '1189', geocod : '2616308', nome : "Vicência"},  
5460: {id : '9417', geocod : '2605301', nome : "Viração"},  
5461: {id : '1933', geocod : '2616407', nome : "Vitória de S. Antão"},  
5462: {id : '9428', geocod : '2611705', nome : "Vitorino"},  
5463: {id : '9438', geocod : '2608057', nome : "Volta do Moxotó"},  
5464: {id : '1911', geocod : '2616506', nome : "Xexéu"},  
5465: {id : '9456', geocod : '2601706', nome : "Xucuru"},  
5466: {id : '9460', geocod : '2605301', nome : "Zé Gomes"}  
5467: ],  
5468: PI : [  
5469: // UF = PI  
5470: {id : '1847', geocod : '2200053', nome : "Acauã"},  
5471: {id : '608', geocod : '2200103', nome : "Agricolândia"},  
5472: {id : '609', geocod : '2200202', nome : "Água Branca"},  
5473: {id : '703', geocod : '2200251', nome : "Alagoinha do Piauí"},  
5474: {id : '644', geocod : '2200277', nome : "Alegrete do Piauí"},  
5475: {id : '617', geocod : '2200301', nome : "Alto Longá"},  
5476: {id : '623', geocod : '2200400', nome : "Altos"},  
5477: {id : '1505', geocod : '2200459', nome : "Alvorada do Gurguéia"},  
5478: {id : '602', geocod : '2200509', nome : "Amarante"},  
5479: {id : '603', geocod : '2200608', nome : "Angical do Piauí"},  
5480: {id : '1508', geocod : '2200707', nome : "Anísio de Abreu"},  
5481: {id : '535', geocod : '2200806', nome : "Antônio Almeida"},  
5482: {id : '724', geocod : '2200905', nome : "Aroazes"},  
5483: {id : '5565', geocod : '2200954', nome : "Aroeiras do Itaim"},  
5484: {id : '573', geocod : '2201002', nome : "Arraial"},  
5485: {id : '729', geocod : '2201051', nome : "Assunção do Piauí"},  
5486: {id : '1501', geocod : '2201101', nome : "Avelino Lopes"},  
5487: {id : '526', geocod : '2201150', nome : "Baixa Grande do Ribeiro"},  
5488: {id : '588', geocod : '2201176', nome : "Barra D'Alcântara"},  
5489: {id : '627', geocod : '2201200', nome : "Barras"},  
5490: {id : '1489', geocod : '2201309', nome : "Barreiras do Piauí"},  
5491: {id : '613', geocod : '2201408', nome : "Barro Duro"},  
5492: {id : '465', geocod : '2201507', nome : "Batalha"},  
5493: {id : '664', geocod : '2201556', nome : "Bela Vista do Piauí"},  
5494: {id : '679', geocod : '2201572', nome : "Belém do Piauí"},  
5495: {id : '616', geocod : '2201606', nome : "Beneditinos"},  
5496: {id : '548', geocod : '2201705', nome : "Bertolínia"},  
5497: {id : '1848', geocod : '2201739', nome : "Betânia do Piauí"},  
5498: {id : '630', geocod : '2201770', nome : "Boa Hora"},  
5499: {id : '696', geocod : '2201804', nome : "Bocaina"},  
5500: {id : '1500', geocod : '2201903', nome : "Bom Jesus"},  
5501: {id : '299', geocod : '2201919', nome : "Bom Princípio do Piauí"},  
5502: {id : '1514', geocod : '2201929', nome : "Bonfim do Piauí"},  
5503: {id : '629', geocod : '2201945', nome : "Boqueirão do Piauí"},  
5504: {id : '935', geocod : '2201960', nome : "Brasileira"},  
5505: {id : '1524', geocod : '2201988', nome : "Brejo do Piauí"},  
5506: {id : '296', geocod : '2202000', nome : "Buriti dos Lopes"},  
5507: {id : '731', geocod : '2202026', nome : "Buriti dos Montes"},  
5508: {id : '626', geocod : '2202059', nome : "Cabeceiras do Piauí"},  
5509: {id : '576', geocod : '2202075', nome : "Cajazeiras do Piauí"},  
5510: {id : '305', geocod : '2202083', nome : "Cajueiro da Praia"},  
5511: {id : '712', geocod : '2202091', nome : "Caldeirão Grande do Piauí"},  
5512: {id : '667', geocod : '2202109', nome : "Campinas do Piauí"},  
5513: {id : '1844', geocod : '2202117', nome : "Campo Alegre do Fidalgo"},  
5514: {id : '699', geocod : '2202133', nome : "Campo Grande do Piauí"},  
5515: {id : '284', geocod : '2202174', nome : "Campo Largo do Piauí"},  
5516: {id : '625', geocod : '2202208', nome : "Campo Maior"},  
5517: {id : '550', geocod : '2202251', nome : "Canavieira"},  
5518: {id : '1525', geocod : '2202307', nome : "Canto do Buriti"},  
5519: {id : '735', geocod : '2202406', nome : "Capitão de Campos"},  
5520: {id : '1586', geocod : '2202455', nome : "Capitão Gervásio Oliveira"},  
5521: {id : '1506', geocod : '2202505', nome : "Caracol"},  
5522: {id : '295', geocod : '2202539', nome : "Caraúbas do Piauí"},  
5523: {id : '675', geocod : '2202554', nome : "Caridade do Piauí"},  
5524: {id : '727', geocod : '2202604', nome : "Castelo do Piauí"},  
5525: {id : '294', geocod : '2202653', nome : "Caxingó"},  
5526: {id : '297', geocod : '2202703', nome : "Cocal"},  
5527: {id : '642', geocod : '2202711', nome : "Cocal de Telha"},  
5528: {id : '298', geocod : '2202729', nome : "Cocal dos Alves"},  
5529: {id : '624', geocod : '2202737', nome : "Coivaras"},  
5530: {id : '1521', geocod : '2202752', nome : "Colônia do Gurguéia"},  
5531: {id : '574', geocod : '2202778', nome : "Colônia do Piauí"},  
5532: {id : '665', geocod : '2202802', nome : "Conceição do Canindé"},  
5533: {id : '1518', geocod : '2202851', nome : "Coronel José Dias"},  
5534: {id : '1487', geocod : '2202901', nome : "Corrente"},  
5535: {id : '1482', geocod : '2203008', nome : "Cristalândia do Piauí"},  
5536: {id : '1447', geocod : '2203107', nome : "Cristino Castro"},  
5537: {id : '1499', geocod : '2203206', nome : "Curimatá"},  
5538: {id : '1445', geocod : '2203230', nome : "Currais"},  
5539: {id : '676', geocod : '2203271', nome : "Curral Novo do Piauí"},  
5540: {id : '578', geocod : '2203255', nome : "Curralinhos"},  
5541: {id : '581', geocod : '2203305', nome : "Demerval Lobão"},  
5542: {id : '1519', geocod : '2203354', nome : "Dirceu Arcoverde"},  
5543: {id : '686', geocod : '2203404', nome : "Dom Expedito Lopes"},  
5544: {id : '1533', geocod : '2203453', nome : "Dom Inocêncio"},  
5545: {id : '741', geocod : '2203420', nome : "Domingos Mourão"},  
5546: {id : '583', geocod : '2203503', nome : "Elesbão Veloso"},  
5547: {id : '1523', geocod : '2203602', nome : "Eliseu Martins"},  
5548: {id : '136', geocod : '2203701', nome : "Esperantina"},  
5549: {id : '1511', geocod : '2203750', nome : "Fartura do Piauí"},  
5550: {id : '564', geocod : '2203800', nome : "Flores do Piauí"},  
5551: {id : '669', geocod : '2203859', nome : "Floresta do Piauí"},  
5552: {id : '561', geocod : '2203909', nome : "Floriano"},  
5553: {id : '587', geocod : '2204006', nome : "Francinópolis"},  
5554: {id : '572', geocod : '2204105', nome : "Francisco Ayres"},  
5555: {id : '645', geocod : '2204154', nome : "Francisco Macedo"},  
5556: {id : '701', geocod : '2204204', nome : "Francisco Santos"},  
5557: {id : '713', geocod : '2204303', nome : "Fronteiras"},  
5558: {id : '694', geocod : '2204352', nome : "Geminiano"},  
5559: {id : '1490', geocod : '2204402', nome : "Gilbués"},  
5560: {id : '557', geocod : '2204501', nome : "Guadalupe"},  
5561: {id : '1504', geocod : '2204550', nome : "Guaribas"},  
5562: {id : '611', geocod : '2204600', nome : "Hugo Napoleão"},  
5563: {id : '303', geocod : '2204659', nome : "Ilha Grande"},  
5564: {id : '690', geocod : '2204709', nome : "Inhuma"},  
5565: {id : '685', geocod : '2204808', nome : "Ipiranga do Piauí"},  
5566: {id : '670', geocod : '2204907', nome : "Isaías Coelho"},  
5567: {id : '672', geocod : '2205003', nome : "Itainópolis"},  
5568: {id : '553', geocod : '2205102', nome : "Itaueira"},  
5569: {id : '673', geocod : '2205151', nome : "Jacobina do Piauí"},  
5570: {id : '678', geocod : '2205201', nome : "Jaicós"},  
5571: {id : '640', geocod : '2205250', nome : "Jardim do Mulato"},  
5572: {id : '733', geocod : '2205276', nome : "Jatobá do Piauí"},  
5573: {id : '560', geocod : '2205300', nome : "Jerumenha"},  
5574: {id : '1520', geocod : '2205359', nome : "João Costa"},  
5575: {id : '142', geocod : '2205409', nome : "Joaquim Pires"},  
5576: {id : '138', geocod : '2205458', nome : "Joca Marques"},  
5577: {id : '620', geocod : '2205508', nome : "José de Freitas"},  
5578: {id : '732', geocod : '2205516', nome : "Juazeiro do Piauí"},  
5579: {id : '1498', geocod : '2205524', nome : "Júlio Borges"},  
5580: {id : '1507', geocod : '2205532', nome : "Jurema"},  
5581: {id : '622', geocod : '2205557', nome : "Lagoa Alegre"},  
5582: {id : '737', geocod : '2205573', nome : "Lagoa de São Francisco"},  
5583: {id : '1587', geocod : '2205565', nome : "Lagoa do Barro do Piauí"},  
5584: {id : '580', geocod : '2205581', nome : "Lagoa do Piauí"},  
5585: {id : '693', geocod : '2205599', nome : "Lagoa do Sítio"},  
5586: {id : '610', geocod : '2205540', nome : "Lagoinha do Piauí"},  
5587: {id : '554', geocod : '2205607', nome : "Landri Sales"},  
5588: {id : '304', geocod : '2205706', nome : "Luís Correia"},  
5589: {id : '139', geocod : '2205805', nome : "Luzilândia"},  
5590: {id : '288', geocod : '2205854', nome : "Madeiro"},  
5591: {id : '1522', geocod : '2205904', nome : "Manoel Emídio"},  
5592: {id : '706', geocod : '2205953', nome : "Marcolândia"},  
5593: {id : '555', geocod : '2206001', nome : "Marcos Parente"},  
5594: {id : '677', geocod : '2206050', nome : "Massapê do Piauí"},  
5595: {id : '285', geocod : '2206100', nome : "Matias Olímpio"},  
5596: {id : '639', geocod : '2206209', nome : "Miguel Alves"},  
5597: {id : '577', geocod : '2206308', nome : "Miguel Leão"},  
5598: {id : '740', geocod : '2206357', nome : "Milton Brandão"},  
5599: {id : '579', geocod : '2206407', nome : "Monsenhor Gil"},  
5600: {id : '702', geocod : '2206506', nome : "Monsenhor Hipólito"},  
5601: {id : '1491', geocod : '2206605', nome : "Monte Alegre do Piauí"},  
5602: {id : '7695', geocod : '2201739', nome : "Monte Orebe"},  
5603: {id : '1502', geocod : '2206654', nome : "Morro Cabeça no Tempo"},  
5604: {id : '137', geocod : '2206670', nome : "Morro do Chapéu do Piauí"},  
5605: {id : '143', geocod : '2206696', nome : "Murici dos Portelas"},  
5606: {id : '571', geocod : '2206704', nome : "Nazaré do Piauí"},  
5607: {id : '5561', geocod : '2206720', nome : "Nazária"},  
5608: {id : '628', geocod : '2206753', nome : "Nossa Senhora de Nazaré"},  
5609: {id : '282', geocod : '2206803', nome : "Nossa Senhora dos Remédios"},  
5610: {id : '1528', geocod : '2207959', nome : "Nova Santa Rita"},  
5611: {id : '691', geocod : '2206902', nome : "Novo Oriente do Piauí"},  
5612: {id : '726', geocod : '2206951', nome : "Novo Santo Antônio"},  
5613: {id : '575', geocod : '2207009', nome : "Oeiras"},  
5614: {id : '612', geocod : '2207108', nome : "Olho D'Água do Piauí"},  
5615: {id : '680', geocod : '2207207', nome : "Padre Marcos"},  
5616: {id : '568', geocod : '2207306', nome : "Paes Landim"},  
5617: {id : '565', geocod : '2207355', nome : "Pajeú do Piauí"},  
5618: {id : '1448', geocod : '2207405', nome : "Palmeira do Piauí"},  
5619: {id : '592', geocod : '2207504', nome : "Palmeirais"},  
5620: {id : '683', geocod : '2207553', nome : "Paquetá"},  
5621: {id : '1496', geocod : '2207603', nome : "Parnaguá"},  
5622: {id : '302', geocod : '2207702', nome : "Parnaíba"},  
5623: {id : '582', geocod : '2207751', nome : "Passagem Franca do Piauí"},  
5624: {id : '674', geocod : '2207777', nome : "Patos do Piauí"},  
5625: {id : '5499', geocod : '2207793', nome : "Pau D'arco do Piauí"},  
5626: {id : '1846', geocod : '2207801', nome : "Paulistana"},  
5627: {id : '551', geocod : '2207850', nome : "Pavussu"},  
5628: {id : '738', geocod : '2207900', nome : "Pedro II"},  
5629: {id : '1527', geocod : '2207934', nome : "Pedro Laurentino"},  
5630: {id : '688', geocod : '2208007', nome : "Picos"},  
5631: {id : '728', geocod : '2208106', nome : "Pimenteiras"},  
5632: {id : '715', geocod : '2208205', nome : "Pio IX"},  
5633: {id : '292', geocod : '2208304', nome : "Piracuruca"},  
5634: {id : '736', geocod : '2208403', nome : "Piripiri"},  
5635: {id : '283', geocod : '2208502', nome : "Porto"},  
5636: {id : '536', geocod : '2208551', nome : "Porto Alegre do Piauí"},  
5637: {id : '618', geocod : '2208601', nome : "Prata do Piauí"},  
5638: {id : '1589', geocod : '2208650', nome : "Queimada Nova"},  
5639: {id : '1497', geocod : '2208700', nome : "Redenção do Gurguéia"},  
5640: {id : '604', geocod : '2208809', nome : "Regeneração"},  
5641: {id : '1495', geocod : '2208858', nome : "Riacho Frio"},  
5642: {id : '566', geocod : '2208874', nome : "Ribeira do Piauí"},  
5643: {id : '527', geocod : '2208908', nome : "Ribeiro Gonçalves"},  
5644: {id : '552', geocod : '2209005', nome : "Rio Grande do Piauí"},  
5645: {id : '682', geocod : '2209104', nome : "Santa Cruz do Piauí"},  
5646: {id : '641', geocod : '2209153', nome : "Santa Cruz dos Milagres"},  
5647: {id : '1492', geocod : '2209203', nome : "Santa Filomena"},  
5648: {id : '1446', geocod : '2209302', nome : "Santa Luz"},  
5649: {id : '584', geocod : '2209377', nome : "Santa Rosa do Piauí"},  
5650: {id : '687', geocod : '2209351', nome : "Santana do Piauí"},  
5651: {id : '698', geocod : '2209401', nome : "Santo Antônio de Lisboa"},  
5652: {id : '605', geocod : '2209450', nome : "Santo Antônio dos Milagres"},  
5653: {id : '668', geocod : '2209500', nome : "Santo Inácio do Piauí"},  
5654: {id : '1509', geocod : '2209559', nome : "São Braz do Piauí"},  
5655: {id : '614', geocod : '2209609', nome : "São Félix do Piauí"},  
5656: {id : '1845', geocod : '2209658', nome : "São Francisco de Assis do Pi"},  
5657: {id : '570', geocod : '2209708', nome : "São Francisco do Piauí"},  
5658: {id : '1488', geocod : '2209757', nome : "São Gonçalo do Gurguéia"},  
5659: {id : '606', geocod : '2209807', nome : "São Gonçalo do Piauí"},  
5660: {id : '704', geocod : '2209856', nome : "São João da Canabrava"},  
5661: {id : '300', geocod : '2209872', nome : "São João da Fronteira"},  
5662: {id : '725', geocod : '2209906', nome : "São João da Serra"},  
5663: {id : '684', geocod : '2209955', nome : "São João da Varjota"},  
5664: {id : '135', geocod : '2209971', nome : "São João do Arraial"},  
5665: {id : '1526', geocod : '2210003', nome : "São João do Piauí"},  
5666: {id : '293', geocod : '2210052', nome : "São José do Divino"},  
5667: {id : '567', geocod : '2210102', nome : "São José do Peixe"},  
5668: {id : '689', geocod : '2210201', nome : "São José do Piauí"},  
5669: {id : '711', geocod : '2210300', nome : "São Julião"},  
5670: {id : '1516', geocod : '2210359', nome : "São Lourenço do Piauí"},  
5671: {id : '697', geocod : '2210375', nome : "São Luis do Piauí"},  
5672: {id : '615', geocod : '2210383', nome : "São Miguel da Baixa Grande"},  
5673: {id : '569', geocod : '2210391', nome : "São Miguel do Fidalgo"},  
5674: {id : '730', geocod : '2210409', nome : "São Miguel do Tapuio"},  
5675: {id : '607', geocod : '2210508', nome : "São Pedro do Piauí"},  
5676: {id : '1515', geocod : '2210607', nome : "São Raimundo Nonato"},  
5677: {id : '1484', geocod : '2210623', nome : "Sebastião Barros"},  
5678: {id : '549', geocod : '2210631', nome : "Sebastião Leal"},  
5679: {id : '734', geocod : '2210656', nome : "Sigefredo Pacheco"},  
5680: {id : '705', geocod : '2210706', nome : "Simões"},  
5681: {id : '666', geocod : '2210805', nome : "Simplício Mendes"},  
5682: {id : '563', geocod : '2210904', nome : "Socorro do Piauí"},  
5683: {id : '695', geocod : '2210938', nome : "Sussuapara"},  
5684: {id : '1517', geocod : '2210953', nome : "Tamboril do Piauí"},  
5685: {id : '585', geocod : '2210979', nome : "Tanque do Piauí"},  
5686: {id : '5486', geocod : '2211001', nome : "Teresina"},  
5687: {id : '621', geocod : '2211100', nome : "União"},  
5688: {id : '533', geocod : '2211209', nome : "Uruçuí"},  
5689: {id : '692', geocod : '2211308', nome : "Valença do Piauí"},  
5690: {id : '1513', geocod : '2211357', nome : "Várzea Branca"},  
5691: {id : '586', geocod : '2211407', nome : "Várzea Grande"},  
5692: {id : '671', geocod : '2211506', nome : "Vera Mendes"},  
5693: {id : '700', geocod : '2211605', nome : "Vila Nova do Piauí"},  
5694: {id : '681', geocod : '2211704', nome : "Wall Ferraz"}  
5695: ],  
5696: PR : [  
5697: // UF = PR  
5698: {id : '5568', geocod : '4104907', nome : "Abapã"},  
5699: {id : '3451', geocod : '4100103', nome : "Abatiá"},  
5700: {id : '5824', geocod : '4122206', nome : "Açungui"},  
5701: {id : '5579', geocod : '4127304', nome : "Adhemar de Barros"},  
5702: {id : '5047', geocod : '4100202', nome : "Adrianópolis"},  
5703: {id : '9462', geocod : '4113205', nome : "Água Azul"},  
5704: {id : '9463', geocod : '4117503', nome : "Água Boa"},  
5705: {id : '9479', geocod : '4110805', nome : "Águas de Jurema"},  
5706: {id : '4975', geocod : '4100301', nome : "Agudos do Sul"},  
5707: {id : '5611', geocod : '4107009', nome : "Alecrim"},  
5708: {id : '5616', geocod : '4126603', nome : "Alemoa"},  
5709: {id : '5617', geocod : '4118204', nome : "Alexandra"},  
5710: {id : '4984', geocod : '4100400', nome : "Almirante Tamandaré"},  
5711: {id : '4846', geocod : '4100459', nome : "Altamira do Paraná"},  
5712: {id : '5629', geocod : '4101002', nome : "Alto Alegre"},  
5713: {id : '5631', geocod : '4104600', nome : "Alto Alegre do Iguaçu"},  
5714: {id : '5657', geocod : '4127502', nome : "Alto do Amparo"},  
5715: {id : '5658', geocod : '4122503', nome : "Alto do São João"},  
5716: {id : '3215', geocod : '4100608', nome : "Alto Paraná"},  
5717: {id : '4856', geocod : '4100707', nome : "Alto Piquiri"},  
5718: {id : '5647', geocod : '4111506', nome : "Alto Porã"},  
5719: {id : '5649', geocod : '4117222', nome : "Alto Santa Fé"},  
5720: {id : '3148', geocod : '4100509', nome : "Altônia"},  
5721: {id : '5663', geocod : '4108304', nome : "Alvorada do Iguaçu"},  
5722: {id : '3321', geocod : '4100806', nome : "Alvorada do Sul"},  
5723: {id : '3173', geocod : '4100905', nome : "Amaporã"},  
5724: {id : '5682', geocod : '4109708', nome : "Amorinha"},  
5725: {id : '4678', geocod : '4101002', nome : "Ampére"},  
5726: {id : '4835', geocod : '4101051', nome : "Anahy"},  
5727: {id : '3332', geocod : '4101101', nome : "Andirá"},  
5728: {id : '3233', geocod : '4101150', nome : "Ângulo"},  
5729: {id : '5034', geocod : '4101200', nome : "Antonina"},  
5730: {id : '5706', geocod : '4112702', nome : "Antônio Brandão de Oiveira"},  
5731: {id : '4966', geocod : '4101309', nome : "Antônio Olinto"},  
5732: {id : '5714', geocod : '4123808', nome : "Anunciação"},  
5733: {id : '5717', geocod : '4127908', nome : "Aparecida dOeste"},  
5734: {id : '5726', geocod : '4110102', nome : "Apiaba"},  
5735: {id : '3251', geocod : '4101408', nome : "Apucarana"},  
5736: {id : '3252', geocod : '4101507', nome : "Arapongas"},  
5737: {id : '5042', geocod : '4101606', nome : "Arapoti"},  
5738: {id : '4873', geocod : '4101655', nome : "Arapuã"},  
5739: {id : '5746', geocod : '4112207', nome : "Arapuan - Janiópolis"},  
5740: {id : '5750', geocod : '4109500', nome : "Ararapira"},  
5741: {id : '3164', geocod : '4101705', nome : "Araruna"},  
5742: {id : '4981', geocod : '4101804', nome : "Araucária"},  
5743: {id : '5767', geocod : '4114302', nome : "Areia Branca dos Assis"},  
5744: {id : '4876', geocod : '4101853', nome : "Ariranha do Ivaí"},  
5745: {id : '3271', geocod : '4101903', nome : "Assaí"},  
5746: {id : '4826', geocod : '4102000', nome : "Assis Chateaubriand"},  
5747: {id : '3248', geocod : '4102109', nome : "Astorga"},  
5748: {id : '3232', geocod : '4102208', nome : "Atalaia"},  
5749: {id : '5813', geocod : '4125704', nome : "Aurora do Iguaçu"},  
5750: {id : '5826', geocod : '4106902', nome : "Bacacheri"},  
5751: {id : '4978', geocod : '4102307', nome : "Balsa Nova"},  
5752: {id : '5856', geocod : '4117057', nome : "Bananas"},  
5753: {id : '3282', geocod : '4102406', nome : "Bandeirantes"},  
5754: {id : '5939', geocod : '4116901', nome : "Barão de Lucena"},  
5755: {id : '4880', geocod : '4102505', nome : "Barbosa Ferraz"},  
5756: {id : '5875', geocod : '4119608', nome : "Barra Bonita"},  
5757: {id : '3453', geocod : '4102703', nome : "Barra do Jacaré"},  
5758: {id : '5881', geocod : '4111209', nome : "Barra Grande"},  
5759: {id : '5882', geocod : '4119806', nome : "Barra Grande - Planalto"},  
5760: {id : '5886', geocod : '4114500', nome : "Barra Santa Salete"},  
5761: {id : '4605', geocod : '4102604', nome : "Barracão"},  
5762: {id : '5914', geocod : '4117305', nome : "Barreiro"},  
5763: {id : '5948', geocod : '4104204', nome : "Bateias"},  
5764: {id : '5953', geocod : '4121604', nome : "Baulândia"},  
5765: {id : '5968', geocod : '4109302', nome : "Bela Vista"},  
5766: {id : '4615', geocod : '4102752', nome : "Bela Vista da Caroba"},  
5767: {id : '5978', geocod : '4107702', nome : "Bela Vista do Ivaí"},  
5768: {id : '3319', geocod : '4102802', nome : "Bela Vista do Paraíso"},  
5769: {id : '5979', geocod : '4103909', nome : "Bela Vista do Piquirí"},  
5770: {id : '5981', geocod : '4126801', nome : "Bela Vista do Tapira"},  
5771: {id : '5996', geocod : '4109203', nome : "Bentópolis"},  
5772: {id : '4773', geocod : '4102901', nome : "Bituruna"},  
5773: {id : '4853', geocod : '4103008', nome : "Boa Esperança"},  
5774: {id : '4688', geocod : '4103024', nome : "Boa Esperança do Iguaçu"},  
5775: {id : '4869', geocod : '4103040', nome : "Boa Ventura de São Roque"},  
5776: {id : '6023', geocod : '4120408', nome : "Boa Vista"},  
5777: {id : '4687', geocod : '4103057', nome : "Boa Vista da Aparecida"},  
5778: {id : '5029', geocod : '4103107', nome : "Bocaiúva do Sul"},  
5779: {id : '6050', geocod : '4111407', nome : "Bom Jardim do Sul"},  
5780: {id : '4613', geocod : '4103156', nome : "Bom Jesus do Sul"},  
5781: {id : '6064', geocod : '4119301', nome : "Bom Retiro"},  
5782: {id : '3238', geocod : '4103206', nome : "Bom Sucesso"},  
5783: {id : '4692', geocod : '4103222', nome : "Bom Sucesso do Sul"},  
5784: {id : '6088', geocod : '4120804', nome : "Borda do Campo"},  
5785: {id : '6089', geocod : '4125506', nome : "Borda do Campo de São José"},  
5786: {id : '6091', geocod : '4109302', nome : "Borman"},  
5787: {id : '3240', geocod : '4103305', nome : "Borrazópolis"},  
5788: {id : '6100', geocod : '4102505', nome : "Bourbonia"},  
5789: {id : '4834', geocod : '4103354', nome : "Braganey"},  
5790: {id : '6101', geocod : '4102000', nome : "Bragantina"},  
5791: {id : '4827', geocod : '4103370', nome : "Brasilândia do Sul"},  
5792: {id : '6130', geocod : '4102307', nome : "Bugre"},  
5793: {id : '6153', geocod : '4101200', nome : "Cacatu"},  
5794: {id : '6160', geocod : '4101200', nome : "Cachoeira de Cima"},  
5795: {id : '6164', geocod : '4125506', nome : "Cachoeira de São José"},  
5796: {id : '6168', geocod : '4121802', nome : "Cachoeira do Espírito Santo"},  
5797: {id : '6180', geocod : '4127502', nome : "Caetano Mendes"},  
5798: {id : '3301', geocod : '4103404', nome : "Cafeara"},  
5799: {id : '6182', geocod : '4106605', nome : "Cafeeiros"},  
5800: {id : '4839', geocod : '4103453', nome : "Cafelândia"},  
5801: {id : '3151', geocod : '4103479', nome : "Cafezal do Sul"},  
5802: {id : '6193', geocod : '4125605', nome : "Caitá"},  
5803: {id : '3258', geocod : '4103503', nome : "Califórnia"},  
5804: {id : '6215', geocod : '4101606', nome : "Calógeras"},  
5805: {id : '3476', geocod : '4103602', nome : "Cambará"},  
5806: {id : '3261', geocod : '4103701', nome : "Cambé"},  
5807: {id : '3250', geocod : '4103800', nome : "Cambira"},  
5808: {id : '4851', geocod : '4103909', nome : "Campina da Lagoa"},  
5809: {id : '4867', geocod : '4103958', nome : "Campina do Simão"},  
5810: {id : '5028', geocod : '4104006', nome : "Campina Grande do Sul"},  
5811: {id : '4833', geocod : '4104055', nome : "Campo Bonito"},  
5812: {id : '6248', geocod : '4106902', nome : "Campo Comprido"},  
5813: {id : '4972', geocod : '4104105', nome : "Campo do Tenente"},  
5814: {id : '4980', geocod : '4104204', nome : "Campo Largo"},  
5815: {id : '6251', geocod : '4125506', nome : "Campo Largo da Roseira"},  
5816: {id : '4983', geocod : '4104253', nome : "Campo Magro"},  
5817: {id : '4878', geocod : '4104303', nome : "Campo Mourão"},  
5818: {id : '6280', geocod : '4127700', nome : "Candeia"},  
5819: {id : '4885', geocod : '4104402', nome : "Cândido de Abreu"},  
5820: {id : '4769', geocod : '4104428', nome : "Candói"},  
5821: {id : '6282', geocod : '4121604', nome : "Canela"},  
5822: {id : '4770', geocod : '4104451', nome : "Cantagalo"},  
5823: {id : '6299', geocod : '4120358', nome : "Canzianópolis"},  
5824: {id : '4618', geocod : '4104501', nome : "Capanema"},  
5825: {id : '4620', geocod : '4104600', nome : "Capitão Leônidas Marques"},  
5826: {id : '6333', geocod : '4112751', nome : "Carajá"},  
5827: {id : '5038', geocod : '4104659', nome : "Carambeí"},  
5828: {id : '6341', geocod : '4101606', nome : "Caratuva"},  
5829: {id : '6347', geocod : '4114708', nome : "Carbonera"},  
5830: {id : '3440', geocod : '4104709', nome : "Carlópolis"},  
5831: {id : '4830', geocod : '4104808', nome : "Cascavel"},  
5832: {id : '5039', geocod : '4104907', nome : "Castro"},  
5833: {id : '4691', geocod : '4105003', nome : "Catanduvas"},  
5834: {id : '6388', geocod : '4106209', nome : "Catanduvas do Sul"},  
5835: {id : '3303', geocod : '4105102', nome : "Centenário do Sul"},  
5836: {id : '6423', geocod : '4119806', nome : "Centro Novo"},  
5837: {id : '5046', geocod : '4105201', nome : "Cerro Azul"},  
5838: {id : '4816', geocod : '4105300', nome : "Céu Azul"},  
5839: {id : '4700', geocod : '4105409', nome : "Chopinzinho"},  
5840: {id : '3165', geocod : '4105508', nome : "Cianorte"},  
5841: {id : '3168', geocod : '4105607', nome : "Cidade Gaúcha"},  
5842: {id : '6454', geocod : '4117107', nome : "Cintra Pimentel"},  
5843: {id : '4747', geocod : '4105706', nome : "Clevelândia"},  
5844: {id : '5026', geocod : '4105805', nome : "Colombo"},  
5845: {id : '6485', geocod : '4125506', nome : "Colônia Murici"},  
5846: {id : '6487', geocod : '4100301', nome : "Colônia Padre Paulo"},  
5847: {id : '3289', geocod : '4105904', nome : "Colorado"},  
5848: {id : '6494', geocod : '4119707', nome : "Comur"},  
5849: {id : '6512', geocod : '4119004', nome : "Conciolândia"},  
5850: {id : '6518', geocod : '4106407', nome : "Congonhas"},  
5851: {id : '3277', geocod : '4106001', nome : "Congonhinhas"},  
5852: {id : '3446', geocod : '4106100', nome : "Conselheiro Mairinck"},  
5853: {id : '6522', geocod : '4124103', nome : "Conselheiro Zacarias"},  
5854: {id : '4979', geocod : '4106209', nome : "Contenda"},  
5855: {id : '6528', geocod : '4125308', nome : "Copacabana do Norte"},  
5856: {id : '4831', geocod : '4106308', nome : "Corbélia"},  
5857: {id : '3276', geocod : '4106407', nome : "Cornélio Procópio"},  
5858: {id : '4749', geocod : '4106456', nome : "Coronel Domingos Soares"},  
5859: {id : '6539', geocod : '4105706', nome : "Coronel Firmino Martins"},  
5860: {id : '4698', geocod : '4106506', nome : "Coronel Vivida"},  
5861: {id : '6545', geocod : '4101408', nome : "Correia de Freitas"},  
5862: {id : '4879', geocod : '4106555', nome : "Corumbataí do Sul"},  
5863: {id : '6561', geocod : '4114401', nome : "Covó"},  
5864: {id : '6563', geocod : '4111209', nome : "Coxilha Rica"},  
5865: {id : '6580', geocod : '4118402', nome : "Cristo Rei"},  
5866: {id : '6579', geocod : '4104501', nome : "Cristo Rei - Capanema"},  
5867: {id : '4799', geocod : '4106803', nome : "Cruz Machado"},  
5868: {id : '4689', geocod : '4106571', nome : "Cruzeiro do Iguaçu"},  
5869: {id : '6596', geocod : '4128401', nome : "Cruzeiro do Norte"},  
5870: {id : '3160', geocod : '4106605', nome : "Cruzeiro do Oeste"},  
5871: {id : '3286', geocod : '4106704', nome : "Cruzeiro do Sul"},  
5872: {id : '4883', geocod : '4106852', nome : "Cruzmaltina"},  
5873: {id : '5487', geocod : '4106902', nome : "Curitiba"},  
5874: {id : '4895', geocod : '4107009', nome : "Curiúva"},  
5875: {id : '6652', geocod : '4115903', nome : "Deputado José Afonso"},  
5876: {id : '6660', geocod : '4127700', nome : "Dez de Maio"},  
5877: {id : '3198', geocod : '4107108', nome : "Diamante do Norte"},  
5878: {id : '4847', geocod : '4107124', nome : "Diamante do Sul"},  
5879: {id : '4487', geocod : '4107157', nome : "Diamante D'Oeste"},  
5880: {id : '6671', geocod : '4121208', nome : "Doce Grande"},  
5881: {id : '6673', geocod : '4127700', nome : "Dois Irmãos"},  
5882: {id : '6672', geocod : '4124806', nome : "Dois Irmãos - São João"},  
5883: {id : '4685', geocod : '4107207', nome : "Dois Vizinhos"},  
5884: {id : '6691', geocod : '4113908', nome : "Dorizon"},  
5885: {id : '3162', geocod : '4107256', nome : "Douradina"},  
5886: {id : '6697', geocod : '4125209', nome : "Doutor Antônio Paran"},  
5887: {id : '3211', geocod : '4107306', nome : "Doutor Camargo"},  
5888: {id : '6704', geocod : '4108809', nome : "Doutor Oliveira Castro"},  
5889: {id : '5044', geocod : '4128633', nome : "Doutor Ulysses"},  
5890: {id : '6711', geocod : '4112009', nome : "Eduardo Xavier da Silva"},  
5891: {id : '6719', geocod : '4102000', nome : "Encantado dOeste"},  
5892: {id : '4684', geocod : '4107405', nome : "Enéas Marques"},  
5893: {id : '3205', geocod : '4107504', nome : "Engenheiro Beltrão"},  
5894: {id : '6740', geocod : '4109401', nome : "Entre Rios"},  
5895: {id : '4488', geocod : '4107538', nome : "Entre Rios do Oeste"},  
5896: {id : '6754', geocod : '4100806', nome : "Esperança do Norte"},  
5897: {id : '3150', geocod : '4107520', nome : "Esperança Nova"},  
5898: {id : '4702', geocod : '4107546', nome : "Espigão Alto do Iguaçu"},  
5899: {id : '6784', geocod : '4109708', nome : "Euzébio de Oliveira"},  
5900: {id : '4547', geocod : '4107553', nome : "Farol"},  
5901: {id : '4894', geocod : '4107603', nome : "Faxinal"},  
5902: {id : '6796', geocod : '4119301', nome : "Faxinal do Céu"},  
5903: {id : '4982', geocod : '4107652', nome : "Fazenda Rio Grande"},  
5904: {id : '3207', geocod : '4107702', nome : "Fênix"},  
5905: {id : '4813', geocod : '4107736', nome : "Fernandes Pinheiro"},  
5906: {id : '6817', geocod : '4116307', nome : "Fernão Dias"},  
5907: {id : '6820', geocod : '4104204', nome : "Ferraria"},  
5908: {id : '3267', geocod : '4107751', nome : "Figueira"},  
5909: {id : '4539', geocod : '4107850', nome : "Flor da Serra do Sul"},  
5910: {id : '3214', geocod : '4107801', nome : "Floraí"},  
5911: {id : '3209', geocod : '4107900', nome : "Floresta"},  
5912: {id : '3306', geocod : '4108007', nome : "Florestópolis"},  
5913: {id : '6841', geocod : '4115200', nome : "Floriano"},  
5914: {id : '3234', geocod : '4108106', nome : "Flórida"},  
5915: {id : '6846', geocod : '4125605', nome : "Fluviópolis"},  
5916: {id : '4841', geocod : '4108205', nome : "Formosa do Oeste"},  
5917: {id : '4463', geocod : '4108304', nome : "Foz do Iguaçu"},  
5918: {id : '4768', geocod : '4108452', nome : "Foz do Jordão"},  
5919: {id : '4828', geocod : '4108320', nome : "Francisco Alves"},  
5920: {id : '4683', geocod : '4108403', nome : "Francisco Beltrão"},  
5921: {id : '6857', geocod : '4117602', nome : "Francisco Frederico"},  
5922: {id : '6865', geocod : '4112702', nome : "Frei Timóteo"},  
5923: {id : '4787', geocod : '4108502', nome : "General Carneiro"},  
5924: {id : '6905', geocod : '4116802', nome : "Geremia Lunardelli"},  
5925: {id : '4866', geocod : '4108551', nome : "Godoy Moreira"},  
5926: {id : '4844', geocod : '4108601', nome : "Goioerê"},  
5927: {id : '4771', geocod : '4108650', nome : "Goioxim"},  
5928: {id : '6923', geocod : '4110706', nome : "Gonçalves Júnior"},  
5929: {id : '6929', geocod : '4118402', nome : "Graciosa"},  
5930: {id : '4877', geocod : '4108700', nome : "Grandes Rios"},  
5931: {id : '6947', geocod : '4103479', nome : "Guaiporã"},  
5932: {id : '4494', geocod : '4108809', nome : "Guaíra"},  
5933: {id : '3199', geocod : '4108908', nome : "Guairaçá"},  
5934: {id : '6948', geocod : '4109401', nome : "Guairacá - Guarapuava"},  
5935: {id : '6952', geocod : '4101804', nome : "Guajuvira"},  
5936: {id : '4811', geocod : '4108957', nome : "Guamiranga"},  
5937: {id : '6953', geocod : '4110706', nome : "Guamirim"},  
5938: {id : '3454', geocod : '4109005', nome : "Guapirama"},  
5939: {id : '6960', geocod : '4109302', nome : "Guaporé"},  
5940: {id : '3171', geocod : '4109104', nome : "Guaporema"},  
5941: {id : '6980', geocod : '4109401', nome : "Guará"},  
5942: {id : '3298', geocod : '4109203', nome : "Guaraci"},  
5943: {id : '6962', geocod : '4119905', nome : "Guaragi"},  
5944: {id : '6973', geocod : '4117057', nome : "Guaraí"},  
5945: {id : '6963', geocod : '4117057', nome : "Guarani"},  
5946: {id : '4845', geocod : '4109302', nome : "Guaraniaçu"},  
5947: {id : '4774', geocod : '4109401', nome : "Guarapuava"},  
5948: {id : '5521', geocod : '4109500', nome : "Guaraqueçaba"},  
5949: {id : '5032', geocod : '4109609', nome : "Guaratuba"},  
5950: {id : '6972', geocod : '4127007', nome : "Guaraúna"},  
5951: {id : '6971', geocod : '4113700', nome : "Guaravera"},  
5952: {id : '7005', geocod : '4111555', nome : "Herculândia"},  
5953: {id : '7008', geocod : '4103909', nome : "Herveira"},  
5954: {id : '4762', geocod : '4109658', nome : "Honório Serpa"},  
5955: {id : '3443', geocod : '4109708', nome : "Ibaiti"},  
5956: {id : '4832', geocod : '4109757', nome : "Ibema"},  
5957: {id : '7022', geocod : '4120507', nome : "Ibiaci"},  
5958: {id : '3263', geocod : '4109807', nome : "Ibiporã"},  
5959: {id : '7244', geocod : '4102109', nome : "Içara"},  
5960: {id : '3155', geocod : '4109906', nome : "Icaraíma"},  
5961: {id : '7061', geocod : '4121000', nome : "Icatu"},  
5962: {id : '3247', geocod : '4110003', nome : "Iguaraçu"},  
5963: {id : '7081', geocod : '4115200', nome : "Iguatemi"},  
5964: {id : '4836', geocod : '4110052', nome : "Iguatu"},  
5965: {id : '7083', geocod : '4114609', nome : "Iguiporã"},  
5966: {id : '4891', geocod : '4110078', nome : "Imbaú"},  
5967: {id : '4812', geocod : '4110102', nome : "Imbituva"},  
5968: {id : '4802', geocod : '4110201', nome : "Inácio Martins"},  
5969: {id : '3285', geocod : '4110300', nome : "Inajá"},  
5970: {id : '3169', geocod : '4110409', nome : "Indianópolis"},  
5971: {id : '7118', geocod : '4125209', nome : "Iolópolis"},  
5972: {id : '4890', geocod : '4110508', nome : "Ipiranga"},  
5973: {id : '4829', geocod : '4110607', nome : "Iporã"},  
5974: {id : '4838', geocod : '4110656', nome : "Iracema do Oeste"},  
5975: {id : '7140', geocod : '4121109', nome : "Irapuan"},  
5976: {id : '4810', geocod : '4110706', nome : "Irati"},  
5977: {id : '7147', geocod : '4113700', nome : "Irerê"},  
5978: {id : '4864', geocod : '4110805', nome : "Iretama"},  
5979: {id : '3292', geocod : '4110904', nome : "Itaguajé"},  
5980: {id : '7165', geocod : '4119905', nome : "Itaiacoca"},  
5981: {id : '4483', geocod : '4110953', nome : "Itaipulândia"},  
5982: {id : '3346', geocod : '4111001', nome : "Itambaracá"},  
5983: {id : '3210', geocod : '4111100', nome : "Itambé"},  
5984: {id : '7201', geocod : '4110706', nome : "Itapará"},  
5985: {id : '4693', geocod : '4111209', nome : "Itapejara d'Oeste"},  
5986: {id : '4985', geocod : '4111258', nome : "Itaperuçu"},  
5987: {id : '3197', geocod : '4111308', nome : "Itaúna do Sul"},  
5988: {id : '4884', geocod : '4111407', nome : "Ivaí"},  
5989: {id : '4874', geocod : '4111506', nome : "Ivaiporã"},  
5990: {id : '7241', geocod : '4116901', nome : "Ivaitinga"},  
5991: {id : '3161', geocod : '4111555', nome : "Ivaté"},  
5992: {id : '3206', geocod : '4111605', nome : "Ivatuba"},  
5993: {id : '3445', geocod : '4111704', nome : "Jaboti"},  
5994: {id : '3457', geocod : '4111803', nome : "Jacarezinho"},  
5995: {id : '7258', geocod : '4120606', nome : "Jaciaba"},  
5996: {id : '7270', geocod : '4121901', nome : "Jacutinga"},  
5997: {id : '7269', geocod : '4111506', nome : "Jacutinga - Ivaiporã"},  
5998: {id : '3255', geocod : '4111902', nome : "Jaguapitã"},  
5999: {id : '5043', geocod : '4112009', nome : "Jaguariaíva"},  
6000: {id : '3239', geocod : '4112108', nome : "Jandaia do Sul"},  
6001: {id : '7286', geocod : '4103479', nome : "Jangada"},  
6002: {id : '4854', geocod : '4112207', nome : "Janiópolis"},  
6003: {id : '3444', geocod : '4112306', nome : "Japira"},  
6004: {id : '3176', geocod : '4112405', nome : "Japurá"},  
6005: {id : '4875', geocod : '4112504', nome : "Jardim Alegre"},  
6006: {id : '3291', geocod : '4112603', nome : "Jardim Olinda"},  
6007: {id : '7298', geocod : '4100509', nome : "Jardim Paredão"},  
6008: {id : '3264', geocod : '4112702', nome : "Jataizinho"},  
6009: {id : '4837', geocod : '4112751', nome : "Jesuítas"},  
6010: {id : '7334', geocod : '4112801', nome : "Joá"},  
6011: {id : '3456', geocod : '4112801', nome : "Joaquim Távora"},  
6012: {id : '7329', geocod : '4121703', nome : "José Lacerda"},  
6013: {id : '3450', geocod : '4112900', nome : "Jundiaí do Sul"},  
6014: {id : '4850', geocod : '4112959', nome : "Juranda"},  
6015: {id : '3167', geocod : '4113007', nome : "Jussara"},  
6016: {id : '7376', geocod : '4104808', nome : "Juvinópolis"},  
6017: {id : '3237', geocod : '4113106', nome : "Kaloré"},  
6018: {id : '7423', geocod : '4125605', nome : "Lajeado"},  
6019: {id : '7424', geocod : '4117305', nome : "Lajeado Bonito"},  
6020: {id : '7434', geocod : '4126207', nome : "Lambari"},  
6021: {id : '4973', geocod : '4113205', nome : "Lapa"},  
6022: {id : '4849', geocod : '4113254', nome : "Laranjal"},  
6023: {id : '4766', geocod : '4113304', nome : "Laranjeiras do Sul"},  
6024: {id : '7448', geocod : '4119202', nome : "Lavrinha"},  
6025: {id : '3274', geocod : '4113403', nome : "Leópolis"},  
6026: {id : '7450', geocod : '4113700', nome : "Lerroville"},  
6027: {id : '4882', geocod : '4113429', nome : "Lidianópolis"},  
6028: {id : '4622', geocod : '4113452', nome : "Lindoeste"},  
6029: {id : '3189', geocod : '4113502', nome : "Loanda"},  
6030: {id : '3288', geocod : '4113601', nome : "Lobato"},  
6031: {id : '3262', geocod : '4113700', nome : "Londrina"},  
6032: {id : '7474', geocod : '4128104', nome : "Lovat"},  
6033: {id : '7477', geocod : '4125001', nome : "Luar"},  
6034: {id : '4863', geocod : '4113734', nome : "Luiziana"},  
6035: {id : '4881', geocod : '4113759', nome : "Lunardelli"},  
6036: {id : '3302', geocod : '4113809', nome : "Lupionópolis"},  
6037: {id : '7505', geocod : '4113809', nome : "Mairá"},  
6038: {id : '4805', geocod : '4113908', nome : "Mallet"},  
6039: {id : '4855', geocod : '4114005', nome : "Mamborê"},  
6040: {id : '3218', geocod : '4114104', nome : "Mandaguaçu"},  
6041: {id : '3246', geocod : '4114203', nome : "Mandaguari"},  
6042: {id : '4977', geocod : '4114302', nome : "Mandirituba"},  
6043: {id : '4677', geocod : '4114351', nome : "Manfrinópolis"},  
6044: {id : '4763', geocod : '4114401', nome : "Mangueirinha"},  
6045: {id : '4872', geocod : '4114500', nome : "Manoel Ribas"},  
6046: {id : '7538', geocod : '4127908', nome : "Marabá"},  
6047: {id : '7541', geocod : '4116703', nome : "Marajó"},  
6048: {id : '7551', geocod : '4123808', nome : "Maravilha"},  
6049: {id : '7550', geocod : '4113700', nome : "Maravilha - Londrina"},  
6050: {id : '7552', geocod : '4125506', nome : "Marcelino"},  
6051: {id : '7554', geocod : '4124400', nome : "Marcionópolis"},  
6052: {id : '4492', geocod : '4114609', nome : "Marechal Cândido Rondon"},  
6053: {id : '7562', geocod : '4114609', nome : "Margarida"},  
6054: {id : '3159', geocod : '4114708', nome : "Maria Helena"},  
6055: {id : '3244', geocod : '4114807', nome : "Marialva"},  
6056: {id : '7567', geocod : '4113205', nome : "Mariental"},  
6057: {id : '3257', geocod : '4114906', nome : "Marilândia do Sul"},  
6058: {id : '3190', geocod : '4115002', nome : "Marilena"},  
6059: {id : '7569', geocod : '4110805', nome : "Marilu"},  
6060: {id : '4857', geocod : '4115101', nome : "Mariluz"},  
6061: {id : '3219', geocod : '4115200', nome : "Maringá"},  
6062: {id : '4675', geocod : '4115309', nome : "Mariópolis"},  
6063: {id : '4823', geocod : '4115358', nome : "Maripá"},  
6064: {id : '7575', geocod : '4100608', nome : "Maristela"},  
6065: {id : '7576', geocod : '4125803', nome : "Mariza"},  
6066: {id : '7579', geocod : '4121406', nome : "Marmelândia"},  
6067: {id : '4681', geocod : '4115408', nome : "Marmeleiro"},  
6068: {id : '4859', geocod : '4115457', nome : "Marquinho"},  
6069: {id : '3241', geocod : '4115507', nome : "Marumbi"},  
6070: {id : '4619', geocod : '4115606', nome : "Matelândia"},  
6071: {id : '5033', geocod : '4115705', nome : "Matinhos"},  
6072: {id : '4861', geocod : '4115739', nome : "Mato Rico"},  
6073: {id : '3256', geocod : '4115754', nome : "Mauá da Serra"},  
6074: {id : '4481', geocod : '4115804', nome : "Medianeira"},  
6075: {id : '4491', geocod : '4115853', nome : "Mercedes"},  
6076: {id : '3172', geocod : '4115903', nome : "Mirador"},  
6077: {id : '7649', geocod : '4100707', nome : "Mirante do Piquiri"},  
6078: {id : '3304', geocod : '4116000', nome : "Miraselva"},  
6079: {id : '4485', geocod : '4116059', nome : "Missal"},  
6080: {id : '7669', geocod : '4117305', nome : "Monjolinho"},  
6081: {id : '7697', geocod : '4124103', nome : "Monte Real"},  
6082: {id : '4858', geocod : '4116109', nome : "Moreira Sales"},  
6083: {id : '5030', geocod : '4116208', nome : "Morretes"},  
6084: {id : '3249', geocod : '4116307', nome : "Munhoz de Melo"},  
6085: {id : '7766', geocod : '4117305', nome : "Natingui"},  
6086: {id : '7780', geocod : '4110607', nome : "Nilza"},  
6087: {id : '7784', geocod : '4100905', nome : "Nordestina"},  
6088: {id : '7785', geocod : '4101101', nome : "Nossa Senhora Aparecida - An"},  
6089: {id : '7787', geocod : '4122404', nome : "Nossa Senhora da Aparecida -"},  
6090: {id : '7788', geocod : '4102406', nome : "Nossa Senhora da Candelária"},  
6091: {id : '3297', geocod : '4116406', nome : "Nossa Senhora das Graças"},  
6092: {id : '3178', geocod : '4116505', nome : "Nova Aliança do Ivaí"},  
6093: {id : '7804', geocod : '4107603', nome : "Nova Altamira"},  
6094: {id : '3275', geocod : '4116604', nome : "Nova América da Colina"},  
6095: {id : '7806', geocod : '4114906', nome : "Nova Amoreira"},  
6096: {id : '4840', geocod : '4116703', nome : "Nova Aurora"},  
6097: {id : '7812', geocod : '4107801', nome : "Nova Bilac"},  
6098: {id : '7815', geocod : '4104709', nome : "Nova Brasília do Ita"},  
6099: {id : '4848', geocod : '4116802', nome : "Nova Cantu"},  
6100: {id : '7818', geocod : '4108403', nome : "Nova Concordia"},  
6101: {id : '3216', geocod : '4116901', nome : "Nova Esperança"},  
6102: {id : '4679', geocod : '4116950', nome : "Nova Esperança do Sudoeste"},  
6103: {id : '3278', geocod : '4117008', nome : "Nova Fátima"},  
6104: {id : '4704', geocod : '4117057', nome : "Nova Laranjeiras"},  
6105: {id : '3191', geocod : '4117107', nome : "Nova Londrina"},  
6106: {id : '7831', geocod : '4124806', nome : "Nova Lourdes"},  
6107: {id : '3163', geocod : '4117206', nome : "Nova Olímpia"},  
6108: {id : '4686', geocod : '4117255', nome : "Nova Prata do Iguaçu"},  
6109: {id : '3268', geocod : '4117214', nome : "Nova Santa Bárbara"},  
6110: {id : '4822', geocod : '4117222', nome : "Nova Santa Rosa"},  
6111: {id : '4865', geocod : '4117271', nome : "Nova Tebas"},  
6112: {id : '7856', geocod : '4114609', nome : "Novo Horizonte"},  
6113: {id : '3242', geocod : '4117297', nome : "Novo Itacolomi"},  
6114: {id : '7858', geocod : '4112306', nome : "Novo Jardim"},  
6115: {id : '7859', geocod : '4106902', nome : "Novo Mundo"},  
6116: {id : '7861', geocod : '4127700', nome : "Novo Sarandi"},  
6117: {id : '7862', geocod : '4127700', nome : "Novo Sobradinho"},  
6118: {id : '7863', geocod : '4114609', nome : "Novo Três Passos"},  
6119: {id : '7878', geocod : '4102000', nome : "Oliveira Barros"},  
6120: {id : '7888', geocod : '4110607', nome : "Oroite"},  
6121: {id : '4888', geocod : '4117305', nome : "Ortigueira"},  
6122: {id : '7895', geocod : '4102505', nome : "Ourilândia"},  
6123: {id : '3213', geocod : '4117404', nome : "Ourizona"},  
6124: {id : '7897', geocod : '4126306', nome : "Ouro Verde"},  
6125: {id : '4818', geocod : '4117453', nome : "Ouro Verde do Oeste"},  
6126: {id : '7899', geocod : '4106308', nome : "Ouro Verde do Piquiri"},  
6127: {id : '7921', geocod : '4106456', nome : "Padre Ponciano"},  
6128: {id : '3217', geocod : '4117503', nome : "Paiçandu"},  
6129: {id : '7929', geocod : '4113700', nome : "Paiquerê"},  
6130: {id : '4748', geocod : '4117602', nome : "Palmas"},  
6131: {id : '4970', geocod : '4117701', nome : "Palmeira"},  
6132: {id : '7949', geocod : '4109401', nome : "Palmeirinha"},  
6133: {id : '7951', geocod : '4125100', nome : "Palmira"},  
6134: {id : '4860', geocod : '4117800', nome : "Palmital"},  
6135: {id : '7953', geocod : '4116703', nome : "Palmitópolis"},  
6136: {id : '4824', geocod : '4117909', nome : "Palotina"},  
6137: {id : '7957', geocod : '4121208', nome : "Pangaré"},  
6138: {id : '7961', geocod : '4117701', nome : "Papagaios Novos"},  
6139: {id : '3177', geocod : '4118006', nome : "Paraíso do Norte"},  
6140: {id : '7973', geocod : '4116109', nome : "Paraná dOeste"},  
6141: {id : '3287', geocod : '4118105', nome : "Paranacity"},  
6142: {id : '7970', geocod : '4126405', nome : "Paranagi"},  
6143: {id : '5036', geocod : '4118204', nome : "Paranaguá"},  
6144: {id : '3290', geocod : '4118303', nome : "Paranapoema"},  
6145: {id : '3181', geocod : '4118402', nome : "Paranavaí"},  
6146: {id : '8007', geocod : '4113304', nome : "Passo Liso"},  
6147: {id : '4490', geocod : '4118451', nome : "Pato Bragado"},  
6148: {id : '4674', geocod : '4118501', nome : "Pato Branco"},  
6149: {id : '8027', geocod : '4120606', nome : "Patos Velhos"},  
6150: {id : '8035', geocod : '4101903', nome : "Pau dAlho do Sul"},  
6151: {id : '4801', geocod : '4118600', nome : "Paula Freitas"},  
6152: {id : '8038', geocod : '4100707', nome : "Paulistânia"},  
6153: {id : '4804', geocod : '4118709', nome : "Paulo Frontin"},  
6154: {id : '3204', geocod : '4118808', nome : "Peabiru"},  
6155: {id : '8046', geocod : '4109609', nome : "Pedra Branca do Arara"},  
6156: {id : '3157', geocod : '4118857', nome : "Perobal"},  
6157: {id : '3152', geocod : '4118907', nome : "Pérola"},  
6158: {id : '4610', geocod : '4119004', nome : "Pérola d'Oeste"},  
6159: {id : '8305', geocod : '4115358', nome : "Pérola Independente"},  
6160: {id : '4974', geocod : '4119103', nome : "Piên"},  
6161: {id : '8110', geocod : '4106803', nome : "Pinaré"},  
6162: {id : '5024', geocod : '4119152', nome : "Pinhais"},  
6163: {id : '4614', geocod : '4119251', nome : "Pinhal de São Bento"},  
6164: {id : '3447', geocod : '4119202', nome : "Pinhalão"},  
6165: {id : '8118', geocod : '4107405', nome : "Pinhalzinho - Enéas Marques"},  
6166: {id : '8119', geocod : '4119301', nome : "Pinhalzinho - Pinhão"},  
6167: {id : '4772', geocod : '4119301', nome : "Pinhão"},  
6168: {id : '8123', geocod : '4106902', nome : "Pinheirinho"},  
6169: {id : '8126', geocod : '4104501', nome : "Pinheiro"},  
6170: {id : '8143', geocod : '4104303', nome : "Piquirivaí"},  
6171: {id : '5041', geocod : '4119400', nome : "Piraí do Sul"},  
6172: {id : '8156', geocod : '4101408', nome : "Pirapó"},  
6173: {id : '5025', geocod : '4119509', nome : "Piraquara"},  
6174: {id : '8161', geocod : '4119905', nome : "Piriquitos"},  
6175: {id : '4870', geocod : '4119608', nome : "Pitanga"},  
6176: {id : '3253', geocod : '4119657', nome : "Pitangueiras"},  
6177: {id : '3196', geocod : '4119707', nome : "Planaltina do Paraná"},  
6178: {id : '4611', geocod : '4119806', nome : "Planalto"},  
6179: {id : '8184', geocod : '4102505', nome : "Pocinho"},  
6180: {id : '8186', geocod : '4117271', nome : "Poema"},  
6181: {id : '5037', geocod : '4119905', nome : "Ponta Grossa"},  
6182: {id : '5035', geocod : '4119954', nome : "Pontal do Paraná"},  
6183: {id : '3320', geocod : '4120002', nome : "Porecatu"},  
6184: {id : '8213', geocod : '4107702', nome : "Porteira Preta"},  
6185: {id : '4969', geocod : '4120101', nome : "Porto Amazonas"},  
6186: {id : '4765', geocod : '4120150', nome : "Porto Barreiro"},  
6187: {id : '8218', geocod : '4121000', nome : "Porto Brasílio"},  
6188: {id : '8219', geocod : '4109906', nome : "Porto Camargo"},  
6189: {id : '8237', geocod : '4116208', nome : "Porto de Cima"},  
6190: {id : '8221', geocod : '4114609', nome : "Porto Mendes"},  
6191: {id : '3187', geocod : '4120200', nome : "Porto Rico"},  
6192: {id : '8228', geocod : '4125555', nome : "Porto São Carlos"},  
6193: {id : '8229', geocod : '4125902', nome : "Porto São José"},  
6194: {id : '4800', geocod : '4120309', nome : "Porto Vitória"},  
6195: {id : '3305', geocod : '4120333', nome : "Prado Ferreira"},  
6196: {id : '4609', geocod : '4120358', nome : "Pranchita"},  
6197: {id : '8274', geocod : '4103701', nome : "Prata"},  
6198: {id : '3220', geocod : '4120408', nome : "Presidente Castelo Branco"},  
6199: {id : '8280', geocod : '4128609', nome : "Presidente Kennedy"},  
6200: {id : '3333', geocod : '4120507', nome : "Primeiro de Maio"},  
6201: {id : '4803', geocod : '4120606', nome : "Prudentópolis"},  
6202: {id : '4843', geocod : '4120655', nome : "Quarto Centenário"},  
6203: {id : '3455', geocod : '4120705', nome : "Quatiguá"},  
6204: {id : '5027', geocod : '4120804', nome : "Quatro Barras"},  
6205: {id : '4821', geocod : '4120853', nome : "Quatro Pontes"},  
6206: {id : '4701', geocod : '4120903', nome : "Quedas do Iguaçu"},  
6207: {id : '3156', geocod : '4121000', nome : "Querência do Norte"},  
6208: {id : '3208', geocod : '4121109', nome : "Quinta do Sol"},  
6209: {id : '4976', geocod : '4121208', nome : "Quitandinha"},  
6210: {id : '4486', geocod : '4121257', nome : "Ramilândia"},  
6211: {id : '3334', geocod : '4121307', nome : "Rancho Alegre"},  
6212: {id : '4852', geocod : '4121356', nome : "Rancho Alegre D'Oeste"},  
6213: {id : '4617', geocod : '4121406', nome : "Realeza"},  
6214: {id : '4809', geocod : '4121505', nome : "Rebouças"},  
6215: {id : '8349', geocod : '4126306', nome : "Reianópolis"},  
6216: {id : '4682', geocod : '4121604', nome : "Renascença"},  
6217: {id : '4889', geocod : '4121703', nome : "Reserva"},  
6218: {id : '4764', geocod : '4121752', nome : "Reserva do Iguaçu"},  
6219: {id : '8379', geocod : '4108700', nome : "Ribeirão Bonito"},  
6220: {id : '3441', geocod : '4121802', nome : "Ribeirão Claro"},  
6221: {id : '3279', geocod : '4121901', nome : "Ribeirão do Pinhal"},  
6222: {id : '4806', geocod : '4122008', nome : "Rio Azul"},  
6223: {id : '3243', geocod : '4122107', nome : "Rio Bom"},  
6224: {id : '8405', geocod : '4108320', nome : "Rio Bonito"},  
6225: {id : '4703', geocod : '4122156', nome : "Rio Bonito do Iguaçu"},  
6226: {id : '4886', geocod : '4122172', nome : "Rio Branco do Ivaí"},  
6227: {id : '4986', geocod : '4122206', nome : "Rio Branco do Sul"},  
6228: {id : '8409', geocod : '4113908', nome : "Rio Claro do Sul"},  
6229: {id : '8430', geocod : '4123808', nome : "Rio da Prata"},  
6230: {id : '8429', geocod : '4117057', nome : "Rio da Prata - Nova Laranjeiras"},  
6231: {id : '8435', geocod : '4104808', nome : "Rio do Salto"},  
6232: {id : '4971', geocod : '4122305', nome : "Rio Negro"},  
6233: {id : '8414', geocod : '4121703', nome : "Rio Novo"},  
6234: {id : '8416', geocod : '4115309', nome : "Rio Pinheiro"},  
6235: {id : '8423', geocod : '4112959', nome : "Rio Verde"},  
6236: {id : '8442', geocod : '4128104', nome : "Roberto Silveira"},  
6237: {id : '3260', geocod : '4122404', nome : "Rolândia"},  
6238: {id : '4862', geocod : '4122503', nome : "Roncador"},  
6239: {id : '3170', geocod : '4122602', nome : "Rondon"},  
6240: {id : '4887', geocod : '4122651', nome : "Rosário do Ivaí"},  
6241: {id : '3254', geocod : '4122701', nome : "Sabáudia"},  
6242: {id : '8471', geocod : '4119806', nome : "Sagrada Família"},  
6243: {id : '4676', geocod : '4122800', nome : "Salgado Filho"},  
6244: {id : '8477', geocod : '4103909', nome : "Salles de Oliveira"},  
6245: {id : '8481', geocod : '4100707', nome : "Saltinho do Oeste"},  
6246: {id : '3435', geocod : '4122909', nome : "Salto do Itararé"},  
6247: {id : '4680', geocod : '4123006', nome : "Salto do Lontra"},  
6248: {id : '3281', geocod : '4123105', nome : "Santa Amélia"},  
6249: {id : '3269', geocod : '4123204', nome : "Santa Cecília do Pavão"},  
6250: {id : '3185', geocod : '4123303', nome : "Santa Cruz de Monte Castelo"},  
6251: {id : '8517', geocod : '4128104', nome : "Santa Eliza"},  
6252: {id : '8520', geocod : '4123303', nome : "Santa Esmeralda"},  
6253: {id : '3296', geocod : '4123402', nome : "Santa Fé"},  
6254: {id : '8529', geocod : '4114807', nome : "Santa Fé do Pirapó"},  
6255: {id : '8522', geocod : '4106902', nome : "Santa Felicidade"},  
6256: {id : '4484', geocod : '4123501', nome : "Santa Helena"},  
6257: {id : '3300', geocod : '4123600', nome : "Santa Inês"},  
6258: {id : '3188', geocod : '4123709', nome : "Santa Isabel do Ivaí"},  
6259: {id : '4616', geocod : '4123808', nome : "Santa Izabel do Oeste"},  
6260: {id : '4621', geocod : '4123824', nome : "Santa Lúcia"},  
6261: {id : '8544', geocod : '4125001', nome : "Santa Luzia da Alvorada"},  
6262: {id : '8556', geocod : '4102802', nome : "Santa Margarida"},  
6263: {id : '8560', geocod : '4100608', nome : "Santa Maria"},  
6264: {id : '4868', geocod : '4123857', nome : "Santa Maria do Oeste"},  
6265: {id : '8565', geocod : '4106001', nome : "Santa Maria do Rio do Peixe"},  
6266: {id : '3280', geocod : '4123907', nome : "Santa Mariana"},  
6267: {id : '3144', geocod : '4123956', nome : "Santa Mônica"},  
6268: {id : '8567', geocod : '4106902', nome : "Santa Quitéria"},  
6269: {id : '8572', geocod : '4127403', nome : "Santa Rita dOeste"},  
6270: {id : '4820', geocod : '4124020', nome : "Santa Tereza do Oeste"},  
6271: {id : '4479', geocod : '4124053', nome : "Santa Terezinha de Itaipu"},  
6272: {id : '8600', geocod : '4102109', nome : "Santa Zélia"},  
6273: {id : '8602', geocod : '4106803', nome : "Santana"},  
6274: {id : '3434', geocod : '4124004', nome : "Santana do Itararé"},  
6275: {id : '3452', geocod : '4124103', nome : "Santo Antônio da Platina"},  
6276: {id : '3284', geocod : '4124202', nome : "Santo Antônio do Caiuá"},  
6277: {id : '8656', geocod : '4102901', nome : "Santo Antônio do Ira"},  
6278: {id : '8661', geocod : '4122107', nome : "Santo Antônio do Palmital"},  
6279: {id : '3272', geocod : '4124301', nome : "Santo Antônio do Paraíso"},  
6280: {id : '4612', geocod : '4124400', nome : "Santo Antônio do Sudoeste"},  
6281: {id : '3299', geocod : '4124509', nome : "Santo Inácio"},  
6282: {id : '8683', geocod : '4116802', nome : "Santo Rei"},  
6283: {id : '8828', geocod : '4117909', nome : "São Camilo"},  
6284: {id : '3179', geocod : '4124608', nome : "São Carlos do Ivaí"},  
6285: {id : '8830', geocod : '4123501', nome : "São Clemente"},  
6286: {id : '8831', geocod : '4128203', nome : "São Cristovão"},  
6287: {id : '8839', geocod : '4128203', nome : "São Domingos"},  
6288: {id : '8844', geocod : '4105409', nome : "São Francisco"},  
6289: {id : '8852', geocod : '4106001', nome : "São Francisco de Imbaú"},  
6290: {id : '8853', geocod : '4105706', nome : "São Francisco de Sales"},  
6291: {id : '3265', geocod : '4124707', nome : "São Jerônimo da Serra"},  
6292: {id : '4696', geocod : '4124806', nome : "São João"},  
6293: {id : '8940', geocod : '4100509', nome : "São João - Altônia"},  
6294: {id : '3283', geocod : '4124905', nome : "São João do Caiuá"},  
6295: {id : '3236', geocod : '4125001', nome : "São João do Ivaí"},  
6296: {id : '8967', geocod : '4124707', nome : "São João do Pinhal"},  
6297: {id : '4968', geocod : '4125100', nome : "São João do Triunfo"},  
6298: {id : '8947', geocod : '4104808', nome : "São João dOeste"},  
6299: {id : '8879', geocod : '4111001', nome : "São Joaquim do Pontal"},  
6300: {id : '3212', geocod : '4125308', nome : "São Jorge do Ivaí"},  
6301: {id : '3149', geocod : '4125357', nome : "São Jorge do Patrocínio"},  
6302: {id : '4695', geocod : '4125209', nome : "São Jorge d'Oeste"},  
6303: {id : '8887', geocod : '4112108', nome : "São José"},  
6304: {id : '3433', geocod : '4125407', nome : "São José da Boa Vista"},  
6305: {id : '4489', geocod : '4125456', nome : "São José das Palmeiras"},  
6306: {id : '8919', geocod : '4125704', nome : "São José do Iguaçu"},  
6307: {id : '8921', geocod : '4110953', nome : "São José do Itavó"},  
6308: {id : '8923', geocod : '4123709', nome : "São José do Ivaí"},  
6309: {id : '5023', geocod : '4125506', nome : "São José dos Pinhais"},  
6310: {id : '8972', geocod : '4124301', nome : "São Judas Tadeu"},  
6311: {id : '8974', geocod : '4105508', nome : "São Lourenço"},  
6312: {id : '8978', geocod : '4104501', nome : "São Luiz - Capanema"},  
6313: {id : '8979', geocod : '4105409', nome : "São Luiz - Chopinzinho"},  
6314: {id : '8980', geocod : '4113700', nome : "São Luiz - Londrina"},  
6315: {id : '8981', geocod : '4114807', nome : "São Luiz - Marialva"},  
6316: {id : '8982', geocod : '4115101', nome : "São Luiz - Mariluz"},  
6317: {id : '8984', geocod : '4102307', nome : "São Luiz do Purunã"},  
6318: {id : '3175', geocod : '4125555', nome : "São Manoel do Paraná"},  
6319: {id : '8990', geocod : '4125506', nome : "São Marcos"},  
6320: {id : '8993', geocod : '4122404', nome : "São Martinho"},  
6321: {id : '4808', geocod : '4125605', nome : "São Mateus do Sul"},  
6322: {id : '9001', geocod : '4127700', nome : "São Miguel"},  
6323: {id : '9008', geocod : '4114807', nome : "São Miguel do Cambuí"},  
6324: {id : '4480', geocod : '4125704', nome : "São Miguel do Iguaçu"},  
6325: {id : '9013', geocod : '4101002', nome : "São Pedro - Ampére"},  
6326: {id : '9014', geocod : '4101408', nome : "São Pedro Apucarana"},  
6327: {id : '9032', geocod : '4124400', nome : "São Pedro do Florido"},  
6328: {id : '4817', geocod : '4125753', nome : "São Pedro do Iguaçu"},  
6329: {id : '3235', geocod : '4125803', nome : "São Pedro do Ivaí"},  
6330: {id : '3186', geocod : '4125902', nome : "São Pedro do Paraná"},  
6331: {id : '9046', geocod : '4107207', nome : "São Roque - Dois Vizinhos"},  
6332: {id : '9047', geocod : '4114609', nome : "São Roque - M. Cândido Rondo"},  
6333: {id : '9048', geocod : '4121406', nome : "São Roque - Realeza"},  
6334: {id : '9052', geocod : '4118501', nome : "São Roque do Chopim"},  
6335: {id : '9054', geocod : '4101002', nome : "São Salvador"},  
6336: {id : '9056', geocod : '4105201', nome : "São Sebastião"},  
6337: {id : '3270', geocod : '4126009', nome : "São Sebastião da Amoreira"},  
6338: {id : '9090', geocod : '4104204', nome : "São Silvestre - Campo Largo"},  
6339: {id : '9091', geocod : '4106605', nome : "São Silvestre - Cruzeiro do"},  
6340: {id : '3174', geocod : '4126108', nome : "São Tomé"},  
6341: {id : '9098', geocod : '4107207', nome : "São Valentim"},  
6342: {id : '9102', geocod : '4101705', nome : "São Vicente"},  
6343: {id : '8697', geocod : '4127809', nome : "Sapé"},  
6344: {id : '3266', geocod : '4126207', nome : "Sapopema"},  
6345: {id : '3245', geocod : '4126256', nome : "Sarandi"},  
6346: {id : '8700', geocod : '4123808', nome : "Sarandi - Santa Izabel do Oe"},  
6347: {id : '4699', geocod : '4126272', nome : "Saudade do Iguaçu"},  
6348: {id : '8710', geocod : '4104808', nome : "Sede Alvorada"},  
6349: {id : '8713', geocod : '4125209', nome : "Sede Nova SantAna"},  
6350: {id : '8714', geocod : '4128609', nome : "Sede Progresso"},  
6351: {id : '4911', geocod : '4126306', nome : "Sengés"},  
6352: {id : '8719', geocod : '4115309', nome : "Senhor Bom Jesus dos Gramado"},  
6353: {id : '8743', geocod : '4128104', nome : "Serra dos Dourados"},  
6354: {id : '8733', geocod : '4109500', nome : "Serra Negra"},  
6355: {id : '4482', geocod : '4126355', nome : "Serranópolis do Iguaçu"},  
6356: {id : '3341', geocod : '4126405', nome : "Sertaneja"},  
6357: {id : '3323', geocod : '4126504', nome : "Sertanópolis"},  
6358: {id : '8756', geocod : '4107504', nome : "Sertãozinho"},  
6359: {id : '8773', geocod : '4102604', nome : "Siqueira Belo"},  
6360: {id : '3449', geocod : '4126603', nome : "Siqueira Campos"},  
6361: {id : '8781', geocod : '4104907', nome : "Socavão"},  
6362: {id : '4697', geocod : '4126652', nome : "Sulina"},  
6363: {id : '8806', geocod : '4118402', nome : "Sumaré"},  
6364: {id : '8812', geocod : '4107504', nome : "Sussuí"},  
6365: {id : '3259', geocod : '4126678', nome : "Tamarana"},  
6366: {id : '3180', geocod : '4126702', nome : "Tamboara"},  
6367: {id : '3147', geocod : '4126801', nome : "Tapejara"},  
6368: {id : '3145', geocod : '4126900', nome : "Tapira"},  
6369: {id : '9175', geocod : '4106902', nome : "Tatuquara"},  
6370: {id : '4814', geocod : '4127007', nome : "Teixeira Soares"},  
6371: {id : '4892', geocod : '4127106', nome : "Telêmaco Borba"},  
6372: {id : '9186', geocod : '4102505', nome : "Tereza Breda"},  
6373: {id : '9187', geocod : '4104402', nome : "Tereza Cristina"},  
6374: {id : '3166', geocod : '4127205', nome : "Terra Boa"},  
6375: {id : '9191', geocod : '4124707', nome : "Terra Nova"},  
6376: {id : '3200', geocod : '4127304', nome : "Terra Rica"},  
6377: {id : '4493', geocod : '4127403', nome : "Terra Roxa"},  
6378: {id : '4893', geocod : '4127502', nome : "Tibagi"},  
6379: {id : '5021', geocod : '4127601', nome : "Tijucas do Sul"},  
6380: {id : '9210', geocod : '4122800', nome : "Tiradentes"},  
6381: {id : '4819', geocod : '4127700', nome : "Toledo"},  
6382: {id : '3448', geocod : '4127809', nome : "Tomazina"},  
6383: {id : '4690', geocod : '4127858', nome : "Três Barras do Paraná"},  
6384: {id : '9246', geocod : '4104402', nome : "Três Bicos"},  
6385: {id : '9247', geocod : '4104204', nome : "Três Córregos"},  
6386: {id : '9241', geocod : '4107504', nome : "Triângulo"},  
6387: {id : '9237', geocod : '4121901', nome : "Triolândia"},  
6388: {id : '5045', geocod : '4127882', nome : "Tunas do Paraná"},  
6389: {id : '3146', geocod : '4127908', nome : "Tuneiras do Oeste"},  
6390: {id : '4825', geocod : '4127957', nome : "Tupãssi"},  
6391: {id : '9267', geocod : '4102109', nome : "Tupinambá"},  
6392: {id : '4871', geocod : '4127965', nome : "Turvo"},  
6393: {id : '9276', geocod : '4125001', nome : "Ubaúna"},  
6394: {id : '4842', geocod : '4128005', nome : "Ubiratã"},  
6395: {id : '9285', geocod : '4106902', nome : "Umbará"},  
6396: {id : '3158', geocod : '4128104', nome : "Umuarama"},  
6397: {id : '4790', geocod : '4128203', nome : "União da Vitória"},  
6398: {id : '9294', geocod : '4123808', nome : "União do Oeste"},  
6399: {id : '3221', geocod : '4128302', nome : "Uniflor"},  
6400: {id : '3273', geocod : '4128401', nome : "Uraí"},  
6401: {id : '9308', geocod : '4119905', nome : "Uvaia"},  
6402: {id : '9327', geocod : '4119806', nome : "Valério"},  
6403: {id : '9339', geocod : '4109708', nome : "Vassoural"},  
6404: {id : '5040', geocod : '4128534', nome : "Ventania"},  
6405: {id : '4815', geocod : '4128559', nome : "Vera Cruz do Oeste"},  
6406: {id : '9354', geocod : '4118709', nome : "Vera Guarani"},  
6407: {id : '4694', geocod : '4128609', nome : "Verê"},  
6408: {id : '9363', geocod : '4126207', nome : "Vida Nova"},  
6409: {id : '9365', geocod : '4105508', nome : "Vidigal"},  
6410: {id : '3154', geocod : '4128625', nome : "Vila Alta"},  
6411: {id : '9376', geocod : '4109708', nome : "Vila Guay"},  
6412: {id : '9382', geocod : '4119608', nome : "Vila Nova - Pitanga"},  
6413: {id : '9383', geocod : '4127700', nome : "Vila Nova - Toledo"},  
6414: {id : '9388', geocod : '4124707', nome : "Vila Nova de Florençia"},  
6415: {id : '9392', geocod : '4124806', nome : "Vila Paraíso"},  
6416: {id : '9396', geocod : '4101408', nome : "Vila Reis"},  
6417: {id : '9399', geocod : '4109906', nome : "Vila Rica do Ivaí"},  
6418: {id : '4767', geocod : '4128658', nome : "Virmond"},  
6419: {id : '9422', geocod : '4107405', nome : "Vista Alegre - Enéas Marques"},  
6420: {id : '9421', geocod : '4106506', nome : "Vista Alegre Coronel Vivida"},  
6421: {id : '4667', geocod : '4128708', nome : "Vitorino"},  
6422: {id : '9449', geocod : '4113700', nome : "Warta"},  
6423: {id : '3458', geocod : '4128500', nome : "Wenceslau Braz"},  
6424: {id : '3153', geocod : '4128807', nome : "Xambrê"},  
6425: {id : '9457', geocod : '4128005', nome : "Yolanda"}  
6426: ],  
6427: RJ : [  
6428: // UF = RJ  
6429: {id : '5569', geocod : '3304508', nome : "Abarracamento"},  
6430: {id : '5572', geocod : '3300100', nome : "Abraão"},  
6431: {id : '5583', geocod : '3300951', nome : "Afonso Arinos"},  
6432: {id : '5592', geocod : '3304201', nome : "Agulhas Negras"},  
6433: {id : '5609', geocod : '3305604', nome : "Aldeia Velha"},  
6434: {id : '5683', geocod : '3303401', nome : "Amparo"},  
6435: {id : '5689', geocod : '3306206', nome : "Andrade Pinto"},  
6436: {id : '4075', geocod : '3300100', nome : "Angra dos Reis"},  
6437: {id : '5700', geocod : '3305406', nome : "Anta"},  
6438: {id : '5712', geocod : '3300407', nome : "Antônio Rocha"},  
6439: {id : '4241', geocod : '3300159', nome : "Aperibé"},  
6440: {id : '4198', geocod : '3300209', nome : "Araruama"},  
6441: {id : '4137', geocod : '3300225', nome : "Areal"},  
6442: {id : '4402', geocod : '3300233', nome : "Armação dos Búzios"},  
6443: {id : '4202', geocod : '3300258', nome : "Arraial do Cabo"},  
6444: {id : '5815', geocod : '3303856', nome : "Avelar"},  
6445: {id : '5829', geocod : '3305505', nome : "Bacaxá"},  
6446: {id : '5854', geocod : '3304706', nome : "Baltazar"},  
6447: {id : '5866', geocod : '3300506', nome : "Banquete"},  
6448: {id : '5938', geocod : '3306107', nome : "Barão de Juparanã"},  
6449: {id : '5869', geocod : '3305000', nome : "Barcelos"},  
6450: {id : '5874', geocod : '3300506', nome : "Barra Alegre"},  
6451: {id : '5892', geocod : '3302403', nome : "Barra de Macaé"},  
6452: {id : '5893', geocod : '3301306', nome : "Barra de São João"},  
6453: {id : '4116', geocod : '3300308', nome : "Barra do Piraí"},  
6454: {id : '4079', geocod : '3300407', nome : "Barra Mansa"},  
6455: {id : '5888', geocod : '3304755', nome : "Barra Seca"},  
6456: {id : '5944', geocod : '3302106', nome : "Batatal"},  
6457: {id : '4125', geocod : '3300456', nome : "Belford Roxo"},  
6458: {id : '5991', geocod : '3306008', nome : "Bemposta"},  
6459: {id : '6012', geocod : '3304300', nome : "Boa Esperança"},  
6460: {id : '6019', geocod : '3301108', nome : "Boa Sorte"},  
6461: {id : '6038', geocod : '3302205', nome : "Boaventura"},  
6462: {id : '4205', geocod : '3300506', nome : "Bom Jardim"},  
6463: {id : '4411', geocod : '3300605', nome : "Bom Jesus do Itabapoana"},  
6464: {id : '6058', geocod : '3303104', nome : "Bom Jesus do Querendo"},  
6465: {id : '4203', geocod : '3300704', nome : "Cabo Frio"},  
6466: {id : '6152', geocod : '3301900', nome : "Cabuçu"},  
6467: {id : '4194', geocod : '3300803', nome : "Cachoeiras de Macacu"},  
6468: {id : '6177', geocod : '3302403', nome : "Cachoeiros"},  
6469: {id : '6222', geocod : '3304805', nome : "Cambiasca"},  
6470: {id : '4407', geocod : '3300902', nome : "Cambuci"},  
6471: {id : '6257', geocod : '3303401', nome : "Campo do Coelho"},  
6472: {id : '4416', geocod : '3301009', nome : "Campos dos Goytacazes"},  
6473: {id : '6260', geocod : '3301702', nome : "Campos Elyseos"},  
6474: {id : '4233', geocod : '3301108', nome : "Cantagalo"},  
6475: {id : '6326', geocod : '3300605', nome : "Carabuçu"},  
6476: {id : '4405', geocod : '3300936', nome : "Carapebus"},  
6477: {id : '4412', geocod : '3301157', nome : "Cardoso Moreira"},  
6478: {id : '4213', geocod : '3301207', nome : "Carmo"},  
6479: {id : '6379', geocod : '3303906', nome : "Cascatinha"},  
6480: {id : '4204', geocod : '3301306', nome : "Casimiro de Abreu"},  
6481: {id : '6404', geocod : '3303500', nome : "Cava"},  
6482: {id : '6474', geocod : '3305109', nome : "Coelho da Rocha"},  
6483: {id : '6482', geocod : '3304805', nome : "Colônia"},  
6484: {id : '4136', geocod : '3300951', nome : "Comendador Levy Gasparian"},  
6485: {id : '6493', geocod : '3302205', nome : "Comendador Venâncio"},  
6486: {id : '6503', geocod : '3302601', nome : "Conceição de Jacareí"},  
6487: {id : '4404', geocod : '3301405', nome : "Conceição de Macabu"},  
6488: {id : '6519', geocod : '3302908', nome : "Conrado"},  
6489: {id : '6521', geocod : '3303401', nome : "Conselheiro Paulino"},  
6490: {id : '6523', geocod : '3306107', nome : "Conservatória"},  
6491: {id : '4206', geocod : '3301504', nome : "Cordeiro"},  
6492: {id : '6631', geocod : '3301207', nome : "Córrego da Prata"},  
6493: {id : '6635', geocod : '3302403', nome : "Córrego do Ouro"},  
6494: {id : '6547', geocod : '3305604', nome : "Correntezas"},  
6495: {id : '6608', geocod : '3300100', nome : "Cunhambebe"},  
6496: {id : '6692', geocod : '3300308', nome : "Dorândia"},  
6497: {id : '6689', geocod : '3301009', nome : "Dores de Macabu"},  
6498: {id : '6701', geocod : '3305901', nome : "Doutor Elias"},  
6499: {id : '6702', geocod : '3304607', nome : "Doutor Loréti"},  
6500: {id : '4197', geocod : '3301603', nome : "Duas Barras"},  
6501: {id : '4127', geocod : '3301702', nome : "Duque de Caxias"},  
6502: {id : '6727', geocod : '3304201', nome : "Engenheiro Passos"},  
6503: {id : '4113', geocod : '3301801', nome : "Engo. Paulo de Frontin"},  
6504: {id : '6771', geocod : '3302106', nome : "Estrada Nova"},  
6505: {id : '6788', geocod : '3304128', nome : "Falcão"},  
6506: {id : '6840', geocod : '3300407', nome : "Floriano"},  
6507: {id : '6869', geocod : '3304201', nome : "Fumaça"},  
6508: {id : '6871', geocod : '3300902', nome : "Funil"},  
6509: {id : '6900', geocod : '3305604', nome : "Gaviões"},  
6510: {id : '6908', geocod : '3304409', nome : "Getulândia"},  
6511: {id : '6912', geocod : '3302403', nome : "Glicério"},  
6512: {id : '6928', geocod : '3302908', nome : "Governador Portela"},  
6513: {id : '4191', geocod : '3301850', nome : "Guapimirim"},  
6514: {id : '6992', geocod : '3302502', nome : "Guia de Pacobaíba"},  
6515: {id : '7040', geocod : '3304706', nome : "Ibitiguaçu"},  
6516: {id : '7042', geocod : '3301009', nome : "Ibitioca"},  
6517: {id : '7052', geocod : '3302007', nome : "Ibituporanga"},  
6518: {id : '4199', geocod : '3301876', nome : "Iguaba Grande"},  
6519: {id : '7092', geocod : '3301702', nome : "Imbariê"},  
6520: {id : '7095', geocod : '3303708', nome : "Inconfidência"},  
6521: {id : '7110', geocod : '3302502', nome : "Inhomirim"},  
6522: {id : '7114', geocod : '3302700', nome : "Inoã"},  
6523: {id : '7121', geocod : '3300308', nome : "Ipiabas"},  
6524: {id : '7122', geocod : '3304904', nome : "Ipiiba"},  
6525: {id : '7125', geocod : '3305307', nome : "Ipituna"},  
6526: {id : '7131', geocod : '3304805', nome : "Ipuca"},  
6527: {id : '4187', geocod : '3301900', nome : "Itaboraí"},  
6528: {id : '7162', geocod : '3302601', nome : "Itacurussá"},  
6529: {id : '3716', geocod : '3302007', nome : "Itaguaí"},  
6530: {id : '7174', geocod : '3303906', nome : "Itaipava"},  
6531: {id : '7175', geocod : '3303302', nome : "Itaipu"},  
6532: {id : '7180', geocod : '3302205', nome : "Itajara"},  
6533: {id : '4409', geocod : '3302056', nome : "Italva"},  
6534: {id : '7189', geocod : '3301900', nome : "Itambi"},  
6535: {id : '4239', geocod : '3302106', nome : "Itaocara"},  
6536: {id : '4410', geocod : '3302205', nome : "Itaperuna"},  
6537: {id : '4073', geocod : '3302254', nome : "Itatiaia"},  
6538: {id : '7265', geocod : '3300100', nome : "Jacuecanga"},  
6539: {id : '7275', geocod : '3302106', nome : "Jaguarembé"},  
6540: {id : '7284', geocod : '3305406', nome : "Jamapará"},  
6541: {id : '3719', geocod : '3302270', nome : "Japeri"},  
6542: {id : '7292', geocod : '3300803', nome : "Japuíba"},  
6543: {id : '4247', geocod : '3302304', nome : "Laje do Muriaé"},  
6544: {id : '7439', geocod : '3302106', nome : "Laranjais"},  
6545: {id : '7490', geocod : '3304409', nome : "Lídice"},  
6546: {id : '7484', geocod : '3303401', nome : "Lumiar"},  
6547: {id : '7491', geocod : '3301405', nome : "Macabuzinho"},  
6548: {id : '4403', geocod : '3302403', nome : "Macaé"},  
6549: {id : '4234', geocod : '3302452', nome : "Macuco"},  
6550: {id : '4130', geocod : '3302502', nome : "Magé"},  
6551: {id : '7513', geocod : '3300100', nome : "Mambucaba"},  
6552: {id : '4078', geocod : '3302601', nome : "Mangaratiba"},  
6553: {id : '7526', geocod : '3301900', nome : "Manilha"},  
6554: {id : '7528', geocod : '3304755', nome : "Maniva"},  
6555: {id : '7532', geocod : '3302700', nome : "Manoel Ribeiro"},  
6556: {id : '7535', geocod : '3304508', nome : "Manuel Duarte"},  
6557: {id : '7543', geocod : '3304706', nome : "Marangatu"},  
6558: {id : '4186', geocod : '3302700', nome : "Maricá"},  
6559: {id : '4118', geocod : '3302809', nome : "Mendes"},  
6560: {id : '4123', geocod : '3302858', nome : "Mesquita"},  
6561: {id : '4131', geocod : '3302908', nome : "Miguel Pereira"},  
6562: {id : '4246', geocod : '3303005', nome : "Miracema"},  
6563: {id : '7667', geocod : '3301603', nome : "Monerá"},  
6564: {id : '7671', geocod : '3304904', nome : "Monjolo"},  
6565: {id : '7679', geocod : '3304706', nome : "Monte Alegre"},  
6566: {id : '7704', geocod : '3300902', nome : "Monte Verde"},  
6567: {id : '7709', geocod : '3304003', nome : "Monumento"},  
6568: {id : '7711', geocod : '3301009', nome : "Morangaba"},  
6569: {id : '7727', geocod : '3301009', nome : "Morro do Coco"},  
6570: {id : '7720', geocod : '3300209', nome : "Morro Grande"},  
6571: {id : '7755', geocod : '3301009', nome : "Mussurepe"},  
6572: {id : '4386', geocod : '3303104', nome : "Natividade"},  
6573: {id : '7776', geocod : '3304904', nome : "Neves"},  
6574: {id : '4124', geocod : '3303203', nome : "Nilópolis"},  
6575: {id : '4128', geocod : '3303302', nome : "Niterói"},  
6576: {id : '7786', geocod : '3305406', nome : "Nossa Senhora da Aparecida"},  
6577: {id : '7790', geocod : '3302205', nome : "Nossa Senhora da Penha"},  
6578: {id : '7796', geocod : '3300407', nome : "Nossa Senhora do Amparo"},  
6579: {id : '4195', geocod : '3303401', nome : "Nova Friburgo"},  
6580: {id : '4122', geocod : '3303500', nome : "Nova Iguaçu"},  
6581: {id : '7876', geocod : '3303203', nome : "Olinda"},  
6582: {id : '7901', geocod : '3303104', nome : "Ourânia"},  
6583: {id : '7906', geocod : '3301900', nome : "Pacheco"},  
6584: {id : '4112', geocod : '3303609', nome : "Paracambi"},  
6585: {id : '4133', geocod : '3303708', nome : "Paraíba do Sul"},  
6586: {id : '7986', geocod : '3303005', nome : "Paraíso do Tobias"},  
6587: {id : '7974', geocod : '3304706', nome : "Paraoquena"},  
6588: {id : '3980', geocod : '3303807', nome : "Parati"},  
6589: {id : '7978', geocod : '3303807', nome : "Parati Mirim"},  
6590: {id : '7998', geocod : '3304409', nome : "Passa Três"},  
6591: {id : '4132', geocod : '3303856', nome : "Paty do Alferes"},  
6592: {id : '8050', geocod : '3304201', nome : "Pedra Selada"},  
6593: {id : '8065', geocod : '3303906', nome : "Pedro do Rio"},  
6594: {id : '8079', geocod : '3306107', nome : "Pentagna"},  
6595: {id : '4134', geocod : '3303906', nome : "Petrópolis"},  
6596: {id : '8174', geocod : '3305406', nome : "Pião"},  
6597: {id : '4115', geocod : '3303955', nome : "Pinheiral"},  
6598: {id : '8142', geocod : '3305000', nome : "Pipeiras"},  
6599: {id : '3717', geocod : '3304003', nome : "Piraí"},  
6600: {id : '8153', geocod : '3300605', nome : "Pirapetinga de Bom Jesus"},  
6601: {id : '4314', geocod : '3304102', nome : "Porciúncula"},  
6602: {id : '8214', geocod : '3302106', nome : "Portela"},  
6603: {id : '8234', geocod : '3301900', nome : "Porto das Caixas"},  
6604: {id : '4084', geocod : '3304110', nome : "Porto Real"},  
6605: {id : '8231', geocod : '3301207', nome : "Porto Velho do Cunha"},  
6606: {id : '8240', geocod : '3303906', nome : "Posse"},  
6607: {id : '8270', geocod : '3300100', nome : "Praia de Araçatiba"},  
6608: {id : '8298', geocod : '3304805', nome : "Pureza"},  
6609: {id : '8299', geocod : '3304102', nome : "Purilândia"},  
6610: {id : '4085', geocod : '3304128', nome : "Quatis"},  
6611: {id : '4114', geocod : '3304144', nome : "Queimados"},  
6612: {id : '4406', geocod : '3304151', nome : "Quissamã"},  
6613: {id : '8340', geocod : '3302205', nome : "Raposo"},  
6614: {id : '8352', geocod : '3304607', nome : "Renascença"},  
6615: {id : '4081', geocod : '3304201', nome : "Resende"},  
6616: {id : '8357', geocod : '3302205', nome : "Retiro do Muriaé"},  
6617: {id : '8372', geocod : '3300407', nome : "Rialto"},  
6618: {id : '8383', geocod : '3304128', nome : "Ribeirão de São Joaquim"},  
6619: {id : '4190', geocod : '3304300', nome : "Rio Bonito"},  
6620: {id : '4080', geocod : '3304409', nome : "Rio Claro"},  
6621: {id : '4121', geocod : '3304508', nome : "Rio das Flores"},  
6622: {id : '4384', geocod : '3304524', nome : "Rio das Ostras"},  
6623: {id : '5488', geocod : '3304557', nome : "Rio de Janeiro"},  
6624: {id : '8436', geocod : '3303401', nome : "Riograndina"},  
6625: {id : '8452', geocod : '3300605', nome : "Rosal"},  
6626: {id : '8468', geocod : '3301801', nome : "Sacra Família do Tinguá"},  
6627: {id : '8483', geocod : '3303708', nome : "Salutaris"},  
6628: {id : '8487', geocod : '3301900', nome : "Sambaetiba"},  
6629: {id : '8490', geocod : '3305505', nome : "Sampaio Correia"},  
6630: {id : '8491', geocod : '3302403', nome : "Sana"},  
6631: {id : '8501', geocod : '3304102', nome : "Santa Clara"},  
6632: {id : '8505', geocod : '3304706', nome : "Santa Cruz"},  
6633: {id : '8536', geocod : '3306107', nome : "Santa Isabel do Rio"},  
6634: {id : '8559', geocod : '3301009', nome : "Santa Maria"},  
6635: {id : '4238', geocod : '3304607', nome : "Santa Maria Madalena"},  
6636: {id : '8574', geocod : '3301108', nome : "Santa Rita da Flores"},  
6637: {id : '8616', geocod : '3304003', nome : "Santanésia"},  
6638: {id : '8623', geocod : '3302502', nome : "Santo Aleixo"},  
6639: {id : '8624', geocod : '3301009', nome : "Santo Amaro de Campo"},  
6640: {id : '4240', geocod : '3304706', nome : "Santo Antônio de Pádua"},  
6641: {id : '8655', geocod : '3304607', nome : "Santo Antônio do Imb"},  
6642: {id : '8676', geocod : '3301009', nome : "Santo Eduardo"},  
6643: {id : '4408', geocod : '3304805', nome : "São Fidélis"},  
6644: {id : '4454', geocod : '3304755', nome : "São Francisco de Itabapoana"},  
6645: {id : '4129', geocod : '3304904', nome : "São Gonçalo"},  
6646: {id : '4417', geocod : '3305000', nome : "São João da Barra"},  
6647: {id : '4126', geocod : '3305109', nome : "São João de Meriti"},  
6648: {id : '8966', geocod : '3300902', nome : "São João do Paraíso"},  
6649: {id : '8945', geocod : '3304409', nome : "São João Marcos"},  
6650: {id : '8875', geocod : '3301157', nome : "São Joaquim"},  
6651: {id : '4385', geocod : '3305133', nome : "São José de Ubá"},  
6652: {id : '8930', geocod : '3300506', nome : "São José do Ribeirão"},  
6653: {id : '8936', geocod : '3300308', nome : "São José do Turvo"},  
6654: {id : '4193', geocod : '3305158', nome : "São José do Vale do Rio Pret"},  
6655: {id : '8996', geocod : '3305109', nome : "São Mateus"},  
6656: {id : '4201', geocod : '3305208', nome : "São Pedro da Aldeia"},  
6657: {id : '9023', geocod : '3303401', nome : "São Pedro da Serra"},  
6658: {id : '9025', geocod : '3304706', nome : "São Pedro de Alcântara"},  
6659: {id : '9069', geocod : '3301009', nome : "São Sebastião de Campos"},  
6660: {id : '4237', geocod : '3305307', nome : "São Sebastião do Alto"},  
6661: {id : '9080', geocod : '3301108', nome : "São Sebastião do Par"},  
6662: {id : '9086', geocod : '3306206', nome : "São Sebastião dos Ferreiros"},  
6663: {id : '9106', geocod : '3300209', nome : "São Vicente de Paula"},  
6664: {id : '4208', geocod : '3305406', nome : "Sapucaia"},  
6665: {id : '4188', geocod : '3305505', nome : "Saquarema"},  
6666: {id : '8707', geocod : '3306206', nome : "Sebastião de Lacerda"},  
6667: {id : '3718', geocod : '3305554', nome : "Seropédica"},  
6668: {id : '8747', geocod : '3301009', nome : "Serrinha"},  
6669: {id : '8761', geocod : '3304904', nome : "Sete Pontes"},  
6670: {id : '4200', geocod : '3305604', nome : "Silva Jardim"},  
6671: {id : '8783', geocod : '3305901', nome : "Sodrelândia"},  
6672: {id : '8790', geocod : '3304607', nome : "Sossego"},  
6673: {id : '8798', geocod : '3300803', nome : "Subaio"},  
6674: {id : '4196', geocod : '3305703', nome : "Sumidouro"},  
6675: {id : '8807', geocod : '3302502', nome : "Suruí"},  
6676: {id : '9127', geocod : '3304508', nome : "Taboas"},  
6677: {id : '9145', geocod : '3300704', nome : "Tamoios"},  
6678: {id : '4189', geocod : '3305752', nome : "Tanguá"},  
6679: {id : '9171', geocod : '3303807', nome : "Tarituba"},  
6680: {id : '4192', geocod : '3305802', nome : "Teresópolis"},  
6681: {id : '9214', geocod : '3301009', nome : "Tocos"},  
6682: {id : '4207', geocod : '3305901', nome : "Trajano de Morais"},  
6683: {id : '9232', geocod : '3301009', nome : "Travessão"},  
6684: {id : '9249', geocod : '3300902', nome : "Três Irmãos"},  
6685: {id : '4135', geocod : '3306008', nome : "Três Rios"},  
6686: {id : '9238', geocod : '3304607', nome : "Triunfo"},  
6687: {id : '9326', geocod : '3305307', nome : "Valão do Barro"},  
6688: {id : '9318', geocod : '3305802', nome : "Vale de Bonsucesso"},  
6689: {id : '9319', geocod : '3305802', nome : "Vale do Paquequer"},  
6690: {id : '4120', geocod : '3306107', nome : "Valença"},  
6691: {id : '9330', geocod : '3300308', nome : "Vargem Alegre"},  
6692: {id : '4422', geocod : '3306156', nome : "Varre-Sai"},  
6693: {id : '4119', geocod : '3306206', nome : "Vassouras"},  
6694: {id : '9348', geocod : '3303005', nome : "Venda das Flores"},  
6695: {id : '9410', geocod : '3305901', nome : "Vila da Grama"},  
6696: {id : '9379', geocod : '3302601', nome : "Vila Muriqui"},  
6697: {id : '9387', geocod : '3301009', nome : "Vila Nova de Campos"},  
6698: {id : '9418', geocod : '3305901', nome : "Visconde de Imbé"},  
6699: {id : '9419', geocod : '3301900', nome : "Visconde de Itaboraí"},  
6700: {id : '4086', geocod : '3306305', nome : "Volta Redonda"},  
6701: {id : '9450', geocod : '3303708', nome : "Werneck"},  
6702: {id : '9454', geocod : '3301702', nome : "Xerém"}  
6703: ],  
6704: RN : [  
6705: // UF = RN  
6706: {id : '1044', geocod : '2400109', nome : "Acari"},  
6707: {id : '1078', geocod : '2400208', nome : "Açu"},  
6708: {id : '1100', geocod : '2400307', nome : "Afonso Bezerra"},  
6709: {id : '886', geocod : '2400406', nome : "Água Nova"},  
6710: {id : '851', geocod : '2400505', nome : "Alexandria"},  
6711: {id : '908', geocod : '2400604', nome : "Almino Afonso"},  
6712: {id : '1081', geocod : '2400703', nome : "Alto do Rodrigues"},  
6713: {id : '1098', geocod : '2400802', nome : "Angicos"},  
6714: {id : '902', geocod : '2400901', nome : "Antônio Martins"},  
6715: {id : '918', geocod : '2401008', nome : "Apodi"},  
6716: {id : '1089', geocod : '2401107', nome : "Areia Branca"},  
6717: {id : '1275', geocod : '2401206', nome : "Arês"},  
6718: {id : '1067', geocod : '2401305', nome : "Augusto Severo"},  
6719: {id : '1262', geocod : '2401404', nome : "Baía Formosa"},  
6720: {id : '5940', geocod : '2411403', nome : "Barão de Serra Branca"},  
6721: {id : '1083', geocod : '2401453', nome : "Baraúna"},  
6722: {id : '1165', geocod : '2401503', nome : "Barcelona"},  
6723: {id : '5985', geocod : '2401602', nome : "Belo Horizonte"},  
6724: {id : '1109', geocod : '2401602', nome : "Bento Fernandes"},  
6725: {id : '1097', geocod : '2401651', nome : "Bodó"},  
6726: {id : '1269', geocod : '2401701', nome : "Bom Jesus"},  
6727: {id : '1272', geocod : '2401800', nome : "Brejinho"},  
6728: {id : '663', geocod : '2401859', nome : "Caiçara do Norte"},  
6729: {id : '1107', geocod : '2401909', nome : "Caiçara do Rio do Vento"},  
6730: {id : '1010', geocod : '2402006', nome : "Caicó"},  
6731: {id : '1096', geocod : '2402105', nome : "Campo Redondo"},  
6732: {id : '1249', geocod : '2402204', nome : "Canguaretama"},  
6733: {id : '1064', geocod : '2402303', nome : "Caraúbas"},  
6734: {id : '1045', geocod : '2402402', nome : "Carnaúba dos Dantas"},  
6735: {id : '1080', geocod : '2402501', nome : "Carnaubais"},  
6736: {id : '1283', geocod : '2402600', nome : "Ceará-Mirim"},  
6737: {id : '1095', geocod : '2402709', nome : "Cerro Corá"},  
6738: {id : '1053', geocod : '2402808', nome : "Coronel Ezequiel"},  
6739: {id : '883', geocod : '2402907', nome : "Coronel João Pessoa"},  
6740: {id : '6632', geocod : '2405306', nome : "Córrego de São Mateus"},  
6741: {id : '1017', geocod : '2403004', nome : "Cruzeta"},  
6742: {id : '1092', geocod : '2403103', nome : "Currais Novos"},  
6743: {id : '887', geocod : '2403202', nome : "Dr. Severiano"},  
6744: {id : '888', geocod : '2403301', nome : "Encanto"},  
6745: {id : '1006', geocod : '2403400', nome : "Equador"},  
6746: {id : '1247', geocod : '2403509', nome : "Espírito Santo"},  
6747: {id : '1285', geocod : '2403608', nome : "Extremoz"},  
6748: {id : '1068', geocod : '2403707', nome : "Felipe Guerra"},  
6749: {id : '1099', geocod : '2403756', nome : "Fernando Pedroza"},  
6750: {id : '6830', geocod : '2406700', nome : "Firmamento"},  
6751: {id : '1074', geocod : '2403806', nome : "Florânia"},  
6752: {id : '898', geocod : '2403905', nome : "Francisco Dantas"},  
6753: {id : '906', geocod : '2404002', nome : "Frutuoso Gomes"},  
6754: {id : '662', geocod : '2404101', nome : "Galinhos"},  
6755: {id : '6886', geocod : '2413904', nome : "Gameleira"},  
6756: {id : '1274', geocod : '2404200', nome : "Goianinha"},  
6757: {id : '1069', geocod : '2404309', nome : "Governador Dix-Sept Rosado"},  
6758: {id : '1088', geocod : '2404408', nome : "Grossos"},  
6759: {id : '661', geocod : '2404507', nome : "Guamaré"},  
6760: {id : '1270', geocod : '2404606', nome : "Ielmo Marinho"},  
6761: {id : '7077', geocod : '2412005', nome : "Igreja Nova"},  
6762: {id : '1079', geocod : '2404705', nome : "Ipanguaçu"},  
6763: {id : '7123', geocod : '2412302', nome : "Ipiranga"},  
6764: {id : '1007', geocod : '2404804', nome : "Ipueira"},  
6765: {id : '1077', geocod : '2404853', nome : "Itajá"},  
6766: {id : '912', geocod : '2404903', nome : "Itaú"},  
6767: {id : '1052', geocod : '2405009', nome : "Jaçanã"},  
6768: {id : '659', geocod : '2405108', nome : "Jandaíra"},  
6769: {id : '1066', geocod : '2405207', nome : "Janduís"},  
6770: {id : '1263', geocod : '2405306', nome : "Januário Cicco"},  
6771: {id : '1158', geocod : '2405405', nome : "Japi"},  
6772: {id : '1108', geocod : '2405504', nome : "Jardim de Angicos"},  
6773: {id : '974', geocod : '2405603', nome : "Jardim de Piranhas"},  
6774: {id : '1015', geocod : '2405702', nome : "Jardim do Seridó"},  
6775: {id : '1110', geocod : '2405801', nome : "João Câmara"},  
6776: {id : '905', geocod : '2405900', nome : "João Dias"},  
6777: {id : '836', geocod : '2406007', nome : "José da Penha"},  
6778: {id : '1071', geocod : '2406106', nome : "Jucurutu"},  
6779: {id : '5559', geocod : '2406155', nome : "Jundiá"},  
6780: {id : '7356', geocod : '2406155', nome : "Jundiá de Cima"},  
6781: {id : '1238', geocod : '2406205', nome : "Lagoa d'Anta"},  
6782: {id : '1266', geocod : '2406304', nome : "Lagoa de Pedras"},  
6783: {id : '1171', geocod : '2406403', nome : "Lagoa de Velhos"},  
6784: {id : '1093', geocod : '2406502', nome : "Lagoa Nova"},  
6785: {id : '1265', geocod : '2406601', nome : "Lagoa Salgada"},  
6786: {id : '1102', geocod : '2406700', nome : "Lajes"},  
6787: {id : '1103', geocod : '2406809', nome : "Lajes Pintadas"},  
6788: {id : '907', geocod : '2406908', nome : "Lucrécia"},  
6789: {id : '834', geocod : '2407005', nome : "Luís Gomes"},  
6790: {id : '1280', geocod : '2407104', nome : "Macaíba"},  
6791: {id : '660', geocod : '2407203', nome : "Macau"},  
6792: {id : '7507', geocod : '2406007', nome : "Major Felipe"},  
6793: {id : '835', geocod : '2407252', nome : "Major Sales"},  
6794: {id : '895', geocod : '2407302', nome : "Marcelino Vieira"},  
6795: {id : '904', geocod : '2407401', nome : "Martins"},  
6796: {id : '7598', geocod : '2414100', nome : "Mata de São Braz"},  
6797: {id : '1286', geocod : '2407500', nome : "Maxaranguape"},  
6798: {id : '1062', geocod : '2407609', nome : "Messias Targino"},  
6799: {id : '1245', geocod : '2407708', nome : "Montanhas"},  
6800: {id : '1273', geocod : '2407807', nome : "Monte Alegre"},  
6801: {id : '1161', geocod : '2407906', nome : "Monte das Gameleiras"},  
6802: {id : '1084', geocod : '2408003', nome : "Mossoró"},  
6803: {id : '5489', geocod : '2408102', nome : "Natal"},  
6804: {id : '1277', geocod : '2408201', nome : "Nísia Floresta"},  
6805: {id : '1239', geocod : '2408300', nome : "Nova Cruz"},  
6806: {id : '1063', geocod : '2408409', nome : "Olho-d'Água do Borges"},  
6807: {id : '1012', geocod : '2408508', nome : "Ouro Branco"},  
6808: {id : '833', geocod : '2408607', nome : "Paraná"},  
6809: {id : '1073', geocod : '2408706', nome : "Paraú"},  
6810: {id : '1113', geocod : '2408805', nome : "Parazinho"},  
6811: {id : '1042', geocod : '2408904', nome : "Parelhas"},  
6812: {id : '1279', geocod : '2403251', nome : "Parnamirim"},  
6813: {id : '1237', geocod : '2409100', nome : "Passa e Fica"},  
6814: {id : '1271', geocod : '2409209', nome : "Passagem"},  
6815: {id : '1060', geocod : '2409308', nome : "Patu"},  
6816: {id : '896', geocod : '2409407', nome : "Pau dos Ferros"},  
6817: {id : '1112', geocod : '2409506', nome : "Pedra Grande"},  
6818: {id : '658', geocod : '2409605', nome : "Pedra Preta"},  
6819: {id : '1101', geocod : '2409704', nome : "Pedro Avelino"},  
6820: {id : '1248', geocod : '2409803', nome : "Pedro Velho"},  
6821: {id : '1082', geocod : '2409902', nome : "Pendências"},  
6822: {id : '897', geocod : '2410009', nome : "Pilões"},  
6823: {id : '1111', geocod : '2410108', nome : "Poço Branco"},  
6824: {id : '909', geocod : '2410207', nome : "Portalegre"},  
6825: {id : '1090', geocod : '2410256', nome : "Porto do Mangue"},  
6826: {id : '1169', geocod : '2410306', nome : "Presidente Juscelino"},  
6827: {id : '1284', geocod : '2410405', nome : "Pureza"},  
6828: {id : '894', geocod : '2410504', nome : "Rafael Fernandes"},  
6829: {id : '1059', geocod : '2410603', nome : "Rafael Godeiro"},  
6830: {id : '911', geocod : '2410702', nome : "Riacho da Cruz"},  
6831: {id : '885', geocod : '2410801', nome : "Riacho de Santana"},  
6832: {id : '1172', geocod : '2410900', nome : "Riachuelo"},  
6833: {id : '1287', geocod : '2408953', nome : "Rio do Fogo"},  
6834: {id : '901', geocod : '2411007', nome : "Rodolfo Fernandes"},  
6835: {id : '8456', geocod : '2405900', nome : "Rosário"},  
6836: {id : '1166', geocod : '2411106', nome : "Ruy Barbosa"},  
6837: {id : '8484', geocod : '2407401', nome : "Salva Vida"},  
6838: {id : '1104', geocod : '2411205', nome : "Santa Cruz"},  
6839: {id : '8527', geocod : '2406155', nome : "Santa Fé"},  
6840: {id : '1174', geocod : '2409332', nome : "Santa Maria"},  
6841: {id : '8589', geocod : '2411403', nome : "Santa Tereza"},  
6842: {id : '1094', geocod : '2411403', nome : "Santana do Matos"},  
6843: {id : '1014', geocod : '2411429', nome : "Santana do Seridó"},  
6844: {id : '1240', geocod : '2411502', nome : "Santo Antônio"},  
6845: {id : '8627', geocod : '2412005', nome : "Santo Antonio do Potengi"},  
6846: {id : '5500', geocod : '2411601', nome : "São Bento do Norte"},  
6847: {id : '1058', geocod : '2411700', nome : "São Bento do Trairí"},  
6848: {id : '8822', geocod : '2407005', nome : "São Bernardo - Luís Gomes"},  
6849: {id : '1011', geocod : '2411809', nome : "São Fernando"},  
6850: {id : '899', geocod : '2411908', nome : "São Francisco do Oeste"},  
6851: {id : '8858', geocod : '2402303', nome : "São Geraldo"},  
6852: {id : '1281', geocod : '2412005', nome : "São Gonçalo do Amarante"},  
6853: {id : '1008', geocod : '2412104', nome : "São João do Sabugi"},  
6854: {id : '8901', geocod : '2411403', nome : "São José da Passagem"},  
6855: {id : '1276', geocod : '2412203', nome : "São José de Mipibu"},  
6856: {id : '1164', geocod : '2412302', nome : "São José do Campestre"},  
6857: {id : '1016', geocod : '2412401', nome : "São José do Seridó"},  
6858: {id : '882', geocod : '2412500', nome : "São Miguel"},  
6859: {id : '1289', geocod : '2412559', nome : "São Miguel de Touros"},  
6860: {id : '1173', geocod : '2412609', nome : "São Paulo do Potengi"},  
6861: {id : '1268', geocod : '2412708', nome : "São Pedro"},  
6862: {id : '1076', geocod : '2412807', nome : "São Rafael"},  
6863: {id : '1105', geocod : '2412906', nome : "São Tomé"},  
6864: {id : '1091', geocod : '2413003', nome : "São Vicente"},  
6865: {id : '1170', geocod : '2413102', nome : "Senador Elói de Souza"},  
6866: {id : '1278', geocod : '2413201', nome : "Senador Georgino Avelino"},  
6867: {id : '8738', geocod : '2413706', nome : "Serra da Tapuia"},  
6868: {id : '1163', geocod : '2413300', nome : "Serra de São Bento"},  
6869: {id : '1087', geocod : '2413359', nome : "Serra do Mel"},  
6870: {id : '971', geocod : '2413409', nome : "Serra Negra do Norte"},  
6871: {id : '1264', geocod : '2413508', nome : "Serrinha"},  
6872: {id : '903', geocod : '2413557', nome : "Serrinha dos Pintos"},  
6873: {id : '913', geocod : '2413607', nome : "Severiano Melo"},  
6874: {id : '1168', geocod : '2413706', nome : "Sítio Novo"},  
6875: {id : '900', geocod : '2413805', nome : "Taboleiro Grande"},  
6876: {id : '1282', geocod : '2413904', nome : "Taipu"},  
6877: {id : '1167', geocod : '2414001', nome : "Tangará"},  
6878: {id : '849', geocod : '2414100', nome : "Tenente Ananias"},  
6879: {id : '1075', geocod : '2414159', nome : "Tenente Laurentino Cruz"},  
6880: {id : '1085', geocod : '2411056', nome : "Tibau"},  
6881: {id : '1288', geocod : '2414209', nome : "Tibau do Sul"},  
6882: {id : '973', geocod : '2414308', nome : "Timbaúba dos Batistas"},  
6883: {id : '1290', geocod : '2414407', nome : "Touros"},  
6884: {id : '9227', geocod : '2414001', nome : "Trairi"},  
6885: {id : '1072', geocod : '2414456', nome : "Triunfo Potiguar"},  
6886: {id : '914', geocod : '2414506', nome : "Umarizal"},  
6887: {id : '1070', geocod : '2414605', nome : "Upanema"},  
6888: {id : '1246', geocod : '2414704', nome : "Várzea"},  
6889: {id : '832', geocod : '2414753', nome : "Venha-Ver"},  
6890: {id : '1267', geocod : '2414803', nome : "Vera Cruz"},  
6891: {id : '910', geocod : '2414902', nome : "Viçosa"},  
6892: {id : '1261', geocod : '2415008', nome : "Vila Flor"}  
6893: ],  
6894: RO : [  
6895: // UF = RO  
6896: {id : '5574', geocod : '1100205', nome : "Abunã"},  
6897: {id : '1322', geocod : '1100015', nome : "Alta Floresta D'Oeste"},  
6898: {id : '1951', geocod : '1100379', nome : "Alto Alegre dos Parecis"},  
6899: {id : '1316', geocod : '1100403', nome : "Alto Paraíso"},  
6900: {id : '1327', geocod : '1100346', nome : "Alvorada D'Oeste"},  
6901: {id : '5751', geocod : '1100338', nome : "Araras"},  
6902: {id : '1317', geocod : '1100023', nome : "Ariquemes"},  
6903: {id : '1314', geocod : '1100452', nome : "Buritis"},  
6904: {id : '1948', geocod : '1100031', nome : "Cabixi"},  
6905: {id : '1342', geocod : '1100601', nome : "Cacaulândia"},  
6906: {id : '1339', geocod : '1100049', nome : "Cacoal"},  
6907: {id : '6204', geocod : '1100205', nome : "Calama"},  
6908: {id : '1313', geocod : '1100700', nome : "Campo Novo de Rondônia"},  
6909: {id : '1318', geocod : '1100809', nome : "Candeias do Jamari"},  
6910: {id : '1329', geocod : '1100908', nome : "Castanheiras"},  
6911: {id : '1947', geocod : '1100056', nome : "Cerejeiras"},  
6912: {id : '1953', geocod : '1100924', nome : "Chupinguaia"},  
6913: {id : '1949', geocod : '1100064', nome : "Colorado do Oeste"},  
6914: {id : '1946', geocod : '1100072', nome : "Corumbiara"},  
6915: {id : '1943', geocod : '1100080', nome : "Costa Marques"},  
6916: {id : '1349', geocod : '1100940', nome : "Cujubim"},  
6917: {id : '1341', geocod : '1100098', nome : "Espigão D'Oeste"},  
6918: {id : '1326', geocod : '1101005', nome : "Governador Jorge Teixeira"},  
6919: {id : '1310', geocod : '1100106', nome : "Guajará-Mirim"},  
6920: {id : '1319', geocod : '1101104', nome : "Itapuã do Oeste"},  
6921: {id : '7257', geocod : '1100205', nome : "Jaci Paraná"},  
6922: {id : '1343', geocod : '1100114', nome : "Jaru"},  
6923: {id : '1333', geocod : '1100122', nome : "Ji-Paraná"},  
6924: {id : '1348', geocod : '1100130', nome : "Machadinho D'Oeste"},  
6925: {id : '7555', geocod : '1100189', nome : "Marco Rondon"},  
6926: {id : '1340', geocod : '1101203', nome : "Ministro Andreazza"},  
6927: {id : '1324', geocod : '1101302', nome : "Mirante da Serra"},  
6928: {id : '1315', geocod : '1101401', nome : "Monte Negro"},  
6929: {id : '1321', geocod : '1100148', nome : "Nova Brasilândia D'Oeste"},  
6930: {id : '1311', geocod : '1100338', nome : "Nova Mamoré"},  
6931: {id : '1325', geocod : '1101435', nome : "Nova União"},  
6932: {id : '1323', geocod : '1100502', nome : "Novo Horizonte do Oeste"},  
6933: {id : '1332', geocod : '1100155', nome : "Ouro Preto do Oeste"},  
6934: {id : '1952', geocod : '1101450', nome : "Parecis"},  
6935: {id : '8099', geocod : '1100338', nome : "Pic Sidney Girão"},  
6936: {id : '1338', geocod : '1100189', nome : "Pimenta Bueno"},  
6937: {id : '1945', geocod : '1101468', nome : "Pimenteiras do Oeste"},  
6938: {id : '5490', geocod : '1100205', nome : "Porto Velho"},  
6939: {id : '1330', geocod : '1100254', nome : "Presidente Médici"},  
6940: {id : '1337', geocod : '1101476', nome : "Primavera de Rondônia"},  
6941: {id : '8295', geocod : '1100080', nome : "Príncipe da Beira"},  
6942: {id : '1344', geocod : '1100262', nome : "Rio Crespo"},  
6943: {id : '8438', geocod : '1100049', nome : "Riozinho"},  
6944: {id : '1335', geocod : '1100288', nome : "Rolim de Moura"},  
6945: {id : '1334', geocod : '1100296', nome : "Santa Luzia D'Oeste"},  
6946: {id : '8613', geocod : '1100320', nome : "Santana do Guaporé"},  
6947: {id : '1336', geocod : '1101484', nome : "São Felipe D'Oeste"},  
6948: {id : '1944', geocod : '1101492', nome : "São Francisco do Guaporé"},  
6949: {id : '1320', geocod : '1100320', nome : "São Miguel do Guaporé"},  
6950: {id : '1312', geocod : '1101500', nome : "Seringueiras"},  
6951: {id : '1331', geocod : '1101559', nome : "Teixeirópolis"},  
6952: {id : '1345', geocod : '1101609', nome : "Theobroma"},  
6953: {id : '1328', geocod : '1101708', nome : "Urupá"},  
6954: {id : '1347', geocod : '1101757', nome : "Vale do Anari"},  
6955: {id : '1346', geocod : '1101807', nome : "Vale do Paraíso"},  
6956: {id : '1950', geocod : '1100304', nome : "Vilhena"}  
6957: ],  
6958: RR : [  
6959: // UF = RR  
6960: {id : '4', geocod : '1400050', nome : "Alto Alegre"},  
6961: {id : '5', geocod : '1400027', nome : "Amajari"},  
6962: {id : '5491', geocod : '1400100', nome : "Boa Vista"},  
6963: {id : '11', geocod : '1400159', nome : "Bonfim"},  
6964: {id : '8', geocod : '1400175', nome : "Cantá"},  
6965: {id : '1', geocod : '1400209', nome : "Caracaraí"},  
6966: {id : '10', geocod : '1400233', nome : "Caroebe"},  
6967: {id : '2', geocod : '1400282', nome : "Iracema"},  
6968: {id : '3', geocod : '1400308', nome : "Mucajaí"},  
6969: {id : '12', geocod : '1400407', nome : "Normandia"},  
6970: {id : '24', geocod : '1400456', nome : "Pacaraima"},  
6971: {id : '6', geocod : '1400472', nome : "Rorainópolis"},  
6972: {id : '9', geocod : '1400506', nome : "São João da Baliza"},  
6973: {id : '7', geocod : '1400605', nome : "São Luiz"},  
6974: {id : '25', geocod : '1400704', nome : "Uiramutã"}  
6975: ],  
6976: RS : [  
6977: // UF = RS  
6978: {id : '5548', geocod : '4300034', nome : "Aceguá"},  
6979: {id : '5584', geocod : '4318903', nome : "Afonso Rodrigues"},  
6980: {id : '5383', geocod : '4300059', nome : "Água Santa"},  
6981: {id : '5589', geocod : '4313508', nome : "Aguapés"},  
6982: {id : '9476', geocod : '4323002', nome : "Águas Claras"},  
6983: {id : '5153', geocod : '4300109', nome : "Agudo"},  
6984: {id : '5291', geocod : '4300208', nome : "Ajuricaba"},  
6985: {id : '5600', geocod : '4315701', nome : "Albardão"},  
6986: {id : '4461', geocod : '4300307', nome : "Alecrim"},  
6987: {id : '5082', geocod : '4300406', nome : "Alegrete"},  
6988: {id : '4467', geocod : '4300455', nome : "Alegria"},  
6989: {id : '5619', geocod : '4310009', nome : "Alfredo Brenner"},  
6990: {id : '5522', geocod : '4300471', nome : "Almirante Tamandaré do Sul"},  
6991: {id : '4631', geocod : '4300505', nome : "Alpestre"},  
6992: {id : '5306', geocod : '4300554', nome : "Alto Alegre"},  
6993: {id : '5655', geocod : '4310207', nome : "Alto da União"},  
6994: {id : '5639', geocod : '4321907', nome : "Alto Erval Novo"},  
6995: {id : '5256', geocod : '4300570', nome : "Alto Feliz"},  
6996: {id : '5644', geocod : '4316808', nome : "Alto Paredão"},  
6997: {id : '5648', geocod : '4316105', nome : "Alto Recreio"},  
6998: {id : '5652', geocod : '4321477', nome : "Alto Uruguai"},  
6999: {id : '5241', geocod : '4300604', nome : "Alvorada"},  
7000: {id : '5182', geocod : '4300638', nome : "Amaral Ferrador"},  
7001: {id : '4571', geocod : '4300646', nome : "Ametista do Sul"},  
7002: {id : '5188', geocod : '4300661', nome : "André da Rocha"},  
7003: {id : '5337', geocod : '4300703', nome : "Anta Gorda"},  
7004: {id : '5709', geocod : '4305900', nome : "Antônio Kerpel"},  
7005: {id : '5386', geocod : '4300802', nome : "Antônio Prado"},  
7006: {id : '5192', geocod : '4300851', nome : "Arambaré"},  
7007: {id : '5270', geocod : '4300877', nome : "Araricá"},  
7008: {id : '4716', geocod : '4300901', nome : "Aratiba"},  
7009: {id : '5763', geocod : '4304804', nome : "Arco Verde"},  
7010: {id : '5764', geocod : '4307302', nome : "Arco-Íris"},  
7011: {id : '5782', geocod : '4320800', nome : "Arlindo"},  
7012: {id : '5785', geocod : '4313953', nome : "Aroeiras"},  
7013: {id : '5786', geocod : '4301651', nome : "Arroio Canoas"},  
7014: {id : '5210', geocod : '4301008', nome : "Arroio do Meio"},  
7015: {id : '5560', geocod : '4301073', nome : "Arroio do Padre"},  
7016: {id : '5410', geocod : '4301057', nome : "Arroio do Sal"},  
7017: {id : '5791', geocod : '4316907', nome : "Arroio do Sol"},  
7018: {id : '5296', geocod : '4301206', nome : "Arroio do Tigre"},  
7019: {id : '5212', geocod : '4301107', nome : "Arroio dos Ratos"},  
7020: {id : '5464', geocod : '4301305', nome : "Arroio Grande"},  
7021: {id : '5787', geocod : '4316907', nome : "Arroio Grande - Santa Maria"},  
7022: {id : '5788', geocod : '4320305', nome : "Arroio Grande - Selbach"},  
7023: {id : '5789', geocod : '4314175', nome : "Arroio Mau e Candiota"},  
7024: {id : '5790', geocod : '4304630', nome : "Arroio Teixeira"},  
7025: {id : '5797', geocod : '4317301', nome : "Arvore Só"},  
7026: {id : '5328', geocod : '4301404', nome : "Arvorezinha"},  
7027: {id : '5805', geocod : '4317509', nome : "Atafona"},  
7028: {id : '5807', geocod : '4320107', nome : "Atiaçu"},  
7029: {id : '5808', geocod : '4317301', nome : "Atlântico"},  
7030: {id : '5287', geocod : '4301503', nome : "Augusto Pestana"},  
7031: {id : '4717', geocod : '4301552', nome : "Áurea"},  
7032: {id : '5816', geocod : '4310876', nome : "Avelino Paranhos"},  
7033: {id : '5819', geocod : '4318309', nome : "Azevedo Sodré"},  
7034: {id : '5831', geocod : '4313656', nome : "Bacupari"},  
7035: {id : '5093', geocod : '4301602', nome : "Bagé"},  
7036: {id : '5853', geocod : '4308706', nome : "Baliza"},  
7037: {id : '5398', geocod : '4301636', nome : "Balneário Pinhal"},  
7038: {id : '5864', geocod : '4303509', nome : "Bandeirinha"},  
7039: {id : '5865', geocod : '4303509', nome : "Banhado do Colégio"},  
7040: {id : '5233', geocod : '4301651', nome : "Barão"},  
7041: {id : '4712', geocod : '4301701', nome : "Barão de Cotegipe"},  
7042: {id : '5211', geocod : '4301750', nome : "Barão do Triunfo"},  
7043: {id : '4591', geocod : '4301859', nome : "Barra do Guarita"},  
7044: {id : '5079', geocod : '4301875', nome : "Barra do Quaraí"},  
7045: {id : '5237', geocod : '4301909', nome : "Barra do Ribeiro"},  
7046: {id : '4714', geocod : '4301925', nome : "Barra do Rio Azul"},  
7047: {id : '4497', geocod : '4301958', nome : "Barra Funda"},  
7048: {id : '4533', geocod : '4301800', nome : "Barracão"},  
7049: {id : '5908', geocod : '4311759', nome : "Barragem do Itú"},  
7050: {id : '5912', geocod : '4320107', nome : "Barreirinho"},  
7051: {id : '5915', geocod : '4310207', nome : "Barreiro"},  
7052: {id : '5931', geocod : '4309209', nome : "Barro Vermelho"},  
7053: {id : '5930', geocod : '4303004', nome : "Barro Vermelho -Cachoeira do Su"},  
7054: {id : '5308', geocod : '4302006', nome : "Barros Cassal"},  
7055: {id : '5943', geocod : '4307104', nome : "Basílio"},  
7056: {id : '5962', geocod : '4304358', nome : "Baú"},  
7057: {id : '5970', geocod : '4310876', nome : "Bela Vista - Jacuizinho"},  
7058: {id : '5971', geocod : '4311700', nome : "Bela Vista - Machadinho"},  
7059: {id : '5972', geocod : '4314100', nome : "Bela Vista - Passo Fundo"},  
7060: {id : '5973', geocod : '4320263', nome : "Bela Vista - Segredo"},  
7061: {id : '5974', geocod : '4321907', nome : "Bela Vista - Tres Passos"},  
7062: {id : '5987', geocod : '4318101', nome : "Beluno"},  
7063: {id : '4588', geocod : '4302055', nome : "Benjamin Constant do Sul"},  
7064: {id : '5363', geocod : '4302105', nome : "Bento Gonçalves"},  
7065: {id : '6002', geocod : '4315701', nome : "Bexiga"},  
7066: {id : '6013', geocod : '4316006', nome : "Boa Esperança"},  
7067: {id : '6024', geocod : '4316808', nome : "Boa Vista - Santa Cruz do Su"},  
7068: {id : '6025', geocod : '4318101', nome : "Boa Vista - São Francisco de"},  
7069: {id : '6026', geocod : '4318804', nome : "Boa Vista - São Lourenço do"},  
7070: {id : '4499', geocod : '4302154', nome : "Boa Vista das Missões"},  
7071: {id : '4475', geocod : '4302204', nome : "Boa Vista do Buricá"},  
7072: {id : '5537', geocod : '4302220', nome : "Boa Vista do Cadeado"},  
7073: {id : '5538', geocod : '4302238', nome : "Boa Vista do Incra"},  
7074: {id : '5353', geocod : '4302253', nome : "Boa Vista do Sul"},  
7075: {id : '6041', geocod : '4316907', nome : "Boca do Monte"},  
7076: {id : '6043', geocod : '4305306', nome : "Boi Preto"},  
7077: {id : '6046', geocod : '4318507', nome : "Bojuru"},  
7078: {id : '6049', geocod : '4302808', nome : "Bom Jardim"},  
7079: {id : '5396', geocod : '4302303', nome : "Bom Jesus"},  
7080: {id : '5253', geocod : '4302352', nome : "Bom Princípio"},  
7081: {id : '4554', geocod : '4302378', nome : "Bom Progresso"},  
7082: {id : '6065', geocod : '4303608', nome : "Bom Retiro"},  
7083: {id : '5207', geocod : '4302402', nome : "Bom Retiro do Sul"},  
7084: {id : '6067', geocod : '4306056', nome : "Bom Será"},  
7085: {id : '6082', geocod : '4303509', nome : "Bonito"},  
7086: {id : '6083', geocod : '4308805', nome : "Boqueirão - General Câmara"},  
7087: {id : '6084', geocod : '4311304', nome : "Boqueirão - Lagoa Vermelha"},  
7088: {id : '6085', geocod : '4318804', nome : "Boqueirão - São Lourenço do"},  
7089: {id : '5323', geocod : '4302451', nome : "Boqueirão do Leão"},  
7090: {id : '6092', geocod : '4311718', nome : "Bororé"},  
7091: {id : '6093', geocod : '4303004', nome : "Bosque"},  
7092: {id : '5098', geocod : '4302501', nome : "Bossoroca"},  
7093: {id : '6095', geocod : '4307104', nome : "Bote"},  
7094: {id : '6097', geocod : '4304200', nome : "Botucaraí"},  
7095: {id : '5545', geocod : '4302584', nome : "Bozano"},  
7096: {id : '4556', geocod : '4302600', nome : "Braga"},  
7097: {id : '5228', geocod : '4302659', nome : "Brochier"},  
7098: {id : '6122', geocod : '4318606', nome : "Brugnarotto"},  
7099: {id : '6131', geocod : '4320800', nome : "Bugre"},  
7100: {id : '6136', geocod : '4317509', nome : "Buriti"},  
7101: {id : '5199', geocod : '4302709', nome : "Butiá"},  
7102: {id : '6143', geocod : '4321352', nome : "Butiás"},  
7103: {id : '5139', geocod : '4302808', nome : "Caçapava do Sul"},  
7104: {id : '5088', geocod : '4302907', nome : "Cacequi"},  
7105: {id : '5169', geocod : '4303004', nome : "Cachoeira do Sul"},  
7106: {id : '5242', geocod : '4303103', nome : "Cachoeirinha"},  
7107: {id : '4526', geocod : '4303202', nome : "Cacique Doble"},  
7108: {id : '6178', geocod : '4302220', nome : "Cadeado"},  
7109: {id : '5109', geocod : '4303301', nome : "Caibaté"},  
7110: {id : '4503', geocod : '4303400', nome : "Caiçara"},  
7111: {id : '5183', geocod : '4303509', nome : "Camaquã"},  
7112: {id : '5339', geocod : '4303558', nome : "Camargo"},  
7113: {id : '5413', geocod : '4303608', nome : "Cambará do Sul"},  
7114: {id : '6229', geocod : '4319158', nome : "Campestre"},  
7115: {id : '5391', geocod : '4303673', nome : "Campestre da Serra"},  
7116: {id : '4457', geocod : '4303707', nome : "Campina das Missões"},  
7117: {id : '6235', geocod : '4307500', nome : "Campina Redonda"},  
7118: {id : '6239', geocod : '4311304', nome : "Campinas - Lagoa Vermelha"},  
7119: {id : '6240', geocod : '4316436', nome : "Campinas - Saldanha Marinho"},  
7120: {id : '4583', geocod : '4303806', nome : "Campinas do Sul"},  
7121: {id : '6241', geocod : '4315800', nome : "Campinho"},  
7122: {id : '6243', geocod : '4305801', nome : "Campinzal"},  
7123: {id : '5264', geocod : '4303905', nome : "Campo Bom"},  
7124: {id : '6247', geocod : '4315156', nome : "Campo Branco"},  
7125: {id : '6258', geocod : '4308854', nome : "Campo do Meio"},  
7126: {id : '4555', geocod : '4304002', nome : "Campo Novo"},  
7127: {id : '6253', geocod : '4308300', nome : "Campo Novo - Fontoura Xavier"},  
7128: {id : '6255', geocod : '4305900', nome : "Campo Santo - Coronel Bicaco"},  
7129: {id : '6256', geocod : '4316402', nome : "Campo Sêco"},  
7130: {id : '5301', geocod : '4304101', nome : "Campos Borges"},  
7131: {id : '5172', geocod : '4304200', nome : "Candelária"},  
7132: {id : '6627', geocod : '4309001', nome : "Cândido Freire"},  
7133: {id : '4458', geocod : '4304309', nome : "Cândido Godói"},  
7134: {id : '5137', geocod : '4304358', nome : "Candiota"},  
7135: {id : '5393', geocod : '4304408', nome : "Canela"},  
7136: {id : '5150', geocod : '4304507', nome : "Canguçu"},  
7137: {id : '6288', geocod : '4313102', nome : "Canhemborá"},  
7138: {id : '5239', geocod : '4304606', nome : "Canoas"},  
7139: {id : '6298', geocod : '4304614', nome : "Canudos"},  
7140: {id : '5533', geocod : '4304614', nome : "Canudos do Vale"},  
7141: {id : '6300', geocod : '4303004', nome : "Capané"},  
7142: {id : '6315', geocod : '4316709', nome : "Capão Alto"},  
7143: {id : '6316', geocod : '4304622', nome : "Capão Bonito - Capão Bonito"},  
7144: {id : '6317', geocod : '4316451', nome : "Capão Bonito - Salto do Jacu"},  
7145: {id : '5523', geocod : '4304622', nome : "Capão Bonito do Sul"},  
7146: {id : '6318', geocod : '4321352', nome : "Capão Comprido"},  
7147: {id : '5408', geocod : '4304630', nome : "Capão da Canoa"},  
7148: {id : '6323', geocod : '4323002', nome : "Capão da Porteira"},  
7149: {id : '6324', geocod : '4311304', nome : "Capão do Cedro"},  
7150: {id : '5536', geocod : '4304655', nome : "Capão do Cipó"},  
7151: {id : '5134', geocod : '4304663', nome : "Capão do Leão"},  
7152: {id : '6325', geocod : '4302303', nome : "Capão do Tigre"},  
7153: {id : '6320', geocod : '4304630', nome : "Capão Novo"},  
7154: {id : '5245', geocod : '4304689', nome : "Capela de Santana"},  
7155: {id : '6301', geocod : '4318903', nome : "Capela São Paulo"},  
7156: {id : '6302', geocod : '4303509', nome : "Capela São Pedro"},  
7157: {id : '6303', geocod : '4305447', nome : "Capela Velha"},  
7158: {id : '5332', geocod : '4304697', nome : "Capitão"},  
7159: {id : '6307', geocod : '4307302', nome : "Capivara"},  
7160: {id : '5274', geocod : '4304671', nome : "Capivari do Sul"},  
7161: {id : '6309', geocod : '4307005', nome : "Capo Erê"},  
7162: {id : '5282', geocod : '4304713', nome : "Caraá"},  
7163: {id : '6334', geocod : '4302808', nome : "Carajá Seival"},  
7164: {id : '5320', geocod : '4304705', nome : "Carazinho"},  
7165: {id : '5364', geocod : '4304804', nome : "Carlos Barbosa"},  
7166: {id : '4514', geocod : '4304853', nome : "Carlos Gomes"},  
7167: {id : '6364', geocod : '4304655', nome : "Caroví"},  
7168: {id : '6376', geocod : '4302303', nome : "Casa Branca"},  
7169: {id : '5346', geocod : '4304903', nome : "Casca"},  
7170: {id : '6377', geocod : '4309605', nome : "Cascata - Horizontina"},  
7171: {id : '6378', geocod : '4314407', nome : "Cascata - Pelotas"},  
7172: {id : '5161', geocod : '4304952', nome : "Caseiros"},  
7173: {id : '6382', geocod : '4308508', nome : "Castelinho"},  
7174: {id : '6387', geocod : '4317608', nome : "Catanduva Grande"},  
7175: {id : '6389', geocod : '4317103', nome : "Cati"},  
7176: {id : '5128', geocod : '4305009', nome : "Catuípe"},  
7177: {id : '6405', geocod : '4319802', nome : "Cavajuretã"},  
7178: {id : '6408', geocod : '4316402', nome : "Caverá"},  
7179: {id : '5164', geocod : '4305108', nome : "Caxias do Sul"},  
7180: {id : '6412', geocod : '4318200', nome : "Cazuza Ferreira"},  
7181: {id : '4513', geocod : '4305116', nome : "Centenário"},  
7182: {id : '6422', geocod : '4322608', nome : "Centro Linha Brasil"},  
7183: {id : '5133', geocod : '4305124', nome : "Cerrito"},  
7184: {id : '6427', geocod : '4314407', nome : "Cerrito Alegre"},  
7185: {id : '6428', geocod : '4319604', nome : "Cerrito do Ouro"},  
7186: {id : '6429', geocod : '4322301', nome : "Cerro Alto"},  
7187: {id : '6430', geocod : '4318606', nome : "Cerro Azul"},  
7188: {id : '5171', geocod : '4305132', nome : "Cerro Branco"},  
7189: {id : '6431', geocod : '4307104', nome : "Cerro Chato"},  
7190: {id : '6432', geocod : '4319406', nome : "Cerro Claro"},  
7191: {id : '6433', geocod : '4317004', nome : "Cerro do Martins"},  
7192: {id : '6434', geocod : '4302709', nome : "Cerro do Roque"},  
7193: {id : '4509', geocod : '4305157', nome : "Cerro Grande"},  
7194: {id : '5191', geocod : '4305173', nome : "Cerro Grande do Sul"},  
7195: {id : '5108', geocod : '4305207', nome : "Cerro Largo"},  
7196: {id : '5159', geocod : '4305306', nome : "Chapada"},  
7197: {id : '6437', geocod : '4311122', nome : "Chapada - Jaquirana"},  
7198: {id : '5218', geocod : '4305355', nome : "Charqueadas"},  
7199: {id : '4711', geocod : '4305371', nome : "Charrua"},  
7200: {id : '4548', geocod : '4305405', nome : "Chiapetta"},  
7201: {id : '6440', geocod : '4317608', nome : "Chicolomã"},  
7202: {id : '6444', geocod : '4310207', nome : "Chorão"},  
7203: {id : '5465', geocod : '4305439', nome : "Chuí"},  
7204: {id : '5130', geocod : '4305447', nome : "Chuvisca"},  
7205: {id : '5401', geocod : '4305454', nome : "Cidreira"},  
7206: {id : '6452', geocod : '4304804', nome : "Cinco da Boa Vista"},  
7207: {id : '6453', geocod : '4322301', nome : "Cinquentenário"},  
7208: {id : '5375', geocod : '4305504', nome : "Ciríaco"},  
7209: {id : '6459', geocod : '4312104', nome : "Clara"},  
7210: {id : '6465', geocod : '4311304', nome : "Clemente Argolo"},  
7211: {id : '6475', geocod : '4319158', nome : "Coimbra"},  
7212: {id : '5222', geocod : '4305587', nome : "Colinas"},  
7213: {id : '6479', geocod : '4309407', nome : "Colombo"},  
7214: {id : '6491', geocod : '4305009', nome : "Colônia das Almas"},  
7215: {id : '6483', geocod : '4310405', nome : "Colônia Medeiros"},  
7216: {id : '6484', geocod : '4317509', nome : "Colônia Municipal"},  
7217: {id : '6486', geocod : '4300034', nome : "Colônia Nova"},  
7218: {id : '6489', geocod : '4306106', nome : "Colônia São João"},  
7219: {id : '6490', geocod : '4314407', nome : "Colônia Z/3"},  
7220: {id : '6480', geocod : '4301206', nome : "Coloninha"},  
7221: {id : '5185', geocod : '4305603', nome : "Colorado"},  
7222: {id : '6492', geocod : '4317509', nome : "Comandaí"},  
7223: {id : '5294', geocod : '4305702', nome : "Condor"},  
7224: {id : '6524', geocod : '4321808', nome : "Consolata"},  
7225: {id : '4511', geocod : '4305801', nome : "Constantina"},  
7226: {id : '6525', geocod : '4315156', nome : "Constantino"},  
7227: {id : '5532', geocod : '4305835', nome : "Coqueiro Baixo"},  
7228: {id : '5321', geocod : '4305850', nome : "Coqueiros do Sul"},  
7229: {id : '6533', geocod : '4306056', nome : "Cordeiro"},  
7230: {id : '6535', geocod : '4303004', nome : "Cordilheira"},  
7231: {id : '5124', geocod : '4305871', nome : "Coronel Barros"},  
7232: {id : '4561', geocod : '4305900', nome : "Coronel Bicaco"},  
7233: {id : '6538', geocod : '4307302', nome : "Coronel Finzito"},  
7234: {id : '5531', geocod : '4305934', nome : "Coronel Pilar"},  
7235: {id : '6542', geocod : '4311908', nome : "Coronel Teixeira"},  
7236: {id : '6552', geocod : '4322004', nome : "Costa da Cadeia"},  
7237: {id : '6553', geocod : '4312401', nome : "Costa da Serra"},  
7238: {id : '6556', geocod : '4307807', nome : "Costão - Estrela"},  
7239: {id : '6557', geocod : '4312443', nome : "Costão - Morrinhos do Sul"},  
7240: {id : '5359', geocod : '4305959', nome : "Cotiporã"},  
7241: {id : '5380', geocod : '4305975', nome : "Coxilha"},  
7242: {id : '6565', geocod : '4307104', nome : "Coxilha do Lageado"},  
7243: {id : '6562', geocod : '4322509', nome : "Coxilha Grande"},  
7244: {id : '6564', geocod : '4321634', nome : "Coxilha Seca"},  
7245: {id : '6568', geocod : '4313656', nome : "Cr 1"},  
7246: {id : '4477', geocod : '4306007', nome : "Crissiumal"},  
7247: {id : '5181', geocod : '4306056', nome : "Cristal"},  
7248: {id : '4505', geocod : '4306072', nome : "Cristal do Sul"},  
7249: {id : '6578', geocod : '4309407', nome : "Cristo Redentor"},  
7250: {id : '6581', geocod : '4317509', nome : "Cristo Rei"},  
7251: {id : '6584', geocod : '4305108', nome : "Criúva"},  
7252: {id : '5289', geocod : '4306106', nome : "Cruz Alta"},  
7253: {id : '5524', geocod : '4306130', nome : "Cruzaltense"},  
7254: {id : '6594', geocod : '4305504', nome : "Cruzaltinha"},  
7255: {id : '5206', geocod : '4306205', nome : "Cruzeiro do Sul"},  
7256: {id : '6613', geocod : '4317301', nome : "Curral Alto"},  
7257: {id : '6617', geocod : '4304630', nome : "Curumim"},  
7258: {id : '6642', geocod : '4321402', nome : "Daltro Filho"},  
7259: {id : '5374', geocod : '4306304', nome : "David Canabarro"},  
7260: {id : '6648', geocod : '4307807', nome : "Delfina"},  
7261: {id : '6650', geocod : '4322608', nome : "Deodoro"},  
7262: {id : '6653', geocod : '4307500', nome : "Depósito"},  
7263: {id : '4559', geocod : '4306320', nome : "Derrubadas"},  
7264: {id : '5102', geocod : '4306353', nome : "Dezesseis de Novembro"},  
7265: {id : '5115', geocod : '4306379', nome : "Dilermando de Aguiar"},  
7266: {id : '5266', geocod : '4306403', nome : "Dois Irmãos"},  
7267: {id : '4563', geocod : '4306429', nome : "Dois Irmãos das Missões"},  
7268: {id : '5356', geocod : '4306452', nome : "Dois Lajeados"},  
7269: {id : '5129', geocod : '4306502', nome : "Dom Feliciano"},  
7270: {id : '5084', geocod : '4306601', nome : "Dom Pedrito"},  
7271: {id : '5412', geocod : '4306551', nome : "Dom Pedro de Alcântara"},  
7272: {id : '5156', geocod : '4306700', nome : "Dona Francisca"},  
7273: {id : '6686', geocod : '4316303', nome : "Dona Otília"},  
7274: {id : '6693', geocod : '4301909', nome : "Douradilho"},  
7275: {id : '6695', geocod : '4300901', nome : "Dourado"},  
7276: {id : '6698', geocod : '4302584', nome : "Doutor Bozano"},  
7277: {id : '6700', geocod : '4312500', nome : "Doutor Edgardo Pereira"},  
7278: {id : '5336', geocod : '4306759', nome : "Doutor Ricardo"},  
7279: {id : '4472', geocod : '4306734', nome : "Dr. Maurício Cardoso"},  
7280: {id : '5217', geocod : '4306767', nome : "Eldorado do Sul"},  
7281: {id : '6712', geocod : '4318200', nome : "Eletra"},  
7282: {id : '5350', geocod : '4306809', nome : "Encantado"},  
7283: {id : '6720', geocod : '4311718', nome : "Encruzilhada"},  
7284: {id : '5151', geocod : '4306908', nome : "Encruzilhada do Sul"},  
7285: {id : '6733', geocod : '4300059', nome : "Engenho Grande"},  
7286: {id : '4575', geocod : '4306924', nome : "Engenho Velho"},  
7287: {id : '4578', geocod : '4306957', nome : "Entre Rios do Sul"},  
7288: {id : '5121', geocod : '4306932', nome : "Entre-Ijuís"},  
7289: {id : '6743', geocod : '4321204', nome : "Entrepelado"},  
7290: {id : '4706', geocod : '4306973', nome : "Erebango"},  
7291: {id : '4713', geocod : '4307005', nome : "Erechim"},  
7292: {id : '5319', geocod : '4307054', nome : "Ernestina"},  
7293: {id : '6748', geocod : '4317400', nome : "Ernesto Alves"},  
7294: {id : '4589', geocod : '4307203', nome : "Erval Grande"},  
7295: {id : '4564', geocod : '4307302', nome : "Erval Seco"},  
7296: {id : '5397', geocod : '4307401', nome : "Esmeralda"},  
7297: {id : '6753', geocod : '4318804', nome : "Esperança"},  
7298: {id : '4558', geocod : '4307450', nome : "Esperança do Sul"},  
7299: {id : '6756', geocod : '4323002', nome : "Espigão"},  
7300: {id : '6757', geocod : '4301800', nome : "Espigão Alto"},  
7301: {id : '6759', geocod : '4317103', nome : "Espinilho"},  
7302: {id : '6760', geocod : '4319158', nome : "Espinilho Grande"},  
7303: {id : '6762', geocod : '4300455', nome : "Espírito Santo"},  
7304: {id : '5309', geocod : '4307500', nome : "Espumoso"},  
7305: {id : '6764', geocod : '4310405', nome : "Esquina Araújo"},  
7306: {id : '6765', geocod : '4305009', nome : "Esquina Bom Sucesso"},  
7307: {id : '6766', geocod : '4306007', nome : "Esquina Gaúcha"},  
7308: {id : '6767', geocod : '4302501', nome : "Esquina Piratini"},  
7309: {id : '4708', geocod : '4307559', nome : "Estação"},  
7310: {id : '6779', geocod : '4300604', nome : "Estância Grande"},  
7311: {id : '5250', geocod : '4307609', nome : "Estância Velha"},  
7312: {id : '6780', geocod : '4321600', nome : "Estância Velha - Tramandaí"},  
7313: {id : '5246', geocod : '4307708', nome : "Esteio"},  
7314: {id : '6772', geocod : '4318507', nome : "Estreito"},  
7315: {id : '5208', geocod : '4307807', nome : "Estrela"},  
7316: {id : '6774', geocod : '4322509', nome : "Estrela - Vacaria"},  
7317: {id : '5297', geocod : '4307815', nome : "Estrela Velha"},  
7318: {id : '5123', geocod : '4307831', nome : "Eugênio de Castro"},  
7319: {id : '6785', geocod : '4304903', nome : "Evangelista"},  
7320: {id : '5360', geocod : '4307864', nome : "Fagundes Varela"},  
7321: {id : '6790', geocod : '4302105', nome : "Faria Lemos"},  
7322: {id : '6792', geocod : '4300505', nome : "Farinhas"},  
7323: {id : '5163', geocod : '4307906', nome : "Farroupilha"},  
7324: {id : '6794', geocod : '4307054', nome : "Faxinal - Ernestina"},  
7325: {id : '6795', geocod : '4318804', nome : "Faxinal - São Lourenço do Su"},  
7326: {id : '5154', geocod : '4308003', nome : "Faxinal do Soturno"},  
7327: {id : '4587', geocod : '4308052', nome : "Faxinalzinho"},  
7328: {id : '6797', geocod : '4308458', nome : "Fazenda Colorado"},  
7329: {id : '6798', geocod : '4321204', nome : "Fazenda Fialho"},  
7330: {id : '6802', geocod : '4314159', nome : "Fazenda São José"},  
7331: {id : '6801', geocod : '4305108', nome : "Fazenda Souza"},  
7332: {id : '5221', geocod : '4308078', nome : "Fazenda Vilanova"},  
7333: {id : '5255', geocod : '4308102', nome : "Feliz"},  
7334: {id : '6821', geocod : '4303004', nome : "Ferreira"},  
7335: {id : '5387', geocod : '4308201', nome : "Flores da Cunha"},  
7336: {id : '6836', geocod : '4310207', nome : "Floresta - Ijuí"},  
7337: {id : '6837', geocod : '4321907', nome : "Floresta - Três Passos"},  
7338: {id : '4710', geocod : '4308250', nome : "Floriano Peixoto"},  
7339: {id : '6844', geocod : '4317400', nome : "Florida"},  
7340: {id : '5326', geocod : '4308300', nome : "Fontoura Xavier"},  
7341: {id : '5142', geocod : '4308409', nome : "Formigueiro"},  
7342: {id : '6849', geocod : '4302808', nome : "Forninho"},  
7343: {id : '6850', geocod : '4301008', nome : "Forqueta"},  
7344: {id : '5534', geocod : '4308433', nome : "Forquetinha"},  
7345: {id : '6851', geocod : '4312401', nome : "Fortaleza - Montenegro"},  
7346: {id : '6852', geocod : '4320206', nome : "Fortaleza - Seberi"},  
7347: {id : '5302', geocod : '4308458', nome : "Fortaleza dos Valos"},  
7348: {id : '6854', geocod : '4301651', nome : "Francesa Alta"},  
7349: {id : '6855', geocod : '4301651', nome : "Francesa Baixa"},  
7350: {id : '4504', geocod : '4308508', nome : "Frederico Westphalen"},  
7351: {id : '6863', geocod : '4313656', nome : "Frei Sebastião"},  
7352: {id : '5361', geocod : '4308607', nome : "Garibaldi"},  
7353: {id : '6895', geocod : '4308607', nome : "Garibaldina"},  
7354: {id : '5078', geocod : '4308656', nome : "Garruchos"},  
7355: {id : '4718', geocod : '4308706', nome : "Gaurama"},  
7356: {id : '6901', geocod : '4312757', nome : "General Cadorna"},  
7357: {id : '5214', geocod : '4308805', nome : "General Câmara"},  
7358: {id : '6902', geocod : '4301651', nome : "General Neto"},  
7359: {id : '5348', geocod : '4308854', nome : "Gentil"},  
7360: {id : '4709', geocod : '4308904', nome : "Getúlio Vargas"},  
7361: {id : '5127', geocod : '4309001', nome : "Giruá"},  
7362: {id : '6914', geocod : '4307807', nome : "Glória - Estrela"},  
7363: {id : '6915', geocod : '4321501', nome : "Glória - Torres"},  
7364: {id : '5275', geocod : '4309050', nome : "Glorinha"},  
7365: {id : '6921', geocod : '4307203', nome : "Goio án"},  
7366: {id : '5272', geocod : '4309100', nome : "Gramado"},  
7367: {id : '4580', geocod : '4309126', nome : "Gramado dos Loureiros"},  
7368: {id : '6931', geocod : '4308300', nome : "Gramado São Pedro"},  
7369: {id : '5307', geocod : '4309159', nome : "Gramado Xavier"},  
7370: {id : '5243', geocod : '4309209', nome : "Gravataí"},  
7371: {id : '5376', geocod : '4309258', nome : "Guabiju"},  
7372: {id : '5238', geocod : '4309308', nome : "Guaíba"},  
7373: {id : '5357', geocod : '4309407', nome : "Guaporé"},  
7374: {id : '5110', geocod : '4309506', nome : "Guarani das Missões"},  
7375: {id : '6982', geocod : '4319406', nome : "Guassupi"},  
7376: {id : '5231', geocod : '4309555', nome : "Harmonia"},  
7377: {id : '7000', geocod : '4318804', nome : "Harmonia - São Lourenço do Sul"},  
7378: {id : '5463', geocod : '4307104', nome : "Herval"},  
7379: {id : '5176', geocod : '4309571', nome : "Herveiras"},  
7380: {id : '7010', geocod : '4304663', nome : "Hidráulica"},  
7381: {id : '4471', geocod : '4309605', nome : "Horizontina"},  
7382: {id : '5136', geocod : '4309654', nome : "Hulha Negra"},  
7383: {id : '4553', geocod : '4309704', nome : "Humaitá"},  
7384: {id : '5166', geocod : '4309753', nome : "Ibarama"},  
7385: {id : '7019', geocod : '4311502', nome : "Ibaré"},  
7386: {id : '5160', geocod : '4309803', nome : "Ibiaçá"},  
7387: {id : '7031', geocod : '4317103', nome : "Ibicuí"},  
7388: {id : '5190', geocod : '4309902', nome : "Ibiraiaras"},  
7389: {id : '5318', geocod : '4309951', nome : "Ibirapuitã"},  
7390: {id : '5305', geocod : '4310009', nome : "Ibirubá"},  
7391: {id : '5278', geocod : '4310108', nome : "Igrejinha"},  
7392: {id : '7078', geocod : '4305850', nome : "Igrejinha - Coqueiros do Sul"},  
7393: {id : '7087', geocod : '4311106', nome : "Ijucapirama"},  
7394: {id : '5288', geocod : '4310207', nome : "Ijuí"},  
7395: {id : '7090', geocod : '4315602', nome : "Ilha dos Marinheiros"},  
7396: {id : '5335', geocod : '4310306', nome : "Ilópolis"},  
7397: {id : '5404', geocod : '4310330', nome : "Imbé"},  
7398: {id : '5351', geocod : '4310363', nome : "Imigrante"},  
7399: {id : '4465', geocod : '4310405', nome : "Independência"},  
7400: {id : '4466', geocod : '4310413', nome : "Inhacorá"},  
7401: {id : '5389', geocod : '4310439', nome : "Ipê"},  
7402: {id : '7124', geocod : '4309209', nome : "Ipiranga"},  
7403: {id : '4705', geocod : '4310462', nome : "Ipiranga do Sul"},  
7404: {id : '7130', geocod : '4303400', nome : "Ipuaçu"},  
7405: {id : '4625', geocod : '4310504', nome : "Iraí"},  
7406: {id : '7150', geocod : '4315701', nome : "Iruí"},  
7407: {id : '5145', geocod : '4310538', nome : "Itaara"},  
7408: {id : '7161', geocod : '4309209', nome : "Itacolomi"},  
7409: {id : '5095', geocod : '4310553', nome : "Itacurubi"},  
7410: {id : '7230', geocod : '4310207', nome : "Itaí"},  
7411: {id : '7231', geocod : '4316709', nome : "Itaíba"},  
7412: {id : '7170', geocod : '4302303', nome : "Itaimbezinho"},  
7413: {id : '7232', geocod : '4310603', nome : "Itaó"},  
7414: {id : '7217', geocod : '4323002', nome : "Itapuã"},  
7415: {id : '5338', geocod : '4310579', nome : "Itapuca"},  
7416: {id : '7215', geocod : '4300703', nome : "Itapuca - Anta Gorda"},  
7417: {id : '5092', geocod : '4310603', nome : "Itaqui"},  
7418: {id : '5540', geocod : '4310652', nome : "Itati"},  
7419: {id : '4590', geocod : '4310702', nome : "Itatiba do Sul"},  
7420: {id : '7242', geocod : '4302501', nome : "Ivaí"},  
7421: {id : '5147', geocod : '4310751', nome : "Ivorá"},  
7422: {id : '5260', geocod : '4310801', nome : "Ivoti"},  
7423: {id : '4500', geocod : '4310850', nome : "Jaboticaba"},  
7424: {id : '5542', geocod : '4310876', nome : "Jacuizinho"},  
7425: {id : '4584', geocod : '4310900', nome : "Jacutinga"},  
7426: {id : '5467', geocod : '4311007', nome : "Jaguarão"},  
7427: {id : '7280', geocod : '4307104', nome : "Jaguarão Chico"},  
7428: {id : '7276', geocod : '4307005', nome : "Jaguaretê"},  
7429: {id : '5091', geocod : '4311106', nome : "Jaguarí"},  
7430: {id : '7288', geocod : '4307906', nome : "Jansen"},  
7431: {id : '5394', geocod : '4311122', nome : "Jaquirana"},  
7432: {id : '7296', geocod : '4318606', nome : "Jardim Alegre"},  
7433: {id : '5118', geocod : '4311130', nome : "Jari"},  
7434: {id : '7307', geocod : '4319604', nome : "Jazidas"},  
7435: {id : '7336', geocod : '4322400', nome : "João Arregui"},  
7436: {id : '7339', geocod : '4315701', nome : "João Rodrigues"},  
7437: {id : '7322', geocod : '4309654', nome : "Joca Tavares"},  
7438: {id : '5122', geocod : '4311155', nome : "Jóia"},  
7439: {id : '7331', geocod : '4301602', nome : "José Otávio"},  
7440: {id : '7377', geocod : '4318200', nome : "Juá"},  
7441: {id : '7381', geocod : '4316451', nome : "Júlio Borges"},  
7442: {id : '5286', geocod : '4311205', nome : "Júlio de Castilhos"},  
7443: {id : '7385', geocod : '4305959', nome : "Lageado Bonito"},  
7444: {id : '7392', geocod : '4311239', nome : "Lagoa Bonita"},  
7445: {id : '5539', geocod : '4311239', nome : "Lagoa Bonita do Sul"},  
7446: {id : '5313', geocod : '4311270', nome : "Lagoa dos Três Cantos"},  
7447: {id : '5162', geocod : '4311304', nome : "Lagoa Vermelha"},  
7448: {id : '5186', geocod : '4311254', nome : "Lagoão"},  
7449: {id : '5209', geocod : '4311403', nome : "Lajeado"},  
7450: {id : '7425', geocod : '4321477', nome : "Lajeado Bonito"},  
7451: {id : '7426', geocod : '4317509', nome : "Lajeado Cerne"},  
7452: {id : '4506', geocod : '4311429', nome : "Lajeado do Bugre"},  
7453: {id : '7427', geocod : '4306007', nome : "Lajeado Grande - Crissiumal"},  
7454: {id : '7428', geocod : '4318200', nome : "Lajeado Grande - São Francis"},  
7455: {id : '7429', geocod : '4317509', nome : "Lajeado Micuim"},  
7456: {id : '7440', geocod : '4323101', nome : "Laranjeira"},  
7457: {id : '5138', geocod : '4311502', nome : "Lavras do Sul"},  
7458: {id : '4512', geocod : '4311601', nome : "Liberato Salzano"},  
7459: {id : '5258', geocod : '4311627', nome : "Lindolfo Collor"},  
7460: {id : '7457', geocod : '4300554', nome : "Linha Bonita"},  
7461: {id : '7458', geocod : '4304101', nome : "Linha Ferrari"},  
7462: {id : '7459', geocod : '4311270', nome : "Linha Glória"},  
7463: {id : '5262', geocod : '4311643', nome : "Linha Nova"},  
7464: {id : '7460', geocod : '4308300', nome : "Linha Silveira"},  
7465: {id : '7461', geocod : '4300471', nome : "Linha Vitória"},  
7466: {id : '7470', geocod : '4319802', nome : "Loreto"},  
7467: {id : '5076', geocod : '4311718', nome : "Maçambara"},  
7468: {id : '4527', geocod : '4311700', nome : "Machadinho"},  
7469: {id : '5416', geocod : '4311734', nome : "Mampituba"},  
7470: {id : '7514', geocod : '4321808', nome : "Manchinha"},  
7471: {id : '7522', geocod : '4316402', nome : "Mangueiras"},  
7472: {id : '5085', geocod : '4311759', nome : "Manoel Viana"},  
7473: {id : '5405', geocod : '4311775', nome : "Maquiné"},  
7474: {id : '5229', geocod : '4311791', nome : "Maratá"},  
7475: {id : '5340', geocod : '4311809', nome : "Marau"},  
7476: {id : '4519', geocod : '4311908', nome : "Marcelino Ramos"},  
7477: {id : '7558', geocod : '4308607', nome : "Marcorama"},  
7478: {id : '5216', geocod : '4311981', nome : "Mariana Pimentel"},  
7479: {id : '4518', geocod : '4312005', nome : "Mariano Moro"},  
7480: {id : '7564', geocod : '4322608', nome : "Mariante"},  
7481: {id : '7577', geocod : '4323804', nome : "Mariápolis"},  
7482: {id : '7578', geocod : '4318606', nome : "Marmeleiro"},  
7483: {id : '5329', geocod : '4312054', nome : "Marques de Souza"},  
7484: {id : '5114', geocod : '4312104', nome : "Mata"},  
7485: {id : '7600', geocod : '4314209', nome : "Matarazzo"},  
7486: {id : '5381', geocod : '4312138', nome : "Mato Castelhano"},  
7487: {id : '7604', geocod : '4309001', nome : "Mato Grande - Giruá"},  
7488: {id : '7605', geocod : '4319158', nome : "Mato Grande - São Miguel das"},  
7489: {id : '5204', geocod : '4312153', nome : "Mato Leitão"},  
7490: {id : '7606', geocod : '4308201', nome : "Mato Perso"},  
7491: {id : '5543', geocod : '4312179', nome : "Mato Queimado"},  
7492: {id : '7608', geocod : '4301305', nome : "Mauá - Arroio Grande"},  
7493: {id : '7609', geocod : '4310207', nome : "Mauá - Ijuí"},  
7494: {id : '4524', geocod : '4312203', nome : "Maximiliano de Almeida"},  
7495: {id : '7612', geocod : '4300208', nome : "Medianeira"},  
7496: {id : '5198', geocod : '4312252', nome : "Minas do Leão"},  
7497: {id : '7635', geocod : '4307104', nome : "Mingote"},  
7498: {id : '4565', geocod : '4312302', nome : "Miraguaí"},  
7499: {id : '7640', geocod : '4317608', nome : "Miraguaia"},  
7500: {id : '7654', geocod : '4320602', nome : "Mirim"},  
7501: {id : '5342', geocod : '4312351', nome : "Montauri"},  
7502: {id : '5395', geocod : '4312377', nome : "Monte Alegre dos Campos"},  
7503: {id : '7682', geocod : '4316808', nome : "Monte Alverne"},  
7504: {id : '7683', geocod : '4320230', nome : "Monte Belo"},  
7505: {id : '5362', geocod : '4312385', nome : "Monte Belo do Sul"},  
7506: {id : '7684', geocod : '4314407', nome : "Monte Bonito"},  
7507: {id : '7689', geocod : '4313953', nome : "Monte Castelo"},  
7508: {id : '5226', geocod : '4312401', nome : "Montenegro"},  
7509: {id : '5312', geocod : '4312427', nome : "Mormaço"},  
7510: {id : '7715', geocod : '4318408', nome : "Morrinhos"},  
7511: {id : '5415', geocod : '4312443', nome : "Morrinhos do Sul"},  
7512: {id : '7724', geocod : '4321667', nome : "Morro de Dentro"},  
7513: {id : '7729', geocod : '4312443', nome : "Morro do Forno"},  
7514: {id : '5135', geocod : '4312450', nome : "Morro Redondo"},  
7515: {id : '5267', geocod : '4312476', nome : "Morro Reuter"},  
7516: {id : '7731', geocod : '4309209', nome : "Morungava"},  
7517: {id : '5236', geocod : '4312500', nome : "Mostardas"},  
7518: {id : '5352', geocod : '4312609', nome : "Muçum"},  
7519: {id : '5390', geocod : '4312617', nome : "Muitos Capões"},  
7520: {id : '5378', geocod : '4312625', nome : "Muliterno"},  
7521: {id : '7745', geocod : '4304101', nome : "Mundo Novo"},  
7522: {id : '5314', geocod : '4312658', nome : "Não-Me-Toque"},  
7523: {id : '5158', geocod : '4312674', nome : "Nicolau Vergueiro"},  
7524: {id : '4582', geocod : '4312708', nome : "Nonoai"},  
7525: {id : '7791', geocod : '4309902', nome : "Nossa Senhora da Saúde"},  
7526: {id : '7793', geocod : '4311304', nome : "Nossa Senhora de Fátima"},  
7527: {id : '5341', geocod : '4312757', nome : "Nova Alvorada"},  
7528: {id : '5371', geocod : '4312807', nome : "Nova Araçá"},  
7529: {id : '5370', geocod : '4312906', nome : "Nova Bassano"},  
7530: {id : '4498', geocod : '4312955', nome : "Nova Boa Vista"},  
7531: {id : '5333', geocod : '4313003', nome : "Nova Bréscia"},  
7532: {id : '4476', geocod : '4313011', nome : "Nova Candelária"},  
7533: {id : '5089', geocod : '4313037', nome : "Nova Esperança do Sul"},  
7534: {id : '5271', geocod : '4313060', nome : "Nova Hartz"},  
7535: {id : '7834', geocod : '4307906', nome : "Nova Milano"},  
7536: {id : '5385', geocod : '4313086', nome : "Nova Pádua"},  
7537: {id : '5157', geocod : '4313102', nome : "Nova Palma"},  
7538: {id : '5268', geocod : '4313201', nome : "Nova Petrópolis"},  
7539: {id : '5187', geocod : '4313300', nome : "Nova Prata"},  
7540: {id : '5292', geocod : '4313334', nome : "Nova Ramada"},  
7541: {id : '5367', geocod : '4313359', nome : "Nova Roma do Sul"},  
7542: {id : '5244', geocod : '4313375', nome : "Nova Santa Rita"},  
7543: {id : '7840', geocod : '4307906', nome : "Nova Sardenha"},  
7544: {id : '7842', geocod : '4321600', nome : "Nova Tramandaí"},  
7545: {id : '4496', geocod : '4313490', nome : "Novo Barreiro"},  
7546: {id : '5170', geocod : '4313391', nome : "Novo Cabrais"},  
7547: {id : '5251', geocod : '4313409', nome : "Novo Hamburgo"},  
7548: {id : '7857', geocod : '4310876', nome : "Novo Horizonte"},  
7549: {id : '4470', geocod : '4313425', nome : "Novo Machado"},  
7550: {id : '7860', geocod : '4321477', nome : "Novo Planalto"},  
7551: {id : '4510', geocod : '4313441', nome : "Novo Tiradentes"},  
7552: {id : '5547', geocod : '4313466', nome : "Novo Xingu"},  
7553: {id : '7866', geocod : '4322350', nome : "Oeste"},  
7554: {id : '7877', geocod : '4305108', nome : "Oliva"},  
7555: {id : '7884', geocod : '4310876', nome : "Oralina"},  
7556: {id : '5402', geocod : '4313508', nome : "Osório"},  
7557: {id : '7889', geocod : '4308508', nome : "Osvaldo Cruz"},  
7558: {id : '7890', geocod : '4303608', nome : "Osvaldo Kroeff"},  
7559: {id : '7892', geocod : '4308201', nome : "Otávio Rocha"},  
7560: {id : '7905', geocod : '4303509', nome : "Pacheca"},  
7561: {id : '7910', geocod : '4321204', nome : "Padilha"},  
7562: {id : '7915', geocod : '4321907', nome : "Padre Gonzales"},  
7563: {id : '4525', geocod : '4313607', nome : "Paim Filho"},  
7564: {id : '7926', geocod : '4316907', nome : "Pains"},  
7565: {id : '7935', geocod : '4323408', nome : "Palanque"},  
7566: {id : '5273', geocod : '4313656', nome : "Palmares do Sul"},  
7567: {id : '7940', geocod : '4301008', nome : "Palmas - Arroio do Meio"},  
7568: {id : '7941', geocod : '4301602', nome : "Palmas - Bagé"},  
7569: {id : '4495', geocod : '4313706', nome : "Palmeira das Missões"},  
7570: {id : '4567', geocod : '4313805', nome : "Palmitinho"},  
7571: {id : '7954', geocod : '4317103', nome : "Pampeiro"},  
7572: {id : '5293', geocod : '4313904', nome : "Panambi"},  
7573: {id : '5196', geocod : '4313953', nome : "Pantano Grande"},  
7574: {id : '5369', geocod : '4314001', nome : "Paraí"},  
7575: {id : '5165', geocod : '4314027', nome : "Paraíso do Sul"},  
7576: {id : '5227', geocod : '4314035', nome : "Pareci Novo"},  
7577: {id : '5276', geocod : '4314050', nome : "Parobé"},  
7578: {id : '5168', geocod : '4314068', nome : "Passa Sete"},  
7579: {id : '8005', geocod : '4313508', nome : "Passinhos"},  
7580: {id : '8006', geocod : '4305009', nome : "Passo Burmann"},  
7581: {id : '8013', geocod : '4315701', nome : "Passo da Areia - Rio Pardo"},  
7582: {id : '8014', geocod : '4323002', nome : "Passo da Areia - Viamão"},  
7583: {id : '8015', geocod : '4304663', nome : "Passo das Pedras"},  
7584: {id : '8016', geocod : '4321501', nome : "Passo de Torres"},  
7585: {id : '8017', geocod : '4315701', nome : "Passo do Adão"},  
7586: {id : '8018', geocod : '4315958', nome : "Passo do Faxinal"},  
7587: {id : '8019', geocod : '4306056', nome : "Passo do Mendonça"},  
7588: {id : '8020', geocod : '4315958', nome : "Passo do Quaresma"},  
7589: {id : '8021', geocod : '4323002', nome : "Passo do Sabão"},  
7590: {id : '5201', geocod : '4314076', nome : "Passo do Sobrado"},  
7591: {id : '8022', geocod : '4316907', nome : "Passo do Verde"},  
7592: {id : '5379', geocod : '4314100', nome : "Passo Fundo"},  
7593: {id : '8009', geocod : '4300406', nome : "Passo Novo"},  
7594: {id : '8010', geocod : '4322004', nome : "Passo Raso"},  
7595: {id : '8011', geocod : '4316451', nome : "Passo Real"},  
7596: {id : '8012', geocod : '4304358', nome : "Passo Real de Candio"},  
7597: {id : '5525', geocod : '4314134', nome : "Paulo Bento"},  
7598: {id : '5224', geocod : '4314159', nome : "Paverama"},  
7599: {id : '5549', geocod : '4314175', nome : "Pedras Altas"},  
7600: {id : '8056', geocod : '4319000', nome : "Pedras Brancas"},  
7601: {id : '8058', geocod : '4301305', nome : "Pedreiras"},  
7602: {id : '8061', geocod : '4302600', nome : "Pedro Gárcia"},  
7603: {id : '5132', geocod : '4314209', nome : "Pedro Osório"},  
7604: {id : '8063', geocod : '4317806', nome : "Pedro Paiva"},  
7605: {id : '8066', geocod : '4321204', nome : "Pega Fogo"},  
7606: {id : '5290', geocod : '4314308', nome : "Pejuçara"},  
7607: {id : '5178', geocod : '4314407', nome : "Pelotas"},  
7608: {id : '8091', geocod : '4312401', nome : "Pesqueiro"},  
7609: {id : '5263', geocod : '4314423', nome : "Picada Café"},  
7610: {id : '4569', geocod : '4314456', nome : "Pinhal"},  
7611: {id : '8115', geocod : '4302402', nome : "Pinhal - Bom Retiro do Sul"},  
7612: {id : '8116', geocod : '4320800', nome : "Pinhal - Soledade"},  
7613: {id : '8117', geocod : '4313201', nome : "Pinhal Alto"},  
7614: {id : '5526', geocod : '4314464', nome : "Pinhal da Serra"},  
7615: {id : '5295', geocod : '4314472', nome : "Pinhal Grande"},  
7616: {id : '8120', geocod : '4307203', nome : "Pinhalzinho - Erval Grande"},  
7617: {id : '8121', geocod : '4311601', nome : "Pinhalzinho - Liberato Salza"},  
7618: {id : '4593', geocod : '4314498', nome : "Pinheirinho do Vale"},  
7619: {id : '8125', geocod : '4317608', nome : "Pinheirinhos"},  
7620: {id : '5131', geocod : '4314506', nome : "Pinheiro Machado"},  
7621: {id : '8128', geocod : '4319307', nome : "Pinheiro Machado - São Paulo"},  
7622: {id : '8129', geocod : '4304705', nome : "Pinheiro Marcado"},  
7623: {id : '8137', geocod : '4302105', nome : "Pinto Bandeira"},  
7624: {id : '8158', geocod : '4306601', nome : "Piraí"},  
7625: {id : '8148', geocod : '4311759', nome : "Pirajú"},  
7626: {id : '5103', geocod : '4314555', nome : "Pirapó"},  
7627: {id : '5148', geocod : '4314605', nome : "Piratini"},  
7628: {id : '8166', geocod : '4306734', nome : "Pitanga"},  
7629: {id : '4572', geocod : '4314704', nome : "Planalto"},  
7630: {id : '8177', geocod : '4306007', nome : "Planalto - Crissiumal"},  
7631: {id : '8180', geocod : '4322400', nome : "Plano Alto"},  
7632: {id : '5225', geocod : '4314753', nome : "Poço das Antas"},  
7633: {id : '8256', geocod : '4314050', nome : "Poço Fundo"},  
7634: {id : '8187', geocod : '4321469', nome : "Poligôno do Erval"},  
7635: {id : '5322', geocod : '4314779', nome : "Pontão"},  
7636: {id : '4585', geocod : '4314787', nome : "Ponte Preta"},  
7637: {id : '5249', geocod : '4314803', nome : "Portão"},  
7638: {id : '5492', geocod : '4314902', nome : "Porto Alegre"},  
7639: {id : '8217', geocod : '4322004', nome : "Porto Batista"},  
7640: {id : '4456', geocod : '4315008', nome : "Porto Lucena"},  
7641: {id : '4462', geocod : '4315057', nome : "Porto Mauá"},  
7642: {id : '8227', geocod : '4321477', nome : "Porto Soberbo"},  
7643: {id : '4460', geocod : '4315073', nome : "Porto Vera Cruz"},  
7644: {id : '4455', geocod : '4315107', nome : "Porto Xavier"},  
7645: {id : '5325', geocod : '4315131', nome : "Pouso Novo"},  
7646: {id : '8245', geocod : '4315602', nome : "Povo Novo"},  
7647: {id : '8246', geocod : '4310702', nome : "Povoado Tozzo"},  
7648: {id : '8264', geocod : '4318804', nome : "Prado Novo"},  
7649: {id : '8272', geocod : '4306734', nome : "Pranchada"},  
7650: {id : '8276', geocod : '4313425', nome : "Pratos"},  
7651: {id : '5261', geocod : '4315149', nome : "Presidente Lucena"},  
7652: {id : '5324', geocod : '4315156', nome : "Progresso"},  
7653: {id : '8287', geocod : '4301206', nome : "Progresso - Arroio do Tigre"},  
7654: {id : '8288', geocod : '4304655', nome : "Progresso - Capão do Cipó"},  
7655: {id : '8289', geocod : '4321808', nome : "Progresso - Três de Maio"},  
7656: {id : '5189', geocod : '4315172', nome : "Protásio Alves"},  
7657: {id : '8296', geocod : '4307054', nome : "Pulador - Ernestina"},  
7658: {id : '8297', geocod : '4320404', nome : "Pulador - Serafina Corrêa"},  
7659: {id : '5334', geocod : '4315206', nome : "Putinga"},  
7660: {id : '5081', geocod : '4315305', nome : "Quaraí"},  
7661: {id : '8306', geocod : '4321808', nome : "Quaraim"},  
7662: {id : '5527', geocod : '4315313', nome : "Quatro Irmãos"},  
7663: {id : '8313', geocod : '4305116', nome : "Quebrado"},  
7664: {id : '5119', geocod : '4315321', nome : "Quevedos"},  
7665: {id : '8319', geocod : '4314407', nome : "Quilômbo"},  
7666: {id : '8324', geocod : '4315602', nome : "Quinta"},  
7667: {id : '8326', geocod : '4313656', nome : "Quintão"},  
7668: {id : '5304', geocod : '4315354', nome : "Quinze de Novembro"},  
7669: {id : '8327', geocod : '4309001', nome : "Quinze de Novembro - Giruá"},  
7670: {id : '8331', geocod : '4318408', nome : "Quitéria"},  
7671: {id : '4562', geocod : '4315404', nome : "Redentora"},  
7672: {id : '8346', geocod : '4322509', nome : "Refugiado"},  
7673: {id : '5331', geocod : '4315453', nome : "Relvado"},  
7674: {id : '8355', geocod : '4317509', nome : "Ressaca Buriti"},  
7675: {id : '5152', geocod : '4315503', nome : "Restinga Seca"},  
7676: {id : '8356', geocod : '4317509', nome : "Restinga Seca - Santo Ângelo"},  
7677: {id : '8390', geocod : '4314407', nome : "Rincão da Cruz"},  
7678: {id : '8391', geocod : '4318903', nome : "Rincão de São Pedro"},  
7679: {id : '8392', geocod : '4315701', nome : "Rincão del Rei"},  
7680: {id : '8393', geocod : '4312500', nome : "Rincão do Cristovão"},  
7681: {id : '8394', geocod : '4316451', nome : "Rincão do Ivaí"},  
7682: {id : '8395', geocod : '4318002', nome : "Rincão do Meio"},  
7683: {id : '8396', geocod : '4300471', nome : "Rincão do Segredo"},  
7684: {id : '8388', geocod : '4317756', nome : "Rincão Doce"},  
7685: {id : '8397', geocod : '4310876', nome : "Rincão dos Costas"},  
7686: {id : '8398', geocod : '4318200', nome : "Rincão dos Kroeff"},  
7687: {id : '8399', geocod : '4317509', nome : "Rincão dos Mendes"},  
7688: {id : '8400', geocod : '4319158', nome : "Rincão dos Moraes"},  
7689: {id : '8401', geocod : '4320578', nome : "Rincão dos Roratos"},  
7690: {id : '8402', geocod : '4304101', nome : "Rincão dos Toledos"},  
7691: {id : '8389', geocod : '4316303', nome : "Rincão Vermelho"},  
7692: {id : '8404', geocod : '4300901', nome : "Rio Azul"},  
7693: {id : '8407', geocod : '4313300', nome : "Rio Branco"},  
7694: {id : '8428', geocod : '4321204', nome : "Rio da Ilha"},  
7695: {id : '4581', geocod : '4315552', nome : "Rio dos Índios"},  
7696: {id : '5468', geocod : '4315602', nome : "Rio Grande"},  
7697: {id : '8415', geocod : '4316808', nome : "Rio Pardinho"},  
7698: {id : '5197', geocod : '4315701', nome : "Rio Pardo"},  
7699: {id : '8420', geocod : '4309803', nome : "Rio Telha"},  
7700: {id : '8421', geocod : '4316600', nome : "Rio Tigre"},  
7701: {id : '8422', geocod : '4308904', nome : "Rio Toldo"},  
7702: {id : '5283', geocod : '4315750', nome : "Riozinho"},  
7703: {id : '5349', geocod : '4315800', nome : "Roca Sales"},  
7704: {id : '4570', geocod : '4315909', nome : "Rodeio Bonito"},  
7705: {id : '5544', geocod : '4315958', nome : "Rolador"},  
7706: {id : '5281', geocod : '4316006', nome : "Rolante"},  
7707: {id : '8448', geocod : '4316006', nome : "Rolantinho"},  
7708: {id : '4577', geocod : '4316105', nome : "Ronda Alta"},  
7709: {id : '4574', geocod : '4316204', nome : "Rondinha"},  
7710: {id : '5104', geocod : '4316303', nome : "Roque Gonzales"},  
7711: {id : '8457', geocod : '4301503', nome : "Rosário"},  
7712: {id : '5087', geocod : '4316402', nome : "Rosário do Sul"},  
7713: {id : '4508', geocod : '4316428', nome : "Sagrada Família"},  
7714: {id : '8472', geocod : '4302907', nome : "Saicã"},  
7715: {id : '5184', geocod : '4316436', nome : "Saldanha Marinho"},  
7716: {id : '8479', geocod : '4310702', nome : "Saltinho"},  
7717: {id : '8480', geocod : '4315909', nome : "Saltinho - Rodeio Bonito"},  
7718: {id : '8482', geocod : '4302584', nome : "Salto"},  
7719: {id : '5300', geocod : '4316451', nome : "Salto do Jacuí"},  
7720: {id : '5106', geocod : '4316477', nome : "Salvador das Missões"},  
7721: {id : '5232', geocod : '4316501', nome : "Salvador do Sul"},  
7722: {id : '4520', geocod : '4316600', nome : "Sananduva"},  
7723: {id : '8498', geocod : '4302808', nome : "Santa Bárbara - Caçapava do"},  
7724: {id : '8499', geocod : '4319711', nome : "Santa Bárbara - São Valentin"},  
7725: {id : '5303', geocod : '4316709', nome : "Santa Bárbara do Sul"},  
7726: {id : '8500', geocod : '4316733', nome : "Santa Cecília"},  
7727: {id : '5528', geocod : '4316733', nome : "Santa Cecília do Sul"},  
7728: {id : '8502', geocod : '4315354', nome : "Santa Clara do Ingaí"},  
7729: {id : '5205', geocod : '4316758', nome : "Santa Clara do Sul"},  
7730: {id : '8503', geocod : '4314050', nome : "Santa Cristina"},  
7731: {id : '8507', geocod : '4321204', nome : "Santa Cruz da Concórdia"},  
7732: {id : '5174', geocod : '4316808', nome : "Santa Cruz do Sul"},  
7733: {id : '8519', geocod : '4323408', nome : "Santa Emília"},  
7734: {id : '8525', geocod : '4316907', nome : "Santa Flora"},  
7735: {id : '8530', geocod : '4318051', nome : "Santa Gema"},  
7736: {id : '8534', geocod : '4318903', nome : "Santa Inês"},  
7737: {id : '8538', geocod : '4301305', nome : "Santa Izabel do Sul"},  
7738: {id : '8553', geocod : '4300554', nome : "Santa Lúcia - Alto Alegre"},  
7739: {id : '8554', geocod : '4302584', nome : "Santa Lúcia - Bozano"},  
7740: {id : '8555', geocod : '4305108', nome : "Santa Lúcia do Piaí"},  
7741: {id : '8540', geocod : '4304804', nome : "Santa Luiza - Carlos Barbosa"},  
7742: {id : '8541', geocod : '4311304', nome : "Santa Luzia - Lagoa Vermelha"},  
7743: {id : '8542', geocod : '4313508', nome : "Santa Luzia - Osório"},  
7744: {id : '5535', geocod : '4316972', nome : "Santa Margarida do Sul"},  
7745: {id : '5143', geocod : '4316907', nome : "Santa Maria"},  
7746: {id : '5269', geocod : '4316956', nome : "Santa Maria do Herval"},  
7747: {id : '8580', geocod : '4300851', nome : "Santa Rita do Sul"},  
7748: {id : '4464', geocod : '4317202', nome : "Santa Rosa"},  
7749: {id : '8585', geocod : '4314407', nome : "Santa Silvana"},  
7750: {id : '8587', geocod : '4313706', nome : "Santa Teresinha"},  
7751: {id : '5354', geocod : '4317251', nome : "Santa Tereza"},  
7752: {id : '8590', geocod : '4305009', nome : "Santa Tereza - Catuípe"},  
7753: {id : '8596', geocod : '4317558', nome : "Santa Terezinha - Santo Antô"},  
7754: {id : '8597', geocod : '4320800', nome : "Santa Terezinha - Soledade"},  
7755: {id : '5466', geocod : '4317301', nome : "Santa Vitória do Palmar"},  
7756: {id : '8603', geocod : '4305306', nome : "Santana - Chapada"},  
7757: {id : '8604', geocod : '4307203', nome : "Santana - Erval Grande"},  
7758: {id : '8605', geocod : '4310207', nome : "Santana - Ijuí"},  
7759: {id : '5149', geocod : '4317004', nome : "Santana da Boa Vista"},  
7760: {id : '5083', geocod : '4317103', nome : "Santana do Livramento"},  
7761: {id : '8494', geocod : '4303509', nome : "SantAuta"},  
7762: {id : '5097', geocod : '4317400', nome : "Santiago"},  
7763: {id : '8625', geocod : '4308805', nome : "Santo Amaro do Sul"},  
7764: {id : '5126', geocod : '4317509', nome : "Santo Ângelo"},  
7765: {id : '8630', geocod : '4305207', nome : "Santo Antônio - Cerro Largo"},  
7766: {id : '8631', geocod : '4309407', nome : "Santo Antônio - Guaporé"},  
7767: {id : '8632', geocod : '4310207', nome : "Santo Antônio - Ijuí"},  
7768: {id : '8633', geocod : '4317806', nome : "Santo Antônio - Santo August"},  
7769: {id : '8634', geocod : '4321907', nome : "Santo Antônio - Três Passos"},  
7770: {id : '5280', geocod : '4317608', nome : "Santo Antônio da Patrulha"},  
7771: {id : '5096', geocod : '4317707', nome : "Santo Antônio das Missões"},  
7772: {id : '8675', geocod : '4304804', nome : "Santo Antônio de Castro"},  
7773: {id : '8650', geocod : '4310009', nome : "Santo Antônio do Bom"},  
7774: {id : '5347', geocod : '4317558', nome : "Santo Antônio do Palma"},  
7775: {id : '5316', geocod : '4317756', nome : "Santo Antônio do Planalto"},  
7776: {id : '8674', geocod : '4300059', nome : "Santo Antônio dos Pinhais"},  
7777: {id : '4550', geocod : '4317806', nome : "Santo Augusto"},  
7778: {id : '4459', geocod : '4317905', nome : "Santo Cristo"},  
7779: {id : '4522', geocod : '4317954', nome : "Santo Expedito do Sul"},  
7780: {id : '8679', geocod : '4302303', nome : "Santo Inácio"},  
7781: {id : '8680', geocod : '4311106', nome : "Santo Isidro - Jaguari"},  
7782: {id : '8681', geocod : '4319208', nome : "Santo Isidro - São Nicolau"},  
7783: {id : '8684', geocod : '4312401', nome : "Santo Reis"},  
7784: {id : '8818', geocod : '4304705', nome : "São Bento"},  
7785: {id : '8824', geocod : '4307302', nome : "São Bom Jesus"},  
7786: {id : '5077', geocod : '4318002', nome : "São Borja"},  
7787: {id : '8829', geocod : '4311981', nome : "São Carlos"},  
7788: {id : '8834', geocod : '4314175', nome : "São Diogo - Pedras Altas"},  
7789: {id : '8835', geocod : '4317103', nome : "São Diogo - Sant' Ana do Liv"},  
7790: {id : '5372', geocod : '4318051', nome : "São Domingos do Sul"},  
7791: {id : '8845', geocod : '4305207', nome : "São Francisco"},  
7792: {id : '5086', geocod : '4318101', nome : "São Francisco de Assis"},  
7793: {id : '5284', geocod : '4318200', nome : "São Francisco de Paula"},  
7794: {id : '5113', geocod : '4318309', nome : "São Gabriel"},  
7795: {id : '5213', geocod : '4318408', nome : "São Jerônimo"},  
7796: {id : '8941', geocod : '4302154', nome : "São João - Boa Vista das Mis"},  
7797: {id : '8942', geocod : '4315404', nome : "São João - Redentora"},  
7798: {id : '8944', geocod : '4305504', nome : "São João Bosco"},  
7799: {id : '4521', geocod : '4318424', nome : "São João da Urtiga"},  
7800: {id : '8954', geocod : '4319158', nome : "São João das Missões"},  
7801: {id : '5155', geocod : '4318432', nome : "São João do Polêsine"},  
7802: {id : '5377', geocod : '4318440', nome : "São Jorge"},  
7803: {id : '8888', geocod : '4300554', nome : "São José - Alto Alegre"},  
7804: {id : '8890', geocod : '4317707', nome : "São José - Santo Antônio das"},  
7805: {id : '8889', geocod : '4317558', nome : "São José - Santo Antônio do"},  
7806: {id : '8891', geocod : '4319158', nome : "São José - São Miguel das Mi"},  
7807: {id : '8899', geocod : '4323200', nome : "São José da Glória"},  
7808: {id : '4507', geocod : '4318457', nome : "São José das Missões"},  
7809: {id : '8906', geocod : '4308607', nome : "São José de Costa Real"},  
7810: {id : '8915', geocod : '4312658', nome : "São José do Centro"},  
7811: {id : '5327', geocod : '4318465', nome : "São José do Herval"},  
7812: {id : '5259', geocod : '4318481', nome : "São José do Hortêncio"},  
7813: {id : '4474', geocod : '4318499', nome : "São José do Inhacorá"},  
7814: {id : '8924', geocod : '4318614', nome : "São José do Maratá"},  
7815: {id : '5469', geocod : '4318507', nome : "São José do Norte"},  
7816: {id : '4532', geocod : '4318606', nome : "São José do Ouro"},  
7817: {id : '5530', geocod : '4318614', nome : "São José do Sul"},  
7818: {id : '5418', geocod : '4318622', nome : "São José dos Ausentes"},  
7819: {id : '8937', geocod : '4304101', nome : "São José dos Campos"},  
7820: {id : '5248', geocod : '4318705', nome : "São Leopoldo"},  
7821: {id : '8975', geocod : '4318903', nome : "São Lourenço das Missões"},  
7822: {id : '5180', geocod : '4318804', nome : "São Lourenço do Sul"},  
7823: {id : '8986', geocod : '4303202', nome : "São Luís Rei"},  
7824: {id : '8983', geocod : '4309902', nome : "São Luiz"},  
7825: {id : '5099', geocod : '4318903', nome : "São Luiz Gonzaga"},  
7826: {id : '5388', geocod : '4319000', nome : "São Marcos"},  
7827: {id : '8991', geocod : '4322400', nome : "São Marcos - Uruguaiana"},  
7828: {id : '8992', geocod : '4305801', nome : "São Marcos Baixo"},  
7829: {id : '4551', geocod : '4319109', nome : "São Martinho"},  
7830: {id : '8994', geocod : '4316808', nome : "São Martinho - Santa Cruz do"},  
7831: {id : '5144', geocod : '4319125', nome : "São Martinho da Serra"},  
7832: {id : '9002', geocod : '4305306', nome : "São Miguel - Chapada"},  
7833: {id : '9003', geocod : '4318606', nome : "São Miguel - São José do Our"},  
7834: {id : '5100', geocod : '4319158', nome : "São Miguel das Missões"},  
7835: {id : '5101', geocod : '4319208', nome : "São Nicolau"},  
7836: {id : '5107', geocod : '4319307', nome : "São Paulo das Missões"},  
7837: {id : '9010', geocod : '4309001', nome : "São Paulo das Tunas"},  
7838: {id : '9015', geocod : '4302105', nome : "São Pedro - Bento Gonçalves"},  
7839: {id : '9016', geocod : '4319364', nome : "São Pedro - São Pedro das Mi"},  
7840: {id : '9017', geocod : '4321006', nome : "São Pedro - Tapera"},  
7841: {id : '9018', geocod : '4321402', nome : "São Pedro - Tenete Portela"},  
7842: {id : '5230', geocod : '4319356', nome : "São Pedro da Serra"},  
7843: {id : '5546', geocod : '4319364', nome : "São Pedro das Missões"},  
7844: {id : '5105', geocod : '4319372', nome : "São Pedro do Butiá"},  
7845: {id : '9034', geocod : '4318606', nome : "São Pedro do Iraxim"},  
7846: {id : '5116', geocod : '4319406', nome : "São Pedro do Sul"},  
7847: {id : '9050', geocod : '4314100', nome : "São Roque"},  
7848: {id : '9057', geocod : '4307401', nome : "São Sebastião - Esmeralda"},  
7849: {id : '9058', geocod : '4309902', nome : "São Sebastião - Ibiraiaras"},  
7850: {id : '5252', geocod : '4319505', nome : "São Sebastião do Caí"},  
7851: {id : '5141', geocod : '4319604', nome : "São Sepé"},  
7852: {id : '9093', geocod : '4312500', nome : "São Simão"},  
7853: {id : '4586', geocod : '4319703', nome : "São Valentim"},  
7854: {id : '9099', geocod : '4310405', nome : "São Valentim - Independência"},  
7855: {id : '5358', geocod : '4319711', nome : "São Valentim do Sul"},  
7856: {id : '4549', geocod : '4319737', nome : "São Valério do Sul"},  
7857: {id : '5254', geocod : '4319752', nome : "São Vendelino"},  
7858: {id : '5090', geocod : '4319802', nome : "São Vicente do Sul"},  
7859: {id : '5265', geocod : '4319901', nome : "Sapiranga"},  
7860: {id : '5247', geocod : '4320008', nome : "Sapucaia do Sul"},  
7861: {id : '8699', geocod : '4322608', nome : "Saraiva"},  
7862: {id : '4573', geocod : '4320107', nome : "Sarandi"},  
7863: {id : '4502', geocod : '4320206', nome : "Seberi"},  
7864: {id : '8708', geocod : '4305108', nome : "Seca"},  
7865: {id : '8711', geocod : '4315354', nome : "Sede Aurora"},  
7866: {id : '8712', geocod : '4314100', nome : "Sede Independência"},  
7867: {id : '4552', geocod : '4320230', nome : "Sede Nova"},  
7868: {id : '5298', geocod : '4320263', nome : "Segredo"},  
7869: {id : '8715', geocod : '4304358', nome : "Seival"},  
7870: {id : '5310', geocod : '4320305', nome : "Selbach"},  
7871: {id : '5112', geocod : '4320321', nome : "Senador Salgado Filho"},  
7872: {id : '5193', geocod : '4320354', nome : "Sentinela do Sul"},  
7873: {id : '5344', geocod : '4320404', nome : "Serafina Corrêa"},  
7874: {id : '5203', geocod : '4320453', nome : "Sério"},  
7875: {id : '8744', geocod : '4314464', nome : "Serra dos Gregórios"},  
7876: {id : '8748', geocod : '4315958', nome : "Serrinha"},  
7877: {id : '8750', geocod : '4320263', nome : "Serrinha Velha"},  
7878: {id : '4707', geocod : '4320503', nome : "Sertão"},  
7879: {id : '5194', geocod : '4320552', nome : "Sertão Santana"},  
7880: {id : '8757', geocod : '4300505', nome : "Sertãozinho"},  
7881: {id : '5125', geocod : '4320578', nome : "Sete de Setembro"},  
7882: {id : '8762', geocod : '4307203', nome : "Sete de Setembro - Erval Gra"},  
7883: {id : '8763', geocod : '4317202', nome : "Sete de Setembro - Santa Ros"},  
7884: {id : '8760', geocod : '4310702', nome : "Sete Lagoas"},  
7885: {id : '4517', geocod : '4320602', nome : "Severiano de Almeida"},  
7886: {id : '8765', geocod : '4320404', nome : "Silva Jardim"},  
7887: {id : '5146', geocod : '4320651', nome : "Silveira Martins"},  
7888: {id : '5177', geocod : '4320677', nome : "Sinimbu"},  
7889: {id : '9110', geocod : '4317905', nome : "Sírio - Santo Cristo"},  
7890: {id : '9111', geocod : '4301206', nome : "Sítio - Arroio do Tigre"},  
7891: {id : '9113', geocod : '4315404', nome : "Sítio Cassemiro"},  
7892: {id : '9114', geocod : '4312302', nome : "Sítio Gabriel"},  
7893: {id : '5167', geocod : '4320701', nome : "Sobradinho"},  
7894: {id : '5317', geocod : '4320800', nome : "Soledade"},  
7895: {id : '8791', geocod : '4317509', nome : "Sossego"},  
7896: {id : '8793', geocod : '4314787', nome : "Souto Neto"},  
7897: {id : '8795', geocod : '4308904', nome : "Souza Ramos"},  
7898: {id : '8808', geocod : '4318309', nome : "Suspiro"},  
7899: {id : '5220', geocod : '4320859', nome : "Tabaí"},  
7900: {id : '9125', geocod : '4316451', nome : "Tabajara"},  
7901: {id : '9137', geocod : '4306908', nome : "Taim - Encruzilhada do Sul"},  
7902: {id : '9138', geocod : '4315602', nome : "Taim - Rio Grande"},  
7903: {id : '9139', geocod : '4318200', nome : "Tainhas"},  
7904: {id : '9142', geocod : '4320263', nome : "Tamanduá"},  
7905: {id : '9147', geocod : '4318606', nome : "Tanque"},  
7906: {id : '5384', geocod : '4320909', nome : "Tapejara"},  
7907: {id : '5311', geocod : '4321006', nome : "Tapera"},  
7908: {id : '9151', geocod : '4321352', nome : "Tapera - Tavares"},  
7909: {id : '5195', geocod : '4321105', nome : "Tapes"},  
7910: {id : '5277', geocod : '4321204', nome : "Taquara"},  
7911: {id : '9163', geocod : '4318804', nome : "Taquaral"},  
7912: {id : '5219', geocod : '4321303', nome : "Taquari"},  
7913: {id : '9167', geocod : '4319802', nome : "Taquarichim"},  
7914: {id : '4501', geocod : '4321329', nome : "Taquaruçu do Sul"},  
7915: {id : '5235', geocod : '4321352', nome : "Tavares"},  
7916: {id : '4560', geocod : '4321402', nome : "Tenente Portela"},  
7917: {id : '5409', geocod : '4321436', nome : "Terra de Areia"},  
7918: {id : '9193', geocod : '4305306', nome : "Tesouras"},  
7919: {id : '5223', geocod : '4321451', nome : "Teutônia"},  
7920: {id : '9195', geocod : '4318309', nome : "Tiaraju"},  
7921: {id : '9201', geocod : '4302501', nome : "Timbaúva"},  
7922: {id : '5529', geocod : '4321469', nome : "Tio Hugo"},  
7923: {id : '4478', geocod : '4321477', nome : "Tiradentes do Sul"},  
7924: {id : '5117', geocod : '4321493', nome : "Toropi"},  
7925: {id : '9222', geocod : '4318101', nome : "Toroquá"},  
7926: {id : '9223', geocod : '4306601', nome : "Torquato Severo"},  
7927: {id : '5420', geocod : '4321501', nome : "Torres"},  
7928: {id : '9225', geocod : '4314506', nome : "Torrinhas"},  
7929: {id : '9226', geocod : '4316402', nome : "Touro Passo"},  
7930: {id : '5403', geocod : '4321600', nome : "Tramandaí"},  
7931: {id : '5330', geocod : '4321626', nome : "Travesseiro"},  
7932: {id : '9233', geocod : '4310850', nome : "Trentin"},  
7933: {id : '4516', geocod : '4321634', nome : "Três Arroios"},  
7934: {id : '9245', geocod : '4300901', nome : "Três Barras"},  
7935: {id : '5411', geocod : '4321667', nome : "Três Cachoeiras"},  
7936: {id : '5279', geocod : '4321709', nome : "Três Coroas"},  
7937: {id : '4473', geocod : '4321808', nome : "Três de Maio"},  
7938: {id : '5406', geocod : '4321832', nome : "Três Forquilhas"},  
7939: {id : '4576', geocod : '4321857', nome : "Três Palmeiras"},  
7940: {id : '4557', geocod : '4321907', nome : "Três Passos"},  
7941: {id : '9251', geocod : '4308300', nome : "Três Pinheiros - Fontoura Xa"},  
7942: {id : '9252', geocod : '4310652', nome : "Três Pinheiros - Itati"},  
7943: {id : '9254', geocod : '4303004', nome : "Três Vendas"},  
7944: {id : '9234', geocod : '4300554', nome : "Treze de Maio"},  
7945: {id : '4579', geocod : '4321956', nome : "Trindade do Sul"},  
7946: {id : '5215', geocod : '4322004', nome : "Triunfo"},  
7947: {id : '9242', geocod : '4312302', nome : "Tronqueiras"},  
7948: {id : '4469', geocod : '4322103', nome : "Tucunduva"},  
7949: {id : '9259', geocod : '4302105', nome : "Tuiutí"},  
7950: {id : '5299', geocod : '4322152', nome : "Tunas"},  
7951: {id : '4523', geocod : '4322186', nome : "Tupanci do Sul"},  
7952: {id : '5285', geocod : '4322202', nome : "Tupanciretã"},  
7953: {id : '9262', geocod : '4319604', nome : "Tupancy e Vila Block"},  
7954: {id : '5234', geocod : '4322251', nome : "Tupandi"},  
7955: {id : '9263', geocod : '4317400', nome : "Tupantuba"},  
7956: {id : '4468', geocod : '4322301', nome : "Tuparendi"},  
7957: {id : '9268', geocod : '4311304', nome : "Tupinambá"},  
7958: {id : '5179', geocod : '4322327', nome : "Turuçu"},  
7959: {id : '9272', geocod : '4305900', nome : "Turvinho"},  
7960: {id : '5111', geocod : '4322343', nome : "Ubiretama"},  
7961: {id : '9286', geocod : '4302907', nome : "Umbu"},  
7962: {id : '9293', geocod : '4317509', nome : "União"},  
7963: {id : '5343', geocod : '4322350', nome : "União da Serra"},  
7964: {id : '5094', geocod : '4322376', nome : "Unistalda"},  
7965: {id : '9295', geocod : '4317103', nome : "Upamaroti"},  
7966: {id : '5080', geocod : '4322400', nome : "Uruguaiana"},  
7967: {id : '9310', geocod : '4318309', nome : "Vacacaí"},  
7968: {id : '5392', geocod : '4322509', nome : "Vacaria"},  
7969: {id : '9314', geocod : '4306809', nome : "Valdástico"},  
7970: {id : '9320', geocod : '4313201', nome : "Vale do Rio Caí"},  
7971: {id : '5175', geocod : '4322533', nome : "Vale do Sol"},  
7972: {id : '9322', geocod : '4302105', nome : "Vale dos Vinhedos"},  
7973: {id : '5257', geocod : '4322541', nome : "Vale Real"},  
7974: {id : '9317', geocod : '4318432', nome : "Vale Vêneto"},  
7975: {id : '5200', geocod : '4322525', nome : "Vale Verde"},  
7976: {id : '5373', geocod : '4322558', nome : "Vanini"},  
7977: {id : '9329', geocod : '4304101', nome : "Varame"},  
7978: {id : '9439', geocod : '4313953', nome : "Várzea"},  
7979: {id : '5202', geocod : '4322608', nome : "Venâncio Aires"},  
7980: {id : '9349', geocod : '4312401', nome : "Vendinha"},  
7981: {id : '5173', geocod : '4322707', nome : "Vera Cruz"},  
7982: {id : '5365', geocod : '4322806', nome : "Veranópolis"},  
7983: {id : '9359', geocod : '4322400', nome : "Vertentes"},  
7984: {id : '5355', geocod : '4322855', nome : "Vespasiano Correa"},  
7985: {id : '4515', geocod : '4322905', nome : "Viadutos"},  
7986: {id : '5240', geocod : '4323002', nome : "Viamão"},  
7987: {id : '9361', geocod : '4323002', nome : "Viamópolis"},  
7988: {id : '4623', geocod : '4323101', nome : "Vicente Dutra"},  
7989: {id : '5315', geocod : '4323200', nome : "Victor Graeff"},  
7990: {id : '9368', geocod : '4317905', nome : "Vila Bom Princípio B"},  
7991: {id : '9369', geocod : '4320909', nome : "Vila Campos"},  
7992: {id : '9372', geocod : '4305108', nome : "Vila Cristina"},  
7993: {id : '9373', geocod : '4313102', nome : "Vila Cruz"},  
7994: {id : '5366', geocod : '4323309', nome : "Vila Flores"},  
7995: {id : '5382', geocod : '4323358', nome : "Vila Lângaro"},  
7996: {id : '9377', geocod : '4317905', nome : "Vila Laranjeira"},  
7997: {id : '5345', geocod : '4323408', nome : "Vila Maria"},  
7998: {id : '5140', geocod : '4323457', nome : "Vila Nova do Sul"},  
7999: {id : '9397', geocod : '4305306', nome : "Vila Rica"},  
8000: {id : '9403', geocod : '4310439', nome : "Vila São Paulo"},  
8001: {id : '9400', geocod : '4310439', nome : "Vila Segredo"},  
8002: {id : '9402', geocod : '4318606', nome : "Vila Souza"},  
8003: {id : '9404', geocod : '4304002', nome : "Vila Turvo"},  
8004: {id : '4568', geocod : '4323507', nome : "Vista Alegre"},  
8005: {id : '9423', geocod : '4305603', nome : "Vista Alegre - Colorado"},  
8006: {id : '5368', geocod : '4323606', nome : "Vista Alegre do Prata"},  
8007: {id : '4566', geocod : '4323705', nome : "Vista Gaúcha"},  
8008: {id : '9425', geocod : '4307302', nome : "Vista Gaúcha - Erval Seco"},  
8009: {id : '9426', geocod : '4306007', nome : "Vista Nova"},  
8010: {id : '9430', geocod : '4309803', nome : "Vitória"},  
8011: {id : '5120', geocod : '4323754', nome : "Vitória das Missões"},  
8012: {id : '9433', geocod : '4307500', nome : "Volta Alegre"},  
8013: {id : '9434', geocod : '4300901', nome : "Volta Fechada"},  
8014: {id : '9436', geocod : '4300505', nome : "Volta Grande"},  
8015: {id : '9437', geocod : '4304101', nome : "Volta Vitória"},  
8016: {id : '5541', geocod : '4323770', nome : "Westfalia"},  
8017: {id : '9451', geocod : '4305850', nome : "Xadrez"},  
8018: {id : '5407', geocod : '4323804', nome : "Xangri-lá"},  
8019: {id : '9452', geocod : '4315206', nome : "Xarqueada"},  
8020: {id : '9453', geocod : '4315156', nome : "Xaxim"},  
8021: {id : '9455', geocod : '4313466', nome : "Xingu"}  
8022: ],  
8023: SC : [  
8024: // UF = SC  
8025: {id : '4723', geocod : '4200051', nome : "Abdon Batista"},  
8026: {id : '4746', geocod : '4200101', nome : "Abelardo Luz"},  
8027: {id : '4919', geocod : '4200200', nome : "Agrolândia"},  
8028: {id : '4927', geocod : '4200309', nome : "Agronômica"},  
8029: {id : '9466', geocod : '4215000', nome : "Água Clara"},  
8030: {id : '4760', geocod : '4200408', nome : "Água Doce"},  
8031: {id : '9475', geocod : '4218905', nome : "Águas Brancas"},  
8032: {id : '4633', geocod : '4200507', nome : "Águas de Chapecó"},  
8033: {id : '4649', geocod : '4200556', nome : "Águas Frias"},  
8034: {id : '4939', geocod : '4200606', nome : "Águas Mornas"},  
8035: {id : '5593', geocod : '4211504', nome : "Aguti"},  
8036: {id : '5596', geocod : '4206108', nome : "Aiurê"},  
8037: {id : '4930', geocod : '4200705', nome : "Alfredo Wagner"},  
8038: {id : '5630', geocod : '4203907', nome : "Alto Alegre"},  
8039: {id : '4528', geocod : '4200754', nome : "Alto Bela Vista"},  
8040: {id : '5654', geocod : '4204202', nome : "Alto da Serra"},  
8041: {id : '4643', geocod : '4200804', nome : "Anchieta"},  
8042: {id : '4938', geocod : '4200903', nome : "Angelina"},  
8043: {id : '4721', geocod : '4201000', nome : "Anita Garibaldi"},  
8044: {id : '4934', geocod : '4201109', nome : "Anitápolis"},  
8045: {id : '5701', geocod : '4219309', nome : "Anta Gorda"},  
8046: {id : '4944', geocod : '4201208', nome : "Antônio Carlos"},  
8047: {id : '4908', geocod : '4201257', nome : "Apiúna"},  
8048: {id : '4737', geocod : '4201273', nome : "Arabutã"},  
8049: {id : '5017', geocod : '4201307', nome : "Araquari"},  
8050: {id : '5429', geocod : '4201406', nome : "Araranguá"},  
8051: {id : '5457', geocod : '4201505', nome : "Armazém"},  
8052: {id : '5783', geocod : '4200705', nome : "Arnópolis"},  
8053: {id : '4781', geocod : '4201604', nome : "Arroio Trinta"},  
8054: {id : '4655', geocod : '4201653', nome : "Arvoredo"},  
8055: {id : '4909', geocod : '4201703', nome : "Ascurra"},  
8056: {id : '4925', geocod : '4201802', nome : "Atalanta"},  
8057: {id : '5806', geocod : '4213708', nome : "Aterrado Torto"},  
8058: {id : '4929', geocod : '4201901', nome : "Aurora"},  
8059: {id : '5818', geocod : '4212403', nome : "Azambuja"},  
8060: {id : '5960', geocod : '4213401', nome : "Baía Alta"},  
8061: {id : '5428', geocod : '4201950', nome : "Balneário Arroio do Silva"},  
8062: {id : '5019', geocod : '4202057', nome : "Balneário Barra do Sul"},  
8063: {id : '5011', geocod : '4202008', nome : "Balneário Camboriú"},  
8064: {id : '5424', geocod : '4202073', nome : "Balneário Gaivota"},  
8065: {id : '4597', geocod : '4202081', nome : "Bandeirante"},  
8066: {id : '4640', geocod : '4202099', nome : "Barra Bonita"},  
8067: {id : '5876', geocod : '4200903', nome : "Barra Clara"},  
8068: {id : '5890', geocod : '4219358', nome : "Barra da Prata"},  
8069: {id : '5879', geocod : '4205209', nome : "Barra Fria"},  
8070: {id : '5883', geocod : '4205308', nome : "Barra Grande"},  
8071: {id : '5016', geocod : '4202107', nome : "Barra Velha"},  
8072: {id : '5918', geocod : '4216602', nome : "Barreiros"},  
8073: {id : '5926', geocod : '4209607', nome : "Barro Branco"},  
8074: {id : '5949', geocod : '4203303', nome : "Bateias de Baixo"},  
8075: {id : '5969', geocod : '4203600', nome : "Bela Vista"},  
8076: {id : '5980', geocod : '4210100', nome : "Bela Vista do Sul"},  
8077: {id : '4798', geocod : '4202131', nome : "Bela Vista do Toldo"},  
8078: {id : '4599', geocod : '4202156', nome : "Belmonte"},  
8079: {id : '4961', geocod : '4202206', nome : "Benedito Novo"},  
8080: {id : '4949', geocod : '4202305', nome : "Biguaçu"},  
8081: {id : '4991', geocod : '4202404', nome : "Blumenau"},  
8082: {id : '4918', geocod : '4202438', nome : "Bocaina do Sul"},  
8083: {id : '6045', geocod : '4210209', nome : "Boiteuxburgo"},  
8084: {id : '5433', geocod : '4202503', nome : "Bom Jardim da Serra"},  
8085: {id : '4744', geocod : '4202537', nome : "Bom Jesus"},  
8086: {id : '4542', geocod : '4202578', nome : "Bom Jesus do Oeste"},  
8087: {id : '4922', geocod : '4202602', nome : "Bom Retiro"},  
8088: {id : '5009', geocod : '4202453', nome : "Bombinhas"},  
8089: {id : '4987', geocod : '4202701', nome : "Botuverá"},  
8090: {id : '5456', geocod : '4202800', nome : "Braço do Norte"},  
8091: {id : '4920', geocod : '4202859', nome : "Braço do Trombudo"},  
8092: {id : '4725', geocod : '4202875', nome : "Brunópolis"},  
8093: {id : '4993', geocod : '4202909', nome : "Brusque"},  
8094: {id : '4786', geocod : '4203006', nome : "Caçador"},  
8095: {id : '6162', geocod : '4213807', nome : "Cachoeira de Fátima"},  
8096: {id : '6165', geocod : '4205407', nome : "Cachoeira do Bom Jesus"},  
8097: {id : '4627', geocod : '4203105', nome : "Caibi"},  
8098: {id : '4789', geocod : '4203154', nome : "Calmon"},  
8099: {id : '5007', geocod : '4203204', nome : "Camboriú"},  
8100: {id : '6223', geocod : '4219507', nome : "Cambuinzal"},  
8101: {id : '6238', geocod : '4216602', nome : "Campinas"},  
8102: {id : '5020', geocod : '4203303', nome : "Campo Alegre"},  
8103: {id : '4726', geocod : '4203402', nome : "Campo Belo do Sul"},  
8104: {id : '4546', geocod : '4203501', nome : "Campo Erê"},  
8105: {id : '4722', geocod : '4203600', nome : "Campos Novos"},  
8106: {id : '6276', geocod : '4205407', nome : "Canasvieiras"},  
8107: {id : '4994', geocod : '4203709', nome : "Canelinha"},  
8108: {id : '6294', geocod : '4202602', nome : "Canoas"},  
8109: {id : '4807', geocod : '4203808', nome : "Canoinhas"},  
8110: {id : '4727', geocod : '4203253', nome : "Capão Alto"},  
8111: {id : '4536', geocod : '4203907', nome : "Capinzal"},  
8112: {id : '5452', geocod : '4203956', nome : "Capivari de Baixo"},  
8113: {id : '6343', geocod : '4217501', nome : "Caraíba"},  
8114: {id : '4754', geocod : '4204004', nome : "Catanduvas"},  
8115: {id : '6402', geocod : '4200705', nome : "Catuíra"},  
8116: {id : '4645', geocod : '4204103', nome : "Caxambu do Sul"},  
8117: {id : '6416', geocod : '4214706', nome : "Cedro Alto"},  
8118: {id : '4720', geocod : '4204152', nome : "Celso Ramos"},  
8119: {id : '4719', geocod : '4204178', nome : "Cerro Negro"},  
8120: {id : '4924', geocod : '4204194', nome : "Chapadão do Lageado"},  
8121: {id : '4653', geocod : '4204202', nome : "Chapecó"},  
8122: {id : '6461', geocod : '4211504', nome : "Claraíba"},  
8123: {id : '5437', geocod : '4204251', nome : "Cocal do Sul"},  
8124: {id : '4740', geocod : '4204301', nome : "Concórdia"},  
8125: {id : '4656', geocod : '4204350', nome : "Cordilheira Alta"},  
8126: {id : '4652', geocod : '4204400', nome : "Coronel Freitas"},  
8127: {id : '4669', geocod : '4204459', nome : "Coronel Martins"},  
8128: {id : '4729', geocod : '4204558', nome : "Correia Pinto"},  
8129: {id : '4999', geocod : '4204509', nome : "Corupá"},  
8130: {id : '6570', geocod : '4215679', nome : "Craveiro"},  
8131: {id : '5436', geocod : '4204608', nome : "Criciúma"},  
8132: {id : '4634', geocod : '4204707', nome : "Cunha Porã"},  
8133: {id : '4635', geocod : '4204756', nome : "Cunhataí"},  
8134: {id : '4730', geocod : '4204806', nome : "Curitibanos"},  
8135: {id : '6641', geocod : '4206900', nome : "Dalbérgia"},  
8136: {id : '6639', geocod : '4203600', nome : "DalPai"},  
8137: {id : '4600', geocod : '4204905', nome : "Descanso"},  
8138: {id : '4604', geocod : '4205001', nome : "Dionísio Cerqueira"},  
8139: {id : '4904', geocod : '4205100', nome : "Dona Emma"},  
8140: {id : '4962', geocod : '4205159', nome : "Doutor Pedrinho"},  
8141: {id : '6736', geocod : '4204301', nome : "Engenho Velho"},  
8142: {id : '6739', geocod : '4211900', nome : "Enseada de Brito"},  
8143: {id : '4670', geocod : '4205175', nome : "Entre Rios"},  
8144: {id : '5427', geocod : '4205191', nome : "Ermo"},  
8145: {id : '4537', geocod : '4205209', nome : "Erval Velho"},  
8146: {id : '6758', geocod : '4203600', nome : "Espinilho"},  
8147: {id : '6769', geocod : '4211207', nome : "Estação Cocal"},  
8148: {id : '4736', geocod : '4205308', nome : "Faxinal dos Guedes"},  
8149: {id : '6813', geocod : '4203808', nome : "Felipe Schmidt"},  
8150: {id : '6827', geocod : '4204202', nome : "Figueira"},  
8151: {id : '4628', geocod : '4205357', nome : "Flor do Sertão"},  
8152: {id : '5493', geocod : '4205407', nome : "Florianópolis"},  
8153: {id : '4662', geocod : '4205431', nome : "Formosa do Sul"},  
8154: {id : '5434', geocod : '4205456', nome : "Forquilhinha"},  
8155: {id : '4784', geocod : '4205506', nome : "Fraiburgo"},  
8156: {id : '6859', geocod : '4216909', nome : "Frederico Wastner"},  
8157: {id : '4792', geocod : '4205555', nome : "Frei Rogério"},  
8158: {id : '4673', geocod : '4205605', nome : "Galvão"},  
8159: {id : '6892', geocod : '4200903', nome : "Garcia"},  
8160: {id : '5462', geocod : '4205704', nome : "Garopaba"},  
8161: {id : '5022', geocod : '4205803', nome : "Garuva"},  
8162: {id : '4995', geocod : '4205902', nome : "Gaspar"},  
8163: {id : '6999', geocod : '4204202', nome : "Gôio En"},  
8164: {id : '4950', geocod : '4206009', nome : "Governador Celso Ramos"},  
8165: {id : '5455', geocod : '4206108', nome : "Grão Pará"},  
8166: {id : '6939', geocod : '4212239', nome : "Grápia"},  
8167: {id : '5450', geocod : '4206207', nome : "Gravatal"},  
8168: {id : '4992', geocod : '4206306', nome : "Guabiruba"},  
8169: {id : '6959', geocod : '4202305', nome : "Guaporanga"},  
8170: {id : '4603', geocod : '4206405', nome : "Guaraciaba"},  
8171: {id : '5004', geocod : '4206504', nome : "Guaramirim"},  
8172: {id : '4608', geocod : '4206603', nome : "Guarujá do Sul"},  
8173: {id : '6984', geocod : '4209607', nome : "Guatá"},  
8174: {id : '4647', geocod : '4206652', nome : "Guatambú"},  
8175: {id : '7006', geocod : '4201406', nome : "Hercílio Luz"},  
8176: {id : '7004', geocod : '4200408', nome : "Herciliópolis"},  
8177: {id : '4757', geocod : '4206702', nome : "Herval d'Oeste"},  
8178: {id : '4776', geocod : '4206751', nome : "Ibiam"},  
8179: {id : '4775', geocod : '4206801', nome : "Ibicaré"},  
8180: {id : '4907', geocod : '4206900', nome : "Ibirama"},  
8181: {id : '5435', geocod : '4207007', nome : "Içara"},  
8182: {id : '4996', geocod : '4207106', nome : "Ilhota"},  
8183: {id : '5454', geocod : '4207205', nome : "Imaruí"},  
8184: {id : '5461', geocod : '4207304', nome : "Imbituba"},  
8185: {id : '4931', geocod : '4207403', nome : "Imbuia"},  
8186: {id : '4988', geocod : '4207502', nome : "Indaial"},  
8187: {id : '9483', geocod : '4209300', nome : "Índios"},  
8188: {id : '7103', geocod : '4205407', nome : "Ingleses do Rio Vermelho"},  
8189: {id : '7116', geocod : '4206108', nome : "Invernada"},  
8190: {id : '4779', geocod : '4207577', nome : "Iomerê"},  
8191: {id : '4529', geocod : '4207601', nome : "Ipira"},  
8192: {id : '7128', geocod : '4214409', nome : "Ipoméia"},  
8193: {id : '4598', geocod : '4207650', nome : "Iporã do Oeste"},  
8194: {id : '4671', geocod : '4207684', nome : "Ipuaçu"},  
8195: {id : '4738', geocod : '4207700', nome : "Ipumirim"},  
8196: {id : '4629', geocod : '4207759', nome : "Iraceminha"},  
8197: {id : '7138', geocod : '4217907', nome : "Irakitan"},  
8198: {id : '4751', geocod : '4207809', nome : "Irani"},  
8199: {id : '7141', geocod : '4208104', nome : "Iraputã"},  
8200: {id : '4660', geocod : '4207858', nome : "Irati"},  
8201: {id : '4797', geocod : '4207908', nome : "Irineópolis"},  
8202: {id : '4715', geocod : '4208005', nome : "Itá"},  
8203: {id : '7179', geocod : '4208104', nome : "Itaió"},  
8204: {id : '4898', geocod : '4208104', nome : "Itaiópolis"},  
8205: {id : '5012', geocod : '4208203', nome : "Itajaí"},  
8206: {id : '7184', geocod : '4204905', nome : "Itajubá"},  
8207: {id : '5010', geocod : '4208302', nome : "Itapema"},  
8208: {id : '4592', geocod : '4208401', nome : "Itapiranga"},  
8209: {id : '5031', geocod : '4208450', nome : "Itapoá"},  
8210: {id : '7213', geocod : '4201307', nome : "Itapocu"},  
8211: {id : '7236', geocod : '4202404', nome : "Itoupava"},  
8212: {id : '4928', geocod : '4208500', nome : "Ituporanga"},  
8213: {id : '4752', geocod : '4208609', nome : "Jaborá"},  
8214: {id : '5425', geocod : '4208708', nome : "Jacinto Machado"},  
8215: {id : '5446', geocod : '4208807', nome : "Jaguaruna"},  
8216: {id : '7287', geocod : '4213609', nome : "Jangada do Sul"},  
8217: {id : '4998', geocod : '4208906', nome : "Jaraguá do Sul"},  
8218: {id : '4661', geocod : '4208955', nome : "Jardinópolis"},  
8219: {id : '4756', geocod : '4209003', nome : "Joaçaba"},  
8220: {id : '5005', geocod : '4209102', nome : "Joinville"},  
8221: {id : '4905', geocod : '4209151', nome : "José Boiteux"},  
8222: {id : '4666', geocod : '4209177', nome : "Jupiá"},  
8223: {id : '4755', geocod : '4209201', nome : "Lacerdópolis"},  
8224: {id : '4913', geocod : '4209300', nome : "Lages"},  
8225: {id : '7389', geocod : '4205407', nome : "Lagoa"},  
8226: {id : '7403', geocod : '4201000', nome : "Lagoa da Estiva"},  
8227: {id : '5453', geocod : '4209409', nome : "Laguna"},  
8228: {id : '4658', geocod : '4209458', nome : "Lajeado Grande"},  
8229: {id : '4900', geocod : '4209508', nome : "Laurentino"},  
8230: {id : '5440', geocod : '4209607', nome : "Lauro Muller"},  
8231: {id : '7452', geocod : '4203600', nome : "Leão"},  
8232: {id : '4793', geocod : '4209706', nome : "Lebon Régis"},  
8233: {id : '4936', geocod : '4209805', nome : "Leoberto Leal"},  
8234: {id : '4739', geocod : '4209854', nome : "Lindóia do Sul"},  
8235: {id : '7462', geocod : '4219606', nome : "Linha dasas Palmeira"},  
8236: {id : '4906', geocod : '4209904', nome : "Lontras"},  
8237: {id : '7471', geocod : '4219309', nome : "Lourdes"},  
8238: {id : '5002', geocod : '4210001', nome : "Luiz Alves"},  
8239: {id : '4758', geocod : '4210035', nome : "Luzerna"},  
8240: {id : '7498', geocod : '4211306', nome : "Machados"},  
8241: {id : '4782', geocod : '4210050', nome : "Macieira"},  
8242: {id : '4967', geocod : '4210100', nome : "Mafra"},  
8243: {id : '4942', geocod : '4210209', nome : "Major Gercino"},  
8244: {id : '4958', geocod : '4210308', nome : "Major Vieira"},  
8245: {id : '5430', geocod : '4210407', nome : "Maracajá"},  
8246: {id : '7545', geocod : '4217907', nome : "Marari"},  
8247: {id : '7547', geocod : '4216107', nome : "Maratá"},  
8248: {id : '4636', geocod : '4210506', nome : "Maravilha"},  
8249: {id : '7559', geocod : '4203808', nome : "Marcílio Dias"},  
8250: {id : '7560', geocod : '4204202', nome : "Marechal Bormann"},  
8251: {id : '4657', geocod : '4210555', nome : "Marema"},  
8252: {id : '7568', geocod : '4216701', nome : "Mariflor"},  
8253: {id : '7580', geocod : '4204806', nome : "Marombas Bossardi"},  
8254: {id : '5001', geocod : '4210605', nome : "Massaranduba"},  
8255: {id : '4788', geocod : '4210704', nome : "Matos Costa"},  
8256: {id : '5432', geocod : '4210803', nome : "Meleiro"},  
8257: {id : '7637', geocod : '4214003', nome : "Mirador"},  
8258: {id : '7653', geocod : '4207304', nome : "Mirim"},  
8259: {id : '4951', geocod : '4210852', nome : "Mirim Doce"},  
8260: {id : '4638', geocod : '4210902', nome : "Modelo"},  
8261: {id : '4624', geocod : '4211009', nome : "Mondaí"},  
8262: {id : '4783', geocod : '4211058', nome : "Monte Carlo"},  
8263: {id : '4959', geocod : '4211108', nome : "Monte Castelo"},  
8264: {id : '7719', geocod : '4211256', nome : "Morro Chato"},  
8265: {id : '5444', geocod : '4211207', nome : "Morro da Fumaça"},  
8266: {id : '5431', geocod : '4211256', nome : "Morro Grande"},  
8267: {id : '7721', geocod : '4215455', nome : "Morro Grande - Sangão"},  
8268: {id : '5013', geocod : '4211306', nome : "Navegantes"},  
8269: {id : '7798', geocod : '4211603', nome : "Nossa Senhora do Carmo"},  
8270: {id : '7820', geocod : '4212205', nome : "Nova Cultura"},  
8271: {id : '4648', geocod : '4211405', nome : "Nova Erechim"},  
8272: {id : '7827', geocod : '4217709', nome : "Nova Guarita"},  
8273: {id : '4651', geocod : '4211454', nome : "Nova Itaberaba"},  
8274: {id : '7838', geocod : '4209003', nome : "Nova Petrópolis"},  
8275: {id : '7841', geocod : '4217501', nome : "Nova Teutônia"},  
8276: {id : '4943', geocod : '4211504', nome : "Nova Trento"},  
8277: {id : '5399', geocod : '4211603', nome : "Nova Veneza"},  
8278: {id : '4664', geocod : '4211652', nome : "Novo Horizonte"},  
8279: {id : '5441', geocod : '4211702', nome : "Orleans"},  
8280: {id : '4917', geocod : '4211751', nome : "Otacílio Costa"},  
8281: {id : '4535', geocod : '4211801', nome : "Ouro"},  
8282: {id : '4745', geocod : '4211850', nome : "Ouro Verde"},  
8283: {id : '7898', geocod : '4206405', nome : "Ouro Verde - Guaraciaba"},  
8284: {id : '4654', geocod : '4211876', nome : "Paial"},  
8285: {id : '4914', geocod : '4211892', nome : "Painel"},  
8286: {id : '4947', geocod : '4211900', nome : "Palhoça"},  
8287: {id : '4538', geocod : '4212007', nome : "Palma Sola"},  
8288: {id : '4916', geocod : '4212056', nome : "Palmeira"},  
8289: {id : '4630', geocod : '4212106', nome : "Palmitos"},  
8290: {id : '8300', geocod : '4205407', nome : "Pántano do Sul"},  
8291: {id : '4960', geocod : '4212205', nome : "Papanduva"},  
8292: {id : '4601', geocod : '4212239', nome : "Paraíso"},  
8293: {id : '5421', geocod : '4212254', nome : "Passo de Torres"},  
8294: {id : '8008', geocod : '4217808', nome : "Passo Manso"},  
8295: {id : '4742', geocod : '4212270', nome : "Passos Maia"},  
8296: {id : '8036', geocod : '4203808', nome : "Paula Pereira"},  
8297: {id : '4946', geocod : '4212304', nome : "Paulo Lopes"},  
8298: {id : '5448', geocod : '4212403', nome : "Pedras Grandes"},  
8299: {id : '5015', geocod : '4212502', nome : "Penha"},  
8300: {id : '8084', geocod : '4216503', nome : "Pericó"},  
8301: {id : '4530', geocod : '4212601', nome : "Peritiba"},  
8302: {id : '8090', geocod : '4209409', nome : "Pescaria Brava"},  
8303: {id : '4923', geocod : '4212700', nome : "Petrolândia"},  
8304: {id : '5014', geocod : '4212809', nome : "Piçarras"},  
8305: {id : '8113', geocod : '4211702', nome : "Pindotiba"},  
8306: {id : '4639', geocod : '4212908', nome : "Pinhalzinho"},  
8307: {id : '8122', geocod : '4210209', nome : "Pinheiral"},  
8308: {id : '4778', geocod : '4213005', nome : "Pinheiro Preto"},  
8309: {id : '8131', geocod : '4203808', nome : "Pinheiros"},  
8310: {id : '8144', geocod : '4209102', nome : "Pirabeiraba"},  
8311: {id : '4531', geocod : '4213104', nome : "Piratuba"},  
8312: {id : '8176', geocod : '4204301', nome : "Planalto"},  
8313: {id : '4646', geocod : '4213153', nome : "Planalto Alegre"},  
8314: {id : '8258', geocod : '4207908', nome : "Poço Preto"},  
8315: {id : '4997', geocod : '4213203', nome : "Pomerode"},  
8316: {id : '4731', geocod : '4213302', nome : "Ponte Alta"},  
8317: {id : '4794', geocod : '4213351', nome : "Ponte Alta do Norte"},  
8318: {id : '4743', geocod : '4213401', nome : "Ponte Serrada"},  
8319: {id : '5008', geocod : '4213500', nome : "Porto Belo"},  
8320: {id : '4791', geocod : '4213609', nome : "Porto União"},  
8321: {id : '4954', geocod : '4213708', nome : "Pouso Redondo"},  
8322: {id : '5414', geocod : '4213807', nome : "Praia Grande"},  
8323: {id : '4750', geocod : '4213906', nome : "Presidente Castelo Branco"},  
8324: {id : '4902', geocod : '4214003', nome : "Presidente Getúlio"},  
8325: {id : '8279', geocod : '4216909', nome : "Presidente Juscelino"},  
8326: {id : '8281', geocod : '4204301', nome : "Presidente Kennedy"},  
8327: {id : '4933', geocod : '4214102', nome : "Presidente Nereu"},  
8328: {id : '4606', geocod : '4214151', nome : "Princesa"},  
8329: {id : '4663', geocod : '4214201', nome : "Quilombo"},  
8330: {id : '4937', geocod : '4214300', nome : "Rancho Queimado"},  
8331: {id : '8341', geocod : '4205407', nome : "Ratones"},  
8332: {id : '8354', geocod : '4211108', nome : "Residência Fuck"},  
8333: {id : '8381', geocod : '4205407', nome : "Ribeirão da Ilha"},  
8334: {id : '8380', geocod : '4209409', nome : "Ribeirão Pequeno"},  
8335: {id : '8403', geocod : '4212700', nome : "Rio Antinha"},  
8336: {id : '8406', geocod : '4208500', nome : "Rio Bonito"},  
8337: {id : '4785', geocod : '4214409', nome : "Rio das Antas"},  
8338: {id : '8431', geocod : '4211702', nome : "Rio das Furnas"},  
8339: {id : '4952', geocod : '4214508', nome : "Rio do Campo"},  
8340: {id : '4899', geocod : '4214607', nome : "Rio do Oeste"},  
8341: {id : '4901', geocod : '4214805', nome : "Rio do Sul"},  
8342: {id : '4990', geocod : '4214706', nome : "Rio dos Cedros"},  
8343: {id : '8425', geocod : '4207205', nome : "Rio dUna"},  
8344: {id : '5458', geocod : '4214904', nome : "Rio Fortuna"},  
8345: {id : '8411', geocod : '4204608', nome : "Rio Maina"},  
8346: {id : '4963', geocod : '4215000', nome : "Rio Negrinho"},  
8347: {id : '8419', geocod : '4210100', nome : "Rio Preto do Sul"},  
8348: {id : '4921', geocod : '4215059', nome : "Rio Rufino"},  
8349: {id : '4626', geocod : '4215075', nome : "Riqueza"},  
8350: {id : '4910', geocod : '4215109', nome : "Rodeio"},  
8351: {id : '4641', geocod : '4215208', nome : "Romelândia"},  
8352: {id : '8705', geocod : '4216206', nome : "Saí"},  
8353: {id : '4955', geocod : '4215307', nome : "Salete"},  
8354: {id : '4545', geocod : '4215356', nome : "Saltinho"},  
8355: {id : '4761', geocod : '4215406', nome : "Salto Veloso"},  
8356: {id : '8493', geocod : '4201406', nome : "Sanga da Toca"},  
8357: {id : '5445', geocod : '4215455', nome : "Sangão"},  
8358: {id : '4795', geocod : '4215505', nome : "Santa Cecília"},  
8359: {id : '8513', geocod : '4213609', nome : "Santa Cruz do Timbó"},  
8360: {id : '4596', geocod : '4215554', nome : "Santa Helena"},  
8361: {id : '8532', geocod : '4209003', nome : "Santa Helena - Joaçaba"},  
8362: {id : '8537', geocod : '4216503', nome : "Santa Izabel"},  
8363: {id : '8552', geocod : '4212106', nome : "Santa Lúcia"},  
8364: {id : '8561', geocod : '4202206', nome : "Santa Maria"},  
8365: {id : '5459', geocod : '4215604', nome : "Santa Rosa de Lima"},  
8366: {id : '5422', geocod : '4215653', nome : "Santa Rosa do Sul"},  
8367: {id : '4956', geocod : '4215679', nome : "Santa Terezinha"},  
8368: {id : '4541', geocod : '4215687', nome : "Santa Terezinha do Progresso"},  
8369: {id : '4668', geocod : '4215695', nome : "Santiago do Sul"},  
8370: {id : '4940', geocod : '4215703', nome : "Santo Amaro da Imperatriz"},  
8371: {id : '8648', geocod : '4205407', nome : "Santo Antônio de Lisboa"},  
8372: {id : '8819', geocod : '4211603', nome : "São Bento Baixo"},  
8373: {id : '4964', geocod : '4215802', nome : "São Bento do Sul"},  
8374: {id : '4644', geocod : '4215752', nome : "São Bernardino"},  
8375: {id : '4935', geocod : '4215901', nome : "São Bonifácio"},  
8376: {id : '4632', geocod : '4216008', nome : "São Carlos"},  
8377: {id : '8832', geocod : '4218301', nome : "São Cristovão"},  
8378: {id : '4732', geocod : '4216057', nome : "São Cristovão do Sul"},  
8379: {id : '4672', geocod : '4216107', nome : "São Domingos"},  
8380: {id : '5018', geocod : '4216206', nome : "São Francisco do Sul"},  
8381: {id : '8856', geocod : '4218400', nome : "São Gabriel"},  
8382: {id : '4945', geocod : '4216305', nome : "São João Batista"},  
8383: {id : '5003', geocod : '4216354', nome : "São João do Itaperiú"},  
8384: {id : '4594', geocod : '4216255', nome : "São João do Oeste"},  
8385: {id : '8969', geocod : '4205407', nome : "São João do Rio Vermelho"},  
8386: {id : '5417', geocod : '4216404', nome : "São João do Sul"},  
8387: {id : '5442', geocod : '4216503', nome : "São Joaquim"},  
8388: {id : '4948', geocod : '4216602', nome : "São José"},  
8389: {id : '4607', geocod : '4216701', nome : "São José do Cedro"},  
8390: {id : '4728', geocod : '4216800', nome : "São José do Cerrito"},  
8391: {id : '8973', geocod : '4200705', nome : "São Leonardo"},  
8392: {id : '4665', geocod : '4216909', nome : "São Lourenço do Oeste"},  
8393: {id : '5449', geocod : '4217006', nome : "São Ludgero"},  
8394: {id : '5460', geocod : '4217105', nome : "São Martinho"},  
8395: {id : '4642', geocod : '4217154', nome : "São Miguel da Boa Vista"},  
8396: {id : '9007', geocod : '4213609', nome : "São Miguel da Serra"},  
8397: {id : '4602', geocod : '4217204', nome : "São Miguel do Oeste"},  
8398: {id : '4941', geocod : '4217253', nome : "São Pedro de Alcântara"},  
8399: {id : '9021', geocod : '4205001', nome : "São Pedro Tobias"},  
8400: {id : '9049', geocod : '4216909', nome : "São Roque"},  
8401: {id : '9072', geocod : '4216503', nome : "São Sebastião do Arvoredo"},  
8402: {id : '9085', geocod : '4209706', nome : "São Sebastião do Sul"},  
8403: {id : '8687', geocod : '4210803', nome : "Sapiranga"},  
8404: {id : '4637', geocod : '4217303', nome : "Saudades"},  
8405: {id : '5000', geocod : '4217402', nome : "Schroeder"},  
8406: {id : '4733', geocod : '4217501', nome : "Seara"},  
8407: {id : '4543', geocod : '4217550', nome : "Serra Alta"},  
8408: {id : '5400', geocod : '4217600', nome : "Siderópolis"},  
8409: {id : '5423', geocod : '4217709', nome : "Sombrio"},  
8410: {id : '8789', geocod : '4202305', nome : "Sorocaba do Sul"},  
8411: {id : '4544', geocod : '4217758', nome : "Sul Brasil"},  
8412: {id : '4953', geocod : '4217808', nome : "Taió"},  
8413: {id : '4777', geocod : '4217907', nome : "Tangará"},  
8414: {id : '9162', geocod : '4203006', nome : "Taquara Verde"},  
8415: {id : '9165', geocod : '4214300', nome : "Taquaras"},  
8416: {id : '4540', geocod : '4217956', nome : "Tigrinhos"},  
8417: {id : '9198', geocod : '4216305', nome : "Tijipió"},  
8418: {id : '5006', geocod : '4218004', nome : "Tijucas"},  
8419: {id : '5419', geocod : '4218103', nome : "Timbé do Sul"},  
8420: {id : '4989', geocod : '4218202', nome : "Timbó"},  
8421: {id : '4796', geocod : '4218251', nome : "Timbó Grande"},  
8422: {id : '4965', geocod : '4218301', nome : "Três Barras"},  
8423: {id : '5438', geocod : '4218350', nome : "Treviso"},  
8424: {id : '5447', geocod : '4218400', nome : "Treze de Maio"},  
8425: {id : '4759', geocod : '4218509', nome : "Treze Tílias"},  
8426: {id : '4926', geocod : '4218608', nome : "Trombudo Central"},  
8427: {id : '5451', geocod : '4218707', nome : "Tubarão"},  
8428: {id : '4595', geocod : '4218756', nome : "Tunápolis"},  
8429: {id : '9270', geocod : '4203600', nome : "Tupitinga"},  
8430: {id : '5426', geocod : '4218806', nome : "Turvo"},  
8431: {id : '4650', geocod : '4218855', nome : "União do Oeste"},  
8432: {id : '5443', geocod : '4218905', nome : "Urubici"},  
8433: {id : '9302', geocod : '4213104', nome : "Uruguai"},  
8434: {id : '4915', geocod : '4218954', nome : "Urupema"},  
8435: {id : '5439', geocod : '4219002', nome : "Urussanga"},  
8436: {id : '4741', geocod : '4219101', nome : "Vargeão"},  
8437: {id : '4724', geocod : '4219150', nome : "Vargem"},  
8438: {id : '4753', geocod : '4219176', nome : "Vargem Bonita"},  
8439: {id : '9334', geocod : '4217105', nome : "Vargem do Cedro"},  
8440: {id : '4932', geocod : '4219200', nome : "Vidal Ramos"},  
8441: {id : '4780', geocod : '4219309', nome : "Videira"},  
8442: {id : '9370', geocod : '4216404', nome : "Vila Conceição"},  
8443: {id : '9411', geocod : '4215000', nome : "Vila de Volta Grande"},  
8444: {id : '9378', geocod : '4216107', nome : "Vila Milani"},  
8445: {id : '9384', geocod : '4207304', nome : "Vila Nova"},  
8446: {id : '4957', geocod : '4219358', nome : "Vitor Meireles"},  
8447: {id : '4903', geocod : '4219408', nome : "Witmarsum"},  
8448: {id : '4735', geocod : '4219507', nome : "Xanxerê"},  
8449: {id : '4734', geocod : '4219606', nome : "Xavantina"},  
8450: {id : '4659', geocod : '4219705', nome : "Xaxim"},  
8451: {id : '4534', geocod : '4219853', nome : "Zortéa"}  
8452: ],  
8453: SE : [  
8454: // UF = SE  
8455: {id : '5660', geocod : '2801405', nome : "Altos Verdes"},  
8456: {id : '1750', geocod : '2800100', nome : "Amparo de São Francisco"},  
8457: {id : '1739', geocod : '2800209', nome : "Aquidabã"},  
8458: {id : '5494', geocod : '2800308', nome : "Aracaju"},  
8459: {id : '1684', geocod : '2800407', nome : "Arauá"},  
8460: {id : '1699', geocod : '2800506', nome : "Areia Branca"},  
8461: {id : '1707', geocod : '2800605', nome : "Barra dos Coqueiros"},  
8462: {id : '5907', geocod : '2801306', nome : "Barracas"},  
8463: {id : '1687', geocod : '2800670', nome : "Boquim"},  
8464: {id : '1791', geocod : '2800704', nome : "Brejo Grande"},  
8465: {id : '1695', geocod : '2801009', nome : "Campo do Brito"},  
8466: {id : '1741', geocod : '2801108', nome : "Canhoba"},  
8467: {id : '1665', geocod : '2801207', nome : "Canindé de São Francisco"},  
8468: {id : '1717', geocod : '2801306', nome : "Capela"},  
8469: {id : '1720', geocod : '2801405', nome : "Carira"},  
8470: {id : '1716', geocod : '2801504', nome : "Carmópolis"},  
8471: {id : '1748', geocod : '2801603', nome : "Cedro de São João"},  
8472: {id : '1642', geocod : '2801702', nome : "Cristinápolis"},  
8473: {id : '1737', geocod : '2801900', nome : "Cumbe"},  
8474: {id : '1711', geocod : '2802007', nome : "Divina Pastora"},  
8475: {id : '1689', geocod : '2802106', nome : "Estância"},  
8476: {id : '1724', geocod : '2802205', nome : "Feira Nova"},  
8477: {id : '1697', geocod : '2802304', nome : "Frei Paulo"},  
8478: {id : '1743', geocod : '2802403', nome : "Gararu"},  
8479: {id : '1715', geocod : '2802502', nome : "General Maynard"},  
8480: {id : '1738', geocod : '2802601', nome : "Gracho Cardoso"},  
8481: {id : '1790', geocod : '2802700', nome : "Ilha das Flores"},  
8482: {id : '1683', geocod : '2802809', nome : "Indiaroba"},  
8483: {id : '1700', geocod : '2802908', nome : "Itabaiana"},  
8484: {id : '1645', geocod : '2803005', nome : "Itabaianinha"},  
8485: {id : '1740', geocod : '2803104', nome : "Itabi"},  
8486: {id : '7196', geocod : '2802809', nome : "Itanhi"},  
8487: {id : '1698', geocod : '2803203', nome : "Itaporanga d'Ajuda"},  
8488: {id : '1719', geocod : '2803302', nome : "Japaratuba"},  
8489: {id : '1749', geocod : '2803401', nome : "Japoatã"},  
8490: {id : '1691', geocod : '2803500', nome : "Lagarto"},  
8491: {id : '7396', geocod : '2804706', nome : "Lagoa Funda"},  
8492: {id : '1705', geocod : '2803609', nome : "Laranjeiras"},  
8493: {id : '1696', geocod : '2803708', nome : "Macambira"},  
8494: {id : '1746', geocod : '2803807', nome : "Malhada dos Bois"},  
8495: {id : '1702', geocod : '2803906', nome : "Malhador"},  
8496: {id : '1713', geocod : '2804003', nome : "Maruim"},  
8497: {id : '7646', geocod : '2801306', nome : "Miranda"},  
8498: {id : '1703', geocod : '2804102', nome : "Moita Bonita"},  
8499: {id : '1721', geocod : '2804201', nome : "Monte Alegre de Sergipe"},  
8500: {id : '1745', geocod : '2804300', nome : "Muribeca"},  
8501: {id : '1788', geocod : '2804409', nome : "Neópolis"},  
8502: {id : '1722', geocod : '2804458', nome : "Nossa Senhora Aparecida"},  
8503: {id : '1725', geocod : '2804508', nome : "Nossa Senhora da Glória"},  
8504: {id : '1736', geocod : '2804607', nome : "Nossa Senhora das Dores"},  
8505: {id : '1742', geocod : '2804706', nome : "Nossa Senhora de Lourdes"},  
8506: {id : '1706', geocod : '2804805', nome : "Nossa Senhora do Socorro"},  
8507: {id : '1786', geocod : '2804904', nome : "Pacatuba"},  
8508: {id : '7939', geocod : '2805802', nome : "Palmares"},  
8509: {id : '1694', geocod : '2805000', nome : "Pedra Mole"},  
8510: {id : '8054', geocod : '2801306', nome : "Pedras"},  
8511: {id : '1685', geocod : '2805109', nome : "Pedrinhas"},  
8512: {id : '1693', geocod : '2805208', nome : "Pinhão"},  
8513: {id : '1718', geocod : '2805307', nome : "Pirambu"},  
8514: {id : '1726', geocod : '2805406', nome : "Poço Redondo"},  
8515: {id : '1646', geocod : '2805505', nome : "Poço Verde"},  
8516: {id : '1728', geocod : '2805604', nome : "Porto da Folha"},  
8517: {id : '1752', geocod : '2805703', nome : "Propriá"},  
8518: {id : '1686', geocod : '2805802', nome : "Riachão do Dantas"},  
8519: {id : '1709', geocod : '2805901', nome : "Riachuelo"},  
8520: {id : '1701', geocod : '2806008', nome : "Ribeirópolis"},  
8521: {id : '1714', geocod : '2806107', nome : "Rosário do Catete"},  
8522: {id : '1690', geocod : '2806206', nome : "Salgado"},  
8523: {id : '8486', geocod : '2807402', nome : "Samambaia"},  
8524: {id : '1688', geocod : '2806305', nome : "Santa Luzia do Itanhy"},  
8525: {id : '1710', geocod : '2806503', nome : "Santa Rosa de Lima"},  
8526: {id : '1787', geocod : '2806404', nome : "Santana do São Francisco"},  
8527: {id : '1708', geocod : '2806602', nome : "Santo Amaro das Brotas"},  
8528: {id : '1704', geocod : '2806701', nome : "São Cristóvão"},  
8529: {id : '1692', geocod : '2806800', nome : "São Domingos"},  
8530: {id : '1747', geocod : '2806909', nome : "São Francisco"},  
8531: {id : '8998', geocod : '2802403', nome : "São Mateus da Palestina"},  
8532: {id : '1723', geocod : '2807006', nome : "São Miguel do Aleixo"},  
8533: {id : '1552', geocod : '2807105', nome : "Simão Dias"},  
8534: {id : '1712', geocod : '2807204', nome : "Siriri"},  
8535: {id : '1751', geocod : '2807303', nome : "Telha"},  
8536: {id : '1644', geocod : '2807402', nome : "Tobias Barreto"},  
8537: {id : '1643', geocod : '2807501', nome : "Tomar do Geru"},  
8538: {id : '1682', geocod : '2807600', nome : "Umbaúba"}  
8539: ],  
8540: SP : [  
8541: // UF = SP  
8542: {id : '3364', geocod : '3500105', nome : "Adamantina"},  
8543: {id : '9461', geocod : '3552205', nome : "Áden"},  
8544: {id : '3541', geocod : '3500204', nome : "Adolfo"},  
8545: {id : '5587', geocod : '3542206', nome : "Agissê"},  
8546: {id : '9473', geocod : '3548906', nome : "Água Vermelha"},  
8547: {id : '3867', geocod : '3500303', nome : "Aguaí"},  
8548: {id : '3903', geocod : '3500402', nome : "Águas da Prata"},  
8549: {id : '3885', geocod : '3500501', nome : "Águas de Lindóia"},  
8550: {id : '3549', geocod : '3500550', nome : "Águas de Santa Bárbara"},  
8551: {id : '3742', geocod : '3500600', nome : "Águas de São Pedro"},  
8552: {id : '3555', geocod : '3500709', nome : "Agudos"},  
8553: {id : '5591', geocod : '3515608', nome : "Agulha"},  
8554: {id : '5598', geocod : '3543907', nome : "Ajapi"},  
8555: {id : '3759', geocod : '3500758', nome : "Alambari"},  
8556: {id : '5602', geocod : '3505500', nome : "Alberto Moreira"},  
8557: {id : '5608', geocod : '3505708', nome : "Aldeia"},  
8558: {id : '5610', geocod : '3510609', nome : "Aldeia de Carapicuíba"},  
8559: {id : '5620', geocod : '3526803', nome : "Alfredo Guedes"},  
8560: {id : '3228', geocod : '3500808', nome : "Alfredo Marcondes"},  
8561: {id : '3620', geocod : '3500907', nome : "Altair"},  
8562: {id : '3852', geocod : '3501004', nome : "Altinópolis"},  
8563: {id : '3517', geocod : '3501103', nome : "Alto Alegre"},  
8564: {id : '5653', geocod : '3522406', nome : "Alto da Brancal"},  
8565: {id : '5646', geocod : '3537008', nome : "Alto Porã"},  
8566: {id : '3777', geocod : '3501152', nome : "Alumínio"},  
8567: {id : '3654', geocod : '3501202', nome : "Álvares Florence"},  
8568: {id : '3313', geocod : '3501301', nome : "Álvares Machado"},  
8569: {id : '3506', geocod : '3501400', nome : "Álvaro de Carvalho"},  
8570: {id : '3503', geocod : '3501509', nome : "Alvinlândia"},  
8571: {id : '5664', geocod : '3529005', nome : "Amadeu Amaral"},  
8572: {id : '5669', geocod : '3530102', nome : "Amandaba"},  
8573: {id : '5679', geocod : '3541406', nome : "Ameliópolis"},  
8574: {id : '3692', geocod : '3501608', nome : "Americana"},  
8575: {id : '3627', geocod : '3501707', nome : "Américo Brasiliense"},  
8576: {id : '3670', geocod : '3501806', nome : "Américo de Campos"},  
8577: {id : '3860', geocod : '3501905', nome : "Amparo"},  
8578: {id : '5686', geocod : '3523305', nome : "Ana Dias"},  
8579: {id : '3700', geocod : '3502002', nome : "Analândia"},  
8580: {id : '5716', geocod : '3519204', nome : "Anápolis"},  
8581: {id : '3395', geocod : '3502101', nome : "Andradina"},  
8582: {id : '3329', geocod : '3502200', nome : "Angatuba"},  
8583: {id : '3471', geocod : '3502309', nome : "Anhembi"},  
8584: {id : '3311', geocod : '3502408', nome : "Anhumas"},  
8585: {id : '4006', geocod : '3502507', nome : "Aparecida"},  
8586: {id : '5720', geocod : '3550100', nome : "Aparecida de São Manuel"},  
8587: {id : '5721', geocod : '3547403', nome : "Aparecida do Bonito"},  
8588: {id : '5722', geocod : '3531308', nome : "Aparecida do Monte Alto"},  
8589: {id : '3399', geocod : '3502606', nome : "Aparecida d'Oeste"},  
8590: {id : '5053', geocod : '3502705', nome : "Apiaí"},  
8591: {id : '5727', geocod : '3510203', nome : "Apiaí Mirim"},  
8592: {id : '5731', geocod : '3534807', nome : "Arabela"},  
8593: {id : '5735', geocod : '3508009', nome : "Aracaçu"},  
8594: {id : '5761', geocod : '3502705', nome : "Araçaíba"},  
8595: {id : '3791', geocod : '3502754', nome : "Araçariguama"},  
8596: {id : '3388', geocod : '3502804', nome : "Araçatuba"},  
8597: {id : '3771', geocod : '3502903', nome : "Araçoiaba da Serra"},  
8598: {id : '3938', geocod : '3503000', nome : "Aramina"},  
8599: {id : '3468', geocod : '3503109', nome : "Arandu"},  
8600: {id : '4076', geocod : '3503158', nome : "Arapeí"},  
8601: {id : '3626', geocod : '3503208', nome : "Araraquara"},  
8602: {id : '3828', geocod : '3503307', nome : "Araras"},  
8603: {id : '5757', geocod : '3541208', nome : "Araxãs"},  
8604: {id : '5762', geocod : '3501905', nome : "Arcadas"},  
8605: {id : '3378', geocod : '3503356', nome : "Arco-Íris"},  
8606: {id : '3557', geocod : '3503406', nome : "Arealva"},  
8607: {id : '4063', geocod : '3503505', nome : "Areias"},  
8608: {id : '3560', geocod : '3503604', nome : "Areiópolis"},  
8609: {id : '3590', geocod : '3503703', nome : "Ariranha"},  
8610: {id : '5777', geocod : '3509908', nome : "Ariri"},  
8611: {id : '5794', geocod : '3538709', nome : "Artemis"},  
8612: {id : '3822', geocod : '3503802', nome : "Artur Nogueira"},  
8613: {id : '3735', geocod : '3503901', nome : "Arujá"},  
8614: {id : '3421', geocod : '3503950', nome : "Aspásia"},  
8615: {id : '3349', geocod : '3504008', nome : "Assis"},  
8616: {id : '5801', geocod : '3543907', nome : "Assistência"},  
8617: {id : '3729', geocod : '3504107', nome : "Atibaia"},  
8618: {id : '5809', geocod : '3516002', nome : "Atlântida"},  
8619: {id : '3411', geocod : '3504206', nome : "Auriflama"},  
8620: {id : '3513', geocod : '3504305', nome : "Avaí"},  
8621: {id : '3526', geocod : '3504404', nome : "Avanhandava"},  
8622: {id : '3469', geocod : '3504503', nome : "Avaré"},  
8623: {id : '5817', geocod : '3529005', nome : "Avencas"},  
8624: {id : '5827', geocod : '3521002', nome : "Bacaetava"},  
8625: {id : '5832', geocod : '3508801', nome : "Bacuriti"},  
8626: {id : '3663', geocod : '3504602', nome : "Bady Bassitt"},  
8627: {id : '5835', geocod : '3533908', nome : "Baguaçu"},  
8628: {id : '5839', geocod : '3532306', nome : "Bairro Alto"},  
8629: {id : '3689', geocod : '3504701', nome : "Balbinos"},  
8630: {id : '3660', geocod : '3504800', nome : "Bálsamo"},  
8631: {id : '4077', geocod : '3504909', nome : "Bananal"},  
8632: {id : '5861', geocod : '3552304', nome : "Bandeirante dOeste"},  
8633: {id : '5935', geocod : '3522604', nome : "Barão Ataliba Nogueira"},  
8634: {id : '3436', geocod : '3505005', nome : "Barão de Antonina"},  
8635: {id : '5937', geocod : '3509502', nome : "Barão de Geraldo"},  
8636: {id : '3527', geocod : '3505104', nome : "Barbosa"},  
8637: {id : '3568', geocod : '3505203', nome : "Bariri"},  
8638: {id : '3565', geocod : '3505302', nome : "Barra Bonita"},  
8639: {id : '5050', geocod : '3505351', nome : "Barra do Chapéu"},  
8640: {id : '5056', geocod : '3505401', nome : "Barra do Turvo"},  
8641: {id : '5877', geocod : '3532504', nome : "Barra Dourada"},  
8642: {id : '5934', geocod : '3508702', nome : "Barrânia"},  
8643: {id : '3679', geocod : '3505500', nome : "Barretos"},  
8644: {id : '3633', geocod : '3505609', nome : "Barrinha"},  
8645: {id : '3798', geocod : '3505708', nome : "Barueri"},  
8646: {id : '3372', geocod : '3505807', nome : "Bastos"},  
8647: {id : '3940', geocod : '3505906', nome : "Batatais"},  
8648: {id : '5946', geocod : '3538600', nome : "Batatuba"},  
8649: {id : '5951', geocod : '3533809', nome : "Batista Botelho"},  
8650: {id : '3554', geocod : '3506003', nome : "Bauru"},  
8651: {id : '3680', geocod : '3506102', nome : "Bebedouro"},  
8652: {id : '5977', geocod : '3548906', nome : "Bela Vista de São Carlos"},  
8653: {id : '3383', geocod : '3506201', nome : "Bento de Abreu"},  
8654: {id : '3498', geocod : '3506300', nome : "Bernardino de Campos"},  
8655: {id : '3726', geocod : '3506359', nome : "Bertioga"},  
8656: {id : '3386', geocod : '3506409', nome : "Bilac"},  
8657: {id : '3387', geocod : '3506508', nome : "Birigui"},  
8658: {id : '6005', geocod : '3530607', nome : "Biritiba Ussu"},  
8659: {id : '3962', geocod : '3506607', nome : "Biritiba-Mirim"},  
8660: {id : '3596', geocod : '3506706', nome : "Boa Esperança do Sul"},  
8661: {id : '6017', geocod : '3508900', nome : "Boa Esperança dOeste"},  
8662: {id : '6034', geocod : '3501202', nome : "Boa Vista dos Andrades"},  
8663: {id : '6029', geocod : '3552502', nome : "Boa Vista Paulista"},  
8664: {id : '3569', geocod : '3506805', nome : "Bocaina"},  
8665: {id : '3331', geocod : '3506904', nome : "Bofete"},  
8666: {id : '3773', geocod : '3507001', nome : "Boituva"},  
8667: {id : '6048', geocod : '3508405', nome : "Bom Fim do Bom Jesus"},  
8668: {id : '3730', geocod : '3507100', nome : "Bom Jesus dos Perdões"},  
8669: {id : '6066', geocod : '3502200', nome : "Bom Retiro da Esperança"},  
8670: {id : '5051', geocod : '3507159', nome : "Bom Sucesso de Itararé"},  
8671: {id : '6076', geocod : '3543402', nome : "Bonfim Paulista"},  
8672: {id : '3354', geocod : '3507209', nome : "Borá"},  
8673: {id : '3558', geocod : '3507308', nome : "Boracéia"},  
8674: {id : '3576', geocod : '3507407', nome : "Borborema"},  
8675: {id : '3551', geocod : '3507456', nome : "Borebi"},  
8676: {id : '6094', geocod : '3506102', nome : "Botafogo"},  
8677: {id : '6096', geocod : '3545605', nome : "Botelho"},  
8678: {id : '3562', geocod : '3507506', nome : "Botucatu"},  
8679: {id : '6098', geocod : '3509601', nome : "Botujuru"},  
8680: {id : '6104', geocod : '3514809', nome : "Braço"},  
8681: {id : '3871', geocod : '3507605', nome : "Bragança Paulista"},  
8682: {id : '6124', geocod : '3530607', nome : "Brás Cubas"},  
8683: {id : '6102', geocod : '3515509', nome : "Brasitânia"},  
8684: {id : '3516', geocod : '3507704', nome : "Braúna"},  
8685: {id : '3524', geocod : '3507753', nome : "Brejo Alegre"},  
8686: {id : '6120', geocod : '3552205', nome : "Brigadeiro Tobias"},  
8687: {id : '3848', geocod : '3507803', nome : "Brodowski"},  
8688: {id : '3474', geocod : '3507902', nome : "Brotas"},  
8689: {id : '6128', geocod : '3503208', nome : "Bueno de Andrada"},  
8690: {id : '3325', geocod : '3508009', nome : "Buri"},  
8691: {id : '3525', geocod : '3508108', nome : "Buritama"},  
8692: {id : '3947', geocod : '3508207', nome : "Buritizal"},  
8693: {id : '3511', geocod : '3508306', nome : "Cabrália Paulista"},  
8694: {id : '3794', geocod : '3508405', nome : "Cabreúva"},  
8695: {id : '3970', geocod : '3508504', nome : "Caçapava"},  
8696: {id : '6161', geocod : '3539301', nome : "Cachoeira de Emas"},  
8697: {id : '4059', geocod : '3508603', nome : "Cachoeira Paulista"},  
8698: {id : '3918', geocod : '3508702', nome : "Caconde"},  
8699: {id : '3532', geocod : '3508801', nome : "Cafelândia"},  
8700: {id : '6184', geocod : '3508801', nome : "Cafesópolis"},  
8701: {id : '3355', geocod : '3508900', nome : "Caiabu"},  
8702: {id : '3803', geocod : '3509007', nome : "Caieiras"},  
8703: {id : '3137', geocod : '3509106', nome : "Caiuá"},  
8704: {id : '3801', geocod : '3509205', nome : "Cajamar"},  
8705: {id : '5059', geocod : '3509254', nome : "Cajati"},  
8706: {id : '3622', geocod : '3509304', nome : "Cajobi"},  
8707: {id : '3851', geocod : '3509403', nome : "Cajuru"},  
8708: {id : '6201', geocod : '3552205', nome : "Cajuru do Sul"},  
8709: {id : '6220', geocod : '3520400', nome : "Cambaquara"},  
8710: {id : '6221', geocod : '3519600', nome : "Cambaratiba"},  
8711: {id : '6234', geocod : '3513900', nome : "Campestrinho"},  
8712: {id : '6236', geocod : '3543006', nome : "Campina de Fora"},  
8713: {id : '3328', geocod : '3509452', nome : "Campina do Monte Alegre"},  
8714: {id : '6237', geocod : '3541307', nome : "Campinal"},  
8715: {id : '3853', geocod : '3509502', nome : "Campinas"},  
8716: {id : '3808', geocod : '3509601', nome : "Campo Limpo Paulista"},  
8717: {id : '6261', geocod : '3513603', nome : "Campos de Cunha"},  
8718: {id : '4003', geocod : '3509700', nome : "Campos do Jordão"},  
8719: {id : '3482', geocod : '3509809', nome : "Campos Novos Paulista"},  
8720: {id : '5062', geocod : '3509908', nome : "Cananéia"},  
8721: {id : '4058', geocod : '3509957', nome : "Canas"},  
8722: {id : '6626', geocod : '3540200', nome : "Cândia"},  
8723: {id : '3348', geocod : '3510005', nome : "Cândido Mota"},  
8724: {id : '3600', geocod : '3510104', nome : "Cândido Rodrigues"},  
8725: {id : '6287', geocod : '3550605', nome : "Canguera"},  
8726: {id : '3493', geocod : '3510153', nome : "Canitar"},  
8727: {id : '4897', geocod : '3510203', nome : "Capão Bonito"},  
8728: {id : '3761', geocod : '3510302', nome : "Capela do Alto"},  
8729: {id : '3749', geocod : '3510401', nome : "Capivari"},  
8730: {id : '6308', geocod : '3524105', nome : "Capivari da Mata"},  
8731: {id : '6312', geocod : '3546405', nome : "Caporanga"},  
8732: {id : '6313', geocod : '3529401', nome : "Capuava"},  
8733: {id : '3973', geocod : '3510500', nome : "Caraguatatuba"},  
8734: {id : '3799', geocod : '3510609', nome : "Carapicuíba"},  
8735: {id : '6348', geocod : '3514908', nome : "Cardeal"},  
8736: {id : '3655', geocod : '3510708', nome : "Cardoso"},  
8737: {id : '3899', geocod : '3510807', nome : "Casa Branca"},  
8738: {id : '3907', geocod : '3510906', nome : "Cássia dos Coqueiros"},  
8739: {id : '3392', geocod : '3511003', nome : "Castilho"},  
8740: {id : '3587', geocod : '3511102', nome : "Catanduva"},  
8741: {id : '3585', geocod : '3511201', nome : "Catiguá"},  
8742: {id : '6401', geocod : '3550001', nome : "Catuçaba"},  
8743: {id : '6403', geocod : '3513009', nome : "Caucaia do Alto"},  
8744: {id : '3616', geocod : '3511300', nome : "Cedral"},  
8745: {id : '3548', geocod : '3511409', nome : "Cerqueira César"},  
8746: {id : '3765', geocod : '3511508', nome : "Cerquilho"},  
8747: {id : '3762', geocod : '3511607', nome : "Cesário Lange"},  
8748: {id : '6436', geocod : '3530607', nome : "Cezar de Souza"},  
8749: {id : '3743', geocod : '3511706', nome : "Charqueada"},  
8750: {id : '3494', geocod : '3557204', nome : "Chavantes"},  
8751: {id : '6449', geocod : '3539806', nome : "Cidade Kemel"},  
8752: {id : '6456', geocod : '3515103', nome : "Cipó Guaçu"},  
8753: {id : '6463', geocod : '3546405', nome : "Clarínia"},  
8754: {id : '3382', geocod : '3511904', nome : "Clementina"},  
8755: {id : '6467', geocod : '3551108', nome : "Cocaes"},  
8756: {id : '3678', geocod : '3512001', nome : "Colina"},  
8757: {id : '3685', geocod : '3512100', nome : "Colômbia"},  
8758: {id : '6505', geocod : '3535507', nome : "Conceição de Monte Alegre"},  
8759: {id : '3863', geocod : '3512209', nome : "Conchal"},  
8760: {id : '3470', geocod : '3512308', nome : "Conchas"},  
8761: {id : '3697', geocod : '3512407', nome : "Cordeirópolis"},  
8762: {id : '3522', geocod : '3512506', nome : "Coroados"},  
8763: {id : '6540', geocod : '3501301', nome : "Coronel Goulart"},  
8764: {id : '3439', geocod : '3512605', nome : "Coronel Macedo"},  
8765: {id : '6544', geocod : '3538907', nome : "Corredeira"},  
8766: {id : '6629', geocod : '3524303', nome : "Córrego Rico"},  
8767: {id : '3699', geocod : '3512704', nome : "Corumbataí"},  
8768: {id : '3695', geocod : '3512803', nome : "Cosmópolis"},  
8769: {id : '3661', geocod : '3512902', nome : "Cosmorama"},  
8770: {id : '6550', geocod : '3530201', nome : "Costa Machado"},  
8771: {id : '3785', geocod : '3513009', nome : "Cotia"},  
8772: {id : '3835', geocod : '3513108', nome : "Cravinhos"},  
8773: {id : '3950', geocod : '3513207', nome : "Cristais Paulista"},  
8774: {id : '6591', geocod : '3551702', nome : "Cruz das Posses"},  
8775: {id : '3344', geocod : '3513306', nome : "Cruzália"},  
8776: {id : '4060', geocod : '3513405', nome : "Cruzeiro"},  
8777: {id : '3810', geocod : '3513504', nome : "Cubatão"},  
8778: {id : '6601', geocod : '3530201', nome : "Cuiabá Paulista"},  
8779: {id : '3979', geocod : '3513603', nome : "Cunha"},  
8780: {id : '6621', geocod : '3552700', nome : "Curupá"},  
8781: {id : '6640', geocod : '3535200', nome : "Dalas"},  
8782: {id : '3838', geocod : '3513702', nome : "Descalvado"},  
8783: {id : '3811', geocod : '3513801', nome : "Diadema"},  
8784: {id : '3408', geocod : '3513850', nome : "Dirce Reis"},  
8785: {id : '6665', geocod : '3529005', nome : "Dirceu"},  
8786: {id : '3906', geocod : '3513900', nome : "Divinolândia"},  
8787: {id : '3598', geocod : '3514007', nome : "Dobrada"},  
8788: {id : '3571', geocod : '3514106', nome : "Dois Córregos"},  
8789: {id : '3429', geocod : '3514205', nome : "Dolcinópolis"},  
8790: {id : '6684', geocod : '3500709', nome : "Domélia"},  
8791: {id : '3572', geocod : '3514304', nome : "Dourado"},  
8792: {id : '3315', geocod : '3514403', nome : "Dracena"},  
8793: {id : '3510', geocod : '3514502', nome : "Duartina"},  
8794: {id : '3833', geocod : '3514601', nome : "Dumont"},  
8795: {id : '6708', geocod : '3535002', nome : "Duplo Céu"},  
8796: {id : '3484', geocod : '3514700', nome : "Echaporã"},  
8797: {id : '5061', geocod : '3514809', nome : "Eldorado"},  
8798: {id : '6713', geocod : '3522604', nome : "Eleutério"},  
8799: {id : '3753', geocod : '3514908', nome : "Elias Fausto"},  
8800: {id : '3584', geocod : '3514924', nome : "Elisiário"},  
8801: {id : '3591', geocod : '3514957', nome : "Embaúba"},  
8802: {id : '3787', geocod : '3515004', nome : "Embu"},  
8803: {id : '3784', geocod : '3515103', nome : "Embu-Guaçu"},  
8804: {id : '3229', geocod : '3515129', nome : "Emilianópolis"},  
8805: {id : '6721', geocod : '3541406', nome : "Eneida"},  
8806: {id : '6722', geocod : '3531407', nome : "Engenheiro Balduíno"},  
8807: {id : '6729', geocod : '3549805', nome : "Engenheiro Schmidt"},  
8808: {id : '3829', geocod : '3515152', nome : "Engo. Coelho"},  
8809: {id : '6746', geocod : '3550308', nome : "Ermelino Matarazzo"},  
8810: {id : '6749', geocod : '3544509', nome : "Esmeralda"},  
8811: {id : '6755', geocod : '3542404', nome : "Espigão"},  
8812: {id : '3869', geocod : '3515186', nome : "Espírito Santo do Pinhal"},  
8813: {id : '3501', geocod : '3515194', nome : "Espírito Santo do Turvo"},  
8814: {id : '3866', geocod : '3557303', nome : "Estiva Gerbi"},  
8815: {id : '3308', geocod : '3515301', nome : "Estrela do Norte"},  
8816: {id : '3427', geocod : '3515202', nome : "Estrela d'Oeste"},  
8817: {id : '3201', geocod : '3515350', nome : "Euclides da Cunha Paulista"},  
8818: {id : '6782', geocod : '3549904', nome : "Eugênio de Melo"},  
8819: {id : '3459', geocod : '3515400', nome : "Fartura"},  
8820: {id : '6876', geocod : '3517307', nome : "Fátima"},  
8821: {id : '6877', geocod : '3555307', nome : "Fátima Paulista"},  
8822: {id : '6803', geocod : '3511607', nome : "Fazenda Velha"},  
8823: {id : '3601', geocod : '3515608', nome : "Fernando Prestes"},  
8824: {id : '3647', geocod : '3515509', nome : "Fernandópolis"},  
8825: {id : '3509', geocod : '3515657', nome : "Fernão"},  
8826: {id : '3731', geocod : '3515707', nome : "Ferraz de Vasconcelos"},  
8827: {id : '3231', geocod : '3515806', nome : "Flora Rica"},  
8828: {id : '3641', geocod : '3515905', nome : "Floreal"},  
8829: {id : '6838', geocod : '3541406', nome : "Floresta do Sul"},  
8830: {id : '3359', geocod : '3516002', nome : "Flórida Paulista"},  
8831: {id : '3343', geocod : '3516101', nome : "Florínia"},  
8832: {id : '3944', geocod : '3516200', nome : "Franca"},  
8833: {id : '3807', geocod : '3516309', nome : "Francisco Morato"},  
8834: {id : '3806', geocod : '3516408', nome : "Franco da Rocha"},  
8835: {id : '6868', geocod : '3510005', nome : "Frutal do Campo"},  
8836: {id : '3381', geocod : '3516507', nome : "Gabriel Monteiro"},  
8837: {id : '3505', geocod : '3516606', nome : "Gália"},  
8838: {id : '3507', geocod : '3516705', nome : "Garça"},  
8839: {id : '6894', geocod : '3542206', nome : "Gardênia"},  
8840: {id : '3639', geocod : '3516804', nome : "Gastão Vidigal"},  
8841: {id : '3597', geocod : '3516853', nome : "Gavião Peixoto"},  
8842: {id : '3412', geocod : '3516903', nome : "General Salgado"},  
8843: {id : '3519', geocod : '3517000', nome : "Getulina"},  
8844: {id : '3521', geocod : '3517109', nome : "Glicério"},  
8845: {id : '6930', geocod : '3522307', nome : "Gramadinho"},  
8846: {id : '6940', geocod : '3529203', nome : "Guachos"},  
8847: {id : '6943', geocod : '3536703', nome : "Guaianás"},  
8848: {id : '6942', geocod : '3550308', nome : "Guaianases"},  
8849: {id : '3534', geocod : '3517208', nome : "Guaiçara"},  
8850: {id : '3518', geocod : '3517307', nome : "Guaimbê"},  
8851: {id : '3686', geocod : '3517406', nome : "Guaíra"},  
8852: {id : '6954', geocod : '3538907', nome : "Guamium"},  
8853: {id : '3618', geocod : '3517505', nome : "Guapiaçu"},  
8854: {id : '5058', geocod : '3517604', nome : "Guapiara"},  
8855: {id : '6958', geocod : '3527108', nome : "Guapiranga"},  
8856: {id : '3936', geocod : '3517703', nome : "Guará"},  
8857: {id : '3368', geocod : '3517802', nome : "Guaraçaí"},  
8858: {id : '3624', geocod : '3517901', nome : "Guaraci"},  
8859: {id : '6961', geocod : '3555109', nome : "Guaraciaba d'Oeste"},  
8860: {id : '3431', geocod : '3518008', nome : "Guarani d'Oeste"},  
8861: {id : '3531', geocod : '3518107', nome : "Guarantã"},  
8862: {id : '6966', geocod : '3542909', nome : "Guarapiranga"},  
8863: {id : '6967', geocod : '3514106', nome : "Guarapuã"},  
8864: {id : '3385', geocod : '3518206', nome : "Guararapes"},  
8865: {id : '3963', geocod : '3518305', nome : "Guararema"},  
8866: {id : '4007', geocod : '3518404', nome : "Guaratinguetá"},  
8867: {id : '3611', geocod : '3518503', nome : "Guareí"},  
8868: {id : '3631', geocod : '3518602', nome : "Guariba"},  
8869: {id : '6976', geocod : '3541109', nome : "Guaricanga"},  
8870: {id : '6977', geocod : '3553708', nome : "Guariroba"},  
8871: {id : '6979', geocod : '3522406', nome : "Guarizinho"},  
8872: {id : '3819', geocod : '3518701', nome : "Guarujá"},  
8873: {id : '3727', geocod : '3518800', nome : "Guarulhos"},  
8874: {id : '5517', geocod : '3518859', nome : "Guatapará"},  
8875: {id : '3407', geocod : '3518909', nome : "Guzolândia"},  
8876: {id : '3379', geocod : '3519006', nome : "Herculândia"},  
8877: {id : '3824', geocod : '3519055', nome : "Holambra"},  
8878: {id : '3690', geocod : '3519071', nome : "Hortolândia"},  
8879: {id : '3575', geocod : '3519105', nome : "Iacanga"},  
8880: {id : '3374', geocod : '3519204', nome : "Iacri"},  
8881: {id : '3550', geocod : '3519253', nome : "Iaras"},  
8882: {id : '3831', geocod : '3519303', nome : "Ibaté"},  
8883: {id : '7034', geocod : '3553401', nome : "Ibiporanga"},  
8884: {id : '3583', geocod : '3519402', nome : "Ibirá"},  
8885: {id : '3477', geocod : '3519501', nome : "Ibirarema"},  
8886: {id : '3578', geocod : '3519600', nome : "Ibitinga"},  
8887: {id : '7046', geocod : '3538709', nome : "Ibitiruna"},  
8888: {id : '7048', geocod : '3539509', nome : "Ibitiúva"},  
8889: {id : '7049', geocod : '3505500', nome : "Ibitu"},  
8890: {id : '3770', geocod : '3519709', nome : "Ibiúna"},  
8891: {id : '3625', geocod : '3519808', nome : "Icém"},  
8892: {id : '7065', geocod : '3532603', nome : "Ida Iolanda"},  
8893: {id : '3335', geocod : '3519907', nome : "Iepê"},  
8894: {id : '7076', geocod : '3537008', nome : "Igaçaba"},  
8895: {id : '3564', geocod : '3520004', nome : "Igaraçu do Tietê"},  
8896: {id : '7072', geocod : '3530508', nome : "Igaraí"},  
8897: {id : '3939', geocod : '3520103', nome : "Igarapava"},  
8898: {id : '3739', geocod : '3520202', nome : "Igaratá"},  
8899: {id : '5067', geocod : '3520301', nome : "Iguape"},  
8900: {id : '5066', geocod : '3520426', nome : "Ilha Comprida"},  
8901: {id : '3400', geocod : '3520442', nome : "Ilha Solteira"},  
8902: {id : '3972', geocod : '3520400', nome : "Ilhabela"},  
8903: {id : '7096', geocod : '3516002', nome : "Indaiá do Aguapeí"},  
8904: {id : '3780', geocod : '3520509', nome : "Indaiatuba"},  
8905: {id : '3338', geocod : '3520608', nome : "Indiana"},  
8906: {id : '2545', geocod : '3520707', nome : "Indiaporã"},  
8907: {id : '7104', geocod : '3533007', nome : "Ingás"},  
8908: {id : '3362', geocod : '3520806', nome : "Inúbia Paulista"},  
8909: {id : '3495', geocod : '3520905', nome : "Ipaussu"},  
8910: {id : '3772', geocod : '3521002', nome : "Iperó"},  
8911: {id : '3744', geocod : '3521101', nome : "Ipeúna"},  
8912: {id : '3667', geocod : '3521150', nome : "Ipiguá"},  
8913: {id : '5057', geocod : '3521200', nome : "Iporanga"},  
8914: {id : '5520', geocod : '3521309', nome : "Ipuã"},  
8915: {id : '3752', geocod : '3521408', nome : "Iracemápolis"},  
8916: {id : '7143', geocod : '3557204', nome : "Irapé"},  
8917: {id : '3544', geocod : '3521507', nome : "Irapuã"},  
8918: {id : '3357', geocod : '3521606', nome : "Irapuru"},  
8919: {id : '3464', geocod : '3521705', nome : "Itaberá"},  
8920: {id : '7155', geocod : '3543006', nome : "Itabôa"},  
8921: {id : '3467', geocod : '3521804', nome : "Itaí"},  
8922: {id : '7169', geocod : '3550308', nome : "Itaim Paulista"},  
8923: {id : '7177', geocod : '3531407', nome : "Itaiúba"},  
8924: {id : '3582', geocod : '3521903', nome : "Itajobi"},  
8925: {id : '3577', geocod : '3522000', nome : "Itaju"},  
8926: {id : '5073', geocod : '3522109', nome : "Itanhaém"},  
8927: {id : '5052', geocod : '3522158', nome : "Itaóca"},  
8928: {id : '3786', geocod : '3522208', nome : "Itapecerica da Serra"},  
8929: {id : '3612', geocod : '3522307', nome : "Itapetininga"},  
8930: {id : '7208', geocod : '3514809', nome : "Itapeúna"},  
8931: {id : '3465', geocod : '3522406', nome : "Itapeva"},  
8932: {id : '3792', geocod : '3522505', nome : "Itapevi"},  
8933: {id : '3868', geocod : '3522604', nome : "Itapira"},  
8934: {id : '5049', geocod : '3522653', nome : "Itapirapuã Paulista"},  
8935: {id : '3579', geocod : '3522703', nome : "Itápolis"},  
8936: {id : '3438', geocod : '3522802', nome : "Itaporanga"},  
8937: {id : '3567', geocod : '3522901', nome : "Itapuí"},  
8938: {id : '3393', geocod : '3523008', nome : "Itapura"},  
8939: {id : '3733', geocod : '3523107', nome : "Itaquaquecetuba"},  
8940: {id : '7220', geocod : '3550308', nome : "Itaquera"},  
8941: {id : '7221', geocod : '3523602', nome : "Itaqueri da Serra"},  
8942: {id : '4912', geocod : '3523206', nome : "Itararé"},  
8943: {id : '5071', geocod : '3523305', nome : "Itariri"},  
8944: {id : '3825', geocod : '3523404', nome : "Itatiba"},  
8945: {id : '3327', geocod : '3523503', nome : "Itatinga"},  
8946: {id : '3745', geocod : '3523602', nome : "Itirapina"},  
8947: {id : '3946', geocod : '3523701', nome : "Itirapuã"},  
8948: {id : '3900', geocod : '3523800', nome : "Itobi"},  
8949: {id : '7235', geocod : '3539202', nome : "Itororó do Paranapanema"},  
8950: {id : '3778', geocod : '3523909', nome : "Itu"},  
8951: {id : '3795', geocod : '3524006', nome : "Itupeva"},  
8952: {id : '3937', geocod : '3524105', nome : "Ituverava"},  
8953: {id : '7240', geocod : '3508900', nome : "Iubatinga"},  
8954: {id : '3683', geocod : '3524204', nome : "Jaborandi"},  
8955: {id : '3608', geocod : '3524303', nome : "Jaboticabal"},  
8956: {id : '7255', geocod : '3508405', nome : "Jacaré"},  
8957: {id : '3965', geocod : '3524402', nome : "Jacareí"},  
8958: {id : '3659', geocod : '3524501', nome : "Jaci"},  
8959: {id : '7261', geocod : '3514403', nome : "Jaciporã"},  
8960: {id : '7264', geocod : '3503406', nome : "Jacuba"},  
8961: {id : '5060', geocod : '3524600', nome : "Jacupiranga"},  
8962: {id : '7272', geocod : '3516705', nome : "Jafa"},  
8963: {id : '3823', geocod : '3524709', nome : "Jaguariúna"},  
8964: {id : '3424', geocod : '3524808', nome : "Jales"},  
8965: {id : '7283', geocod : '3514403', nome : "Jamaica"},  
8966: {id : '3969', geocod : '3524907', nome : "Jambeiro"},  
8967: {id : '3797', geocod : '3525003', nome : "Jandira"},  
8968: {id : '7293', geocod : '3550308', nome : "Jaraguá"},  
8969: {id : '7297', geocod : '3505708', nome : "Jardim Belval"},  
8970: {id : '7299', geocod : '3518800', nome : "Jardim Presidente Dutra"},  
8971: {id : '7300', geocod : '3543303', nome : "Jardim Santa Luzia"},  
8972: {id : '7301', geocod : '3505708', nome : "Jardim Silveira"},  
8973: {id : '3837', geocod : '3525102', nome : "Jardinópolis"},  
8974: {id : '3809', geocod : '3525201', nome : "Jarinu"},  
8975: {id : '7305', geocod : '3501103', nome : "Jatobá"},  
8976: {id : '3566', geocod : '3525300', nome : "Jaú"},  
8977: {id : '3948', geocod : '3525409', nome : "Jeriquara"},  
8978: {id : '3877', geocod : '3525508', nome : "Joanópolis"},  
8979: {id : '3350', geocod : '3525607', nome : "João Ramalho"},  
8980: {id : '7319', geocod : '3509502', nome : "Joaquim Egídio"},  
8981: {id : '7323', geocod : '3509205', nome : "Jordanésia"},  
8982: {id : '3540', geocod : '3525706', nome : "José Bonifácio"},  
8983: {id : '7350', geocod : '3519006', nome : "Juliânia"},  
8984: {id : '3530', geocod : '3525805', nome : "Júlio Mesquita"},  
8985: {id : '3764', geocod : '3525854', nome : "Jumirim"},  
8986: {id : '3805', geocod : '3525904', nome : "Jundiaí"},  
8987: {id : '7354', geocod : '3530607', nome : "Jundiapeba"},  
8988: {id : '7357', geocod : '3531407', nome : "Junqueira"},  
8989: {id : '3138', geocod : '3526001', nome : "Junqueirópolis"},  
8990: {id : '5068', geocod : '3526100', nome : "Juquiá"},  
8991: {id : '7359', geocod : '3512308', nome : "Juquiratiba"},  
8992: {id : '3781', geocod : '3526209', nome : "Juquitiba"},  
8993: {id : '7364', geocod : '3517109', nome : "Juritis"},  
8994: {id : '7365', geocod : '3525102', nome : "Jurucê"},  
8995: {id : '7368', geocod : '3535002', nome : "Jurupeba"},  
8996: {id : '7369', geocod : '3553708', nome : "Jurupema"},  
8997: {id : '7489', geocod : '3529005', nome : "Lácio"},  
8998: {id : '7390', geocod : '3534609', nome : "Lagoa Azul"},  
8999: {id : '7394', geocod : '3510807', nome : "Lagoa Branca"},  
9000: {id : '3977', geocod : '3526308', nome : "Lagoinha"},  
9001: {id : '3740', geocod : '3526407', nome : "Laranjal Paulista"},  
9002: {id : '7443', geocod : '3526407', nome : "Laras"},  
9003: {id : '7445', geocod : '3511904', nome : "Lauro Penteado"},  
9004: {id : '3370', geocod : '3526506', nome : "Lavínia"},  
9005: {id : '4061', geocod : '3526605', nome : "Lavrinhas"},  
9006: {id : '3830', geocod : '3526704', nome : "Leme"},  
9007: {id : '3552', geocod : '3526803', nome : "Lençóis Paulista"},  
9008: {id : '3693', geocod : '3526902', nome : "Limeira"},  
9009: {id : '3884', geocod : '3527009', nome : "Lindóia"},  
9010: {id : '3533', geocod : '3527108', nome : "Lins"},  
9011: {id : '7466', geocod : '3523503', nome : "Lobo"},  
9012: {id : '4008', geocod : '3527207', nome : "Lorena"},  
9013: {id : '3635', geocod : '3527256', nome : "Lourdes"},  
9014: {id : '3796', geocod : '3527306', nome : "Louveira"},  
9015: {id : '3365', geocod : '3527405', nome : "Lucélia"},  
9016: {id : '3508', geocod : '3527504', nome : "Lucianópolis"},  
9017: {id : '3840', geocod : '3527603', nome : "Luís Antônio"},  
9018: {id : '3515', geocod : '3527702', nome : "Luiziânia"},  
9019: {id : '3490', geocod : '3527801', nome : "Lupércio"},  
9020: {id : '7486', geocod : '3524303', nome : "Lusitânia"},  
9021: {id : '3353', geocod : '3527900', nome : "Lutécia"},  
9022: {id : '3563', geocod : '3528007', nome : "Macatuba"},  
9023: {id : '3642', geocod : '3528106', nome : "Macaubal"},  
9024: {id : '3651', geocod : '3528205', nome : "Macedônia"},  
9025: {id : '7500', geocod : '3517000', nome : "Macucos"},  
9026: {id : '3640', geocod : '3528304', nome : "Magda"},  
9027: {id : '7504', geocod : '3550605', nome : "Mailasqui"},  
9028: {id : '3789', geocod : '3528403', nome : "Mairinque"},  
9029: {id : '3728', geocod : '3528502', nome : "Mairiporã"},  
9030: {id : '3499', geocod : '3528601', nome : "Manduri"},  
9031: {id : '7519', geocod : '3533007', nome : "Mangaratu"},  
9032: {id : '3295', geocod : '3528700', nome : "Marabá Paulista"},  
9033: {id : '3345', geocod : '3528809', nome : "Maracaí"},  
9034: {id : '3581', geocod : '3528858', nome : "Marapoama"},  
9035: {id : '7557', geocod : '3531506', nome : "Marcondésia"},  
9036: {id : '7561', geocod : '3550704', nome : "Maresias"},  
9037: {id : '3356', geocod : '3528908', nome : "Mariápolis"},  
9038: {id : '3491', geocod : '3529005', nome : "Marília"},  
9039: {id : '3414', geocod : '3529104', nome : "Marinópolis"},  
9040: {id : '7574', geocod : '3526407', nome : "Maristela"},  
9041: {id : '7584', geocod : '3530805', nome : "Martim Francisco"},  
9042: {id : '7585', geocod : '3530706', nome : "Martinho Prado Júnior"},  
9043: {id : '3339', geocod : '3529203', nome : "Martinópolis"},  
9044: {id : '3599', geocod : '3529302', nome : "Matão"},  
9045: {id : '3815', geocod : '3529401', nome : "Mauá"},  
9046: {id : '3542', geocod : '3529500', nome : "Mendonça"},  
9047: {id : '3648', geocod : '3529609', nome : "Meridiano"},  
9048: {id : '2493', geocod : '3529658', nome : "Mesópolis"},  
9049: {id : '3688', geocod : '3529708', nome : "Miguelópolis"},  
9050: {id : '3570', geocod : '3529807', nome : "Mineiros do Tietê"},  
9051: {id : '2546', geocod : '3530003', nome : "Mira Estrela"},  
9052: {id : '5069', geocod : '3529906', nome : "Miracatu"},  
9053: {id : '7643', geocod : '3532504', nome : "Miraluz"},  
9054: {id : '3369', geocod : '3530102', nome : "Mirandópolis"},  
9055: {id : '3294', geocod : '3530201', nome : "Mirante do Paranapanema"},  
9056: {id : '3664', geocod : '3530300', nome : "Mirassol"},  
9057: {id : '3666', geocod : '3530409', nome : "Mirassolândia"},  
9058: {id : '3908', geocod : '3530508', nome : "Mococa"},  
9059: {id : '3865', geocod : '3530706', nome : "Mogi Guaçu"},  
9060: {id : '3736', geocod : '3530607', nome : "Moji das Cruzes"},  
9061: {id : '3864', geocod : '3530805', nome : "Moji-Mirim"},  
9062: {id : '3750', geocod : '3530904', nome : "Mombuca"},  
9063: {id : '3638', geocod : '3531001', nome : "Monções"},  
9064: {id : '5074', geocod : '3531100', nome : "Mongaguá"},  
9065: {id : '7676', geocod : '3541406', nome : "Montalvão"},  
9066: {id : '3861', geocod : '3531209', nome : "Monte Alegre do Sul"},  
9067: {id : '3603', geocod : '3531308', nome : "Monte Alto"},  
9068: {id : '3657', geocod : '3531407', nome : "Monte Aprazível"},  
9069: {id : '3677', geocod : '3531506', nome : "Monte Azul Paulista"},  
9070: {id : '3317', geocod : '3531605', nome : "Monte Castelo"},  
9071: {id : '3754', geocod : '3531803', nome : "Monte Mor"},  
9072: {id : '7705', geocod : '3509304', nome : "Monte Verde Paulista"},  
9073: {id : '3983', geocod : '3531704', nome : "Monteiro Lobato"},  
9074: {id : '7713', geocod : '3538006', nome : "Moreira César"},  
9075: {id : '5518', geocod : '3531902', nome : "Morro Agudo"},  
9076: {id : '7726', geocod : '3522307', nome : "Morro do Alto"},  
9077: {id : '3856', geocod : '3532009', nome : "Morungaba"},  
9078: {id : '7733', geocod : '3531209', nome : "Mostardas"},  
9079: {id : '3628', geocod : '3532058', nome : "Motuca"},  
9080: {id : '7734', geocod : '3528908', nome : "Mourão"},  
9081: {id : '3367', geocod : '3532108', nome : "Murutinga do Sul"},  
9082: {id : '3322', geocod : '3532157', nome : "Nantes"},  
9083: {id : '3309', geocod : '3532207', nome : "Narandiba"},  
9084: {id : '3974', geocod : '3532306', nome : "Natividade da Serra"},  
9085: {id : '3737', geocod : '3532405', nome : "Nazaré Paulista"},  
9086: {id : '3658', geocod : '3532504', nome : "Neves Paulista"},  
9087: {id : '3645', geocod : '3532603', nome : "Nhandeara"},  
9088: {id : '3656', geocod : '3532702', nome : "Nipoã"},  
9089: {id : '7781', geocod : '3504305', nome : "Nogueira"},  
9090: {id : '7801', geocod : '3545001', nome : "Nossa Senhora do Remédio"},  
9091: {id : '7802', geocod : '3510005', nome : "Nova Alexandria"},  
9092: {id : '3545', geocod : '3532801', nome : "Nova Aliança"},  
9093: {id : '7807', geocod : '3522703', nome : "Nova América"},  
9094: {id : '7809', geocod : '3509502', nome : "Nova Aparecida"},  
9095: {id : '5054', geocod : '3532827', nome : "Nova Campina"},  
9096: {id : '3401', geocod : '3532843', nome : "Nova Canaã Paulista"},  
9097: {id : '7816', geocod : '3521903', nome : "Nova Cardoso"},  
9098: {id : '3410', geocod : '3532868', nome : "Nova Castilho"},  
9099: {id : '3594', geocod : '3532900', nome : "Nova Europa"},  
9100: {id : '3669', geocod : '3533007', nome : "Nova Granada"},  
9101: {id : '3143', geocod : '3533106', nome : "Nova Guataporanga"},  
9102: {id : '3318', geocod : '3533205', nome : "Nova Independência"},  
9103: {id : '7828', geocod : '3532801', nome : "Nova Itapirema"},  
9104: {id : '3636', geocod : '3533304', nome : "Nova Luzitânia"},  
9105: {id : '3694', geocod : '3533403', nome : "Nova Odessa"},  
9106: {id : '7839', geocod : '3541208', nome : "Nova Pátria"},  
9107: {id : '7844', geocod : '3552403', nome : "Nova Veneza"},  
9108: {id : '3589', geocod : '3533254', nome : "Novais"},  
9109: {id : '7851', geocod : '3540002', nome : "Novo Cravinhos"},  
9110: {id : '3580', geocod : '3533502', nome : "Novo Horizonte"},  
9111: {id : '3934', geocod : '3533601', nome : "Nuporanga"},  
9112: {id : '7904', geocod : '3555109', nome : "Oásis"},  
9113: {id : '3489', geocod : '3533700', nome : "Ocauçu"},  
9114: {id : '3500', geocod : '3533809', nome : "Óleo"},  
9115: {id : '3621', geocod : '3533908', nome : "Olímpia"},  
9116: {id : '7882', geocod : '3533007', nome : "Onda Branca"},  
9117: {id : '3668', geocod : '3534005', nome : "Onda Verde"},  
9118: {id : '3487', geocod : '3534104', nome : "Oriente"},  
9119: {id : '3673', geocod : '3534203', nome : "Orindiúva"},  
9120: {id : '3701', geocod : '3534302', nome : "Orlândia"},  
9121: {id : '3802', geocod : '3534401', nome : "Osasco"},  
9122: {id : '3485', geocod : '3534500', nome : "Oscar Bressane"},  
9123: {id : '3363', geocod : '3534609', nome : "Osvaldo Cruz"},  
9124: {id : '3481', geocod : '3534708', nome : "Ourinhos"},  
9125: {id : '7896', geocod : '3552502', nome : "Ouro Fino Paulista"},  
9126: {id : '3141', geocod : '3534807', nome : "Ouro Verde"},  
9127: {id : '3432', geocod : '3534757', nome : "Ouroeste"},  
9128: {id : '3358', geocod : '3534906', nome : "Pacaembu"},  
9129: {id : '7919', geocod : '3529005', nome : "Padre Nóbrega"},  
9130: {id : '3672', geocod : '3535002', nome : "Palestina"},  
9131: {id : '3592', geocod : '3535101', nome : "Palmares Paulista"},  
9132: {id : '3415', geocod : '3535200', nome : "Palmeira d'Oeste"},  
9133: {id : '7947', geocod : '3552502', nome : "Palmeiras de São Paulo"},  
9134: {id : '3478', geocod : '3535309', nome : "Palmital"},  
9135: {id : '3139', geocod : '3535408', nome : "Panorama"},  
9136: {id : '3352', geocod : '3535507', nome : "Paraguaçu Paulista"},  
9137: {id : '3968', geocod : '3535606', nome : "Paraibuna"},  
9138: {id : '3593', geocod : '3535705', nome : "Paraíso"},  
9139: {id : '7966', geocod : '3511706', nome : "Paraisolândia"},  
9140: {id : '7969', geocod : '3520400', nome : "Paranabi"},  
9141: {id : '3326', geocod : '3535804', nome : "Paranapanema"},  
9142: {id : '7971', geocod : '3547809', nome : "Paranapiacaba"},  
9143: {id : '3428', geocod : '3535903', nome : "Paranapuã"},  
9144: {id : '3373', geocod : '3536000', nome : "Parapuã"},  
9145: {id : '3330', geocod : '3536109', nome : "Pardinho"},  
9146: {id : '7988', geocod : '3550308', nome : "Parelheiros"},  
9147: {id : '5063', geocod : '3536208', nome : "Pariquera-Açu"},  
9148: {id : '3653', geocod : '3536257', nome : "Parisi"},  
9149: {id : '7990', geocod : '3555000', nome : "Parnaso"},  
9150: {id : '7993', geocod : '3524402', nome : "Parque Meia Lua"},  
9151: {id : '7994', geocod : '3519709', nome : "Paruru"},  
9152: {id : '3945', geocod : '3536307', nome : "Patrocínio Paulista"},  
9153: {id : '3140', geocod : '3536406', nome : "Paulicéia"},  
9154: {id : '3821', geocod : '3536505', nome : "Paulínia"},  
9155: {id : '3502', geocod : '3536570', nome : "Paulistânia"},  
9156: {id : '3674', geocod : '3536604', nome : "Paulo de Faria"},  
9157: {id : '8040', geocod : '3540002', nome : "Paulópolis"},  
9158: {id : '3556', geocod : '3536703', nome : "Pederneiras"},  
9159: {id : '3874', geocod : '3536802', nome : "Pedra Bela"},  
9160: {id : '8045', geocod : '3523206', nome : "Pedra Branca de Itararé"},  
9161: {id : '3650', geocod : '3536901', nome : "Pedranópolis"},  
9162: {id : '3949', geocod : '3537008', nome : "Pedregulho"},  
9163: {id : '3858', geocod : '3537107', nome : "Pedreira"},  
9164: {id : '3342', geocod : '3537156', nome : "Pedrinhas Paulista"},  
9165: {id : '8060', geocod : '3529906', nome : "Pedro Barros"},  
9166: {id : '5070', geocod : '3537206', nome : "Pedro de Toledo"},  
9167: {id : '3523', geocod : '3537305', nome : "Penápolis"},  
9168: {id : '3396', geocod : '3537404', nome : "Pereira Barreto"},  
9169: {id : '3763', geocod : '3537503', nome : "Pereiras"},  
9170: {id : '5072', geocod : '3537602', nome : "Peruíbe"},  
9171: {id : '8089', geocod : '3550308', nome : "Perus"},  
9172: {id : '3376', geocod : '3537701', nome : "Piacatu"},  
9173: {id : '8101', geocod : '3555406', nome : "Picinguaba"},  
9174: {id : '3769', geocod : '3537800', nome : "Piedade"},  
9175: {id : '3757', geocod : '3537909', nome : "Pilar do Sul"},  
9176: {id : '4002', geocod : '3538006', nome : "Pindamonhangaba"},  
9177: {id : '3588', geocod : '3538105', nome : "Pindorama"},  
9178: {id : '3873', geocod : '3538204', nome : "Pinhalzinho"},  
9179: {id : '8130', geocod : '3526605', nome : "Pinheiros"},  
9180: {id : '8141', geocod : '3517703', nome : "Pioneiros"},  
9181: {id : '3225', geocod : '3538303', nome : "Piquerobi"},  
9182: {id : '4009', geocod : '3538501', nome : "Piquete"},  
9183: {id : '3876', geocod : '3538600', nome : "Piracaia"},  
9184: {id : '3751', geocod : '3538709', nome : "Piracicaba"},  
9185: {id : '3463', geocod : '3538808', nome : "Piraju"},  
9186: {id : '3535', geocod : '3538907', nome : "Pirajuí"},  
9187: {id : '8149', geocod : '3502309', nome : "Pirambóia"},  
9188: {id : '3605', geocod : '3539004', nome : "Pirangi"},  
9189: {id : '3793', geocod : '3539103', nome : "Pirapora do Bom Jesus"},  
9190: {id : '3312', geocod : '3539202', nome : "Pirapozinho"},  
9191: {id : '3842', geocod : '3539301', nome : "Pirassununga"},  
9192: {id : '3553', geocod : '3539400', nome : "Piratininga"},  
9193: {id : '3634', geocod : '3539509', nome : "Pitangueiras"},  
9194: {id : '3529', geocod : '3539608', nome : "Planalto"},  
9195: {id : '8179', geocod : '3554300', nome : "Planalto do Sul"},  
9196: {id : '3479', geocod : '3539707', nome : "Platina"},  
9197: {id : '3732', geocod : '3539806', nome : "Poá"},  
9198: {id : '3644', geocod : '3539905', nome : "Poloni"},  
9199: {id : '8188', geocod : '3509205', nome : "Polvilho"},  
9200: {id : '3488', geocod : '3540002', nome : "Pompéia"},  
9201: {id : '3537', geocod : '3540101', nome : "Pongaí"},  
9202: {id : '5519', geocod : '3540200', nome : "Pontal"},  
9203: {id : '3423', geocod : '3540259', nome : "Pontalinda"},  
9204: {id : '3671', geocod : '3540309', nome : "Pontes Gestal"},  
9205: {id : '2494', geocod : '3540408', nome : "Populina"},  
9206: {id : '3614', geocod : '3540507', nome : "Porangaba"},  
9207: {id : '8215', geocod : '3510302', nome : "Porto"},  
9208: {id : '3774', geocod : '3540606', nome : "Porto Feliz"},  
9209: {id : '3839', geocod : '3540705', nome : "Porto Ferreira"},  
9210: {id : '8223', geocod : '3510500', nome : "Porto Novo"},  
9211: {id : '4005', geocod : '3540754', nome : "Potim"},  
9212: {id : '3547', geocod : '3540804', nome : "Potirendaba"},  
9213: {id : '8243', geocod : '3525300', nome : "Potunduva"},  
9214: {id : '3360', geocod : '3540853', nome : "Pracinha"},  
9215: {id : '8266', geocod : '3538907', nome : "Pradínia"},  
9216: {id : '3632', geocod : '3540903', nome : "Pradópolis"},  
9217: {id : '5075', geocod : '3541000', nome : "Praia Grande"},  
9218: {id : '3559', geocod : '3541059', nome : "Pratânia"},  
9219: {id : '3512', geocod : '3541109', nome : "Presidente Alves"},  
9220: {id : '3226', geocod : '3541208', nome : "Presidente Bernardes"},  
9221: {id : '3136', geocod : '3541307', nome : "Presidente Epitácio"},  
9222: {id : '3134', geocod : '3541406', nome : "Presidente Prudente"},  
9223: {id : '3223', geocod : '3541505', nome : "Presidente Venceslau"},  
9224: {id : '3520', geocod : '3541604', nome : "Promissão"},  
9225: {id : '8293', geocod : '3516903', nome : "Prudêncio e Morais"},  
9226: {id : '3615', geocod : '3541653', nome : "Quadra"},  
9227: {id : '3351', geocod : '3541703', nome : "Quatá"},  
9228: {id : '3514', geocod : '3541802', nome : "Queiroz"},  
9229: {id : '4072', geocod : '3541901', nome : "Queluz"},  
9230: {id : '3486', geocod : '3542008', nome : "Quintana"},  
9231: {id : '8328', geocod : '3554102', nome : "Quiririm"},  
9232: {id : '3748', geocod : '3542107', nome : "Rafard"},  
9233: {id : '3340', geocod : '3542206', nome : "Rancharia"},  
9234: {id : '8344', geocod : '3522307', nome : "Rechan"},  
9235: {id : '3975', geocod : '3542305', nome : "Redenção da Serra"},  
9236: {id : '3337', geocod : '3542404', nome : "Regente Feijó"},  
9237: {id : '3574', geocod : '3542503', nome : "Reginópolis"},  
9238: {id : '5065', geocod : '3542602', nome : "Registro"},  
9239: {id : '3943', geocod : '3542701', nome : "Restinga"},  
9240: {id : '8362', geocod : '3548708', nome : "Riacho Grande"},  
9241: {id : '5048', geocod : '3542800', nome : "Ribeira"},  
9242: {id : '3475', geocod : '3542909', nome : "Ribeirão Bonito"},  
9243: {id : '5055', geocod : '3543006', nome : "Ribeirão Branco"},  
9244: {id : '3942', geocod : '3543105', nome : "Ribeirão Corrente"},  
9245: {id : '3483', geocod : '3543204', nome : "Ribeirão do Sul"},  
9246: {id : '3227', geocod : '3543238', nome : "Ribeirão dos Índios"},  
9247: {id : '4896', geocod : '3543253', nome : "Ribeirão Grande"},  
9248: {id : '3816', geocod : '3543303', nome : "Ribeirão Pires"},  
9249: {id : '3836', geocod : '3543402', nome : "Ribeirão Preto"},  
9250: {id : '8375', geocod : '3518206', nome : "Ribeiro do Vale"},  
9251: {id : '8376', geocod : '3533908', nome : "Ribeiro dos Santos"},  
9252: {id : '3952', geocod : '3543600', nome : "Rifaina"},  
9253: {id : '3630', geocod : '3543709', nome : "Rincão"},  
9254: {id : '3375', geocod : '3543808', nome : "Rinópolis"},  
9255: {id : '3698', geocod : '3543907', nome : "Rio Claro"},  
9256: {id : '3747', geocod : '3544004', nome : "Rio das Pedras"},  
9257: {id : '3820', geocod : '3544103', nome : "Rio Grande da Serra"},  
9258: {id : '2547', geocod : '3544202', nome : "Riolândia"},  
9259: {id : '3437', geocod : '3543501', nome : "Riversul"},  
9260: {id : '8441', geocod : '3538105', nome : "Roberto"},  
9261: {id : '8455', geocod : '3529005', nome : "Rosália"},  
9262: {id : '3192', geocod : '3544251', nome : "Rosana"},  
9263: {id : '4004', geocod : '3544301', nome : "Roseira"},  
9264: {id : '3384', geocod : '3544400', nome : "Rubiácea"},  
9265: {id : '8464', geocod : '3507506', nome : "Rubião Júnior"},  
9266: {id : '3405', geocod : '3544509', nome : "Rubinéia"},  
9267: {id : '8465', geocod : '3530300', nome : "Ruilândia"},  
9268: {id : '8466', geocod : '3530607', nome : "Sabaúna"},  
9269: {id : '3538', geocod : '3544608', nome : "Sabino"},  
9270: {id : '3361', geocod : '3544707', nome : "Sagres"},  
9271: {id : '3543', geocod : '3544806', nome : "Sales"},  
9272: {id : '3702', geocod : '3544905', nome : "Sales Oliveira"},  
9273: {id : '3967', geocod : '3545001', nome : "Salesópolis"},  
9274: {id : '3366', geocod : '3545100', nome : "Salmourão"},  
9275: {id : '3746', geocod : '3545159', nome : "Saltinho"},  
9276: {id : '3779', geocod : '3545209', nome : "Salto"},  
9277: {id : '3768', geocod : '3545308', nome : "Salto de Pirapora"},  
9278: {id : '3480', geocod : '3545407', nome : "Salto Grande"},  
9279: {id : '3307', geocod : '3545506', nome : "Sandovalina"},  
9280: {id : '3586', geocod : '3545605', nome : "Santa Adélia"},  
9281: {id : '3422', geocod : '3545704', nome : "Santa Albertina"},  
9282: {id : '8496', geocod : '3517000', nome : "Santa América"},  
9283: {id : '3691', geocod : '3545803', nome : "Santa Barbara D'Oeste"},  
9284: {id : '3964', geocod : '3546009', nome : "Santa Branca"},  
9285: {id : '3406', geocod : '3546108', nome : "Santa Clara d'Oeste"},  
9286: {id : '3827', geocod : '3546207', nome : "Santa Cruz da Conceição"},  
9287: {id : '3850', geocod : '3546256', nome : "Santa Cruz da Esperança"},  
9288: {id : '8508', geocod : '3547502', nome : "Santa Cruz da Estrel"},  
9289: {id : '3843', geocod : '3546306', nome : "Santa Cruz das Palmeiras"},  
9290: {id : '3496', geocod : '3546405', nome : "Santa Cruz do Rio Pardo"},  
9291: {id : '8514', geocod : '3523206', nome : "Santa Cruz dos Lopes"},  
9292: {id : '3607', geocod : '3546504', nome : "Santa Ernestina"},  
9293: {id : '8521', geocod : '3548906', nome : "Santa Eudóxia"},  
9294: {id : '3402', geocod : '3546603', nome : "Santa Fé do Sul"},  
9295: {id : '3696', geocod : '3546702', nome : "Santa Gertrudes"},  
9296: {id : '3738', geocod : '3546801', nome : "Santa Isabel"},  
9297: {id : '8535', geocod : '3536901', nome : "Santa Isabel do Marinheiro"},  
9298: {id : '3629', geocod : '3546900', nome : "Santa Lúcia"},  
9299: {id : '8557', geocod : '3515707', nome : "Santa Margarida Paulista"},  
9300: {id : '3472', geocod : '3547007', nome : "Santa Maria da Serra"},  
9301: {id : '8564', geocod : '3541604', nome : "Santa Maria do Gurupi"},  
9302: {id : '3142', geocod : '3547106', nome : "Santa Mercedes"},  
9303: {id : '3841', geocod : '3547502', nome : "Santa Rita do Passa Quatro"},  
9304: {id : '8578', geocod : '3529906', nome : "Santa Rita do Ribeira"},  
9305: {id : '3420', geocod : '3547403', nome : "Santa Rita d'Oeste"},  
9306: {id : '3849', geocod : '3547601', nome : "Santa Rosa de Viterbo"},  
9307: {id : '3418', geocod : '3547650', nome : "Santa Salete"},  
9308: {id : '8588', geocod : '3538709', nome : "Santa Teresinha de Piracicab"},  
9309: {id : '3416', geocod : '3547205', nome : "Santana da Ponte Pensa"},  
9310: {id : '3800', geocod : '3547304', nome : "Santana de Parnaíba"},  
9311: {id : '8618', geocod : '3536703', nome : "Santelmo"},  
9312: {id : '3224', geocod : '3547700', nome : "Santo Anastácio"},  
9313: {id : '3813', geocod : '3547809', nome : "Santo André"},  
9314: {id : '3910', geocod : '3547908', nome : "Santo Antônio da Alegria"},  
9315: {id : '8642', geocod : '3538907', nome : "Santo Antônio da Estiva"},  
9316: {id : '8649', geocod : '3510005', nome : "Santo Antônio de Paranapanem"},  
9317: {id : '3859', geocod : '3548005', nome : "Santo Antônio de Posse"},  
9318: {id : '3409', geocod : '3548054', nome : "Santo Antônio do Aracanguá"},  
9319: {id : '3870', geocod : '3548104', nome : "Santo Antônio do Jardim"},  
9320: {id : '3984', geocod : '3548203', nome : "Santo Antônio do Pinhal"},  
9321: {id : '8640', geocod : '3515707', nome : "Santo Antônio Paulista"},  
9322: {id : '3230', geocod : '3548302', nome : "Santo Expedito"},  
9323: {id : '3380', geocod : '3548401', nome : "Santópolis do Aguapeí"},  
9324: {id : '3818', geocod : '3548500', nome : "Santos"},  
9325: {id : '8816', geocod : '3524105', nome : "São Benedito da Cachoeirinha"},  
9326: {id : '8817', geocod : '3530508', nome : "São Benedito das Areias"},  
9327: {id : '3988', geocod : '3548609', nome : "São Bento do Sapucaí"},  
9328: {id : '3812', geocod : '3548708', nome : "São Bernardo do Campo"},  
9329: {id : '8823', geocod : '3528601', nome : "São Berto"},  
9330: {id : '3814', geocod : '3548807', nome : "São Caetano do Sul"},  
9331: {id : '3832', geocod : '3548906', nome : "São Carlos"},  
9332: {id : '3417', geocod : '3549003', nome : "São Francisco"},  
9333: {id : '8851', geocod : '3550704', nome : "São Francisco da Praia"},  
9334: {id : '8849', geocod : '3549904', nome : "São Francisco Xavier"},  
9335: {id : '3901', geocod : '3549102', nome : "São João da Boa Vista"},  
9336: {id : '3426', geocod : '3549201', nome : "São João das Duas Pontes"},  
9337: {id : '3413', geocod : '3549250', nome : "São João de Iracema"},  
9338: {id : '8957', geocod : '3556008', nome : "São João de Itaguaçu"},  
9339: {id : '8964', geocod : '3510708', nome : "São João do Marinheiro"},  
9340: {id : '3314', geocod : '3549300', nome : "São João do Pau d'Alho"},  
9341: {id : '8946', geocod : '3550605', nome : "São João Novo"},  
9342: {id : '3935', geocod : '3549409', nome : "São Joaquim da Barra"},  
9343: {id : '3941', geocod : '3549508', nome : "São José da Bela Vista"},  
9344: {id : '8903', geocod : '3528809', nome : "São José das Laranjeiras"},  
9345: {id : '4064', geocod : '3549607', nome : "São José do Barreiro"},  
9346: {id : '3905', geocod : '3549706', nome : "São José do Rio Pardo"},  
9347: {id : '3665', geocod : '3549805', nome : "São José do Rio Preto"},  
9348: {id : '3966', geocod : '3549904', nome : "São José dos Campos"},  
9349: {id : '3782', geocod : '3549953', nome : "São Lourenço da Serra"},  
9350: {id : '8977', geocod : '3529302', nome : "São Lourenço do Turvo"},  
9351: {id : '3976', geocod : '3550001', nome : "São Luís do Paraitinga"},  
9352: {id : '3561', geocod : '3550100', nome : "São Manuel"},  
9353: {id : '8995', geocod : '3501103', nome : "São Martinho dOeste"},  
9354: {id : '8997', geocod : '3550308', nome : "São Mateus"},  
9355: {id : '9000', geocod : '3550308', nome : "São Miguel"},  
9356: {id : '3756', geocod : '3550209', nome : "São Miguel Arcanjo"},  
9357: {id : '5495', geocod : '3550308', nome : "São Paulo"},  
9358: {id : '3741', geocod : '3550407', nome : "São Pedro"},  
9359: {id : '3497', geocod : '3550506', nome : "São Pedro do Turvo"},  
9360: {id : '3790', geocod : '3550605', nome : "São Roque"},  
9361: {id : '9051', geocod : '3500402', nome : "São Roque da Fartura"},  
9362: {id : '3971', geocod : '3550704', nome : "São Sebastião"},  
9363: {id : '3904', geocod : '3550803', nome : "São Sebastião da Grama"},  
9364: {id : '9064', geocod : '3507902', nome : "São Sebastião da Serra"},  
9365: {id : '9092', geocod : '3524402', nome : "São Silvestre de Jacareí"},  
9366: {id : '3845', geocod : '3550902', nome : "São Simão"},  
9367: {id : '3817', geocod : '3551009', nome : "São Vicente"},  
9368: {id : '8686', geocod : '3535507', nome : "Sapezal"},  
9369: {id : '8688', geocod : '3550308', nome : "Sapopemba"},  
9370: {id : '3758', geocod : '3551108', nome : "Sarapuí"},  
9371: {id : '3462', geocod : '3551207', nome : "Sarutaiá"},  
9372: {id : '3646', geocod : '3551306', nome : "Sebastianópolis do Sul"},  
9373: {id : '3846', geocod : '3551405', nome : "Serra Azul"},  
9374: {id : '3862', geocod : '3551603', nome : "Serra Negra"},  
9375: {id : '3847', geocod : '3551504', nome : "Serrana"},  
9376: {id : '3834', geocod : '3551702', nome : "Sertãozinho"},  
9377: {id : '5064', geocod : '3551801', nome : "Sete Barras"},  
9378: {id : '3623', geocod : '3551900', nome : "Severínia"},  
9379: {id : '4062', geocod : '3552007', nome : "Silveiras"},  
9380: {id : '8772', geocod : '3508801', nome : "Simões"},  
9381: {id : '8770', geocod : '3557105', nome : "Simonsen"},  
9382: {id : '3875', geocod : '3552106', nome : "Socorro"},  
9383: {id : '8784', geocod : '3546405', nome : "Sodrélia"},  
9384: {id : '8786', geocod : '3541000', nome : "Solemar"},  
9385: {id : '3776', geocod : '3552205', nome : "Sorocaba"},  
9386: {id : '8796', geocod : '3509502', nome : "Souzas"},  
9387: {id : '3398', geocod : '3552304', nome : "Sud Mennucci"},  
9388: {id : '8805', geocod : '3500907', nome : "Suinana"},  
9389: {id : '3755', geocod : '3552403', nome : "Sumaré"},  
9390: {id : '8811', geocod : '3535309', nome : "Sussuí"},  
9391: {id : '3397', geocod : '3552551', nome : "Suzanápolis"},  
9392: {id : '3734', geocod : '3552502', nome : "Suzano"},  
9393: {id : '9124', geocod : '3526506', nome : "Tabajara"},  
9394: {id : '3619', geocod : '3552601', nome : "Tabapuã"},  
9395: {id : '3595', geocod : '3552700', nome : "Tabatinga"},  
9396: {id : '3788', geocod : '3552809', nome : "Taboão da Serra"},  
9397: {id : '3336', geocod : '3552908', nome : "Taciba"},  
9398: {id : '3460', geocod : '3553005', nome : "Taguaí"},  
9399: {id : '3606', geocod : '3553104', nome : "Taiaçu"},  
9400: {id : '9135', geocod : '3530607', nome : "Taiaçupeba"},  
9401: {id : '3609', geocod : '3553203', nome : "Taiúva"},  
9402: {id : '9140', geocod : '3549805', nome : "Talhado"},  
9403: {id : '3844', geocod : '3553302', nome : "Tambaú"},  
9404: {id : '3662', geocod : '3553401', nome : "Tanabi"},  
9405: {id : '9153', geocod : '3522703', nome : "Tapinas"},  
9406: {id : '3767', geocod : '3553500', nome : "Tapiraí"},  
9407: {id : '3914', geocod : '3553609', nome : "Tapiratiba"},  
9408: {id : '3610', geocod : '3553658', nome : "Taquaral"},  
9409: {id : '3602', geocod : '3553708', nome : "Taquaritinga"},  
9410: {id : '3466', geocod : '3553807', nome : "Taquarituba"},  
9411: {id : '3324', geocod : '3553856', nome : "Taquarivaí"},  
9412: {id : '3310', geocod : '3553906', nome : "Tarabai"},  
9413: {id : '3347', geocod : '3553955', nome : "Tarumã"},  
9414: {id : '3760', geocod : '3554003', nome : "Tatuí"},  
9415: {id : '4000', geocod : '3554102', nome : "Taubaté"},  
9416: {id : '9194', geocod : '3529203', nome : "Teçaindá"},  
9417: {id : '3461', geocod : '3554201', nome : "Tejupá"},  
9418: {id : '3293', geocod : '3554300', nome : "Teodoro Sampaio"},  
9419: {id : '9188', geocod : '3519402', nome : "Termas de Ibirá"},  
9420: {id : '9192', geocod : '3547106', nome : "Terra Nova dOeste"},  
9421: {id : '3682', geocod : '3554409', nome : "Terra Roxa"},  
9422: {id : '9196', geocod : '3506003', nome : "Tibiriçá"},  
9423: {id : '9197', geocod : '3538808', nome : "Tibiriçá do Paranapa"},  
9424: {id : '3766', geocod : '3554508', nome : "Tietê"},  
9425: {id : '3442', geocod : '3554607', nome : "Timburi"},  
9426: {id : '3613', geocod : '3554656', nome : "Torre de Pedra"},  
9427: {id : '3473', geocod : '3554706', nome : "Torrinha"},  
9428: {id : '3573', geocod : '3554755', nome : "Trabiju"},  
9429: {id : '4001', geocod : '3554805', nome : "Tremembé"},  
9430: {id : '3403', geocod : '3554904', nome : "Três Fronteiras"},  
9431: {id : '9253', geocod : '3531209', nome : "Três Pontes"},  
9432: {id : '9243', geocod : '3530102', nome : "Trs Alianças"},  
9433: {id : '3857', geocod : '3554953', nome : "Tuiuti"},  
9434: {id : '9260', geocod : '3512209', nome : "Tujuguaba"},  
9435: {id : '3377', geocod : '3555000', nome : "Tupã"},  
9436: {id : '9266', geocod : '3538709', nome : "Tupi"},  
9437: {id : '3316', geocod : '3555109', nome : "Tupi Paulista"},  
9438: {id : '9271', geocod : '3521705', nome : "Turiba do Sul"},  
9439: {id : '3637', geocod : '3555208', nome : "Turiúba"},  
9440: {id : '3430', geocod : '3555307', nome : "Turmalina"},  
9441: {id : '9273', geocod : '3506102', nome : "Turvínia"},  
9442: {id : '3539', geocod : '3555356', nome : "Ubarana"},  
9443: {id : '3978', geocod : '3555406', nome : "Ubatuba"},  
9444: {id : '3504', geocod : '3555505', nome : "Ubirajara"},  
9445: {id : '3617', geocod : '3555604', nome : "Uchoa"},  
9446: {id : '3643', geocod : '3555703', nome : "União Paulista"},  
9447: {id : '9292', geocod : '3555000', nome : "Universo"},  
9448: {id : '3419', geocod : '3555802', nome : "Urânia"},  
9449: {id : '3536', geocod : '3555901', nome : "Uru"},  
9450: {id : '3546', geocod : '3556008', nome : "Urupês"},  
9451: {id : '9304', geocod : '3545605', nome : "Ururaí"},  
9452: {id : '9315', geocod : '3533502', nome : "Vale Formoso"},  
9453: {id : '3649', geocod : '3556107', nome : "Valentim Gentil"},  
9454: {id : '3854', geocod : '3556206', nome : "Valinhos"},  
9455: {id : '3371', geocod : '3556305', nome : "Valparaíso"},  
9456: {id : '9328', geocod : '3536703', nome : "Vanglória"},  
9457: {id : '3872', geocod : '3556354', nome : "Vargem"},  
9458: {id : '3902', geocod : '3556404', nome : "Vargem Grande do Sul"},  
9459: {id : '3783', geocod : '3556453', nome : "Vargem Grande Paulista"},  
9460: {id : '9336', geocod : '3555000', nome : "Varpa"},  
9461: {id : '3804', geocod : '3556503', nome : "Várzea Paulista"},  
9462: {id : '9346', geocod : '3510807', nome : "Venda Branca"},  
9463: {id : '3492', geocod : '3556602', nome : "Vera Cruz"},  
9464: {id : '9362', geocod : '3518701', nome : "Vicente de Carvalho"},  
9465: {id : '9374', geocod : '3510609', nome : "Vila Dirce"},  
9466: {id : '9381', geocod : '3548906', nome : "Vila Nery"},  
9467: {id : '9409', geocod : '3503208', nome : "Vila Xavier"},  
9468: {id : '3855', geocod : '3556701', nome : "Vinhedo"},  
9469: {id : '3681', geocod : '3556800', nome : "Viradouro"},  
9470: {id : '3604', geocod : '3556909', nome : "Vista Alegre do Alto"},  
9471: {id : '3425', geocod : '3556958', nome : "Vitória Brasil"},  
9472: {id : '9427', geocod : '3507506', nome : "Vitoriana"},  
9473: {id : '3775', geocod : '3557006', nome : "Votorantim"},  
9474: {id : '3652', geocod : '3557105', nome : "Votuporanga"},  
9475: {id : '3528', geocod : '3557154', nome : "Zacarias"}  
9476: ],  
9477: TO : [  
9478: // UF = TO  
9479: {id : '1416', geocod : '1700251', nome : "Abreulândia"},  
9480: {id : '481', geocod : '1700301', nome : "Aguiarnópolis"},  
9481: {id : '1396', geocod : '1700350', nome : "Aliança do Tocantins"},  
9482: {id : '1464', geocod : '1700400', nome : "Almas"},  
9483: {id : '2101', geocod : '1700707', nome : "Alvorada"},  
9484: {id : '426', geocod : '1701002', nome : "Ananás"},  
9485: {id : '478', geocod : '1701051', nome : "Angico"},  
9486: {id : '1471', geocod : '1701101', nome : "Aparecida do Rio Negro"},  
9487: {id : '403', geocod : '1701309', nome : "Aragominas"},  
9488: {id : '1409', geocod : '1701903', nome : "Araguacema"},  
9489: {id : '2083', geocod : '1702000', nome : "Araguaçu"},  
9490: {id : '424', geocod : '1702109', nome : "Araguaína"},  
9491: {id : '405', geocod : '1702158', nome : "Araguanã"},  
9492: {id : '428', geocod : '1702208', nome : "Araguatins"},  
9493: {id : '397', geocod : '1702307', nome : "Arapoema"},  
9494: {id : '2114', geocod : '1702406', nome : "Arraias"},  
9495: {id : '498', geocod : '1702554', nome : "Augustinópolis"},  
9496: {id : '2121', geocod : '1702703', nome : "Aurora do Tocantins"},  
9497: {id : '497', geocod : '1702901', nome : "Axixá do Tocantins"},  
9498: {id : '475', geocod : '1703008', nome : "Babaçulândia"},  
9499: {id : '398', geocod : '1703057', nome : "Bandeirantes do Tocantins"},  
9500: {id : '473', geocod : '1703073', nome : "Barra do Ouro"},  
9501: {id : '1417', geocod : '1703107', nome : "Barrolândia"},  
9502: {id : '396', geocod : '1703206', nome : "Bernardo Sayão"},  
9503: {id : '1437', geocod : '1703305', nome : "Bom Jesus do Tocantins"},  
9504: {id : '1431', geocod : '1703602', nome : "Brasilândia do Tocantins"},  
9505: {id : '1404', geocod : '1703701', nome : "Brejinho de Nazaré"},  
9506: {id : '429', geocod : '1703800', nome : "Buriti do Tocantins"},  
9507: {id : '6140', geocod : '1721000', nome : "Buritirana"},  
9508: {id : '496', geocod : '1703826', nome : "Cachoeirinha"},  
9509: {id : '487', geocod : '1703842', nome : "Campos Lindos"},  
9510: {id : '6265', geocod : '1702406', nome : "Cana Brava"},  
9511: {id : '6283', geocod : '1718907', nome : "Cangas"},  
9512: {id : '1391', geocod : '1703867', nome : "Cariri do Tocantins"},  
9513: {id : '404', geocod : '1703883', nome : "Carmolândia"},  
9514: {id : '431', geocod : '1703891', nome : "Carrasco Bonito"},  
9515: {id : '6369', geocod : '1709005', nome : "Cartucho"},  
9516: {id : '1407', geocod : '1703909', nome : "Caseara"},  
9517: {id : '1477', geocod : '1704105', nome : "Centenário"},  
9518: {id : '1461', geocod : '1705102', nome : "Chapada da Natividade"},  
9519: {id : '1411', geocod : '1704600', nome : "Chapada de Areia"},  
9520: {id : '1443', geocod : '1705508', nome : "Colinas do Tocantins"},  
9521: {id : '1424', geocod : '1716703', nome : "Colméia"},  
9522: {id : '2119', geocod : '1705557', nome : "Combinado"},  
9523: {id : '2145', geocod : '1705607', nome : "Conceição do Tocantins"},  
9524: {id : '1441', geocod : '1706001', nome : "Couto de Magalhães"},  
9525: {id : '6569', geocod : '1709005', nome : "Craolândia"},  
9526: {id : '1398', geocod : '1706100', nome : "Cristalândia"},  
9527: {id : '1397', geocod : '1706258', nome : "Crixás do Tocantins"},  
9528: {id : '477', geocod : '1706506', nome : "Darcinópolis"},  
9529: {id : '1466', geocod : '1707009', nome : "Dianópolis"},  
9530: {id : '1415', geocod : '1707108', nome : "Divinópolis do Tocantins"},  
9531: {id : '1422', geocod : '1707207', nome : "Dois Irmãos do Tocantins"},  
9532: {id : '1395', geocod : '1707306', nome : "Dueré"},  
9533: {id : '417', geocod : '1707405', nome : "Esperantina"},  
9534: {id : '1400', geocod : '1707553', nome : "Fátima"},  
9535: {id : '2102', geocod : '1707652', nome : "Figueirópolis"},  
9536: {id : '479', geocod : '1707702', nome : "Filadélfia"},  
9537: {id : '1389', geocod : '1708205', nome : "Formoso do Araguaia"},  
9538: {id : '1427', geocod : '1708254', nome : "Fortaleza do Tabocão"},  
9539: {id : '1423', geocod : '1708304', nome : "Goianorte"},  
9540: {id : '474', geocod : '1709005', nome : "Goiatins"},  
9541: {id : '1428', geocod : '1709302', nome : "Guaraí"},  
9542: {id : '1392', geocod : '1709500', nome : "Gurupi"},  
9543: {id : '1403', geocod : '1709807', nome : "Ipueiras"},  
9544: {id : '1476', geocod : '1710508', nome : "Itacajá"},  
9545: {id : '502', geocod : '1710706', nome : "Itaguatins"},  
9546: {id : '1444', geocod : '1710904', nome : "Itapiratins"},  
9547: {id : '1429', geocod : '1711100', nome : "Itaporã do Tocantins"},  
9548: {id : '2095', geocod : '1711506', nome : "Jaú do Tocantins"},  
9549: {id : '1442', geocod : '1711803', nome : "Juarina"},  
9550: {id : '1390', geocod : '1711902', nome : "Lagoa da Confusão"},  
9551: {id : '1472', geocod : '1711951', nome : "Lagoa do Tocantins"},  
9552: {id : '1419', geocod : '1712009', nome : "Lajeado"},  
9553: {id : '2120', geocod : '1712157', nome : "Lavandeira"},  
9554: {id : '1480', geocod : '1712405', nome : "Lizarda"},  
9555: {id : '495', geocod : '1712454', nome : "Luzinópolis"},  
9556: {id : '1406', geocod : '1712504', nome : "Marianópolis do Tocantins"},  
9557: {id : '1469', geocod : '1712702', nome : "Mateiros"},  
9558: {id : '501', geocod : '1712801', nome : "Maurilândia do Tocantins"},  
9559: {id : '1420', geocod : '1713205', nome : "Miracema do Tocantins"},  
9560: {id : '7647', geocod : '1709302', nome : "Mirandópolis"},  
9561: {id : '1418', geocod : '1713304', nome : "Miranorte"},  
9562: {id : '1434', geocod : '1713601', nome : "Monte do Carmo"},  
9563: {id : '1413', geocod : '1713700', nome : "Monte Santo do Tocantins"},  
9564: {id : '7725', geocod : '1718907', nome : "Morro de São João"},  
9565: {id : '402', geocod : '1713957', nome : "Muricilândia"},  
9566: {id : '7765', geocod : '1702208', nome : "Natal"},  
9567: {id : '1460', geocod : '1714203', nome : "Natividade"},  
9568: {id : '483', geocod : '1714302', nome : "Nazaré"},  
9569: {id : '399', geocod : '1714880', nome : "Nova Olinda"},  
9570: {id : '1402', geocod : '1715002', nome : "Nova Rosalândia"},  
9571: {id : '1473', geocod : '1715101', nome : "Novo Acordo"},  
9572: {id : '2118', geocod : '1715150', nome : "Novo Alegre"},  
9573: {id : '1467', geocod : '1715259', nome : "Novo Jardim"},  
9574: {id : '1401', geocod : '1715507', nome : "Oliveira de Fátima"},  
9575: {id : '5496', geocod : '1721000', nome : "Palmas"},  
9576: {id : '472', geocod : '1715705', nome : "Palmeirante"},  
9577: {id : '5498', geocod : '1713809', nome : "Palmeiras do Tocantins"},  
9578: {id : '2096', geocod : '1715754', nome : "Palmeirópolis"},  
9579: {id : '1414', geocod : '1716109', nome : "Paraíso do Tocantins"},  
9580: {id : '2139', geocod : '1716208', nome : "Paranã"},  
9581: {id : '392', geocod : '1716307', nome : "Pau D'Arco"},  
9582: {id : '1436', geocod : '1716505', nome : "Pedro Afonso"},  
9583: {id : '2103', geocod : '1716604', nome : "Peixe"},  
9584: {id : '1425', geocod : '1716653', nome : "Pequizeiro"},  
9585: {id : '1462', geocod : '1717008', nome : "Pindorama do Tocantins"},  
9586: {id : '408', geocod : '1717206', nome : "Piraquê"},  
9587: {id : '1410', geocod : '1717503', nome : "Pium"},  
9588: {id : '2148', geocod : '1717800', nome : "Ponte Alta do Bom Jesus"},  
9589: {id : '1463', geocod : '1717909', nome : "Ponte Alta do Tocantins"},  
9590: {id : '1465', geocod : '1718006', nome : "Porto Alegre do Tocantins"},  
9591: {id : '1405', geocod : '1718204', nome : "Porto Nacional"},  
9592: {id : '8224', geocod : '1716505', nome : "Porto Real"},  
9593: {id : '500', geocod : '1718303', nome : "Praia Norte"},  
9594: {id : '1430', geocod : '1718402', nome : "Presidente Kennedy"},  
9595: {id : '8294', geocod : '1714203', nome : "Príncipe"},  
9596: {id : '1412', geocod : '1718451', nome : "Pugmil"},  
9597: {id : '1478', geocod : '1718501', nome : "Recursolândia"},  
9598: {id : '425', geocod : '1718550', nome : "Riachinho"},  
9599: {id : '1468', geocod : '1718659', nome : "Rio da Conceição"},  
9600: {id : '1426', geocod : '1718709', nome : "Rio dos Bois"},  
9601: {id : '1474', geocod : '1718758', nome : "Rio Sono"},  
9602: {id : '499', geocod : '1718808', nome : "Sampaio"},  
9603: {id : '2033', geocod : '1718840', nome : "Sandolândia"},  
9604: {id : '401', geocod : '1718865', nome : "Santa Fé do Araguaia"},  
9605: {id : '1475', geocod : '1718881', nome : "Santa Maria do Tocantins"},  
9606: {id : '1399', geocod : '1718899', nome : "Santa Rita do Tocantins"},  
9607: {id : '1432', geocod : '1718907', nome : "Santa Rosa do Tocantins"},  
9608: {id : '1470', geocod : '1719004', nome : "Santa Tereza do Tocantins"},  
9609: {id : '482', geocod : '1720002', nome : "Santa Terezinha do Tocantins"},  
9610: {id : '427', geocod : '1720101', nome : "São Bento do Tocantins"},  
9611: {id : '1479', geocod : '1720150', nome : "São Félix do Tocantins"},  
9612: {id : '506', geocod : '1720200', nome : "São Miguel do Tocantins"},  
9613: {id : '2097', geocod : '1720259', nome : "São Salvador do Tocantins"},  
9614: {id : '430', geocod : '1720309', nome : "São Sebastião do Tocantins"},  
9615: {id : '1394', geocod : '1720499', nome : "São Valério da Natividade"},  
9616: {id : '1433', geocod : '1720655', nome : "Silvanópolis"},  
9617: {id : '505', geocod : '1720804', nome : "Sítio Novo do Tocantins"},  
9618: {id : '1393', geocod : '1720853', nome : "Sucupira"},  
9619: {id : '2147', geocod : '1720903', nome : "Taguatinga"},  
9620: {id : '2146', geocod : '1720937', nome : "Taipas do Tocantins"},  
9621: {id : '2041', geocod : '1720978', nome : "Talismã"},  
9622: {id : '9169', geocod : '1721000', nome : "Taquarussu do Porto"},  
9623: {id : '1421', geocod : '1721109', nome : "Tocantínia"},  
9624: {id : '485', geocod : '1721208', nome : "Tocantinópolis"},  
9625: {id : '1435', geocod : '1721257', nome : "Tupirama"},  
9626: {id : '1438', geocod : '1721307', nome : "Tupiratins"},  
9627: {id : '476', geocod : '1722081', nome : "Wanderlândia"},  
9628: {id : '407', geocod : '1722107', nome : "Xambioá"}  
9629: ]  
9630:};

## .\mobile\www\js\produtos.js

1:var lista\_produtos = [  
 2: {id : '1', nome : "Abacaxi (a Granel)"},  
 3: {id : '2', nome : "Abacaxi (Doces e Conservas)"},  
 4: {id : '3', nome : "Abacaxi (Embalado)"},  
 5: {id : '4', nome : "Abre-cartas para escritorio"},  
 6: {id : '5', nome : "Açaí (em Conserva)"},  
 7: {id : '6', nome : "Açaí (In Natura)"},  
 8: {id : '7', nome : "Acessórios de Outros Metais não Ferrosos"},  
 9: {id : '8', nome : "Acessórios e Peças para Instrumentos Musicais"},  
 10: {id : '9', nome : "Açúcar a Granel (Cristal, Demerara e Refinado)"},  
 11: {id : '10', nome : "Açúcar Embalado (Cristal, Demerara e Refinado)"},  
 12: {id : '11', nome : "Açucar Invertido (Xarope)"},  
 13: {id : '12', nome : "Adesivos e Selantes"},  
 14: {id : '13', nome : "Aditivos de Uso Industrial (Quimicos)"},  
 15: {id : '14', nome : "Adubos e Fertilizantes"},  
 16: {id : '15', nome : "Aeronaves"},  
 17: {id : '16', nome : "Agrotóxicos"},  
 18: {id : '17', nome : "Aguardente de Cana (Envasado)"},  
 19: {id : '18', nome : "Aguardente de Cana (Granéis Líquidos)"},  
 20: {id : '19', nome : "Águas Minerais"},  
 21: {id : '20', nome : "Agulhas de Costura, Croche, Trico"},  
 22: {id : '21', nome : "Álcool (Embalado)"},  
 23: {id : '22', nome : "Álcool (Etanol)"},  
 24: {id : '23', nome : "Alfinetes"},  
 25: {id : '24', nome : "Algodão (a Granel)"},  
 26: {id : '25', nome : "Algodão Beneficiado"},  
 27: {id : '26', nome : "Algodão em Caroço (a Granel)"},  
 28: {id : '27', nome : "Alho (a Granel)"},  
 29: {id : '28', nome : "Alho (Embalado)"},  
 30: {id : '29', nome : "Alicates"},  
 31: {id : '30', nome : "Alicates para Unhas"},  
 32: {id : '31', nome : "Alizares e Soleiras de Aço, Aluminio ou Ferro"},  
 33: {id : '32', nome : "Alumínio e suas Ligas"},  
 34: {id : '33', nome : "Alviões (Ferramentas)"},  
 35: {id : '34', nome : "Amendoim (a Granel)"},  
 36: {id : '35', nome : "Amendoim (Doces, Conservas e Salgados)"},  
 37: {id : '36', nome : "Amianto"},  
 38: {id : '37', nome : "Amido de Milho"},  
 39: {id : '38', nome : "Amidos e Féculas"},  
 40: {id : '39', nome : "Amônia"},  
 41: {id : '40', nome : "Ancinhos"},  
 42: {id : '41', nome : "Âncoras Metálicas (Peças e Volumes)"},  
 43: {id : '42', nome : "Andaimes tubulares de Metal"},  
 44: {id : '43', nome : "Animais Domésticos"},  
 45: {id : '44', nome : "Animais vivos (outros)"},  
 46: {id : '45', nome : "Aparelhos de barbear"},  
 47: {id : '46', nome : "Aparelhos de Som e Imagem (televisão, DVD player, blue ray)"},  
 48: {id : '47', nome : "Aparelhos e Equipamentos não-Elétricos para Instalações Térmicas"},  
 49: {id : '48', nome : "Aparelhos e Equipamentos Para Sinalização e Alarme"},  
 50: {id : '49', nome : "Aparelhos/ Instrumentos de Medida"},  
 51: {id : '50', nome : "Aparelhos/ Instrumentos Médico-Hospitalar"},  
 52: {id : '51', nome : "Aparelhos/ Instrumentos Ópticos"},  
 53: {id : '52', nome : "Aplainadeiras"},  
 54: {id : '53', nome : "Arames de Aço"},  
 55: {id : '54', nome : "Arames Farpados (aço, ferro)"},  
 56: {id : '55', nome : "Arames Lisos (aço, ferro)"},  
 57: {id : '56', nome : "Arcos de pua (ferramenta)"},  
 58: {id : '57', nome : "Ardósia"},  
 59: {id : '58', nome : "Areia (a Granel)"},  
 60: {id : '59', nome : "Areia (Embalada)"},  
 61: {id : '60', nome : "Armas brancas (martelos, os machados, etc.)"},  
 62: {id : '61', nome : "Armas e Munições ( revólveres, pistolas, rifles)"},  
 63: {id : '62', nome : "Arroz (a Granel)"},  
 64: {id : '63', nome : "Arroz Beneficiado"},  
 65: {id : '64', nome : "Arroz Beneficiado - Produtos Derivados (pré-cozido, risoto, doce, biscoito)"},  
 66: {id : '65', nome : "Arroz em Casca"},  
 67: {id : '66', nome : "Arruelas não roscados de Aço, Cobre e Ferro"},  
 68: {id : '67', nome : "Artefatos de Couro"},  
 69: {id : '68', nome : "Artefatos de Joalheria e Ourivesaria (jóias)"},  
 70: {id : '69', nome : "Artefatos de Zinco para Construção (Peças e Volumes)"},  
 71: {id : '70', nome : "Artefatos para Caça, Pesca e Esporte"},  
 72: {id : '71', nome : "Artigos de Borracha (outros)"},  
 73: {id : '72', nome : "Artigos de Funilaria e de Metal para Usos Doméstico e Pessoal"},  
 74: {id : '73', nome : "Artigos de Serralheria (exceto esquadrias)"},  
 75: {id : '74', nome : "Artigos plasticos para Escritório"},  
 76: {id : '75', nome : "Artigos plasticos para Higiene"},  
 77: {id : '76', nome : "Artigos plasticos para uso doméstico"},  
 78: {id : '77', nome : "Artigos plasticos para uso escolar"},  
 79: {id : '78', nome : "Assadeiras"},  
 80: {id : '79', nome : "Assentos plasticos"},  
 81: {id : '80', nome : "Automóveis, Camionetas e Utilitários"},  
 82: {id : '81', nome : "Aves Abatidas"},  
 83: {id : '82', nome : "Aves vivas (Outras)"},  
 84: {id : '83', nome : "Avestruz"},  
 85: {id : '84', nome : "Aviamentos Para Costura"},  
 86: {id : '85', nome : "Azulejos e Pisos"},  
 87: {id : '86', nome : "Bacias de Aluminio"},  
 88: {id : '87', nome : "Baionetas"},  
 89: {id : '88', nome : "Baixelas de Aço, Aluminio"},  
 90: {id : '89', nome : "Baixelas de Cobre, Estanho, Ferro"},  
 91: {id : '90', nome : "Balas"},  
 92: {id : '91', nome : "Balcoes e Balaustradas (material metálico)"},  
 93: {id : '92', nome : "Baldes (metal)"},  
 94: {id : '93', nome : "Banana ( a Granel)"},  
 95: {id : '94', nome : "Banana (Doces e Conservas)"},  
 96: {id : '95', nome : "Bandejas (metal)"},  
 97: {id : '96', nome : "Banheiras de ferro ou aço"},  
 98: {id : '97', nome : "Banheiras plasticas"},  
 99: {id : '98', nome : "Barcos"},  
 100: {id : '99', nome : "Barras de Aço"},  
 101: {id : '100', nome : "Barras de Aluminio"},  
 102: {id : '101', nome : "Barras de Outros Metais não Ferrosos"},  
 103: {id : '102', nome : "Barris de Metal"},  
 104: {id : '103', nome : "Barris de Plastico"},  
 105: {id : '104', nome : "Batata (Doces, Conservas e Salgados)"},  
 106: {id : '105', nome : "Batata (a Granel)"},  
 107: {id : '106', nome : "Baterias de Veículos"},  
 108: {id : '107', nome : "Bebidas Outras (Envasadas)"},  
 109: {id : '108', nome : "Bebidas Outras (Granéis Líquidos)"},  
 110: {id : '109', nome : "Bezerros ou Bezerras"},  
 111: {id : '110', nome : "Bicicletas e Triciclos não Motorizados"},  
 112: {id : '111', nome : "Metais para Bidês (metal)"},  
 113: {id : '112', nome : "Bigornas"},  
 114: {id : '113', nome : "Bisnagas flexiveis para embalagem (metal)"},  
 115: {id : '114', nome : "Blocos de ferro"},  
 116: {id : '115', nome : "Bobinas de Aço"},  
 117: {id : '116', nome : "Bodes"},  
 118: {id : '117', nome : "Bois"},  
 119: {id : '118', nome : "Bolachas e Biscoitos"},  
 120: {id : '119', nome : "Bolsas de Plastico"},  
 121: {id : '120', nome : "Bolsas e Cintos"},  
 122: {id : '121', nome : "Bovinos vivos"},  
 123: {id : '122', nome : "Braçadeiras"},  
 124: {id : '123', nome : "Brinquedos e outros Jogos Recreativos"},  
 125: {id : '124', nome : "Búfalos"},  
 126: {id : '125', nome : "Cabines, Carrocerias e Reboques"},  
 127: {id : '126', nome : "Fios, Cabos e Condutores Elétricos Isolados"},  
 128: {id : '127', nome : "Cabos para facas"},  
 129: {id : '128', nome : "Cabras"},  
 130: {id : '129', nome : "Cabritos e Cabritas"},  
 131: {id : '130', nome : "Cacau"},  
 132: {id : '131', nome : "Cachorros"},  
 133: {id : '132', nome : "Cadeados"},  
 134: {id : '133', nome : "Café em grão"},  
 135: {id : '134', nome : "Café solúvel"},  
 136: {id : '135', nome : "Café torrado e moído"},  
 137: {id : '136', nome : "Caixa de descarga de plastico"},  
 138: {id : '137', nome : "Caixas de Correio de Ferro e Aço Fundidas"},  
 139: {id : '138', nome : "Caixas de Metal"},  
 140: {id : '139', nome : "Caixas e Caixotes de Plastico"},  
 141: {id : '140', nome : "Caixas plasticas"},  
 142: {id : '141', nome : "Cajú (a Granel)"},  
 143: {id : '142', nome : "Cajú (Doces, Conservas e Salgados)"},  
 144: {id : '143', nome : "Cal"},  
 145: {id : '144', nome : "Cal hidratado"},  
 146: {id : '145', nome : "Cal Virgem"},  
 147: {id : '146', nome : "Calçados e Tênis"},  
 148: {id : '147', nome : "Calcário"},  
 149: {id : '148', nome : "Calcário Agrícola"},  
 150: {id : '149', nome : "Calcário Siderúrgico"},  
 151: {id : '150', nome : "Caldeiras"},  
 152: {id : '151', nome : "Calhas Metálicas (Peças e Volumes)"},  
 153: {id : '152', nome : "Camarões"},  
 154: {id : '153', nome : "Caminhões"},  
 155: {id : '154', nome : "Campainhas Metálicas (Peças e Volumes)"},  
 156: {id : '155', nome : "Cana-de-açúcar"},  
 157: {id : '156', nome : "Canecas de plastico"},  
 158: {id : '157', nome : "Canecas para cozinha (metal, aço, ferro, aluminio)"},  
 159: {id : '158', nome : "Canecas plasticas"},  
 160: {id : '159', nome : "Canos de Ferro, Aço"},  
 161: {id : '160', nome : "Canos e Perfis Ocos (ferro, aç, inox)"},  
 162: {id : '161', nome : "Cantoneiras de plastico"},  
 163: {id : '162', nome : "Capacetes de plastico"},  
 164: {id : '163', nome : "Caprinos"},  
 165: {id : '164', nome : "Capsulas Plasticas"},  
 166: {id : '165', nome : "Carga fracionada"},  
 167: {id : '166', nome : "Carne bovina"},  
 168: {id : '167', nome : "Carne de aves"},  
 169: {id : '168', nome : "Carne de Bode"},  
 170: {id : '169', nome : "Carne de Cabra"},  
 171: {id : '170', nome : "Carne de Ovelha"},  
 172: {id : '171', nome : "Carne de Ovinos e Caprinos"},  
 173: {id : '172', nome : "Carne suína"},  
 174: {id : '173', nome : "Carneiros Abatidos (resfriado, frigorificado, congelado)"},  
 175: {id : '174', nome : "Carnes (outras)"},  
 176: {id : '175', nome : "Carretéis de Plastico"},  
 177: {id : '177', nome : "Carrocerias para Ônibus"},  
 178: {id : '178', nome : "Cartuchos de Plastico"},  
 179: {id : '179', nome : "Carvão Mineral"},  
 180: {id : '180', nome : "Carvão vegetal a granel"},  
 181: {id : '181', nome : "Carvão vegetal embalado"},  
 182: {id : '182', nome : "Cascalho"},  
 183: {id : '183', nome : "Castanha de Caju (a Granel)"},  
 184: {id : '184', nome : "Castanha de Caju (Conservas)"},  
 185: {id : '185', nome : "Castanha de Caju (Embalada)"},  
 186: {id : '186', nome : "Castanha do Pará (a Granel)"},  
 187: {id : '187', nome : "Castanha do Pará (Conservas)"},  
 188: {id : '188', nome : "Castanha do Pará (Embalada)"},  
 189: {id : '189', nome : "Catalisadores"},  
 190: {id : '190', nome : "Cavaco de madeira"},  
 191: {id : '191', nome : "Cavalos"},  
 192: {id : '192', nome : "Cebola (a Granel)"},  
 193: {id : '193', nome : "Cebola (Conservas)"},  
 194: {id : '194', nome : "Cebola (Embalada)"},  
 195: {id : '195', nome : "Celulose"},  
 196: {id : '196', nome : "Centrífugas (Eletrodoméstico)"},  
 197: {id : '197', nome : "Cerâmicas"},  
 198: {id : '198', nome : "Cerveja e Chopp (Tanque)"},  
 199: {id : '199', nome : "Cervejas (Em Barris)"},  
 200: {id : '200', nome : "Cervejas (Envasadas)"},  
 201: {id : '201', nome : "Cestas de plastico para lixo"},  
 202: {id : '202', nome : "Chá e outras Ervas para Infusão"},  
 203: {id : '203', nome : "Chafarizes para Moinhos de Ferro e Aço Fundidas"},  
 204: {id : '204', nome : "Chapas de Aço"},  
 205: {id : '205', nome : "Chapas de Aluminio"},  
 206: {id : '206', nome : "Chapas de Granito"},  
 207: {id : '207', nome : "Chapas de Outros Metais não Ferrosos"},  
 208: {id : '208', nome : "Chapas de Plástico"},  
 209: {id : '209', nome : "Chapas e tiras, distendidas, de cobre e niquel"},  
 210: {id : '210', nome : "Chapas e tiras, distendidas, de ferro e aço"},  
 211: {id : '211', nome : "Charutos"},  
 212: {id : '212', nome : "Chassis com Motor para Automóveis, Camionetas e Utilitários"},  
 213: {id : '213', nome : "Chaves"},  
 214: {id : '214', nome : "Chaves de caixa intercambiáveis"},  
 215: {id : '215', nome : "Chaves de fenda"},  
 216: {id : '216', nome : "Chaves Metálicas (Embalados)"},  
 217: {id : '217', nome : "Chaves para porcas"},  
 218: {id : '218', nome : "Chester"},  
 219: {id : '219', nome : "Chinelos e Sandálias"},  
 220: {id : '220', nome : "Chocolates e Doces"},  
 221: {id : '221', nome : "Chopp (Em Barris)"},  
 222: {id : '222', nome : "Chumbo"},  
 223: {id : '223', nome : "Cigarros"},  
 224: {id : '224', nome : "Cimento a granel"},  
 225: {id : '225', nome : "Cimento ensacado"},  
 226: {id : '226', nome : "Escória de cimento"},  
 227: {id : '227', nome : "Cinzas de Aluminio"},  
 228: {id : '228', nome : "Cinzas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 229: {id : '229', nome : "Cisalhas para metais (ferramenta)"},  
 230: {id : '230', nome : "Clarabóias Metálicas (Peças e Volumes)"},  
 231: {id : '231', nome : "Classificadores Metálicos (Embalados)"},  
 232: {id : '232', nome : "Clinquer"},  
 233: {id : '233', nome : "Clipes Metálicos (Embalados)"},  
 234: {id : '234', nome : "Cloreto de Potássio"},  
 235: {id : '235', nome : "Cloro e Álcalis"},  
 236: {id : '236', nome : "Coalhadas"},  
 237: {id : '237', nome : "Cobalto"},  
 238: {id : '238', nome : "Cobre"},  
 239: {id : '239', nome : "Côco (a Granel)"},  
 240: {id : '240', nome : "Côco (Doces e Conservas)"},  
 241: {id : '241', nome : "Codornas"},  
 242: {id : '242', nome : "Cofres e Caixas de Segurança e Artefatos Semelhantes Metálicos (Peças e Volumes)"},  
 243: {id : '243', nome : "Colheres"},  
 244: {id : '244', nome : "Comadres de Aluminio"},  
 245: {id : '245', nome : "Computadores e Notebooks, Impressoras"},  
 246: {id : '246', nome : "Conchas para cozinha"},  
 247: {id : '247', nome : "Condensadores de máquinas à vapor"},  
 248: {id : '248', nome : "Cabos Elétricos"},  
 249: {id : '249', nome : "Conexões de Aluminio"},  
 250: {id : '250', nome : "Conexões para encanamento (metal, ferro, aço, inox)"},  
 251: {id : '251', nome : "Conexões para encanamento (plástico, pvc)"},  
 252: {id : '252', nome : "Conservas e Doces de Outros Legumes e Vegetais"},  
 253: {id : '253', nome : "Construcoes pre-fabricadas de Ferro e Aço"},  
 254: {id : '254', nome : "Construcoes pré-fabricadas de Madeira"},  
 255: {id : '255', nome : "Copa (Salsicharia)"},  
 256: {id : '256', nome : "Copos de Alumínio"},  
 257: {id : '257', nome : "Copos e outros pequenos Artefatos de Vidro"},  
 258: {id : '258', nome : "Copos plasticos"},  
 259: {id : '259', nome : "Coque (Carvão)"},  
 260: {id : '260', nome : "Correias de transmissão - Plastico"},  
 261: {id : '261', nome : "Correio, Malote e Demais Serviços de Entrega Expressa"},  
 262: {id : '262', nome : "Correntes cortantes para serra"},  
 263: {id : '263', nome : "Cortadores para unhas"},  
 264: {id : '264', nome : "Corta-papeis para escritorio"},  
 265: {id : '265', nome : "Corta-pinos (ferramenta)"},  
 266: {id : '266', nome : "Corta-tubos (ferramenta)"},  
 267: {id : '267', nome : "Cotovelos de Aluminio"},  
 268: {id : '268', nome : "Cotovelos para Tubos de Ferro e Aço"},  
 269: {id : '269', nome : "Couros"},  
 270: {id : '270', nome : "Cromo"},  
 271: {id : '271', nome : "Crustáceos e Moluscos"},  
 272: {id : '272', nome : "Crustáceos industrializado"},  
 273: {id : '273', nome : "Cubas e recipientes"},  
 274: {id : '274', nome : "Cumeeiras Metálicas (Peças e Volumes)"},  
 275: {id : '275', nome : "Defensivos agrícolas (agrotóxicos)"},  
 276: {id : '276', nome : "Dendê (a Granel)"},  
 277: {id : '277', nome : "Desperdícios de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 278: {id : '278', nome : "Dextrina"},  
 279: {id : '279', nome : "Dobradiças"},  
 280: {id : '280', nome : "Eguas"},  
 281: {id : '281', nome : "Eletrodomésticos"},  
 282: {id : '282', nome : "Eletrodos"},  
 283: {id : '283', nome : "Embalagens de Papel ou Papelão"},  
 284: {id : '284', nome : "Embalagens de Plástico"},  
 285: {id : '285', nome : "Embalagens de Vidro"},  
 286: {id : '286', nome : "Embalagens Metálicas"},  
 287: {id : '287', nome : "Embalagens Plasticas"},  
 288: {id : '288', nome : "Embarcações"},  
 289: {id : '289', nome : "Engradados de Plastico"},  
 290: {id : '290', nome : "Enlatados"},  
 291: {id : '291', nome : "Entulho"},  
 292: {id : '292', nome : "Enxofre"},  
 293: {id : '293', nome : "Equinos"},  
 294: {id : '294', nome : "Equipamento Bélico (grande porte)"},  
 295: {id : '295', nome : "Equipamentos de Ar Condicionado"},  
 296: {id : '296', nome : "Equipamentos de Informática"},  
 297: {id : '297', nome : "Erva-Mate (a Granel)"},  
 298: {id : '298', nome : "Erva-Mate (Embalada)"},  
 299: {id : '299', nome : "Escadas Metálicas (Peças e Volumes)"},  
 300: {id : '300', nome : "Escamas de Aluminio"},  
 301: {id : '301', nome : "Escamas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 302: {id : '302', nome : "Escórias de Metal (restos de funilaria)"},  
 303: {id : '303', nome : "Escorredores de Louças"},  
 304: {id : '304', nome : "Escovas, Pincéis e Vassouras"},  
 305: {id : '305', nome : "Escumadeiras para cozinha"},  
 306: {id : '306', nome : "Esferas para Moinhos de Ferro e Aço Fundidas"},  
 307: {id : '307', nome : "Espadas"},  
 308: {id : '308', nome : "Espatulas para escritorio"},  
 309: {id : '309', nome : "Espatulas para Unhas"},  
 310: {id : '310', nome : "Esponjas de Aço para limpeza e polimento"},  
 311: {id : '311', nome : "Espumas de Poliuretano"},  
 312: {id : '312', nome : "Esquadrias de Metal"},  
 313: {id : '313', nome : "Esquadrias e Venezianas de Madeira"},  
 314: {id : '314', nome : "Estamparia de Metais"},  
 315: {id : '315', nome : "Estanho"},  
 316: {id : '316', nome : "Estatuetas e Outros Objetos Metálicos para Ornamentação (Embalados)"},  
 317: {id : '317', nome : "Estruturas de Farois Maritimos"},  
 318: {id : '318', nome : "Estruturas Metálicas"},  
 319: {id : '319', nome : "Estruturas para navios"},  
 320: {id : '320', nome : "Estruturas Pré-Moldadas de Concreto Armado, em Série ou Sob Encomenda"},  
 321: {id : '321', nome : "Estufas Elétricas Industriais"},  
 322: {id : '322', nome : "Artefatos de trefilados metálicos (peças pequenas para industria, residencia, escritorio)"},  
 323: {id : '323', nome : "Artefatos diversos de plástico (escritorio, doméstico, uso industrial)"},  
 324: {id : '324', nome : "Outros Produtos Alimentícios"},  
 325: {id : '325', nome : "Facas cortantes para máquinas"},  
 326: {id : '326', nome : "Facas e Artigos de Cutelaria"},  
 327: {id : '327', nome : "Facas para uso profissional (ferramenta)"},  
 328: {id : '328', nome : "Farelo de Soja"},  
 329: {id : '329', nome : "Farinha de Mandioca"},  
 330: {id : '330', nome : "Farinha de Milho"},  
 331: {id : '331', nome : "Farinha de Trigo"},  
 332: {id : '332', nome : "Fateixas Metálicas e Suas Partes (Peças e Volumes)"},  
 333: {id : '333', nome : "Fechaduras"},  
 334: {id : '334', nome : "Feijão (em Sacos)"},  
 335: {id : '335', nome : "Feijão Beneficiado (Embalado)"},  
 336: {id : '336', nome : "Fermentos, Leveduras e Coalhos"},  
 337: {id : '337', nome : "Ferragens para Construção Civil"},  
 338: {id : '338', nome : "Ferragens para Encadernação Metálicos (Embalados)"},  
 339: {id : '339', nome : "Ferragens para Linhas Elétricas"},  
 340: {id : '340', nome : "Ferramentas de Chanfrar"},  
 341: {id : '341', nome : "Ferramentas de Fresar"},  
 342: {id : '342', nome : "Ferramentas de moldurar, ranchurar"},  
 343: {id : '343', nome : "Ferramentas de roscar"},  
 344: {id : '344', nome : "Ferramentas Manuais"},  
 345: {id : '345', nome : "Ferramentas para jardinagem"},  
 346: {id : '346', nome : "Ferramentas para pedreiros"},  
 347: {id : '347', nome : "Ferramentas para tornear"},  
 348: {id : '348', nome : "Ferrolhos"},  
 349: {id : '349', nome : "Ferroligas"},  
 350: {id : '350', nome : "Fertilizantes"},  
 351: {id : '351', nome : "Fichários Metálicos (Embalados)"},  
 352: {id : '352', nome : "Fio-máquina de Aço"},  
 353: {id : '353', nome : "Fios de Aluminio"},  
 354: {id : '354', nome : "Fios de níquel e de fios de ferro e aço"},  
 355: {id : '355', nome : "Fios e Folhas de Outros Metais não Ferrosos"},  
 356: {id : '356', nome : "Fios e têxteis"},  
 357: {id : '357', nome : "Fios Elétricos"},  
 358: {id : '358', nome : "Flores e Plantas Ornamentais"},  
 359: {id : '359', nome : "Fogões"},  
 360: {id : '360', nome : "Fogos de Artifício"},  
 361: {id : '361', nome : "Foices"},  
 362: {id : '362', nome : "Folhas de Aluminio"},  
 363: {id : '363', nome : "Folhas de Plástico"},  
 364: {id : '364', nome : "Folhas de serra (ferramenta)"},  
 365: {id : '365', nome : "Folhas-de-flandres de Aço"},  
 366: {id : '366', nome : "Forcados (ferramenta)"},  
 367: {id : '367', nome : "Forjados de Aço"},  
 368: {id : '368', nome : "Forjados de Metais Não Ferrosos"},  
 369: {id : '369', nome : "Forjaria de Metais"},  
 370: {id : '370', nome : "Forjas portateis"},  
 371: {id : '371', nome : "Formões"},  
 372: {id : '372', nome : "Fornos Elétricos Industriais"},  
 373: {id : '373', nome : "Fornos Industriais não-Elétricos"},  
 374: {id : '374', nome : "Forquilhas (ferramenta)"},  
 375: {id : '375', nome : "Fosfato"},  
 376: {id : '376', nome : "Fósforos de Segurança"},  
 377: {id : '377', nome : "Frango Abatido (resfriado, frigorificado, congelado)"},  
 378: {id : '378', nome : "Frangos para Corte"},  
 379: {id : '379', nome : "Frascos Plasticos"},  
 380: {id : '380', nome : "Frigideiras"},  
 381: {id : '381', nome : "Frutose"},  
 382: {id : '382', nome : "Fumo em folha"},  
 383: {id : '383', nome : "Furadeiras"},  
 384: {id : '384', nome : "Furadores para bordar"},  
 385: {id : '385', nome : "Galinhas"},  
 386: {id : '386', nome : "Galos"},  
 387: {id : '387', nome : "Ganchos e Pinos"},  
 388: {id : '388', nome : "Garfos"},  
 389: {id : '389', nome : "Garrafas e Garrafoes de Plastico"},  
 390: {id : '390', nome : "Gás de cozinha"},  
 391: {id : '391', nome : "Gás liquefeito de petróleo (GLP)"},  
 392: {id : '392', nome : "Gases Industriais"},  
 393: {id : '393', nome : "Gasoálcool"},  
 394: {id : '394', nome : "Gasolina"},  
 395: {id : '395', nome : "Gatos"},  
 396: {id : '396', nome : "Geladeiras"},  
 397: {id : '397', nome : "Gêlo / alimentos e sucos congelados"},  
 398: {id : '398', nome : "Gesso"},  
 399: {id : '399', nome : "Gipsita"},  
 400: {id : '400', nome : "Glicose"},  
 401: {id : '401', nome : "Glútem de Trigo"},  
 402: {id : '402', nome : "Gonzos Metálicas (Peças e Volumes)"},  
 403: {id : '403', nome : "Grades de Esgotos de Ferro e Aço Fundidas"},  
 404: {id : '404', nome : "Grades e redes de fios de cobre"},  
 405: {id : '405', nome : "Grampeadores e Semelhantes Metálicos (Embalados)"},  
 406: {id : '406', nome : "Grampos em Barretas e Artefatos Semelhantes Metálicos (Embalados)"},  
 407: {id : '407', nome : "Granalhas de Ferro"},  
 408: {id : '408', nome : "Granalhas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 409: {id : '409', nome : "Grandes Estruturas e Peças de Caldeiraria Pesada"},  
 410: {id : '410', nome : "Granitos"},  
 411: {id : '411', nome : "Gravuras e Semelhantes Metálicos (Embalados)"},  
 412: {id : '412', nome : "Guaraná (In Natura)"},  
 413: {id : '413', nome : "Guaraná (Pó, Sementes e Doces)"},  
 414: {id : '414', nome : "Guaraná (Refrigerante)"},  
 415: {id : '415', nome : "Guarniçoes de Aço"},  
 416: {id : '416', nome : "Hélices Metálicas para Embarcações (Peças e Volumes)"},  
 417: {id : '417', nome : "Hortaliças (a Granel)"},  
 418: {id : '418', nome : "Hortaliças (Embalado)"},  
 419: {id : '419', nome : "Instrumentos Musicais"},  
 420: {id : '420', nome : "Intermediários para Fertilizantes (Outros)"},  
 421: {id : '421', nome : "Iogurtes"},  
 422: {id : '422', nome : "Jogos Recreativos"},  
 423: {id : '423', nome : "Jornais e Revistas"},  
 424: {id : '424', nome : "Jumentos e Jumentas"},  
 425: {id : '425', nome : "Juta"},  
 426: {id : '426', nome : "Lagostas"},  
 427: {id : '427', nome : "Lagostins"},  
 428: {id : '428', nome : "Laminados de Alumínio"},  
 429: {id : '429', nome : "Laminados de Outros Metais não Ferrosos"},  
 430: {id : '430', nome : "Laminados e Semi Acabados de Aço"},  
 431: {id : '431', nome : "Laminados e Semi Acabados de Aço Inoxidável"},  
 432: {id : '432', nome : "Laminados e Semi Acabados de Ferro"},  
 433: {id : '433', nome : "Laminados e Semi Acabados de Outras Ligas de Aço"},  
 434: {id : '434', nome : "Laminados Planos de Aço"},  
 435: {id : '435', nome : "Laminados Planos e Tubulares de Plástico"},  
 436: {id : '436', nome : "Laminas cortantes para máquinas"},  
 437: {id : '437', nome : "Lâminas de Plástico"},  
 438: {id : '438', nome : "Laminas para barbear"},  
 439: {id : '439', nome : "Laminas para facas"},  
 440: {id : '440', nome : "Lâmpadas"},  
 441: {id : '441', nome : "Lâmpadas, lamparina para soldagem"},  
 442: {id : '442', nome : "Lanças"},  
 443: {id : '443', nome : "Lanchas"},  
 444: {id : '444', nome : "Laranja (a Granel)"},  
 445: {id : '445', nome : "Laranja (Doces e Conservas)"},  
 446: {id : '446', nome : "Laranja (Embalada)"},  
 447: {id : '447', nome : "Latas de Aluminio, Aço, Ferro"},  
 448: {id : '448', nome : "Látex (Borracha)"},  
 449: {id : '449', nome : "Laticínios (Derivados do Leite)"},  
 450: {id : '450', nome : "Lavatórios de ferro ou aço"},  
 451: {id : '451', nome : "Lavatórios de plastico"},  
 452: {id : '452', nome : "Leite em Pó"},  
 453: {id : '453', nome : "Leite In Natura (em Latões ou Tarros)"},  
 454: {id : '454', nome : "Leite longa vida"},  
 455: {id : '455', nome : "Leite Pasteurizado (Envasado ou Ensacado)"},  
 456: {id : '456', nome : "Leite Pasteurizado (Granéis Líquidos)"},  
 457: {id : '457', nome : "Leitões e Leitoas vivos"},  
 458: {id : '458', nome : "Letras, Números e Sinais diversos Metálicos (Embalados)"},  
 459: {id : '459', nome : "Limão (a Granel)"},  
 460: {id : '460', nome : "Limão (Doces e Conservas)"},  
 461: {id : '461', nome : "Limão (Embalado)"},  
 462: {id : '462', nome : "Limas para afiar"},  
 463: {id : '463', nome : "Lingotes de Alumínio"},  
 464: {id : '464', nome : "Lingotes de Ferro"},  
 465: {id : '465', nome : "Lingotes de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 466: {id : '466', nome : "Linhas e Fios Para Costurar e Bordar"},  
 467: {id : '467', nome : "Liquidificadores"},  
 468: {id : '468', nome : "Lítio"},  
 469: {id : '469', nome : "Livros"},  
 470: {id : '470', nome : "Lixo Doméstico"},  
 471: {id : '471', nome : "Lixos e Resíduos Hospitalares"},  
 472: {id : '472', nome : "Lixos e Resíduos Industriais"},  
 473: {id : '473', nome : "Locomotivas"},  
 474: {id : '474', nome : "Louças Sanitárias"},  
 475: {id : '475', nome : "Luminárias e Equipamentos de Iluminação - Exceto Para Veículos"},  
 476: {id : '476', nome : "Maçã (a Granel)"},  
 477: {id : '477', nome : "Maçã (Doces e Conservas)"},  
 478: {id : '478', nome : "Maçã (Embalada)"},  
 479: {id : '479', nome : "Maçaricos"},  
 480: {id : '480', nome : "Macarrão e Massas Alimentícias"},  
 481: {id : '481', nome : "Machados"},  
 482: {id : '482', nome : "Madeira (Lenha)"},  
 483: {id : '483', nome : "Madeira Desdobrada (serrada)"},  
 484: {id : '484', nome : "Madeira em toras"},  
 485: {id : '485', nome : "Madeira Industrializada (Compensados e Aglomerados)"},  
 486: {id : '486', nome : "Madeira para Indústria de Celulose"},  
 487: {id : '487', nome : "Malte"},  
 488: {id : '488', nome : "Mamão (a Granel)"},  
 489: {id : '489', nome : "Mamão (Doces e Conservas)"},  
 490: {id : '490', nome : "Mamão (Embalado)"},  
 491: {id : '491', nome : "Mandioca ou Aipim (a Granel)"},  
 492: {id : '492', nome : "Mandioca ou Aipim (Embalada)"},  
 493: {id : '493', nome : "Manga (a Granel)"},  
 494: {id : '494', nome : "Manga (Doces e Conservas)"},  
 495: {id : '495', nome : "Manga (Embalada)"},  
 496: {id : '496', nome : "Manganês"},  
 497: {id : '497', nome : "Manteiga"},  
 498: {id : '498', nome : "Máquinas agrícolas"},  
 499: {id : '499', nome : "Máquinas de Apontar Lápis Metálicos (Embalados)"},  
 500: {id : '500', nome : "Máquinas de Lavar e Secar para Uso Doméstico"},  
 501: {id : '501', nome : "Máquinas e equipamentos (outros)"},  
 502: {id : '502', nome : "Máquinas para Escritório"},  
 503: {id : '503', nome : "Maracujá (a Granel)"},  
 504: {id : '504', nome : "Maracujá (Doces e Conservas)"},  
 505: {id : '505', nome : "Maracujá (Embalado)"},  
 506: {id : '506', nome : "Margarina"},  
 507: {id : '507', nome : "Mariscos"},  
 508: {id : '508', nome : "Mármores"},  
 509: {id : '509', nome : "Marretas"},  
 510: {id : '510', nome : "Martelos"},  
 511: {id : '511', nome : "Materiais de construção (outros)"},  
 512: {id : '512', nome : "Materiais de escritório (outros)"},  
 513: {id : '513', nome : "Materiais Elétricos (outros)"},  
 514: {id : '514', nome : "Materiais Impressos"},  
 515: {id : '515', nome : "Material Elétrico para Veículos (outros)"},  
 516: {id : '516', nome : "Material Eletrônico Básico"},  
 517: {id : '517', nome : "Material para Andaimes, Armacoes e Escoramento"},  
 518: {id : '518', nome : "Medicamentos Para Uso Humano"},  
 519: {id : '519', nome : "Medicamentos Para Uso Veterinário"},  
 520: {id : '520', nome : "Melancia (a Granel)"},  
 521: {id : '521', nome : "Melancia (Doces e Conservas)"},  
 522: {id : '522', nome : "Melancia (Embalada)"},  
 523: {id : '523', nome : "Melão (a Granel)"},  
 524: {id : '524', nome : "Melão (Doces e Conservas)"},  
 525: {id : '525', nome : "Melão (Embalado)"},  
 526: {id : '526', nome : "Metalurgia de Metais"},  
 527: {id : '527', nome : "Metalurgia de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 528: {id : '528', nome : "Milho (a Granel)"},  
 529: {id : '529', nome : "Milho (Doces, Conservas e Salgados)"},  
 530: {id : '530', nome : "Milho (Embalado)"},  
 531: {id : '531', nome : "Minerais (outros)"},  
 532: {id : '532', nome : "Minério de Alumínio (Bauxita)"},  
 533: {id : '533', nome : "Minério de Ferro"},  
 534: {id : '534', nome : "Molas de cobre e de níquel"},  
 535: {id : '535', nome : "Molas e folhas de molas de ferro e aço"},  
 536: {id : '536', nome : "Moldadores (ferramenta)"},  
 537: {id : '537', nome : "Molduras e Semelhantes Metálicos (Embalados)"},  
 538: {id : '538', nome : "Molhos, Temperos e Condimentos"},  
 539: {id : '539', nome : "Moluscos industrializado"},  
 540: {id : '540', nome : "Morango (a Granel)"},  
 541: {id : '541', nome : "Morango (Doces e Conservas)"},  
 542: {id : '542', nome : "Morango (Embalado)"},  
 543: {id : '543', nome : "Mortadela"},  
 544: {id : '544', nome : "Mos com armação (ferramenta)"},  
 545: {id : '545', nome : "Motocicletas"},  
 546: {id : '546', nome : "Motores Elétricos, Inclusive Peças"},  
 547: {id : '547', nome : "Motores para Automóveis, Camionetas e Utilitários"},  
 548: {id : '548', nome : "Motores para Caminhões e Ônibus"},  
 549: {id : '549', nome : "Motores, Bombas e Compressores para Máquinas e Equipamentos"},  
 550: {id : '550', nome : "Moveis de Aço, Aluminio ou Ferro"},  
 551: {id : '551', nome : "Móveis de Madeira"},  
 552: {id : '552', nome : "Mulas e Burros"},  
 553: {id : '553', nome : "Navalhas para barbear"},  
 554: {id : '554', nome : "Nióbio"},  
 555: {id : '555', nome : "Níquel"},  
 556: {id : '556', nome : "Novilhos ou Novilhas"},  
 557: {id : '557', nome : "Óleo combustível"},  
 558: {id : '558', nome : "Óleo de Algodão (Granéis Líquidos)"},  
 559: {id : '559', nome : "Óleo de Algodão (Envasado)"},  
 560: {id : '560', nome : "Óleo de Amendoim (Granéis Líquidos)"},  
 561: {id : '561', nome : "Óleo de Amendoim (Envasado)"},  
 562: {id : '562', nome : "Óleo de Dendê (Granéis Líquidos)"},  
 563: {id : '563', nome : "Óleo de Dendê (Envasado)"},  
 564: {id : '564', nome : "Óleo de Girassol (Granéis Líquidos)"},  
 565: {id : '565', nome : "Óleo de Girassol (Envasado)"},  
 566: {id : '566', nome : "Óleo de Mamona"},  
 567: {id : '567', nome : "Óleo de Milho (Envasado)"},  
 568: {id : '568', nome : "Óleo de Milho (Granéis Líquidos)"},  
 569: {id : '569', nome : "Óleo de Soja (Envasado)"},  
 570: {id : '570', nome : "Óleo de Soja (Granéis Líquidos)"},  
 571: {id : '571', nome : "Óleo Diesel"},  
 572: {id : '572', nome : "Óleos envazados (milho, amidos e féculas vegetais)"},  
 573: {id : '573', nome : "Óleos envasados (vegetal e animal)"},  
 574: {id : '574', nome : "Óleos outros (Granéis Líquidos)"},  
 575: {id : '575', nome : "Ônibus"},  
 576: {id : '576', nome : "Ostras"},  
 577: {id : '577', nome : "Ouro"},  
 578: {id : '578', nome : "Outras Frutas (Doces e Conservas)"},  
 579: {id : '579', nome : "Outras Frutas (a Granel)"},  
 580: {id : '580', nome : "Outras Pedras em Placas"},  
 581: {id : '581', nome : "Outros Animais Domésticos"},  
 582: {id : '582', nome : "Outros Produtos Metálicos (Embalados)"},  
 583: {id : '583', nome : "Outros Produtos Metálicos (Peças e Volumes)"},  
 584: {id : '584', nome : "Ovinos"},  
 585: {id : '585', nome : "Ovos de galinha e de outras aves"},  
 586: {id : '586', nome : "Pães e Bolos Industrializados"},  
 587: {id : '587', nome : "Palha (lã) de ferro e aço"},  
 588: {id : '588', nome : "Palmito (In Natura)"},  
 589: {id : '589', nome : "Palmito em Conserva (embalado)"},  
 590: {id : '590', nome : "Panelas"},  
 591: {id : '591', nome : "Papel e papelão e artefatos"},  
 592: {id : '592', nome : "Parafusos"},  
 593: {id : '593', nome : "Pás"},  
 594: {id : '594', nome : "Peças de Caldeiraria Pesada"},  
 595: {id : '595', nome : "Peças de plastico"},  
 596: {id : '597', nome : "Peças e acessórios para automóveis"},  
 597: {id : '598', nome : "Peças e acessórios para caminhões"},  
 598: {id : '599', nome : "Peças e Acessórios para Ferrovias"},  
 599: {id : '600', nome : "Peças e acessórios para motocicletas"},  
 600: {id : '601', nome : "Peças e acessórios para ônibus"},  
 601: {id : '602', nome : "Peças e acessórios para veículos automotores"},  
 602: {id : '603', nome : "Peças e acessórios para veículos utilitários"},  
 603: {id : '604', nome : "Peças Fundidas de Ferro e Aço"},  
 604: {id : '605', nome : "Peças Fundidas de Metais Não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 605: {id : '606', nome : "Peças e equipamentos para Embarcações"},  
 606: {id : '607', nome : "Peças para Reatores nucleares"},  
 607: {id : '608', nome : "Pedra Britada"},  
 608: {id : '609', nome : "Pedras para Revestimento/ Calçamento (Piso)"},  
 609: {id : '610', nome : "Peito de Perú"},  
 610: {id : '611', nome : "Peixe industrializado"},  
 611: {id : '612', nome : "Peixes"},  
 612: {id : '613', nome : "Peixes vivos"},  
 613: {id : '614', nome : "Películas de Plástico"},  
 614: {id : '615', nome : "Pellets Cítricos (Farinhas)"},  
 615: {id : '616', nome : "Pequenos Roedores"},  
 616: {id : '617', nome : "Perdizes"},  
 617: {id : '618', nome : "Perfis de Aço"},  
 618: {id : '619', nome : "Perfis de Outros Metais não Ferrosos"},  
 619: {id : '620', nome : "Cosméticos e artigos de perfumaria"},  
 620: {id : '621', nome : "Perfumes"},  
 621: {id : '622', nome : "Perfuradoras (ferramenta)"},  
 622: {id : '623', nome : "Perfuradores Metálicos (Embalados)"},  
 623: {id : '624', nome : "Peruas (ave)"},  
 624: {id : '625', nome : "Perus"},  
 625: {id : '626', nome : "Pescado industrializado"},  
 626: {id : '627', nome : "Pêssego (a Granel)"},  
 627: {id : '628', nome : "Pêssego (Doces e Conservas)"},  
 628: {id : '629', nome : "Pêssego (Embalado)"},  
 629: {id : '630', nome : "Petróleo Bruto"},  
 630: {id : '631', nome : "Petroquimicos Básicos (Granéis Líquidos)"},  
 631: {id : '632', nome : "Petroquimicos Básicos (Granéis Sólidos)"},  
 632: {id : '633', nome : "Pias (metal)"},  
 633: {id : '634', nome : "Picaretas"},  
 634: {id : '635', nome : "Pimenta do Reino (a Granel)"},  
 635: {id : '636', nome : "Pimenta do Reino (Embalada)"},  
 636: {id : '637', nome : "Pinças e artefatos"},  
 637: {id : '638', nome : "Pinças e ferramentas semelhantes"},  
 638: {id : '639', nome : "Pintos"},  
 639: {id : '640', nome : "Placas de Aço"},  
 640: {id : '641', nome : "Placas de Aluminio"},  
 641: {id : '642', nome : "Placas de Ferro"},  
 642: {id : '643', nome : "Placas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 643: {id : '644', nome : "Placas Indicadoras Metálicas (Peças e Volumes)"},  
 644: {id : '645', nome : "Placas Metálicas para Veículos (Embalados)"},  
 645: {id : '646', nome : "Placas Sinalizadoras e Semelhantes Metálicas (Peças e Volumes)"},  
 646: {id : '647', nome : "Plainas"},  
 647: {id : '648', nome : "Mudas de Plantas ou Árvores"},  
 648: {id : '649', nome : "Plaquetas de Aluminio"},  
 649: {id : '650', nome : "Plaquetas de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 650: {id : '651', nome : "Plásticos (outros)"},  
 651: {id : '652', nome : "Pneus e Câmaras de Ar Novos"},  
 652: {id : '653', nome : "Pneus Recondicionados"},  
 653: {id : '654', nome : "Pneus Usados"},  
 654: {id : '655', nome : "Pólvoras, Explosivos e Detonantes"},  
 655: {id : '656', nome : "Pontes e elementos de pontes, de ferro e aço"},  
 656: {id : '657', nome : "Porcas e Arruelas"},  
 657: {id : '658', nome : "Porcos e Porcas vivos"},  
 658: {id : '659', nome : "Porta-canetas e Semelhantes Metálicos (Embalados)"},  
 659: {id : '660', nome : "Portas Blindadas e Compartimentos para Casas-Fortes (Peças e Volumes)"},  
 660: {id : '661', nome : "Portas de diques"},  
 661: {id : '662', nome : "Portas de Eclusas"},  
 662: {id : '663', nome : "Portas e janelas de Madeida"},  
 663: {id : '664', nome : "Portas e Janelas de Metal"},  
 664: {id : '665', nome : "Portas e Portoes de Aço, Aluminio ou Ferro"},  
 665: {id : '666', nome : "Pós de Ferro"},  
 666: {id : '667', nome : "Pós de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 667: {id : '668', nome : "Potros"},  
 668: {id : '669', nome : "Prata"},  
 669: {id : '670', nome : "Pratos (metal)"},  
 670: {id : '671', nome : "Pré-Moldados de concreto"},  
 671: {id : '672', nome : "Presunto"},  
 672: {id : '673', nome : "Processadores de Alimentos (Eletrodoméstico)"},  
 673: {id : '674', nome : "Produtos agrícolas (outros)"},  
 674: {id : '675', nome : "Produtos da floresta (outros)"},  
 675: {id : '676', nome : "Produtos de Higiene"},  
 676: {id : '677', nome : "Produtos de Madeira"},  
 677: {id : '678', nome : "Produtos de Salsicharia"},  
 678: {id : '679', nome : "Produtos Dietéticos"},  
 679: {id : '680', nome : "Produtos farmacêuticos"},  
 680: {id : '681', nome : "Produtos para fotografia"},  
 681: {id : '682', nome : "Produtos químicos outros (Granéis Líquidos)"},  
 682: {id : '683', nome : "Produtos químicos outros (Granéis Sólidos)"},  
 683: {id : '684', nome : "Produtos Têxteis (outros)"},  
 684: {id : '685', nome : "Quadros e Artigos para decoração"},  
 685: {id : '686', nome : "Queijos"},  
 686: {id : '687', nome : "Querosene de aviação"},  
 687: {id : '688', nome : "Ração Animal"},  
 688: {id : '689', nome : "Rádios e Aparelhos de Som"},  
 689: {id : '690', nome : "Raspadeiras (ferramenta)"},  
 690: {id : '691', nome : "Reatores nucleares"},  
 691: {id : '692', nome : "Rebites de Aço, Cobre e Ferro"},  
 692: {id : '694', nome : "Recipientes de Aço, Aluminio ou Ferro"},  
 693: {id : '695', nome : "Refrigerantes (Envasados)"},  
 694: {id : '696', nome : "Refrigerantes (Granéis Líquidos)"},  
 695: {id : '697', nome : "Regadores e semelhantes, de zinco"},  
 696: {id : '698', nome : "Regadores e similares, de plastico"},  
 697: {id : '699', nome : "Reservatorios Metalicos"},  
 698: {id : '700', nome : "Resíduos de Aluminio"},  
 699: {id : '701', nome : "Resíduos de Metais não-Ferrosos (cobre, latão, alumínio, bronze, outros)"},  
 700: {id : '702', nome : "Resinas e Elastômeros"},  
 701: {id : '703', nome : "Rocha Fosfática"},  
 702: {id : '704', nome : "Rocha Ornamental"},  
 703: {id : '705', nome : "Rodízios de Aço, Aluminio ou Ferro"},  
 704: {id : '706', nome : "Rolhas e Tampas Metalicas"},  
 705: {id : '707', nome : "Rolhas e Tampas Plasticas"},  
 706: {id : '708', nome : "Rosbife"},  
 707: {id : '709', nome : "Roupas e confecções"},  
 708: {id : '710', nome : "Sabões, Detergentes Produtos de Limpeza"},  
 709: {id : '711', nome : "Saboneteiras de Aluminio"},  
 710: {id : '712', nome : "Sabonetes e Shampoos"},  
 711: {id : '713', nome : "Saca-bocados (ferramenta)"},  
 712: {id : '714', nome : "Sachos (ferramenta)"},  
 713: {id : '715', nome : "Sacos e Sacolas Plasticas"},  
 714: {id : '716', nome : "Sal (a Granel)"},  
 715: {id : '717', nome : "Sal (Embalado)"},  
 716: {id : '718', nome : "Salame"},  
 717: {id : '719', nome : "Saleiros (metal)"},  
 718: {id : '720', nome : "Salsicha"},  
 719: {id : '721', nome : "Sapatos"},  
 720: {id : '722', nome : "Sargentos (ferramentas)"},  
 721: {id : '723', nome : "Semeadores manuais"},  
 722: {id : '724', nome : "Sementes Para Lavoura Temporária"},  
 723: {id : '725', nome : "Sementes Para Pastagem"},  
 724: {id : '726', nome : "Serras manuais"},  
 725: {id : '727', nome : "Siderúrgicos (outros)"},  
 726: {id : '728', nome : "Sinos Metálicas (Peças e Volumes)"},  
 727: {id : '729', nome : "Soja em grão"},  
 728: {id : '730', nome : "Sorgo"},  
 729: {id : '731', nome : "Sucatas de Ferro e Aço"},  
 730: {id : '732', nome : "Sucatas Metálicas"},  
 731: {id : '733', nome : "Suco de Laranja (Envasado)"},  
 732: {id : '734', nome : "Suco de Laranja (Granéis Líquidos)"},  
 733: {id : '735', nome : "Suco de Uva (Envasado)"},  
 734: {id : '736', nome : "Suco de Uva (Granéis Líquidos)"},  
 735: {id : '737', nome : "Sucos Outros (Envasado)"},  
 736: {id : '738', nome : "Sucos Outros (Granéis Líquidos)"},  
 737: {id : '739', nome : "Suínos vivos"},  
 738: {id : '740', nome : "Suportes (ganchos) Metálicos para Vestuário (Embalados)"},  
 739: {id : '741', nome : "Suportes de Metal"},  
 740: {id : '742', nome : "Suportes de Plastico"},  
 741: {id : '743', nome : "Tambores de Metal"},  
 742: {id : '744', nome : "Tambores de Plastico"},  
 743: {id : '745', nome : "Tampas plasticas para vaso Sanitário"},  
 744: {id : '746', nome : "Tangerina (a Granel)"},  
 745: {id : '747', nome : "Tangerina (Doces e Conservas)"},  
 746: {id : '748', nome : "Tangerina (Embalada)"},  
 747: {id : '749', nome : "Tanques de Cozinha (metal)"},  
 748: {id : '750', nome : "Tanques de grande capacidade (metal)"},  
 749: {id : '751', nome : "Tântalo"},  
 750: {id : '752', nome : "Tapetes"},  
 751: {id : '753', nome : "Tapioca e seus sucedândeos"},  
 752: {id : '754', nome : "Tecidos e Seus Artefatos (Exceto Vestuário)"},  
 753: {id : '755', nome : "Tecidos têxteis"},  
 754: {id : '756', nome : "Telas de fios de cobre, niquel"},  
 755: {id : '757', nome : "Telas metálicas tecidas"},  
 756: {id : '758', nome : "Telefones e Sistemas de Comunicação"},  
 757: {id : '759', nome : "Televisores"},  
 758: {id : '760', nome : "Telhas"},  
 759: {id : '761', nome : "Tenazes (ferramenta)"},  
 760: {id : '762', nome : "Tesouras"},  
 761: {id : '763', nome : "Tesouras-serrote"},  
 762: {id : '764', nome : "Tijolos"},  
 763: {id : '765', nome : "Tintas, vernizes, esmaltes e lacas"},  
 764: {id : '766', nome : "Tiras de Aço"},  
 765: {id : '767', nome : "Tiras de Aluminio"},  
 766: {id : '768', nome : "Tiras de Outros Metais não Ferrosos"},  
 767: {id : '769', nome : "Tiras de Plástico"},  
 768: {id : '770', nome : "Titânio"},  
 769: {id : '771', nome : "Tomate (a Granel)"},  
 770: {id : '772', nome : "Tomate (Doces e Conservas)"},  
 771: {id : '773', nome : "Tomate (Embalado)"},  
 772: {id : '774', nome : "Tonéis"},  
 773: {id : '775', nome : "Torneiras e Válvulas"},  
 774: {id : '776', nome : "Tornos"},  
 775: {id : '777', nome : "Torres de telegrafia"},  
 776: {id : '778', nome : "Torres e pórticos (pilares), de ferro e aço"},  
 777: {id : '779', nome : "Trailers"},  
 778: {id : '780', nome : "Tratores e Implementos para Agricultura (Inclusive Peças)"},  
 779: {id : '781', nome : "Travessas"},  
 780: {id : '782', nome : "Trigo em grão"},  
 781: {id : '783', nome : "Trilhos de Aço"},  
 782: {id : '784', nome : "Tubos de Ferro e Aço"},  
 783: {id : '785', nome : "Tubos de Outros Metais não Ferrosos"},  
 784: {id : '786', nome : "Tubos de Plástico"},  
 785: {id : '787', nome : "Tubos e Conexões de Ferro, Aço e Aço Inoxidável"},  
 786: {id : '788', nome : "Tubos Flexiveis de Chumbo para Embalagens"},  
 787: {id : '789', nome : "Tubos Flexiveis de Metal"},  
 788: {id : '790', nome : "Tungstênio"},  
 789: {id : '791', nome : "Uniões de Aluminio"},  
 790: {id : '792', nome : "Uniões para Tubos de Ferro e Aço"},  
 791: {id : '793', nome : "Uréia"},  
 792: {id : '794', nome : "Utensílios de plastico"},  
 793: {id : '795', nome : "Utensílios p/Cozinha Movidos a Eletricidade"},  
 794: {id : '796', nome : "Uva (a Granel)"},  
 795: {id : '797', nome : "Uva (Doces e Conservas)"},  
 796: {id : '798', nome : "Uva (Embalado)"},  
 797: {id : '799', nome : "Vacas"},  
 798: {id : '800', nome : "Vagões Ferroviarios"},  
 799: {id : '801', nome : "Veículos e Material Rodante para Ferrovias"},  
 800: {id : '802', nome : "Veículos e Utilitários"},  
 801: {id : '803', nome : "Vergalhões de Aço"},  
 802: {id : '804', nome : "Vidros e Espelhos"},  
 803: {id : '805', nome : "Vidros Laminados para Veículos Automotores, Aeronaves, etc"},  
 804: {id : '806', nome : "Vidros Planos"},  
 805: {id : '807', nome : "Vinhos (Envasados)"},  
 806: {id : '808', nome : "Vinhos (Granéis Líquidos)"},  
 807: {id : '809', nome : "Vôngoles"},  
 808: {id : '810', nome : "Xarope de Glicose"},  
 809: {id : '811', nome : "Xícaras (metal)"},  
 810: {id : '812', nome : "Xicaras de plastico"},  
 811: {id : '813', nome : "Zinco"},  
 812: {id : '814', nome : "Zircônio"},  
 813: {id : '815', nome : "Equipamento Bélico (pequeno porte, armas e munição)"},  
 814: {id : '816', nome : "Balcoes e Balaustradas (madeira)"},  
 815: {id : '817', nome : "Figo (a Granel)"},  
 816: {id : '818', nome : "Figo (Doces e Conservas)"},  
 817: {id : '819', nome : "Figo (Embalada)"},  
 818: {id : '820', nome : "Goiaba (a Granel)"},  
 819: {id : '821', nome : "Goiaba (Doces e Conservas)"},  
 820: {id : '822', nome : "Goiaba (Embalada)"},  
 821: {id : '823', nome : "Outras frutas (Embalada)"},  
 822: {id : '824', nome : "Baldes (plástico)"},  
 823: {id : '825', nome : "Bidês (louça, porcelana, cerâmica)"},  
 824: {id : '826', nome : "Pias (louça, porcelana, cerâmica)"},  
 825: {id : '827', nome : "Pratos (louça, porcelana, cerâmica)"},  
 826: {id : '828', nome : "Artigos banheiro - saboneteiras, porta escova (louça, porcelana, cerâmica )"},  
 827: {id : '829', nome : "Saleiros (louça, porcelana, cerâmica)"},  
 828: {id : '830', nome : "Artigos banheiro - saboneteiras, porta escova (plástico)"},  
 829: {id : '831', nome : "Saleiros (plástico)"},  
 830: {id : '832', nome : "Tanques de Cozinha (plástico)"},  
 831: {id : '833', nome : "Tanques de Cozinha (louça, porcelana, cerâmica )"},  
 832: {id : '834', nome : "Tanques de grande capacidade (plástico, fibra de vidro, outros)"},  
 833: {id : '835', nome : "Xicaras (louça, porcelana, cerâmica )"},  
 834: {id : '836', nome : "Cajú (embalado)"},  
 835: {id : '837', nome : "Bisnagas flexiveis para embalagem (plástico)"},  
 836: {id : '838', nome : "Louças"},  
 837: {id : '839', nome : "Aparelhos Celulares"},  
 838: {id : '840', nome : "Cristaleria"},  
 839: {id : '841', nome : "Frutas (a Granel)"},  
 840: {id : '842', nome : "Remédios"},  
 841: {id : '844', nome : "Objetos pessoais"},  
 842: {id : '845', nome : "Malas"},  
 843: {id : '846', nome : "Mobiliários (mudanças)"},  
 844: {id : '848', nome : "Cadáver (Corpo humano)"},  
 845: {id : '849', nome : "Restos incinerados"},  
 846: {id : '850', nome : "Cinzas do corpo humano"},  
 847: {id : '851', nome : "Platina"},  
 848: {id : '853', nome : "Bônus"},  
 849: {id : '855', nome : "Obras de arte"},  
 850: {id : '856', nome : "Esculturas"},  
 851: {id : '857', nome : "Jóias"},  
 852: {id : '858', nome : "Pedras Preciosas ou Semi-preciosas"},  
 853: {id : '859', nome : "Jaulas"},  
 854: {id : '860', nome : "Gaiolas"},  
 855: {id : '861', nome : "Petardos"},  
 856: {id : '863', nome : "Extintores"},  
 857: {id : '864', nome : "Aerossóis"},  
 858: {id : '865', nome : "Inaladores"},  
 859: {id : '866', nome : "Isqueiros"},  
 860: {id : '867', nome : "Garrafas de Oxigênio"},  
 861: {id : '868', nome : "Garrafas de Ar Comprimido"},  
 862: {id : '869', nome : "Líquidos Inflamáveis"},  
 863: {id : '870', nome : "Colas"},  
 864: {id : '871', nome : "Acetona"},  
 865: {id : '873', nome : "Produtos Petrolíferos"},  
 866: {id : '874', nome : "Sólidos Inflamáveis"},  
 867: {id : '875', nome : "Magnésio"},  
 868: {id : '876', nome : "Sódio"},  
 869: {id : '877', nome : "Peróxidos Orgânicos"},  
 870: {id : '878', nome : "Substâncias Oxidantes"},  
 871: {id : '879', nome : "Nitrato Amoníaco"},  
 872: {id : '880', nome : "Geradores de Oxigênio"},  
 873: {id : '881', nome : "Tóxicos e Substâncias infecciosas"},  
 874: {id : '884', nome : "Desinfetantes"},  
 875: {id : '885', nome : "Sangue Contaminado"},  
 876: {id : '886', nome : "Amostras Infectadas"},  
 877: {id : '887', nome : "Material Radioativo"},  
 878: {id : '888', nome : "Detectores de fumaça"},  
 879: {id : '889', nome : "Plutônio"},  
 880: {id : '890', nome : "Urânio"},  
 881: {id : '891', nome : "Materiais Corrosivos"},  
 882: {id : '892', nome : "Ácidos"},  
 883: {id : '893', nome : "Mercúrio"},  
 884: {id : '894', nome : "Amoníaco"},  
 885: {id : '895', nome : "Produtos de Limpeza"},  
 886: {id : '896', nome : "Baterias e pilhas"},  
 887: {id : '897', nome : "Mercadorias Perigosas"},  
 888: {id : '898', nome : "Material Magnetizado"},  
 889: {id : '899', nome : "Equipamentos de Salvamento"},  
 890: {id : '900', nome : "Combustíveis e seus derivados"},  
 891: {id : '902', nome : "Cristais"},  
 892: {id : '903', nome : "Documentos, Títulos e Apólices"},  
 893: {id : '904', nome : "Peróxido de Hidrogênio"},  
 894: {id : '905', nome : "Água de Côco"},  
 895: {id : '906', nome : "Água Oxigenada"},  
 896: {id : '907', nome : "Alimentos Industrializados"},  
 897: {id : '908', nome : "Argamassa"},  
 898: {id : '910', nome : "Amianto"},  
 899: {id : '911', nome : "Antenas e Equipamentos de Telecomunicações"},  
 900: {id : '912', nome : "Massa Asfáltica"},  
 901: {id : '913', nome : "Elevadores"},  
 902: {id : '914', nome : "Aveia"},  
 903: {id : '915', nome : "Azeite"},  
 904: {id : '916', nome : "Azeitonas"},  
 905: {id : '917', nome : "Balanças"},  
 906: {id : '918', nome : "Bauxita"},  
 907: {id : '919', nome : "Paletes de Madeira"},  
 908: {id : '920', nome : "Bambu"},  
 909: {id : '921', nome : "Estofamentos / Poltronas / bancos para veículos"},  
 910: {id : '922', nome : "Estofados / Sofás / Poltronas / Almofadas"},  
 911: {id : '923', nome : "Barracas e Equipamentos para Festas"},  
 912: {id : '924', nome : "Botija ou Botijão (Vazios)"},  
 913: {id : '925', nome : "Bicarbonato de Sódio"},  
 914: {id : '926', nome : "Betoneiras"},  
 915: {id : '927', nome : "Beterrabas"},  
 916: {id : '928', nome : "Cenouras"},  
 917: {id : '929', nome : "Legumes / Verduras"},  
 918: {id : '930', nome : "Camas ou Beliches"},  
 919: {id : '931', nome : "Extrato de Tomate / Polpa de Tomate"},  
 920: {id : '932', nome : "Ervilhas"},  
 921: {id : '933', nome : "Adubo Orgânico / Esterco"},  
 922: {id : '934', nome : "Polpa de Frutas"},  
 923: {id : '935', nome : "Polietileno"},  
 924: {id : '936', nome : "Pó de Serra / Serragem"},  
 925: {id : '937', nome : "Cevada"},  
 926: {id : '938', nome : "Soda Cáustica ou Hidróxido de Sódio"},  
 927: {id : '939', nome : "Sulfato de Amônia"},  
 928: {id : '940', nome : "Sulfato de Alumínio"},  
 929: {id : '941', nome : "Sulfato de Cálcio"},  
 930: {id : '942', nome : "Tolueno ou metil benzeno"},  
 931: {id : '943', nome : "Solvente, Dissolvente ou Dispersante"},  
 932: {id : '944', nome : "Cereais"},  
 933: {id : '945', nome : "Sebo bovino"},  
 934: {id : '946', nome : "Sebo Ácido"},  
 935: {id : '948', nome : "Cesta Básica"},  
 936: {id : '949', nome : "Caixa Eletrônico"},  
 937: {id : '950', nome : "Caixão / Urna Funerária"},  
 938: {id : '951', nome : "Caixa de Papelão"},  
 939: {id : '952', nome : "Material Escolar / Cadernos"},  
 940: {id : '953', nome : "Cabos para Vassouras / Rodos"},  
 941: {id : '954', nome : "Cabides"},  
 942: {id : '955', nome : "Móveis / Cadeiras / Mesas"},  
 943: {id : '956', nome : "Mudanças"},  
 944: {id : '957', nome : "Cadeiras de Plástico"},  
 945: {id : '958', nome : "Caixas Térmicas"},  
 946: {id : '959', nome : "Embalagens ou Caixas de Isopor"},  
 947: {id : '960', nome : "Câmara Frigorífica"},  
 948: {id : '961', nome : "Capim, Grama ou Sementes de Capim"},  
 949: {id : '962', nome : "Carbonatos - Sódio / Cálcio / Cobre / Zinco / Guanidina / Cobalto/ Magnésio / Bário / Silício"},  
 950: {id : '963', nome : "Caixotes de Madeira"},  
 951: {id : '964', nome : "Churrasqueiras"},  
 952: {id : '965', nome : "Silagem ou Silos"},  
 953: {id : '966', nome : "Cianeto de sódio"},  
 954: {id : '967', nome : "Chaminés"},  
 955: {id : '968', nome : "Cilindros"},  
 956: {id : '969', nome : "Climatizadores / Aquecedores / Condicionadores de Ar"},  
 957: {id : '970', nome : "Colchões"},  
 958: {id : '971', nome : "Cobertores e Tecidos Cama, Mesa e Banho / Travesseiros"},  
 959: {id : '972', nome : "Congelados ou Frios"},  
 960: {id : '973', nome : "Glicerina"},  
 961: {id : '974', nome : "Cogumelo / Champagnion"},  
 962: {id : '975', nome : "Caixas para Entulho"},  
 963: {id : '976', nome : "Conservas"},  
 964: {id : '977', nome : "Cordas e Cabos"},  
 965: {id : '978', nome : "Dióxido de carbono / Anidrido Carbônico / Gás carbônico"},  
 966: {id : '979', nome : "Foscálcio"},  
 967: {id : '980', nome : "Formol"},  
 968: {id : '981', nome : "Lubrificantes, Graxas"},  
 969: {id : '982', nome : "Farelo ou Gérmem de Milho"},  
 970: {id : '983', nome : "Hipocloreto ou Cloreto de Sódio"},  
 971: {id : '984', nome : "Equipamentos de Informática"},  
 972: {id : '985', nome : "Lã de Vidro"},  
 973: {id : '986', nome : "Nitrato de Alumínio"},  
 974: {id : '987', nome : "Lisina"},  
 975: {id : '988', nome : "Betume ou Piche"},  
 976: {id : '989', nome : "Piscinas"},  
 977: {id : '990', nome : "Policloreto de Alumínio"},  
 978: {id : '991', nome : "Polvilho"},  
 979: {id : '992', nome : "Pregos e Grampos"},  
 980: {id : '993', nome : "Material / produtos para Reciclagem"},  
 981: {id : '994', nome : "Proteína de Soja"},  
 982: {id : '995', nome : "Sacaria - Sacos"},  
 983: {id : '996', nome : "Vasilhames"},  
 984: {id : '997', nome : "Vinagre"},  
 985: {id : '998', nome : "Artesanatos"},  
 986: {id : '999', nome : "Biodiesel"},  
 987: {id : '1000', nome : "Bijouterias e Acessórios"},  
 988: {id : '1001', nome : "Biomassas"},  
 989: {id : '1002', nome : "Bobinas de Plástico"},  
 990: {id : '1003', nome : "Borras Algodão / Soja"},  
 991: {id : '1004', nome : "Emulsão"},  
 992: {id : '1005', nome : "Etileno Glicol Mono Butil Éter"},  
 993: {id : '1006', nome : "Carrinhos para Compras"},  
 994: {id : '1007', nome : "Descartáveis"},  
 995: {id : '1008', nome : "Dormentes"},  
 996: {id : '1009', nome : "Embutidos / Enlatados"},  
 997: {id : '1010', nome : "Aparelhos para Ginástica"},  
 998: {id : '1011', nome : "Farelo de Algodão"},  
 999: {id : '1012', nome : "Farelo de Arroz"},  
1000: {id : '1013', nome : "Farelo ou Farinha de Osso"},  
1001: {id : '1014', nome : "Farelo ou Farinha de Carne"},  
1002: {id : '1015', nome : "Feno"},  
1003: {id : '1016', nome : "Fibra de Côco"},  
1004: {id : '1017', nome : "Gôndolas / Prateleiras"},  
1005: {id : '1018', nome : "Dolomita"},  
1006: {id : '1019', nome : "Metanol / Hidroximetano / Álcool metílico / Carbinol"},  
1007: {id : '1020', nome : "Ferro Gusa"},  
1008: {id : '1021', nome : "Nitrogênio ou Azoto"},  
1009: {id : '1022', nome : "Água (Granel Líquido)"},  
1010:// {id : '3000', nome : "VAZIO"},  
1011: {id : '4000', nome : "NÃO SOUBE INFORMAR"},  
1012: {id : '5000', nome : "OUTROS"}  
1013:];

## .\mobile\www\js\propriedadeVeiculo.js

1:var lista\_propriedade\_veiculo = [  
 2: {id : '1', nome : "Alugado/Fretado", simples : true, onibus : true, carga\_cinza : false, carga\_vermelha : false},  
 3: {id : '2', nome : "Empresa", simples : true, onibus : true, carga\_cinza : false, carga\_vermelha : false},  
 4: {id : '3', nome : "Empresa/Frota própria", simples : false, onibus : false, carga\_cinza : true, carga\_vermelha : false},  
 5: {id : '4', nome : "Próprio", simples : true, onibus : true, carga\_cinza : false, carga\_vermelha : false},  
 6: {id : '5', nome : "Próprio, alugado/fretado", simples : false, onibus : false, carga\_cinza : false, carga\_vermelha : false},  
 7: {id : '7', nome : "Próprio/Particular/Autônomo", simples : false, onibus : false, carga\_cinza : false, carga\_vermelha : true},  
 8: {id : '8', nome : "Serviços Públicos", simples : true, onibus : true, carga\_cinza : false, carga\_vermelha : false},  
 9: {id : '9', nome : "Taxi", simples : true, onibus : false, carga\_cinza : false, carga\_vermelha : false},  
 10: {id : '10', nome : "Transportadora", simples : false, onibus : false, carga\_cinza : false, carga\_vermelha : true},  
 11: {id : '11', nome : "Outros", simples : true, onibus : true, carga\_cinza : false, carga\_vermelha : false}  
 12:];

## .\mobile\www\js\util.js

1:var util = {  
 2:  
 3: // Funções para o inicializa  
 4: inicializaSelect : function(nome\_campo, lista) {  
 5: var insert\_inicial = "<option value='-1'>Selecione</option>\n";  
 6: $.each(lista, function(index, item) {  
 7: insert\_inicial += "<option value='" + (index + 1) + "'>" + item + "</option>\n";  
 8: });  
 9: $("#" + nome\_campo).html(insert\_inicial).selectmenu("refresh", true);  
 10: },  
 11:  
 12: inicializaSelectCustom : function(nome\_campo, lista, mensagem) {  
 13: var insert\_inicial = "<option value='-1'>" + mensagem + "</option>\n";  
 14: $.each(lista, function(index, item) {  
 15: insert\_inicial += "<option value='" + (index + 1) + "'>" + item + "</option>\n";  
 16: });  
 17: $("#" + nome\_campo).html(insert\_inicial).selectmenu("refresh", true);  
 18: },  
 19:  
 20: inicializaSelectCustomValueAsIndex : function(nome\_campo, lista, mensagem) {  
 21: var insert\_inicial = "<option value='-1'>" + mensagem + "</option>\n";  
 22: $.each(lista, function(index, item) {  
 23: insert\_inicial += "<option value='" + item + "'>" + item + "</option>\n";  
 24: });  
 25: $("#" + nome\_campo).html(insert\_inicial).selectmenu("refresh", true);  
 26: },  
 27:  
 28: inicializaSelectPais : function(nome\_registro, nome\_campo, inclui\_Brasil, mensagem) {  
 29: var lista = paises.listados();  
 30: var insert\_inicial = "";  
 31: if (!inclui\_Brasil) {  
 32: if (util.isEmpty(mensagem)) {  
 33: msgSelecione = "Selecione";  
 34: } else {  
 35: msgSelecione = mensagem;  
 36: }  
 37: insert\_inicial = "<option value='-1'>" + msgSelecione + "</option>\n";  
 38: }  
 39: $.each(lista, function(index, item) {  
 40: if (inclui\_Brasil || (item != 'Brasil')) {  
 41: var select\_Brasil = "";  
 42: if (item == 'Brasil') {  
 43: select\_Brasil = " selected";  
 44: app.setAtributo(nome\_registro, index + 1);  
 45: }  
 46: insert\_inicial += "<option value='" + (index + 1) + "'" + select\_Brasil + ">" + item + "</option>\n";  
 47: }  
 48: });  
 49: $("#" + nome\_campo).html(insert\_inicial).selectmenu("refresh", true);  
 50: },  
 51:  
 52: inicializaSelectMunicipio : function(nome\_campo, uf\_sigla, mensagem) {  
 53: var lista = lista\_municipios[uf\_sigla];  
 54: var insert\_inicial = "<option value='-1'>" + mensagem + "</option>\n";  
 55: $.each(lista, function(index, item) {  
 56: insert\_inicial += "<option value='" + item.id + "'>" + item.nome + "</option>\n";  
 57: });  
 58: $("#" + nome\_campo).html(insert\_inicial).selectmenu("refresh", true);  
 59: },  
 60:  
 61: getIdxArray : function(value, array) {  
 62: return $.inArray(value, array) + 1;  
 63: },  
 64:  
 65: contains : function(value, array) {  
 66: return ($.inArray(value, array) >= 0);  
 67: },  
 68:  
 69: getListaFrequencias : function() {  
 70: var lista\_frequencias = [ 'Dia', 'Semana', 'Mês', 'Ano', 'Eventualmente' ];  
 71: return lista\_frequencias;  
 72: },  
 73:  
 74: inicializaSelectFrequencia : function(tipo\_fluxo) {  
 75: var lista\_frequencias = util.getListaFrequencias();  
 76: var nome\_campo = 'frequencia\_sel\_' + tipo\_fluxo;  
 77: util.inicializaSelect(nome\_campo, lista\_frequencias);  
 78: $("#" + nome\_campo).change(function() {  
 79: var freq\_num = $('#frequencia\_num\_' + tipo\_fluxo);  
 80: if ($(this).val() == util.getIdxArray('Eventualmente', lista\_frequencias)) {  
 81: freq\_num.val(1);  
 82: app.setAtributo('frequenciaQtd', 1);  
 83: }  
 84: });  
 85: },  
 86:  
 87: inicializaSelectCargaRiscoOnu : function(nome\_campo, mensagem, lista) {  
 88: var insert\_inicial = "<option value='-1'>" + mensagem + "</option>\n";  
 89: $.each(lista, function(index, item) {  
 90: insert\_inicial += "<option value='" + item.id + "'>" + item.numeroid + "</option>\n";  
 91: });  
 92: $("#" + nome\_campo).html(insert\_inicial).selectmenu("refresh", true);  
 93: },  
 94:  
 95: inicializaSelectTipoDeCarga : function(nome\_campo, mensagem, lista) {  
 96: var insert\_inicial = "<option value='-1'>" + mensagem + "</option>\n";  
 97: $.each(lista, function(index, item) {  
 98: insert\_inicial += "<option value='" + item.id + "'>" + item.nomeProduto + "</option>\n";  
 99: });  
 100: $("#" + nome\_campo).html(insert\_inicial).selectmenu("refresh", true);  
 101: },  
 102:  
 103: inicializaPlacas : function(tipo\_fluxo) {  
 104: // $('.input\_placa').keyup(function() {  
 105: // var input = $(this);  
 106: // input.val(input.val().toUpperCase());  
 107: // });  
 108: $('#placa\_letras\_' + tipo\_fluxo).keyup(function() {  
 109: var input = $(this);  
 110: var regex = new RegExp("[^a-zA-Z]+");  
 111: var value = null;  
 112: if (!util.isEmpty(input.val())) {  
 113: input.val(input.val().replace(regex, ''));  
 114: value = input.val();  
 115: }  
 116: if (util.isEmpty(value) || value.length < Number(input.attr("maxlength"))) {  
 117: $('#grupo\_placa\_numeros\_' + tipo\_fluxo).hide();  
 118: $('#placa\_numeros\_' + tipo\_fluxo).val("");  
 119: // app.setAtributo('placa\_numeros', null);  
 120: } else {  
 121: app.setAtributo('placa\_letras', input.val());  
 122: $('#grupo\_placa\_numeros\_' + tipo\_fluxo).show();  
 123: $('#placa\_letras' + tipo\_fluxo).trigger('change');  
 124: $('#placa\_numeros\_' + tipo\_fluxo).focus();  
 125: }  
 126: // $('#placa\_letras\_' + tipo\_fluxo).change(function() {  
 127: // app.setAtributo('placa\_letras', $(this).val());  
 128: // });  
 129: });  
 130: $('#placa\_numeros\_' + tipo\_fluxo).keyup(function() {  
 131: var input = $(this);  
 132: var regex = new RegExp("[^0-9]+");  
 133: input.val(input.val().replace(regex, ''));  
 134: var max\_len = Number(input.attr("maxlength"));  
 135: if (input.val().length >= max\_len) {  
 136: input.val(input.val().substring(0, max\_len));  
 137: $('#placa\_numeros\_' + tipo\_fluxo).trigger('change');  
 138: // app.setAtributo('placa\_numeros', input.val());  
 139: }  
 140: });  
 141: $('#placa\_unica\_' + tipo\_fluxo).attr("maxlength","20");  
 142: },  
 143:  
 144: inicializaTabelaAuxiliar : function(nome\_campo, mensagem, lista\_tb\_aux, nome\_fluxo) {  
 145: var insert\_inicial = "<option value='-1'>" + mensagem + "</option>\n";  
 146: $.each(lista\_tb\_aux, function(index, item) {  
 147: if (nome\_fluxo == "simples" && item.simples) {  
 148: insert\_inicial += "<option value='" + item.id + "'>" + item.nome + "</option>\n";  
 149: }  
 150: if (nome\_fluxo == "onibus" && item.onibus) {  
 151: insert\_inicial += "<option value='" + item.id + "'>" + item.nome + "</option>\n";  
 152: }  
 153: if (nome\_fluxo == "carga" && item.carga) {  
 154: insert\_inicial += "<option value='" + item.id + "'>" + item.nome + "</option>\n";  
 155: }  
 156: if (nome\_fluxo == "carga\_cinza" && item.carga\_cinza) {  
 157: insert\_inicial += "<option value='" + item.id + "'>" + item.nome + "</option>\n";  
 158: }  
 159: if (nome\_fluxo == "carga\_vermelha" && item.carga\_vermelha) {  
 160: insert\_inicial += "<option value='" + item.id + "'>" + item.nome + "</option>\n";  
 161: }  
 162: if (nome\_fluxo == "moto" && item.moto) {  
 163: insert\_inicial += "<option value='" + item.id + "'>" + item.nome + "</option>\n";  
 164: }  
 165: });  
 166: $("#" + nome\_campo).html(insert\_inicial).selectmenu("refresh", true);  
 167: },  
 168:  
 169: getIdFromTabelaAuxiliar : function(valor, lista\_tb\_aux) {  
 170:   
 171: if (valor == null ) {  
 172: app.logger.log("[ERRO] getIdFromTabelaAuxiliar: valor null");  
 173: }  
 174: if ( lista\_tb\_aux == null ) {  
 175: app.logger.log("[ERRO] getIdFromTabelaAuxiliar: lista\_tb\_aux null");  
 176: }  
 177:   
 178: for ( var item in lista\_tb\_aux) {  
 179: if (lista\_tb\_aux[item].nome == valor) {  
 180: var id = Number(lista\_tb\_aux[item].id);  
 181: if (!isNaN(id)) {  
 182: return Number(id);  
 183: }  
 184: return id;  
 185: }  
 186: }  
 187: return null;  
 188: },  
 189:  
 190: // Funções para o progresso  
 191: /\*\*  
 192: \*   
 193: \* @param nome\_registro  
 194: \* nome do atributo da variável global registro  
 195: \* @param nome\_campo  
 196: \* id do componente html  
 197: \* @param grupo\_proximo  
 198: \* id da div que irá sofrer show/hide  
 199: \*/  
 200: progressoRadioSimNao : function(nome\_registro, nome\_campo, grupo\_proximo) {  
 201: $('#' + nome\_campo + '\_sim').click(function() {  
 202: $('#' + grupo\_proximo).show();  
 203: app.setAtributo(nome\_registro, true);  
 204: });  
 205: $('#' + nome\_campo + '\_nao').click(function() {  
 206: $('#' + grupo\_proximo).show();  
 207: app.setAtributo(nome\_registro, false);  
 208: });  
 209: },  
 210:  
 211: progressoRadioSimNaoAlternado : function(nome\_registro, nome\_campo, grupo\_proximo\_sim, grupo\_proximo\_nao) {  
 212: $('#' + nome\_campo + '\_sim').click(function() {  
 213: $('#' + grupo\_proximo\_sim).show();  
 214: $('#' + grupo\_proximo\_nao).hide();  
 215: app.setAtributo(nome\_registro, true);  
 216: });  
 217: $('#' + nome\_campo + '\_nao').click(function() {  
 218: $('#' + grupo\_proximo\_sim).hide();  
 219: $('#' + grupo\_proximo\_nao).show();  
 220: app.setAtributo(nome\_registro, false);  
 221: });  
 222: },  
 223:  
 224: progressoRadioPlacaEstrangeira : function(tipo\_fluxo) {  
 225: $('#placa\_estrangeira\_' + tipo\_fluxo + '\_sim').click(function() {  
 226: $('#grupo\_pais\_' + tipo\_fluxo).show();  
 227: $('#grupo\_placa\_' + tipo\_fluxo).hide();  
 228: app.setAtributo('placaEstrangeira', true);  
 229: app.setAtributo('placa\_letras', null);  
 230: app.setAtributo('placa\_numeros', null);  
 231: $('#placa\_letras\_' + tipo\_fluxo).val(null);  
 232: $('#placa\_numeros\_' + tipo\_fluxo).val(null);  
 233:  
 234: if (Number($('#pais\_' + tipo\_fluxo).val()) == -1) {  
 235: $("#grupo\_placa\_unica\_" + tipo\_fluxo).hide();  
 236: }  
 237: });  
 238: $('#placa\_estrangeira\_' + tipo\_fluxo + '\_nao').click(function() {  
 239: $('#grupo\_pais\_' + tipo\_fluxo).hide();  
 240: $('#grupo\_placa\_unica\_' + tipo\_fluxo).hide();  
 241: if (util.isEmpty(registro.placaEstrangeira) || registro.placaEstrangeira) {  
 242: $('#grupo\_placa\_numeros\_' + tipo\_fluxo).hide();  
 243: }  
 244: app.setAtributo('idPaisPlacaEstrangeira', null);  
 245: app.setAtributo('placaEstrangeira', false);  
 246: app.setAtributo('placa\_unica', null);  
 247: $('#placa\_unica\_' + tipo\_fluxo).val(null);  
 248: util.progressoRestartSelect("pais\_" + tipo\_fluxo, "Selecione");  
 249: $("#grupo\_placa\_" + tipo\_fluxo).show();  
 250: });  
 251: },  
 252:  
 253: progressoCheckboxAlternado : function(nome\_registro, nome\_campo, grupo\_proximo\_check, grupo\_proximo\_uncheck) {  
 254: $('#' + nome\_campo).click(function() {  
 255: if ($(this).is(':checked')) {  
 256: $('#' + grupo\_proximo\_check).show();  
 257: $('#' + grupo\_proximo\_uncheck).hide();  
 258: app.setAtributo(nome\_registro, true);  
 259: } else {  
 260: $('#' + grupo\_proximo\_check).hide();  
 261: $('#' + grupo\_proximo\_uncheck).show();  
 262: app.setAtributo(nome\_registro, false);  
 263: }  
 264: });  
 265: },  
 266:  
 267: /\*\*  
 268: \*   
 269: \* @param nome\_registro  
 270: \* nome do atributo da variável global registro  
 271: \* @param nome\_campo  
 272: \* id do componente html  
 273: \* @param grupo\_proximo  
 274: \* id da div que irá sofrer show/hide  
 275: \*/  
 276: progressoSelect : function(nome\_registro, nome\_campo, grupo\_proximo) {  
 277: $('#' + nome\_campo).change(function() {  
 278: if (Number($(this).val()) != -1) {  
 279: app.setAtributo(nome\_registro, $(this).val());  
 280: $("#" + grupo\_proximo).show();  
 281: } else {  
 282: $("#" + grupo\_proximo).hide();  
 283: app.setAtributo(nome\_registro, null);  
 284: }  
 285: });  
 286: },  
 287:  
 288: progressoSelectPais : function(nome\_registro\_pais, nome\_registro\_municipio, nome\_campo, proximo\_imediato, proximo\_imediato2,  
 289: grupo\_proximo, fluxo) {  
 290: $('#' + nome\_campo).change(function() {  
 291: grupo\_proximo\_imediato = "grupo\_" + proximo\_imediato + "\_" + fluxo;  
 292: grupo\_proximo\_imediato2 = "grupo\_" + proximo\_imediato2 + "\_" + fluxo;  
 293: if (Number($(this).val()) == -1) {  
 294: $("#" + grupo\_proximo\_imediato).hide();  
 295: $("#" + grupo\_proximo\_imediato2).hide();  
 296: $("#" + grupo\_proximo).hide();  
 297: app.setAtributo(nome\_registro\_pais, null);  
 298: app.setAtributo(proximo\_imediato, null);  
 299: app.setAtributo(nome\_registro\_municipio, null);  
 300: } else if (Number($(this).val()) != 1) { // País diferente de Brasil  
 301: $("#" + grupo\_proximo\_imediato).hide();  
 302: $("#" + grupo\_proximo\_imediato2).hide();  
 303: $("#" + grupo\_proximo).show();  
 304: app.setAtributo(nome\_registro\_pais, $(this).val());  
 305: app.setAtributo(proximo\_imediato, null);  
 306: app.setAtributo(nome\_registro\_municipio, null);  
 307: util.inicializaSelectCustomValueAsIndex(proximo\_imediato + "\_" + fluxo, lista\_estados, "UF");  
 308: $("#" + proximo\_imediato2 + "\_" + fluxo).val("");  
 309: } else { // País é Brasil  
 310: $("#" + grupo\_proximo\_imediato).show();  
 311: $("#" + grupo\_proximo).hide();  
 312: app.setAtributo(nome\_registro\_pais, $(this).val());  
 313: }  
 314: });  
 315: },  
 316:  
 317: progressoRestartSelect : function(nome\_campo, mensagem) {  
 318: $("#" + nome\_campo + " option:contains('" + mensagem + "')").prop({  
 319: selected : true  
 320: });  
 321: $("select#" + nome\_campo).selectmenu("refresh", true);  
 322: },  
 323:  
 324: /\*\*  
 325: \*   
 326: \* @param nome\_registro  
 327: \* nome do atributo da variável global registro  
 328: \* @param nome\_campo  
 329: \* id do componente html  
 330: \* @param grupo\_proximo  
 331: \* id da div que irá sofrer show/hide  
 332: \* @param autocomplete  
 333: \* bool para indicar se é um campo autocomplete  
 334: \*/  
 335: progressoInputText : function(nome\_registro, nome\_campo, grupo\_proximo, autocomplete) {  
 336:  
 337: if (autocomplete) { // autocomplete  
 338: $('#' + nome\_campo).change(function() {  
 339: progride($(this).val());  
 340: app.setAtributo(nome\_registro, $(this).val());  
 341: });  
 342: } else { // textinput comum  
 343: $('#' + nome\_campo).keyup(function() {  
 344: progride($(this).val());  
 345: });  
 346: $('#' + nome\_campo).change(function() {  
 347: app.setAtributo(nome\_registro, $(this).val());  
 348: });  
 349: }  
 350:  
 351: function progride(value) {  
 352: if (util.isEmpty(value)) {  
 353: $("#" + grupo\_proximo).hide();  
 354: } else {  
 355: $("#" + grupo\_proximo).show();  
 356: }  
 357: }  
 358: },  
 359:  
 360: progressoInputTextLen : function(nome\_registro, nome\_campo, grupo\_proximo) {  
 361: $('#' + nome\_campo).keyup(function() {  
 362: var input = $(this);  
 363: var max\_len = Number(input.attr("maxlength"));  
 364: var min\_len = input.attr("minlength");  
 365:   
 366: if(typeof min\_len !== typeof undefined && min\_len !== false){  
 367: min\_len=Number(min\_len);  
 368: }else  
 369: min\_len=max\_len;  
 370:   
 371: if (progride(input.val(),min\_len)) {  
 372: input.val(String(input.val()).trim().substring(0, max\_len));  
 373: input.trigger('change');  
 374: }  
 375: });  
 376: $('#' + nome\_campo).change(function() {  
 377: app.setAtributo(nome\_registro, $(this).val());  
 378: });  
 379:  
 380: function progride(value, min\_len) {  
 381: if (util.isEmpty(value) || (String(value).trim().length < min\_len)) {  
 382: $("#" + grupo\_proximo).hide();  
 383: } else {  
 384: $("#" + grupo\_proximo).show();  
 385: return true;  
 386: }  
 387: return false;  
 388: }  
 389: },  
 390:  
 391: progressoInputMoney : function(nome\_registro, nome\_campo, grupo\_proximo, autocomplete) {  
 392: $('#' + nome\_campo).keyup(function() {  
 393: progride($(this).maskMoney('unmasked')[0]);  
 394: });  
 395: $('#' + nome\_campo).change(function() {  
 396: app.setAtributo(nome\_registro, $(this).maskMoney('unmasked')[0]);  
 397: });  
 398:  
 399: function progride(value) {  
 400: if (util.isEmpty(value) || (Number(value) == 0)) {  
 401: $("#" + grupo\_proximo).hide();  
 402: } else {  
 403: $("#" + grupo\_proximo).show();  
 404: }  
 405: }  
 406: },  
 407:  
 408: progressoPlacaNumeros : function(tipo\_fluxo, grupo\_proximo) {  
 409: util.progressoInputTextLen("placa\_numeros", "placa\_numeros\_" + tipo\_fluxo, grupo\_proximo);  
 410: },  
 411:  
 412: // Funções para validação dos componentes  
 413: validaSelect : function(nome\_campo, campo\_aviso) {  
 414: if (Number($('#' + nome\_campo).val()) != -1) {  
 415: return true;  
 416: } else {  
 417: if (!util.isEmpty(campo\_aviso)) {  
 418: util.alerta\_msg(campo\_aviso);  
 419: }  
 420: return false;  
 421: }  
 422: },  
 423:  
 424: validaRadioSimNao : function(nome\_campo, campo\_aviso) {  
 425: var option = $('input[name=' + nome\_campo + ']:checked').val();  
 426: if (option != null) {  
 427: return true;  
 428: } else {  
 429: util.alerta\_msg(campo\_aviso);  
 430: return false;  
 431: }  
 432: },  
 433:  
 434: validaRadioChecked : function(nome\_campo, campo\_aviso) {  
 435: var option = $('input[name=' + nome\_campo + ']').val();  
 436: if (option != null) {  
 437: return true;  
 438: } else {  
 439: util.alerta\_msg(campo\_aviso);  
 440: return false;  
 441: }  
 442: },  
 443:  
 444: validaInputText : function(nome\_campo, campo\_aviso) {  
 445: var value = $.trim($('#' + nome\_campo).val());  
 446: var max\_len = Number($.trim($('#' + nome\_campo).attr("maxlength")))  
 447: if (value.length > 0) {  
 448: if ((max\_len > 0) && (value.length > max\_len)) {  
 449: if (!util.isEmpty(campo\_aviso)) {  
 450: util.alerta\_msg(campo\_aviso, "O campo deve ter no máximo " + max\_len + " caracteres.");  
 451: }  
 452: return false;  
 453: } else {  
 454: return true;  
 455: }  
 456: } else {  
 457: util.alerta\_msg(campo\_aviso);  
 458: if (!util.isEmpty(campo\_aviso)) {  
 459: return false;  
 460: }  
 461: }  
 462: },  
 463:  
 464: validaLenInputText : function(nome\_campo, campo\_aviso) {  
 465: var value = $.trim($('#' + nome\_campo).val());  
 466: var len = Number($.trim($('#' + nome\_campo).attr("maxlength")));  
 467: var min\_len = $('#' + nome\_campo).attr("minlength");  
 468:   
 469: if(typeof min\_len !== typeof undefined && min\_len !== false){  
 470: min\_len=Number(min\_len);  
 471: }else  
 472: min\_len=len;  
 473:   
 474: if (len > 0) {  
 475: if (value.length < min\_len || value.length > len) {  
 476: if(min\_len===len)  
 477: util.alerta\_msg(campo\_aviso, "O campo deve ter exatamente " + len + " caracteres.");  
 478: else  
 479: util.alerta\_msg(campo\_aviso, "O campo deve ter entre " + min\_len + " e "+len+" caracteres.");  
 480: return false;  
 481: }  
 482: }  
 483: return true;  
 484: },  
 485:  
 486: validaInputNumberRange : function(nome\_campo, campo\_aviso, min, max) {  
 487: var value = $.trim($('#' + nome\_campo).val())  
 488: if ((!util.isEmpty(min) && !util.isEmpty(max) && Number(value) >= min && Number(value) <= max)  
 489: || (!util.isEmpty(min) && util.isEmpty(max) && Number(value) >= min)  
 490: || (!util.isEmpty(max) && util.isEmpty(min) && Number(value) <= max)) {  
 491: return true;  
 492: } else if (!util.isEmpty(min) || !util.isEmpty(max)) {  
 493: var msgComplemento;  
 494: if (!util.isEmpty(min) && !util.isEmpty(max)) {  
 495: if (min >= max) {  
 496: msgComplemento = "O valor do campo não deve ser diferente de " + min;  
 497: } else {  
 498: msgComplemento = "O valor do campo deve estar entre " + min + " e " + max;  
 499: }  
 500: } else if (!util.isEmpty(min)) {  
 501: msgComplemento = "O menor valor possível do campo é " + min;  
 502: } else if (!util.isEmpty(max)) {  
 503: msgComplemento = "O maior valor possível do campo é " + max;  
 504: }  
 505: util.alerta\_msg(campo\_aviso, msgComplemento);  
 506: return false;  
 507: } else {  
 508: return validaInputText(nome\_campo, campo\_aviso);  
 509: }  
 510: },  
 511:  
 512: validaValueInList : function(nome\_campo, campo\_aviso, lista, idRegistro) {  
 513: var value = $.trim($('#' + nome\_campo).val());  
 514: app.setAtributo(idRegistro,value);  
 515:   
 516: var idItemLista = util.findValueInList(value, lista);  
 517: if (util.isEmpty(idItemLista)) {  
 518: util.alerta\_msg(campo\_aviso, "O valor " + value + " informado não é válido.");  
 519: return false;  
 520: }  
 521: return true;  
 522: },  
 523:   
 524: findValueInList : function(valorLista, lista) {  
 525: var idItemLista = null;  
 526: if (!util.isEmpty(valorLista)) {  
 527: var value = valorLista.toUpperCase();  
 528: $.each(lista, function(index, item) {  
 529: if (value == item.numeroid) {  
 530: idItemLista = item.id;  
 531: return false; // funciona como break para o sair do each()  
 532: }  
 533: });  
 534: return idItemLista;  
 535: } else {  
 536: return null;  
 537: }   
 538: },  
 539:   
 540: validaTemPessoasVeiculo : function(nome\_campo, campo\_aviso) {  
 541: if (util.isEmpty($.trim($('#' + nome\_campo).val())) || !($.trim($('#' + nome\_campo).val()) > 0)) {  
 542: util.alerta\_msg(campo\_aviso, "O campo deve ser preenchido com um valor positivo maior que zero");  
 543: return false;  
 544: }  
 545: return true;  
 546: },  
 547:  
 548: validaLimitePessoas : function(nome\_campo, campo\_aviso) {  
 549: var limitePessoas = app.getAtributo('limitePessoas');  
 550: if (!util.isEmpty(limitePessoas)) {  
 551: if ($.trim($('#' + nome\_campo).val()) > Number(limitePessoas)) {  
 552: util.alerta\_msg(campo\_aviso, "O limite de pessoas para esse tipo de veículo é " + limitePessoas);  
 553: return false;  
 554: }  
 555: }  
 556: return true;  
 557: },  
 558:  
 559: limitaTamanhoCampo : function(nome\_campo, nome\_atributo){  
 560: var limite = app.getAtributo(nome\_atributo);  
 561: util.limitaTamanhoCampoPorValor(nome\_campo,limite);  
 562: },  
 563:  
 564: limitaTamanhoCampoPorValor : function(nome\_campo, valor){  
 565: if (!util.isEmpty(valor)) {  
 566: $('#'+nome\_campo).attr("maxlength",String(Number(valor)).length);  
 567: }  
 568: },  
 569:   
 570: validacaoLimiteMaximoTempoReal: function(nome\_campo,campo\_aviso, funcao\_validar){  
 571: $("#"+nome\_campo).change(function(){  
 572: if(funcao\_validar(nome\_campo,campo\_aviso)){  
 573: $(this).removeClass("invalido");  
 574: }else  
 575: $(this).addClass("invalido");  
 576: })  
 577: },  
 578:  
 579: isFilterRunning : false, // controla se o filtro já terminou  
 580:  
 581: autocomplete : function(nome\_do\_campo, lista, title, txt\_content) {  
 582: title = (title == null) ? "Busca" : title;  
 583: txt\_content = (txt\_content == null) ? "Entre com o início da palavra." : txt\_content;  
 584: var field = $('#' + nome\_do\_campo);  
 585: var overlayInput = $.confirm({  
 586: title : title,  
 587: content : txt\_content,  
 588: confirmButton : null,  
 589: cancelButton : null,  
 590: animation : 'none',  
 591: closeAnimation : 'none',  
 592: animationBounce : 1,  
 593: icon : 'fa fa-search',  
 594: closeIcon : true,  
 595: onClose : function() {  
 596: util.isFilterRunning = false;  
 597: },  
 598:  
 599: isCentered : false,  
 600: });  
 601: overlayInput.$body.addClass("ui-page-theme-a");  
 602:  
 603: var extraHtml = '<form class="ui-filterable" > \  
 604: <input id="filtro" type="text" autofocus data-type="search"> \  
 605: </form> \  
 606: <ul id="filtro\_autocomplete" data-role="listview" data-filter="true" data-input="#filtro" data-inset="true"></ul>';  
 607: overlayInput.$content.append(extraHtml);  
 608: var txtInput = overlayInput.$content.find('#filtro');  
 609: app.logger.log(txtInput);  
 610: txtInput.textinput();  
 611: /\*  
 612: \* txtInput.one('load', function(){ $(this).focus(); alert('tentou focar'); });  
 613: \*/  
 614: overlayInput.$content.find('#filtro\_autocomplete').listview();  
 615: overlayInput.$b.css('margin-top', '0px');  
 616:  
 617: $("#filtro\_autocomplete").on("filterablebeforefilter", function(e, data) {  
 618: app.logger.log('entrou no filtro'); // TODO ja podemos excluir essa linha?  
 619: var ul\_list = $(this);  
 620: var auto\_input = $(data.input);  
 621: var auto\_value = auto\_input.val();  
 622: var html = "";  
 623: var minLength = 2;// só pesquisa quando a entrada for maior  
 624: var newText = "";  
 625: var waitCicle = false;  
 626: ul\_list.html("");  
 627:  
 628: if (auto\_value && auto\_value.length > minLength) {  
 629: if (!util.isFilterRunning) {  
 630: util.isFilterRunning = true;  
 631: doFilter(auto\_value);  
 632: util.isFilterRunning = false;  
 633: } else {  
 634: newText = auto\_value;  
 635: if (!waitCicle) {  
 636: waitCicle = setInterval(function() {  
 637: if (!util.isFilterRunning) {  
 638: util.isFilterRunning = true;  
 639: clearInterval(waitCicle);  
 640: waitCicle = false;  
 641: doFilter(newText);  
 642: util.isFilterRunning = false;  
 643: }  
 644: }, 300);  
 645: }  
 646: }  
 647: }  
 648:  
 649: function doFilter(inputedTxt) {  
 650:  
 651: ul\_list.html("<li><div class='ui-loader'><span class='ui-icon ui-icon-loading'></span></div></li>");  
 652: ul\_list.listview("refresh");  
 653: var matcher = new RegExp(util.escapeRegex(inputedTxt), "i");  
 654: var res = $.grep(lista, function(value) {  
 655: value = value.label || value.value || value.nome || value;  
 656: return matcher.test(value) || matcher.test(util.replaceDiacritics(value));  
 657: });  
 658: app.logger.log(JSON.stringify(res), null, '\t');  
 659: $.each(res, function(i, val) {  
 660: if (val.id != null) { // municipio  
 661: html += "<li li\_id='" + val.id + "' >" + val.nome + "</li>";  
 662: } else {  
 663: html += "<li>" + val.nome + "</li>";  
 664: }  
 665: // overlayInput.setDialogDimension();  
 666: });  
 667: ul\_list.html(html);  
 668: ul\_list.children("li").click(function() {  
 669: var txt;  
 670: if ($(this).attr('li\_id') != null) { // municipio  
 671: txt = $(this).html() + " | " + $(this).attr('li\_id');  
 672: } else {  
 673: txt = $(this).html();  
 674: }  
 675: field.val(txt).trigger('change').scrollLeft(0);  
 676: // overlayInput.$content.find('#filtro').focus();  
 677: overlayInput.close();  
 678: });  
 679: ul\_list.listview("refresh");  
 680: ul\_list.trigger("updatelayout");  
 681: }  
 682:  
 683: });  
 684:  
 685: },  
 686:  
 687: // Outras funções  
 688: alerta\_msg : function(campo\_aviso, msgComplemento) {  
 689: if (util.isEmpty(msgComplemento)) {  
 690: alert("O campo " + campo\_aviso + " não foi preenchido.", "Erro no preenchimento!", null, 'error');  
 691: } else {  
 692: alert("O campo " + campo\_aviso + " não foi preenchido corretamente.<br />" + msgComplemento,  
 693: 'Erro no preenchimento!', null, 'error');  
 694: }  
 695: },  
 696:  
 697: isEmpty : function(valor) {  
 698: try {  
 699: emp = (valor == undefined) || (valor == null) || (String(valor).trim().length == 0);  
 700:// if (emp) {  
 701:// app.logger.log("isEmpty is TRUE");  
 702:// }  
 703: return emp;  
 704: } catch(exc) {  
 705: app.logger.log("[ERRO] Erro no isEmpty. Detalhes:");  
 706: app.logger.log(exc);  
 707: return null;  
 708: }  
 709: },  
 710:  
 711: isFunction : function(functionToCheck) {  
 712: var getType = {};  
 713: return functionToCheck && getType.toString.call(functionToCheck) === '[object Function]';  
 714: },  
 715:  
 716: getTimeFormated : function(date, format) {  
 717: return moment(date).format(format);  
 718: },  
 719:  
 720: getTimeDefaultFormated : function(date) {  
 721: return util.getTimeFormated(date, "YYYY-MM-DD HH:mm:ss");  
 722: },  
 723:  
 724: getTimeUnixTimestamp : function(date) {  
 725: return util.getTimeFormated(date, "X");  
 726: },  
 727:  
 728: getTimeToExportFormated : function(date) {  
 729: return util.getTimeFormated(date, "YYYY-MM-DD\_HH-mm-ss");  
 730: },  
 731:  
 732: alphabet : {  
 733: a : /[\u0061\u24D0\uFF41\u1E9A\u00E0\u00E1\u00E2\u1EA7\u1EA5\u1EAB\u1EA9\u00E3\u0101\u0103\u1EB1\u1EAF\u1EB5\u1EB3\u0227\u01E1\u00E4\u01DF\u1EA3\u00E5\u01FB\u01CE\u0201\u0203\u1EA1\u1EAD\u1EB7\u1E01\u0105\u2C65\u0250]/ig,  
 734: aa : /[\uA733]/ig,  
 735: ae : /[\u00E6\u01FD\u01E3]/ig,  
 736: ao : /[\uA735]/ig,  
 737: au : /[\uA737]/ig,  
 738: av : /[\uA739\uA73B]/ig,  
 739: ay : /[\uA73D]/ig,  
 740: b : /[\u0062\u24D1\uFF42\u1E03\u1E05\u1E07\u0180\u0183\u0253]/ig,  
 741: c : /[\u0063\u24D2\uFF43\u0107\u0109\u010B\u010D\u00E7\u1E09\u0188\u023C\uA73F\u2184]/ig,  
 742: d : /[\u0064\u24D3\uFF44\u1E0B\u010F\u1E0D\u1E11\u1E13\u1E0F\u0111\u018C\u0256\u0257\uA77A]/ig,  
 743: dz : /[\u01F3\u01C6]/ig,  
 744: e : /[\u0065\u24D4\uFF45\u00E8\u00E9\u00EA\u1EC1\u1EBF\u1EC5\u1EC3\u1EBD\u0113\u1E15\u1E17\u0115\u0117\u00EB\u1EBB\u011B\u0205\u0207\u1EB9\u1EC7\u0229\u1E1D\u0119\u1E19\u1E1B\u0247\u025B\u01DD]/ig,  
 745: f : /[\u0066\u24D5\uFF46\u1E1F\u0192\uA77C]/ig,  
 746: g : /[\u0067\u24D6\uFF47\u01F5\u011D\u1E21\u011F\u0121\u01E7\u0123\u01E5\u0260\uA7A1\u1D79\uA77F]/ig,  
 747: h : /[\u0068\u24D7\uFF48\u0125\u1E23\u1E27\u021F\u1E25\u1E29\u1E2B\u1E96\u0127\u2C68\u2C76\u0265]/ig,  
 748: hv : /[\u0195]/ig,  
 749: i : /[\u0069\u24D8\uFF49\u00EC\u00ED\u00EE\u0129\u012B\u012D\u00EF\u1E2F\u1EC9\u01D0\u0209\u020B\u1ECB\u012F\u1E2D\u0268\u0131]/ig,  
 750: j : /[\u006A\u24D9\uFF4A\u0135\u01F0\u0249]/ig,  
 751: k : /[\u006B\u24DA\uFF4B\u1E31\u01E9\u1E33\u0137\u1E35\u0199\u2C6A\uA741\uA743\uA745\uA7A3]/ig,  
 752: l : /[\u006C\u24DB\uFF4C\u0140\u013A\u013E\u1E37\u1E39\u013C\u1E3D\u1E3B\u017F\u0142\u019A\u026B\u2C61\uA749\uA781\uA747]/ig,  
 753: lj : /[\u01C9]/ig,  
 754: m : /[\u006D\u24DC\uFF4D\u1E3F\u1E41\u1E43\u0271\u026F]/ig,  
 755: n : /[\u006E\u24DD\uFF4E\u01F9\u0144\u00F1\u1E45\u0148\u1E47\u0146\u1E4B\u1E49\u019E\u0272\u0149\uA791\uA7A5]/ig,  
 756: nj : /[\u01CC]/ig,  
 757: o : /[\u006F\u24DE\uFF4F\u00F2\u00F3\u00F4\u1ED3\u1ED1\u1ED7\u1ED5\u00F5\u1E4D\u022D\u1E4F\u014D\u1E51\u1E53\u014F\u022F\u0231\u00F6\u022B\u1ECF\u0151\u01D2\u020D\u020F\u01A1\u1EDD\u1EDB\u1EE1\u1EDF\u1EE3\u1ECD\u1ED9\u01EB\u01ED\u00F8\u01FF\u0254\uA74B\uA74D\u0275]/ig,  
 758: oi : /[\u01A3]/ig,  
 759: ou : /[\u0223]/ig,  
 760: oo : /[\uA74F]/ig,  
 761: p : /[\u0070\u24DF\uFF50\u1E55\u1E57\u01A5\u1D7D\uA751\uA753\uA755]/ig,  
 762: q : /[\u0071\u24E0\uFF51\u024B\uA757\uA759]/ig,  
 763: r : /[\u0072\u24E1\uFF52\u0155\u1E59\u0159\u0211\u0213\u1E5B\u1E5D\u0157\u1E5F\u024D\u027D\uA75B\uA7A7\uA783]/ig,  
 764: s : /[\u0073\u24E2\uFF53\u015B\u1E65\u015D\u1E61\u0161\u1E67\u1E63\u1E69\u0219\u015F\u023F\uA7A9\uA785\u1E9B]/ig,  
 765: ss : /[\u00DF\u1E9E]/ig,  
 766: t : /[\u0074\u24E3\uFF54\u1E6B\u1E97\u0165\u1E6D\u021B\u0163\u1E71\u1E6F\u0167\u01AD\u0288\u2C66\uA787]/ig,  
 767: tz : /[\uA729]/ig,  
 768: u : /[\u0075\u24E4\uFF55\u00F9\u00FA\u00FB\u0169\u1E79\u016B\u1E7B\u016D\u00FC\u01DC\u01D8\u01D6\u01DA\u1EE7\u016F\u0171\u01D4\u0215\u0217\u01B0\u1EEB\u1EE9\u1EEF\u1EED\u1EF1\u1EE5\u1E73\u0173\u1E77\u1E75\u0289]/ig,  
 769: v : /[\u0076\u24E5\uFF56\u1E7D\u1E7F\u028B\uA75F\u028C]/ig,  
 770: vy : /[\uA761]/ig,  
 771: w : /[\u0077\u24E6\uFF57\u1E81\u1E83\u0175\u1E87\u1E85\u1E98\u1E89\u2C73]/ig,  
 772: x : /[\u0078\u24E7\uFF58\u1E8B\u1E8D]/ig,  
 773: y : /[\u0079\u24E8\uFF59\u1EF3\u00FD\u0177\u1EF9\u0233\u1E8F\u00FF\u1EF7\u1E99\u1EF5\u01B4\u024F\u1EFF]/ig,  
 774: z : /[\u007A\u24E9\uFF5A\u017A\u1E91\u017C\u017E\u1E93\u1E95\u01B6\u0225\u0240\u2C6C\uA763]/ig,  
 775: '' : /[\u0300\u0301\u0302\u0303\u0308]/ig  
 776: },  
 777:  
 778: replaceDiacritics : function(str) {  
 779: for ( var letter in util.alphabet) {  
 780: str = str.replace(util.alphabet[letter], letter);  
 781: }  
 782: return str;  
 783: },  
 784:  
 785: escapeRegex : function(value) {  
 786: return value.replace(/[\-\[\]{}()\*+?.,\\\^$|#\s]/g, "\\$&");  
 787: },  
 788:  
 789: reverse : function(s) {  
 790: var o = [];  
 791: for (var i = 0, len = s.length; i <= len; i++)  
 792: o.push(s.charAt(len - i));  
 793: return o.join('');  
 794: },  
 795:  
 796: secToFrase : function (d,extended) {  
 797: var sec\_num = parseInt(d, 10);  
 798: var hours = Math.floor(sec\_num / 3600);  
 799: var minutes = Math.floor((sec\_num - (hours \* 3600)) / 60);  
 800: var seconds = sec\_num - (hours \* 3600) - (minutes \* 60);  
 801: var frase = "";  
 802:   
 803: if(extended == true) {  
 804: if(hours > 0) {frase = frase + hours + " hora(s), ";}  
 805: if(minutes > 0) {frase = frase + minutes + " minuto(s) e ";}  
 806: if(seconds > 0) {frase = frase + seconds + " segundos";}  
 807: } else {  
 808: if(hours > 0) {frase = frase + hours + "h";}  
 809: if(minutes > 0) {frase = frase + minutes + "m";}  
 810: if(seconds > 0) {frase = frase + seconds + "s";}  
 811: }  
 812:   
 813: return frase;  
 814: },  
 815:  
 816: formatDateSumario : function (fulldate) {  
 817: var d = new Date(fulldate.replace(/-/g, '/'));  
 818: return util.formateDateOnly(d);  
 819: },  
 820:  
 821: formatDateAndTimeSumario : function (fulldate) {  
 822: var d = new Date(fulldate.replace(/-/g, '/'));  
 823: hora = d.getHours();  
 824: min = d.getMinutes();  
 825: sec = d.getSeconds();  
 826: if (hora< 10) {hora = "0"+hora;}  
 827: if (min< 10) {min = "0"+min;}  
 828: if (sec< 10) {sec = "0"+sec;}  
 829:  
 830: return util.formateDateOnly(d) + ', às ' + hora + ':' + min + ':' + sec;  
 831: },  
 832:  
 833: formateDateOnly : function (d) {  
 834: dia = d.getDate();  
 835: mes = d.getMonth() + 1;  
 836: ano = d.getFullYear();  
 837: if (dia < 10) {dia = "0"+dia;}  
 838: if (mes< 10) {mes = "0"+mes;}  
 839:  
 840: return dia + '/' + mes + '/' + ano;  
 841: },  
 842:   
 843: formatDateTimeToDisplay : function(){  
 844: var d = new Date();  
 845: var hora = d.getHours();  
 846: var min = d.getMinutes();  
 847: if (hora< 10) {hora = "0"+hora;}  
 848: if (min< 10) {min = "0"+min;}  
 849: return util.formateDateOnly(d) + ' ' + hora + ':' + min;  
 850: },  
 851:   
 852: geraListaAnos : function(){  
 853: var lista\_anos = [];  
 854: var ano = new Date().getFullYear();  
 855: while (ano > 1899){  
 856: lista\_anos = lista\_anos.concat(ano);  
 857: ano--;  
 858: }  
 859: return lista\_anos;  
 860: },  
 861:};

## .\concentrador\novos\textunderscore od\textunderscore 1.json

1:[{"tipo":"p1","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":0,"id":"0010011463271193","frequenciaPeriodo":"Eventualmente","idTipoCarroceria":0,"frequenciaQtd":1,"idMotivoDaViagem":6,"numeroDePessoasATrabalho":3,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":5478,"indoPegarCarga":0,"idRendaMedia":6,"possuiCargaPerigosa":0,"idOrigemMunicipio":5475,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:13:13","idOrigemPais":1,"valorDaCarga":0,"numeroDePessoasNoVeiculo":3,"idNumeroDaOnu":0,"idCombustivel":1,"idPosto":211,"placaVermelha":0,"placa":"ERF-2345","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":0,"categoria":"Leve","possuiReboque":0,"anoDeFabricacao":2010,"idDestinoPais":1,"idPaisPlacaEstrangeira":0,"idCargaAnterior":0,"tipoCaminhao":"null","dataFimPesq":"2016-05-14 21:14:51"}]

## .\concentrador\novos\textunderscore volumetrica.json

1:[{"Pesquisador1":"6","Posto":1,"Hora":"4","Local":"outro","Data":"2016-05-21","DadosVolumetrica":  
 2:[{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"P1"},  
 3:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"P3"},  
 4:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"P2"},  
 5:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"M"},  
 6:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"O1"},  
 7:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"O2"},  
 8:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"O3"},  
 9:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C1"},  
 10:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C2"},  
 11:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C3"},  
 12:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C4"},  
 13:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C5"},  
 14:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S1"},  
 15:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S2"},  
 16:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S3"},  
 17:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S4"},  
 18:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S5"},  
 19:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S6"},  
 20:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE1"},  
 21:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE2"},  
 22:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE3"},  
 23:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE4"},  
 24:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE5"},  
 25:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R1"},  
 26:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R2"},  
 27:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R3"},  
 28:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R4"},  
 29:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R5"},  
 30:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R6"}],"Pesquisador2":"4","Pista":"D","Sentido":"BA"}]

## .\concentrador\novos\textunderscore volumetrica\textunderscore 1.json

1:[{"Pesquisador1":"primeiro","Posto":1,"Hora":"2","Local":"qualquer","Data":"2016-05-15","DadosVolumetrica":  
 2:[{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"P1"},  
 3:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"P3"},  
 4:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"P2"},  
 5:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"M"},  
 6:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"O1"},  
 7:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"O2"},  
 8:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"O3"},  
 9:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C1"},  
 10:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C2"},  
 11:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C3"},  
 12:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C4"},  
 13:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C5"},  
 14:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"S1"},  
 15:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"S2"},  
 16:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"S3"},  
 17:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"S4"},  
 18:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"S5"},  
 19:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"S6"},  
 20:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE1"},  
 21:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE2"},  
 22:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE3"},  
 23:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE4"},  
 24:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE5"},  
 25:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R1"},  
 26:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R2"},  
 27:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R3"},  
 28:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R4"},  
 29:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R5"},  
 30:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R6"}],"Pesquisador2":"segundo","Pista":"S","Sentido":"AB"},  
 31:{"Pesquisador1":"tenente","Posto":1,"Hora":"2","Local":"outro local","Data":"2016-05-15","DadosVolumetrica":  
 32:[{"Hora1":4,"Valores":"1,1,1,1,1,1,1,1","Hora2":4,"Veiculo":"P1"},  
 33:{"Hora1":4,"Valores":"1,1,1,1,1,1,1,1","Hora2":4,"Veiculo":"P3"},  
 34:{"Hora1":8,"Valores":"2,2,2,2,2,2,2,2","Hora2":8,"Veiculo":"P2"},  
 35:{"Hora1":8,"Valores":"2,2,2,2,2,2,2,2","Hora2":8,"Veiculo":"M"},  
 36:{"Hora1":3,"Valores":"2,1,0,0,0,0,0,0","Hora2":0,"Veiculo":"O1"},  
 37:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"O2"},  
 38:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"O3"},  
 39:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C1"},  
 40:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C2"},  
 41:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C3"},  
 42:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C4"},  
 43:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"C5"},  
 44:{"Hora1":84,"Valores":"21,21,21,21,21,21,21,21","Hora2":84,"Veiculo":"S1"},  
 45:{"Hora1":84,"Valores":"21,21,21,21,21,21,21,21","Hora2":84,"Veiculo":"S2"},  
 46:{"Hora1":12,"Valores":"3,3,3,3,3,33,3,3","Hora2":42,"Veiculo":"S3"},  
 47:{"Hora1":12,"Valores":"3,3,3,3,3,3,3,3","Hora2":12,"Veiculo":"S4"},  
 48:{"Hora1":12,"Valores":"3,3,3,3,3,3,3,3","Hora2":12,"Veiculo":"S5"},  
 49:{"Hora1":12,"Valores":"3,3,3,3,3,3,33,0","Hora2":39,"Veiculo":"S6"},  
 50:{"Hora1":3,"Valores":"3,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE1"},  
 51:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE2"},  
 52:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE3"},  
 53:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE4"},  
 54:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"SE5"},  
 55:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R1"},  
 56:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R2"},  
 57:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R3"},  
 58:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R4"},  
 59:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R5"},  
 60:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veiculo":"R6"}],"Pesquisador2":"capitao","Pista":"D","Sentido":"BA"}]

## .\concentrador\todos\textunderscore od.json

1:[{"tipo":"p1","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":0,"idNumeroDeRisco":0,"id":"0010011463271193","frequenciaPeriodo":"Eventualmente","idTipoCarroceria":0,"frequenciaQtd":1,"idMotivoDaViagem":6,"numeroDePessoasATrabalho":3,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":5478,"indoPegarCarga":0,"idRendaMedia":6,"possuiCargaPerigosa":0,"idOrigemMunicipio":5475,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:13:13","idOrigemPais":1,"valorDaCarga":0,"numeroDePessoasNoVeiculo":3,"idNumeroDaOnu":0,"idCombustivel":1,"idPosto":211,"placaVermelha":0,"placa":"ERF-2345","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":0,"categoria":"Leve","possuiReboque":0,"anoDeFabricacao":2010,"idDestinoPais":1,"idPaisPlacaEstrangeira":0,"idCargaAnterior":0,"tipoCaminhao":"null","dataFimPesq":"2016-05-14 21:14:51"},  
 2:{"tipo":"p2","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":0,"idNumeroDeRisco":0,"id":"0010011463271294","frequenciaPeriodo":"Ano","idTipoCarroceria":0,"frequenciaQtd":2,"idMotivoDaViagem":8,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":4425,"indoPegarCarga":0,"idRendaMedia":5,"possuiCargaPerigosa":0,"idOrigemMunicipio":0,"idMotivoDeEscolhaDaRota":6,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:14:54","idOrigemPais":4,"valorDaCarga":0,"numeroDePessoasNoVeiculo":9,"idNumeroDaOnu":0,"idCombustivel":5,"idPosto":211,"placaVermelha":0,"placa":"TRUE","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":0,"categoria":"Leve","possuiReboque":0,"anoDeFabricacao":2013,"idDestinoPais":1,"idPaisPlacaEstrangeira":3,"idCargaAnterior":0,"tipoCaminhao":"null","dataFimPesq":"2016-05-14 21:16:37"},  
 3:{"tipo":"m","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":4,"estaNoNote":0,"idNumeroDeRisco":0,"id":"0010011463271413","frequenciaPeriodo":"Semana","idTipoCarroceria":0,"frequenciaQtd":3,"idMotivoDaViagem":6,"numeroDePessoasATrabalho":2,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":0,"indoPegarCarga":0,"idRendaMedia":6,"possuiCargaPerigosa":0,"idOrigemMunicipio":0,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:16:53","idOrigemPais":5,"valorDaCarga":0,"numeroDePessoasNoVeiculo":2,"idNumeroDaOnu":0,"idCombustivel":4,"idPosto":211,"placaVermelha":0,"placa":"DDE-4932","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":0,"categoria":"Moto","possuiReboque":0,"anoDeFabricacao":2014,"idDestinoPais":9,"idPaisPlacaEstrangeira":0,"idCargaAnterior":0,"tipoCaminhao":"null","dataFimPesq":"2016-05-14 21:18:56"},  
 4:{"tipo":"o1","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":0,"idNumeroDeRisco":0,"id":"0010011463271542","frequenciaPeriodo":"Semana","idTipoCarroceria":0,"frequenciaQtd":4,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":1,"idDestinoMunicipio":5991,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":6860,"idMotivoDeEscolhaDaRota":0,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:19:02","idOrigemPais":1,"valorDaCarga":0,"numeroDePessoasNoVeiculo":56,"idNumeroDaOnu":0,"idCombustivel":0,"idPosto":211,"placaVermelha":0,"placa":"RTI-2476","placaEstrangeira":0,"idTipoDeViagemOuServico":3,"municipioDesembarqueCarga":0,"categoria":"Onibus","possuiReboque":0,"anoDeFabricacao":2016,"idDestinoPais":1,"idPaisPlacaEstrangeira":0,"idCargaAnterior":0,"tipoCaminhao":"null","dataFimPesq":"2016-05-14 21:21:29"},  
 5:{"tipo":"o3","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":1,"estaNoNote":0,"idNumeroDeRisco":0,"id":"0010011463271701","frequenciaPeriodo":"Dia","idTipoCarroceria":0,"frequenciaQtd":1,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":11,"idDestinoMunicipio":0,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":0,"idMotivoDeEscolhaDaRota":0,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:21:41","idOrigemPais":13,"valorDaCarga":0,"numeroDePessoasNoVeiculo":45,"idNumeroDaOnu":0,"idCombustivel":0,"idPosto":211,"placaVermelha":0,"placa":"FALSE","placaEstrangeira":0,"idTipoDeViagemOuServico":1,"municipioDesembarqueCarga":0,"categoria":"Onibus","possuiReboque":0,"anoDeFabricacao":1900,"idDestinoPais":2,"idPaisPlacaEstrangeira":13,"idCargaAnterior":0,"tipoCaminhao":"null","dataFimPesq":"2016-05-14 21:22:56"},  
 6:{"tipo":"c1","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":2,"estaNoNote":0,"idNumeroDeRisco":0,"id":"0010011463271801","frequenciaPeriodo":"Semana","idTipoCarroceria":4,"frequenciaQtd":5,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":0,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":2265,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:23:21","idOrigemPais":1,"valorDaCarga":0,"numeroDePessoasNoVeiculo":0,"idNumeroDaOnu":0,"idCombustivel":0,"idPosto":211,"placaVermelha":0,"placa":"ERR-9807","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":0,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2015,"idDestinoPais":8,"idPaisPlacaEstrangeira":0,"idCargaAnterior":0,"tipoCaminhao":"Leve","dataFimPesq":"2016-05-14 21:25:05"},  
 7:{"tipo":"s3","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"ETC12345678","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":1,"estaNoNote":0,"idNumeroDeRisco":0,"id":"0010011463271914","frequenciaPeriodo":"Ano","idTipoCarroceria":6,"frequenciaQtd":7,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":5475,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":10,"idDestinoMunicipio":5472,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":0,"idMotivoDeEscolhaDaRota":9,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:25:14","idOrigemPais":4,"valorDaCarga":0,"numeroDePessoasNoVeiculo":0,"idNumeroDaOnu":0,"idCombustivel":0,"idPosto":211,"placaVermelha":0,"placa":"RIF","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":0,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2016,"idDestinoPais":1,"idPaisPlacaEstrangeira":4,"idCargaAnterior":1,"tipoCaminhao":"Semirreboque","dataFimPesq":"2016-05-14 21:27:51"},  
 8:{"tipo":"se3","valorDoFrete":0.029999999329447746,"idAgregado":0,"idLocalEmbarqueCarga":1,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":400,"idProduto":9,"municipioEmbarqueCarga":1883,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":1,"estaNoNote":0,"idNumeroDeRisco":18,"id":"0010011463272086","frequenciaPeriodo":"Dia","idTipoCarroceria":13,"frequenciaQtd":2,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":5473,"paradaObrigatoriaMunicipio2":6228,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":0,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":0,"idMotivoDeEscolhaDaRota":1,"idLocalDesembarqueCarga":7,"dataIniPesq":"2016-05-14 21:28:06","idOrigemPais":4,"valorDaCarga":0.07999999821186066,"numeroDePessoasNoVeiculo":0,"idNumeroDaOnu":4,"idCombustivel":0,"idPosto":211,"placaVermelha":0,"placa":"FRT-5437","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":6232,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2011,"idDestinoPais":8,"idPaisPlacaEstrangeira":0,"idCargaAnterior":0,"tipoCaminhao":"Semirreboque Especial","dataFimPesq":"2016-05-14 21:43:26"},  
 9:{"tipo":"se3","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"CTC87654321","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":0,"idNumeroDeRisco":27,"id":"0010011463273019","frequenciaPeriodo":"Eventualmente","idTipoCarroceria":9,"frequenciaQtd":1,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":10,"idDestinoMunicipio":0,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":7338,"idMotivoDeEscolhaDaRota":5,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:43:39","idOrigemPais":1,"valorDaCarga":0,"numeroDePessoasNoVeiculo":0,"idNumeroDaOnu":2185,"idCombustivel":0,"idPosto":211,"placaVermelha":0,"placa":"DRTTY","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":0,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2009,"idDestinoPais":7,"idPaisPlacaEstrangeira":2,"idCargaAnterior":782,"tipoCaminhao":"Semirreboque Especial","dataFimPesq":"2016-05-14 21:46:35"},  
 10:{"tipo":"r4","valorDoFrete":7.559999942779541,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"ETC57843276","pesoDaCarga":450,"idProduto":729,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":2,"estaNoNote":0,"idNumeroDeRisco":18,"id":"0010011463273198","frequenciaPeriodo":"Eventualmente","idTipoCarroceria":10,"frequenciaQtd":1,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":4,"idPropriedadesDoVeiculo":7,"idDestinoMunicipio":0,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":0,"idMotivoDeEscolhaDaRota":1,"idLocalDesembarqueCarga":1,"dataIniPesq":"2016-05-14 21:46:38","idOrigemPais":5,"valorDaCarga":9.880000114440918,"numeroDePessoasNoVeiculo":0,"idNumeroDaOnu":4,"idCombustivel":0,"idPosto":211,"placaVermelha":0,"placa":"RTY-5799","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":3790,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2013,"idDestinoPais":8,"idPaisPlacaEstrangeira":0,"idCargaAnterior":0,"tipoCaminhao":"Reboque","dataFimPesq":"2016-05-14 21:51:43"},  
 11:{"tipo":"r1","valorDoFrete":0,"idAgregado":0,"idLocalEmbarqueCarga":0,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":0,"idProduto":0,"municipioEmbarqueCarga":0,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":1,"estaNoNote":0,"idNumeroDeRisco":0,"id":"0010011463273506","frequenciaPeriodo":"Eventualmente","idTipoCarroceria":2,"frequenciaQtd":1,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":0,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":0,"idMotivoDeEscolhaDaRota":3,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:51:46","idOrigemPais":8,"valorDaCarga":0,"numeroDePessoasNoVeiculo":0,"idNumeroDaOnu":0,"idCombustivel":0,"idPosto":211,"placaVermelha":0,"placa":"JHFJHF","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":0,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2013,"idDestinoPais":4,"idPaisPlacaEstrangeira":2,"idCargaAnterior":0,"tipoCaminhao":"Reboque","dataFimPesq":"2016-05-14 21:52:54"},  
 12:{"tipo":"se2","valorDoFrete":474.4700012207031,"idAgregado":0,"idLocalEmbarqueCarga":1,"utilizaParadaEspecial":0,"rntrc":"null","pesoDaCarga":4500,"idProduto":165,"municipioEmbarqueCarga":5486,"cancelado":0,"idIpad":"001001","sentido":"AB","idPerguntaExtra":2,"estaNoNote":0,"idNumeroDeRisco":18,"id":"0010011463273577","frequenciaPeriodo":"Mês","idTipoCarroceria":12,"frequenciaQtd":3,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":3012,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":0,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":5494,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":6586,"idMotivoDeEscolhaDaRota":3,"idLocalDesembarqueCarga":0,"dataIniPesq":"2016-05-14 21:52:57","idOrigemPais":1,"valorDaCarga":6.47473728E8,"numeroDePessoasNoVeiculo":0,"idNumeroDaOnu":4,"idCombustivel":0,"idPosto":211,"placaVermelha":0,"placa":"JHF-7647","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":0,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2011,"idDestinoPais":1,"idPaisPlacaEstrangeira":0,"idCargaAnterior":0,"tipoCaminhao":"Semirreboque Especial","dataFimPesq":"2016-05-14 21:55:42"},  
 13:{"tipo":"c4","valorDoFrete":3.448372736E9,"idAgregado":0,"idLocalEmbarqueCarga":7,"utilizaParadaEspecial":0,"rntrc":"ETC34543900","pesoDaCarga":384.70001220703125,"idProduto":18,"municipioEmbarqueCarga":5471,"cancelado":0,"idIpad":"001001","sentido":"BA","idPerguntaExtra":0,"estaNoNote":0,"idNumeroDeRisco":0,"id":"0010011463273868","frequenciaPeriodo":"Eventualmente","idTipoCarroceria":10,"frequenciaQtd":1,"idMotivoDaViagem":0,"numeroDePessoasATrabalho":0,"paradaObrigatoriaMunicipio1":0,"paradaObrigatoriaMunicipio2":0,"idTipoDeContainer":2,"idPropriedadesDoVeiculo":10,"idDestinoMunicipio":0,"indoPegarCarga":0,"idRendaMedia":0,"possuiCargaPerigosa":0,"idOrigemMunicipio":0,"idMotivoDeEscolhaDaRota":7,"idLocalDesembarqueCarga":1,"dataIniPesq":"2016-05-14 21:57:48","idOrigemPais":2,"valorDaCarga":3.838384E7,"numeroDePessoasNoVeiculo":0,"idNumeroDaOnu":0,"idCombustivel":0,"idPosto":91,"placaVermelha":0,"placa":"ERT","placaEstrangeira":0,"idTipoDeViagemOuServico":0,"municipioDesembarqueCarga":1609,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2012,"idDestinoPais":13,"idPaisPlacaEstrangeira":5,"idCargaAnterior":0,"tipoCaminhao":"Leve","dataFimPesq":"2016-05-14 22:04:39"}]

## .\concentrador\todos\textunderscore od\textunderscore 8.json

1:[{"tipo":"p1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":null,"frequenciaQtd":1,"idMotivoDaViagem":6,"numeroDePessoasATrabalho":3,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":5478,"indoPegarCarga":null,"idRendaMedia":6,"possuiCargaPerigosa":0,"idOrigemMunicipio":5475,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:13:13","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":3,"idNumeroDaOnu":null,"idCombustivel":1,"idPosto":3,"placaVermelha":0,"placa":"ERF-2345","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Leve","possuiReboque":0,"anoDeFabricacao":2010,"idDestinoPais":1,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:14:51"},  
 2:{"tipo":"p2","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Ano","idTipoCarroceria":null,"frequenciaQtd":2,"idMotivoDaViagem":8,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":4425,"indoPegarCarga":null,"idRendaMedia":5,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":6,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:14:54","idOrigemPais":4,"valorDaCarga":null,"numeroDePessoasNoVeiculo":9,"idNumeroDaOnu":null,"idCombustivel":5,"idPosto":3,"placaVermelha":0,"placa":"TRUE","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Leve","possuiReboque":0,"anoDeFabricacao":2013,"idDestinoPais":1,"idPaisPlacaEstrangeira":3,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:16:37"},  
 3:{"tipo":"m","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":4,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Semana","idTipoCarroceria":null,"frequenciaQtd":3,"idMotivoDaViagem":6,"numeroDePessoasATrabalho":2,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":6,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:16:53","idOrigemPais":5,"valorDaCarga":null,"numeroDePessoasNoVeiculo":2,"idNumeroDaOnu":null,"idCombustivel":4,"idPosto":3,"placaVermelha":0,"placa":"DDE-4932","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Moto","possuiReboque":0,"anoDeFabricacao":2014,"idDestinoPais":9,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:18:56"},  
 4:{"tipo":"o1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Semana","idTipoCarroceria":null,"frequenciaQtd":4,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":1,"idDestinoMunicipio":5991,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":6860,"idMotivoDeEscolhaDaRota":null,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:19:02","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":56,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"RTI-2476","placaEstrangeira":0,"idTipoDeViagemOuServico":3,"municipioDesembarqueCarga":null,"categoria":"Onibus","possuiReboque":0,"anoDeFabricacao":2016,"idDestinoPais":1,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:21:29"},  
 5:{"tipo":"o3","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":1,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Dia","idTipoCarroceria":null,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":11,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":null,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:21:41","idOrigemPais":13,"valorDaCarga":null,"numeroDePessoasNoVeiculo":45,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"FALSE","placaEstrangeira":0,"idTipoDeViagemOuServico":1,"municipioDesembarqueCarga":null,"categoria":"Onibus","possuiReboque":0,"anoDeFabricacao":1900,"idDestinoPais":2,"idPaisPlacaEstrangeira":13,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:22:56"},  
 6:{"tipo":"c1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":2,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Semana","idTipoCarroceria":4,"frequenciaQtd":5,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":null,"indoPegarCarga":0,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":2265,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:23:21","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"ERR-9807","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2015,"idDestinoPais":8,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Leve","dataFimPesq":"2016-05-14 21:25:05"},  
 7:{"tipo":"s3","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":"ETC12345678","pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":1,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Ano","idTipoCarroceria":6,"frequenciaQtd":7,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":5475,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":10,"idDestinoMunicipio":5472,"indoPegarCarga":1,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":9,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:25:14","idOrigemPais":4,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"RIF","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2016,"idDestinoPais":1,"idPaisPlacaEstrangeira":4,"idCargaAnterior":1,"tipoCaminhao":"Semirreboque","dataFimPesq":"2016-05-14 21:27:51"},  
 8:{"tipo":"se3","valorDoFrete":0.03,"idAgregado":null,"idLocalEmbarqueCarga":1,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":400,"idProduto":9,"municipioEmbarqueCarga":1883,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":1,"estaNoNote":1,"idNumeroDeRisco":18,"frequenciaPeriodo":"Dia","idTipoCarroceria":13,"frequenciaQtd":2,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":5473,"paradaObrigatoriaMunicipio2":6228,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":1,"idLocalDesembarqueCarga":7,"dataIniPesq":"2016-05-14 21:28:06","idOrigemPais":4,"valorDaCarga":0.08,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":4,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"FRT-5437","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":6232,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2011,"idDestinoPais":8,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Semirreboque Especial","dataFimPesq":"2016-05-14 21:43:26"},  
 9:{"tipo":"se3","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":"CTC87654321","pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":27,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":9,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":10,"idDestinoMunicipio":null,"indoPegarCarga":1,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":7338,"idMotivoDeEscolhaDaRota":5,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:43:39","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":2185,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"DRTTY","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2009,"idDestinoPais":7,"idPaisPlacaEstrangeira":2,"idCargaAnterior":782,"tipoCaminhao":"Semirreboque Especial","dataFimPesq":"2016-05-14 21:46:35"},  
 10:{"tipo":"r4","valorDoFrete":7.56,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":"ETC57843276","pesoDaCarga":450,"idProduto":729,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":2,"estaNoNote":1,"idNumeroDeRisco":18,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":10,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":4,"idPropriedadesDoVeiculo":7,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":1,"idLocalDesembarqueCarga":1,"dataIniPesq":"2016-05-14 21:46:38","idOrigemPais":5,"valorDaCarga":9.88,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":4,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"RTY-5799","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":3790,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2013,"idDestinoPais":8,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Reboque","dataFimPesq":"2016-05-14 21:51:43"},  
 11:{"tipo":"r1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":1,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":2,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":null,"indoPegarCarga":1,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":3,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:51:46","idOrigemPais":8,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"JHFJHF","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2013,"idDestinoPais":4,"idPaisPlacaEstrangeira":2,"idCargaAnterior":null,"tipoCaminhao":"Reboque","dataFimPesq":"2016-05-14 21:52:54"},  
 12:{"tipo":"se2","valorDoFrete":474.47,"idAgregado":null,"idLocalEmbarqueCarga":1,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":4500,"idProduto":165,"municipioEmbarqueCarga":5486,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":2,"estaNoNote":1,"idNumeroDeRisco":18,"frequenciaPeriodo":"Mês","idTipoCarroceria":12,"frequenciaQtd":3,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":3012,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":5494,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":6586,"idMotivoDeEscolhaDaRota":3,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:52:57","idOrigemPais":1,"valorDaCarga":6.4747373727E8,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":4,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"JHF-7647","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2011,"idDestinoPais":1,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Semirreboque Especial","dataFimPesq":"2016-05-14 21:55:42"},  
 13:{"tipo":"c4","valorDoFrete":3.44837273738E9,"idAgregado":null,"idLocalEmbarqueCarga":7,"utilizaParadaEspecial":0,"rntrc":"ETC34543900","pesoDaCarga":384.7,"idProduto":18,"municipioEmbarqueCarga":5471,"cancelado":0,"idIpad":"002008","sentido":"BA","idPerguntaExtra":null,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":10,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":2,"idPropriedadesDoVeiculo":10,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":7,"idLocalDesembarqueCarga":1,"dataIniPesq":"2016-05-14 21:57:48","idOrigemPais":2,"valorDaCarga":3.838383838E7,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"ERT","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":1609,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2012,"idDestinoPais":13,"idPaisPlacaEstrangeira":5,"idCargaAnterior":null,"tipoCaminhao":"Leve","dataFimPesq":"2016-05-14 22:04:39"},  
 14:{"tipo":"S1","valorDoFrete":543.35,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":8800,"idProduto":1,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"002008","sentido":"AB","idPerguntaExtra":null,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":4,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":5,"idLocalDesembarqueCarga":6,"dataIniPesq":"2016-05-15 00:48:00","idOrigemPais":4,"valorDaCarga":646.57,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":"TTT-5555","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":1609,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2016,"idDestinoPais":12,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Semirreboque","dataFimPesq":"2016-05-15 00:49:18"},  
 15:{"tipo":"C4","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":1,"idIpad":"002008","sentido":"AB","idPerguntaExtra":null,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":null,"idTipoCarroceria":10,"frequenciaQtd":null,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":null,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":null,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 12:33:56","idOrigemPais":null,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":null,"placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":null,"idDestinoPais":null,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Leve","dataFimPesq":"2016-05-15 12:34:24"},  
 16:{"tipo":null,"valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":1,"idIpad":"002008","sentido":"AB","idPerguntaExtra":null,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":null,"idTipoCarroceria":null,"frequenciaQtd":null,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":null,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":null,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 12:34:27","idOrigemPais":null,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":3,"placaVermelha":0,"placa":null,"placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":null,"possuiReboque":0,"anoDeFabricacao":null,"idDestinoPais":null,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-15 12:34:37"}]

## .\concentrador\todos\textunderscore od\textunderscore 9.json

1:[{"tipo":"p1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":null,"frequenciaQtd":1,"idMotivoDaViagem":6,"numeroDePessoasATrabalho":3,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":5478,"indoPegarCarga":null,"idRendaMedia":6,"possuiCargaPerigosa":0,"idOrigemMunicipio":5475,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:13:13","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":3,"idNumeroDaOnu":null,"idCombustivel":1,"idPosto":17,"placaVermelha":0,"placa":"ERF-2345","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Leve","possuiReboque":0,"anoDeFabricacao":2010,"idDestinoPais":1,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:14:51"},  
 2:{"tipo":"p2","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Ano","idTipoCarroceria":null,"frequenciaQtd":2,"idMotivoDaViagem":8,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":4425,"indoPegarCarga":null,"idRendaMedia":5,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":6,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:14:54","idOrigemPais":4,"valorDaCarga":null,"numeroDePessoasNoVeiculo":9,"idNumeroDaOnu":null,"idCombustivel":5,"idPosto":17,"placaVermelha":0,"placa":"TRUE","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Leve","possuiReboque":0,"anoDeFabricacao":2013,"idDestinoPais":1,"idPaisPlacaEstrangeira":3,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:16:37"},  
 3:{"tipo":"m","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":4,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Semana","idTipoCarroceria":null,"frequenciaQtd":3,"idMotivoDaViagem":6,"numeroDePessoasATrabalho":2,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":6,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:16:53","idOrigemPais":5,"valorDaCarga":null,"numeroDePessoasNoVeiculo":2,"idNumeroDaOnu":null,"idCombustivel":4,"idPosto":17,"placaVermelha":0,"placa":"DDE-4932","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Moto","possuiReboque":0,"anoDeFabricacao":2014,"idDestinoPais":9,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:18:56"},  
 4:{"tipo":"o1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Semana","idTipoCarroceria":null,"frequenciaQtd":4,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":1,"idDestinoMunicipio":5991,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":6860,"idMotivoDeEscolhaDaRota":null,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:19:02","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":56,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"RTI-2476","placaEstrangeira":0,"idTipoDeViagemOuServico":3,"municipioDesembarqueCarga":null,"categoria":"Onibus","possuiReboque":0,"anoDeFabricacao":2016,"idDestinoPais":1,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:21:29"},  
 5:{"tipo":"o3","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":1,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Dia","idTipoCarroceria":null,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":11,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":null,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:21:41","idOrigemPais":13,"valorDaCarga":null,"numeroDePessoasNoVeiculo":45,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"FALSE","placaEstrangeira":0,"idTipoDeViagemOuServico":1,"municipioDesembarqueCarga":null,"categoria":"Onibus","possuiReboque":0,"anoDeFabricacao":1900,"idDestinoPais":2,"idPaisPlacaEstrangeira":13,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-14 21:22:56"},  
 6:{"tipo":"c1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":2,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Semana","idTipoCarroceria":4,"frequenciaQtd":5,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":null,"indoPegarCarga":0,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":2265,"idMotivoDeEscolhaDaRota":4,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:23:21","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"ERR-9807","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2015,"idDestinoPais":8,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Leve","dataFimPesq":"2016-05-14 21:25:05"},  
 7:{"tipo":"s3","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":"ETC12345678","pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":1,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Ano","idTipoCarroceria":6,"frequenciaQtd":7,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":5475,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":10,"idDestinoMunicipio":5472,"indoPegarCarga":1,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":9,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:25:14","idOrigemPais":4,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"RIF","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2016,"idDestinoPais":1,"idPaisPlacaEstrangeira":4,"idCargaAnterior":1,"tipoCaminhao":"Semirreboque","dataFimPesq":"2016-05-14 21:27:51"},  
 8:{"tipo":"se3","valorDoFrete":0.03,"idAgregado":null,"idLocalEmbarqueCarga":1,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":400,"idProduto":9,"municipioEmbarqueCarga":1883,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":1,"estaNoNote":1,"idNumeroDeRisco":18,"frequenciaPeriodo":"Dia","idTipoCarroceria":13,"frequenciaQtd":2,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":5473,"paradaObrigatoriaMunicipio2":6228,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":1,"idLocalDesembarqueCarga":7,"dataIniPesq":"2016-05-14 21:28:06","idOrigemPais":4,"valorDaCarga":0.08,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":4,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"FRT-5437","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":6232,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2011,"idDestinoPais":8,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Semirreboque Especial","dataFimPesq":"2016-05-14 21:43:26"},  
 9:{"tipo":"se3","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":"CTC87654321","pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":27,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":9,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":10,"idDestinoMunicipio":null,"indoPegarCarga":1,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":7338,"idMotivoDeEscolhaDaRota":5,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:43:39","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":2185,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"DRTTY","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2009,"idDestinoPais":7,"idPaisPlacaEstrangeira":2,"idCargaAnterior":782,"tipoCaminhao":"Semirreboque Especial","dataFimPesq":"2016-05-14 21:46:35"},  
 10:{"tipo":"r4","valorDoFrete":7.56,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":"ETC57843276","pesoDaCarga":450,"idProduto":729,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":2,"estaNoNote":1,"idNumeroDeRisco":18,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":10,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":4,"idPropriedadesDoVeiculo":7,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":1,"idLocalDesembarqueCarga":1,"dataIniPesq":"2016-05-14 21:46:38","idOrigemPais":5,"valorDaCarga":9.88,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":4,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"RTY-5799","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":3790,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2013,"idDestinoPais":8,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Reboque","dataFimPesq":"2016-05-14 21:51:43"},  
 11:{"tipo":"r1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":1,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":2,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":null,"indoPegarCarga":1,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":3,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:51:46","idOrigemPais":8,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"JHFJHF","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2013,"idDestinoPais":4,"idPaisPlacaEstrangeira":2,"idCargaAnterior":null,"tipoCaminhao":"Reboque","dataFimPesq":"2016-05-14 21:52:54"},  
 12:{"tipo":"se2","valorDoFrete":474.47,"idAgregado":null,"idLocalEmbarqueCarga":1,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":4500,"idProduto":165,"municipioEmbarqueCarga":5486,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":2,"estaNoNote":1,"idNumeroDeRisco":18,"frequenciaPeriodo":"Mês","idTipoCarroceria":12,"frequenciaQtd":3,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":3012,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":5494,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":6586,"idMotivoDeEscolhaDaRota":3,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-14 21:52:57","idOrigemPais":1,"valorDaCarga":6.4747373727E8,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":4,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"JHF-7647","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2011,"idDestinoPais":1,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Semirreboque Especial","dataFimPesq":"2016-05-14 21:55:42"},  
 13:{"tipo":"c4","valorDoFrete":3.44837273738E9,"idAgregado":null,"idLocalEmbarqueCarga":7,"utilizaParadaEspecial":0,"rntrc":"ETC34543900","pesoDaCarga":384.7,"idProduto":18,"municipioEmbarqueCarga":5471,"cancelado":0,"idIpad":"015001","sentido":"BA","idPerguntaExtra":null,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":10,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":2,"idPropriedadesDoVeiculo":10,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":7,"idLocalDesembarqueCarga":1,"dataIniPesq":"2016-05-14 21:57:48","idOrigemPais":2,"valorDaCarga":3.838383838E7,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":91,"placaVermelha":0,"placa":"ERT","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":1609,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2012,"idDestinoPais":13,"idPaisPlacaEstrangeira":5,"idCargaAnterior":null,"tipoCaminhao":"Leve","dataFimPesq":"2016-05-14 22:04:39"},  
 14:{"tipo":"S1","valorDoFrete":543.35,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":8800,"idProduto":1,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":null,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":4,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":5,"idLocalDesembarqueCarga":6,"dataIniPesq":"2016-05-15 00:48:00","idOrigemPais":4,"valorDaCarga":646.57,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":91,"placaVermelha":0,"placa":"TTT-5555","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":1609,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":2016,"idDestinoPais":12,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Semirreboque","dataFimPesq":"2016-05-15 00:49:18"},  
 15:{"tipo":"C4","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":1,"idIpad":"015001","sentido":"AB","idPerguntaExtra":null,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":null,"idTipoCarroceria":10,"frequenciaQtd":null,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":null,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":null,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 12:33:56","idOrigemPais":null,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":0,"placaVermelha":0,"placa":null,"placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":0,"anoDeFabricacao":null,"idDestinoPais":null,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":"Leve","dataFimPesq":"2016-05-15 12:34:24"},  
 16:{"tipo":null,"valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":0,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":1,"idIpad":"015001","sentido":"AB","idPerguntaExtra":null,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":null,"idTipoCarroceria":null,"frequenciaQtd":null,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":null,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":null,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 12:34:27","idOrigemPais":null,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":0,"placaVermelha":0,"placa":null,"placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":null,"possuiReboque":0,"anoDeFabricacao":null,"idDestinoPais":null,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-15 12:34:37"},  
 17:{"tipo":"P1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":null,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":3,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":null,"frequenciaQtd":1,"idMotivoDaViagem":6,"numeroDePessoasATrabalho":9,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":2,"idDestinoMunicipio":4710,"indoPegarCarga":null,"idRendaMedia":6,"possuiCargaPerigosa":null,"idOrigemMunicipio":5484,"idMotivoDeEscolhaDaRota":5,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 16:35:08","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":9,"idNumeroDaOnu":null,"idCombustivel":3,"idPosto":17,"placaVermelha":null,"placa":"RET-6547","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Leve","possuiReboque":0,"anoDeFabricacao":2012,"idDestinoPais":1,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-15 16:37:09"},  
 18:{"tipo":"P1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":null,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":4,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Ano","idTipoCarroceria":null,"frequenciaQtd":2,"idMotivoDaViagem":3,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":11,"idDestinoMunicipio":5493,"indoPegarCarga":null,"idRendaMedia":8,"possuiCargaPerigosa":null,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":6,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 16:37:32","idOrigemPais":2,"valorDaCarga":null,"numeroDePessoasNoVeiculo":7,"idNumeroDaOnu":null,"idCombustivel":1,"idPosto":17,"placaVermelha":null,"placa":"FEDRT","placaEstrangeira":1,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Leve","possuiReboque":1,"anoDeFabricacao":2007,"idDestinoPais":1,"idPaisPlacaEstrangeira":2,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-15 16:38:52"},  
 19:{"tipo":"O1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":null,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":2,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":null,"frequenciaQtd":1,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":4,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":null,"possuiCargaPerigosa":null,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":null,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 17:32:07","idOrigemPais":4,"valorDaCarga":null,"numeroDePessoasNoVeiculo":9999,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":17,"placaVermelha":null,"placa":"TTT-5455","placaEstrangeira":0,"idTipoDeViagemOuServico":3,"municipioDesembarqueCarga":null,"categoria":"Onibus","possuiReboque":null,"anoDeFabricacao":2014,"idDestinoPais":6,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-15 17:33:04"},  
 20:{"tipo":"P1","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":null,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":4,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":null,"frequenciaQtd":1,"idMotivoDaViagem":6,"numeroDePessoasATrabalho":7,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":4,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":3,"possuiCargaPerigosa":null,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":2,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 17:33:08","idOrigemPais":4,"valorDaCarga":null,"numeroDePessoasNoVeiculo":7,"idNumeroDaOnu":null,"idCombustivel":2,"idPosto":17,"placaVermelha":null,"placa":"GGG-5556","placaEstrangeira":0,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Leve","possuiReboque":0,"anoDeFabricacao":2011,"idDestinoPais":9,"idPaisPlacaEstrangeira":null,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-15 17:35:30"},  
 21:{"tipo":"M","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":null,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":2,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Eventualmente","idTipoCarroceria":null,"frequenciaQtd":1,"idMotivoDaViagem":5,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":4,"idDestinoMunicipio":null,"indoPegarCarga":null,"idRendaMedia":4,"possuiCargaPerigosa":null,"idOrigemMunicipio":null,"idMotivoDeEscolhaDaRota":6,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 17:36:11","idOrigemPais":4,"valorDaCarga":null,"numeroDePessoasNoVeiculo":2,"idNumeroDaOnu":null,"idCombustivel":2,"idPosto":17,"placaVermelha":null,"placa":"T","placaEstrangeira":1,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Moto","possuiReboque":0,"anoDeFabricacao":2014,"idDestinoPais":9,"idPaisPlacaEstrangeira":5,"idCargaAnterior":null,"tipoCaminhao":null,"dataFimPesq":"2016-05-15 17:37:28"},  
 22:{"tipo":"R3","valorDoFrete":null,"idAgregado":null,"idLocalEmbarqueCarga":null,"utilizaParadaEspecial":null,"rntrc":null,"pesoDaCarga":null,"idProduto":null,"municipioEmbarqueCarga":null,"cancelado":0,"idIpad":"015001","sentido":"AB","idPerguntaExtra":2,"estaNoNote":1,"idNumeroDeRisco":null,"frequenciaPeriodo":"Semana","idTipoCarroceria":13,"frequenciaQtd":6,"idMotivoDaViagem":null,"numeroDePessoasATrabalho":null,"paradaObrigatoriaMunicipio1":null,"paradaObrigatoriaMunicipio2":null,"idTipoDeContainer":null,"idPropriedadesDoVeiculo":3,"idDestinoMunicipio":6627,"indoPegarCarga":0,"idRendaMedia":null,"possuiCargaPerigosa":0,"idOrigemMunicipio":5967,"idMotivoDeEscolhaDaRota":7,"idLocalDesembarqueCarga":null,"dataIniPesq":"2016-05-15 17:37:36","idOrigemPais":1,"valorDaCarga":null,"numeroDePessoasNoVeiculo":null,"idNumeroDaOnu":null,"idCombustivel":null,"idPosto":17,"placaVermelha":0,"placa":"YTR","placaEstrangeira":1,"idTipoDeViagemOuServico":null,"municipioDesembarqueCarga":null,"categoria":"Pesado","possuiReboque":null,"anoDeFabricacao":2013,"idDestinoPais":1,"idPaisPlacaEstrangeira":4,"idCargaAnterior":null,"tipoCaminhao":"Reboque","dataFimPesq":"2016-05-15 17:38:59"}]

## .\concentrador\todos\textunderscore volumetrica.json

1:[{"Pesquisador1":"a","Posto":1,"Hora":"2","Local":"qualquer","Data":"2016-05-15","DadosVolumetrica":  
 2:[{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"P1"},  
 3:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"P3"},  
 4:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"P2"},  
 5:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"M"},  
 6:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"O1"},  
 7:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"O2"},  
 8:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"O3"},  
 9:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C1"},  
 10:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C2"},  
 11:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C3"},  
 12:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C4"},  
 13:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"C5"},  
 14:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S1"},  
 15:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S2"},  
 16:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S3"},  
 17:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S4"},  
 18:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S5"},  
 19:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"S6"},  
 20:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE1"},  
 21:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE2"},  
 22:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE3"},  
 23:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE4"},  
 24:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"SE5"},  
 25:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R1"},  
 26:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R2"},  
 27:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R3"},  
 28:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R4"},  
 29:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R5"},  
 30:{"Hora1":0,"Valores":"0,0,0,0,0,0,0,0","Hora2":0,"Veículo":"R6"}],"Pesquisador2":"d","Pista":"S","Sentido":"AB"}]

## .\concentrador\src\logins\users.json

1:{  
 2: "logins": [  
 3: {  
 4: "trecho": "ENTR AC-463/465 (BOM FUTURO) - ENTR BR-317",  
 5: "pwd": "1590",  
 6: "ladoA": "",  
 7: "fase": "3",  
 8: "ladoB": "",  
 9: "usr": "001"  
 10: },  
 11: {  
 12: "trecho": "LUCAS DO RIO VERDE – INÍCIO DA TRAVESSIA URBANA DE SORRISO",  
 13: "pwd": "0808",  
 14: "ladoA": "LUCAS DO RIO VERDE",  
 15: "fase": "1",  
 16: "ladoB": "INÍCIO DA TRAVESSIA URBANA DE SORRISO",  
 17: "usr": "002"  
 18: },  
 19: {  
 20: "trecho": "ENTR BR-104(A) - ENTR BR-104(B)",  
 21: "pwd": "0630",  
 22: "ladoA": "",  
 23: "fase": "2",  
 24: "ladoB": "",  
 25: "usr": "003"  
 26: },  
 27: {  
 28: "trecho": "ENTR AL-225(A) - ENTR AL-225(B) (P/PORTO REAL DO COLÉGIO)",  
 29: "pwd": "0890",  
 30: "ladoA": "",  
 31: "fase": "2",  
 32: "ladoB": "",  
 33: "usr": "004"  
 34: },  
 35: {  
 36: "trecho": "ENTR AL-115(A) - ENTR AL-115(B) (PALMEIRA DOS ÍNDIOS)",  
 37: "pwd": "0950",  
 38: "ladoA": "",  
 39: "fase": "3",  
 40: "ladoB": "",  
 41: "usr": "005"  
 42: },  
 43: {  
 44: "trecho": "ENTR BR-101(A) - ENTR BR-101(B) (P/ALAGOINHAS)",  
 45: "pwd": "0750",  
 46: "ladoA": "",  
 47: "fase": "4",  
 48: "ladoB": "",  
 49: "usr": "006"  
 50: },  
 51: {  
 52: "trecho": "ENTR BA-120(A)/262(B) (P/ITAJUÍPE) - ENTR BR-415/BA-120(B) (ITABUNA)",  
 53: "pwd": "1810",  
 54: "ladoA": "",  
 55: "fase": "3",  
 56: "ladoB": "",  
 57: "usr": "007"  
 58: },  
 59: {  
 60: "trecho": "ENTR BR-324(A) - ENTR BR-324(B)/BA-502/503 (FEIRA DE SANTANA)",  
 61: "pwd": "0330",  
 62: "ladoA": "",  
 63: "fase": "2",  
 64: "ladoB": "",  
 65: "usr": "008"  
 66: },  
 67: {  
 68: "trecho": "ENTR BA-643 (PLANALTO) - ENTR BA-641",  
 69: "pwd": "0934",  
 70: "ladoA": "",  
 71: "fase": "3",  
 72: "ladoB": "",  
 73: "usr": "009"  
 74: },  
 75: {  
 76: "trecho": "ENTR BR-242(A) - ENTR BR-330/BA-148 (SEABRA)",  
 77: "pwd": "0365",  
 78: "ladoA": "",  
 79: "fase": "3",  
 80: "ladoB": "",  
 81: "usr": "010"  
 82: },  
 83: {  
 84: "trecho": "JAGUARARI - ENTR BA-220 (SENHOR DO BONFIM)",  
 85: "pwd": "0300",  
 86: "ladoA": "",  
 87: "fase": "2",  
 88: "ladoB": "",  
 89: "usr": "011"  
 90: },  
 91: {  
 92: "trecho": "ENTR CE-187 (P/MARRECOS) - ENTR BR-404/CE-176/187/363 (P/TAUÁ)",  
 93: "pwd": "0530",  
 94: "ladoA": "",  
 95: "fase": "3",  
 96: "ladoB": "",  
 97: "usr": "012"  
 98: },  
 99: {  
 100: "trecho": "ENTR CE-253 (PACAJÚS) - ENTR BR-122(A)/CE-354 (CHORÓZINHO)",  
 101: "pwd": "0090",  
 102: "ladoA": "",  
 103: "fase": "3",  
 104: "ladoB": "",  
 105: "usr": "013"  
 106: },  
 107: {  
 108: "trecho": "ENTR BR-226(B)/CE-275(B) (P/PEREIRO) - ENTR BR-404/434 (ICÓ)",  
 109: "pwd": "0260",  
 110: "ladoA": "",  
 111: "fase": "2",  
 112: "ladoB": "",  
 113: "usr": "014"  
 114: },  
 115: {  
 116: "trecho": "ENTR ES-245(B) - ENTR ES-440",  
 117: "pwd": "2195",  
 118: "ladoA": "",  
 119: "fase": "2",  
 120: "ladoB": "",  
 121: "usr": "015"  
 122: },  
 123: {  
 124: "trecho": "ENTR ES-375(A) (P/PIÚMA) - ENTR ES-375(B) (ICONHA)",  
 125: "pwd": "2470",  
 126: "ladoA": "",  
 127: "fase": "3",  
 128: "ladoB": "",  
 129: "usr": "016"  
 130: },  
 131: {  
 132: "trecho": "ENTR BR-101(B) - ENTR ES-465 (P/DOMINGOS MARTINS)",  
 133: "pwd": "0070",  
 134: "ladoA": "",  
 135: "fase": "3",  
 136: "ladoB": "",  
 137: "usr": "017"  
 138: },  
 139: {  
 140: "trecho": "KM 15,3 - ENTR BR-484 (P/PONTE SOBRE RIO DOCE)",  
 141: "pwd": "0015",  
 142: "ladoA": "",  
 143: "fase": "2",  
 144: "ladoB": "",  
 145: "usr": "018"  
 146: },  
 147: {  
 148: "trecho": "ENTR BR-030(A)/GO-116 (P/FORMOSA) - ENTR GO-346 (P/CABEÇEIRAS)",  
 149: "pwd": "0110",  
 150: "ladoA": "",  
 151: "fase": "2",  
 152: "ladoB": "",  
 153: "usr": "019"  
 154: },  
 155: {  
 156: "trecho": "ENTR BR-457(B)/GO-219 - ENTR GO-020(A)",  
 157: "pwd": "0090",  
 158: "ladoA": "",  
 159: "fase": "2",  
 160: "ladoB": "",  
 161: "usr": "020",  
 162: "perguntaExtra": true  
 163: },  
 164: {  
 165: "trecho": "ENTR GO-338 (ABADIÂNIA - FIM TRAVESSIA URBANA) - ENTR BR-153(A) (P/ANÁPOLIS)",  
 166: "pwd": "0112",  
 167: "ladoA": "",  
 168: "fase": "4",  
 169: "ladoB": "",  
 170: "usr": "021"  
 171: },  
 172: {  
 173: "trecho": "ENTR GO-353(A) (LINDA VISTA) - ENTR BR-414(A)/GO-151/244/353(B) (PORANGATU)",  
 174: "pwd": "0340",  
 175: "ladoA": "",  
 176: "fase": "3",  
 177: "ladoB": "",  
 178: "usr": "022"  
 179: },  
 180: {  
 181: "trecho": "ENTR BR-154/452(A)/483 - ENTR BR-452(B) (DIV GO/MG) (ITUMBIARA)",  
 182: "pwd": "0770",  
 183: "ladoA": "",  
 184: "fase": "2",  
 185: "ladoB": "",  
 186: "usr": "023"  
 187: },  
 188: {  
 189: "trecho": "ENTR BR-060(B)/158(B)/GO-184(B) (P/ESTREITO) - ENTR GO-050(B)",  
 190: "pwd": "0510",  
 191: "ladoA": "",  
 192: "fase": "2",  
 193: "ladoB": "",  
 194: "usr": "024"  
 195: },  
 196: {  
 197: "trecho": "FIM DA DUPLICAÇÃO - ENTR BR-402/MA-110 (BACABEIRA)",  
 198: "pwd": "0050",  
 199: "ladoA": "",  
 200: "fase": "2",  
 201: "ladoB": "",  
 202: "usr": "025"  
 203: },  
 204: {  
 205: "trecho": "ENTR BR-316(A) (CACHUCHA) - ENTR BR-316(B) (PERITORÓ)",  
 206: "pwd": "0150",  
 207: "ladoA": "",  
 208: "fase": "3",  
 209: "ladoB": "",  
 210: "usr": "026"  
 211: },  
 212: {  
 213: "trecho": "RIO PINDARÉ - ENTR BR-010(A) (AÇAILÂNDIA)",  
 214: "pwd": "0680",  
 215: "ladoA": "",  
 216: "fase": "3",  
 217: "ladoB": "",  
 218: "usr": "027"  
 219: },  
 220: {  
 221: "trecho": "ENTR BR-251(A) (CANACÍ) - ENTR BR-135/251(B)/308/365 (MONTES CLAROS)",  
 222: "pwd": "0670",  
 223: "ladoA": "",  
 224: "fase": "2",  
 225: "ladoB": "",  
 226: "usr": "028"  
 227: },  
 228: {  
 229: "trecho": "ENTR MA-026 (DEZESSETE) - ENTR MA-034(A)/127/349 (CAXIAS)",  
 230: "pwd": "0370",  
 231: "ladoA": "",  
 232: "fase": "3",  
 233: "ladoB": "",  
 234: "usr": "029"  
 235: },  
 236: {  
 237: "trecho": "ENTR MG-410 (P/PONTO DIAMANTE) - ENTR MG-181 (JOÃO PINHEIRO)",  
 238: "pwd": "0130",  
 239: "ladoA": "",  
 240: "fase": "2",  
 241: "ladoB": "",  
 242: "usr": "030"  
 243: },  
 244: {  
 245: "trecho": "ENTR MG-238 (P/SETE LAGOAS) - ENTR MG-432 (P/ESMERALDAS)",  
 246: "pwd": "0330",  
 247: "ladoA": "",  
 248: "fase": "3",  
 249: "ladoB": "",  
 250: "usr": "031"  
 251: },  
 252: {  
 253: "trecho": "ENTR BR-356(A) (P/BELO HORIZONTE) - ENTR BR-356(B)",  
 254: "pwd": "0040",  
 255: "ladoA": "",  
 256: "fase": "2",  
 257: "ladoB": "",  
 258: "usr": "032"  
 259: },  
 260: {  
 261: "trecho": "ENTR BR-499 (SANTOS DUMONT) - ENTR ANT UNIÃO E INDÚSTRIA (B. TRIUNFO)",  
 262: "pwd": "0570",  
 263: "ladoA": "",  
 264: "fase": "3",  
 265: "ladoB": "",  
 266: "usr": "033"  
 267: },  
 268: {  
 269: "trecho": "ENTR BR-365(B)/452(B) - RIO TIJUCO",  
 270: "pwd": "0250",  
 271: "ladoA": "",  
 272: "fase": "4",  
 273: "ladoB": "",  
 274: "usr": "034"  
 275: },  
 276: {  
 277: "trecho": "ENTR BR-342(B)/418/MG-217 (RIB STO ANTÔNIO) (TEÓFILO OTONI) - ACESSO ITAMBACURI",  
 278: "pwd": "1130",  
 279: "ladoA": "",  
 280: "fase": "2",  
 281: "ladoB": "",  
 282: "usr": "035"  
 283: },  
 284: {  
 285: "trecho": "FIM PISTA DUPLA - DOM CORRÊA",  
 286: "pwd": "1280",  
 287: "ladoA": "",  
 288: "fase": "4",  
 289: "ladoB": "",  
 290: "usr": "036"  
 291: },  
 292: {  
 293: "trecho": "ENTR BR-116(A)/120 (LEOPOLDINA) - ENTR BR-116(B)",  
 294: "pwd": "1430",  
 295: "ladoA": "",  
 296: "fase": "3",  
 297: "ladoB": "",  
 298: "usr": "037"  
 299: },  
 300: {  
 301: "trecho": "ENTR BR-259(B) (CURVELO) - ENTR BR-040(A)",  
 302: "pwd": "0850",  
 303: "ladoA": "",  
 304: "fase": "3",  
 305: "ladoB": "",  
 306: "usr": "038"  
 307: },  
 308: {  
 309: "trecho": "ENTR BR-364(A)/262(A) (P/COMENDADOR GOMES) - ENTR BR-364(B)/262(B) (P/FRUTAL)",  
 310: "pwd": "0290",  
 311: "ladoA": "",  
 312: "fase": "2",  
 313: "ladoB": "",  
 314: "usr": "039"  
 315: },  
 316: {  
 317: "trecho": "ENTR MG-187 (P/IBIÁ) - ENTR BR-146 (P/ARAXÁ)",  
 318: "pwd": "0890",  
 319: "ladoA": "",  
 320: "fase": "4",  
 321: "ladoB": "",  
 322: "usr": "040"  
 323: },  
 324: {  
 325: "trecho": "ENTR MG-451 (ITUTINGA) - ENTR BR-354 (LAVRAS)",  
 326: "pwd": "0230",  
 327: "ladoA": "",  
 328: "fase": "3",  
 329: "ladoB": "",  
 330: "usr": "041"  
 331: },  
 332: {  
 333: "trecho": "ACESSO CARMO DO PARANAIBA - ENTR MG-230",  
 334: "pwd": "0215",  
 335: "ladoA": "",  
 336: "fase": "4",  
 337: "ladoB": "",  
 338: "usr": "042"  
 339: },  
 340: {  
 341: "trecho": "PRESIDENTE OLEGÁRIO - ENTR BR-146(A)/365 (PATOS DE MINAS)",  
 342: "pwd": "0100",  
 343: "ladoA": "",  
 344: "fase": "3",  
 345: "ladoB": "",  
 346: "usr": "043"  
 347: },  
 348: {  
 349: "trecho": "ENTR BR-153 - ENTR BR-154(A)",  
 350: "pwd": "0370",  
 351: "ladoA": "",  
 352: "fase": "3",  
 353: "ladoB": "",  
 354: "usr": "044"  
 355: },  
 356: {  
 357: "trecho": "ENTR MG-425 (CORONEL FABRICIANO) - ENTR MG-320 (P/JAGUARAÇU)",  
 358: "pwd": "0230",  
 359: "ladoA": "",  
 360: "fase": "2",  
 361: "ladoB": "",  
 362: "usr": "045"  
 363: },  
 364: {  
 365: "trecho": "ENTR BR-262(B) (BETIM) - ENTR MG-155",  
 366: "pwd": "0490",  
 367: "ladoA": "",  
 368: "fase": "2",  
 369: "ladoB": "",  
 370: "usr": "046"  
 371: },  
 372: {  
 373: "trecho": "ENTR MG-458 (CAREAÇU) - ENTR BR-459 (P/POUSO ALEGRE)",  
 374: "pwd": "0750",  
 375: "ladoA": "",  
 376: "fase": "3",  
 377: "ladoB": "",  
 378: "usr": "047"  
 379: },  
 380: {  
 381: "trecho": "ACESSO COLÂNIA AGRICOLA SEXTA LINHA - FINAL PISTA DUPLA (VILA VARGAS)",  
 382: "pwd": "0324",  
 383: "ladoA": "VILA SÃO PEDRO",  
 384: "fase": "1",  
 385: "ladoB": "ACESSO COLÔNIA AGRÍCOLA SEXTA LINHA",  
 386: "usr": "048"  
 387: },  
 388: {  
 389: "trecho": "ENTR MS-441(B) (P/BANDEIRANTES) - ENTR MS-244 (P/BONFIM)",  
 390: "pwd": "0450",  
 391: "ladoA": "ENTR MS-441(B) (P/BANDEIRANTES)",  
 392: "fase": "1",  
 393: "ladoB": "ENTR MS-244 (P/BONFIM)",  
 394: "usr": "049"  
 395: },  
 396: {  
 397: "trecho": "ENTR BR-158(B)/MS-395 (P/BRASILÂNDIA) - ENTR MS-453/459 (P/ARAPUÁ)",  
 398: "pwd": "1290",  
 399: "ladoA": "",  
 400: "fase": "2",  
 401: "ladoB": "",  
 402: "usr": "050"  
 403: },  
 404: {  
 405: "trecho": "FIM TRAV RIO PARANÁ (PONTE M. JOPPERT) - INCÍO DA PISTA DUPLA",  
 406: "pwd": "0880",  
 407: "ladoA": "",  
 408: "fase": "2",  
 409: "ladoB": "",  
 410: "usr": "051"  
 411: },  
 412: {  
 413: "trecho": "ENTR BR-070(B) - ENTR MT-170 (CARAMÚJO)",  
 414: "pwd": "0030",  
 415: "ladoA": "ENTR BR-070(B)",  
 416: "fase": "1",  
 417: "ladoB": "ENTR MT-170 (CARAMÚJO)",  
 418: "usr": "052"  
 419: },  
 420: {  
 421: "trecho": "ENTR MT-270(B) - ENTR MT-469(A)",  
 422: "pwd": "0592",  
 423: "ladoA": "ENTR MT-270(B)",  
 424: "fase": "1",  
 425: "ladoB": "ENTR MT-469(A)",  
 426: "usr": "053"  
 427: },  
 428: {  
 429: "trecho": "JANGADA - ENTR MT-246(B)",  
 430: "pwd": "0840",  
 431: "ladoA": "",  
 432: "fase": "3",  
 433: "ladoB": "",  
 434: "usr": "054"  
 435: },  
 436: {  
 437: "trecho": "ENTR BR-222(B)/PA-332 (DOM ELISEU) - ENTR PA-125/263 (GURUPIZINHO)",  
 438: "pwd": "0510",  
 439: "ladoA": "",  
 440: "fase": "3",  
 441: "ladoB": "",  
 442: "usr": "055"  
 443: },  
 444: {  
 445: "trecho": "ENTR PB-018 (P/CONDE) - ENTR PB-034",  
 446: "pwd": "0335",  
 447: "ladoA": "",  
 448: "fase": "3",  
 449: "ladoB": "",  
 450: "usr": "056"  
 451: },  
 452: {  
 453: "trecho": "ENTR PB-100 (P/GALANTE) - ENTR BR-104(A)/408(B)/PB-095 (CAMPINA GRANDE)",  
 454: "pwd": "0010",  
 455: "ladoA": "",  
 456: "fase": "4",  
 457: "ladoB": "",  
 458: "usr": "057"  
 459: },  
 460: {  
 461: "trecho": "ENTR PE-063 (P/AMARAJI) - ENTR PE-064/085 (RIBEIRÃO)",  
 462: "pwd": "0510",  
 463: "ladoA": "",  
 464: "fase": "3",  
 465: "ladoB": "",  
 466: "usr": "058"  
 467: },  
 468: {  
 469: "trecho": "ENTR PE-460 - ENTR BR-316/428 (P/CABROBÓ)",  
 470: "pwd": "0460",  
 471: "ladoA": "",  
 472: "fase": "3",  
 473: "ladoB": "",  
 474: "usr": "059"  
 475: },  
 476: {  
 477: "trecho": "ENTR PE-103 (P/BONITO) - ENTR BR-104/423(A) (CARUARÚ)",  
 478: "pwd": "0140",  
 479: "ladoA": "",  
 480: "fase": "3",  
 481: "ladoB": "",  
 482: "usr": "060"  
 483: },  
 484: {  
 485: "trecho": "ENTR PE-450 (P/VERDEJANTE) - ENTR BR-116/361 (SALGUEIRO)",  
 486: "pwd": "0420",  
 487: "ladoA": "",  
 488: "fase": "4",  
 489: "ladoB": "",  
 490: "usr": "061"  
 491: },  
 492: {  
 493: "trecho": "ENTR BR-428(A) (LAGOA GRANDE) - ENTR BR-235/407/423/428(B) (DIV PE/BA) (PETROLINA/JUAZEIRO)",  
 494: "pwd": "0370",  
 495: "ladoA": "",  
 496: "fase": "3",  
 497: "ladoB": "",  
 498: "usr": "062"  
 499: },  
 500: {  
 501: "trecho": "ENTR BR-407(B)/PI-238/245(B) (PICOS) - ENTR PI-242",  
 502: "pwd": "0830",  
 503: "ladoA": "",  
 504: "fase": "4",  
 505: "ladoB": "",  
 506: "usr": "063"  
 507: },  
 508: {  
 509: "trecho": "ENTR BR-404(B)/407(B) - INÍCIO PISTA DUPLA (CAPITÃO CAMPOS)",  
 510: "pwd": "0130",  
 511: "ladoA": "",  
 512: "fase": "3",  
 513: "ladoB": "",  
 514: "usr": "064"  
 515: },  
 516: {  
 517: "trecho": "DEMERVAL LOBÃO - ENTR PI-223 (MONSENHOR GIL)",  
 518: "pwd": "0420",  
 519: "ladoA": "",  
 520: "fase": "3",  
 521: "ladoB": "",  
 522: "usr": "065"  
 523: },  
 524: {  
 525: "trecho": "DIV SP/PR (CAB NORTE PONTE S/ RIO PARDINHO) - INÍCIO VARIANTE DO ALPINO(PISTA DIREITA)",  
 526: "pwd": "2710",  
 527: "ladoA": "DIV SP/PR (CAB NORTE PONTE S/ RIO PARDINHO)",  
 528: "fase": "1",  
 529: "ladoB": "INÍCIO VARIANTE DO ALPINO(PISTA DIREITA)",  
 530: "usr": "066"  
 531: },  
 532: {  
 533: "trecho": "ENTR PR-427 (CAMPO DO TENENTE) - DIV PR/SC (RIO NEGRO/MAFRA)",  
 534: "pwd": "2810",  
 535: "ladoA": "ENTR PR-427 (CAMPO DO TENENTE)",  
 536: "fase": "1",  
 537: "ladoB": "DIV PR/SC (RIO NEGRO/MAFRA)",  
 538: "usr": "067"  
 539: },  
 540: {  
 541: "trecho": "ACESSO GUARAPUAVA - ENTR BR-466 (P/GUARAPUAVA)",  
 542: "pwd": "0205",  
 543: "ladoA": "ACESSO GUARAPUAVA",  
 544: "fase": "1",  
 545: "ladoB": "ENTR BR-466 (P/GUARAPUAVA)",  
 546: "usr": "068"  
 547: },  
 548: {  
 549: "trecho": "ACESSO OESTE CASCAVEL - ENTR BR-163/PR-182 (P/CAPITÃO LEÔNIDAS MARQUES)",  
 550: "pwd": "0310",  
 551: "ladoA": "ENTR PR-182 (P/CAPITÃO LEÔNIDAS MARQUES)",  
 552: "fase": "1",  
 553: "ladoB": "ENTR BR-163",  
 554: "usr": "069"  
 555: },  
 556: {  
 557: "trecho": "FIM PISTA DUPLA - ENTR PR-431 (P/JACAREZINHO)",  
 558: "pwd": "0480",  
 559: "ladoA": "",  
 560: "fase": "2",  
 561: "ladoB": "",  
 562: "usr": "070"  
 563: },  
 564: {  
 565: "trecho": "ENTR PR-281 (P/TIJUCAS DO SUL) - DIV PR/SC (ENTR BR-101)",  
 566: "pwd": "0490",  
 567: "ladoA": "ENTR PR-281 (P/TIJUCAS DO SUL)",  
 568: "fase": "1",  
 569: "ladoB": "DIV PR/SC (ENTR BR-101)",  
 570: "usr": "071"  
 571: },  
 572: {  
 573: "trecho": "ENTR PR-510/511 (P/CONTENDA) - ENTR PR-428 (LAPA)",  
 574: "pwd": "0095",  
 575: "ladoA": "ENTR PR-510/511 (P/CONTENDA)",  
 576: "fase": "1",  
 577: "ladoB": "ENTR PR-428 (LAPA)",  
 578: "usr": "072"  
 579: },  
 580: {  
 581: "trecho": "ENTR BR-393(B) (TREVO MOURA BRASIL) - ENTR BR-492(A) (TREVO P/ AREAL)",  
 582: "pwd": "0730",  
 583: "ladoA": "",  
 584: "fase": "2",  
 585: "ladoB": "",  
 586: "usr": "073"  
 587: },  
 588: {  
 589: "trecho": "ENTR RJ-162 (P/RIO DOURADO) - ENTR BR-120 (CASIMIRO DE ABREU)",  
 590: "pwd": "2930",  
 591: "ladoA": "",  
 592: "fase": "3",  
 593: "ladoB": "",  
 594: "usr": "074"  
 595: },  
 596: {  
 597: "trecho": "ENTR RJ-159 (FLORIANO) - ENTR RJ-161 (RESENDE)",  
 598: "pwd": "2130",  
 599: "ladoA": "",  
 600: "fase": "2",  
 601: "ladoB": "",  
 602: "usr": "075"  
 603: },  
 604: {  
 605: "trecho": "FIM DA PISTA DUPLA (ITAPERUNA) - ENTR BR-393/484(B)/RJ-186",  
 606: "pwd": "0310",  
 607: "ladoA": "",  
 608: "fase": "2",  
 609: "ladoB": "",  
 610: "usr": "076"  
 611: },  
 612: {  
 613: "trecho": "SAPUCAIA - ENTR BR-040(A)",  
 614: "pwd": "0330",  
 615: "ladoA": "",  
 616: "fase": "3",  
 617: "ladoB": "",  
 618: "usr": "077"  
 619: },  
 620: {  
 621: "trecho": "ACESSO À LAGOA DO BONFIM - ENTR RN-315/002(A) (SÃO JOSÉ DE MIPIBÚ/NÍSIA FLORESTA)",  
 622: "pwd": "0167",  
 623: "ladoA": "",  
 624: "fase": "3",  
 625: "ladoB": "",  
 626: "usr": "078"  
 627: },  
 628: {  
 629: "trecho": "ENTR RN-203 (P/SÃO PEDRO) - ACESSO BOM JESUS",  
 630: "pwd": "0070",  
 631: "ladoA": "",  
 632: "fase": "3",  
 633: "ladoB": "",  
 634: "usr": "079"  
 635: },  
 636: {  
 637: "trecho": "ENTR BR-104(B) (LAJES) - ENTR RN-023 (CAIÇARA DO RIO DO VENTO)",  
 638: "pwd": "0270",  
 639: "ladoA": "",  
 640: "fase": "3",  
 641: "ladoB": "",  
 642: "usr": "080"  
 643: },  
 644: {  
 645: "trecho": "ENTR RN-233 (P/CARAÚBAS) - ENTR RN-031/177(A)",  
 646: "pwd": "0070",  
 647: "ladoA": "",  
 648: "fase": "3",  
 649: "ladoB": "",  
 650: "usr": "081"  
 651: },  
 652: {  
 653: "trecho": "ENTR BR-429(B) (JI-PARANÁ) - ENTR RO-473 (P/URUPÁ)",  
 654: "pwd": "1210",  
 655: "ladoA": "",  
 656: "fase": "3",  
 657: "ladoB": "",  
 658: "usr": "082"  
 659: },  
 660: {  
 661: "trecho": "ENTR ERS-703 (GUAIBA) - ENTR ERS-709 (P/ BARRA DO RIBEIRO)",  
 662: "pwd": "3275",  
 663: "ladoA": "",  
 664: "fase": "3",  
 665: "ladoB": "",  
 666: "usr": "083"  
 667: },  
 668: {  
 669: "trecho": "ENTR RSC-472 (P/ PALMITINHO) - ENTR ERS-585/587 (SEBERI)",  
 670: "pwd": "1150",  
 671: "ladoA": "ENTR BR-472 (P/PALMITINHO)",  
 672: "fase": "1",  
 673: "ladoB": "ENTR RS-585/587 (SEBERI)",  
 674: "usr": "084"  
 675: },  
 676: {  
 677: "trecho": "ENTR ERS-348 (VAL DE SERRA) - ENTR ERS-509 (P/ SANTA MARIA)",  
 678: "pwd": "1310",  
 679: "ladoA": "ENTR RS-348 (VAL DE SERRA)",  
 680: "fase": "1",  
 681: "ladoB": "ENTR RS-509 (P/SANTA MARIA)",  
 682: "usr": "085"  
 683: },  
 684: {  
 685: "trecho": "ACESSO A MINAS DO LEAO - ENTR BRS-471 (PANTANO GRANDE)",  
 686: "pwd": "0180",  
 687: "ladoA": "ACESSO A MINAS DO LEÃO",  
 688: "fase": "1",  
 689: "ladoB": "ENTR BR-471 (PANTANO GRANDE)",  
 690: "usr": "086"  
 691: },  
 692: {  
 693: "trecho": "ENTR ERS-705 (GERIBA) - ENTR BRS-153(B) (P/ BAGE)",  
 694: "pwd": "1840",  
 695: "ladoA": "ENTR RS-705 (P/GERIBA)",  
 696: "fase": "1",  
 697: "ladoB": "ENTR BR-290(B)",  
 698: "usr": "087"  
 699: },  
 700: {  
 701: "trecho": "ENTR BRS-293(A) (P/ QUARAI) - ENTR BRS-472(A) (P/ BARRA DO QUARAI)",  
 702: "pwd": "0410",  
 703: "ladoA": "ENTR BR-293(A)",  
 704: "fase": "1",  
 705: "ladoB": "ENTR BR-472(A)",  
 706: "usr": "088"  
 707: },  
 708: {  
 709: "trecho": "ENTR RSC-473 (P/ BAGE) - ENTR ERS-630/634 (P/ DOM PEDRITO)",  
 710: "pwd": "0130",  
 711: "ladoA": "ENTR BR-473 (P/BAGÉ)",  
 712: "fase": "1",  
 713: "ladoB": "ENTR RS-630 (P/DOM PEDRITO)",  
 714: "usr": "089"  
 715: },  
 716: {  
 717: "trecho": "ENTR BRS-471(A) (QUINTA) - ACESSO SUL A PELOTAS",  
 718: "pwd": "0070",  
 719: "ladoA": "ENTR BR-471(A) (QUINTA)",  
 720: "fase": "1",  
 721: "ladoB": "ACESSO PELOTAS",  
 722: "usr": "090"  
 723: },  
 724: {  
 725: "trecho": "ACESSO LESTE A ITAQUI - ENTR BR-290(A)/293(A) (URUGUAIANA)",  
 726: "pwd": "0210",  
 727: "ladoA": "ACESSO LESTE A ITAQUI",  
 728: "fase": "1",  
 729: "ladoB": "ENTR BR-290(A)/293(A) (URUGUAIANA)",  
 730: "usr": "091"  
 731: },  
 732: {  
 733: "trecho": "GARUVA - ENTR SC-430 (PIRABEIRABA)",  
 734: "pwd": "3830",  
 735: "ladoA": "GARUVA",  
 736: "fase": "1",  
 737: "ladoB": "ENTR SC-430 (PIRABEIRABA)",  
 738: "usr": "092"  
 739: },  
 740: {  
 741: "trecho": "BALNEÁRIO DE CAMBORIÚ - P/PORTO BELO",  
 742: "pwd": "4010",  
 743: "ladoA": "BALNEÁRIO DE CAMBORIÚ",  
 744: "fase": "1",  
 745: "ladoB": "P/PORTO BELO",  
 746: "usr": "093"  
 747: },  
 748: {  
 749: "trecho": "PONTE SOBRE O RIO DA MADRE - PONTE SOBRE O RIO ARAÇATUBA",  
 750: "pwd": "4125",  
 751: "ladoA": "PONTE SOBRE O RIO DA MADRE",  
 752: "fase": "1",  
 753: "ladoB": "PONTE SOBRE O RIO ARAÇATUBA",  
 754: "usr": "094"  
 755: },  
 756: {  
 757: "trecho": "ENTR BR-285(A)/SC-449 (ARARANGUÁ) - ENTR BR-285(B)/SC-285 (P/ERMO)",  
 758: "pwd": "4240",  
 759: "ladoA": "ENTR BR-285(A) (ARARANGUÁ)",  
 760: "fase": "1",  
 761: "ladoB": "ENTR BR-285(B)/SC-285 (SANGA DA TOCA)",  
 762: "usr": "095"  
 763: },  
 764: {  
 765: "trecho": "ENTR SC-352(B) (P/TAIÓ) - SÃO CRISTOVÃO DO SUL",  
 766: "pwd": "2910",  
 767: "ladoA": "",  
 768: "fase": "2",  
 769: "ladoB": "",  
 770: "usr": "096"  
 771: },  
 772: {  
 773: "trecho": "ENTR SC-463 (P/JABORÁ) - ENTR BR-283 (P/CONCÓRDIA)",  
 774: "pwd": "1590",  
 775: "ladoA": "",  
 776: "fase": "3",  
 777: "ladoB": "",  
 778: "usr": "097"  
 779: },  
 780: {  
 781: "trecho": "ENTR SC-114(A) (ÍNDIOS) - ENTR SC-114(B) (P/PAINEL)",  
 782: "pwd": "0020",  
 783: "ladoA": "",  
 784: "fase": "3",  
 785: "ladoB": "",  
 786: "usr": "098"  
 787: },  
 788: {  
 789: "trecho": "ENTR SC-157(B) (P/MODELO) - ENTR BR-158(A) (P/CUNHA PORÃ)",  
 790: "pwd": "0377",  
 791: "ladoA": "ENTR SC-157(B) (P/MODELO)",  
 792: "fase": "1",  
 793: "ladoB": "ENTR BR-158(A) (P/CUNHA PORÃ)",  
 794: "usr": "099"  
 795: },  
 796: {  
 797: "trecho": "ENTR SC-114(B) (P/OTACÍLIO COSTA) - ENTR BR-116",  
 798: "pwd": "0190",  
 799: "ladoA": "",  
 800: "fase": "4",  
 801: "ladoB": "",  
 802: "usr": "100"  
 803: },  
 804: {  
 805: "trecho": "ENTR SE-464 (P/SÃO CRISTOVÃO) - ENTR BR-349(B)/SE-265 (ITAPORANGA D`AJUDA)",  
 806: "pwd": "0040",  
 807: "ladoA": "",  
 808: "fase": "3",  
 809: "ladoB": "",  
 810: "usr": "101"  
 811: },  
 812: {  
 813: "trecho": "ENTR SE-160 - ENTR SE-170(A)",  
 814: "pwd": "0060",  
 815: "ladoA": "",  
 816: "fase": "3",  
 817: "ladoB": "",  
 818: "usr": "102"  
 819: },  
 820: {  
 821: "trecho": "ENTR SP-070 - ENTR SP-103 (CAÇAPAVA)",  
 822: "pwd": "2410",  
 823: "ladoA": "",  
 824: "fase": "3",  
 825: "ladoB": "",  
 826: "usr": "103"  
 827: },  
 828: {  
 829: "trecho": "ENTR SP-228 - ENTR SP-057 (P/SIDERÚRGICA)",  
 830: "pwd": "2570",  
 831: "ladoA": "",  
 832: "fase": "3",  
 833: "ladoB": "",  
 834: "usr": "104"  
 835: },  
 836: {  
 837: "trecho": "ACESSO GUAIÇARA - ENTR BR-154/267 (LINS)",  
 838: "pwd": "1043",  
 839: "ladoA": "",  
 840: "fase": "3",  
 841: "ladoB": "",  
 842: "usr": "105"  
 843: },  
 844: {  
 845: "trecho": "DIV MG/SP - ENTR SP-063 (P/BRAGANÇA PAULISTA)",  
 846: "pwd": "0810",  
 847: "ladoA": "",  
 848: "fase": "3",  
 849: "ladoB": "",  
 850: "usr": "106"  
 851: },  
 852: {  
 853: "trecho": "ENTR TO-420 - INICIO PISTA DUPLA ARAGUAINA",  
 854: "pwd": "0105",  
 855: "ladoA": "",  
 856: "fase": "2",  
 857: "ladoB": "",  
 858: "usr": "107"  
 859: },  
 860: {  
 861: "trecho": "ENTR TO-342(B) (MIRANORTE) - ENTR TO-348 (BARROLÂNDIA)",  
 862: "pwd": "0205",  
 863: "ladoA": "",  
 864: "fase": "3",  
 865: "ladoB": "",  
 866: "usr": "108"  
 867: },  
 868: {  
 869: "trecho": "ENTR TO-483 (FIGUEIRÓPOLIS) - ENTR TO-296(A)/373 (ALVORADA)",  
 870: "pwd": "0300",  
 871: "ladoA": "",  
 872: "fase": "3",  
 873: "ladoB": "",  
 874: "usr": "109"  
 875: },  
 876: {  
 877: "trecho": "ENTR PI-260 (P/BARREIRAS DO PIAUÍ) - ENTR PI-255 (CORRENTE)",  
 878: "pwd": "0495",  
 879: "ladoA": "",  
 880: "fase": "3",  
 881: "ladoB": "",  
 882: "usr": "110"  
 883: },  
 884: {  
 885: "trecho": "FIM PISTA DUPLA - ENTR GO-156(B)",  
 886: "pwd": "0150",  
 887: "ladoA": "",  
 888: "fase": "3",  
 889: "ladoB": "",  
 890: "usr": "111"  
 891: },  
 892: {  
 893: "trecho": "ENTR PR-508 (P/MATINHOS) - ENTR BR-101/PR-408 (P/MORRETES)",  
 894: "pwd": "0025",  
 895: "ladoA": "ENTR PR-508 (P/MATINHOS)",  
 896: "fase": "1",  
 897: "ladoB": "ENTR BR-101/PR-408 (P/MORRETES)",  
 898: "usr": "112"  
 899: },  
 900: {  
 901: "trecho": "ACESSO SANTA - ENTR BR-376(B)/PR-428 (SÃO LUIS PURUNÃ)",  
 902: "pwd": "0075",  
 903: "ladoA": "ACESSO SANTA",  
 904: "fase": "1",  
 905: "ladoB": "ENTR BR-376(B)/PR-428 (SÃO LUIS PURUNÃ)",  
 906: "usr": "113"  
 907: },  
 908: {  
 909: "trecho": "ENTR MS-375 (ZUZU) - ENTR BR-163(A) (NOVA ALVORADA)",  
 910: "pwd": "0950",  
 911: "ladoA": "ENTR MS-375 (ZUZU)",  
 912: "fase": "1",  
 913: "ladoB": "ENTR BR-163(A) (NOVA ALVORADA DO SUL)",  
 914: "usr": "114"  
 915: },  
 916: {  
 917: "trecho": "ENTR MS-434 (RAIMUNDO) - ENTR BR-483/497 (INÍCIO DA PISTA DUPLA)",  
 918: "pwd": "0440",  
 919: "ladoA": "ENTR MS-434 (RAIMUNDO)",  
 920: "fase": "1",  
 921: "ladoB": "ENTR BR-483/497 (INÍCIO DA PISTA DUPLA)",  
 922: "usr": "115"  
 923: },  
 924: {  
 925: "trecho": "ENTR SP-148 - ENTR SP-160 (ENTR PISTA INVERSA (A))",  
 926: "pwd": "0810",  
 927: "ladoA": "",  
 928: "fase": "3",  
 929: "ladoB": "",  
 930: "usr": "116"  
 931: },  
 932: {  
 933: "trecho": "ENTR BR-493(A) (P/MAGÉ) - ENTR RJ-107 (IMBARIÊ)",  
 934: "pwd": "0070",  
 935: "ladoA": "",  
 936: "fase": "3",  
 937: "ladoB": "",  
 938: "usr": "117"  
 939: },  
 940: {  
 941: "trecho": "ENTR BR-101 - ENTR SC-108(A) (NEUDOR)",  
 942: "pwd": "0040",  
 943: "ladoA": "ENTR BR-101",  
 944: "fase": "1",  
 945: "ladoB": "ENTR SC-108(A) (NEUDOR)",  
 946: "usr": "118"  
 947: },  
 948: {  
 949: "trecho": "ENTR BR-153(A)/PR-446 - ENTR PR-170",  
 950: "pwd": "1490",  
 951: "ladoA": "ENTR BR-280(A)/PR-446",  
 952: "fase": "1",  
 953: "ladoB": "ENTR PR-170",  
 954: "usr": "119"  
 955: },  
 956: {  
 957: "trecho": "ENTR BR-262(B) - ENTR BR-050(B)",  
 958: "pwd": "0060",  
 959: "ladoA": "",  
 960: "fase": "2",  
 961: "ladoB": "",  
 962: "usr": "120"  
 963: },  
 964: {  
 965: "trecho": "ENTR MT-423 - ITAUBA",  
 966: "pwd": "0834",  
 967: "ladoA": "ENTR MT-423",  
 968: "fase": "1",  
 969: "ladoB": "ITAUBA",  
 970: "usr": "121"  
 971: },  
 972: {  
 973: "trecho": "MATUPÁ - GUARANTÃ DO NORTE",  
 974: "pwd": "0852",  
 975: "ladoA": "",  
 976: "fase": "3",  
 977: "ladoB": "",  
 978: "usr": "122"  
 979: },  
 980: {  
 981: "trecho": "ENTR AL-215(B) - ENTR AL-220(A) (P/BARRA DE SÃO MIGUEL)",  
 982: "pwd": "0790",  
 983: "ladoA": "",  
 984: "fase": "3",  
 985: "ladoB": "",  
 986: "usr": "123"  
 987: },  
 988: {  
 989: "trecho": "ENTR PE-270 (P/BUÍQUE) - ENTR BR-110 (CRUZEIRO DO NORDESTE)",  
 990: "pwd": "0260",  
 991: "ladoA": "",  
 992: "fase": "4",  
 993: "ladoB": "",  
 994: "usr": "124"  
 995: },  
 996: {  
 997: "trecho": "ENTR PE-218 (P/BOM CONSELHO) - ENTR PE-187 (P/PALMEIRINA)",  
 998: "pwd": "0090",  
 999: "ladoA": "",  
1000: "fase": "4",  
1001: "ladoB": "",  
1002: "usr": "125"  
1003: },  
1004: {  
1005: "trecho": "ENTR BR-420(B) (P/LAJE) - ENTR BA-542 (P/GUERÉM)",  
1006: "pwd": "1695",  
1007: "ladoA": "",  
1008: "fase": "3",  
1009: "ladoB": "",  
1010: "usr": "126"  
1011: },  
1012: {  
1013: "trecho": "ENTR BA-493 (ITATIM) - ENTR BA-245",  
1014: "pwd": "0790",  
1015: "ladoA": "",  
1016: "fase": "4",  
1017: "ladoB": "",  
1018: "usr": "127"  
1019: },  
1020: {  
1021: "trecho": "MAIRI - ENTR BR-349",  
1022: "pwd": "0334",  
1023: "ladoA": "",  
1024: "fase": "2",  
1025: "ladoB": "",  
1026: "usr": "128"  
1027: },  
1028: {  
1029: "trecho": "GAVIÃO - ENTR BR-349 (NOVA FÁTIMA)",  
1030: "pwd": "0270",  
1031: "ladoA": "",  
1032: "fase": "2",  
1033: "ladoB": "",  
1034: "usr": "129"  
1035: },  
1036: {  
1037: "trecho": "DIV BA/MG - ENTR BR-251(A) (P/SALINAS)",  
1038: "pwd": "1010",  
1039: "ladoA": "",  
1040: "fase": "2",  
1041: "ladoB": "",  
1042: "usr": "130"  
1043: },  
1044: {  
1045: "trecho": "ENTR BA-937 (P/PAJEÚ DO VENTO) - ENTR BR-122(B)/430/BA-569 (CAETITÉ)",  
1046: "pwd": "0530",  
1047: "ladoA": "",  
1048: "fase": "2",  
1049: "ladoB": "",  
1050: "usr": "131"  
1051: },  
1052: {  
1053: "trecho": "ENTR BA-131 (P/CAÉM) - ENTR BA-417",  
1054: "pwd": "0245",  
1055: "ladoA": "",  
1056: "fase": "3",  
1057: "ladoB": "",  
1058: "usr": "132"  
1059: },  
1060: {  
1061: "trecho": "ENTR BR-235 - ENTR BA-220 (EUCLIDES DA CUNHA)",  
1062: "pwd": "0550",  
1063: "ladoA": "",  
1064: "fase": "2",  
1065: "ladoB": "",  
1066: "usr": "133"  
1067: },  
1068: {  
1069: "trecho": "ENTR CE-373 (CARMELÓPOLIS) - ENTR CE-187/292 (CAMPOS SALES)",  
1070: "pwd": "0690",  
1071: "ladoA": "",  
1072: "fase": "3",  
1073: "ladoB": "",  
1074: "usr": "134"  
1075: },  
1076: {  
1077: "trecho": "ENTR CE-266 (SUCESSO) - ENTR BR-226/404 (CRATEÚS)",  
1078: "pwd": "0190",  
1079: "ladoA": "",  
1080: "fase": "2",  
1081: "ladoB": "",  
1082: "usr": "135"  
1083: },  
1084: {  
1085: "trecho": "ENTR CE-060(B) (MINEIROLÂNDIA) - ENTR CE-168 (PEDRA BRANCA)",  
1086: "pwd": "0610",  
1087: "ladoA": "",  
1088: "fase": "4",  
1089: "ladoB": "",  
1090: "usr": "136"  
1091: },  
1092: {  
1093: "trecho": "ENTR CE-154 (PIRANGI) - ENTR CE-060(A)/265 (QUIXADÁ)",  
1094: "pwd": "0050",  
1095: "ladoA": "",  
1096: "fase": "4",  
1097: "ladoB": "",  
1098: "usr": "137"  
1099: },  
1100: {  
1101: "trecho": "ENTR CE-456 (P/TARGINOS) - ENTR CE-257 (CANINDÉ)",  
1102: "pwd": "0590",  
1103: "ladoA": "",  
1104: "fase": "2",  
1105: "ladoB": "",  
1106: "usr": "138"  
1107: },  
1108: {  
1109: "trecho": "ENTR CE-341 (CROATÁ) - ENTR CE-162 (SÃO LUÍS DO CURU)",  
1110: "pwd": "0070",  
1111: "ladoA": "",  
1112: "fase": "4",  
1113: "ladoB": "",  
1114: "usr": "139"  
1115: },  
1116: {  
1117: "trecho": "ENTR CE-263 (P/JAGUARUANA) - RUSSAS",  
1118: "pwd": "0160",  
1119: "ladoA": "",  
1120: "fase": "3",  
1121: "ladoB": "",  
1122: "usr": "140"  
1123: },  
1124: {  
1125: "trecho": "ENTR BR-381 (SÃO MATEUS) - ENTR ES-430 (P/JAGUARÉ)",  
1126: "pwd": "2130",  
1127: "ladoA": "",  
1128: "fase": "2",  
1129: "ladoB": "",  
1130: "usr": "141"  
1131: },  
1132: {  
1133: "trecho": "ENTR ES-137 - ENTR ES-130 (P/NOVA VENÉCIA)",  
1134: "pwd": "0300",  
1135: "ladoA": "",  
1136: "fase": "4",  
1137: "ladoB": "",  
1138: "usr": "142"  
1139: },  
1140: {  
1141: "trecho": "ENTR BA-290 (TEIXEIRA DE FREITAS) - ENTR BR-418 (P/POSTO DA MATA)",  
1142: "pwd": "1990",  
1143: "ladoA": "",  
1144: "fase": "3",  
1145: "ladoB": "",  
1146: "usr": "143"  
1147: },  
1148: {  
1149: "trecho": "MORRO GRANDE (ACESSO IV C. ITAPEMIRIM) - ENTR ES-166 (COUTINHO)",  
1150: "pwd": "0030",  
1151: "ladoA": "",  
1152: "fase": "2",  
1153: "ladoB": "",  
1154: "usr": "144"  
1155: },  
1156: {  
1157: "trecho": "ENTR MG-422 (RESPLENDOR) - ENTR BR-458 (CONSELHEIRO PENA)",  
1158: "pwd": "0130",  
1159: "ladoA": "",  
1160: "fase": "2",  
1161: "ladoB": "",  
1162: "usr": "145"  
1163: },  
1164: {  
1165: "trecho": "ENTR GO-230 - ENTR GO-237",  
1166: "pwd": "0125",  
1167: "ladoA": "",  
1168: "fase": "3",  
1169: "ladoB": "",  
1170: "usr": "146"  
1171: },  
1172: {  
1173: "trecho": "ENTR GO-114 (P/FLORES DE GOIÁS) - SANTA MARIA",  
1174: "pwd": "0160",  
1175: "ladoA": "",  
1176: "fase": "2",  
1177: "ladoB": "",  
1178: "usr": "147"  
1179: },  
1180: {  
1181: "trecho": "DIV MG/GO - ENTR GO-010(A)",  
1182: "pwd": "0470",  
1183: "ladoA": "",  
1184: "fase": "3",  
1185: "ladoB": "",  
1186: "usr": "148"  
1187: },  
1188: {  
1189: "trecho": "ENTR GO-010 (P/LUZIÂNIA) - ENTR BR-050(B)/354/457/GO-309 (CRISTALINA)",  
1190: "pwd": "0050",  
1191: "ladoA": "",  
1192: "fase": "4",  
1193: "ladoB": "",  
1194: "usr": "149"  
1195: },  
1196: {  
1197: "trecho": "URUTAÍ - ENTR BR-490/GO-213/307 (IPAMERI)",  
1198: "pwd": "0080",  
1199: "ladoA": "",  
1200: "fase": "3",  
1201: "ladoB": "",  
1202: "usr": "150"  
1203: },  
1204: {  
1205: "trecho": "ENTR GO-080(B) (P/SAO FRANCISCO DE GOIÁS) - ENTR GO-431 (P/PIRENOPOLIS)",  
1206: "pwd": "0550",  
1207: "ladoA": "",  
1208: "fase": "2",  
1209: "ladoB": "",  
1210: "usr": "151"  
1211: },  
1212: {  
1213: "trecho": "ENTR BR-153(A)/GO-342(B) - ENTR BR-153(B)/GO-237 (URUAÇÚ)",  
1214: "pwd": "0190",  
1215: "ladoA": "",  
1216: "fase": "2",  
1217: "ladoB": "",  
1218: "usr": "152"  
1219: },  
1220: {  
1221: "trecho": "ENTR GO-210(B) (P/PANAMÁ) - ENTR BR-154/452(A)/483",  
1222: "pwd": "0750",  
1223: "ladoA": "",  
1224: "fase": "2",  
1225: "ladoB": "",  
1226: "usr": "153"  
1227: },  
1228: {  
1229: "trecho": "ENTR GO-040 - ENTR BR-154(A)/483(A)/GO-206(A) (P/CACHOEIRA DOURADA)",  
1230: "pwd": "0050",  
1231: "ladoA": "",  
1232: "fase": "2",  
1233: "ladoB": "",  
1234: "usr": "154"  
1235: },  
1236: {  
1237: "trecho": "ENTR GO-194 (P/PORTELÂNDIA) - INÍCIO PISTA DUPLA",  
1238: "pwd": "0570",  
1239: "ladoA": "",  
1240: "fase": "3",  
1241: "ladoB": "",  
1242: "usr": "155"  
1243: },  
1244: {  
1245: "trecho": "ENTR GO-174(B) - ENTR GO-206 (P/CAÇU)",  
1246: "pwd": "0440",  
1247: "ladoA": "",  
1248: "fase": "3",  
1249: "ladoB": "",  
1250: "usr": "156"  
1251: },  
1252: {  
1253: "trecho": "ENTR MA-006(A) (COCALINHO) - ENTR MA-216 (BOM JARDIM)",  
1254: "pwd": "0280",  
1255: "ladoA": "",  
1256: "fase": "2",  
1257: "ladoB": "",  
1258: "usr": "157"  
1259: },  
1260: {  
1261: "trecho": "ENTR MA-259 (P/TUNTUM) - ENTR MA-012/272 (BARRA DO CORDA)",  
1262: "pwd": "0860",  
1263: "ladoA": "",  
1264: "fase": "3",  
1265: "ladoB": "",  
1266: "usr": "158"  
1267: },  
1268: {  
1269: "trecho": "ENTR BR-222(B) (ESTACA ZERO) - ENTR MA-240 (PIO XII)",  
1270: "pwd": "0300",  
1271: "ladoA": "",  
1272: "fase": "2",  
1273: "ladoB": "",  
1274: "usr": "159"  
1275: },  
1276: {  
1277: "trecho": "BOM JESUS DO TOCANTINS - RIO JACUNDÁ (R MÃE MARIA)",  
1278: "pwd": "0730",  
1279: "ladoA": "",  
1280: "fase": "3",  
1281: "ladoB": "",  
1282: "usr": "160"  
1283: },  
1284: {  
1285: "trecho": "ENTR BR-122/251/365 (MONTES CLAROS) - ENTR BR-451 (BOCAIÚVA)",  
1286: "pwd": "0750",  
1287: "ladoA": "",  
1288: "fase": "3",  
1289: "ladoB": "",  
1290: "usr": "161"  
1291: },  
1292: {  
1293: "trecho": "ENTR BR-342/MG-404 (SALINAS) - RIO VACARIA",  
1294: "pwd": "0240",  
1295: "ladoA": "",  
1296: "fase": "4",  
1297: "ladoB": "",  
1298: "usr": "162"  
1299: },  
1300: {  
1301: "trecho": "ENTR MA-132/270 (COLINAS) - ENTR MA-134 (PEIXE)",  
1302: "pwd": "0290",  
1303: "ladoA": "",  
1304: "fase": "2",  
1305: "ladoB": "",  
1306: "usr": "163"  
1307: },  
1308: {  
1309: "trecho": "ENTR BR-040(B) - ENTR MG-155 (SÃO BRÁS DO SUAÇUÍ)",  
1310: "pwd": "0030",  
1311: "ladoA": "",  
1312: "fase": "3",  
1313: "ladoB": "",  
1314: "usr": "164"  
1315: },  
1316: {  
1317: "trecho": "ENTR MG-434 - ENTR MG-435 (P/CAETÉ)",  
1318: "pwd": "0335",  
1319: "ladoA": "",  
1320: "fase": "2",  
1321: "ladoB": "",  
1322: "usr": "165"  
1323: },  
1324: {  
1325: "trecho": "ENTR MG-275 (P/CARANDAÍ) - ACESSO ALTO DOCE (INÍCIO PISTA DUPLA)",  
1326: "pwd": "0510",  
1327: "ladoA": "",  
1328: "fase": "2",  
1329: "ladoB": "",  
1330: "usr": "166"  
1331: },  
1332: {  
1333: "trecho": "ENTR BR-040(B) - ENTR MG-135",  
1334: "pwd": "0130",  
1335: "ladoA": "",  
1336: "fase": "3",  
1337: "ladoB": "",  
1338: "usr": "167"  
1339: },  
1340: {  
1341: "trecho": "ENTR MG-190 (RIO BAGAGEM) - ACESSO INDIANÓPOLIS",  
1342: "pwd": "0250",  
1343: "ladoA": "",  
1344: "fase": "3",  
1345: "ladoB": "",  
1346: "usr": "168"  
1347: },  
1348: {  
1349: "trecho": "ENTR BR-352(B)/GO-210(B)/330 (CATALÃO) - ENTR GO-402",  
1350: "pwd": "0152",  
1351: "ladoA": "",  
1352: "fase": "3",  
1353: "ladoB": "",  
1354: "usr": "169"  
1355: },  
1356: {  
1357: "trecho": "ENTR BR-050/365/452/455 (UBERLÂNDIA) - ENTR BR-153/464 (PRATA)",  
1358: "pwd": "0010",  
1359: "ladoA": "",  
1360: "fase": "3",  
1361: "ladoB": "",  
1362: "usr": "170"  
1363: },  
1364: {  
1365: "trecho": "ENTR BR-050(B)/455/497 (UBERLÂNDIA) - ENTR BR-452(B)",  
1366: "pwd": "0310",  
1367: "ladoA": "",  
1368: "fase": "4",  
1369: "ladoB": "",  
1370: "usr": "171"  
1371: },  
1372: {  
1373: "trecho": "ENTR MG-190 (P/NOVA PONTE) - ACESSO SANTA JULIANA",  
1374: "pwd": "0230",  
1375: "ladoA": "",  
1376: "fase": "4",  
1377: "ladoB": "",  
1378: "usr": "172"  
1379: },  
1380: {  
1381: "trecho": "ACESSO CARLOS CHAGAS - ENTR MG-412",  
1382: "pwd": "0110",  
1383: "ladoA": "",  
1384: "fase": "3",  
1385: "ladoB": "",  
1386: "usr": "173"  
1387: },  
1388: {  
1389: "trecho": "ENTR BR-451(B)/MG-416 (P/SÃO PEDRO DO SUAÇUÍ) - ENTR MG-117 (SÃO JOÃO EVANGELISTA)",  
1390: "pwd": "0110",  
1391: "ladoA": "",  
1392: "fase": "2",  
1393: "ladoB": "",  
1394: "usr": "174"  
1395: },  
1396: {  
1397: "trecho": "TREVO DE ACESSO SÃO JOÃO DO MANHUAÇU - ENTR MG-265 (P/DIVINO)",  
1398: "pwd": "1310",  
1399: "ladoA": "",  
1400: "fase": "4",  
1401: "ladoB": "",  
1402: "usr": "175"  
1403: },  
1404: {  
1405: "trecho": "ENTR ES-185(B) (P/IÚNA) - DIV ES/MG",  
1406: "pwd": "0210",  
1407: "ladoA": "",  
1408: "fase": "2",  
1409: "ladoB": "",  
1410: "usr": "176"  
1411: },  
1412: {  
1413: "trecho": "ACESSO ITANHOMI - ENTR BR-458(A) (TURUAÇÚ)",  
1414: "pwd": "1195",  
1415: "ladoA": "",  
1416: "fase": "3",  
1417: "ladoB": "",  
1418: "usr": "177"  
1419: },  
1420: {  
1421: "trecho": "ENTR BR-116(B) - ENTR MG-126 (BICAS)",  
1422: "pwd": "0030",  
1423: "ladoA": "",  
1424: "fase": "2",  
1425: "ladoB": "",  
1426: "usr": "178"  
1427: },  
1428: {  
1429: "trecho": "ENTR BR-265/356 (MURIAÉ) - ENTR MG-285 (LARANJAL)",  
1430: "pwd": "1370",  
1431: "ladoA": "",  
1432: "fase": "2",  
1433: "ladoB": "",  
1434: "usr": "179"  
1435: },  
1436: {  
1437: "trecho": "PONTE S/ RIO PARÁ - ENTR MG-423(A) (PITANGUI)",  
1438: "pwd": "0310",  
1439: "ladoA": "",  
1440: "fase": "3",  
1441: "ladoB": "",  
1442: "usr": "180"  
1443: },  
1444: {  
1445: "trecho": "ACESSO MORADA NOVA DE MINAS - ACESSO SÃO JOSÉ DO BURITI",  
1446: "pwd": "0195",  
1447: "ladoA": "",  
1448: "fase": "2",  
1449: "ladoB": "",  
1450: "usr": "181"  
1451: },  
1452: {  
1453: "trecho": "ENTR BR-367(B) (GOUVEIA) - ACESSO NOSSA SENHORA DA GLÓRIA",  
1454: "pwd": "0330",  
1455: "ladoA": "",  
1456: "fase": "2",  
1457: "ladoB": "",  
1458: "usr": "182"  
1459: },  
1460: {  
1461: "trecho": "ENTR MG-255 - ENTR BR-262(B) (DIV MG/SP)",  
1462: "pwd": "0910",  
1463: "ladoA": "",  
1464: "fase": "2",  
1465: "ladoB": "",  
1466: "usr": "183"  
1467: },  
1468: {  
1469: "trecho": "ENTR BR-462(B) - ENTR BR-262",  
1470: "pwd": "0250",  
1471: "ladoA": "",  
1472: "fase": "3",  
1473: "ladoB": "",  
1474: "usr": "184",  
1475: "perguntaExtra": true  
1476: },  
1477: {  
1478: "trecho": "ENTR BR-452 (P/UBERLÂNDIA) - ENTR BR-462 (PERDIZES)",  
1479: "pwd": "0930",  
1480: "ladoA": "",  
1481: "fase": "4",  
1482: "ladoB": "",  
1483: "usr": "185",  
1484: "perguntaExtra": true  
1485: },  
1486: {  
1487: "trecho": "ENTR BR-265(B) - CAMPOS GERAIS",  
1488: "pwd": "0070",  
1489: "ladoA": "",  
1490: "fase": "2",  
1491: "ladoB": "",  
1492: "usr": "186"  
1493: },  
1494: {  
1495: "trecho": "ENTR MG-354(B) - ENTR BR-265(A)",  
1496: "pwd": "0030",  
1497: "ladoA": "",  
1498: "fase": "2",  
1499: "ladoB": "",  
1500: "usr": "187"  
1501: },  
1502: {  
1503: "trecho": "ENTR MG-135/338 (BARBACENA) - ACESSO TIRADENTES",  
1504: "pwd": "0190",  
1505: "ladoA": "",  
1506: "fase": "2",  
1507: "ladoB": "",  
1508: "usr": "188"  
1509: },  
1510: {  
1511: "trecho": "ENTR BR-265 (LAVRAS) - ENTR MG-167(A) (P/TRÊS CORAÇÕES)",  
1512: "pwd": "0670",  
1513: "ladoA": "",  
1514: "fase": "4",  
1515: "ladoB": "",  
1516: "usr": "189"  
1517: },  
1518: {  
1519: "trecho": "ACESSO FORMIGA - ENTR MG-164 (CANDEIAS)",  
1520: "pwd": "0350",  
1521: "ladoA": "",  
1522: "fase": "2",  
1523: "ladoB": "",  
1524: "usr": "190"  
1525: },  
1526: {  
1527: "trecho": "ENTR MG-270 (CARMÓPOLIS DE MINAS) - ENTR BR-494(A) (P/OLIVEIRA)",  
1528: "pwd": "0590",  
1529: "ladoA": "",  
1530: "fase": "2",  
1531: "ladoB": "",  
1532: "usr": "191"  
1533: },  
1534: {  
1535: "trecho": "ENTR BR-354(A) (P/BAMBUÍ) - ENTR BR-354(B)",  
1536: "pwd": "0230",  
1537: "ladoA": "",  
1538: "fase": "3",  
1539: "ladoB": "",  
1540: "usr": "192"  
1541: },  
1542: {  
1543: "trecho": "ACESSO GUIMARÂNIA - ENTR MG-188",  
1544: "pwd": "0205",  
1545: "ladoA": "",  
1546: "fase": "2",  
1547: "ladoB": "",  
1548: "usr": "193"  
1549: },  
1550: {  
1551: "trecho": "ENTR MG-410 - PRESIDENTE OLEGÁRIO",  
1552: "pwd": "0090",  
1553: "ladoA": "",  
1554: "fase": "3",  
1555: "ladoB": "",  
1556: "usr": "194"  
1557: },  
1558: {  
1559: "trecho": "ENTR MG-060 (P/SÃO GONÇALO DO ABAETÉ) - ACESSO GALENA",  
1560: "pwd": "0150",  
1561: "ladoA": "",  
1562: "fase": "4",  
1563: "ladoB": "",  
1564: "usr": "195"  
1565: },  
1566: {  
1567: "trecho": "ENTR BR-154(B)/461/464 (ITUIUTABA) - ACESSO GURINHATÃ",  
1568: "pwd": "0410",  
1569: "ladoA": "",  
1570: "fase": "2",  
1571: "ladoB": "",  
1572: "usr": "196"  
1573: },  
1574: {  
1575: "trecho": "ENTR BR-365 (P/MONTE ALEGRE DE MINAS) - ENTR BR-464/497 (P/PRATA)",  
1576: "pwd": "0830",  
1577: "ladoA": "",  
1578: "fase": "2",  
1579: "ladoB": "",  
1580: "usr": "197"  
1581: },  
1582: {  
1583: "trecho": "ENTR MG-229 (P/SENHORA DO PORTO) - ENTR MG-232 (P/CARMESIA)",  
1584: "pwd": "0170",  
1585: "ladoA": "",  
1586: "fase": "3",  
1587: "ladoB": "",  
1588: "usr": "198"  
1589: },  
1590: {  
1591: "trecho": "ENTR MG-320 (P/SÃO JOSÉ DO GOIABAL) - ENTR BR-120 (VARGEM LINDA)",  
1592: "pwd": "0370",  
1593: "ladoA": "",  
1594: "fase": "2",  
1595: "ladoB": "",  
1596: "usr": "199"  
1597: },  
1598: {  
1599: "trecho": "ACESSO À GOV. VALADARES - PERIQUITO",  
1600: "pwd": "0170",  
1601: "ladoA": "",  
1602: "fase": "2",  
1603: "ladoB": "",  
1604: "usr": "200"  
1605: },  
1606: {  
1607: "trecho": "ACESSO FLORESTAL - ENTR BR-352 (PARÁ DE MINAS)",  
1608: "pwd": "0685",  
1609: "ladoA": "",  
1610: "fase": "3",  
1611: "ladoB": "",  
1612: "usr": "201"  
1613: },  
1614: {  
1615: "trecho": "ENTR MG-050(B) (P/DIVINOPÓLIS) - ENTR MG-260 (P/CLÁUDIO)",  
1616: "pwd": "0050",  
1617: "ladoA": "",  
1618: "fase": "2",  
1619: "ladoB": "",  
1620: "usr": "202"  
1621: },  
1622: {  
1623: "trecho": "ACESSO CONCEIÇÃO DO RIO VERDE - ENTR BR-460",  
1624: "pwd": "0270",  
1625: "ladoA": "",  
1626: "fase": "2",  
1627: "ladoB": "",  
1628: "usr": "203"  
1629: },  
1630: {  
1631: "trecho": "ACESSO SANTA RITA DAS CALDAS - ENTR MG-179 (POUSO ALEGRE)",  
1632: "pwd": "0023",  
1633: "ladoA": "",  
1634: "fase": "3",  
1635: "ladoB": "",  
1636: "usr": "204"  
1637: },  
1638: {  
1639: "trecho": "ENTR BR-369 (BOTELHOS) - ENTR BR-267(A) (P/BANDEIRA DO SUL)",  
1640: "pwd": "0270",  
1641: "ladoA": "",  
1642: "fase": "3",  
1643: "ladoB": "",  
1644: "usr": "205"  
1645: },  
1646: {  
1647: "trecho": "ENTR BR-487(A) - ACESSO NAVIRAÍ I",  
1648: "pwd": "0212",  
1649: "ladoA": "",  
1650: "fase": "3",  
1651: "ladoB": "",  
1652: "usr": "206"  
1653: },  
1654: {  
1655: "trecho": "ENTR BR-267(A) - INÍCIO DUPLICAÇÃO",  
1656: "pwd": "0612",  
1657: "ladoA": "ENTR BR-267(A)",  
1658: "fase": "1",  
1659: "ladoB": "INÍCIO DUPLICAÇÃO",  
1660: "usr": "207"  
1661: },  
1662: {  
1663: "trecho": "ENTR MS-141 (IVINHEMA) - VILA AMANDINA",  
1664: "pwd": "0090",  
1665: "ladoA": "",  
1666: "fase": "3",  
1667: "ladoB": "",  
1668: "usr": "208"  
1669: },  
1670: {  
1671: "trecho": "ENTR MS-339/446/448 (MIRANDA) - ENTR MS-185/243 (GUAICURUS)",  
1672: "pwd": "1390",  
1673: "ladoA": "ENTR MS-339/446/448 (MIRANDA)",  
1674: "fase": "1",  
1675: "ladoB": "ENTR MS-185/243 (GUAICURUS)",  
1676: "usr": "209"  
1677: },  
1678: {  
1679: "trecho": "BURITIZINHO - ENTR MS-223(A) (SILVOLÂNDIA)",  
1680: "pwd": "0058",  
1681: "ladoA": "BURITIZINHO",  
1682: "fase": "1",  
1683: "ladoB": "ENTR MS-223(A) (SILVOLÂNDIA)",  
1684: "usr": "210"  
1685: },  
1686: {  
1687: "trecho": "ENTR MS-316 (PARAÍSO) - ENTR MS-324(A)",  
1688: "pwd": "0390",  
1689: "ladoA": "ENTR MS-316 (PARAÍSO DAS ÁGUAS)",  
1690: "fase": "1",  
1691: "ladoB": "ENTR MS-324(A)",  
1692: "usr": "211"  
1693: },  
1694: {  
1695: "trecho": "ENTR BR-158(B) (P/TAMBOARA) - ACESSO ALTO PARANÁ/MARISTELA",  
1696: "pwd": "0180",  
1697: "ladoA": "",  
1698: "fase": "3",  
1699: "ladoB": "",  
1700: "usr": "212"  
1701: },  
1702: {  
1703: "trecho": "ENTR SP-425 (PRESIDENTE PRUDENTE) - ENTR SP-425",  
1704: "pwd": "0050",  
1705: "ladoA": "",  
1706: "fase": "2",  
1707: "ladoB": "",  
1708: "usr": "213"  
1709: },  
1710: {  
1711: "trecho": "ENTR MT-373 - ENTR MT-130(A) (PRIMAVERA DO LESTE)",  
1712: "pwd": "0330",  
1713: "ladoA": "ENTR MT-373",  
1714: "fase": "1",  
1715: "ladoB": "ENTR MT-130(A) (PRIMAVERA DO LESTE)",  
1716: "usr": "214"  
1717: },  
1718: {  
1719: "trecho": "FIM VARIANTE II SERRA DE SÃO VICENTE - ACESSO DISTRITO INDUSTRIAL",  
1720: "pwd": "0765",  
1721: "ladoA": "",  
1722: "fase": "2",  
1723: "ladoB": "",  
1724: "usr": "215"  
1725: },  
1726: {  
1727: "trecho": "ENTR BR-010(B) (P/SANTA MARIA DO PARÁ) - ENTR PA-324 (P/SALINÓPOLIS)",  
1728: "pwd": "0150",  
1729: "ladoA": "",  
1730: "fase": "3",  
1731: "ladoB": "",  
1732: "usr": "216"  
1733: },  
1734: {  
1735: "trecho": "ENTR PB-177 (SOLEDADE) - ENTR PB-195 (JUAZEIRINHO)",  
1736: "pwd": "0270",  
1737: "ladoA": "",  
1738: "fase": "3",  
1739: "ladoB": "",  
1740: "usr": "217"  
1741: },  
1742: {  
1743: "trecho": "BARRA DE SANTANA - ENTR PB-196 (P/RIACHO DE SANTO ANTÔNIO)",  
1744: "pwd": "0322",  
1745: "ladoA": "",  
1746: "fase": "2",  
1747: "ladoB": "",  
1748: "usr": "218"  
1749: },  
1750: {  
1751: "trecho": "ENTR PE-263 (P/ITAPETIM) - ENTR PE-320 (SÃO JOSÉ DO EGITO)",  
1752: "pwd": "0280",  
1753: "ladoA": "",  
1754: "fase": "2",  
1755: "ladoB": "",  
1756: "usr": "219"  
1757: },  
1758: {  
1759: "trecho": "ENTR PE-015 (P/PAULISTA - VIADUTO SOBRE PE-015) - VIADUTO SOBRE AVENIDA CAXANGÁ",  
1760: "pwd": "0410",  
1761: "ladoA": "",  
1762: "fase": "3",  
1763: "ladoB": "",  
1764: "usr": "220"  
1765: },  
1766: {  
1767: "trecho": "DIV PE/AL - ENTR BR-423(A)",  
1768: "pwd": "0530",  
1769: "ladoA": "",  
1770: "fase": "2",  
1771: "ladoB": "",  
1772: "usr": "221"  
1773: },  
1774: {  
1775: "trecho": "ENTR PE-590 (P/IPUBÍ) - ENTR BR-122 (OURICURÍ)",  
1776: "pwd": "0630",  
1777: "ladoA": "",  
1778: "fase": "2",  
1779: "ladoB": "",  
1780: "usr": "222"  
1781: },  
1782: {  
1783: "trecho": "ENTR CE-292(B) - DIV CE/PE",  
1784: "pwd": "0250",  
1785: "ladoA": "",  
1786: "fase": "3",  
1787: "ladoB": "",  
1788: "usr": "223"  
1789: },  
1790: {  
1791: "trecho": "ENTR PE-310/312 (CUSTÓDIA) - ENTR BR-426/PE-340 (SÍTIO DOS NUNES)",  
1792: "pwd": "0300",  
1793: "ladoA": "",  
1794: "fase": "3",  
1795: "ladoB": "",  
1796: "usr": "224"  
1797: },  
1798: {  
1799: "trecho": "RAJADA - ENTR BR-235(A)",  
1800: "pwd": "0250",  
1801: "ladoA": "",  
1802: "fase": "2",  
1803: "ladoB": "",  
1804: "usr": "225"  
1805: },  
1806: {  
1807: "trecho": "ENTR PI-394 - ENTR PI-140 (CANTO DO BURITI)",  
1808: "pwd": "0115",  
1809: "ladoA": "",  
1810: "fase": "2",  
1811: "ladoB": "",  
1812: "usr": "226"  
1813: },  
1814: {  
1815: "trecho": "ENTR PI-211 (CAXINGÓ) - ENTR PI-213(A)",  
1816: "pwd": "0070",  
1817: "ladoA": "",  
1818: "fase": "3",  
1819: "ladoB": "",  
1820: "usr": "227"  
1821: },  
1822: {  
1823: "trecho": "ACESSO OESTE SOBRAL - ENTR BR-403(B)/CE-183",  
1824: "pwd": "0190",  
1825: "ladoA": "",  
1826: "fase": "2",  
1827: "ladoB": "",  
1828: "usr": "228"  
1829: },  
1830: {  
1831: "trecho": "ENTR MA-034 (SÃO JOÃO DOS PATOS) - ENTR BR-230(B)/MA-364(A) (DOIS IRMÃOS)",  
1832: "pwd": "0330",  
1833: "ladoA": "",  
1834: "fase": "2",  
1835: "ladoB": "",  
1836: "usr": "229"  
1837: },  
1838: {  
1839: "trecho": "ENTR PI-130 (P/AMARANTE) - ENTR PI-120",  
1840: "pwd": "0290",  
1841: "ladoA": "",  
1842: "fase": "2",  
1843: "ladoB": "",  
1844: "usr": "230"  
1845: },  
1846: {  
1847: "trecho": "ENTR SP-252 (GUAPIARA) - ENTR SP-165/249 (APIAÍ)",  
1848: "pwd": "0210",  
1849: "ladoA": "",  
1850: "fase": "3",  
1851: "ladoB": "",  
1852: "usr": "231"  
1853: },  
1854: {  
1855: "trecho": "ENTR PR-549 (P/LUIZIANA) - ENTR PR-462 (P/IRETAMA)",  
1856: "pwd": "0180",  
1857: "ladoA": "ENTR PR-549 (P/LUIZIANA)",  
1858: "fase": "1",  
1859: "ladoB": "ENTR PR-462 (P/IRETAMA)",  
1860: "usr": "232"  
1861: },  
1862: {  
1863: "trecho": "ACESSO IVAIPORÃ - ENTR BR-487 (P/MANOEL RIBAS)",  
1864: "pwd": "0080",  
1865: "ladoA": "",  
1866: "fase": "3",  
1867: "ladoB": "",  
1868: "usr": "233"  
1869: },  
1870: {  
1871: "trecho": "ENTR BR-158(B) (LARANJEIRAS DO SUL) - ENTR PR-473",  
1872: "pwd": "0250",  
1873: "ladoA": "ENTR BR-158(B) (LARANJEIRAS DO SUL)",  
1874: "fase": "1",  
1875: "ladoB": "ENTR PR-473",  
1876: "usr": "234"  
1877: },  
1878: {  
1879: "trecho": "ACESSO ÁGUA MINERAL SANTA CLARA - ENTR PR-281(A) (P/CHOPINZINHO)",  
1880: "pwd": "0455",  
1881: "ladoA": "ACESSO ÁGUA MINERAL SANTA CLARA",  
1882: "fase": "1",  
1883: "ladoB": "ENTR PR-281(A) (P/CHOPINZINHO)",  
1884: "usr": "235"  
1885: },  
1886: {  
1887: "trecho": "ENTR PR-468(B) - ENTR PR-180 (GOIO ERÊ)",  
1888: "pwd": "0500",  
1889: "ladoA": "ENTR PR-468(B)",  
1890: "fase": "1",  
1891: "ladoB": "ENTR PR-180 (GOIO ERÊ)",  
1892: "usr": "236"  
1893: },  
1894: {  
1895: "trecho": "ENTR PR-431(B) - ENTR PR-092(A) (P/BARRA DO JACARÉ)",  
1896: "pwd": "1230",  
1897: "ladoA": "ENTR PR-431(B)",  
1898: "fase": "1",  
1899: "ladoB": "ENTR PR-092(A) (P/BARRA DO JACARÉ)",  
1900: "usr": "237"  
1901: },  
1902: {  
1903: "trecho": "ENTR SP-331(A) - ENTR SP-331(B)",  
1904: "pwd": "1140",  
1905: "ladoA": "",  
1906: "fase": "2",  
1907: "ladoB": "",  
1908: "usr": "238"  
1909: },  
1910: {  
1911: "trecho": "ENTR BR-369 - ENTR SP-261 (P/ÁGUAS DE SANTA BÁRBARA)",  
1912: "pwd": "0230",  
1913: "ladoA": "",  
1914: "fase": "3",  
1915: "ladoB": "",  
1916: "usr": "239"  
1917: },  
1918: {  
1919: "trecho": "ACESSO PIRAPÓ - ACESSO CAMBIRA",  
1920: "pwd": "0710",  
1921: "ladoA": "ACESSO PIRAPÓ",  
1922: "fase": "1",  
1923: "ladoB": "ACESSO CAMBIRA",  
1924: "usr": "240"  
1925: },  
1926: {  
1927: "trecho": "ACESSO PARAÍBA DO SUL - ENTR RJ-135 (ANDRADE PINTO)",  
1928: "pwd": "0410",  
1929: "ladoA": "",  
1930: "fase": "3",  
1931: "ladoB": "",  
1932: "usr": "241"  
1933: },  
1934: {  
1935: "trecho": "ENTR RJ-196 - ENTR RJ-182",  
1936: "pwd": "2850",  
1937: "ladoA": "",  
1938: "fase": "4",  
1939: "ladoB": "",  
1940: "usr": "242"  
1941: },  
1942: {  
1943: "trecho": "ITAMONTE - ENTR BR-485 (GARGANTA DO REGISTRO)",  
1944: "pwd": "0590",  
1945: "ladoA": "",  
1946: "fase": "3",  
1947: "ladoB": "",  
1948: "usr": "243"  
1949: },  
1950: {  
1951: "trecho": "ENTR RJ-149 (P/RIO CLARO) - ENTR BR-494(A) (ANGRA DOS REIS)",  
1952: "pwd": "3350",  
1953: "ladoA": "",  
1954: "fase": "3",  
1955: "ladoB": "",  
1956: "usr": "244"  
1957: },  
1958: {  
1959: "trecho": "ENTR RJ-145 (P/BARRA DO PIRAÍ) - ENTR RJ-141 (DORANDIA)",  
1960: "pwd": "0530",  
1961: "ladoA": "",  
1962: "fase": "3",  
1963: "ladoB": "",  
1964: "usr": "245"  
1965: },  
1966: {  
1967: "trecho": "ENTR BR-494 (ARANTINA) - AIURUOCA",  
1968: "pwd": "0170",  
1969: "ladoA": "",  
1970: "fase": "4",  
1971: "ladoB": "",  
1972: "usr": "246"  
1973: },  
1974: {  
1975: "trecho": "ENTR RJ-198(B) - ENTR RJ-200 (MONTE ALEGRE)",  
1976: "pwd": "0170",  
1977: "ladoA": "",  
1978: "fase": "2",  
1979: "ladoB": "",  
1980: "usr": "247"  
1981: },  
1982: {  
1983: "trecho": "ENTR RJ-202 - ENTR RJ-234 (P/PENEDO)",  
1984: "pwd": "0350",  
1985: "ladoA": "",  
1986: "fase": "4",  
1987: "ladoB": "",  
1988: "usr": "248"  
1989: },  
1990: {  
1991: "trecho": "ENTR BR-482(B) - ENTR ES-181",  
1992: "pwd": "0150",  
1993: "ladoA": "",  
1994: "fase": "3",  
1995: "ladoB": "",  
1996: "usr": "249"  
1997: },  
1998: {  
1999: "trecho": "ENTR BR-116/265 (MURIAÉ) - DIV MG/RJ",  
2000: "pwd": "0210",  
2001: "ladoA": "",  
2002: "fase": "3",  
2003: "ladoB": "",  
2004: "usr": "250"  
2005: },  
2006: {  
2007: "trecho": "ENTR RJ-156 (P/VOLTA DO PIÃO) - ENTR BR-492(A)",  
2008: "pwd": "1530",  
2009: "ladoA": "",  
2010: "fase": "2",  
2011: "ladoB": "",  
2012: "usr": "251"  
2013: },  
2014: {  
2015: "trecho": "ACESSO À JENIPABU - ENTR RN-160(B) (P/EXTREMOZ)",  
2016: "pwd": "0065",  
2017: "ladoA": "",  
2018: "fase": "3",  
2019: "ladoB": "",  
2020: "usr": "252"  
2021: },  
2022: {  
2023: "trecho": "TAIPÚ - ENTR RN-064 (P/CEARÁ MIRIM)",  
2024: "pwd": "0110",  
2025: "ladoA": "",  
2026: "fase": "3",  
2027: "ladoB": "",  
2028: "usr": "253"  
2029: },  
2030: {  
2031: "trecho": "ENTR CE-261 - DIV CE/RN",  
2032: "pwd": "0055",  
2033: "ladoA": "",  
2034: "fase": "3",  
2035: "ladoB": "",  
2036: "usr": "254"  
2037: },  
2038: {  
2039: "trecho": "ENTR BR-230(A) (SANTA GERTRUDES) - ENTR PB-275(A) (P/SÃO JOSÉ DE ESPINHARA)",  
2040: "pwd": "0210",  
2041: "ladoA": "",  
2042: "fase": "3",  
2043: "ladoB": "",  
2044: "usr": "255"  
2045: },  
2046: {  
2047: "trecho": "ACESSO USINA DE SAMUEL - INICIO PISTA DUPLA",  
2048: "pwd": "1375",  
2049: "ladoA": "",  
2050: "fase": "3",  
2051: "ladoB": "",  
2052: "usr": "256"  
2053: },  
2054: {  
2055: "trecho": "VISTA ALEGRE DO ABUANÃ - VILA EXTREMA",  
2056: "pwd": "1510",  
2057: "ladoA": "",  
2058: "fase": "3",  
2059: "ladoB": "",  
2060: "usr": "257"  
2061: },  
2062: {  
2063: "trecho": "ENTR BR-421 (ARIQUEMES) - ENTR RO-459(A) (ALTO PARAISO)",  
2064: "pwd": "1300",  
2065: "ladoA": "",  
2066: "fase": "2",  
2067: "ladoB": "",  
2068: "usr": "258"  
2069: },  
2070: {  
2071: "trecho": "ENTR RO-135 (P/NOVA LONDRINA) - FIM DO TRECHO PAVIMENTADO",  
2072: "pwd": "0031",  
2073: "ladoA": "",  
2074: "fase": "2",  
2075: "ladoB": "",  
2076: "usr": "259"  
2077: },  
2078: {  
2079: "trecho": "ENTR RO-471 (P/MINISTRO ANDREAZZA) - ENTR RO-479(A) (P/ROLIM DE MOURA)",  
2080: "pwd": "1148",  
2081: "ladoA": "",  
2082: "fase": "2",  
2083: "ladoB": "",  
2084: "usr": "260"  
2085: },  
2086: {  
2087: "trecho": "ENTR RO-391 (POSTO GUAPORÉ) - MARCO RONDON",  
2088: "pwd": "1090",  
2089: "ladoA": "",  
2090: "fase": "2",  
2091: "ladoB": "",  
2092: "usr": "261"  
2093: },  
2094: {  
2095: "trecho": "PADRONAL - RIO 12 DE OUTUBRO",  
2096: "pwd": "0132",  
2097: "ladoA": "PADRONAL",  
2098: "fase": "1",  
2099: "ladoB": "RIO 12 DE OUTUBRO",  
2100: "usr": "262"  
2101: },  
2102: {  
2103: "trecho": "ENTR ACESSO SUL DE CAPÃO DA CANOA - ENTR BR-290(A) (OSÓRIO)",  
2104: "pwd": "4425",  
2105: "ladoA": "ENTR ACESSO SUL DE CAPÃO DA CANOA",  
2106: "fase": "1",  
2107: "ladoB": "ENTR BR-290(A) (OSÓRIO)",  
2108: "usr": "263"  
2109: },  
2110: {  
2111: "trecho": "ENTR RSC-470 (P/ MONTENEGRO) - ENTR ERS-124 (P/ POLO PETROQUIMICO)",  
2112: "pwd": "0330",  
2113: "ladoA": "ENTR BR-470",  
2114: "fase": "1",  
2115: "ladoB": "ENTR BR-470/116(A) (CANOAS)",  
2116: "usr": "264"  
2117: },  
2118: {  
2119: "trecho": "ENTR BRS-153(B)/ERS-324 (P/ PONTAO) - ENTR BRS-377(A)/386 (P/ CARAZINHO)",  
2120: "pwd": "0210",  
2121: "ladoA": "ENTR BR-153(B)/RS-324 (P/PONTÃO)",  
2122: "fase": "1",  
2123: "ladoB": "ENTR BR-377(A)/386 (CARAZINHO)",  
2124: "usr": "265"  
2125: },  
2126: {  
2127: "trecho": "ENTR RS-168 (P/SÃO PAULO DAS MISSÕES) - ENTR BR-472 (FRONT BRASIL/ARGENTINA) (PORTO XAVIER)",  
2128: "pwd": "0470",  
2129: "ladoA": "ENTR RS-168 (P/SÃO PAULO DAS MISSÕES)",  
2130: "fase": "1",  
2131: "ladoB": "ENTR BR-472 (FRONT BRASIL/ARGENTINA) (PORTO XAVIER)",  
2132: "usr": "266"  
2133: },  
2134: {  
2135: "trecho": "ENTR ERS-533 (P/ CAPAO CIPO) - ENTR BRS-287 (CONTORNO DE SANTIAGO)",  
2136: "pwd": "0140",  
2137: "ladoA": "A. BELTRÃO",  
2138: "fase": "1",  
2139: "ladoB": "ENTR BR-287 (SANTIAGO)",  
2140: "usr": "267"  
2141: },  
2142: {  
2143: "trecho": "ENTR BRS-290(B) (P/ SAO GABRIEL) - ENTR BRS-392 (P/ CACAPAVA DO SUL)",  
2144: "pwd": "1850",  
2145: "ladoA": "ENTR BR-290(B)",  
2146: "fase": "3",  
2147: "ladoB": "ENTR BR-392",  
2148: "usr": "268"  
2149: },  
2150: {  
2151: "trecho": "ENTR BRS-158(A) (P/ AZEVEDO SODRE) - ENTR ERS-640 (P/ CACEQUI)",  
2152: "pwd": "1350",  
2153: "ladoA": "ENTR BR-290(A) (P/ROSÁRIO DO SUL)",  
2154: "fase": "1",  
2155: "ladoB": "ENTR RS-640 (P/CACEQUI)",  
2156: "usr": "269"  
2157: },  
2158: {  
2159: "trecho": "ENTR RSC-473 (VILA SARANDI) - CURRAL ALTO",  
2160: "pwd": "0230",  
2161: "ladoA": "ENTR BR-473 (SARANDI)",  
2162: "fase": "1",  
2163: "ladoB": "CURRAL ALTO",  
2164: "usr": "270"  
2165: },  
2166: {  
2167: "trecho": "ENTR ERS-476 (LAJEADO GRANDE) - ENTR ERS-110 (VARZEA DO CEDRO)",  
2168: "pwd": "0310",  
2169: "ladoA": "ENTR RS-476 (LAJEADO GRANDE)",  
2170: "fase": "1",  
2171: "ladoB": "ENTR RS-110 (VÁRZEA DO CEDRO)",  
2172: "usr": "271"  
2173: },  
2174: {  
2175: "trecho": "ENTR BR-282(B)/470(A) - ENTR SC-455 (P/IBICUÍ)",  
2176: "pwd": "0015",  
2177: "ladoA": "ENTR BR-282(B)/470(A)",  
2178: "fase": "1",  
2179: "ladoB": "ENTR SC-455 (P/IBICUÍ)",  
2180: "usr": "272"  
2181: },  
2182: {  
2183: "trecho": "DIVISA SC/RS (RIO PELOTAS) - ENTR BRS-285(A) (P/ VACARIA)",  
2184: "pwd": "3010",  
2185: "ladoA": "DIV SC/RS (RIO PELOTAS)",  
2186: "fase": "1",  
2187: "ladoB": "ENTR BR-285(A) (P/VACARIA)",  
2188: "usr": "273"  
2189: },  
2190: {  
2191: "trecho": "ENTR BR-486 (BOM RETIRO) - ENTR SC-345(B) (P/URUBICI)",  
2192: "pwd": "0110",  
2193: "ladoA": "ENTR BR-486 (BOM RETIRO)",  
2194: "fase": "1",  
2195: "ladoB": "ENTR SC-345(B) (P/URUBICI)",  
2196: "usr": "274"  
2197: },  
2198: {  
2199: "trecho": "ENTR SC-473 (P/ANCHIETA) - SÃO JOSÉ DO CEDRO",  
2200: "pwd": "0027",  
2201: "ladoA": "ENTR SC-473 (P/ANCHIETA)",  
2202: "fase": "1",  
2203: "ladoB": "SÃO JOSÉ DO CEDRO",  
2204: "usr": "275"  
2205: },  
2206: {  
2207: "trecho": "ENTR BR-280(A) - ENTR BR-280(B) (VITORINO)",  
2208: "pwd": "0290",  
2209: "ladoA": "",  
2210: "fase": "3",  
2211: "ladoB": "",  
2212: "usr": "276"  
2213: },  
2214: {  
2215: "trecho": "ENTR BA-392 (ANTAS) - ENTR BA-220 (CÍCERO DANTAS)",  
2216: "pwd": "0612",  
2217: "ladoA": "",  
2218: "fase": "2",  
2219: "ladoB": "",  
2220: "usr": "277"  
2221: },  
2222: {  
2223: "trecho": "ENTR SP-183 - ENTR BR-116(A) (P/LORENA)",  
2224: "pwd": "0190",  
2225: "ladoA": "",  
2226: "fase": "3",  
2227: "ladoB": "",  
2228: "usr": "278"  
2229: },  
2230: {  
2231: "trecho": "ENTR SP-062 (P/ROSEIRA) - ENTR BR-383(A)/SP-132 (PINDAMONHANGABA)",  
2232: "pwd": "2360",  
2233: "ladoA": "",  
2234: "fase": "3",  
2235: "ladoB": "",  
2236: "usr": "279"  
2237: },  
2238: {  
2239: "trecho": "ENTR SP-065 (P/IGARATÁ) - ENTR SP-056 (ARUJÁ)",  
2240: "pwd": "2470",  
2241: "ladoA": "",  
2242: "fase": "3",  
2243: "ladoB": "",  
2244: "usr": "280"  
2245: },  
2246: {  
2247: "trecho": "ENTR BR-116(A) (JUQUIÁ) - ENTR SP-139 (REGISTRO)",  
2248: "pwd": "2630",  
2249: "ladoA": "",  
2250: "fase": "3",  
2251: "ladoB": "",  
2252: "usr": "281"  
2253: },  
2254: {  
2255: "trecho": "ENTR BR-373(A) (ITAPETININGA) - ENTR BR-373(B)",  
2256: "pwd": "0130",  
2257: "ladoA": "",  
2258: "fase": "3",  
2259: "ladoB": "",  
2260: "usr": "282"  
2261: },  
2262: {  
2263: "trecho": "ACESSO GUAIMBÉ - ENTR SP-333(A)",  
2264: "pwd": "1077",  
2265: "ladoA": "",  
2266: "fase": "3",  
2267: "ladoB": "",  
2268: "usr": "283"  
2269: },  
2270: {  
2271: "trecho": "ENTR SP-355 - ENTR SP-425",  
2272: "pwd": "1010",  
2273: "ladoA": "",  
2274: "fase": "2",  
2275: "ladoB": "",  
2276: "usr": "284"  
2277: },  
2278: {  
2279: "trecho": "ENTR SP-255 (ARARAQUARA) - ENTR BR-364(B)",  
2280: "pwd": "0110",  
2281: "ladoA": "",  
2282: "fase": "3",  
2283: "ladoB": "",  
2284: "usr": "285"  
2285: },  
2286: {  
2287: "trecho": "ACESSO RINÓPOLIS - ENTR SP-294 (PARAPUÃ)",  
2288: "pwd": "0783",  
2289: "ladoA": "",  
2290: "fase": "3",  
2291: "ladoB": "",  
2292: "usr": "286"  
2293: },  
2294: {  
2295: "trecho": "ENTR SP-065 (ATIBAIA) - ENTR SP-023 (MARIPORÃ)",  
2296: "pwd": "0850",  
2297: "ladoA": "",  
2298: "fase": "3",  
2299: "ladoB": "",  
2300: "usr": "287"  
2301: },  
2302: {  
2303: "trecho": "ENTR BR-222/230(A)/PA-150 (MARABÁ) - ENTR PA-405",  
2304: "pwd": "0010",  
2305: "ladoA": "",  
2306: "fase": "3",  
2307: "ladoB": "",  
2308: "usr": "288"  
2309: },  
2310: {  
2311: "trecho": "ENTR BR-010(A) (PORTO FRANCO) - ENTR BR-010(B)/230(A)/MA-138 (ESTREITO)",  
2312: "pwd": "0390",  
2313: "ladoA": "",  
2314: "fase": "2",  
2315: "ladoB": "",  
2316: "usr": "289"  
2317: },  
2318: {  
2319: "trecho": "ENTR MA-012/375 (SÃO RAIMUNDO DAS MANGABEIRAS) - ENTR MA-006(A)",  
2320: "pwd": "1010",  
2321: "ladoA": "",  
2322: "fase": "3",  
2323: "ladoB": "",  
2324: "usr": "290"  
2325: },  
2326: {  
2327: "trecho": "FIM PISTA DUPLA - GURUPÍ - ENTR BR-242(B)/TO-280",  
2328: "pwd": "0290",  
2329: "ladoA": "",  
2330: "fase": "3",  
2331: "ladoB": "",  
2332: "usr": "291"  
2333: },  
2334: {  
2335: "trecho": "ENTR BR-020(A)/135(B)/BA-020 - ENTR BR-020(B)",  
2336: "pwd": "0330",  
2337: "ladoA": "",  
2338: "fase": "4",  
2339: "ladoB": "",  
2340: "usr": "292"  
2341: },  
2342: {  
2343: "trecho": "INDIANÓPOLIS - VALE DO SONHO",  
2344: "pwd": "0272",  
2345: "ladoA": "",  
2346: "fase": "3",  
2347: "ladoB": "",  
2348: "usr": "293"  
2349: },  
2350: {  
2351: "trecho": "ENTR GO-217 (P/MAIRIPOTABA) - ENTR GO-320 (INDIARA)",  
2352: "pwd": "0210",  
2353: "ladoA": "",  
2354: "fase": "4",  
2355: "ladoB": "",  
2356: "usr": "294"  
2357: },  
2358: {  
2359: "trecho": "ENTR BR-153 (P/TIBAGI/IPIRANGA) - ENTR BR-373(A)/487(A) (CAETANO)",  
2360: "pwd": "0370",  
2361: "ladoA": "ENTR BR-153 (P/TIBAGI/IPIRANGA)",  
2362: "fase": "1",  
2363: "ladoB": "ENTR BR-373(A)/487(A) (CAETANO)",  
2364: "usr": "295"  
2365: },  
2366: {  
2367: "trecho": "RIO ANHANDUÍ - ENTR BR-262(A) (CAMPO GRANDE)",  
2368: "pwd": "0390",  
2369: "ladoA": "RIO ANHANDUÍ",  
2370: "fase": "1",  
2371: "ladoB": "ENTR BR-262(A) (CAMPO GRANDE)",  
2372: "usr": "296"  
2373: },  
2374: {  
2375: "trecho": "ACESSO ITAJÁ (GO) - ENTR MS-431 (P/SÃO JOÃO DO APORÉ)",  
2376: "pwd": "0434",  
2377: "ladoA": "",  
2378: "fase": "3",  
2379: "ladoB": "",  
2380: "usr": "297"  
2381: },  
2382: {  
2383: "trecho": "FNM (ENTR PISTA INVERSA) - ENTR BR-116(A)/493/RJ-109",  
2384: "pwd": "0930",  
2385: "ladoA": "",  
2386: "fase": "3",  
2387: "ladoB": "",  
2388: "usr": "298"  
2389: },  
2390: {  
2391: "trecho": "ENTR SC-418 (P/POMERODE) - ENTR BR-477(B) (P/TIMBÓ)",  
2392: "pwd": "0110",  
2393: "ladoA": "ENTR SC-418 (P/POMERODE)",  
2394: "fase": "1",  
2395: "ladoB": "ENTR BR-477(B) (P/TIMBÓ)",  
2396: "usr": "299"  
2397: },  
2398: {  
2399: "trecho": "ENTR BR-116(B) - P/TRES BARRAS",  
2400: "pwd": "0110",  
2401: "ladoA": "ENTR BR-116(B)",  
2402: "fase": "1",  
2403: "ladoB": "P/TRES BARRAS",  
2404: "usr": "300"  
2405: },  
2406: {  
2407: "trecho": "DIV PI/BA - ENTR BR-235/BA-161 (REMANSO)",  
2408: "pwd": "0150",  
2409: "ladoA": "",  
2410: "fase": "2",  
2411: "ladoB": "",  
2412: "usr": "301"  
2413: },  
2414: {  
2415: "trecho": "ENTR BR-369(A) (BARBOSA FERRAZ) - ENTR BR-158(A)/369(B) (ANEL VIÁRIO CAMPO MOURÃO)",  
2416: "pwd": "0770",  
2417: "ladoA": "",  
2418: "fase": "2",  
2419: "ladoB": "",  
2420: "usr": "302"  
2421: }  
2422: ]  
2423:}

## .\concentrador\src\logins\users\textunderscore fase2\textunderscore old.json

1:{  
 2: "logins": [  
 3: {  
 4: "usr" : "001",  
 5: "pwd" : "1590",  
 6: "fase" : "3",  
 7: "ladoA" : "",  
 8: "ladoB" : "",  
 9: "trecho" : "ENTR AC-463/465 (BOM FUTURO) - ENTR BR-317"  
 10: }, {  
 11: "usr" : "002",  
 12: "pwd" : "0808",  
 13: "fase" : "1",  
 14: "ladoA" : "LUCAS DO RIO VERDE",  
 15: "ladoB" : "INÍCIO DA TRAVESSIA URBANA DE SORRISO",  
 16: "trecho" : "LUCAS DO RIO VERDE – INÍCIO DA TRAVESSIA URBANA DE SORRISO"  
 17: }, {  
 18: "usr" : "003",  
 19: "pwd" : "0630",  
 20: "fase" : "2",  
 21: "ladoA" : "",  
 22: "ladoB" : "",  
 23: "trecho" : "ENTR BR-104(A) - ENTR BR-104(B)"  
 24: }, {  
 25: "usr" : "004",  
 26: "pwd" : "0890",  
 27: "fase" : "2",  
 28: "ladoA" : "",  
 29: "ladoB" : "",  
 30: "trecho" : "ENTR AL-225(A) - ENTR AL-225(B) (P/PORTO REAL DO COLÉGIO)"  
 31: }, {  
 32: "usr" : "005",  
 33: "pwd" : "0950",  
 34: "fase" : "3",  
 35: "ladoA" : "",  
 36: "ladoB" : "",  
 37: "trecho" : "ENTR AL-115(A) - ENTR AL-115(B) (PALMEIRA DOS ÍNDIOS)"  
 38: }, {  
 39: "usr" : "006",  
 40: "pwd" : "0750",  
 41: "fase" : "4",  
 42: "ladoA" : "",  
 43: "ladoB" : "",  
 44: "trecho" : "ENTR BR-101(A) - ENTR BR-101(B) (P/ALAGOINHAS)"  
 45: }, {  
 46: "usr" : "007",  
 47: "pwd" : "1810",  
 48: "fase" : "3",  
 49: "ladoA" : "",  
 50: "ladoB" : "",  
 51: "trecho" : "ENTR BA-120(A)/262(B) (P/ITAJUÍPE) - ENTR BR-415/BA-120(B) (ITABUNA)"  
 52: }, {  
 53: "usr" : "008",  
 54: "pwd" : "0330",  
 55: "fase" : "2",  
 56: "ladoA" : "",  
 57: "ladoB" : "",  
 58: "trecho" : "ENTR BR-324(A) - ENTR BR-324(B)/BA-502/503 (FEIRA DE SANTANA)"  
 59: }, {  
 60: "usr" : "009",  
 61: "pwd" : "0934",  
 62: "fase" : "3",  
 63: "ladoA" : "",  
 64: "ladoB" : "",  
 65: "trecho" : "ENTR BA-643 (PLANALTO) - ENTR BA-641"  
 66: }, {  
 67: "usr" : "010",  
 68: "pwd" : "0365",  
 69: "fase" : "3",  
 70: "ladoA" : "",  
 71: "ladoB" : "",  
 72: "trecho" : "ENTR BR-242(A) - ENTR BR-330/BA-148 (SEABRA)"  
 73: }, {  
 74: "usr" : "011",  
 75: "pwd" : "0300",  
 76: "fase" : "2",  
 77: "ladoA" : "",  
 78: "ladoB" : "",  
 79: "trecho" : "JAGUARARI - ENTR BA-220 (SENHOR DO BONFIM)"  
 80: }, {  
 81: "usr" : "012",  
 82: "pwd" : "0530",  
 83: "fase" : "3",  
 84: "ladoA" : "",  
 85: "ladoB" : "",  
 86: "trecho" : "ENTR CE-187 (P/MARRECOS) - ENTR BR-404/CE-176/187/363 (P/TAUÁ)"  
 87: }, {  
 88: "usr" : "013",  
 89: "pwd" : "0090",  
 90: "fase" : "3",  
 91: "ladoA" : "",  
 92: "ladoB" : "",  
 93: "trecho" : "ENTR CE-253 (PACAJÚS) - ENTR BR-122(A)/CE-354 (CHORÓZINHO)"  
 94: }, {  
 95: "usr" : "014",  
 96: "pwd" : "0260",  
 97: "fase" : "2",  
 98: "ladoA" : "",  
 99: "ladoB" : "",  
 100: "trecho" : "ENTR BR-226(B)/CE-275(B) (P/PEREIRO) - ENTR BR-404/434 (ICÓ)"  
 101: }, {  
 102: "usr" : "015",  
 103: "pwd" : "2195",  
 104: "fase" : "2",  
 105: "ladoA" : "",  
 106: "ladoB" : "",  
 107: "trecho" : "ENTR ES-245(B) - ENTR ES-440"  
 108: }, {  
 109: "usr" : "016",  
 110: "pwd" : "2470",  
 111: "fase" : "3",  
 112: "ladoA" : "",  
 113: "ladoB" : "",  
 114: "trecho" : "ENTR ES-375(A) (P/PIÚMA) - ENTR ES-375(B) (ICONHA)"  
 115: }, {  
 116: "usr" : "017",  
 117: "pwd" : "0070",  
 118: "fase" : "3",  
 119: "ladoA" : "",  
 120: "ladoB" : "",  
 121: "trecho" : "ENTR BR-101(B) - ENTR ES-465 (P/DOMINGOS MARTINS)"  
 122: }, {  
 123: "usr" : "018",  
 124: "pwd" : "0015",  
 125: "fase" : "2",  
 126: "ladoA" : "",  
 127: "ladoB" : "",  
 128: "trecho" : "KM 15,3 - ENTR BR-484 (P/PONTE SOBRE RIO DOCE)"  
 129: }, {  
 130: "usr" : "019",  
 131: "pwd" : "0110",  
 132: "fase" : "2",  
 133: "ladoA" : "",  
 134: "ladoB" : "",  
 135: "trecho" : "ENTR GO-116 (P/FORMOSA) - ENTR GO-346 (P/CABEÇEIRAS)"  
 136: }, {  
 137: "usr" : "020",  
 138: "pwd" : "0090",  
 139: "fase" : "2",  
 140: "ladoA" : "",  
 141: "ladoB" : "",  
 142: "trecho" : "ENTR BR-457(B)/GO-219 - ENTR GO-020(A)",  
 143: "perguntaExtra" : true  
 144: }, {  
 145: "usr" : "021",  
 146: "pwd" : "0112",  
 147: "fase" : "4",  
 148: "ladoA" : "",  
 149: "ladoB" : "",  
 150: "trecho" : "ENTR GO-338 (ABADIÂNIA - FIM TRAVESSIA URBANA) - ENTR BR-153(A) (P/ANÁPOLIS)"  
 151: }, {  
 152: "usr" : "022",  
 153: "pwd" : "0340",  
 154: "fase" : "3",  
 155: "ladoA" : "",  
 156: "ladoB" : "",  
 157: "trecho" : "ENTR GO-353(A) (LINDA VISTA) - ENTR BR-414(A)/GO-151/244/353(B) (PORANGATU)"  
 158: }, {  
 159: "usr" : "023",  
 160: "pwd" : "0770",  
 161: "fase" : "2",  
 162: "ladoA" : "",  
 163: "ladoB" : "",  
 164: "trecho" : "ENTR BR-154/452(A)/483 - ENTR BR-452(B) (DIV GO/MG) (ITUMBIARA)"  
 165: }, {  
 166: "usr" : "024",  
 167: "pwd" : "0510",  
 168: "fase" : "2",  
 169: "ladoA" : "",  
 170: "ladoB" : "",  
 171: "trecho" : "ENTR BR-060(B)/158(B)/GO-184(B) (P/ESTREITO) - ENTR GO-050(B)"  
 172: }, {  
 173: "usr" : "025",  
 174: "pwd" : "0050",  
 175: "fase" : "2",  
 176: "ladoA" : "",  
 177: "ladoB" : "",  
 178: "trecho" : "FIM DA DUPLICAÇÃO - ENTR BR-402/MA-110 (BACABEIRA)"  
 179: }, {  
 180: "usr" : "026",  
 181: "pwd" : "0150",  
 182: "fase" : "3",  
 183: "ladoA" : "",  
 184: "ladoB" : "",  
 185: "trecho" : "ENTR BR-316(A) (CACHUCHA) - ENTR BR-316(B) (PERITORÓ)"  
 186: }, {  
 187: "usr" : "027",  
 188: "pwd" : "0680",  
 189: "fase" : "3",  
 190: "ladoA" : "",  
 191: "ladoB" : "",  
 192: "trecho" : "RIO PINDARÉ - ENTR BR-010(A) (AÇAILÂNDIA)"  
 193: }, {  
 194: "usr" : "028",  
 195: "pwd" : "0670",  
 196: "fase" : "2",  
 197: "ladoA" : "",  
 198: "ladoB" : "",  
 199: "trecho" : "ENTR BR-251(A) (CANACÍ) - ENTR BR-135/251(B)/308/365 (MONTES CLAROS)"  
 200: }, {  
 201: "usr" : "029",  
 202: "pwd" : "0370",  
 203: "fase" : "3",  
 204: "ladoA" : "",  
 205: "ladoB" : "",  
 206: "trecho" : "ENTR MA-026 (DEZESSETE) - ENTR MA-034(A)/127/349 (CAXIAS)"  
 207: }, {  
 208: "usr" : "030",  
 209: "pwd" : "0130",  
 210: "fase" : "2",  
 211: "ladoA" : "",  
 212: "ladoB" : "",  
 213: "trecho" : "ENTR MG-410 (P/PONTO DIAMANTE) - ENTR MG-181 (JOÃO PINHEIRO)"  
 214: }, {  
 215: "usr" : "031",  
 216: "pwd" : "0330",  
 217: "fase" : "3",  
 218: "ladoA" : "",  
 219: "ladoB" : "",  
 220: "trecho" : "ENTR MG-238 (P/SETE LAGOAS) - ENTR MG-432 (P/ESMERALDAS)"  
 221: }, {  
 222: "usr" : "032",  
 223: "pwd" : "0040",  
 224: "fase" : "2",  
 225: "ladoA" : "",  
 226: "ladoB" : "",  
 227: "trecho" : "ENTR BR-040(A) (ANEL RODOVIÁRIO DE BELO HORIZONTE) - ENTR BR-040(B)"  
 228: }, {  
 229: "usr" : "033",  
 230: "pwd" : "0570",  
 231: "fase" : "3",  
 232: "ladoA" : "",  
 233: "ladoB" : "",  
 234: "trecho" : "ENTR BR-499 (SANTOS DUMONT) - ENTR ANT UNIÃO E INDÚSTRIA (B. TRIUNFO)"  
 235: }, {  
 236: "usr" : "034",  
 237: "pwd" : "0250",  
 238: "fase" : "4",  
 239: "ladoA" : "",  
 240: "ladoB" : "",  
 241: "trecho" : "ENTR BR-365(B)/452(B) - RIO TIJUCO"  
 242: }, {  
 243: "usr" : "035",  
 244: "pwd" : "1130",  
 245: "fase" : "2",  
 246: "ladoA" : "",  
 247: "ladoB" : "",  
 248: "trecho" : "ENTR BR-342(B)/418/MG-217 (RIB STO ANTÔNIO) (TEÓFILO OTONI) - ACESSO ITAMBACURI"  
 249: }, {  
 250: "usr" : "036",  
 251: "pwd" : "1280",  
 252: "fase" : "4",  
 253: "ladoA" : "",  
 254: "ladoB" : "",  
 255: "trecho" : "FIM PISTA DUPLA - DOM CORRÊA"  
 256: }, {  
 257: "usr" : "037",  
 258: "pwd" : "1430",  
 259: "fase" : "3",  
 260: "ladoA" : "",  
 261: "ladoB" : "",  
 262: "trecho" : "ENTR BR-116(A)/120 (LEOPOLDINA) - ENTR BR-116(B)"  
 263: }, {  
 264: "usr" : "038",  
 265: "pwd" : "0850",  
 266: "fase" : "3",  
 267: "ladoA" : "",  
 268: "ladoB" : "",  
 269: "trecho" : "ENTR BR-259(B) (CURVELO) - ENTR BR-040(A)"  
 270: }, {  
 271: "usr" : "039",  
 272: "pwd" : "0290",  
 273: "fase" : "2",  
 274: "ladoA" : "",  
 275: "ladoB" : "",  
 276: "trecho" : "ENTR BR-364(A)/262(A) (P/COMENDADOR GOMES) - ENTR BR-364(B)/262(B) (P/FRUTAL)"  
 277: }, {  
 278: "usr" : "040",  
 279: "pwd" : "0890",  
 280: "fase" : "4",  
 281: "ladoA" : "",  
 282: "ladoB" : "",  
 283: "trecho" : "ENTR MG-187 (P/IBIÁ) - ENTR BR-146 (P/ARAXÁ)"  
 284: }, {  
 285: "usr" : "041",  
 286: "pwd" : "0230",  
 287: "fase" : "3",  
 288: "ladoA" : "",  
 289: "ladoB" : "",  
 290: "trecho" : "ENTR MG-451 (ITUTINGA) - ENTR BR-354 (LAVRAS)"  
 291: }, {  
 292: "usr" : "042",  
 293: "pwd" : "0215",  
 294: "fase" : "4",  
 295: "ladoA" : "",  
 296: "ladoB" : "",  
 297: "trecho" : "ACESSO CARMO DO PARANAIBA - ENTR MG-230"  
 298: }, {  
 299: "usr" : "043",  
 300: "pwd" : "0100",  
 301: "fase" : "3",  
 302: "ladoA" : "",  
 303: "ladoB" : "",  
 304: "trecho" : "PRESIDENTE OLEGÁRIO - ENTR BR-146(A)/365 (PATOS DE MINAS)"  
 305: }, {  
 306: "usr" : "044",  
 307: "pwd" : "0370",  
 308: "fase" : "3",  
 309: "ladoA" : "",  
 310: "ladoB" : "",  
 311: "trecho" : "ENTR BR-153 - ENTR BR-154(A)"  
 312: }, {  
 313: "usr" : "045",  
 314: "pwd" : "0230",  
 315: "fase" : "2",  
 316: "ladoA" : "",  
 317: "ladoB" : "",  
 318: "trecho" : "ENTR MG-425 (CORONEL FABRICIANO) - ENTR MG-320 (P/JAGUARAÇU)"  
 319: }, {  
 320: "usr" : "046",  
 321: "pwd" : "0490",  
 322: "fase" : "2",  
 323: "ladoA" : "",  
 324: "ladoB" : "",  
 325: "trecho" : "ENTR BR-262(B) (BETIM) - ENTR MG-155"  
 326: }, {  
 327: "usr" : "047",  
 328: "pwd" : "0750",  
 329: "fase" : "3",  
 330: "ladoA" : "",  
 331: "ladoB" : "",  
 332: "trecho" : "ENTR MG-458 (CAREAÇU) - ENTR BR-459 (P/POUSO ALEGRE)"  
 333: }, {  
 334: "usr" : "048",  
 335: "pwd" : "0324",  
 336: "fase" : "1",  
 337: "ladoA" : "VILA SÃO PEDRO",  
 338: "ladoB" : "ACESSO COLÔNIA AGRÍCOLA SEXTA LINHA",  
 339: "trecho" : "ACESSO COLÂNIA AGRICOLA SEXTA LINHA - FINAL PISTA DUPLA (VILA VARGAS)"  
 340: }, {  
 341: "usr" : "049",  
 342: "pwd" : "0450",  
 343: "fase" : "1",  
 344: "ladoA" : "ENTR MS-441(B) (P/BANDEIRANTES)",  
 345: "ladoB" : "ENTR MS-244 (P/BONFIM)",  
 346: "trecho" : "ENTR MS-441(B) (P/BANDEIRANTES) - ENTR MS-244 (P/BONFIM)"  
 347: }, {  
 348: "usr" : "050",  
 349: "pwd" : "1290",  
 350: "fase" : "2",  
 351: "ladoA" : "",  
 352: "ladoB" : "",  
 353: "trecho" : "ENTR BR-158(B)/MS-395 (P/BRASILÂNDIA) - ENTR MS-453/459 (P/ARAPUÁ)"  
 354: }, {  
 355: "usr" : "051",  
 356: "pwd" : "0880",  
 357: "fase" : "2",  
 358: "ladoA" : "",  
 359: "ladoB" : "",  
 360: "trecho" : "FIM TRAV RIO PARANÁ (PONTE M. JOPPERT) - INCÍO DA PISTA DUPLA"  
 361: }, {  
 362: "usr" : "052",  
 363: "pwd" : "0030",  
 364: "fase" : "1",  
 365: "ladoA" : "ENTR BR-070(B)",  
 366: "ladoB" : "ENTR MT-170 (CARAMÚJO)",  
 367: "trecho" : "ENTR BR-070(B) - ENTR MT-170 (CARAMÚJO)"  
 368: }, {  
 369: "usr" : "053",  
 370: "pwd" : "0592",  
 371: "fase" : "1",  
 372: "ladoA" : "ENTR MT-270(B)",  
 373: "ladoB" : "ENTR MT-469(A)",  
 374: "trecho" : "ENTR MT-270(B) - ENTR MT-469(A)"  
 375: }, {  
 376: "usr" : "054",  
 377: "pwd" : "0840",  
 378: "fase" : "3",  
 379: "ladoA" : "",  
 380: "ladoB" : "",  
 381: "trecho" : "JANGADA - ENTR MT-246(B)"  
 382: }, {  
 383: "usr" : "055",  
 384: "pwd" : "0510",  
 385: "fase" : "3",  
 386: "ladoA" : "",  
 387: "ladoB" : "",  
 388: "trecho" : "ENTR BR-222(B)/PA-332 (DOM ELISEU) - ENTR PA-125/263 (GURUPIZINHO)"  
 389: }, {  
 390: "usr" : "056",  
 391: "pwd" : "0335",  
 392: "fase" : "3",  
 393: "ladoA" : "",  
 394: "ladoB" : "",  
 395: "trecho" : "ENTR PB-018 (P/CONDE) - ENTR PB-034"  
 396: }, {  
 397: "usr" : "057",  
 398: "pwd" : "0010",  
 399: "fase" : "4",  
 400: "ladoA" : "",  
 401: "ladoB" : "",  
 402: "trecho" : "ENTR PB-100 (P/GALANTE) - ENTR BR-104(A)/408(B)/PB-095 (CAMPINA GRANDE)"  
 403: }, {  
 404: "usr" : "058",  
 405: "pwd" : "0510",  
 406: "fase" : "3",  
 407: "ladoA" : "",  
 408: "ladoB" : "",  
 409: "trecho" : "ENTR PE-063 (P/AMARAJI) - ENTR PE-064/085 (RIBEIRÃO)"  
 410: }, {  
 411: "usr" : "059",  
 412: "pwd" : "0460",  
 413: "fase" : "3",  
 414: "ladoA" : "",  
 415: "ladoB" : "",  
 416: "trecho" : "ENTR PE-460 - ENTR BR-316/428 (P/CABROBÓ)"  
 417: }, {  
 418: "usr" : "060",  
 419: "pwd" : "0140",  
 420: "fase" : "3",  
 421: "ladoA" : "",  
 422: "ladoB" : "",  
 423: "trecho" : "ENTR PE-103 (P/BONITO) - ENTR BR-104/423(A) (CARUARÚ)"  
 424: }, {  
 425: "usr" : "061",  
 426: "pwd" : "0420",  
 427: "fase" : "4",  
 428: "ladoA" : "",  
 429: "ladoB" : "",  
 430: "trecho" : "ENTR PE-450 (P/VERDEJANTE) - ENTR BR-116/361 (SALGUEIRO)"  
 431: }, {  
 432: "usr" : "062",  
 433: "pwd" : "0370",  
 434: "fase" : "3",  
 435: "ladoA" : "",  
 436: "ladoB" : "",  
 437: "trecho" : "ENTR BR-428(A) (LAGOA GRANDE) - ENTR BR-235/407/423/428(B) (DIV PE/BA) (PETROLINA/JUAZEIRO)"  
 438: }, {  
 439: "usr" : "063",  
 440: "pwd" : "0830",  
 441: "fase" : "4",  
 442: "ladoA" : "",  
 443: "ladoB" : "",  
 444: "trecho" : "ENTR PI-242 - ENTR BR-407(A)/PI-238/245(A) (PICOS)"  
 445: }, {  
 446: "usr" : "064",  
 447: "pwd" : "0130",  
 448: "fase" : "3",  
 449: "ladoA" : "",  
 450: "ladoB" : "",  
 451: "trecho" : "ENTR BR-404(B)/407(B) - INÍCIO PISTA DUPLA (CAPITÃO CAMPOS)"  
 452: }, {  
 453: "usr" : "065",  
 454: "pwd" : "0420",  
 455: "fase" : "3",  
 456: "ladoA" : "",  
 457: "ladoB" : "",  
 458: "trecho" : "DEMERVAL LOBÃO - ENTR PI-223 (MONSENHOR GIL)"  
 459: }, {  
 460: "usr" : "066",  
 461: "pwd" : "2710",  
 462: "fase" : "1",  
 463: "ladoA" : "DIV SP/PR (CAB NORTE PONTE S/ RIO PARDINHO)",  
 464: "ladoB" : "INÍCIO VARIANTE DO ALPINO(PISTA DIREITA)",  
 465: "trecho" : "DIV SP/PR (CAB NORTE PONTE S/ RIO PARDINHO) - INÍCIO VARIANTE DO ALPINO(PISTA DIREITA)"  
 466: }, {  
 467: "usr" : "067",  
 468: "pwd" : "2810",  
 469: "fase" : "1",  
 470: "ladoA" : "ENTR PR-427 (CAMPO DO TENENTE)",  
 471: "ladoB" : "DIV PR/SC (RIO NEGRO/MAFRA)",  
 472: "trecho" : "ENTR PR-427 (CAMPO DO TENENTE) - DIV PR/SC (RIO NEGRO/MAFRA)"  
 473: }, {  
 474: "usr" : "068",  
 475: "pwd" : "0205",  
 476: "fase" : "1",  
 477: "ladoA" : "ACESSO GUARAPUAVA",  
 478: "ladoB" : "ENTR BR-466 (P/GUARAPUAVA)",  
 479: "trecho" : "ACESSO GUARAPUAVA - ENTR BR-466 (P/GUARAPUAVA)"  
 480: }, {  
 481: "usr" : "069",  
 482: "pwd" : "0310",  
 483: "fase" : "1",  
 484: "ladoA" : "ENTR PR-182 (P/CAPITÃO LEÔNIDAS MARQUES)",  
 485: "ladoB" : "ENTR BR-163",  
 486: "trecho" : "ACESSO OESTE CASCAVEL - ENTR BR-163/PR-182 (P/CAPITÃO LEÔNIDAS MARQUES)"  
 487: }, {  
 488: "usr" : "070",  
 489: "pwd" : "0480",  
 490: "fase" : "2",  
 491: "ladoA" : "",  
 492: "ladoB" : "",  
 493: "trecho" : "FIM PISTA DUPLA - ENTR PR-431 (P/JACAREZINHO)"  
 494: }, {  
 495: "usr" : "071",  
 496: "pwd" : "0490",  
 497: "fase" : "1",  
 498: "ladoA" : "ENTR PR-281 (P/TIJUCAS DO SUL)",  
 499: "ladoB" : "DIV PR/SC (ENTR BR-101)",  
 500: "trecho" : "ENTR PR-281 (P/TIJUCAS DO SUL) - DIV PR/SC (ENTR BR-101)"  
 501: }, {  
 502: "usr" : "072",  
 503: "pwd" : "0095",  
 504: "fase" : "1",  
 505: "ladoA" : "ENTR PR-510/511 (P/CONTENDA)",  
 506: "ladoB" : "ENTR PR-428 (LAPA)",  
 507: "trecho" : "ENTR PR-510/511 (P/CONTENDA) - ENTR PR-428 (LAPA)"  
 508: }, {  
 509: "usr" : "073",  
 510: "pwd" : "0730",  
 511: "fase" : "2",  
 512: "ladoA" : "",  
 513: "ladoB" : "",  
 514: "trecho" : "ENTR BR-393(B) (TREVO MOURA BRASIL) - ENTR BR-492(A) (TREVO P/ AREAL)"  
 515: }, {  
 516: "usr" : "074",  
 517: "pwd" : "2930",  
 518: "fase" : "3",  
 519: "ladoA" : "",  
 520: "ladoB" : "",  
 521: "trecho" : "ENTR RJ-162 (P/RIO DOURADO) - ENTR BR-120 (CASIMIRO DE ABREU)"  
 522: }, {  
 523: "usr" : "075",  
 524: "pwd" : "2130",  
 525: "fase" : "2",  
 526: "ladoA" : "",  
 527: "ladoB" : "",  
 528: "trecho" : "ENTR RJ-159 (FLORIANO) - ENTR RJ-161 (RESENDE)"  
 529: }, {  
 530: "usr" : "076",  
 531: "pwd" : "0310",  
 532: "fase" : "2",  
 533: "ladoA" : "",  
 534: "ladoB" : "",  
 535: "trecho" : "ENTR BR-356(A)/393(B) - ENTR BR-356(B) (ITAPERUNA)"  
 536: }, {  
 537: "usr" : "077",  
 538: "pwd" : "0330",  
 539: "fase" : "3",  
 540: "ladoA" : "",  
 541: "ladoB" : "",  
 542: "trecho" : "SAPUCAIA - ENTR BR-040(A)"  
 543: }, {  
 544: "usr" : "078",  
 545: "pwd" : "0167",  
 546: "fase" : "3",  
 547: "ladoA" : "",  
 548: "ladoB" : "",  
 549: "trecho" : "ACESSO À LAGOA DO BONFIM - ENTR RN-315/002(A) (SÃO JOSÉ DE MIPIBÚ/NÍSIA FLORESTA)"  
 550: }, {  
 551: "usr" : "079",  
 552: "pwd" : "0070",  
 553: "fase" : "3",  
 554: "ladoA" : "",  
 555: "ladoB" : "",  
 556: "trecho" : "ENTR RN-203 (P/SÃO PEDRO) - ACESSO BOM JESUS"  
 557: }, {  
 558: "usr" : "080",  
 559: "pwd" : "0270",  
 560: "fase" : "3",  
 561: "ladoA" : "",  
 562: "ladoB" : "",  
 563: "trecho" : "ENTR BR-104(B) (LAJES) - ENTR RN-023 (CAIÇARA DO RIO DO VENTO)"  
 564: }, {  
 565: "usr" : "081",  
 566: "pwd" : "0070",  
 567: "fase" : "3",  
 568: "ladoA" : "",  
 569: "ladoB" : "",  
 570: "trecho" : "ENTR RN-233 (P/CARAÚBAS) - ENTR RN-031/177(A)"  
 571: }, {  
 572: "usr" : "082",  
 573: "pwd" : "1210",  
 574: "fase" : "3",  
 575: "ladoA" : "",  
 576: "ladoB" : "",  
 577: "trecho" : "ENTR BR-429(B) (JI-PARANÁ) - ENTR RO-473 (P/URUPÁ)"  
 578: }, {  
 579: "usr" : "083",  
 580: "pwd" : "3275",  
 581: "fase" : "3",  
 582: "ladoA" : "",  
 583: "ladoB" : "",  
 584: "trecho" : "ENTR ERS-703 (GUAIBA) - ENTR ERS-709 (P/ BARRA DO RIBEIRO)"  
 585: }, {  
 586: "usr" : "084",  
 587: "pwd" : "1150",  
 588: "fase" : "1",  
 589: "ladoA" : "ENTR BR-472 (P/PALMITINHO)",  
 590: "ladoB" : "ENTR RS-585/587 (SEBERI)",  
 591: "trecho" : "ENTR RSC-472 (P/ PALMITINHO) - ENTR ERS-585/587 (SEBERI)"  
 592: }, {  
 593: "usr" : "085",  
 594: "pwd" : "1310",  
 595: "fase" : "1",  
 596: "ladoA" : "ENTR RS-348 (VAL DE SERRA)",  
 597: "ladoB" : "ENTR RS-509 (P/SANTA MARIA)",  
 598: "trecho" : "ENTR ERS-348 (VAL DE SERRA) - ENTR ERS-509 (P/ SANTA MARIA)"  
 599: }, {  
 600: "usr" : "086",  
 601: "pwd" : "0180",  
 602: "fase" : "1",  
 603: "ladoA" : "ACESSO A MINAS DO LEÃO",  
 604: "ladoB" : "ENTR BR-471 (PANTANO GRANDE)",  
 605: "trecho" : "ACESSO A MINAS DO LEAO - ENTR BRS-471 (PANTANO GRANDE)"  
 606: }, {  
 607: "usr" : "087",  
 608: "pwd" : "1840",  
 609: "fase" : "1",  
 610: "ladoA" : "ENTR RS-705 (P/GERIBA)",  
 611: "ladoB" : "ENTR BR-290(B)",  
 612: "trecho" : "ENTR ERS-705 (GERIBA) - ENTR BRS-153(B) (P/ BAGE)"  
 613: }, {  
 614: "usr" : "088",  
 615: "pwd" : "0410",  
 616: "fase" : "1",  
 617: "ladoA" : "ENTR BR-293(A)",  
 618: "ladoB" : "ENTR BR-472(A)",  
 619: "trecho" : "ENTR BRS-293(A) (P/ QUARAI) - ENTR BRS-472(A) (P/ BARRA DO QUARAI)"  
 620: }, {  
 621: "usr" : "089",  
 622: "pwd" : "0130",  
 623: "fase" : "1",  
 624: "ladoA" : "ENTR BR-473 (P/BAGÉ)",  
 625: "ladoB" : "ENTR RS-630 (P/DOM PEDRITO)",  
 626: "trecho" : "ENTR RSC-473 (P/ BAGE) - ENTR ERS-630/634 (P/ DOM PEDRITO)"  
 627: }, {  
 628: "usr" : "090",  
 629: "pwd" : "0070",  
 630: "fase" : "1",  
 631: "ladoA" : "ENTR BR-471(A) (QUINTA)",  
 632: "ladoB" : "ACESSO PELOTAS",  
 633: "trecho" : "ENTR BRS-471(A) (QUINTA) - ACESSO SUL A PELOTAS"  
 634: }, {  
 635: "usr" : "091",  
 636: "pwd" : "0210",  
 637: "fase" : "1",  
 638: "ladoA" : "ACESSO LESTE A ITAQUI",  
 639: "ladoB" : "ENTR BR-290(A)/293(A) (URUGUAIANA)",  
 640: "trecho" : "ACESSO LESTE A ITAQUI - ENTR BR-290(A)/293(A) (URUGUAIANA)"  
 641: }, {  
 642: "usr" : "092",  
 643: "pwd" : "3830",  
 644: "fase" : "1",  
 645: "ladoA" : "GARUVA",  
 646: "ladoB" : "ENTR SC-430 (PIRABEIRABA)",  
 647: "trecho" : "GARUVA - ENTR SC-430 (PIRABEIRABA)"  
 648: }, {  
 649: "usr" : "093",  
 650: "pwd" : "4010",  
 651: "fase" : "1",  
 652: "ladoA" : "BALNEÁRIO DE CAMBORIÚ",  
 653: "ladoB" : "P/PORTO BELO",  
 654: "trecho" : "BALNEÁRIO DE CAMBORIÚ - P/PORTO BELO"  
 655: }, {  
 656: "usr" : "094",  
 657: "pwd" : "4125",  
 658: "fase" : "1",  
 659: "ladoA" : "PONTE SOBRE O RIO DA MADRE",  
 660: "ladoB" : "PONTE SOBRE O RIO ARAÇATUBA",  
 661: "trecho" : "PONTE SOBRE O RIO DA MADRE - PONTE SOBRE O RIO ARAÇATUBA"  
 662: }, {  
 663: "usr" : "095",  
 664: "pwd" : "4240",  
 665: "fase" : "1",  
 666: "ladoA" : "ENTR BR-285(A) (ARARANGUÁ)",  
 667: "ladoB" : "ENTR BR-285(B)/SC-285 (SANGA DA TOCA)",  
 668: "trecho" : "ENTR BR-285(A)/SC-449 (ARARANGUÁ) - ENTR BR-285(B)/SC-285 (P/ERMO)"  
 669: }, {  
 670: "usr" : "096",  
 671: "pwd" : "2910",  
 672: "fase" : "2",  
 673: "ladoA" : "",  
 674: "ladoB" : "",  
 675: "trecho" : "ENTR SC-352(B) (P/TAIÓ) - SÃO CRISTOVÃO DO SUL"  
 676: }, {  
 677: "usr" : "097",  
 678: "pwd" : "1590",  
 679: "fase" : "3",  
 680: "ladoA" : "",  
 681: "ladoB" : "",  
 682: "trecho" : "ENTR SC-463 (P/JABORÁ) - ENTR BR-283 (P/CONCÓRDIA)"  
 683: }, {  
 684: "usr" : "098",  
 685: "pwd" : "0020",  
 686: "fase" : "3",  
 687: "ladoA" : "",  
 688: "ladoB" : "",  
 689: "trecho" : "ENTR SC-114(A) (ÍNDIOS) - ENTR SC-114(B) (P/PAINEL)"  
 690: }, {  
 691: "usr" : "099",  
 692: "pwd" : "0377",  
 693: "fase" : "1",  
 694: "ladoA" : "ENTR SC-157(B) (P/MODELO)",  
 695: "ladoB" : "ENTR BR-158(A) (P/CUNHA PORÃ)",  
 696: "trecho" : "ENTR SC-157(B) (P/MODELO) - ENTR BR-158(A) (P/CUNHA PORÃ)"  
 697: }, {  
 698: "usr" : "100",  
 699: "pwd" : "0190",  
 700: "fase" : "4",  
 701: "ladoA" : "",  
 702: "ladoB" : "",  
 703: "trecho" : "ENTR SC-114(B) (P/OTACÍLIO COSTA) - ENTR BR-116"  
 704: }, {  
 705: "usr" : "101",  
 706: "pwd" : "0040",  
 707: "fase" : "3",  
 708: "ladoA" : "",  
 709: "ladoB" : "",  
 710: "trecho" : "ENTR SE-464 (P/SÃO CRISTOVÃO) - ENTR BR-349(B)/SE-265 (ITAPORANGA D`AJUDA)"  
 711: }, {  
 712: "usr" : "102",  
 713: "pwd" : "0060",  
 714: "fase" : "3",  
 715: "ladoA" : "",  
 716: "ladoB" : "",  
 717: "trecho" : "ENTR SE-160 - ENTR SE-170(A)"  
 718: }, {  
 719: "usr" : "103",  
 720: "pwd" : "2410",  
 721: "fase" : "3",  
 722: "ladoA" : "",  
 723: "ladoB" : "",  
 724: "trecho" : "ENTR SP-070 - ENTR SP-103 (CAÇAPAVA)"  
 725: }, {  
 726: "usr" : "104",  
 727: "pwd" : "2570",  
 728: "fase" : "3",  
 729: "ladoA" : "",  
 730: "ladoB" : "",  
 731: "trecho" : "ENTR SP-228 - ENTR SP-057 (P/SIDERÚRGICA)"  
 732: }, {  
 733: "usr" : "105",  
 734: "pwd" : "1043",  
 735: "fase" : "3",  
 736: "ladoA" : "",  
 737: "ladoB" : "",  
 738: "trecho" : "ACESSO GUAIÇARA - ENTR BR-154/267 (LINS)"  
 739: }, {  
 740: "usr" : "106",  
 741: "pwd" : "0810",  
 742: "fase" : "3",  
 743: "ladoA" : "",  
 744: "ladoB" : "",  
 745: "trecho" : "DIV MG/SP - ENTR SP-063 (P/BRAGANÇA PAULISTA)"  
 746: }, {  
 747: "usr" : "107",  
 748: "pwd" : "0105",  
 749: "fase" : "2",  
 750: "ladoA" : "",  
 751: "ladoB" : "",  
 752: "trecho" : "ENTR TO-420 - INICIO PISTA DUPLA ARAGUAINA"  
 753: }, {  
 754: "usr" : "108",  
 755: "pwd" : "0205",  
 756: "fase" : "3",  
 757: "ladoA" : "",  
 758: "ladoB" : "",  
 759: "trecho" : "ENTR TO-342(B) (MIRANORTE) - ENTR TO-348 (BARROLÂNDIA)"  
 760: }, {  
 761: "usr" : "109",  
 762: "pwd" : "0300",  
 763: "fase" : "3",  
 764: "ladoA" : "",  
 765: "ladoB" : "",  
 766: "trecho" : "ENTR TO-483 (FIGUEIRÓPOLIS) - ENTR TO-296(A)/373 (ALVORADA)"  
 767: }, {  
 768: "usr" : "110",  
 769: "pwd" : "0495",  
 770: "fase" : "3",  
 771: "ladoA" : "",  
 772: "ladoB" : "",  
 773: "trecho" : "ENTR PI-260 (P/BARREIRAS DO PIAUÍ) - ENTR PI-255 (CORRENTE)"  
 774: }, {  
 775: "usr" : "111",  
 776: "pwd" : "0150",  
 777: "fase" : "3",  
 778: "ladoA" : "",  
 779: "ladoB" : "",  
 780: "trecho" : "FIM PISTA DUPLA - ENTR GO-156(B)"  
 781: }, {  
 782: "usr" : "112",  
 783: "pwd" : "0025",  
 784: "fase" : "1",  
 785: "ladoA" : "ENTR PR-508 (P/MATINHOS)",  
 786: "ladoB" : "ENTR BR-101/PR-408 (P/MORRETES)",  
 787: "trecho" : "ENTR PR-508 (P/MATINHOS) - ENTR BR-101/PR-408 (P/MORRETES)"  
 788: }, {  
 789: "usr" : "113",  
 790: "pwd" : "0075",  
 791: "fase" : "1",  
 792: "ladoA" : "ACESSO SANTA",  
 793: "ladoB" : "ENTR BR-376(B)/PR-428 (SÃO LUIS PURUNÃ)",  
 794: "trecho" : "ACESSO SANTA - ENTR BR-376(B)/PR-428 (SÃO LUIS PURUNÃ)"  
 795: }, {  
 796: "usr" : "114",  
 797: "pwd" : "0950",  
 798: "fase" : "1",  
 799: "ladoA" : "ENTR MS-375 (ZUZU)",  
 800: "ladoB" : "ENTR BR-163(A) (NOVA ALVORADA DO SUL)",  
 801: "trecho" : "ENTR MS-375 (ZUZU) - ENTR BR-163(A) (NOVA ALVORADA)"  
 802: }, {  
 803: "usr" : "115",  
 804: "pwd" : "0440",  
 805: "fase" : "1",  
 806: "ladoA" : "ENTR MS-434 (RAIMUNDO)",  
 807: "ladoB" : "ENTR BR-483/497 (INÍCIO DA PISTA DUPLA)",  
 808: "trecho" : "ENTR MS-434 (RAIMUNDO) - ENTR BR-483/497 (INÍCIO DA PISTA DUPLA)"  
 809: }, {  
 810: "usr" : "116",  
 811: "pwd" : "0810",  
 812: "fase" : "3",  
 813: "ladoA" : "",  
 814: "ladoB" : "",  
 815: "trecho" : "ENTR PISTA INVERSA (B) - ENTR BR-101"  
 816: }, {  
 817: "usr" : "117",  
 818: "pwd" : "0070",  
 819: "fase" : "3",  
 820: "ladoA" : "",  
 821: "ladoB" : "",  
 822: "trecho" : "ENTR BR-493(A) (P/MAGÉ) - ENTR RJ-107 (IMBARIÊ)"  
 823: }, {  
 824: "usr" : "118",  
 825: "pwd" : "0040",  
 826: "fase" : "1",  
 827: "ladoA" : "ENTR BR-101",  
 828: "ladoB" : "ENTR SC-108(A) (NEUDOR)",  
 829: "trecho" : "ENTR BR-101 - ENTR SC-108(A) (NEUDOR)"  
 830: }, {  
 831: "usr" : "119",  
 832: "pwd" : "1490",  
 833: "fase" : "1",  
 834: "ladoA" : "ENTR BR-280(A)/PR-446",  
 835: "ladoB" : "ENTR PR-170",  
 836: "trecho" : "ENTR BR-153(A)/PR-446 - ENTR PR-170"  
 837: }, {  
 838: "usr" : "120",  
 839: "pwd" : "0060",  
 840: "fase" : "2",  
 841: "ladoA" : "",  
 842: "ladoB" : "",  
 843: "trecho" : "ENTR BR-262(B) - ENTR BR-050(B)"  
 844: }, {  
 845: "usr" : "121",  
 846: "pwd" : "0834",  
 847: "fase" : "1",  
 848: "ladoA" : "ENTR MT-423",  
 849: "ladoB" : "ITAUBA",  
 850: "trecho" : "ENTR MT-423 - ITAUBA"  
 851: }, {  
 852: "usr" : "122",  
 853: "pwd" : "0852",  
 854: "fase" : "3",  
 855: "ladoA" : "",  
 856: "ladoB" : "",  
 857: "trecho" : "MATUPÁ - GUARANTÃ DO NORTE"  
 858: }, {  
 859: "usr" : "123",  
 860: "pwd" : "0790",  
 861: "fase" : "3",  
 862: "ladoA" : "",  
 863: "ladoB" : "",  
 864: "trecho" : "ENTR AL-215(B) - ENTR AL-220(A) (P/BARRA DE SÃO MIGUEL)"  
 865: }, {  
 866: "usr" : "124",  
 867: "pwd" : "0260",  
 868: "fase" : "4",  
 869: "ladoA" : "",  
 870: "ladoB" : "",  
 871: "trecho" : "ENTR PE-270 (P/BUÍQUE) - ENTR BR-110 (CRUZEIRO DO NORDESTE)"  
 872: }, {  
 873: "usr" : "125",  
 874: "pwd" : "0090",  
 875: "fase" : "4",  
 876: "ladoA" : "",  
 877: "ladoB" : "",  
 878: "trecho" : "ENTR PE-218 (P/BOM CONSELHO) - ENTR PE-187 (P/PALMEIRINA)"  
 879: }, {  
 880: "usr" : "126",  
 881: "pwd" : "1695",  
 882: "fase" : "3",  
 883: "ladoA" : "",  
 884: "ladoB" : "",  
 885: "trecho" : "ENTR BR-420(B) (P/LAJE) - ENTR BA-542 (P/GUERÉM)"  
 886: }, {  
 887: "usr" : "127",  
 888: "pwd" : "0790",  
 889: "fase" : "4",  
 890: "ladoA" : "",  
 891: "ladoB" : "",  
 892: "trecho" : "ENTR BA-493 (ITATIM) - ENTR BA-245"  
 893: }, {  
 894: "usr" : "128",  
 895: "pwd" : "0334",  
 896: "fase" : "2",  
 897: "ladoA" : "",  
 898: "ladoB" : "",  
 899: "trecho" : "MAIRI - ENTR BR-349"  
 900: }, {  
 901: "usr" : "129",  
 902: "pwd" : "0270",  
 903: "fase" : "2",  
 904: "ladoA" : "",  
 905: "ladoB" : "",  
 906: "trecho" : "GAVIÃO - ENTR BR-349 (NOVA FÁTIMA)"  
 907: }, {  
 908: "usr" : "130",  
 909: "pwd" : "1010",  
 910: "fase" : "2",  
 911: "ladoA" : "",  
 912: "ladoB" : "",  
 913: "trecho" : "DIV BA/MG - ENTR BR-251(A) (P/SALINAS)"  
 914: }, {  
 915: "usr" : "131",  
 916: "pwd" : "0530",  
 917: "fase" : "2",  
 918: "ladoA" : "",  
 919: "ladoB" : "",  
 920: "trecho" : "ENTR BA-937 (P/PAJEÚ DO VENTO) - ENTR BR-122(B)/430/BA-569 (CAETITÉ)"  
 921: }, {  
 922: "usr" : "132",  
 923: "pwd" : "0245",  
 924: "fase" : "3",  
 925: "ladoA" : "",  
 926: "ladoB" : "",  
 927: "trecho" : "ENTR BA-131 (P/CAÉM) - ENTR BA-417"  
 928: }, {  
 929: "usr" : "133",  
 930: "pwd" : "0550",  
 931: "fase" : "2",  
 932: "ladoA" : "",  
 933: "ladoB" : "",  
 934: "trecho" : "ENTR BR-235 - ENTR BA-220 (EUCLIDES DA CUNHA)"  
 935: }, {  
 936: "usr" : "134",  
 937: "pwd" : "0690",  
 938: "fase" : "3",  
 939: "ladoA" : "",  
 940: "ladoB" : "",  
 941: "trecho" : "ENTR CE-373 (CARMELÓPOLIS) - ENTR CE-187/292 (CAMPOS SALES)"  
 942: }, {  
 943: "usr" : "135",  
 944: "pwd" : "0190",  
 945: "fase" : "2",  
 946: "ladoA" : "",  
 947: "ladoB" : "",  
 948: "trecho" : "ENTR CE-266 (SUCESSO) - ENTR BR-226/404 (CRATEÚS)"  
 949: }, {  
 950: "usr" : "136",  
 951: "pwd" : "0610",  
 952: "fase" : "4",  
 953: "ladoA" : "",  
 954: "ladoB" : "",  
 955: "trecho" : "ENTR CE-060(B) (MINEIROLÂNDIA) - ENTR CE-168 (PEDRA BRANCA)"  
 956: }, {  
 957: "usr" : "137",  
 958: "pwd" : "0050",  
 959: "fase" : "4",  
 960: "ladoA" : "",  
 961: "ladoB" : "",  
 962: "trecho" : "ENTR CE-154 (PIRANGI) - ENTR CE-060(A)/265 (QUIXADÁ)"  
 963: }, {  
 964: "usr" : "138",  
 965: "pwd" : "0590",  
 966: "fase" : "2",  
 967: "ladoA" : "",  
 968: "ladoB" : "",  
 969: "trecho" : "ENTR CE-456 (P/TARGINOS) - ENTR CE-257 (CANINDÉ)"  
 970: }, {  
 971: "usr" : "139",  
 972: "pwd" : "0070",  
 973: "fase" : "4",  
 974: "ladoA" : "",  
 975: "ladoB" : "",  
 976: "trecho" : "ENTR CE-341 (CROATÁ) - ENTR CE-162 (SÃO LUÍS DO CURU)"  
 977: }, {  
 978: "usr" : "140",  
 979: "pwd" : "0160",  
 980: "fase" : "3",  
 981: "ladoA" : "",  
 982: "ladoB" : "",  
 983: "trecho" : "ENTR CE-263 (P/JAGUARUANA) - RUSSAS"  
 984: }, {  
 985: "usr" : "141",  
 986: "pwd" : "2130",  
 987: "fase" : "2",  
 988: "ladoA" : "",  
 989: "ladoB" : "",  
 990: "trecho" : "ENTR BR-381 (SÃO MATEUS) - ENTR ES-430 (P/JAGUARÉ)"  
 991: }, {  
 992: "usr" : "142",  
 993: "pwd" : "0300",  
 994: "fase" : "4",  
 995: "ladoA" : "",  
 996: "ladoB" : "",  
 997: "trecho" : "ENTR ES-137 - ENTR ES-130 (P/NOVA VENÉCIA)"  
 998: }, {  
 999: "usr" : "143",  
1000: "pwd" : "1990",  
1001: "fase" : "3",  
1002: "ladoA" : "",  
1003: "ladoB" : "",  
1004: "trecho" : "ENTR BA-290 (TEIXEIRA DE FREITAS) - ENTR BR-418 (P/POSTO DA MATA)"  
1005: }, {  
1006: "usr" : "144",  
1007: "pwd" : "0030",  
1008: "fase" : "2",  
1009: "ladoA" : "",  
1010: "ladoB" : "",  
1011: "trecho" : "MORRO GRANDE (ACESSO IV C. ITAPEMIRIM) - ENTR ES-166 (COUTINHO)"  
1012: }, {  
1013: "usr" : "145",  
1014: "pwd" : "0130",  
1015: "fase" : "2",  
1016: "ladoA" : "",  
1017: "ladoB" : "",  
1018: "trecho" : "ENTR MG-422 (RESPLENDOR) - ENTR BR-458 (CONSELHEIRO PENA)"  
1019: }, {  
1020: "usr" : "146",  
1021: "pwd" : "0125",  
1022: "fase" : "3",  
1023: "ladoA" : "",  
1024: "ladoB" : "",  
1025: "trecho" : "ENTR GO-230 - ENTR GO-237"  
1026: }, {  
1027: "usr" : "147",  
1028: "pwd" : "0160",  
1029: "fase" : "2",  
1030: "ladoA" : "",  
1031: "ladoB" : "",  
1032: "trecho" : "ENTR GO-114 (P/FLORES DE GOIÁS) - SANTA MARIA"  
1033: }, {  
1034: "usr" : "148",  
1035: "pwd" : "0470",  
1036: "fase" : "3",  
1037: "ladoA" : "",  
1038: "ladoB" : "",  
1039: "trecho" : "DIV MG/GO - ENTR GO-010(A)"  
1040: }, {  
1041: "usr" : "149",  
1042: "pwd" : "0050",  
1043: "fase" : "4",  
1044: "ladoA" : "",  
1045: "ladoB" : "",  
1046: "trecho" : "ENTR GO-010 (P/LUZIÂNIA) - ENTR BR-050(B)/354/457/GO-309 (CRISTALINA)"  
1047: }, {  
1048: "usr" : "150",  
1049: "pwd" : "0080",  
1050: "fase" : "3",  
1051: "ladoA" : "",  
1052: "ladoB" : "",  
1053: "trecho" : "URUTAÍ - ENTR BR-490/GO-213/307 (IPAMERI)"  
1054: }, {  
1055: "usr" : "151",  
1056: "pwd" : "0550",  
1057: "fase" : "2",  
1058: "ladoA" : "",  
1059: "ladoB" : "",  
1060: "trecho" : "ENTR GO-080(B) (P/SAO FRANCISCO DE GOIÁS) - ENTR GO-431 (P/PIRENOPOLIS)"  
1061: }, {  
1062: "usr" : "152",  
1063: "pwd" : "0190",  
1064: "fase" : "2",  
1065: "ladoA" : "",  
1066: "ladoB" : "",  
1067: "trecho" : "ENTR BR-153(A)/GO-342(B) - ENTR BR-153(B)/GO-237 (URUAÇÚ)"  
1068: }, {  
1069: "usr" : "153",  
1070: "pwd" : "0750",  
1071: "fase" : "2",  
1072: "ladoA" : "",  
1073: "ladoB" : "",  
1074: "trecho" : "ENTR GO-210(B) (P/PANAMÁ) - ENTR BR-154/452(A)/483"  
1075: }, {  
1076: "usr" : "154",  
1077: "pwd" : "0050",  
1078: "fase" : "2",  
1079: "ladoA" : "",  
1080: "ladoB" : "",  
1081: "trecho" : "ENTR GO-040 - ENTR BR-154(A)/483(A)/GO-206(A) (P/CACHOEIRA DOURADA)"  
1082: }, {  
1083: "usr" : "155",  
1084: "pwd" : "0570",  
1085: "fase" : "3",  
1086: "ladoA" : "",  
1087: "ladoB" : "",  
1088: "trecho" : "ENTR GO-194 (P/PORTELÂNDIA) - INÍCIO PISTA DUPLA"  
1089: }, {  
1090: "usr" : "156",  
1091: "pwd" : "0440",  
1092: "fase" : "3",  
1093: "ladoA" : "",  
1094: "ladoB" : "",  
1095: "trecho" : "ENTR GO-174(B) - ENTR GO-206 (P/CAÇU)"  
1096: }, {  
1097: "usr" : "157",  
1098: "pwd" : "0280",  
1099: "fase" : "2",  
1100: "ladoA" : "",  
1101: "ladoB" : "",  
1102: "trecho" : "ENTR MA-006(A) (COCALINHO) - ENTR MA-216 (BOM JARDIM)"  
1103: }, {  
1104: "usr" : "158",  
1105: "pwd" : "0860",  
1106: "fase" : "3",  
1107: "ladoA" : "",  
1108: "ladoB" : "",  
1109: "trecho" : "ENTR MA-259 (P/TUNTUM) - ENTR MA-012/272 (BARRA DO CORDA)"  
1110: }, {  
1111: "usr" : "159",  
1112: "pwd" : "0300",  
1113: "fase" : "2",  
1114: "ladoA" : "",  
1115: "ladoB" : "",  
1116: "trecho" : "ENTR BR-222(B) (ESTACA ZERO) - ENTR MA-240 (PIO XII)"  
1117: }, {  
1118: "usr" : "160",  
1119: "pwd" : "0730",  
1120: "fase" : "3",  
1121: "ladoA" : "",  
1122: "ladoB" : "",  
1123: "trecho" : "BOM JESUS DO TOCANTINS - RIO JACUNDÁ (R MÃE MARIA)"  
1124: }, {  
1125: "usr" : "161",  
1126: "pwd" : "0750",  
1127: "fase" : "3",  
1128: "ladoA" : "",  
1129: "ladoB" : "",  
1130: "trecho" : "ENTR BR-122/251/365 (MONTES CLAROS) - ENTR BR-451 (BOCAIÚVA)"  
1131: }, {  
1132: "usr" : "162",  
1133: "pwd" : "0240",  
1134: "fase" : "4",  
1135: "ladoA" : "",  
1136: "ladoB" : "",  
1137: "trecho" : "ENTR BR-342/MG-404 (SALINAS) - RIO VACARIA"  
1138: }, {  
1139: "usr" : "163",  
1140: "pwd" : "0290",  
1141: "fase" : "2",  
1142: "ladoA" : "",  
1143: "ladoB" : "",  
1144: "trecho" : "ENTR MA-132/270 (COLINAS) - ENTR MA-134 (PEIXE)"  
1145: }, {  
1146: "usr" : "164",  
1147: "pwd" : "0030",  
1148: "fase" : "3",  
1149: "ladoA" : "",  
1150: "ladoB" : "",  
1151: "trecho" : "ENTR BR-040(B) - ENTR MG-155 (SÃO BRÁS DO SUAÇUÍ)"  
1152: }, {  
1153: "usr" : "165",  
1154: "pwd" : "0335",  
1155: "fase" : "2",  
1156: "ladoA" : "",  
1157: "ladoB" : "",  
1158: "trecho" : "ENTR MG-434 - ENTR MG-435 (P/CAETÉ)"  
1159: }, {  
1160: "usr" : "166",  
1161: "pwd" : "0510",  
1162: "fase" : "2",  
1163: "ladoA" : "",  
1164: "ladoB" : "",  
1165: "trecho" : "ENTR MG-275 (P/CARANDAÍ) - ACESSO ALTO DOCE (INÍCIO PISTA DUPLA)"  
1166: }, {  
1167: "usr" : "167",  
1168: "pwd" : "0130",  
1169: "fase" : "3",  
1170: "ladoA" : "",  
1171: "ladoB" : "",  
1172: "trecho" : "ENTR BR-040(B) - ENTR MG-135"  
1173: }, {  
1174: "usr" : "168",  
1175: "pwd" : "0250",  
1176: "fase" : "3",  
1177: "ladoA" : "",  
1178: "ladoB" : "",  
1179: "trecho" : "ENTR MG-190 (RIO BAGAGEM) - ACESSO INDIANÓPOLIS"  
1180: }, {  
1181: "usr" : "169",  
1182: "pwd" : "0152",  
1183: "fase" : "3",  
1184: "ladoA" : "",  
1185: "ladoB" : "",  
1186: "trecho" : "ENTR BR-352(B)/GO-210(B)/330 (CATALÃO) - ENTR GO-402"  
1187: }, {  
1188: "usr" : "170",  
1189: "pwd" : "0010",  
1190: "fase" : "3",  
1191: "ladoA" : "",  
1192: "ladoB" : "",  
1193: "trecho" : "ENTR BR-050/365/452/455 (UBERLÂNDIA) - ENTR BR-153/464 (PRATA)"  
1194: }, {  
1195: "usr" : "171",  
1196: "pwd" : "0310",  
1197: "fase" : "4",  
1198: "ladoA" : "",  
1199: "ladoB" : "",  
1200: "trecho" : "ENTR BR-050(B)/455/497 (UBERLÂNDIA) - ENTR BR-452(B)"  
1201: }, {  
1202: "usr" : "172",  
1203: "pwd" : "0230",  
1204: "fase" : "4",  
1205: "ladoA" : "",  
1206: "ladoB" : "",  
1207: "trecho" : "ENTR MG-190 (P/NOVA PONTE) - ACESSO SANTA JULIANA"  
1208: }, {  
1209: "usr" : "173",  
1210: "pwd" : "0110",  
1211: "fase" : "3",  
1212: "ladoA" : "",  
1213: "ladoB" : "",  
1214: "trecho" : "ACESSO CARLOS CHAGAS - ENTR MG-412"  
1215: }, {  
1216: "usr" : "174",  
1217: "pwd" : "0110",  
1218: "fase" : "2",  
1219: "ladoA" : "",  
1220: "ladoB" : "",  
1221: "trecho" : "ACESSO SÃO JOSÉ DO JACURÍ - ENTR MG-117 (SÃO JOÃO EVANGELISTA)"  
1222: }, {  
1223: "usr" : "175",  
1224: "pwd" : "1310",  
1225: "fase" : "4",  
1226: "ladoA" : "",  
1227: "ladoB" : "",  
1228: "trecho" : "TREVO DE ACESSO SÃO JOÃO DO MANHUAÇU - ENTR MG-265 (P/DIVINO)"  
1229: }, {  
1230: "usr" : "176",  
1231: "pwd" : "0210",  
1232: "fase" : "2",  
1233: "ladoA" : "",  
1234: "ladoB" : "",  
1235: "trecho" : "ENTR ES-185(B) (P/IÚNA) - DIV ES/MG"  
1236: }, {  
1237: "usr" : "177",  
1238: "pwd" : "1195",  
1239: "fase" : "3",  
1240: "ladoA" : "",  
1241: "ladoB" : "",  
1242: "trecho" : "ACESSO ITANHOMI - ENTR BR-458(A) (TURUAÇÚ)"  
1243: }, {  
1244: "usr" : "178",  
1245: "pwd" : "0030",  
1246: "fase" : "2",  
1247: "ladoA" : "",  
1248: "ladoB" : "",  
1249: "trecho" : "ENTR BR-116(B) - ENTR MG-126 (BICAS)"  
1250: }, {  
1251: "usr" : "179",  
1252: "pwd" : "1370",  
1253: "fase" : "2",  
1254: "ladoA" : "",  
1255: "ladoB" : "",  
1256: "trecho" : "ENTR BR-265/356 (MURIAÉ) - ENTR MG-285 (LARANJAL)"  
1257: }, {  
1258: "usr" : "180",  
1259: "pwd" : "0310",  
1260: "fase" : "3",  
1261: "ladoA" : "",  
1262: "ladoB" : "",  
1263: "trecho" : "PONTE S/ RIO PARÁ - ENTR MG-423(A) (PITANGUI)"  
1264: }, {  
1265: "usr" : "181",  
1266: "pwd" : "0195",  
1267: "fase" : "2",  
1268: "ladoA" : "",  
1269: "ladoB" : "",  
1270: "trecho" : "ACESSO MORADA NOVA DE MINAS - ACESSO SÃO JOSÉ DO BURITI"  
1271: }, {  
1272: "usr" : "182",  
1273: "pwd" : "0330",  
1274: "fase" : "2",  
1275: "ladoA" : "",  
1276: "ladoB" : "",  
1277: "trecho" : "ENTR BR-367(B) (GOUVEIA) - ACESSO NOSSA SENHORA DA GLÓRIA"  
1278: }, {  
1279: "usr" : "183",  
1280: "pwd" : "0910",  
1281: "fase" : "2",  
1282: "ladoA" : "",  
1283: "ladoB" : "",  
1284: "trecho" : "ENTR MG-255 - ENTR BR-262(B) (DIV MG/SP)"  
1285: }, {  
1286: "usr" : "184",  
1287: "pwd" : "0250",  
1288: "fase" : "3",  
1289: "ladoA" : "",  
1290: "ladoB" : "",  
1291: "trecho" : "ENTR BR-462(B) - ENTR BR-262",  
1292: "perguntaExtra" : true  
1293: }, {  
1294: "usr" : "185",  
1295: "pwd" : "0930",  
1296: "fase" : "4",  
1297: "ladoA" : "",  
1298: "ladoB" : "",  
1299: "trecho" : "ENTR BR-452 (P/UBERLÂNDIA) - ENTR BR-462 (PERDIZES)",  
1300: "perguntaExtra" : true  
1301: }, {  
1302: "usr" : "186",  
1303: "pwd" : "0070",  
1304: "fase" : "2",  
1305: "ladoA" : "",  
1306: "ladoB" : "",  
1307: "trecho" : "ENTR BR-265(B) - CAMPOS GERAIS"  
1308: }, {  
1309: "usr" : "187",  
1310: "pwd" : "0030",  
1311: "fase" : "2",  
1312: "ladoA" : "",  
1313: "ladoB" : "",  
1314: "trecho" : "ENTR MG-354(B) - ENTR BR-265(A)"  
1315: }, {  
1316: "usr" : "188",  
1317: "pwd" : "0190",  
1318: "fase" : "2",  
1319: "ladoA" : "",  
1320: "ladoB" : "",  
1321: "trecho" : "ENTR MG-135/338 (BARBACENA) - ACESSO TIRADENTES"  
1322: }, {  
1323: "usr" : "189",  
1324: "pwd" : "0670",  
1325: "fase" : "4",  
1326: "ladoA" : "",  
1327: "ladoB" : "",  
1328: "trecho" : "ENTR BR-265 (LAVRAS) - ENTR MG-167(A) (P/TRÊS CORAÇÕES)"  
1329: }, {  
1330: "usr" : "190",  
1331: "pwd" : "0350",  
1332: "fase" : "2",  
1333: "ladoA" : "",  
1334: "ladoB" : "",  
1335: "trecho" : "ACESSO FORMIGA - ENTR MG-164 (CANDEIAS)"  
1336: }, {  
1337: "usr" : "191",  
1338: "pwd" : "0590",  
1339: "fase" : "2",  
1340: "ladoA" : "",  
1341: "ladoB" : "",  
1342: "trecho" : "ENTR MG-270 (CARMÓPOLIS DE MINAS) - ENTR BR-494(A) (P/OLIVEIRA)"  
1343: }, {  
1344: "usr" : "192",  
1345: "pwd" : "0230",  
1346: "fase" : "3",  
1347: "ladoA" : "",  
1348: "ladoB" : "",  
1349: "trecho" : "ENTR BR-354(A) (P/BAMBUÍ) - ENTR BR-354(B)"  
1350: }, {  
1351: "usr" : "193",  
1352: "pwd" : "0205",  
1353: "fase" : "2",  
1354: "ladoA" : "",  
1355: "ladoB" : "",  
1356: "trecho" : "ACESSO GUIMARÂNIA - ENTR MG-188"  
1357: }, {  
1358: "usr" : "194",  
1359: "pwd" : "0090",  
1360: "fase" : "3",  
1361: "ladoA" : "",  
1362: "ladoB" : "",  
1363: "trecho" : "ENTR MG-410 - PRESIDENTE OLEGÁRIO"  
1364: }, {  
1365: "usr" : "195",  
1366: "pwd" : "0150",  
1367: "fase" : "4",  
1368: "ladoA" : "",  
1369: "ladoB" : "",  
1370: "trecho" : "ENTR MG-060 (P/SÃO GONÇALO DO ABAETÉ) - ACESSO GALENA"  
1371: }, {  
1372: "usr" : "196",  
1373: "pwd" : "0410",  
1374: "fase" : "2",  
1375: "ladoA" : "",  
1376: "ladoB" : "",  
1377: "trecho" : "ENTR BR-154(B)/461/464 (ITUIUTABA) - ACESSO GURINHATÃ"  
1378: }, {  
1379: "usr" : "197",  
1380: "pwd" : "0830",  
1381: "fase" : "2",  
1382: "ladoA" : "",  
1383: "ladoB" : "",  
1384: "trecho" : "ENTR BR-365 (P/MONTE ALEGRE DE MINAS) - ENTR BR-464/497 (P/PRATA)"  
1385: }, {  
1386: "usr" : "198",  
1387: "pwd" : "0170",  
1388: "fase" : "3",  
1389: "ladoA" : "",  
1390: "ladoB" : "",  
1391: "trecho" : "ENTR MG-229 (P/SENHORA DO PORTO) - ENTR MG-232 (P/CARMESIA)"  
1392: }, {  
1393: "usr" : "199",  
1394: "pwd" : "0370",  
1395: "fase" : "2",  
1396: "ladoA" : "",  
1397: "ladoB" : "",  
1398: "trecho" : "ENTR MG-320 (P/SÃO JOSÉ DO GOIABAL) - ENTR BR-120 (VARGEM LINDA)"  
1399: }, {  
1400: "usr" : "200",  
1401: "pwd" : "0170",  
1402: "fase" : "2",  
1403: "ladoA" : "",  
1404: "ladoB" : "",  
1405: "trecho" : "ACESSO À GOV. VALADARES - PERIQUITO"  
1406: }, {  
1407: "usr" : "201",  
1408: "pwd" : "0685",  
1409: "fase" : "3",  
1410: "ladoA" : "",  
1411: "ladoB" : "",  
1412: "trecho" : "ACESSO FLORESTAL - ENTR BR-352 (PARÁ DE MINAS)"  
1413: }, {  
1414: "usr" : "202",  
1415: "pwd" : "0050",  
1416: "fase" : "2",  
1417: "ladoA" : "",  
1418: "ladoB" : "",  
1419: "trecho" : "ENTR MG-050(B) (P/DIVINOPÓLIS) - ENTR MG-260 (P/CLÁUDIO)"  
1420: }, {  
1421: "usr" : "203",  
1422: "pwd" : "0270",  
1423: "fase" : "2",  
1424: "ladoA" : "",  
1425: "ladoB" : "",  
1426: "trecho" : "ACESSO CONCEIÇÃO DO RIO VERDE - ENTR BR-460"  
1427: }, {  
1428: "usr" : "204",  
1429: "pwd" : "0023",  
1430: "fase" : "3",  
1431: "ladoA" : "",  
1432: "ladoB" : "",  
1433: "trecho" : "ACESSO SANTA RITA DAS CALDAS - ENTR MG-179 (POUSO ALEGRE)"  
1434: }, {  
1435: "usr" : "205",  
1436: "pwd" : "0270",  
1437: "fase" : "3",  
1438: "ladoA" : "",  
1439: "ladoB" : "",  
1440: "trecho" : "ENTR BR-369 (BOTELHOS) - ENTR BR-267(A) (P/BANDEIRA DO SUL)"  
1441: }, {  
1442: "usr" : "206",  
1443: "pwd" : "0212",  
1444: "fase" : "3",  
1445: "ladoA" : "",  
1446: "ladoB" : "",  
1447: "trecho" : "ENTR BR-487(A) - ACESSO NAVIRAÍ I"  
1448: }, {  
1449: "usr" : "207",  
1450: "pwd" : "0612",  
1451: "fase" : "1",  
1452: "ladoA" : "ENTR BR-267(A)",  
1453: "ladoB" : "INÍCIO DUPLICAÇÃO",  
1454: "trecho" : "ENTR BR-267(A) - INÍCIO DUPLICAÇÃO"  
1455: }, {  
1456: "usr" : "208",  
1457: "pwd" : "0090",  
1458: "fase" : "3",  
1459: "ladoA" : "",  
1460: "ladoB" : "",  
1461: "trecho" : "ENTR MS-141 (IVINHEMA) - VILA AMANDINA"  
1462: }, {  
1463: "usr" : "209",  
1464: "pwd" : "1390",  
1465: "fase" : "1",  
1466: "ladoA" : "ENTR MS-339/446/448 (MIRANDA)",  
1467: "ladoB" : "ENTR MS-185/243 (GUAICURUS)",  
1468: "trecho" : "ENTR MS-339/446/448 (MIRANDA) - ENTR MS-185/243 (GUAICURUS)"  
1469: }, {  
1470: "usr" : "210",  
1471: "pwd" : "0058",  
1472: "fase" : "1",  
1473: "ladoA" : "BURITIZINHO",  
1474: "ladoB" : "ENTR MS-223(A) (SILVOLÂNDIA)",  
1475: "trecho" : "BURITIZINHO - ENTR MS-223(A) (SILVOLÂNDIA)"  
1476: }, {  
1477: "usr" : "211",  
1478: "pwd" : "0390",  
1479: "fase" : "1",  
1480: "ladoA" : "ENTR MS-316 (PARAÍSO DAS ÁGUAS)",  
1481: "ladoB" : "ENTR MS-324(A)",  
1482: "trecho" : "ENTR MS-316 (PARAÍSO) - ENTR MS-324(A)"  
1483: }, {  
1484: "usr" : "212",  
1485: "pwd" : "0180",  
1486: "fase" : "3",  
1487: "ladoA" : "",  
1488: "ladoB" : "",  
1489: "trecho" : "ENTR BR-158(B) (P/TAMBOARA) - ACESSO ALTO PARANÁ/MARISTELA"  
1490: }, {  
1491: "usr" : "213",  
1492: "pwd" : "0050",  
1493: "fase" : "2",  
1494: "ladoA" : "",  
1495: "ladoB" : "",  
1496: "trecho" : "ENTR SP-425 (PRESIDENTE PRUDENTE) - ENTR SP-425"  
1497: }, {  
1498: "usr" : "214",  
1499: "pwd" : "0330",  
1500: "fase" : "1",  
1501: "ladoA" : "ENTR MT-373",  
1502: "ladoB" : "ENTR MT-130(A) (PRIMAVERA DO LESTE)",  
1503: "trecho" : "ENTR MT-373 - ENTR MT-130(A) (PRIMAVERA DO LESTE)"  
1504: }, {  
1505: "usr" : "215",  
1506: "pwd" : "0765",  
1507: "fase" : "2",  
1508: "ladoA" : "",  
1509: "ladoB" : "",  
1510: "trecho" : "FIM VARIANTE II SERRA DE SÃO VICENTE - ACESSO DISTRITO INDUSTRIAL"  
1511: }, {  
1512: "usr" : "216",  
1513: "pwd" : "0150",  
1514: "fase" : "3",  
1515: "ladoA" : "",  
1516: "ladoB" : "",  
1517: "trecho" : "ENTR BR-010(B) (P/SANTA MARIA DO PARÁ) - ENTR PA-324 (P/SALINÓPOLIS)"  
1518: }, {  
1519: "usr" : "217",  
1520: "pwd" : "0270",  
1521: "fase" : "3",  
1522: "ladoA" : "",  
1523: "ladoB" : "",  
1524: "trecho" : "ENTR PB-177 (SOLEDADE) - ENTR PB-195 (JUAZEIRINHO)"  
1525: }, {  
1526: "usr" : "218",  
1527: "pwd" : "0322",  
1528: "fase" : "2",  
1529: "ladoA" : "",  
1530: "ladoB" : "",  
1531: "trecho" : "BARRA DE SANTANA - ENTR PB-196 (P/RIACHO DE SANTO ANTÔNIO)"  
1532: }, {  
1533: "usr" : "219",  
1534: "pwd" : "0280",  
1535: "fase" : "2",  
1536: "ladoA" : "",  
1537: "ladoB" : "",  
1538: "trecho" : "ENTR PE-263 (P/ITAPETIM) - ENTR PE-320 (SÃO JOSÉ DO EGITO)"  
1539: }, {  
1540: "usr" : "220",  
1541: "pwd" : "0410",  
1542: "fase" : "3",  
1543: "ladoA" : "",  
1544: "ladoB" : "",  
1545: "trecho" : "ENTR PE-015 (P/PAULISTA - VIADUTO SOBRE PE-015) - VIADUTO SOBRE AVENIDA CAXANGÁ"  
1546: }, {  
1547: "usr" : "221",  
1548: "pwd" : "0530",  
1549: "fase" : "2",  
1550: "ladoA" : "",  
1551: "ladoB" : "",  
1552: "trecho" : "DIV PE/AL - ENTR BR-423(A)"  
1553: }, {  
1554: "usr" : "222",  
1555: "pwd" : "0630",  
1556: "fase" : "2",  
1557: "ladoA" : "",  
1558: "ladoB" : "",  
1559: "trecho" : "ENTR PE-590 (P/IPUBÍ) - ENTR BR-122 (OURICURÍ)"  
1560: }, {  
1561: "usr" : "223",  
1562: "pwd" : "0250",  
1563: "fase" : "3",  
1564: "ladoA" : "",  
1565: "ladoB" : "",  
1566: "trecho" : "ENTR CE-292(B) - DIV CE/PE"  
1567: }, {  
1568: "usr" : "224",  
1569: "pwd" : "0300",  
1570: "fase" : "3",  
1571: "ladoA" : "",  
1572: "ladoB" : "",  
1573: "trecho" : "ENTR PE-310/312 (CUSTÓDIA) - ENTR BR-426/PE-340 (SÍTIO DOS NUNES)"  
1574: }, {  
1575: "usr" : "225",  
1576: "pwd" : "0250",  
1577: "fase" : "2",  
1578: "ladoA" : "",  
1579: "ladoB" : "",  
1580: "trecho" : "RAJADA - ENTR BR-235(A)"  
1581: }, {  
1582: "usr" : "226",  
1583: "pwd" : "0115",  
1584: "fase" : "2",  
1585: "ladoA" : "",  
1586: "ladoB" : "",  
1587: "trecho" : "ENTR PI-394 - ENTR PI-140 (CANTO DO BURITI)"  
1588: }, {  
1589: "usr" : "227",  
1590: "pwd" : "0070",  
1591: "fase" : "3",  
1592: "ladoA" : "",  
1593: "ladoB" : "",  
1594: "trecho" : "ENTR PI-211 (CAXINGÓ) - ENTR PI-213(A)"  
1595: }, {  
1596: "usr" : "228",  
1597: "pwd" : "0190",  
1598: "fase" : "2",  
1599: "ladoA" : "",  
1600: "ladoB" : "",  
1601: "trecho" : "ACESSO OESTE SOBRAL - ENTR BR-403(B)/CE-183"  
1602: }, {  
1603: "usr" : "229",  
1604: "pwd" : "0330",  
1605: "fase" : "2",  
1606: "ladoA" : "",  
1607: "ladoB" : "",  
1608: "trecho" : "ENTR MA-034 (SÃO JOÃO DOS PATOS) - ENTR BR-230(B)/MA-364(A) (DOIS IRMÃOS)"  
1609: }, {  
1610: "usr" : "230",  
1611: "pwd" : "0290",  
1612: "fase" : "2",  
1613: "ladoA" : "",  
1614: "ladoB" : "",  
1615: "trecho" : "ENTR PI-130 (P/AMARANTE) - ENTR PI-120"  
1616: }, {  
1617: "usr" : "231",  
1618: "pwd" : "0210",  
1619: "fase" : "3",  
1620: "ladoA" : "",  
1621: "ladoB" : "",  
1622: "trecho" : "ENTR SP-252 (GUAPIARA) - ENTR SP-165/249 (APIAÍ)"  
1623: }, {  
1624: "usr" : "232",  
1625: "pwd" : "0180",  
1626: "fase" : "1",  
1627: "ladoA" : "ENTR PR-549 (P/LUIZIANA)",  
1628: "ladoB" : "ENTR PR-462 (P/IRETAMA)",  
1629: "trecho" : "ENTR PR-549 (P/LUIZIANA) - ENTR PR-462 (P/IRETAMA)"  
1630: }, {  
1631: "usr" : "233",  
1632: "pwd" : "0080",  
1633: "fase" : "3",  
1634: "ladoA" : "",  
1635: "ladoB" : "",  
1636: "trecho" : "ACESSO IVAIPORÃ - ENTR BR-487 (P/MANOEL RIBAS)"  
1637: }, {  
1638: "usr" : "234",  
1639: "pwd" : "0250",  
1640: "fase" : "1",  
1641: "ladoA" : "ENTR BR-158(B) (LARANJEIRAS DO SUL)",  
1642: "ladoB" : "ENTR PR-473",  
1643: "trecho" : "ENTR BR-158(B) (LARANJEIRAS DO SUL) - ENTR PR-473"  
1644: }, {  
1645: "usr" : "235",  
1646: "pwd" : "0455",  
1647: "fase" : "1",  
1648: "ladoA" : "ACESSO ÁGUA MINERAL SANTA CLARA",  
1649: "ladoB" : "ENTR PR-281(A) (P/CHOPINZINHO)",  
1650: "trecho" : "ACESSO ÁGUA MINERAL SANTA CLARA - ENTR PR-281(A) (P/CHOPINZINHO)"  
1651: }, {  
1652: "usr" : "236",  
1653: "pwd" : "0500",  
1654: "fase" : "1",  
1655: "ladoA" : "ENTR PR-468(B)",  
1656: "ladoB" : "ENTR PR-180 (GOIO ERÊ)",  
1657: "trecho" : "ENTR PR-468(B) - ENTR PR-180 (GOIO ERÊ)"  
1658: }, {  
1659: "usr" : "237",  
1660: "pwd" : "1230",  
1661: "fase" : "1",  
1662: "ladoA" : "ENTR PR-431(B)",  
1663: "ladoB" : "ENTR PR-092(A) (P/BARRA DO JACARÉ)",  
1664: "trecho" : "ENTR PR-431(B) - ENTR PR-092(A) (P/BARRA DO JACARÉ)"  
1665: }, {  
1666: "usr" : "238",  
1667: "pwd" : "1140",  
1668: "fase" : "2",  
1669: "ladoA" : "",  
1670: "ladoB" : "",  
1671: "trecho" : "ENTR SP-331(A) - ENTR SP-331(B)"  
1672: }, {  
1673: "usr" : "239",  
1674: "pwd" : "0230",  
1675: "fase" : "3",  
1676: "ladoA" : "",  
1677: "ladoB" : "",  
1678: "trecho" : "ENTR BR-369 - ENTR SP-261 (P/ÁGUAS DE SANTA BÁRBARA)"  
1679: }, {  
1680: "usr" : "240",  
1681: "pwd" : "0710",  
1682: "fase" : "1",  
1683: "ladoA" : "ACESSO PIRAPÓ",  
1684: "ladoB" : "ACESSO CAMBIRA",  
1685: "trecho" : "ACESSO PIRAPÓ - ACESSO CAMBIRA"  
1686: }, {  
1687: "usr" : "241",  
1688: "pwd" : "0410",  
1689: "fase" : "3",  
1690: "ladoA" : "",  
1691: "ladoB" : "",  
1692: "trecho" : "ACESSO PARAÍBA DO SUL - ENTR RJ-135 (ANDRADE PINTO)"  
1693: }, {  
1694: "usr" : "242",  
1695: "pwd" : "2850",  
1696: "fase" : "4",  
1697: "ladoA" : "",  
1698: "ladoB" : "",  
1699: "trecho" : "ENTR RJ-196 - ENTR RJ-182"  
1700: }, {  
1701: "usr" : "243",  
1702: "pwd" : "0590",  
1703: "fase" : "3",  
1704: "ladoA" : "",  
1705: "ladoB" : "",  
1706: "trecho" : "ITAMONTE - ENTR BR-485 (GARGANTA DO REGISTRO)"  
1707: }, {  
1708: "usr" : "244",  
1709: "pwd" : "3350",  
1710: "fase" : "3",  
1711: "ladoA" : "",  
1712: "ladoB" : "",  
1713: "trecho" : "ENTR RJ-149 (P/RIO CLARO) - ENTR BR-494(A) (ANGRA DOS REIS)"  
1714: }, {  
1715: "usr" : "245",  
1716: "pwd" : "0530",  
1717: "fase" : "3",  
1718: "ladoA" : "",  
1719: "ladoB" : "",  
1720: "trecho" : "ENTR RJ-145 (P/BARRA DO PIRAÍ) - ENTR RJ-141 (DORANDIA)"  
1721: }, {  
1722: "usr" : "246",  
1723: "pwd" : "0170",  
1724: "fase" : "4",  
1725: "ladoA" : "",  
1726: "ladoB" : "",  
1727: "trecho" : "ENTR BR-494 (ARANTINA) - AIURUOCA"  
1728: }, {  
1729: "usr" : "247",  
1730: "pwd" : "0170",  
1731: "fase" : "2",  
1732: "ladoA" : "",  
1733: "ladoB" : "",  
1734: "trecho" : "ENTR RJ-198(B) - ENTR RJ-200 (MONTE ALEGRE)"  
1735: }, {  
1736: "usr" : "248",  
1737: "pwd" : "0350",  
1738: "fase" : "4",  
1739: "ladoA" : "",  
1740: "ladoB" : "",  
1741: "trecho" : "ENTR RJ-202 - ENTR RJ-234 (P/PENEDO)"  
1742: }, {  
1743: "usr" : "249",  
1744: "pwd" : "0150",  
1745: "fase" : "3",  
1746: "ladoA" : "",  
1747: "ladoB" : "",  
1748: "trecho" : "ENTR BR-482(B) - ENTR ES-181"  
1749: }, {  
1750: "usr" : "250",  
1751: "pwd" : "0210",  
1752: "fase" : "3",  
1753: "ladoA" : "",  
1754: "ladoB" : "",  
1755: "trecho" : "ENTR BR-116/265 (MURIAÉ) - DIV MG/RJ"  
1756: }, {  
1757: "usr" : "251",  
1758: "pwd" : "1530",  
1759: "fase" : "2",  
1760: "ladoA" : "",  
1761: "ladoB" : "",  
1762: "trecho" : "ENTR RJ-156 (P/VOLTA DO PIÃO) - ENTR BR-492(A)"  
1763: }, {  
1764: "usr" : "252",  
1765: "pwd" : "0065",  
1766: "fase" : "3",  
1767: "ladoA" : "",  
1768: "ladoB" : "",  
1769: "trecho" : "ACESSO À JENIPABU - ENTR RN-160(B) (P/EXTREMOZ)"  
1770: }, {  
1771: "usr" : "253",  
1772: "pwd" : "0110",  
1773: "fase" : "3",  
1774: "ladoA" : "",  
1775: "ladoB" : "",  
1776: "trecho" : "TAIPÚ - ENTR RN-064 (P/CEARÁ MIRIM)"  
1777: }, {  
1778: "usr" : "254",  
1779: "pwd" : "0055",  
1780: "fase" : "3",  
1781: "ladoA" : "",  
1782: "ladoB" : "",  
1783: "trecho" : "ENTR CE-261 - DIV CE/RN"  
1784: }, {  
1785: "usr" : "255",  
1786: "pwd" : "0210",  
1787: "fase" : "3",  
1788: "ladoA" : "",  
1789: "ladoB" : "",  
1790: "trecho" : "ENTR BR-230(A) (SANTA GERTRUDES) - ENTR PB-275(A) (P/SÃO JOSÉ DE ESPINHARA)"  
1791: }, {  
1792: "usr" : "256",  
1793: "pwd" : "1375",  
1794: "fase" : "3",  
1795: "ladoA" : "",  
1796: "ladoB" : "",  
1797: "trecho" : "ACESSO USINA DE SAMUEL - INICIO PISTA DUPLA"  
1798: }, {  
1799: "usr" : "257",  
1800: "pwd" : "1510",  
1801: "fase" : "3",  
1802: "ladoA" : "",  
1803: "ladoB" : "",  
1804: "trecho" : "VISTA ALEGRE DO ABUANÃ - VILA EXTREMA"  
1805: }, {  
1806: "usr" : "258",  
1807: "pwd" : "1300",  
1808: "fase" : "2",  
1809: "ladoA" : "",  
1810: "ladoB" : "",  
1811: "trecho" : "ENTR BR-421 (ARIQUEMES) - ENTR RO-459(A) (ALTO PARAISO)"  
1812: }, {  
1813: "usr" : "259",  
1814: "pwd" : "0031",  
1815: "fase" : "2",  
1816: "ladoA" : "",  
1817: "ladoB" : "",  
1818: "trecho" : "ENTR RO-135 (P/NOVA LONDRINA) - FIM DO TRECHO PAVIMENTADO"  
1819: }, {  
1820: "usr" : "260",  
1821: "pwd" : "1148",  
1822: "fase" : "2",  
1823: "ladoA" : "",  
1824: "ladoB" : "",  
1825: "trecho" : "ENTR RO-471 (P/MINISTRO ANDREAZZA) - ENTR RO-479(A) (P/ROLIM DE MOURA)"  
1826: }, {  
1827: "usr" : "261",  
1828: "pwd" : "1090",  
1829: "fase" : "2",  
1830: "ladoA" : "",  
1831: "ladoB" : "",  
1832: "trecho" : "ENTR RO-391 (POSTO GUAPORÉ) - MARCO RONDON"  
1833: }, {  
1834: "usr" : "262",  
1835: "pwd" : "0132",  
1836: "fase" : "1",  
1837: "ladoA" : "PADRONAL",  
1838: "ladoB" : "RIO 12 DE OUTUBRO",  
1839: "trecho" : "PADRONAL - RIO 12 DE OUTUBRO"  
1840: }, {  
1841: "usr" : "263",  
1842: "pwd" : "4425",  
1843: "fase" : "1",  
1844: "ladoA" : "ENTR ACESSO SUL DE CAPÃO DA CANOA",  
1845: "ladoB" : "ENTR BR-290(A) (OSÓRIO)",  
1846: "trecho" : "ENTR ACESSO SUL DE CAPÃO DA CANOA - ENTR BR-290(A) (OSÓRIO)"  
1847: }, {  
1848: "usr" : "264",  
1849: "pwd" : "0330",  
1850: "fase" : "1",  
1851: "ladoA" : "ENTR BR-470",  
1852: "ladoB" : "ENTR BR-470/116(A) (CANOAS)",  
1853: "trecho" : "ENTR RSC-470 (P/ MONTENEGRO) - ENTR ERS-124 (P/ POLO PETROQUIMICO)"  
1854: }, {  
1855: "usr" : "265",  
1856: "pwd" : "0210",  
1857: "fase" : "1",  
1858: "ladoA" : "ENTR BR-153(B)/RS-324 (P/PONTÃO)",  
1859: "ladoB" : "ENTR BR-377(A)/386 (CARAZINHO)",  
1860: "trecho" : "ENTR BRS-153(B)/ERS-324 (P/ PONTAO) - ENTR BRS-377(A)/386 (P/ CARAZINHO)"  
1861: }, {  
1862: "usr" : "266",  
1863: "pwd" : "0470",  
1864: "fase" : "1",  
1865: "ladoA" : "ENTR RS-168 (P/SÃO PAULO DAS MISSÕES)",  
1866: "ladoB" : "ENTR BR-472 (FRONT BRASIL/ARGENTINA) (PORTO XAVIER)",  
1867: "trecho" : "ENTR RS-168 (P/SÃO PAULO DAS MISSÕES) - ENTR BR-472 (FRONT BRASIL/ARGENTINA) (PORTO XAVIER)"  
1868: }, {  
1869: "usr" : "267",  
1870: "pwd" : "0140",  
1871: "fase" : "1",  
1872: "ladoA" : "A. BELTRÃO",  
1873: "ladoB" : "ENTR BR-287 (SANTIAGO)",  
1874: "trecho" : "ENTR ERS-533 (P/ CAPAO CIPO) - ENTR BRS-287 (CONTORNO DE SANTIAGO)"  
1875: }, {  
1876: "usr" : "268",  
1877: "pwd" : "1850",  
1878: "fase" : "3",  
1879: "ladoA" : "ENTR BR-290(B)",  
1880: "ladoB" : "ENTR BR-392",  
1881: "trecho" : "ENTR BRS-290(B) (P/ SAO GABRIEL) - ENTR BRS-392 (P/ CACAPAVA DO SUL)"  
1882: }, {  
1883: "usr" : "269",  
1884: "pwd" : "1350",  
1885: "fase" : "1",  
1886: "ladoA" : "ENTR BR-290(A) (P/ROSÁRIO DO SUL)",  
1887: "ladoB" : "ENTR RS-640 (P/CACEQUI)",  
1888: "trecho" : "ENTR BRS-158(A) (P/ AZEVEDO SODRE) - ENTR ERS-640 (P/ CACEQUI)"  
1889: }, {  
1890: "usr" : "270",  
1891: "pwd" : "0230",  
1892: "fase" : "1",  
1893: "ladoA" : "ENTR BR-473 (SARANDI)",  
1894: "ladoB" : "CURRAL ALTO",  
1895: "trecho" : "ENTR RSC-473 (VILA SARANDI) - CURRAL ALTO"  
1896: }, {  
1897: "usr" : "271",  
1898: "pwd" : "0310",  
1899: "fase" : "1",  
1900: "ladoA" : "ENTR RS-476 (LAJEADO GRANDE)",  
1901: "ladoB" : "ENTR RS-110 (VÁRZEA DO CEDRO)",  
1902: "trecho" : "ENTR ERS-476 (LAJEADO GRANDE) - ENTR ERS-110 (VARZEA DO CEDRO)"  
1903: }, {  
1904: "usr" : "272",  
1905: "pwd" : "0015",  
1906: "fase" : "1",  
1907: "ladoA" : "ENTR BR-282(B)/470(A)",  
1908: "ladoB" : "ENTR SC-455 (P/IBICUÍ)",  
1909: "trecho" : "ENTR BR-282(B)/470(A) - ENTR SC-455 (P/IBICUÍ)"  
1910: }, {  
1911: "usr" : "273",  
1912: "pwd" : "3010",  
1913: "fase" : "1",  
1914: "ladoA" : "DIV SC/RS (RIO PELOTAS)",  
1915: "ladoB" : "ENTR BR-285(A) (P/VACARIA)",  
1916: "trecho" : "DIVISA SC/RS (RIO PELOTAS) - ENTR BRS-285(A) (P/ VACARIA)"  
1917: }, {  
1918: "usr" : "274",  
1919: "pwd" : "0110",  
1920: "fase" : "1",  
1921: "ladoA" : "ENTR BR-486 (BOM RETIRO)",  
1922: "ladoB" : "ENTR SC-345(B) (P/URUBICI)",  
1923: "trecho" : "ENTR BR-486 (BOM RETIRO) - ENTR SC-345(B) (P/URUBICI)"  
1924: }, {  
1925: "usr" : "275",  
1926: "pwd" : "0027",  
1927: "fase" : "1",  
1928: "ladoA" : "ENTR SC-473 (P/ANCHIETA)",  
1929: "ladoB" : "SÃO JOSÉ DO CEDRO",  
1930: "trecho" : "ENTR SC-473 (P/ANCHIETA) - SÃO JOSÉ DO CEDRO"  
1931: }, {  
1932: "usr" : "276",  
1933: "pwd" : "0290",  
1934: "fase" : "3",  
1935: "ladoA" : "",  
1936: "ladoB" : "",  
1937: "trecho" : "ENTR BR-280(A) - ENTR BR-280(B) (VITORINO)"  
1938: }, {  
1939: "usr" : "277",  
1940: "pwd" : "0612",  
1941: "fase" : "2",  
1942: "ladoA" : "",  
1943: "ladoB" : "",  
1944: "trecho" : "ENTR BA-392 (ANTAS) - ENTR BA-220 (CÍCERO DANTAS)"  
1945: }, {  
1946: "usr" : "278",  
1947: "pwd" : "0190",  
1948: "fase" : "3",  
1949: "ladoA" : "",  
1950: "ladoB" : "",  
1951: "trecho" : "ENTR SP-183 - ENTR BR-116(A) (P/LORENA)"  
1952: }, {  
1953: "usr" : "279",  
1954: "pwd" : "2360",  
1955: "fase" : "3",  
1956: "ladoA" : "",  
1957: "ladoB" : "",  
1958: "trecho" : "ENTR SP-062 (P/ROSEIRA) - ENTR BR-383(A)/SP-132 (PINDAMONHANGABA)"  
1959: }, {  
1960: "usr" : "280",  
1961: "pwd" : "2470",  
1962: "fase" : "3",  
1963: "ladoA" : "",  
1964: "ladoB" : "",  
1965: "trecho" : "ENTR SP-065 (P/IGARATÁ) - ENTR SP-056 (ARUJÁ)"  
1966: }, {  
1967: "usr" : "281",  
1968: "pwd" : "2630",  
1969: "fase" : "3",  
1970: "ladoA" : "",  
1971: "ladoB" : "",  
1972: "trecho" : "ENTR BR-116(A) (JUQUIÁ) - ENTR SP-139 (REGISTRO)"  
1973: }, {  
1974: "usr" : "282",  
1975: "pwd" : "0130",  
1976: "fase" : "3",  
1977: "ladoA" : "",  
1978: "ladoB" : "",  
1979: "trecho" : "ENTR BR-373(A) (ITAPETININGA) - ENTR BR-373(B)"  
1980: }, {  
1981: "usr" : "283",  
1982: "pwd" : "1077",  
1983: "fase" : "3",  
1984: "ladoA" : "",  
1985: "ladoB" : "",  
1986: "trecho" : "ACESSO GUAIMBÉ - ENTR SP-333(A)"  
1987: }, {  
1988: "usr" : "284",  
1989: "pwd" : "1010",  
1990: "fase" : "2",  
1991: "ladoA" : "",  
1992: "ladoB" : "",  
1993: "trecho" : "ENTR SP-355 - ENTR SP-425"  
1994: }, {  
1995: "usr" : "285",  
1996: "pwd" : "0110",  
1997: "fase" : "3",  
1998: "ladoA" : "",  
1999: "ladoB" : "",  
2000: "trecho" : "ENTR SP-255 (ARARAQUARA) - ENTR BR-364(B)"  
2001: }, {  
2002: "usr" : "286",  
2003: "pwd" : "0783",  
2004: "fase" : "3",  
2005: "ladoA" : "",  
2006: "ladoB" : "",  
2007: "trecho" : "ACESSO RINÓPOLIS - ENTR SP-294 (PARAPUÃ)"  
2008: }, {  
2009: "usr" : "287",  
2010: "pwd" : "0850",  
2011: "fase" : "3",  
2012: "ladoA" : "",  
2013: "ladoB" : "",  
2014: "trecho" : "ENTR SP-065 (ATIBAIA) - ENTR SP-023 (MARIPORÃ)"  
2015: }, {  
2016: "usr" : "288",  
2017: "pwd" : "0010",  
2018: "fase" : "3",  
2019: "ladoA" : "",  
2020: "ladoB" : "",  
2021: "trecho" : "ENTR BR-222/230(A)/PA-150 (MARABÁ) - ENTR PA-405"  
2022: }, {  
2023: "usr" : "289",  
2024: "pwd" : "0390",  
2025: "fase" : "2",  
2026: "ladoA" : "",  
2027: "ladoB" : "",  
2028: "trecho" : "ENTR BR-010(A) (PORTO FRANCO) - ENTR BR-010(B)/230(A)/MA-138 (ESTREITO)"  
2029: }, {  
2030: "usr" : "290",  
2031: "pwd" : "1010",  
2032: "fase" : "3",  
2033: "ladoA" : "",  
2034: "ladoB" : "",  
2035: "trecho" : "ENTR MA-012/375 (SÃO RAIMUNDO DAS MANGABEIRAS) - ENTR MA-006(A)"  
2036: }, {  
2037: "usr" : "291",  
2038: "pwd" : "0290",  
2039: "fase" : "3",  
2040: "ladoA" : "",  
2041: "ladoB" : "",  
2042: "trecho" : "FIM PISTA DUPLA - GURUPÍ - ENTR BR-242(B)/TO-280"  
2043: }, {  
2044: "usr" : "292",  
2045: "pwd" : "0330",  
2046: "fase" : "4",  
2047: "ladoA" : "",  
2048: "ladoB" : "",  
2049: "trecho" : "ENTR BR-020(A)/135(B)/BA-020 - ENTR BR-020(B)"  
2050: }, {  
2051: "usr" : "293",  
2052: "pwd" : "0272",  
2053: "fase" : "3",  
2054: "ladoA" : "",  
2055: "ladoB" : "",  
2056: "trecho" : "INDIANÓPOLIS - VALE DO SONHO"  
2057: }, {  
2058: "usr" : "294",  
2059: "pwd" : "0210",  
2060: "fase" : "4",  
2061: "ladoA" : "",  
2062: "ladoB" : "",  
2063: "trecho" : "ENTR GO-217 (P/MAIRIPOTABA) - ENTR GO-320 (INDIARA)"  
2064: }, {  
2065: "usr" : "295",  
2066: "pwd" : "0370",  
2067: "fase" : "1",  
2068: "ladoA" : "ENTR BR-153 (P/TIBAGI/IPIRANGA)",  
2069: "ladoB" : "ENTR BR-373(A)/487(A) (CAETANO)",  
2070: "trecho" : "ENTR BR-153 (P/TIBAGI/IPIRANGA) - ENTR BR-373(A)/487(A) (CAETANO)"  
2071: }, {  
2072: "usr" : "296",  
2073: "pwd" : "0390",  
2074: "fase" : "1",  
2075: "ladoA" : "RIO ANHANDUÍ",  
2076: "ladoB" : "ENTR BR-262(A) (CAMPO GRANDE)",  
2077: "trecho" : "RIO ANHANDUÍ - ENTR BR-262(A) (CAMPO GRANDE)"  
2078: }, {  
2079: "usr" : "297",  
2080: "pwd" : "0434",  
2081: "fase" : "3",  
2082: "ladoA" : "",  
2083: "ladoB" : "",  
2084: "trecho" : "ACESSO ITAJÁ (GO) - ENTR MS-431 (P/SÃO JOÃO DO APORÉ)"  
2085: }, {  
2086: "usr" : "298",  
2087: "pwd" : "0930",  
2088: "fase" : "3",  
2089: "ladoA" : "",  
2090: "ladoB" : "",  
2091: "trecho" : "FNM (ENTR PISTA INVERSA) - ENTR BR-116(A)/493/RJ-109"  
2092: }, {  
2093: "usr" : "299",  
2094: "pwd" : "0110",  
2095: "fase" : "1",  
2096: "ladoA" : "ENTR SC-418 (P/POMERODE)",  
2097: "ladoB" : "ENTR BR-477(B) (P/TIMBÓ)",  
2098: "trecho" : "ENTR SC-418 (P/POMERODE) - ENTR BR-477(B) (P/TIMBÓ)"  
2099: }, {  
2100: "usr" : "300",  
2101: "pwd" : "0110",  
2102: "fase" : "1",  
2103: "ladoA" : "ENTR BR-116(B)",  
2104: "ladoB" : "P/TRES BARRAS",  
2105: "trecho" : "ENTR BR-116(B) - P/TRES BARRAS"  
2106: }, {  
2107: "usr" : "301",  
2108: "pwd" : "0150",  
2109: "fase" : "2",  
2110: "ladoA" : "",  
2111: "ladoB" : "",  
2112: "trecho" : "DIV PI/BA - ENTR BR-235/BA-161 (REMANSO)"  
2113: }, {  
2114: "usr" : "302",  
2115: "pwd" : "0770",  
2116: "fase" : "2",  
2117: "ladoA" : "",  
2118: "ladoB" : "",  
2119: "trecho" : "ENTR BR-369(A) (BARBOSA FERRAZ) - ENTR BR-158(A)/369(B) (ANEL VIÁRIO CAMPO MOURÃO)"  
2120: }  
2121: ]  
2122:}

## .\mobile\.esdoc.json

1:{  
 2: "source": "./www",  
 3: "destination": "./esdocs",  
 4: "plugins": [  
 5: {"name": "esdoc-standard-plugin"}  
 6: ]  
 7:}

## .\mobile\package-lock.json

1:{  
 2: "name": "br.ufrj.coppetec.contdnit",  
 3: "version": "2.4.0",  
 4: "lockfileVersion": 1,  
 5: "requires": true,  
 6: "dependencies": {  
 7: "@types/node": {  
 8: "version": "10.1.3",  
 9: "resolved": "https://registry.npmjs.org/@types/node/-/node-10.1.3.tgz",  
 10: "integrity": "sha512-GiCx7dRvta0hbxXoJFAUxz+CKX6bZSCKjM5slq2vPp/5zwK01T4ibYZkGr6EN4F2QmxDQR76/ZHg6q+7iFWCWw==",  
 11: "dev": true  
 12: },  
 13: "abab": {  
 14: "version": "1.0.4",  
 15: "resolved": "https://registry.npmjs.org/abab/-/abab-1.0.4.tgz",  
 16: "integrity": "sha1-X6rZwsB/YN12dw9xzwJbYqY8/U4=",  
 17: "dev": true,  
 18: "optional": true  
 19: },  
 20: "accepts": {  
 21: "version": "1.3.5",  
 22: "resolved": "https://registry.npmjs.org/accepts/-/accepts-1.3.5.tgz",  
 23: "integrity": "sha1-63d99gEXI6OxTopywIBcjoZ0a9I=",  
 24: "dev": true,  
 25: "requires": {  
 26: "mime-types": "2.1.18",  
 27: "negotiator": "0.6.1"  
 28: }  
 29: },  
 30: "acorn": {  
 31: "version": "2.7.0",  
 32: "resolved": "https://registry.npmjs.org/acorn/-/acorn-2.7.0.tgz",  
 33: "integrity": "sha1-q259nYhqrKiwhbwzEreaGYQz8Oc=",  
 34: "dev": true  
 35: },  
 36: "acorn-globals": {  
 37: "version": "1.0.9",  
 38: "resolved": "https://registry.npmjs.org/acorn-globals/-/acorn-globals-1.0.9.tgz",  
 39: "integrity": "sha1-VbtemGkVB7dFedBRNBMhfDgMVM8=",  
 40: "dev": true,  
 41: "optional": true,  
 42: "requires": {  
 43: "acorn": "2.7.0"  
 44: }  
 45: },  
 46: "ajv": {  
 47: "version": "5.5.2",  
 48: "resolved": "https://registry.npmjs.org/ajv/-/ajv-5.5.2.tgz",  
 49: "integrity": "sha1-c7Xuyj+rZT49P5Qis0GtQiBdyWU=",  
 50: "dev": true,  
 51: "optional": true,  
 52: "requires": {  
 53: "co": "4.6.0",  
 54: "fast-deep-equal": "1.1.0",  
 55: "fast-json-stable-stringify": "2.0.0",  
 56: "json-schema-traverse": "0.3.1"  
 57: }  
 58: },  
 59: "ansi-regex": {  
 60: "version": "2.1.1",  
 61: "resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-regex-2.1.1.tgz",  
 62: "integrity": "sha1-w7M6te42DYbg5ijwRorn7yfWVN8=",  
 63: "dev": true  
 64: },  
 65: "ansi-styles": {  
 66: "version": "2.2.1",  
 67: "resolved": "https://registry.npmjs.org/ansi-styles/-/ansi-styles-2.2.1.tgz",  
 68: "integrity": "sha1-tDLdM1i2NM914eRmQ2gkBTPB3b4=",  
 69: "dev": true  
 70: },  
 71: "argparse": {  
 72: "version": "1.0.10",  
 73: "resolved": "https://registry.npmjs.org/argparse/-/argparse-1.0.10.tgz",  
 74: "integrity": "sha512-o5Roy6tNG4SL/FOkCAN6RzjiakZS25RLYFrcMttJqbdd8BWrnA+fGz57iN5Pb06pvBGvl5gQ0B48dJlslXvoTg==",  
 75: "dev": true,  
 76: "requires": {  
 77: "sprintf-js": "1.0.3"  
 78: }  
 79: },  
 80: "array-flatten": {  
 81: "version": "1.1.1",  
 82: "resolved": "https://registry.npmjs.org/array-flatten/-/array-flatten-1.1.1.tgz",  
 83: "integrity": "sha1-ml9pkFGx5wczKPKgCJaLZOopVdI=",  
 84: "dev": true  
 85: },  
 86: "asn1": {  
 87: "version": "0.2.3",  
 88: "resolved": "https://registry.npmjs.org/asn1/-/asn1-0.2.3.tgz",  
 89: "integrity": "sha1-2sh4dxPJlmhJ/IGAd36+nB3fO4Y=",  
 90: "dev": true,  
 91: "optional": true  
 92: },  
 93: "assert-plus": {  
 94: "version": "1.0.0",  
 95: "resolved": "https://registry.npmjs.org/assert-plus/-/assert-plus-1.0.0.tgz",  
 96: "integrity": "sha1-8S4PPF13sLHN2RRpQuTpbB5N1SU=",  
 97: "dev": true  
 98: },  
 99: "async": {  
 100: "version": "0.9.2",  
 101: "resolved": "https://registry.npmjs.org/async/-/async-0.9.2.tgz",  
 102: "integrity": "sha1-rqdNXmHB+JlhO/ZL2mbUx48v0X0=",  
 103: "dev": true,  
 104: "optional": true  
 105: },  
 106: "asynckit": {  
 107: "version": "0.4.0",  
 108: "resolved": "https://registry.npmjs.org/asynckit/-/asynckit-0.4.0.tgz",  
 109: "integrity": "sha1-x57Zf380y48robyXkLzDZkdLS3k=",  
 110: "dev": true,  
 111: "optional": true  
 112: },  
 113: "aws-sign2": {  
 114: "version": "0.7.0",  
 115: "resolved": "https://registry.npmjs.org/aws-sign2/-/aws-sign2-0.7.0.tgz",  
 116: "integrity": "sha1-tG6JCTSpWR8tL2+G1+ap8bP+dqg=",  
 117: "dev": true,  
 118: "optional": true  
 119: },  
 120: "aws4": {  
 121: "version": "1.7.0",  
 122: "resolved": "https://registry.npmjs.org/aws4/-/aws4-1.7.0.tgz",  
 123: "integrity": "sha512-32NDda82rhwD9/JBCCkB+MRYDp0oSvlo2IL6rQWA10PQi7tDUM3eqMSltXmY+Oyl/7N3P3qNtAlv7X0d9bI28w==",  
 124: "dev": true,  
 125: "optional": true  
 126: },  
 127: "babel-code-frame": {  
 128: "version": "6.26.0",  
 129: "resolved": "https://registry.npmjs.org/babel-code-frame/-/babel-code-frame-6.26.0.tgz",  
 130: "integrity": "sha1-Y/1D99weO7fONZR9uP42mj9Yx0s=",  
 131: "dev": true,  
 132: "requires": {  
 133: "chalk": "1.1.3",  
 134: "esutils": "2.0.2",  
 135: "js-tokens": "3.0.2"  
 136: }  
 137: },  
 138: "babel-generator": {  
 139: "version": "6.26.1",  
 140: "resolved": "https://registry.npmjs.org/babel-generator/-/babel-generator-6.26.1.tgz",  
 141: "integrity": "sha512-HyfwY6ApZj7BYTcJURpM5tznulaBvyio7/0d4zFOeMPUmfxkCjHocCuoLa2SAGzBI8AREcH3eP3758F672DppA==",  
 142: "dev": true,  
 143: "requires": {  
 144: "babel-messages": "6.23.0",  
 145: "babel-runtime": "6.26.0",  
 146: "babel-types": "6.26.0",  
 147: "detect-indent": "4.0.0",  
 148: "jsesc": "1.3.0",  
 149: "lodash": "4.17.10",  
 150: "source-map": "0.5.7",  
 151: "trim-right": "1.0.1"  
 152: }  
 153: },  
 154: "babel-messages": {  
 155: "version": "6.23.0",  
 156: "resolved": "https://registry.npmjs.org/babel-messages/-/babel-messages-6.23.0.tgz",  
 157: "integrity": "sha1-8830cDhYA1sqKVHG7F7fbGLyYw4=",  
 158: "dev": true,  
 159: "requires": {  
 160: "babel-runtime": "6.26.0"  
 161: }  
 162: },  
 163: "babel-runtime": {  
 164: "version": "6.26.0",  
 165: "resolved": "https://registry.npmjs.org/babel-runtime/-/babel-runtime-6.26.0.tgz",  
 166: "integrity": "sha1-llxwWGaOgrVde/4E/yM3vItWR/4=",  
 167: "dev": true,  
 168: "requires": {  
 169: "core-js": "2.5.7",  
 170: "regenerator-runtime": "0.11.1"  
 171: }  
 172: },  
 173: "babel-traverse": {  
 174: "version": "6.26.0",  
 175: "resolved": "https://registry.npmjs.org/babel-traverse/-/babel-traverse-6.26.0.tgz",  
 176: "integrity": "sha1-RqnL1+3MYsjlwGTi0tjQ9ANXZu4=",  
 177: "dev": true,  
 178: "requires": {  
 179: "babel-code-frame": "6.26.0",  
 180: "babel-messages": "6.23.0",  
 181: "babel-runtime": "6.26.0",  
 182: "babel-types": "6.26.0",  
 183: "babylon": "6.18.0",  
 184: "debug": "2.6.9",  
 185: "globals": "9.18.0",  
 186: "invariant": "2.2.4",  
 187: "lodash": "4.17.10"  
 188: }  
 189: },  
 190: "babel-types": {  
 191: "version": "6.26.0",  
 192: "resolved": "https://registry.npmjs.org/babel-types/-/babel-types-6.26.0.tgz",  
 193: "integrity": "sha1-o7Bz+Uq0nrb6Vc1lInozQ4BjJJc=",  
 194: "dev": true,  
 195: "requires": {  
 196: "babel-runtime": "6.26.0",  
 197: "esutils": "2.0.2",  
 198: "lodash": "4.17.10",  
 199: "to-fast-properties": "1.0.3"  
 200: }  
 201: },  
 202: "babylon": {  
 203: "version": "6.18.0",  
 204: "resolved": "https://registry.npmjs.org/babylon/-/babylon-6.18.0.tgz",  
 205: "integrity": "sha512-q/UEjfGJ2Cm3oKV71DJz9d25TPnq5rhBVL2Q4fA5wcC3jcrdn7+SssEybFIxwAvvP+YCsCYNKughoF33GxgycQ==",  
 206: "dev": true  
 207: },  
 208: "balanced-match": {  
 209: "version": "1.0.0",  
 210: "resolved": "https://registry.npmjs.org/balanced-match/-/balanced-match-1.0.0.tgz",  
 211: "integrity": "sha1-ibTRmasr7kneFk6gK4nORi1xt2c=",  
 212: "dev": true  
 213: },  
 214: "bcrypt-pbkdf": {  
 215: "version": "1.0.1",  
 216: "resolved": "https://registry.npmjs.org/bcrypt-pbkdf/-/bcrypt-pbkdf-1.0.1.tgz",  
 217: "integrity": "sha1-Y7xdy2EzG5K8Bf1SiVPDNGKgb40=",  
 218: "dev": true,  
 219: "optional": true,  
 220: "requires": {  
 221: "tweetnacl": "0.14.5"  
 222: }  
 223: },  
 224: "bluebird": {  
 225: "version": "3.5.1",  
 226: "resolved": "https://registry.npmjs.org/bluebird/-/bluebird-3.5.1.tgz",  
 227: "integrity": "sha512-MKiLiV+I1AA596t9w1sQJ8jkiSr5+ZKi0WKrYGUn6d1Fx+Ij4tIj+m2WMQSGczs5jZVxV339chE8iwk6F64wjA==",  
 228: "dev": true  
 229: },  
 230: "body-parser": {  
 231: "version": "1.18.2",  
 232: "resolved": "https://registry.npmjs.org/body-parser/-/body-parser-1.18.2.tgz",  
 233: "integrity": "sha1-h2eKGdhLR9hZuDGZvVm84iKxBFQ=",  
 234: "dev": true,  
 235: "requires": {  
 236: "bytes": "3.0.0",  
 237: "content-type": "1.0.4",  
 238: "debug": "2.6.9",  
 239: "depd": "1.1.2",  
 240: "http-errors": "1.6.3",  
 241: "iconv-lite": "0.4.19",  
 242: "on-finished": "2.3.0",  
 243: "qs": "6.5.1",  
 244: "raw-body": "2.3.2",  
 245: "type-is": "1.6.16"  
 246: },  
 247: "dependencies": {  
 248: "qs": {  
 249: "version": "6.5.1",  
 250: "resolved": "https://registry.npmjs.org/qs/-/qs-6.5.1.tgz",  
 251: "integrity": "sha512-eRzhrN1WSINYCDCbrz796z37LOe3m5tmW7RQf6oBntukAG1nmovJvhnwHHRMAfeoItc1m2Hk02WER2aQ/iqs+A==",  
 252: "dev": true  
 253: }  
 254: }  
 255: },  
 256: "boolbase": {  
 257: "version": "1.0.0",  
 258: "resolved": "https://registry.npmjs.org/boolbase/-/boolbase-1.0.0.tgz",  
 259: "integrity": "sha1-aN/1++YMUes3cl6p4+0xDcwed24=",  
 260: "dev": true  
 261: },  
 262: "boom": {  
 263: "version": "0.4.2",  
 264: "resolved": "https://registry.npmjs.org/boom/-/boom-0.4.2.tgz",  
 265: "integrity": "sha1-emNune1O/O+xnO9JR6PGffrukRs=",  
 266: "dev": true,  
 267: "requires": {  
 268: "hoek": "0.9.1"  
 269: }  
 270: },  
 271: "brace-expansion": {  
 272: "version": "1.1.11",  
 273: "resolved": "https://registry.npmjs.org/brace-expansion/-/brace-expansion-1.1.11.tgz",  
 274: "integrity": "sha512-iCuPHDFgrHX7H2vEI/5xpz07zSHB00TpugqhmYtVmMO6518mCuRMoOYFldEBl0g187ufozdaHgWKcYFb61qGiA==",  
 275: "dev": true,  
 276: "requires": {  
 277: "balanced-match": "1.0.0",  
 278: "concat-map": "0.0.1"  
 279: }  
 280: },  
 281: "bytes": {  
 282: "version": "3.0.0",  
 283: "resolved": "https://registry.npmjs.org/bytes/-/bytes-3.0.0.tgz",  
 284: "integrity": "sha1-0ygVQE1olpn4Wk6k+odV3ROpYEg=",  
 285: "dev": true  
 286: },  
 287: "caseless": {  
 288: "version": "0.12.0",  
 289: "resolved": "https://registry.npmjs.org/caseless/-/caseless-0.12.0.tgz",  
 290: "integrity": "sha1-G2gcIf+EAzyCZUMJBolCDRhxUdw=",  
 291: "dev": true,  
 292: "optional": true  
 293: },  
 294: "catharsis": {  
 295: "version": "0.8.9",  
 296: "resolved": "https://registry.npmjs.org/catharsis/-/catharsis-0.8.9.tgz",  
 297: "integrity": "sha1-mMyJDKZS3S7w5ws3klMQ/56Q/Is=",  
 298: "dev": true,  
 299: "requires": {  
 300: "underscore-contrib": "0.3.0"  
 301: }  
 302: },  
 303: "chalk": {  
 304: "version": "1.1.3",  
 305: "resolved": "https://registry.npmjs.org/chalk/-/chalk-1.1.3.tgz",  
 306: "integrity": "sha1-qBFcVeSnAv5NFQq9OHKCKn4J/Jg=",  
 307: "dev": true,  
 308: "requires": {  
 309: "ansi-styles": "2.2.1",  
 310: "escape-string-regexp": "1.0.5",  
 311: "has-ansi": "2.0.0",  
 312: "strip-ansi": "3.0.1",  
 313: "supports-color": "2.0.0"  
 314: }  
 315: },  
 316: "cheerio": {  
 317: "version": "1.0.0-rc.2",  
 318: "resolved": "https://registry.npmjs.org/cheerio/-/cheerio-1.0.0-rc.2.tgz",  
 319: "integrity": "sha1-S59TqBsn5NXawxwP/Qz6A8xoMNs=",  
 320: "dev": true,  
 321: "requires": {  
 322: "css-select": "1.2.0",  
 323: "dom-serializer": "0.1.0",  
 324: "entities": "1.1.1",  
 325: "htmlparser2": "3.9.2",  
 326: "lodash": "4.17.10",  
 327: "parse5": "3.0.3"  
 328: }  
 329: },  
 330: "co": {  
 331: "version": "4.6.0",  
 332: "resolved": "https://registry.npmjs.org/co/-/co-4.6.0.tgz",  
 333: "integrity": "sha1-bqa989hTrlTMuOR7+gvz+QMfsYQ=",  
 334: "dev": true,  
 335: "optional": true  
 336: },  
 337: "color-logger": {  
 338: "version": "0.0.6",  
 339: "resolved": "https://registry.npmjs.org/color-logger/-/color-logger-0.0.6.tgz",  
 340: "integrity": "sha1-5WJF7ymCJlcRDHy3WpzXhstp7Rs=",  
 341: "dev": true  
 342: },  
 343: "combined-stream": {  
 344: "version": "1.0.6",  
 345: "resolved": "https://registry.npmjs.org/combined-stream/-/combined-stream-1.0.6.tgz",  
 346: "integrity": "sha1-cj599ugBrFYTETp+RFqbactjKBg=",  
 347: "dev": true,  
 348: "requires": {  
 349: "delayed-stream": "1.0.0"  
 350: }  
 351: },  
 352: "concat-map": {  
 353: "version": "0.0.1",  
 354: "resolved": "https://registry.npmjs.org/concat-map/-/concat-map-0.0.1.tgz",  
 355: "integrity": "sha1-2Klr13/Wjfd5OnMDajug1UBdR3s=",  
 356: "dev": true  
 357: },  
 358: "content-disposition": {  
 359: "version": "0.5.2",  
 360: "resolved": "https://registry.npmjs.org/content-disposition/-/content-disposition-0.5.2.tgz",  
 361: "integrity": "sha1-DPaLud318r55YcOoUXjLhdunjLQ=",  
 362: "dev": true  
 363: },  
 364: "content-type": {  
 365: "version": "1.0.4",  
 366: "resolved": "https://registry.npmjs.org/content-type/-/content-type-1.0.4.tgz",  
 367: "integrity": "sha512-hIP3EEPs8tB9AT1L+NUqtwOAps4mk2Zob89MWXMHjHWg9milF/j4osnnQLXBCBFBk/tvIG/tUc9mOUJiPBhPXA==",  
 368: "dev": true  
 369: },  
 370: "cookie": {  
 371: "version": "0.3.1",  
 372: "resolved": "https://registry.npmjs.org/cookie/-/cookie-0.3.1.tgz",  
 373: "integrity": "sha1-5+Ch+e9DtMi6klxcWpboBtFoc7s=",  
 374: "dev": true  
 375: },  
 376: "cookie-signature": {  
 377: "version": "1.0.6",  
 378: "resolved": "https://registry.npmjs.org/cookie-signature/-/cookie-signature-1.0.6.tgz",  
 379: "integrity": "sha1-4wOogrNCzD7oylE6eZmXNNqzriw=",  
 380: "dev": true  
 381: },  
 382: "core-js": {  
 383: "version": "2.5.7",  
 384: "resolved": "https://registry.npmjs.org/core-js/-/core-js-2.5.7.tgz",  
 385: "integrity": "sha512-RszJCAxg/PP6uzXVXL6BsxSXx/B05oJAQ2vkJRjyjrEcNVycaqOmNb5OTxZPE3xa5gwZduqza6L9JOCenh/Ecw==",  
 386: "dev": true  
 387: },  
 388: "core-util-is": {  
 389: "version": "1.0.2",  
 390: "resolved": "https://registry.npmjs.org/core-util-is/-/core-util-is-1.0.2.tgz",  
 391: "integrity": "sha1-tf1UIgqivFq1eqtxQMlAdUUDwac=",  
 392: "dev": true  
 393: },  
 394: "cryptiles": {  
 395: "version": "0.2.2",  
 396: "resolved": "https://registry.npmjs.org/cryptiles/-/cryptiles-0.2.2.tgz",  
 397: "integrity": "sha1-7ZH/HxetE9N0gohZT4pIoNJvMlw=",  
 398: "dev": true,  
 399: "optional": true,  
 400: "requires": {  
 401: "boom": "0.4.2"  
 402: }  
 403: },  
 404: "css-select": {  
 405: "version": "1.2.0",  
 406: "resolved": "https://registry.npmjs.org/css-select/-/css-select-1.2.0.tgz",  
 407: "integrity": "sha1-KzoRBTnFNV8c2NMUYj6HCxIeyFg=",  
 408: "dev": true,  
 409: "requires": {  
 410: "boolbase": "1.0.0",  
 411: "css-what": "2.1.0",  
 412: "domutils": "1.5.1",  
 413: "nth-check": "1.0.1"  
 414: }  
 415: },  
 416: "css-what": {  
 417: "version": "2.1.0",  
 418: "resolved": "https://registry.npmjs.org/css-what/-/css-what-2.1.0.tgz",  
 419: "integrity": "sha1-lGfQMsOM+u+58teVASUwYvh/ob0=",  
 420: "dev": true  
 421: },  
 422: "cssom": {  
 423: "version": "0.3.2",  
 424: "resolved": "https://registry.npmjs.org/cssom/-/cssom-0.3.2.tgz",  
 425: "integrity": "sha1-uANhcMefB6kP8vFuIihAJ6JDhIs=",  
 426: "dev": true  
 427: },  
 428: "cssstyle": {  
 429: "version": "0.2.37",  
 430: "resolved": "https://registry.npmjs.org/cssstyle/-/cssstyle-0.2.37.tgz",  
 431: "integrity": "sha1-VBCXI0yyUTyDzu06zdwn/yeYfVQ=",  
 432: "dev": true,  
 433: "optional": true,  
 434: "requires": {  
 435: "cssom": "0.3.2"  
 436: }  
 437: },  
 438: "ctype": {  
 439: "version": "0.5.3",  
 440: "resolved": "https://registry.npmjs.org/ctype/-/ctype-0.5.3.tgz",  
 441: "integrity": "sha1-gsGMJGH3QRTvFsE1IkrQuRRMoS8=",  
 442: "dev": true,  
 443: "optional": true  
 444: },  
 445: "dashdash": {  
 446: "version": "1.14.1",  
 447: "resolved": "https://registry.npmjs.org/dashdash/-/dashdash-1.14.1.tgz",  
 448: "integrity": "sha1-hTz6D3y+L+1d4gMmuN1YEDX24vA=",  
 449: "dev": true,  
 450: "optional": true,  
 451: "requires": {  
 452: "assert-plus": "1.0.0"  
 453: }  
 454: },  
 455: "debug": {  
 456: "version": "2.6.9",  
 457: "resolved": "https://registry.npmjs.org/debug/-/debug-2.6.9.tgz",  
 458: "integrity": "sha512-bC7ElrdJaJnPbAP+1EotYvqZsb3ecl5wi6Bfi6BJTUcNowp6cvspg0jXznRTKDjm/E7AdgFBVeAPVMNcKGsHMA==",  
 459: "dev": true,  
 460: "requires": {  
 461: "ms": "2.0.0"  
 462: }  
 463: },  
 464: "deep-is": {  
 465: "version": "0.1.3",  
 466: "resolved": "https://registry.npmjs.org/deep-is/-/deep-is-0.1.3.tgz",  
 467: "integrity": "sha1-s2nW+128E+7PUk+RsHD+7cNXzzQ=",  
 468: "dev": true,  
 469: "optional": true  
 470: },  
 471: "delayed-stream": {  
 472: "version": "1.0.0",  
 473: "resolved": "https://registry.npmjs.org/delayed-stream/-/delayed-stream-1.0.0.tgz",  
 474: "integrity": "sha1-3zrhmayt+31ECqrgsp4icrJOxhk=",  
 475: "dev": true  
 476: },  
 477: "depd": {  
 478: "version": "1.1.2",  
 479: "resolved": "https://registry.npmjs.org/depd/-/depd-1.1.2.tgz",  
 480: "integrity": "sha1-m81S4UwJd2PnSbJ0xDRu0uVgtak=",  
 481: "dev": true  
 482: },  
 483: "destroy": {  
 484: "version": "1.0.4",  
 485: "resolved": "https://registry.npmjs.org/destroy/-/destroy-1.0.4.tgz",  
 486: "integrity": "sha1-l4hXRCxEdJ5CBmE+N5RiBYJqvYA=",  
 487: "dev": true  
 488: },  
 489: "detect-indent": {  
 490: "version": "4.0.0",  
 491: "resolved": "https://registry.npmjs.org/detect-indent/-/detect-indent-4.0.0.tgz",  
 492: "integrity": "sha1-920GQ1LN9Docts5hnE7jqUdd4gg=",  
 493: "dev": true,  
 494: "requires": {  
 495: "repeating": "2.0.1"  
 496: }  
 497: },  
 498: "dom-serializer": {  
 499: "version": "0.1.0",  
 500: "resolved": "https://registry.npmjs.org/dom-serializer/-/dom-serializer-0.1.0.tgz",  
 501: "integrity": "sha1-BzxpdUbOB4DOI75KKOKT5AvDDII=",  
 502: "dev": true,  
 503: "requires": {  
 504: "domelementtype": "1.1.3",  
 505: "entities": "1.1.1"  
 506: },  
 507: "dependencies": {  
 508: "domelementtype": {  
 509: "version": "1.1.3",  
 510: "resolved": "https://registry.npmjs.org/domelementtype/-/domelementtype-1.1.3.tgz",  
 511: "integrity": "sha1-vSh3PiZCiBrsUVRJJCmcXNgiGFs=",  
 512: "dev": true  
 513: }  
 514: }  
 515: },  
 516: "domelementtype": {  
 517: "version": "1.3.0",  
 518: "resolved": "https://registry.npmjs.org/domelementtype/-/domelementtype-1.3.0.tgz",  
 519: "integrity": "sha1-sXrtguirWeUt2cGbF1bg/BhyBMI=",  
 520: "dev": true  
 521: },  
 522: "domhandler": {  
 523: "version": "2.4.2",  
 524: "resolved": "https://registry.npmjs.org/domhandler/-/domhandler-2.4.2.tgz",  
 525: "integrity": "sha512-JiK04h0Ht5u/80fdLMCEmV4zkNh2BcoMFBmZ/91WtYZ8qVXSKjiw7fXMgFPnHcSZgOo3XdinHvmnDUeMf5R4wA==",  
 526: "dev": true,  
 527: "requires": {  
 528: "domelementtype": "1.3.0"  
 529: }  
 530: },  
 531: "domutils": {  
 532: "version": "1.5.1",  
 533: "resolved": "https://registry.npmjs.org/domutils/-/domutils-1.5.1.tgz",  
 534: "integrity": "sha1-3NhIiib1Y9YQeeSMn3t+Mjc2gs8=",  
 535: "dev": true,  
 536: "requires": {  
 537: "dom-serializer": "0.1.0",  
 538: "domelementtype": "1.3.0"  
 539: }  
 540: },  
 541: "ecc-jsbn": {  
 542: "version": "0.1.1",  
 543: "resolved": "https://registry.npmjs.org/ecc-jsbn/-/ecc-jsbn-0.1.1.tgz",  
 544: "integrity": "sha1-D8c6ntXw1Tw4GTOYUj735UN3dQU=",  
 545: "dev": true,  
 546: "optional": true,  
 547: "requires": {  
 548: "jsbn": "0.1.1"  
 549: }  
 550: },  
 551: "ee-first": {  
 552: "version": "1.1.1",  
 553: "resolved": "https://registry.npmjs.org/ee-first/-/ee-first-1.1.1.tgz",  
 554: "integrity": "sha1-WQxhFWsK4vTwJVcyoViyZrxWsh0=",  
 555: "dev": true  
 556: },  
 557: "encodeurl": {  
 558: "version": "1.0.2",  
 559: "resolved": "https://registry.npmjs.org/encodeurl/-/encodeurl-1.0.2.tgz",  
 560: "integrity": "sha1-rT/0yG7C0CkyL1oCw6mmBslbP1k=",  
 561: "dev": true  
 562: },  
 563: "entities": {  
 564: "version": "1.1.1",  
 565: "resolved": "https://registry.npmjs.org/entities/-/entities-1.1.1.tgz",  
 566: "integrity": "sha1-blwtClYhtdra7O+AuQ7ftc13cvA=",  
 567: "dev": true  
 568: },  
 569: "escape-html": {  
 570: "version": "1.0.3",  
 571: "resolved": "https://registry.npmjs.org/escape-html/-/escape-html-1.0.3.tgz",  
 572: "integrity": "sha1-Aljq5NPQwJdN4cFpGI7wBR0dGYg=",  
 573: "dev": true  
 574: },  
 575: "escape-string-regexp": {  
 576: "version": "1.0.5",  
 577: "resolved": "https://registry.npmjs.org/escape-string-regexp/-/escape-string-regexp-1.0.5.tgz",  
 578: "integrity": "sha1-G2HAViGQqN/2rjuyzwIAyhMLhtQ=",  
 579: "dev": true  
 580: },  
 581: "escodegen": {  
 582: "version": "1.9.1",  
 583: "resolved": "https://registry.npmjs.org/escodegen/-/escodegen-1.9.1.tgz",  
 584: "integrity": "sha512-6hTjO1NAWkHnDk3OqQ4YrCuwwmGHL9S3nPlzBOUG/R44rda3wLNrfvQ5fkSGjyhHFKM7ALPKcKGrwvCLe0lC7Q==",  
 585: "dev": true,  
 586: "optional": true,  
 587: "requires": {  
 588: "esprima": "3.1.3",  
 589: "estraverse": "4.2.0",  
 590: "esutils": "2.0.2",  
 591: "optionator": "0.8.2",  
 592: "source-map": "0.6.1"  
 593: },  
 594: "dependencies": {  
 595: "source-map": {  
 596: "version": "0.6.1",  
 597: "resolved": "https://registry.npmjs.org/source-map/-/source-map-0.6.1.tgz",  
 598: "integrity": "sha512-UjgapumWlbMhkBgzT7Ykc5YXUT46F0iKu8SGXq0bcwP5dz/h0Plj6enJqjz1Zbq2l5WaqYnrVbwWOWMyF3F47g==",  
 599: "dev": true,  
 600: "optional": true  
 601: }  
 602: }  
 603: },  
 604: "esdoc": {  
 605: "version": "1.1.0",  
 606: "resolved": "https://registry.npmjs.org/esdoc/-/esdoc-1.1.0.tgz",  
 607: "integrity": "sha512-vsUcp52XJkOWg9m1vDYplGZN2iDzvmjDL5M/Mp8qkoDG3p2s0yIQCIjKR5wfPBaM3eV14a6zhQNYiNTCVzPnxA==",  
 608: "dev": true,  
 609: "requires": {  
 610: "babel-generator": "6.26.1",  
 611: "babel-traverse": "6.26.0",  
 612: "babylon": "6.18.0",  
 613: "cheerio": "1.0.0-rc.2",  
 614: "color-logger": "0.0.6",  
 615: "escape-html": "1.0.3",  
 616: "fs-extra": "5.0.0",  
 617: "ice-cap": "0.0.4",  
 618: "marked": "0.3.19",  
 619: "minimist": "1.2.0",  
 620: "taffydb": "2.7.3"  
 621: }  
 622: },  
 623: "esdoc-accessor-plugin": {  
 624: "version": "1.0.0",  
 625: "resolved": "https://registry.npmjs.org/esdoc-accessor-plugin/-/esdoc-accessor-plugin-1.0.0.tgz",  
 626: "integrity": "sha1-eRukhy5sQDUVznSbE0jW8Ck62es=",  
 627: "dev": true  
 628: },  
 629: "esdoc-brand-plugin": {  
 630: "version": "1.0.1",  
 631: "resolved": "https://registry.npmjs.org/esdoc-brand-plugin/-/esdoc-brand-plugin-1.0.1.tgz",  
 632: "integrity": "sha512-Yv9j3M7qk5PSLmSeD6MbPsfIsEf8K43EdH8qZpE/GZwnJCRVmDPrZJ1cLDj/fPu6P35YqgcEaJK4E2NL/CKA7g==",  
 633: "dev": true,  
 634: "requires": {  
 635: "cheerio": "0.22.0"  
 636: },  
 637: "dependencies": {  
 638: "cheerio": {  
 639: "version": "0.22.0",  
 640: "resolved": "https://registry.npmjs.org/cheerio/-/cheerio-0.22.0.tgz",  
 641: "integrity": "sha1-qbqoYKP5tZWmuBsahocxIe06Jp4=",  
 642: "dev": true,  
 643: "requires": {  
 644: "css-select": "1.2.0",  
 645: "dom-serializer": "0.1.0",  
 646: "entities": "1.1.1",  
 647: "htmlparser2": "3.9.2",  
 648: "lodash.assignin": "4.2.0",  
 649: "lodash.bind": "4.2.1",  
 650: "lodash.defaults": "4.2.0",  
 651: "lodash.filter": "4.6.0",  
 652: "lodash.flatten": "4.4.0",  
 653: "lodash.foreach": "4.5.0",  
 654: "lodash.map": "4.6.0",  
 655: "lodash.merge": "4.6.1",  
 656: "lodash.pick": "4.4.0",  
 657: "lodash.reduce": "4.6.0",  
 658: "lodash.reject": "4.6.0",  
 659: "lodash.some": "4.6.0"  
 660: }  
 661: }  
 662: }  
 663: },  
 664: "esdoc-coverage-plugin": {  
 665: "version": "1.1.0",  
 666: "resolved": "https://registry.npmjs.org/esdoc-coverage-plugin/-/esdoc-coverage-plugin-1.1.0.tgz",  
 667: "integrity": "sha1-OGmGnNf4eJH5cmJXh2laKZrs5Fw=",  
 668: "dev": true  
 669: },  
 670: "esdoc-external-ecmascript-plugin": {  
 671: "version": "1.0.0",  
 672: "resolved": "https://registry.npmjs.org/esdoc-external-ecmascript-plugin/-/esdoc-external-ecmascript-plugin-1.0.0.tgz",  
 673: "integrity": "sha1-ePVl1KDFGFrGMVJhTc4f4ahmiNs=",  
 674: "dev": true,  
 675: "requires": {  
 676: "fs-extra": "1.0.0"  
 677: },  
 678: "dependencies": {  
 679: "fs-extra": {  
 680: "version": "1.0.0",  
 681: "resolved": "https://registry.npmjs.org/fs-extra/-/fs-extra-1.0.0.tgz",  
 682: "integrity": "sha1-zTzl9+fLYUWIP8rjGR6Yd/hYeVA=",  
 683: "dev": true,  
 684: "requires": {  
 685: "graceful-fs": "4.1.11",  
 686: "jsonfile": "2.4.0",  
 687: "klaw": "1.3.1"  
 688: }  
 689: },  
 690: "jsonfile": {  
 691: "version": "2.4.0",  
 692: "resolved": "https://registry.npmjs.org/jsonfile/-/jsonfile-2.4.0.tgz",  
 693: "integrity": "sha1-NzaitCi4e72gzIO1P6PWM6NcKug=",  
 694: "dev": true,  
 695: "requires": {  
 696: "graceful-fs": "4.1.11"  
 697: }  
 698: }  
 699: }  
 700: },  
 701: "esdoc-integrate-manual-plugin": {  
 702: "version": "1.0.0",  
 703: "resolved": "https://registry.npmjs.org/esdoc-integrate-manual-plugin/-/esdoc-integrate-manual-plugin-1.0.0.tgz",  
 704: "integrity": "sha1-GFSmqhwIEDXXyMUeO91PtlqkcRw=",  
 705: "dev": true  
 706: },  
 707: "esdoc-integrate-test-plugin": {  
 708: "version": "1.0.0",  
 709: "resolved": "https://registry.npmjs.org/esdoc-integrate-test-plugin/-/esdoc-integrate-test-plugin-1.0.0.tgz",  
 710: "integrity": "sha1-4tDQAJD38MNeXS8sAzMnp55T5Ak=",  
 711: "dev": true  
 712: },  
 713: "esdoc-lint-plugin": {  
 714: "version": "1.0.2",  
 715: "resolved": "https://registry.npmjs.org/esdoc-lint-plugin/-/esdoc-lint-plugin-1.0.2.tgz",  
 716: "integrity": "sha512-24AYqD2WbZI9We02I7/6dzAa7yUliRTFUaJCZAcYJMQicJT5gUrNFVaI8XmWEN/mhF3szIn1uZBNWeLul4CmNw==",  
 717: "dev": true  
 718: },  
 719: "esdoc-publish-html-plugin": {  
 720: "version": "1.1.2",  
 721: "resolved": "https://registry.npmjs.org/esdoc-publish-html-plugin/-/esdoc-publish-html-plugin-1.1.2.tgz",  
 722: "integrity": "sha512-hG1fZmTcEp3P/Hv/qKiMdG1qSp8MjnVZMMkxL5P5ry7I2sX0HQ4P9lt2lms+90Lt0r340HHhSuVx107UL7dphg==",  
 723: "dev": true,  
 724: "requires": {  
 725: "babel-generator": "6.11.4",  
 726: "cheerio": "0.22.0",  
 727: "escape-html": "1.0.3",  
 728: "fs-extra": "1.0.0",  
 729: "ice-cap": "0.0.4",  
 730: "marked": "0.3.19",  
 731: "taffydb": "2.7.2"  
 732: },  
 733: "dependencies": {  
 734: "babel-generator": {  
 735: "version": "6.11.4",  
 736: "resolved": "https://registry.npmjs.org/babel-generator/-/babel-generator-6.11.4.tgz",  
 737: "integrity": "sha1-FPaTOrsgxiZm0n47e59bncBxKpo=",  
 738: "dev": true,  
 739: "requires": {  
 740: "babel-messages": "6.23.0",  
 741: "babel-runtime": "6.26.0",  
 742: "babel-types": "6.26.0",  
 743: "detect-indent": "3.0.1",  
 744: "lodash": "4.17.10",  
 745: "source-map": "0.5.7"  
 746: }  
 747: },  
 748: "cheerio": {  
 749: "version": "0.22.0",  
 750: "resolved": "https://registry.npmjs.org/cheerio/-/cheerio-0.22.0.tgz",  
 751: "integrity": "sha1-qbqoYKP5tZWmuBsahocxIe06Jp4=",  
 752: "dev": true,  
 753: "requires": {  
 754: "css-select": "1.2.0",  
 755: "dom-serializer": "0.1.0",  
 756: "entities": "1.1.1",  
 757: "htmlparser2": "3.9.2",  
 758: "lodash.assignin": "4.2.0",  
 759: "lodash.bind": "4.2.1",  
 760: "lodash.defaults": "4.2.0",  
 761: "lodash.filter": "4.6.0",  
 762: "lodash.flatten": "4.4.0",  
 763: "lodash.foreach": "4.5.0",  
 764: "lodash.map": "4.6.0",  
 765: "lodash.merge": "4.6.1",  
 766: "lodash.pick": "4.4.0",  
 767: "lodash.reduce": "4.6.0",  
 768: "lodash.reject": "4.6.0",  
 769: "lodash.some": "4.6.0"  
 770: }  
 771: },  
 772: "detect-indent": {  
 773: "version": "3.0.1",  
 774: "resolved": "https://registry.npmjs.org/detect-indent/-/detect-indent-3.0.1.tgz",  
 775: "integrity": "sha1-ncXl3bzu+DJXZLlFGwK8bVQIT3U=",  
 776: "dev": true,  
 777: "requires": {  
 778: "get-stdin": "4.0.1",  
 779: "minimist": "1.2.0",  
 780: "repeating": "1.1.3"  
 781: }  
 782: },  
 783: "fs-extra": {  
 784: "version": "1.0.0",  
 785: "resolved": "https://registry.npmjs.org/fs-extra/-/fs-extra-1.0.0.tgz",  
 786: "integrity": "sha1-zTzl9+fLYUWIP8rjGR6Yd/hYeVA=",  
 787: "dev": true,  
 788: "requires": {  
 789: "graceful-fs": "4.1.11",  
 790: "jsonfile": "2.4.0",  
 791: "klaw": "1.3.1"  
 792: }  
 793: },  
 794: "jsonfile": {  
 795: "version": "2.4.0",  
 796: "resolved": "https://registry.npmjs.org/jsonfile/-/jsonfile-2.4.0.tgz",  
 797: "integrity": "sha1-NzaitCi4e72gzIO1P6PWM6NcKug=",  
 798: "dev": true,  
 799: "requires": {  
 800: "graceful-fs": "4.1.11"  
 801: }  
 802: },  
 803: "repeating": {  
 804: "version": "1.1.3",  
 805: "resolved": "https://registry.npmjs.org/repeating/-/repeating-1.1.3.tgz",  
 806: "integrity": "sha1-PUEUIYh3U3SU+X93+Xhfq4EPpKw=",  
 807: "dev": true,  
 808: "requires": {  
 809: "is-finite": "1.0.2"  
 810: }  
 811: },  
 812: "taffydb": {  
 813: "version": "2.7.2",  
 814: "resolved": "https://registry.npmjs.org/taffydb/-/taffydb-2.7.2.tgz",  
 815: "integrity": "sha1-e/gQalwaSCUbPjvAoOFzJIn9Dcg=",  
 816: "dev": true  
 817: }  
 818: }  
 819: },  
 820: "esdoc-standard-plugin": {  
 821: "version": "1.0.0",  
 822: "resolved": "https://registry.npmjs.org/esdoc-standard-plugin/-/esdoc-standard-plugin-1.0.0.tgz",  
 823: "integrity": "sha1-ZhIBysfvhokkkCRG/awVJyU8XU0=",  
 824: "dev": true,  
 825: "requires": {  
 826: "esdoc-accessor-plugin": "1.0.0",  
 827: "esdoc-brand-plugin": "1.0.1",  
 828: "esdoc-coverage-plugin": "1.1.0",  
 829: "esdoc-external-ecmascript-plugin": "1.0.0",  
 830: "esdoc-integrate-manual-plugin": "1.0.0",  
 831: "esdoc-integrate-test-plugin": "1.0.0",  
 832: "esdoc-lint-plugin": "1.0.2",  
 833: "esdoc-publish-html-plugin": "1.1.2",  
 834: "esdoc-type-inference-plugin": "1.0.2",  
 835: "esdoc-undocumented-identifier-plugin": "1.0.0",  
 836: "esdoc-unexported-identifier-plugin": "1.0.0"  
 837: }  
 838: },  
 839: "esdoc-type-inference-plugin": {  
 840: "version": "1.0.2",  
 841: "resolved": "https://registry.npmjs.org/esdoc-type-inference-plugin/-/esdoc-type-inference-plugin-1.0.2.tgz",  
 842: "integrity": "sha512-tMIcEHNe1uhUGA7lT1UTWc9hs2dzthnTgmqXpmeUhurk7fL2tinvoH+IVvG/sLROzwOGZQS9zW/F9KWnpMzLIQ==",  
 843: "dev": true  
 844: },  
 845: "esdoc-undocumented-identifier-plugin": {  
 846: "version": "1.0.0",  
 847: "resolved": "https://registry.npmjs.org/esdoc-undocumented-identifier-plugin/-/esdoc-undocumented-identifier-plugin-1.0.0.tgz",  
 848: "integrity": "sha1-guBdNxwy0ShxFA8dXIHsmf2cwsg=",  
 849: "dev": true  
 850: },  
 851: "esdoc-unexported-identifier-plugin": {  
 852: "version": "1.0.0",  
 853: "resolved": "https://registry.npmjs.org/esdoc-unexported-identifier-plugin/-/esdoc-unexported-identifier-plugin-1.0.0.tgz",  
 854: "integrity": "sha1-H5h0xqfCvr+a05fDzrdcnGnaurE=",  
 855: "dev": true  
 856: },  
 857: "esprima": {  
 858: "version": "3.1.3",  
 859: "resolved": "https://registry.npmjs.org/esprima/-/esprima-3.1.3.tgz",  
 860: "integrity": "sha1-/cpRzuYTOJXjyI1TXOSdv/YqRjM=",  
 861: "dev": true,  
 862: "optional": true  
 863: },  
 864: "estraverse": {  
 865: "version": "4.2.0",  
 866: "resolved": "https://registry.npmjs.org/estraverse/-/estraverse-4.2.0.tgz",  
 867: "integrity": "sha1-De4/7TH81GlhjOc0IJn8GvoL2xM=",  
 868: "dev": true,  
 869: "optional": true  
 870: },  
 871: "esutils": {  
 872: "version": "2.0.2",  
 873: "resolved": "https://registry.npmjs.org/esutils/-/esutils-2.0.2.tgz",  
 874: "integrity": "sha1-Cr9PHKpbyx96nYrMbepPqqBLrJs=",  
 875: "dev": true  
 876: },  
 877: "etag": {  
 878: "version": "1.8.1",  
 879: "resolved": "https://registry.npmjs.org/etag/-/etag-1.8.1.tgz",  
 880: "integrity": "sha1-Qa4u62XvpiJorr/qg6x9eSmbCIc=",  
 881: "dev": true  
 882: },  
 883: "express": {  
 884: "version": "4.16.3",  
 885: "resolved": "https://registry.npmjs.org/express/-/express-4.16.3.tgz",  
 886: "integrity": "sha1-avilAjUNsyRuzEvs9rWjTSL37VM=",  
 887: "dev": true,  
 888: "requires": {  
 889: "accepts": "1.3.5",  
 890: "array-flatten": "1.1.1",  
 891: "body-parser": "1.18.2",  
 892: "content-disposition": "0.5.2",  
 893: "content-type": "1.0.4",  
 894: "cookie": "0.3.1",  
 895: "cookie-signature": "1.0.6",  
 896: "debug": "2.6.9",  
 897: "depd": "1.1.2",  
 898: "encodeurl": "1.0.2",  
 899: "escape-html": "1.0.3",  
 900: "etag": "1.8.1",  
 901: "finalhandler": "1.1.1",  
 902: "fresh": "0.5.2",  
 903: "merge-descriptors": "1.0.1",  
 904: "methods": "1.1.2",  
 905: "on-finished": "2.3.0",  
 906: "parseurl": "1.3.2",  
 907: "path-to-regexp": "0.1.7",  
 908: "proxy-addr": "2.0.3",  
 909: "qs": "6.5.1",  
 910: "range-parser": "1.2.0",  
 911: "safe-buffer": "5.1.1",  
 912: "send": "0.16.2",  
 913: "serve-static": "1.13.2",  
 914: "setprototypeof": "1.1.0",  
 915: "statuses": "1.4.0",  
 916: "type-is": "1.6.16",  
 917: "utils-merge": "1.0.1",  
 918: "vary": "1.1.2"  
 919: },  
 920: "dependencies": {  
 921: "qs": {  
 922: "version": "6.5.1",  
 923: "resolved": "https://registry.npmjs.org/qs/-/qs-6.5.1.tgz",  
 924: "integrity": "sha512-eRzhrN1WSINYCDCbrz796z37LOe3m5tmW7RQf6oBntukAG1nmovJvhnwHHRMAfeoItc1m2Hk02WER2aQ/iqs+A==",  
 925: "dev": true  
 926: },  
 927: "safe-buffer": {  
 928: "version": "5.1.1",  
 929: "resolved": "https://registry.npmjs.org/safe-buffer/-/safe-buffer-5.1.1.tgz",  
 930: "integrity": "sha512-kKvNJn6Mm93gAczWVJg7wH+wGYWNrDHdWvpUmHyEsgCtIwwo3bqPtV4tR5tuPaUhTOo/kvhVwd8XwwOllGYkbg==",  
 931: "dev": true  
 932: }  
 933: }  
 934: },  
 935: "extend": {  
 936: "version": "3.0.1",  
 937: "resolved": "https://registry.npmjs.org/extend/-/extend-3.0.1.tgz",  
 938: "integrity": "sha1-p1Xqe8Gt/MWjHOfnYtuq3F5jZEQ=",  
 939: "dev": true,  
 940: "optional": true  
 941: },  
 942: "extsprintf": {  
 943: "version": "1.3.0",  
 944: "resolved": "https://registry.npmjs.org/extsprintf/-/extsprintf-1.3.0.tgz",  
 945: "integrity": "sha1-lpGEQOMEGnpBT4xS48V06zw+HgU=",  
 946: "dev": true  
 947: },  
 948: "fast-deep-equal": {  
 949: "version": "1.1.0",  
 950: "resolved": "https://registry.npmjs.org/fast-deep-equal/-/fast-deep-equal-1.1.0.tgz",  
 951: "integrity": "sha1-wFNHeBfIa1HaqFPIHgWbcz0CNhQ=",  
 952: "dev": true,  
 953: "optional": true  
 954: },  
 955: "fast-json-stable-stringify": {  
 956: "version": "2.0.0",  
 957: "resolved": "https://registry.npmjs.org/fast-json-stable-stringify/-/fast-json-stable-stringify-2.0.0.tgz",  
 958: "integrity": "sha1-1RQsDK7msRifh9OnYREGT4bIu/I=",  
 959: "dev": true,  
 960: "optional": true  
 961: },  
 962: "fast-levenshtein": {  
 963: "version": "2.0.6",  
 964: "resolved": "https://registry.npmjs.org/fast-levenshtein/-/fast-levenshtein-2.0.6.tgz",  
 965: "integrity": "sha1-PYpcZog6FqMMqGQ+hR8Zuqd5eRc=",  
 966: "dev": true,  
 967: "optional": true  
 968: },  
 969: "finalhandler": {  
 970: "version": "1.1.1",  
 971: "resolved": "https://registry.npmjs.org/finalhandler/-/finalhandler-1.1.1.tgz",  
 972: "integrity": "sha512-Y1GUDo39ez4aHAw7MysnUD5JzYX+WaIj8I57kO3aEPT1fFRL4sr7mjei97FgnwhAyyzRYmQZaTHb2+9uZ1dPtg==",  
 973: "dev": true,  
 974: "requires": {  
 975: "debug": "2.6.9",  
 976: "encodeurl": "1.0.2",  
 977: "escape-html": "1.0.3",  
 978: "on-finished": "2.3.0",  
 979: "parseurl": "1.3.2",  
 980: "statuses": "1.4.0",  
 981: "unpipe": "1.0.0"  
 982: }  
 983: },  
 984: "forever-agent": {  
 985: "version": "0.6.1",  
 986: "resolved": "https://registry.npmjs.org/forever-agent/-/forever-agent-0.6.1.tgz",  
 987: "integrity": "sha1-+8cfDEGt6zf5bFd60e1C2P2sypE=",  
 988: "dev": true,  
 989: "optional": true  
 990: },  
 991: "form-data": {  
 992: "version": "2.3.2",  
 993: "resolved": "https://registry.npmjs.org/form-data/-/form-data-2.3.2.tgz",  
 994: "integrity": "sha1-SXBJi+YEwgwAXU9cI67NIda0kJk=",  
 995: "dev": true,  
 996: "optional": true,  
 997: "requires": {  
 998: "asynckit": "0.4.0",  
 999: "combined-stream": "1.0.6",  
1000: "mime-types": "2.1.18"  
1001: }  
1002: },  
1003: "forwarded": {  
1004: "version": "0.1.2",  
1005: "resolved": "https://registry.npmjs.org/forwarded/-/forwarded-0.1.2.tgz",  
1006: "integrity": "sha1-mMI9qxF1ZXuMBXPozszZGw/xjIQ=",  
1007: "dev": true  
1008: },  
1009: "fresh": {  
1010: "version": "0.5.2",  
1011: "resolved": "https://registry.npmjs.org/fresh/-/fresh-0.5.2.tgz",  
1012: "integrity": "sha1-PYyt2Q2XZWn6g1qx+OSyOhBWBac=",  
1013: "dev": true  
1014: },  
1015: "fs-extra": {  
1016: "version": "5.0.0",  
1017: "resolved": "https://registry.npmjs.org/fs-extra/-/fs-extra-5.0.0.tgz",  
1018: "integrity": "sha512-66Pm4RYbjzdyeuqudYqhFiNBbCIuI9kgRqLPSHIlXHidW8NIQtVdkM1yeZ4lXwuhbTETv3EUGMNHAAw6hiundQ==",  
1019: "dev": true,  
1020: "requires": {  
1021: "graceful-fs": "4.1.11",  
1022: "jsonfile": "4.0.0",  
1023: "universalify": "0.1.1"  
1024: }  
1025: },  
1026: "fs.realpath": {  
1027: "version": "1.0.0",  
1028: "resolved": "https://registry.npmjs.org/fs.realpath/-/fs.realpath-1.0.0.tgz",  
1029: "integrity": "sha1-FQStJSMVjKpA20onh8sBQRmU6k8=",  
1030: "dev": true  
1031: },  
1032: "get-stdin": {  
1033: "version": "4.0.1",  
1034: "resolved": "https://registry.npmjs.org/get-stdin/-/get-stdin-4.0.1.tgz",  
1035: "integrity": "sha1-uWjGsKBDhDJJAui/Gl3zJXmkUP4=",  
1036: "dev": true  
1037: },  
1038: "getpass": {  
1039: "version": "0.1.7",  
1040: "resolved": "https://registry.npmjs.org/getpass/-/getpass-0.1.7.tgz",  
1041: "integrity": "sha1-Xv+OPmhNVprkyysSgmBOi6YhSfo=",  
1042: "dev": true,  
1043: "optional": true,  
1044: "requires": {  
1045: "assert-plus": "1.0.0"  
1046: }  
1047: },  
1048: "glob": {  
1049: "version": "7.1.2",  
1050: "resolved": "https://registry.npmjs.org/glob/-/glob-7.1.2.tgz",  
1051: "integrity": "sha512-MJTUg1kjuLeQCJ+ccE4Vpa6kKVXkPYJ2mOCQyUuKLcLQsdrMCpBPUi8qVE6+YuaJkozeA9NusTAw3hLr8Xe5EQ==",  
1052: "dev": true,  
1053: "requires": {  
1054: "fs.realpath": "1.0.0",  
1055: "inflight": "1.0.6",  
1056: "inherits": "2.0.3",  
1057: "minimatch": "3.0.4",  
1058: "once": "1.4.0",  
1059: "path-is-absolute": "1.0.1"  
1060: }  
1061: },  
1062: "globals": {  
1063: "version": "9.18.0",  
1064: "resolved": "https://registry.npmjs.org/globals/-/globals-9.18.0.tgz",  
1065: "integrity": "sha512-S0nG3CLEQiY/ILxqtztTWH/3iRRdyBLw6KMDxnKMchrtbj2OFmehVh0WUCfW3DUrIgx/qFrJPICrq4Z4sTR9UQ==",  
1066: "dev": true  
1067: },  
1068: "graceful-fs": {  
1069: "version": "4.1.11",  
1070: "resolved": "https://registry.npmjs.org/graceful-fs/-/graceful-fs-4.1.11.tgz",  
1071: "integrity": "sha1-Dovf5NHduIVNZOBOp8AOKgJuVlg=",  
1072: "dev": true  
1073: },  
1074: "har-schema": {  
1075: "version": "2.0.0",  
1076: "resolved": "https://registry.npmjs.org/har-schema/-/har-schema-2.0.0.tgz",  
1077: "integrity": "sha1-qUwiJOvKwEeCoNkDVSHyRzW37JI=",  
1078: "dev": true,  
1079: "optional": true  
1080: },  
1081: "har-validator": {  
1082: "version": "5.0.3",  
1083: "resolved": "https://registry.npmjs.org/har-validator/-/har-validator-5.0.3.tgz",  
1084: "integrity": "sha1-ukAsJmGU8VlW7xXg/PJCmT9qff0=",  
1085: "dev": true,  
1086: "optional": true,  
1087: "requires": {  
1088: "ajv": "5.5.2",  
1089: "har-schema": "2.0.0"  
1090: }  
1091: },  
1092: "has-ansi": {  
1093: "version": "2.0.0",  
1094: "resolved": "https://registry.npmjs.org/has-ansi/-/has-ansi-2.0.0.tgz",  
1095: "integrity": "sha1-NPUEnOHs3ysGSa8+8k5F7TVBbZE=",  
1096: "dev": true,  
1097: "requires": {  
1098: "ansi-regex": "2.1.1"  
1099: }  
1100: },  
1101: "hawk": {  
1102: "version": "1.1.1",  
1103: "resolved": "https://registry.npmjs.org/hawk/-/hawk-1.1.1.tgz",  
1104: "integrity": "sha1-h81JH5tG5OKurKM1QWdmiF0tHtk=",  
1105: "dev": true,  
1106: "optional": true,  
1107: "requires": {  
1108: "boom": "0.4.2",  
1109: "cryptiles": "0.2.2",  
1110: "hoek": "0.9.1",  
1111: "sntp": "0.2.4"  
1112: }  
1113: },  
1114: "hoek": {  
1115: "version": "0.9.1",  
1116: "resolved": "https://registry.npmjs.org/hoek/-/hoek-0.9.1.tgz",  
1117: "integrity": "sha1-PTIkYrrfB3Fup+uFuviAec3c5QU=",  
1118: "dev": true  
1119: },  
1120: "htmlparser2": {  
1121: "version": "3.9.2",  
1122: "resolved": "https://registry.npmjs.org/htmlparser2/-/htmlparser2-3.9.2.tgz",  
1123: "integrity": "sha1-G9+HrMoPP55T+k/M6w9LTLsAszg=",  
1124: "dev": true,  
1125: "requires": {  
1126: "domelementtype": "1.3.0",  
1127: "domhandler": "2.4.2",  
1128: "domutils": "1.5.1",  
1129: "entities": "1.1.1",  
1130: "inherits": "2.0.3",  
1131: "readable-stream": "2.3.6"  
1132: }  
1133: },  
1134: "http-errors": {  
1135: "version": "1.6.3",  
1136: "resolved": "https://registry.npmjs.org/http-errors/-/http-errors-1.6.3.tgz",  
1137: "integrity": "sha1-i1VoC7S+KDoLW/TqLjhYC+HZMg0=",  
1138: "dev": true,  
1139: "requires": {  
1140: "depd": "1.1.2",  
1141: "inherits": "2.0.3",  
1142: "setprototypeof": "1.1.0",  
1143: "statuses": "1.4.0"  
1144: }  
1145: },  
1146: "http-signature": {  
1147: "version": "1.2.0",  
1148: "resolved": "https://registry.npmjs.org/http-signature/-/http-signature-1.2.0.tgz",  
1149: "integrity": "sha1-muzZJRFHcvPZW2WmCruPfBj7rOE=",  
1150: "dev": true,  
1151: "optional": true,  
1152: "requires": {  
1153: "assert-plus": "1.0.0",  
1154: "jsprim": "1.4.1",  
1155: "sshpk": "1.14.1"  
1156: }  
1157: },  
1158: "ice-cap": {  
1159: "version": "0.0.4",  
1160: "resolved": "https://registry.npmjs.org/ice-cap/-/ice-cap-0.0.4.tgz",  
1161: "integrity": "sha1-im0xq0ysjUtW3k+pRt8zUlYbbhg=",  
1162: "dev": true,  
1163: "requires": {  
1164: "cheerio": "0.20.0",  
1165: "color-logger": "0.0.3"  
1166: },  
1167: "dependencies": {  
1168: "cheerio": {  
1169: "version": "0.20.0",  
1170: "resolved": "https://registry.npmjs.org/cheerio/-/cheerio-0.20.0.tgz",  
1171: "integrity": "sha1-XHEPK6uVZTJyhCugHG6mGzVF7DU=",  
1172: "dev": true,  
1173: "requires": {  
1174: "css-select": "1.2.0",  
1175: "dom-serializer": "0.1.0",  
1176: "entities": "1.1.1",  
1177: "htmlparser2": "3.8.3",  
1178: "jsdom": "7.2.2",  
1179: "lodash": "4.17.10"  
1180: }  
1181: },  
1182: "color-logger": {  
1183: "version": "0.0.3",  
1184: "resolved": "https://registry.npmjs.org/color-logger/-/color-logger-0.0.3.tgz",  
1185: "integrity": "sha1-2bIt0dlz4Waxi/MT+fSBu6TfIBg=",  
1186: "dev": true  
1187: },  
1188: "domhandler": {  
1189: "version": "2.3.0",  
1190: "resolved": "https://registry.npmjs.org/domhandler/-/domhandler-2.3.0.tgz",  
1191: "integrity": "sha1-LeWaCCLVAn+r/28DLCsloqir5zg=",  
1192: "dev": true,  
1193: "requires": {  
1194: "domelementtype": "1.3.0"  
1195: }  
1196: },  
1197: "htmlparser2": {  
1198: "version": "3.8.3",  
1199: "resolved": "https://registry.npmjs.org/htmlparser2/-/htmlparser2-3.8.3.tgz",  
1200: "integrity": "sha1-mWwosZFRaovoZQGn15dX5ccMEGg=",  
1201: "dev": true,  
1202: "requires": {  
1203: "domelementtype": "1.3.0",  
1204: "domhandler": "2.3.0",  
1205: "domutils": "1.5.1",  
1206: "entities": "1.0.0",  
1207: "readable-stream": "1.1.14"  
1208: },  
1209: "dependencies": {  
1210: "entities": {  
1211: "version": "1.0.0",  
1212: "resolved": "https://registry.npmjs.org/entities/-/entities-1.0.0.tgz",  
1213: "integrity": "sha1-sph6o4ITR/zeZCsk/fyeT7cSvyY=",  
1214: "dev": true  
1215: }  
1216: }  
1217: },  
1218: "isarray": {  
1219: "version": "0.0.1",  
1220: "resolved": "https://registry.npmjs.org/isarray/-/isarray-0.0.1.tgz",  
1221: "integrity": "sha1-ihis/Kmo9Bd+Cav8YDiTmwXR7t8=",  
1222: "dev": true  
1223: },  
1224: "readable-stream": {  
1225: "version": "1.1.14",  
1226: "resolved": "https://registry.npmjs.org/readable-stream/-/readable-stream-1.1.14.tgz",  
1227: "integrity": "sha1-fPTFTvZI44EwhMY23SB54WbAgdk=",  
1228: "dev": true,  
1229: "requires": {  
1230: "core-util-is": "1.0.2",  
1231: "inherits": "2.0.3",  
1232: "isarray": "0.0.1",  
1233: "string\_decoder": "0.10.31"  
1234: }  
1235: },  
1236: "string\_decoder": {  
1237: "version": "0.10.31",  
1238: "resolved": "https://registry.npmjs.org/string\_decoder/-/string\_decoder-0.10.31.tgz",  
1239: "integrity": "sha1-YuIDvEF2bGwoyfyEMB2rHFMQ+pQ=",  
1240: "dev": true  
1241: }  
1242: }  
1243: },  
1244: "iconv-lite": {  
1245: "version": "0.4.19",  
1246: "resolved": "https://registry.npmjs.org/iconv-lite/-/iconv-lite-0.4.19.tgz",  
1247: "integrity": "sha512-oTZqweIP51xaGPI4uPa56/Pri/480R+mo7SeU+YETByQNhDG55ycFyNLIgta9vXhILrxXDmF7ZGhqZIcuN0gJQ==",  
1248: "dev": true  
1249: },  
1250: "inflight": {  
1251: "version": "1.0.6",  
1252: "resolved": "https://registry.npmjs.org/inflight/-/inflight-1.0.6.tgz",  
1253: "integrity": "sha1-Sb1jMdfQLQwJvJEKEHW6gWW1bfk=",  
1254: "dev": true,  
1255: "requires": {  
1256: "once": "1.4.0",  
1257: "wrappy": "1.0.2"  
1258: }  
1259: },  
1260: "inherits": {  
1261: "version": "2.0.3",  
1262: "resolved": "https://registry.npmjs.org/inherits/-/inherits-2.0.3.tgz",  
1263: "integrity": "sha1-Yzwsg+PaQqUC9SRmAiSA9CCCYd4=",  
1264: "dev": true  
1265: },  
1266: "invariant": {  
1267: "version": "2.2.4",  
1268: "resolved": "https://registry.npmjs.org/invariant/-/invariant-2.2.4.tgz",  
1269: "integrity": "sha512-phJfQVBuaJM5raOpJjSfkiD6BpbCE4Ns//LaXl6wGYtUBY83nWS6Rf9tXm2e8VaK60JEjYldbPif/A2B1C2gNA==",  
1270: "dev": true,  
1271: "requires": {  
1272: "loose-envify": "1.3.1"  
1273: }  
1274: },  
1275: "ipaddr.js": {  
1276: "version": "1.6.0",  
1277: "resolved": "https://registry.npmjs.org/ipaddr.js/-/ipaddr.js-1.6.0.tgz",  
1278: "integrity": "sha1-4/o1e3c9phnybpXwSdBVxyeW+Gs=",  
1279: "dev": true  
1280: },  
1281: "is-finite": {  
1282: "version": "1.0.2",  
1283: "resolved": "https://registry.npmjs.org/is-finite/-/is-finite-1.0.2.tgz",  
1284: "integrity": "sha1-zGZ3aVYCvlUO8R6LSqYwU0K20Ko=",  
1285: "dev": true,  
1286: "requires": {  
1287: "number-is-nan": "1.0.1"  
1288: }  
1289: },  
1290: "is-typedarray": {  
1291: "version": "1.0.0",  
1292: "resolved": "https://registry.npmjs.org/is-typedarray/-/is-typedarray-1.0.0.tgz",  
1293: "integrity": "sha1-5HnICFjfDBsR3dppQPlgEfzaSpo=",  
1294: "dev": true,  
1295: "optional": true  
1296: },  
1297: "isarray": {  
1298: "version": "1.0.0",  
1299: "resolved": "https://registry.npmjs.org/isarray/-/isarray-1.0.0.tgz",  
1300: "integrity": "sha1-u5NdSFgsuhaMBoNJV6VKPgcSTxE=",  
1301: "dev": true  
1302: },  
1303: "isstream": {  
1304: "version": "0.1.2",  
1305: "resolved": "https://registry.npmjs.org/isstream/-/isstream-0.1.2.tgz",  
1306: "integrity": "sha1-R+Y/evVa+m+S4VAOaQ64uFKcCZo=",  
1307: "dev": true,  
1308: "optional": true  
1309: },  
1310: "js-tokens": {  
1311: "version": "3.0.2",  
1312: "resolved": "https://registry.npmjs.org/js-tokens/-/js-tokens-3.0.2.tgz",  
1313: "integrity": "sha1-mGbfOVECEw449/mWvOtlRDIJwls=",  
1314: "dev": true  
1315: },  
1316: "js2xmlparser": {  
1317: "version": "3.0.0",  
1318: "resolved": "https://registry.npmjs.org/js2xmlparser/-/js2xmlparser-3.0.0.tgz",  
1319: "integrity": "sha1-P7YOqgicVED5MZ9RdgzNB+JJlzM=",  
1320: "dev": true,  
1321: "requires": {  
1322: "xmlcreate": "1.0.2"  
1323: }  
1324: },  
1325: "jsbn": {  
1326: "version": "0.1.1",  
1327: "resolved": "https://registry.npmjs.org/jsbn/-/jsbn-0.1.1.tgz",  
1328: "integrity": "sha1-peZUwuWi3rXyAdls77yoDA7y9RM=",  
1329: "dev": true,  
1330: "optional": true  
1331: },  
1332: "jsdoc": {  
1333: "version": "3.5.5",  
1334: "resolved": "https://registry.npmjs.org/jsdoc/-/jsdoc-3.5.5.tgz",  
1335: "integrity": "sha512-6PxB65TAU4WO0Wzyr/4/YhlGovXl0EVYfpKbpSroSj0qBxT4/xod/l40Opkm38dRHRdQgdeY836M0uVnJQG7kg==",  
1336: "dev": true,  
1337: "requires": {  
1338: "babylon": "7.0.0-beta.19",  
1339: "bluebird": "3.5.1",  
1340: "catharsis": "0.8.9",  
1341: "escape-string-regexp": "1.0.5",  
1342: "js2xmlparser": "3.0.0",  
1343: "klaw": "2.0.0",  
1344: "marked": "0.3.19",  
1345: "mkdirp": "0.5.1",  
1346: "requizzle": "0.2.1",  
1347: "strip-json-comments": "2.0.1",  
1348: "taffydb": "2.6.2",  
1349: "underscore": "1.8.3"  
1350: },  
1351: "dependencies": {  
1352: "babylon": {  
1353: "version": "7.0.0-beta.19",  
1354: "resolved": "https://registry.npmjs.org/babylon/-/babylon-7.0.0-beta.19.tgz",  
1355: "integrity": "sha512-Vg0C9s/REX6/WIXN37UKpv5ZhRi6A4pjHlpkE34+8/a6c2W1Q692n3hmc+SZG5lKRnaExLUbxtJ1SVT+KaCQ/A==",  
1356: "dev": true  
1357: },  
1358: "klaw": {  
1359: "version": "2.0.0",  
1360: "resolved": "https://registry.npmjs.org/klaw/-/klaw-2.0.0.tgz",  
1361: "integrity": "sha1-WcEo4Nxc5BAgEVEZTuucv4WGUPY=",  
1362: "dev": true,  
1363: "requires": {  
1364: "graceful-fs": "4.1.11"  
1365: }  
1366: },  
1367: "taffydb": {  
1368: "version": "2.6.2",  
1369: "resolved": "https://registry.npmjs.org/taffydb/-/taffydb-2.6.2.tgz",  
1370: "integrity": "sha1-fLy2S1oUG2ou/CxdLGe04VCyomg=",  
1371: "dev": true  
1372: }  
1373: }  
1374: },  
1375: "jsdom": {  
1376: "version": "7.2.2",  
1377: "resolved": "https://registry.npmjs.org/jsdom/-/jsdom-7.2.2.tgz",  
1378: "integrity": "sha1-QLQCdwwr2iNGkJa+6Rq2deOx/G4=",  
1379: "dev": true,  
1380: "optional": true,  
1381: "requires": {  
1382: "abab": "1.0.4",  
1383: "acorn": "2.7.0",  
1384: "acorn-globals": "1.0.9",  
1385: "cssom": "0.3.2",  
1386: "cssstyle": "0.2.37",  
1387: "escodegen": "1.9.1",  
1388: "nwmatcher": "1.4.4",  
1389: "parse5": "1.5.1",  
1390: "request": "2.87.0",  
1391: "sax": "1.2.4",  
1392: "symbol-tree": "3.2.2",  
1393: "tough-cookie": "2.3.4",  
1394: "webidl-conversions": "2.0.1",  
1395: "whatwg-url-compat": "0.6.5",  
1396: "xml-name-validator": "2.0.1"  
1397: },  
1398: "dependencies": {  
1399: "parse5": {  
1400: "version": "1.5.1",  
1401: "resolved": "https://registry.npmjs.org/parse5/-/parse5-1.5.1.tgz",  
1402: "integrity": "sha1-m387DeMr543CQBsXVzzK8Pb1nZQ=",  
1403: "dev": true,  
1404: "optional": true  
1405: }  
1406: }  
1407: },  
1408: "jsesc": {  
1409: "version": "1.3.0",  
1410: "resolved": "https://registry.npmjs.org/jsesc/-/jsesc-1.3.0.tgz",  
1411: "integrity": "sha1-RsP+yMGJKxKwgz25vHYiF226s0s=",  
1412: "dev": true  
1413: },  
1414: "json-schema": {  
1415: "version": "0.2.3",  
1416: "resolved": "https://registry.npmjs.org/json-schema/-/json-schema-0.2.3.tgz",  
1417: "integrity": "sha1-tIDIkuWaLwWVTOcnvT8qTogvnhM=",  
1418: "dev": true,  
1419: "optional": true  
1420: },  
1421: "json-schema-traverse": {  
1422: "version": "0.3.1",  
1423: "resolved": "https://registry.npmjs.org/json-schema-traverse/-/json-schema-traverse-0.3.1.tgz",  
1424: "integrity": "sha1-NJptRMU6Ud6JtAgFxdXlm0F9M0A=",  
1425: "dev": true,  
1426: "optional": true  
1427: },  
1428: "json-stringify-safe": {  
1429: "version": "5.0.1",  
1430: "resolved": "https://registry.npmjs.org/json-stringify-safe/-/json-stringify-safe-5.0.1.tgz",  
1431: "integrity": "sha1-Epai1Y/UXxmg9s4B1lcB4sc1tus=",  
1432: "dev": true  
1433: },  
1434: "jsonfile": {  
1435: "version": "4.0.0",  
1436: "resolved": "https://registry.npmjs.org/jsonfile/-/jsonfile-4.0.0.tgz",  
1437: "integrity": "sha1-h3Gq4HmbZAdrdmQPygWPnBDjPss=",  
1438: "dev": true,  
1439: "requires": {  
1440: "graceful-fs": "4.1.11"  
1441: }  
1442: },  
1443: "jsprim": {  
1444: "version": "1.4.1",  
1445: "resolved": "https://registry.npmjs.org/jsprim/-/jsprim-1.4.1.tgz",  
1446: "integrity": "sha1-MT5mvB5cwG5Di8G3SZwuXFastqI=",  
1447: "dev": true,  
1448: "optional": true,  
1449: "requires": {  
1450: "assert-plus": "1.0.0",  
1451: "extsprintf": "1.3.0",  
1452: "json-schema": "0.2.3",  
1453: "verror": "1.10.0"  
1454: }  
1455: },  
1456: "klaw": {  
1457: "version": "1.3.1",  
1458: "resolved": "https://registry.npmjs.org/klaw/-/klaw-1.3.1.tgz",  
1459: "integrity": "sha1-QIhDO0azsbolnXh4XY6W9zugJDk=",  
1460: "dev": true,  
1461: "requires": {  
1462: "graceful-fs": "4.1.11"  
1463: }  
1464: },  
1465: "levn": {  
1466: "version": "0.3.0",  
1467: "resolved": "https://registry.npmjs.org/levn/-/levn-0.3.0.tgz",  
1468: "integrity": "sha1-OwmSTt+fCDwEkP3UwLxEIeBHZO4=",  
1469: "dev": true,  
1470: "optional": true,  
1471: "requires": {  
1472: "prelude-ls": "1.1.2",  
1473: "type-check": "0.3.2"  
1474: }  
1475: },  
1476: "linkify-it": {  
1477: "version": "1.2.4",  
1478: "resolved": "https://registry.npmjs.org/linkify-it/-/linkify-it-1.2.4.tgz",  
1479: "integrity": "sha1-B3NSbDF8j9E71TTuHRgP+Iq/iBo=",  
1480: "dev": true,  
1481: "requires": {  
1482: "uc.micro": "1.0.5"  
1483: }  
1484: },  
1485: "lodash": {  
1486: "version": "4.17.10",  
1487: "resolved": "https://registry.npmjs.org/lodash/-/lodash-4.17.10.tgz",  
1488: "integrity": "sha512-UejweD1pDoXu+AD825lWwp4ZGtSwgnpZxb3JDViD7StjQz+Nb/6l093lx4OQ0foGWNRoc19mWy7BzL+UAK2iVg==",  
1489: "dev": true  
1490: },  
1491: "lodash.assignin": {  
1492: "version": "4.2.0",  
1493: "resolved": "https://registry.npmjs.org/lodash.assignin/-/lodash.assignin-4.2.0.tgz",  
1494: "integrity": "sha1-uo31+4QesKPoBEIysOJjqNxqKKI=",  
1495: "dev": true  
1496: },  
1497: "lodash.bind": {  
1498: "version": "4.2.1",  
1499: "resolved": "https://registry.npmjs.org/lodash.bind/-/lodash.bind-4.2.1.tgz",  
1500: "integrity": "sha1-euMBfpOWIqwxt9fX3LGzTbFpDTU=",  
1501: "dev": true  
1502: },  
1503: "lodash.defaults": {  
1504: "version": "4.2.0",  
1505: "resolved": "https://registry.npmjs.org/lodash.defaults/-/lodash.defaults-4.2.0.tgz",  
1506: "integrity": "sha1-0JF4cW/+pN3p5ft7N/bwgCJ0WAw=",  
1507: "dev": true  
1508: },  
1509: "lodash.filter": {  
1510: "version": "4.6.0",  
1511: "resolved": "https://registry.npmjs.org/lodash.filter/-/lodash.filter-4.6.0.tgz",  
1512: "integrity": "sha1-ZosdSYFgOuHMWm+nYBQ+SAtMSs4=",  
1513: "dev": true  
1514: },  
1515: "lodash.flatten": {  
1516: "version": "4.4.0",  
1517: "resolved": "https://registry.npmjs.org/lodash.flatten/-/lodash.flatten-4.4.0.tgz",  
1518: "integrity": "sha1-8xwiIlqWMtK7+OSt2+8kCqdlph8=",  
1519: "dev": true  
1520: },  
1521: "lodash.foreach": {  
1522: "version": "4.5.0",  
1523: "resolved": "https://registry.npmjs.org/lodash.foreach/-/lodash.foreach-4.5.0.tgz",  
1524: "integrity": "sha1-Gmo16s5AEoDH8G3d7DUWWrJ+PlM=",  
1525: "dev": true  
1526: },  
1527: "lodash.map": {  
1528: "version": "4.6.0",  
1529: "resolved": "https://registry.npmjs.org/lodash.map/-/lodash.map-4.6.0.tgz",  
1530: "integrity": "sha1-dx7Hg540c9nEzeKLGTlMNWL09tM=",  
1531: "dev": true  
1532: },  
1533: "lodash.merge": {  
1534: "version": "4.6.1",  
1535: "resolved": "https://registry.npmjs.org/lodash.merge/-/lodash.merge-4.6.1.tgz",  
1536: "integrity": "sha512-AOYza4+Hf5z1/0Hztxpm2/xiPZgi/cjMqdnKTUWTBSKchJlxXXuUSxCCl8rJlf4g6yww/j6mA8nC8Hw/EZWxKQ==",  
1537: "dev": true  
1538: },  
1539: "lodash.pick": {  
1540: "version": "4.4.0",  
1541: "resolved": "https://registry.npmjs.org/lodash.pick/-/lodash.pick-4.4.0.tgz",  
1542: "integrity": "sha1-UvBWEP/53tQiYRRB7R/BI6AwAbM=",  
1543: "dev": true  
1544: },  
1545: "lodash.reduce": {  
1546: "version": "4.6.0",  
1547: "resolved": "https://registry.npmjs.org/lodash.reduce/-/lodash.reduce-4.6.0.tgz",  
1548: "integrity": "sha1-8atrg5KZrUj3hKu/R2WW8DuRTTs=",  
1549: "dev": true  
1550: },  
1551: "lodash.reject": {  
1552: "version": "4.6.0",  
1553: "resolved": "https://registry.npmjs.org/lodash.reject/-/lodash.reject-4.6.0.tgz",  
1554: "integrity": "sha1-gNZJLcFHCGS79YNTO2UfQqn1JBU=",  
1555: "dev": true  
1556: },  
1557: "lodash.some": {  
1558: "version": "4.6.0",  
1559: "resolved": "https://registry.npmjs.org/lodash.some/-/lodash.some-4.6.0.tgz",  
1560: "integrity": "sha1-G7nzFO9ri63tE7VJFpsqlF62jk0=",  
1561: "dev": true  
1562: },  
1563: "loose-envify": {  
1564: "version": "1.3.1",  
1565: "resolved": "https://registry.npmjs.org/loose-envify/-/loose-envify-1.3.1.tgz",  
1566: "integrity": "sha1-0aitM/qc4OcT1l/dCsi3SNR4yEg=",  
1567: "dev": true,  
1568: "requires": {  
1569: "js-tokens": "3.0.2"  
1570: }  
1571: },  
1572: "markdown-it": {  
1573: "version": "4.4.0",  
1574: "resolved": "https://registry.npmjs.org/markdown-it/-/markdown-it-4.4.0.tgz",  
1575: "integrity": "sha1-PfNz2+pYepp/7z5WMRtokI91xBQ=",  
1576: "dev": true,  
1577: "requires": {  
1578: "argparse": "1.0.10",  
1579: "entities": "1.1.1",  
1580: "linkify-it": "1.2.4",  
1581: "mdurl": "1.0.1",  
1582: "uc.micro": "1.0.5"  
1583: }  
1584: },  
1585: "marked": {  
1586: "version": "0.3.19",  
1587: "resolved": "https://registry.npmjs.org/marked/-/marked-0.3.19.tgz",  
1588: "integrity": "sha512-ea2eGWOqNxPcXv8dyERdSr/6FmzvWwzjMxpfGB/sbMccXoct+xY+YukPD+QTUZwyvK7BZwcr4m21WBOW41pAkg==",  
1589: "dev": true  
1590: },  
1591: "mdn-links": {  
1592: "version": "0.1.0",  
1593: "resolved": "https://registry.npmjs.org/mdn-links/-/mdn-links-0.1.0.tgz",  
1594: "integrity": "sha1-4kyDuXy0xYhsw58veAcF+/4nOqU=",  
1595: "dev": true  
1596: },  
1597: "mdurl": {  
1598: "version": "1.0.1",  
1599: "resolved": "https://registry.npmjs.org/mdurl/-/mdurl-1.0.1.tgz",  
1600: "integrity": "sha1-/oWy7HWlkDfyrf7BAP1sYBdhFS4=",  
1601: "dev": true  
1602: },  
1603: "media-typer": {  
1604: "version": "0.3.0",  
1605: "resolved": "https://registry.npmjs.org/media-typer/-/media-typer-0.3.0.tgz",  
1606: "integrity": "sha1-hxDXrwqmJvj/+hzgAWhUUmMlV0g=",  
1607: "dev": true  
1608: },  
1609: "merge-descriptors": {  
1610: "version": "1.0.1",  
1611: "resolved": "https://registry.npmjs.org/merge-descriptors/-/merge-descriptors-1.0.1.tgz",  
1612: "integrity": "sha1-sAqqVW3YtEVoFQ7J0blT8/kMu2E=",  
1613: "dev": true  
1614: },  
1615: "methods": {  
1616: "version": "1.1.2",  
1617: "resolved": "https://registry.npmjs.org/methods/-/methods-1.1.2.tgz",  
1618: "integrity": "sha1-VSmk1nZUE07cxSZmVoNbD4Ua/O4=",  
1619: "dev": true  
1620: },  
1621: "mime": {  
1622: "version": "1.4.1",  
1623: "resolved": "https://registry.npmjs.org/mime/-/mime-1.4.1.tgz",  
1624: "integrity": "sha512-KI1+qOZu5DcW6wayYHSzR/tXKCDC5Om4s1z2QJjDULzLcmf3DvzS7oluY4HCTrc+9FiKmWUgeNLg7W3uIQvxtQ==",  
1625: "dev": true  
1626: },  
1627: "mime-db": {  
1628: "version": "1.33.0",  
1629: "resolved": "https://registry.npmjs.org/mime-db/-/mime-db-1.33.0.tgz",  
1630: "integrity": "sha512-BHJ/EKruNIqJf/QahvxwQZXKygOQ256myeN/Ew+THcAa5q+PjyTTMMeNQC4DZw5AwfvelsUrA6B67NKMqXDbzQ==",  
1631: "dev": true  
1632: },  
1633: "mime-types": {  
1634: "version": "2.1.18",  
1635: "resolved": "https://registry.npmjs.org/mime-types/-/mime-types-2.1.18.tgz",  
1636: "integrity": "sha512-lc/aahn+t4/SWV/qcmumYjymLsWfN3ELhpmVuUFjgsORruuZPVSwAQryq+HHGvO/SI2KVX26bx+En+zhM8g8hQ==",  
1637: "dev": true,  
1638: "requires": {  
1639: "mime-db": "1.33.0"  
1640: }  
1641: },  
1642: "minimatch": {  
1643: "version": "3.0.4",  
1644: "resolved": "https://registry.npmjs.org/minimatch/-/minimatch-3.0.4.tgz",  
1645: "integrity": "sha512-yJHVQEhyqPLUTgt9B83PXu6W3rx4MvvHvSUvToogpwoGDOUQ+yDrR0HRot+yOCdCO7u4hX3pWft6kWBBcqh0UA==",  
1646: "dev": true,  
1647: "requires": {  
1648: "brace-expansion": "1.1.11"  
1649: }  
1650: },  
1651: "minimist": {  
1652: "version": "1.2.0",  
1653: "resolved": "https://registry.npmjs.org/minimist/-/minimist-1.2.0.tgz",  
1654: "integrity": "sha1-o1AIsg9BOD7sH7kU9M1d95omQoQ=",  
1655: "dev": true  
1656: },  
1657: "mkdirp": {  
1658: "version": "0.5.1",  
1659: "resolved": "https://registry.npmjs.org/mkdirp/-/mkdirp-0.5.1.tgz",  
1660: "integrity": "sha1-MAV0OOrGz3+MR2fzhkjWaX11yQM=",  
1661: "dev": true,  
1662: "requires": {  
1663: "minimist": "0.0.8"  
1664: },  
1665: "dependencies": {  
1666: "minimist": {  
1667: "version": "0.0.8",  
1668: "resolved": "https://registry.npmjs.org/minimist/-/minimist-0.0.8.tgz",  
1669: "integrity": "sha1-hX/Kv8M5fSYluCKCYuhqp6ARsF0=",  
1670: "dev": true  
1671: }  
1672: }  
1673: },  
1674: "ms": {  
1675: "version": "2.0.0",  
1676: "resolved": "https://registry.npmjs.org/ms/-/ms-2.0.0.tgz",  
1677: "integrity": "sha1-VgiurfwAvmwpAd9fmGF4jeDVl8g=",  
1678: "dev": true  
1679: },  
1680: "negotiator": {  
1681: "version": "0.6.1",  
1682: "resolved": "https://registry.npmjs.org/negotiator/-/negotiator-0.6.1.tgz",  
1683: "integrity": "sha1-KzJxhOiZIQEXeyhWP7XnECrNDKk=",  
1684: "dev": true  
1685: },  
1686: "nth-check": {  
1687: "version": "1.0.1",  
1688: "resolved": "https://registry.npmjs.org/nth-check/-/nth-check-1.0.1.tgz",  
1689: "integrity": "sha1-mSms32KPwsQQmN6rgqxYDPFJquQ=",  
1690: "dev": true,  
1691: "requires": {  
1692: "boolbase": "1.0.0"  
1693: }  
1694: },  
1695: "number-is-nan": {  
1696: "version": "1.0.1",  
1697: "resolved": "https://registry.npmjs.org/number-is-nan/-/number-is-nan-1.0.1.tgz",  
1698: "integrity": "sha1-CXtgK1NCKlIsGvuHkDGDNpQaAR0=",  
1699: "dev": true  
1700: },  
1701: "nwmatcher": {  
1702: "version": "1.4.4",  
1703: "resolved": "https://registry.npmjs.org/nwmatcher/-/nwmatcher-1.4.4.tgz",  
1704: "integrity": "sha512-3iuY4N5dhgMpCUrOVnuAdGrgxVqV2cJpM+XNccjR2DKOB1RUP0aA+wGXEiNziG/UKboFyGBIoKOaNlJxx8bciQ==",  
1705: "dev": true,  
1706: "optional": true  
1707: },  
1708: "oauth-sign": {  
1709: "version": "0.8.2",  
1710: "resolved": "https://registry.npmjs.org/oauth-sign/-/oauth-sign-0.8.2.tgz",  
1711: "integrity": "sha1-Rqarfwrq2N6unsBWV4C31O/rnUM=",  
1712: "dev": true,  
1713: "optional": true  
1714: },  
1715: "on-finished": {  
1716: "version": "2.3.0",  
1717: "resolved": "https://registry.npmjs.org/on-finished/-/on-finished-2.3.0.tgz",  
1718: "integrity": "sha1-IPEzZIGwg811M3mSoWlxqi2QaUc=",  
1719: "dev": true,  
1720: "requires": {  
1721: "ee-first": "1.1.1"  
1722: }  
1723: },  
1724: "once": {  
1725: "version": "1.4.0",  
1726: "resolved": "https://registry.npmjs.org/once/-/once-1.4.0.tgz",  
1727: "integrity": "sha1-WDsap3WWHUsROsF9nFC6753Xa9E=",  
1728: "dev": true,  
1729: "requires": {  
1730: "wrappy": "1.0.2"  
1731: }  
1732: },  
1733: "optionator": {  
1734: "version": "0.8.2",  
1735: "resolved": "https://registry.npmjs.org/optionator/-/optionator-0.8.2.tgz",  
1736: "integrity": "sha1-NkxeQJ0/TWMB1sC0wFu6UBgK62Q=",  
1737: "dev": true,  
1738: "optional": true,  
1739: "requires": {  
1740: "deep-is": "0.1.3",  
1741: "fast-levenshtein": "2.0.6",  
1742: "levn": "0.3.0",  
1743: "prelude-ls": "1.1.2",  
1744: "type-check": "0.3.2",  
1745: "wordwrap": "1.0.0"  
1746: }  
1747: },  
1748: "parse5": {  
1749: "version": "3.0.3",  
1750: "resolved": "https://registry.npmjs.org/parse5/-/parse5-3.0.3.tgz",  
1751: "integrity": "sha512-rgO9Zg5LLLkfJF9E6CCmXlSE4UVceloys8JrFqCcHloC3usd/kJCyPDwH2SOlzix2j3xaP9sUX3e8+kvkuleAA==",  
1752: "dev": true,  
1753: "requires": {  
1754: "@types/node": "10.1.3"  
1755: }  
1756: },  
1757: "parseurl": {  
1758: "version": "1.3.2",  
1759: "resolved": "https://registry.npmjs.org/parseurl/-/parseurl-1.3.2.tgz",  
1760: "integrity": "sha1-/CidTtiZMRlGDBViUyYs3I3mW/M=",  
1761: "dev": true  
1762: },  
1763: "path-is-absolute": {  
1764: "version": "1.0.1",  
1765: "resolved": "https://registry.npmjs.org/path-is-absolute/-/path-is-absolute-1.0.1.tgz",  
1766: "integrity": "sha1-F0uSaHNVNP+8es5r9TpanhtcX18=",  
1767: "dev": true  
1768: },  
1769: "path-to-regexp": {  
1770: "version": "0.1.7",  
1771: "resolved": "https://registry.npmjs.org/path-to-regexp/-/path-to-regexp-0.1.7.tgz",  
1772: "integrity": "sha1-32BBeABfUi8V60SQ5yR6G/qmf4w=",  
1773: "dev": true  
1774: },  
1775: "performance-now": {  
1776: "version": "2.1.0",  
1777: "resolved": "https://registry.npmjs.org/performance-now/-/performance-now-2.1.0.tgz",  
1778: "integrity": "sha1-Ywn04OX6kT7BxpMHrjZLSzd8nns=",  
1779: "dev": true,  
1780: "optional": true  
1781: },  
1782: "prelude-ls": {  
1783: "version": "1.1.2",  
1784: "resolved": "https://registry.npmjs.org/prelude-ls/-/prelude-ls-1.1.2.tgz",  
1785: "integrity": "sha1-IZMqVJ9eUv/ZqCf1cOBL5iqX2lQ=",  
1786: "dev": true  
1787: },  
1788: "process-nextick-args": {  
1789: "version": "2.0.0",  
1790: "resolved": "https://registry.npmjs.org/process-nextick-args/-/process-nextick-args-2.0.0.tgz",  
1791: "integrity": "sha512-MtEC1TqN0EU5nephaJ4rAtThHtC86dNN9qCuEhtshvpVBkAW5ZO7BASN9REnF9eoXGcRub+pFuKEpOHE+HbEMw==",  
1792: "dev": true  
1793: },  
1794: "proxy-addr": {  
1795: "version": "2.0.3",  
1796: "resolved": "https://registry.npmjs.org/proxy-addr/-/proxy-addr-2.0.3.tgz",  
1797: "integrity": "sha512-jQTChiCJteusULxjBp8+jftSQE5Obdl3k4cnmLA6WXtK6XFuWRnvVL7aCiBqaLPM8c4ph0S4tKna8XvmIwEnXQ==",  
1798: "dev": true,  
1799: "requires": {  
1800: "forwarded": "0.1.2",  
1801: "ipaddr.js": "1.6.0"  
1802: }  
1803: },  
1804: "punycode": {  
1805: "version": "1.4.1",  
1806: "resolved": "https://registry.npmjs.org/punycode/-/punycode-1.4.1.tgz",  
1807: "integrity": "sha1-wNWmOycYgArY4esPpSachN1BhF4=",  
1808: "dev": true  
1809: },  
1810: "qs": {  
1811: "version": "6.5.2",  
1812: "resolved": "https://registry.npmjs.org/qs/-/qs-6.5.2.tgz",  
1813: "integrity": "sha512-N5ZAX4/LxJmF+7wN74pUD6qAh9/wnvdQcjq9TZjevvXzSUo7bfmw91saqMjzGS2xq91/odN2dW/WOl7qQHNDGA==",  
1814: "dev": true,  
1815: "optional": true  
1816: },  
1817: "range-parser": {  
1818: "version": "1.2.0",  
1819: "resolved": "https://registry.npmjs.org/range-parser/-/range-parser-1.2.0.tgz",  
1820: "integrity": "sha1-9JvmtIeJTdxA3MlKMi9hEJLgDV4=",  
1821: "dev": true  
1822: },  
1823: "raw-body": {  
1824: "version": "2.3.2",  
1825: "resolved": "https://registry.npmjs.org/raw-body/-/raw-body-2.3.2.tgz",  
1826: "integrity": "sha1-vNYMd9Prk83gBQKVw/N5OJvIj4k=",  
1827: "dev": true,  
1828: "requires": {  
1829: "bytes": "3.0.0",  
1830: "http-errors": "1.6.2",  
1831: "iconv-lite": "0.4.19",  
1832: "unpipe": "1.0.0"  
1833: },  
1834: "dependencies": {  
1835: "depd": {  
1836: "version": "1.1.1",  
1837: "resolved": "https://registry.npmjs.org/depd/-/depd-1.1.1.tgz",  
1838: "integrity": "sha1-V4O04cRZ8G+lyif5kfPQbnoxA1k=",  
1839: "dev": true  
1840: },  
1841: "http-errors": {  
1842: "version": "1.6.2",  
1843: "resolved": "https://registry.npmjs.org/http-errors/-/http-errors-1.6.2.tgz",  
1844: "integrity": "sha1-CgAsyFcHGSp+eUbO7cERVfYOxzY=",  
1845: "dev": true,  
1846: "requires": {  
1847: "depd": "1.1.1",  
1848: "inherits": "2.0.3",  
1849: "setprototypeof": "1.0.3",  
1850: "statuses": "1.4.0"  
1851: }  
1852: },  
1853: "setprototypeof": {  
1854: "version": "1.0.3",  
1855: "resolved": "https://registry.npmjs.org/setprototypeof/-/setprototypeof-1.0.3.tgz",  
1856: "integrity": "sha1-ZlZ+NwQ+608E2RvWWMDL77VbjgQ=",  
1857: "dev": true  
1858: }  
1859: }  
1860: },  
1861: "readable-stream": {  
1862: "version": "2.3.6",  
1863: "resolved": "https://registry.npmjs.org/readable-stream/-/readable-stream-2.3.6.tgz",  
1864: "integrity": "sha512-tQtKA9WIAhBF3+VLAseyMqZeBjW0AHJoxOtYqSUZNJxauErmLbVm2FW1y+J/YA9dUrAC39ITejlZWhVIwawkKw==",  
1865: "dev": true,  
1866: "requires": {  
1867: "core-util-is": "1.0.2",  
1868: "inherits": "2.0.3",  
1869: "isarray": "1.0.0",  
1870: "process-nextick-args": "2.0.0",  
1871: "safe-buffer": "5.1.2",  
1872: "string\_decoder": "1.1.1",  
1873: "util-deprecate": "1.0.2"  
1874: }  
1875: },  
1876: "regenerator-runtime": {  
1877: "version": "0.11.1",  
1878: "resolved": "https://registry.npmjs.org/regenerator-runtime/-/regenerator-runtime-0.11.1.tgz",  
1879: "integrity": "sha512-MguG95oij0fC3QV3URf4V2SDYGJhJnJGqvIIgdECeODCT98wSWDAJ94SSuVpYQUoTcGUIL6L4yNB7j1DFFHSBg==",  
1880: "dev": true  
1881: },  
1882: "repeating": {  
1883: "version": "2.0.1",  
1884: "resolved": "https://registry.npmjs.org/repeating/-/repeating-2.0.1.tgz",  
1885: "integrity": "sha1-UhTFOpJtNVJwdSf7q0FdvAjQbdo=",  
1886: "dev": true,  
1887: "requires": {  
1888: "is-finite": "1.0.2"  
1889: }  
1890: },  
1891: "request": {  
1892: "version": "2.87.0",  
1893: "resolved": "https://registry.npmjs.org/request/-/request-2.87.0.tgz",  
1894: "integrity": "sha512-fcogkm7Az5bsS6Sl0sibkbhcKsnyon/jV1kF3ajGmF0c8HrttdKTPRT9hieOaQHA5HEq6r8OyWOo/o781C1tNw==",  
1895: "dev": true,  
1896: "optional": true,  
1897: "requires": {  
1898: "aws-sign2": "0.7.0",  
1899: "aws4": "1.7.0",  
1900: "caseless": "0.12.0",  
1901: "combined-stream": "1.0.6",  
1902: "extend": "3.0.1",  
1903: "forever-agent": "0.6.1",  
1904: "form-data": "2.3.2",  
1905: "har-validator": "5.0.3",  
1906: "http-signature": "1.2.0",  
1907: "is-typedarray": "1.0.0",  
1908: "isstream": "0.1.2",  
1909: "json-stringify-safe": "5.0.1",  
1910: "mime-types": "2.1.18",  
1911: "oauth-sign": "0.8.2",  
1912: "performance-now": "2.1.0",  
1913: "qs": "6.5.2",  
1914: "safe-buffer": "5.1.2",  
1915: "tough-cookie": "2.3.4",  
1916: "tunnel-agent": "0.6.0",  
1917: "uuid": "3.2.1"  
1918: }  
1919: },  
1920: "requizzle": {  
1921: "version": "0.2.1",  
1922: "resolved": "https://registry.npmjs.org/requizzle/-/requizzle-0.2.1.tgz",  
1923: "integrity": "sha1-aUPDUwxNmn5G8c3dUcFY/GcM294=",  
1924: "dev": true,  
1925: "requires": {  
1926: "underscore": "1.6.0"  
1927: },  
1928: "dependencies": {  
1929: "underscore": {  
1930: "version": "1.6.0",  
1931: "resolved": "https://registry.npmjs.org/underscore/-/underscore-1.6.0.tgz",  
1932: "integrity": "sha1-izixDKze9jM3uLJOT/htRa6lKag=",  
1933: "dev": true  
1934: }  
1935: }  
1936: },  
1937: "rimraf": {  
1938: "version": "2.6.2",  
1939: "resolved": "https://registry.npmjs.org/rimraf/-/rimraf-2.6.2.tgz",  
1940: "integrity": "sha512-lreewLK/BlghmxtfH36YYVg1i8IAce4TI7oao75I1g245+6BctqTVQiBP3YUJ9C6DQOXJmkYR9X9fCLtCOJc5w==",  
1941: "dev": true,  
1942: "requires": {  
1943: "glob": "7.1.2"  
1944: }  
1945: },  
1946: "safe-buffer": {  
1947: "version": "5.1.2",  
1948: "resolved": "https://registry.npmjs.org/safe-buffer/-/safe-buffer-5.1.2.tgz",  
1949: "integrity": "sha512-Gd2UZBJDkXlY7GbJxfsE8/nvKkUEU1G38c1siN6QP6a9PT9MmHB8GnpscSmMJSoF8LOIrt8ud/wPtojys4G6+g==",  
1950: "dev": true  
1951: },  
1952: "sax": {  
1953: "version": "1.2.4",  
1954: "resolved": "https://registry.npmjs.org/sax/-/sax-1.2.4.tgz",  
1955: "integrity": "sha512-NqVDv9TpANUjFm0N8uM5GxL36UgKi9/atZw+x7YFnQ8ckwFGKrl4xX4yWtrey3UJm5nP1kUbnYgLopqWNSRhWw==",  
1956: "dev": true,  
1957: "optional": true  
1958: },  
1959: "send": {  
1960: "version": "0.16.2",  
1961: "resolved": "https://registry.npmjs.org/send/-/send-0.16.2.tgz",  
1962: "integrity": "sha512-E64YFPUssFHEFBvpbbjr44NCLtI1AohxQ8ZSiJjQLskAdKuriYEP6VyGEsRDH8ScozGpkaX1BGvhanqCwkcEZw==",  
1963: "dev": true,  
1964: "requires": {  
1965: "debug": "2.6.9",  
1966: "depd": "1.1.2",  
1967: "destroy": "1.0.4",  
1968: "encodeurl": "1.0.2",  
1969: "escape-html": "1.0.3",  
1970: "etag": "1.8.1",  
1971: "fresh": "0.5.2",  
1972: "http-errors": "1.6.3",  
1973: "mime": "1.4.1",  
1974: "ms": "2.0.0",  
1975: "on-finished": "2.3.0",  
1976: "range-parser": "1.2.0",  
1977: "statuses": "1.4.0"  
1978: }  
1979: },  
1980: "serve-static": {  
1981: "version": "1.13.2",  
1982: "resolved": "https://registry.npmjs.org/serve-static/-/serve-static-1.13.2.tgz",  
1983: "integrity": "sha512-p/tdJrO4U387R9oMjb1oj7qSMaMfmOyd4j9hOFoxZe2baQszgHcSWjuya/CiT5kgZZKRudHNOA0pYXOl8rQ5nw==",  
1984: "dev": true,  
1985: "requires": {  
1986: "encodeurl": "1.0.2",  
1987: "escape-html": "1.0.3",  
1988: "parseurl": "1.3.2",  
1989: "send": "0.16.2"  
1990: }  
1991: },  
1992: "setprototypeof": {  
1993: "version": "1.1.0",  
1994: "resolved": "https://registry.npmjs.org/setprototypeof/-/setprototypeof-1.1.0.tgz",  
1995: "integrity": "sha512-BvE/TwpZX4FXExxOxZyRGQQv651MSwmWKZGqvmPcRIjDqWub67kTKuIMx43cZZrS/cBBzwBcNDWoFxt2XEFIpQ==",  
1996: "dev": true  
1997: },  
1998: "sntp": {  
1999: "version": "0.2.4",  
2000: "resolved": "https://registry.npmjs.org/sntp/-/sntp-0.2.4.tgz",  
2001: "integrity": "sha1-+4hfGLDzqtGJ+CSGJTa87ux1CQA=",  
2002: "dev": true,  
2003: "optional": true,  
2004: "requires": {  
2005: "hoek": "0.9.1"  
2006: }  
2007: },  
2008: "source-map": {  
2009: "version": "0.5.7",  
2010: "resolved": "https://registry.npmjs.org/source-map/-/source-map-0.5.7.tgz",  
2011: "integrity": "sha1-igOdLRAh0i0eoUyA2OpGi6LvP8w=",  
2012: "dev": true  
2013: },  
2014: "sprintf-js": {  
2015: "version": "1.0.3",  
2016: "resolved": "https://registry.npmjs.org/sprintf-js/-/sprintf-js-1.0.3.tgz",  
2017: "integrity": "sha1-BOaSb2YolTVPPdAVIDYzuFcpfiw=",  
2018: "dev": true  
2019: },  
2020: "sshpk": {  
2021: "version": "1.14.1",  
2022: "resolved": "https://registry.npmjs.org/sshpk/-/sshpk-1.14.1.tgz",  
2023: "integrity": "sha1-Ew9Zde3a2WPx1W+SuaxsUfqfg+s=",  
2024: "dev": true,  
2025: "optional": true,  
2026: "requires": {  
2027: "asn1": "0.2.3",  
2028: "assert-plus": "1.0.0",  
2029: "bcrypt-pbkdf": "1.0.1",  
2030: "dashdash": "1.14.1",  
2031: "ecc-jsbn": "0.1.1",  
2032: "getpass": "0.1.7",  
2033: "jsbn": "0.1.1",  
2034: "tweetnacl": "0.14.5"  
2035: }  
2036: },  
2037: "statuses": {  
2038: "version": "1.4.0",  
2039: "resolved": "https://registry.npmjs.org/statuses/-/statuses-1.4.0.tgz",  
2040: "integrity": "sha512-zhSCtt8v2NDrRlPQpCNtw/heZLtfUDqxBM1udqikb/Hbk52LK4nQSwr10u77iopCW5LsyHpuXS0GnEc48mLeew==",  
2041: "dev": true  
2042: },  
2043: "string\_decoder": {  
2044: "version": "1.1.1",  
2045: "resolved": "https://registry.npmjs.org/string\_decoder/-/string\_decoder-1.1.1.tgz",  
2046: "integrity": "sha512-n/ShnvDi6FHbbVfviro+WojiFzv+s8MPMHBczVePfUpDJLwoLT0ht1l4YwBCbi8pJAveEEdnkHyPyTP/mzRfwg==",  
2047: "dev": true,  
2048: "requires": {  
2049: "safe-buffer": "5.1.2"  
2050: }  
2051: },  
2052: "stringstream": {  
2053: "version": "0.0.6",  
2054: "resolved": "https://registry.npmjs.org/stringstream/-/stringstream-0.0.6.tgz",  
2055: "integrity": "sha512-87GEBAkegbBcweToUrdzf3eLhWNg06FJTebl4BVJz/JgWy8CvEr9dRtX5qWphiynMSQlxxi+QqN0z5T32SLlhA==",  
2056: "dev": true,  
2057: "optional": true  
2058: },  
2059: "strip-ansi": {  
2060: "version": "3.0.1",  
2061: "resolved": "https://registry.npmjs.org/strip-ansi/-/strip-ansi-3.0.1.tgz",  
2062: "integrity": "sha1-ajhfuIU9lS1f8F0Oiq+UJ43GPc8=",  
2063: "dev": true,  
2064: "requires": {  
2065: "ansi-regex": "2.1.1"  
2066: }  
2067: },  
2068: "strip-json-comments": {  
2069: "version": "2.0.1",  
2070: "resolved": "https://registry.npmjs.org/strip-json-comments/-/strip-json-comments-2.0.1.tgz",  
2071: "integrity": "sha1-PFMZQukIwml8DsNEhYwobHygpgo=",  
2072: "dev": true  
2073: },  
2074: "supports-color": {  
2075: "version": "2.0.0",  
2076: "resolved": "https://registry.npmjs.org/supports-color/-/supports-color-2.0.0.tgz",  
2077: "integrity": "sha1-U10EXOa2Nj+kARcIRimZXp3zJMc=",  
2078: "dev": true  
2079: },  
2080: "symbol-tree": {  
2081: "version": "3.2.2",  
2082: "resolved": "https://registry.npmjs.org/symbol-tree/-/symbol-tree-3.2.2.tgz",  
2083: "integrity": "sha1-rifbOPZgp64uHDt9G8KQgZuFGeY=",  
2084: "dev": true,  
2085: "optional": true  
2086: },  
2087: "taffydb": {  
2088: "version": "2.7.3",  
2089: "resolved": "https://registry.npmjs.org/taffydb/-/taffydb-2.7.3.tgz",  
2090: "integrity": "sha1-KtNxaWKUmPylvIQkMJbTzeDsOjQ=",  
2091: "dev": true  
2092: },  
2093: "to-fast-properties": {  
2094: "version": "1.0.3",  
2095: "resolved": "https://registry.npmjs.org/to-fast-properties/-/to-fast-properties-1.0.3.tgz",  
2096: "integrity": "sha1-uDVx+k2MJbguIxsG46MFXeTKGkc=",  
2097: "dev": true  
2098: },  
2099: "tough-cookie": {  
2100: "version": "2.3.4",  
2101: "resolved": "https://registry.npmjs.org/tough-cookie/-/tough-cookie-2.3.4.tgz",  
2102: "integrity": "sha512-TZ6TTfI5NtZnuyy/Kecv+CnoROnyXn2DN97LontgQpCwsX2XyLYCC0ENhYkehSOwAp8rTQKc/NUIF7BkQ5rKLA==",  
2103: "dev": true,  
2104: "requires": {  
2105: "punycode": "1.4.1"  
2106: }  
2107: },  
2108: "tr46": {  
2109: "version": "0.0.3",  
2110: "resolved": "https://registry.npmjs.org/tr46/-/tr46-0.0.3.tgz",  
2111: "integrity": "sha1-gYT9NH2snNwYWZLzpmIuFLnZq2o=",  
2112: "dev": true,  
2113: "optional": true  
2114: },  
2115: "trim-right": {  
2116: "version": "1.0.1",  
2117: "resolved": "https://registry.npmjs.org/trim-right/-/trim-right-1.0.1.tgz",  
2118: "integrity": "sha1-yy4SAwZ+DI3h9hQJS5/kVwTqYAM=",  
2119: "dev": true  
2120: },  
2121: "tunnel-agent": {  
2122: "version": "0.6.0",  
2123: "resolved": "https://registry.npmjs.org/tunnel-agent/-/tunnel-agent-0.6.0.tgz",  
2124: "integrity": "sha1-J6XeoGs2sEoKmWZ3SykIaPD8QP0=",  
2125: "dev": true,  
2126: "optional": true,  
2127: "requires": {  
2128: "safe-buffer": "5.1.2"  
2129: }  
2130: },  
2131: "tweetnacl": {  
2132: "version": "0.14.5",  
2133: "resolved": "https://registry.npmjs.org/tweetnacl/-/tweetnacl-0.14.5.tgz",  
2134: "integrity": "sha1-WuaBd/GS1EViadEIr6k/+HQ/T2Q=",  
2135: "dev": true,  
2136: "optional": true  
2137: },  
2138: "type-check": {  
2139: "version": "0.3.2",  
2140: "resolved": "https://registry.npmjs.org/type-check/-/type-check-0.3.2.tgz",  
2141: "integrity": "sha1-WITKtRLPHTVeP7eE8wgEsrUg23I=",  
2142: "dev": true,  
2143: "requires": {  
2144: "prelude-ls": "1.1.2"  
2145: }  
2146: },  
2147: "type-is": {  
2148: "version": "1.6.16",  
2149: "resolved": "https://registry.npmjs.org/type-is/-/type-is-1.6.16.tgz",  
2150: "integrity": "sha512-HRkVv/5qY2G6I8iab9cI7v1bOIdhm94dVjQCPFElW9W+3GeDOSHmy2EBYe4VTApuzolPcmgFTN3ftVJRKR2J9Q==",  
2151: "dev": true,  
2152: "requires": {  
2153: "media-typer": "0.3.0",  
2154: "mime-types": "2.1.18"  
2155: }  
2156: },  
2157: "uc.micro": {  
2158: "version": "1.0.5",  
2159: "resolved": "https://registry.npmjs.org/uc.micro/-/uc.micro-1.0.5.tgz",  
2160: "integrity": "sha512-JoLI4g5zv5qNyT09f4YAvEZIIV1oOjqnewYg5D38dkQljIzpPT296dbIGvKro3digYI1bkb7W6EP1y4uDlmzLg==",  
2161: "dev": true  
2162: },  
2163: "underscore": {  
2164: "version": "1.8.3",  
2165: "resolved": "https://registry.npmjs.org/underscore/-/underscore-1.8.3.tgz",  
2166: "integrity": "sha1-Tz+1OxBuYJf8+ctBCfKl6b36UCI=",  
2167: "dev": true  
2168: },  
2169: "underscore-contrib": {  
2170: "version": "0.3.0",  
2171: "resolved": "https://registry.npmjs.org/underscore-contrib/-/underscore-contrib-0.3.0.tgz",  
2172: "integrity": "sha1-ZltmwkeD+PorGMn4y7Dix9SMJsc=",  
2173: "dev": true,  
2174: "requires": {  
2175: "underscore": "1.6.0"  
2176: },  
2177: "dependencies": {  
2178: "underscore": {  
2179: "version": "1.6.0",  
2180: "resolved": "https://registry.npmjs.org/underscore/-/underscore-1.6.0.tgz",  
2181: "integrity": "sha1-izixDKze9jM3uLJOT/htRa6lKag=",  
2182: "dev": true  
2183: }  
2184: }  
2185: },  
2186: "universalify": {  
2187: "version": "0.1.1",  
2188: "resolved": "https://registry.npmjs.org/universalify/-/universalify-0.1.1.tgz",  
2189: "integrity": "sha1-+nG63UQ3r0wUiEHjs7Fl+enlkLc=",  
2190: "dev": true  
2191: },  
2192: "unpipe": {  
2193: "version": "1.0.0",  
2194: "resolved": "https://registry.npmjs.org/unpipe/-/unpipe-1.0.0.tgz",  
2195: "integrity": "sha1-sr9O6FFKrmFltIF4KdIbLvSZBOw=",  
2196: "dev": true  
2197: },  
2198: "util-deprecate": {  
2199: "version": "1.0.2",  
2200: "resolved": "https://registry.npmjs.org/util-deprecate/-/util-deprecate-1.0.2.tgz",  
2201: "integrity": "sha1-RQ1Nyfpw3nMnYvvS1KKJgUGaDM8=",  
2202: "dev": true  
2203: },  
2204: "utils-merge": {  
2205: "version": "1.0.1",  
2206: "resolved": "https://registry.npmjs.org/utils-merge/-/utils-merge-1.0.1.tgz",  
2207: "integrity": "sha1-n5VxD1CiZ5R7LMwSR0HBAoQn5xM=",  
2208: "dev": true  
2209: },  
2210: "uuid": {  
2211: "version": "3.2.1",  
2212: "resolved": "https://registry.npmjs.org/uuid/-/uuid-3.2.1.tgz",  
2213: "integrity": "sha512-jZnMwlb9Iku/O3smGWvZhauCf6cvvpKi4BKRiliS3cxnI+Gz9j5MEpTz2UFuXiKPJocb7gnsLHwiS05ige5BEA==",  
2214: "dev": true,  
2215: "optional": true  
2216: },  
2217: "vary": {  
2218: "version": "1.1.2",  
2219: "resolved": "https://registry.npmjs.org/vary/-/vary-1.1.2.tgz",  
2220: "integrity": "sha1-IpnwLG3tMNSllhsLn3RSShj2NPw=",  
2221: "dev": true  
2222: },  
2223: "verror": {  
2224: "version": "1.10.0",  
2225: "resolved": "https://registry.npmjs.org/verror/-/verror-1.10.0.tgz",  
2226: "integrity": "sha1-OhBcoXBTr1XW4nDB+CiGguGNpAA=",  
2227: "dev": true,  
2228: "optional": true,  
2229: "requires": {  
2230: "assert-plus": "1.0.0",  
2231: "core-util-is": "1.0.2",  
2232: "extsprintf": "1.3.0"  
2233: }  
2234: },  
2235: "webidl-conversions": {  
2236: "version": "2.0.1",  
2237: "resolved": "https://registry.npmjs.org/webidl-conversions/-/webidl-conversions-2.0.1.tgz",  
2238: "integrity": "sha1-O/glj30xjHRDw28uFpQCoaZwNQY=",  
2239: "dev": true,  
2240: "optional": true  
2241: },  
2242: "whatwg-url-compat": {  
2243: "version": "0.6.5",  
2244: "resolved": "https://registry.npmjs.org/whatwg-url-compat/-/whatwg-url-compat-0.6.5.tgz",  
2245: "integrity": "sha1-AImBEa9om7CXVBzVpFymyHmERb8=",  
2246: "dev": true,  
2247: "optional": true,  
2248: "requires": {  
2249: "tr46": "0.0.3"  
2250: }  
2251: },  
2252: "wordwrap": {  
2253: "version": "1.0.0",  
2254: "resolved": "https://registry.npmjs.org/wordwrap/-/wordwrap-1.0.0.tgz",  
2255: "integrity": "sha1-J1hIEIkUVqQXHI0CJkQa3pDLyus=",  
2256: "dev": true,  
2257: "optional": true  
2258: },  
2259: "wrappy": {  
2260: "version": "1.0.2",  
2261: "resolved": "https://registry.npmjs.org/wrappy/-/wrappy-1.0.2.tgz",  
2262: "integrity": "sha1-tSQ9jz7BqjXxNkYFvA0QNuMKtp8=",  
2263: "dev": true  
2264: },  
2265: "xml-name-validator": {  
2266: "version": "2.0.1",  
2267: "resolved": "https://registry.npmjs.org/xml-name-validator/-/xml-name-validator-2.0.1.tgz",  
2268: "integrity": "sha1-TYuPHszTQZqjYgYb7O9RXh5VljU=",  
2269: "dev": true,  
2270: "optional": true  
2271: },  
2272: "xmlcreate": {  
2273: "version": "1.0.2",  
2274: "resolved": "https://registry.npmjs.org/xmlcreate/-/xmlcreate-1.0.2.tgz",  
2275: "integrity": "sha1-+mv3YqYKQT+z3Y9LA8WyaSONMI8=",  
2276: "dev": true  
2277: },  
2278: "yui": {  
2279: "version": "3.18.1",  
2280: "resolved": "https://registry.npmjs.org/yui/-/yui-3.18.1.tgz",  
2281: "integrity": "sha1-4AAmnsCntvvHQcu4/L0OZRF7AUw=",  
2282: "dev": true,  
2283: "requires": {  
2284: "request": "2.40.0"  
2285: },  
2286: "dependencies": {  
2287: "asn1": {  
2288: "version": "0.1.11",  
2289: "resolved": "https://registry.npmjs.org/asn1/-/asn1-0.1.11.tgz",  
2290: "integrity": "sha1-VZvhg3bQik7E2+gId9J4GGObLfc=",  
2291: "dev": true,  
2292: "optional": true  
2293: },  
2294: "assert-plus": {  
2295: "version": "0.1.5",  
2296: "resolved": "https://registry.npmjs.org/assert-plus/-/assert-plus-0.1.5.tgz",  
2297: "integrity": "sha1-7nQAlBMALYTOxyGcasgRgS5yMWA=",  
2298: "dev": true,  
2299: "optional": true  
2300: },  
2301: "aws-sign2": {  
2302: "version": "0.5.0",  
2303: "resolved": "https://registry.npmjs.org/aws-sign2/-/aws-sign2-0.5.0.tgz",  
2304: "integrity": "sha1-xXED96F/wDfwLXwuZLYC6iI/fWM=",  
2305: "dev": true,  
2306: "optional": true  
2307: },  
2308: "combined-stream": {  
2309: "version": "0.0.7",  
2310: "resolved": "https://registry.npmjs.org/combined-stream/-/combined-stream-0.0.7.tgz",  
2311: "integrity": "sha1-ATfmV7qlp1QcV6w3rF/AfXO03B8=",  
2312: "dev": true,  
2313: "optional": true,  
2314: "requires": {  
2315: "delayed-stream": "0.0.5"  
2316: }  
2317: },  
2318: "delayed-stream": {  
2319: "version": "0.0.5",  
2320: "resolved": "https://registry.npmjs.org/delayed-stream/-/delayed-stream-0.0.5.tgz",  
2321: "integrity": "sha1-1LH0OpPoKW3+AmlPRoC8N6MTxz8=",  
2322: "dev": true,  
2323: "optional": true  
2324: },  
2325: "forever-agent": {  
2326: "version": "0.5.2",  
2327: "resolved": "https://registry.npmjs.org/forever-agent/-/forever-agent-0.5.2.tgz",  
2328: "integrity": "sha1-bQ4JxJIflKJ/Y9O0nF/v8epMUTA=",  
2329: "dev": true  
2330: },  
2331: "form-data": {  
2332: "version": "0.1.4",  
2333: "resolved": "https://registry.npmjs.org/form-data/-/form-data-0.1.4.tgz",  
2334: "integrity": "sha1-kavXiKupcCsaq/qLwBAxoqyeOxI=",  
2335: "dev": true,  
2336: "optional": true,  
2337: "requires": {  
2338: "async": "0.9.2",  
2339: "combined-stream": "0.0.7",  
2340: "mime": "1.2.11"  
2341: }  
2342: },  
2343: "http-signature": {  
2344: "version": "0.10.1",  
2345: "resolved": "https://registry.npmjs.org/http-signature/-/http-signature-0.10.1.tgz",  
2346: "integrity": "sha1-T72sEyVZqoMjEh5UB3nAoBKyfmY=",  
2347: "dev": true,  
2348: "optional": true,  
2349: "requires": {  
2350: "asn1": "0.1.11",  
2351: "assert-plus": "0.1.5",  
2352: "ctype": "0.5.3"  
2353: }  
2354: },  
2355: "mime": {  
2356: "version": "1.2.11",  
2357: "resolved": "https://registry.npmjs.org/mime/-/mime-1.2.11.tgz",  
2358: "integrity": "sha1-WCA+7Ybjpe8XrtK32evUfwpg3RA=",  
2359: "dev": true,  
2360: "optional": true  
2361: },  
2362: "mime-types": {  
2363: "version": "1.0.2",  
2364: "resolved": "https://registry.npmjs.org/mime-types/-/mime-types-1.0.2.tgz",  
2365: "integrity": "sha1-mVrhOSq4r/y/yyZB3QVOlDwNXc4=",  
2366: "dev": true  
2367: },  
2368: "node-uuid": {  
2369: "version": "1.4.8",  
2370: "resolved": "https://registry.npmjs.org/node-uuid/-/node-uuid-1.4.8.tgz",  
2371: "integrity": "sha1-sEDrCSOWivq/jTL7HxfxFn/auQc=",  
2372: "dev": true  
2373: },  
2374: "oauth-sign": {  
2375: "version": "0.3.0",  
2376: "resolved": "https://registry.npmjs.org/oauth-sign/-/oauth-sign-0.3.0.tgz",  
2377: "integrity": "sha1-y1QPk7srIqfVlBaRoojWDo6pOG4=",  
2378: "dev": true,  
2379: "optional": true  
2380: },  
2381: "qs": {  
2382: "version": "1.0.2",  
2383: "resolved": "https://registry.npmjs.org/qs/-/qs-1.0.2.tgz",  
2384: "integrity": "sha1-UKk+K1r2aRwxvOpdrnjubqGQN2g=",  
2385: "dev": true  
2386: },  
2387: "request": {  
2388: "version": "2.40.0",  
2389: "resolved": "https://registry.npmjs.org/request/-/request-2.40.0.tgz",  
2390: "integrity": "sha1-TdZw9pbx5uhC5mtLXoOTAaub62c=",  
2391: "dev": true,  
2392: "requires": {  
2393: "aws-sign2": "0.5.0",  
2394: "forever-agent": "0.5.2",  
2395: "form-data": "0.1.4",  
2396: "hawk": "1.1.1",  
2397: "http-signature": "0.10.1",  
2398: "json-stringify-safe": "5.0.1",  
2399: "mime-types": "1.0.2",  
2400: "node-uuid": "1.4.8",  
2401: "oauth-sign": "0.3.0",  
2402: "qs": "1.0.2",  
2403: "stringstream": "0.0.6",  
2404: "tough-cookie": "2.3.4",  
2405: "tunnel-agent": "0.4.3"  
2406: }  
2407: },  
2408: "tunnel-agent": {  
2409: "version": "0.4.3",  
2410: "resolved": "https://registry.npmjs.org/tunnel-agent/-/tunnel-agent-0.4.3.tgz",  
2411: "integrity": "sha1-Y3PbdpCf5XDgjXNYM2Xtgop07us=",  
2412: "dev": true,  
2413: "optional": true  
2414: }  
2415: }  
2416: },  
2417: "yuidocjs": {  
2418: "version": "0.10.2",  
2419: "resolved": "https://registry.npmjs.org/yuidocjs/-/yuidocjs-0.10.2.tgz",  
2420: "integrity": "sha1-M5JJZ85hkCTNcO9pTiZ9L5iPc/Y=",  
2421: "dev": true,  
2422: "requires": {  
2423: "express": "4.16.3",  
2424: "graceful-fs": "4.1.11",  
2425: "markdown-it": "4.4.0",  
2426: "mdn-links": "0.1.0",  
2427: "minimatch": "3.0.4",  
2428: "rimraf": "2.6.2",  
2429: "yui": "3.18.1"  
2430: }  
2431: }  
2432: }  
2433:}

## .\mobile\package.json

1:{  
 2: "name": "br.ufrj.coppetec.contdnit",  
 3: "version": "2.4.0",  
 4: "displayName": "Contagem Mobile",  
 5: "cordova": {  
 6: "platforms": [],  
 7: "plugins": {  
 8: "cordova-plugin-dialogs": {},  
 9: "cordova-plugin-whitelist": {},  
 10: "cordova-plugin-file": {},  
 11: "io.phasr.cordova.plugin.itunesfilesharing": {},  
 12: "cordova-plugin-splashscreen": {},  
 13: "cordova-plugin-statusbar": {},  
 14: "cordova-sqlite-storage": {},  
 15: "cordova-plugin-battery-status": {},  
 16: "cordova-plugin-cache-clear": {},  
 17: "cordova-plugin-device": {}  
 18: }  
 19: },  
 20: "dependencies": {  
 21: "cordova-plugin-battery-status": "^1.2.4",  
 22: "cordova-plugin-cache-clear": "^1.3.7",  
 23: "cordova-plugin-compat": "^1.1.0",  
 24: "cordova-plugin-device": "^1.1.6",  
 25: "cordova-plugin-dialogs": "~1.3.3",  
 26: "cordova-plugin-file": "^4.3.3",  
 27: "cordova-plugin-itunesfilesharing": "0.0.1",  
 28: "cordova-plugin-splashscreen": "^4.0.3",  
 29: "cordova-plugin-statusbar": "^2.2.3",  
 30: "cordova-plugin-whitelist": "1",  
 31: "cordova-sqlite-storage": "^2.0.4",  
 32: "io.phasr.cordova.plugin.itunesfilesharing": "~0.0.1"  
 33: }  
 34:}

## .\mobile\platforms\platforms.json

1:{}

## .\mobile\www\css\bootstrap.min.css

1:/\*!  
 2: \* Bootstrap v3.3.5 (http://getbootstrap.com)  
 3: \* Copyright 2011-2016 Twitter, Inc.  
 4: \* Licensed under MIT (https://github.com/twbs/bootstrap/blob/master/LICENSE)  
 5: \*/  
 6:  
 7:/\*!  
 8: \* Generated using the Bootstrap Customizer (http://getbootstrap.com/customize/?id=bfda3ef09892c3ccfc142f0544041b19)  
 9: \* Config saved to config.json and https://gist.github.com/bfda3ef09892c3ccfc142f0544041b19  
 10: \*//\*!  
 11: \* Bootstrap v3.3.6 (http://getbootstrap.com)  
 12: \* Copyright 2011-2015 Twitter, Inc.  
 13: \* Licensed under MIT (https://github.com/twbs/bootstrap/blob/master/LICENSE)  
 14: \*//\*! normalize.css v3.0.3 | MIT License | github.com/necolas/normalize.css \*/html{font-family:sans-serif;-ms-text-size-adjust:100%;-webkit-text-size-adjust:100%}body{margin:0}article,aside,details,figcaption,figure,footer,header,hgroup,main,menu,nav,section,summary{display:block}audio,canvas,progress,video{display:inline-block;vertical-align:baseline}audio:not([controls]){display:none;height:0}[hidden],template{display:none}a{background-color:transparent}a:active,a:hover{outline:0}abbr[title]{border-bottom:1px dotted}b,strong{font-weight:bold}dfn{font-style:italic}h1{font-size:2em;margin:0.67em 0}mark{background:#ff0;color:#000}small{font-size:80%}sub,sup{font-size:75%;line-height:0;position:relative;vertical-align:baseline}sup{top:-0.5em}sub{bottom:-0.25em}img{border:0}svg:not(:root){overflow:hidden}figure{margin:1em 40px}hr{-webkit-box-sizing:content-box;-moz-box-sizing:content-box;box-sizing:content-box;height:0}pre{overflow:auto}code,kbd,pre,samp{font-family:monospace, monospace;font-size:1em}button,input,optgroup,select,textarea{color:inherit;font:inherit;margin:0}button{overflow:visible}button,select{text-transform:none}button,html input[type="button"],input[type="reset"],input[type="submit"]{-webkit-appearance:button;cursor:pointer}button[disabled],html input[disabled]{cursor:default}button::-moz-focus-inner,input::-moz-focus-inner{border:0;padding:0}input{line-height:normal}input[type="checkbox"],input[type="radio"]{-webkit-box-sizing:border-box;-moz-box-sizing:border-box;box-sizing:border-box;padding:0}input[type="number"]::-webkit-inner-spin-button,input[type="number"]::-webkit-outer-spin-button{height:auto}input[type="search"]{-webkit-appearance:textfield;-webkit-box-sizing:content-box;-moz-box-sizing:content-box;box-sizing:content-box}input[type="search"]::-webkit-search-cancel-button,input[type="search"]::-webkit-search-decoration{-webkit-appearance:none}fieldset{border:1px solid #c0c0c0;margin:0 2px;padding:0.35em 0.625em 0.75em}legend{border:0;padding:0}textarea{overflow:auto}optgroup{font-weight:bold}table{border-collapse:collapse;border-spacing:0}td,th{padding:0}\*{-webkit-box-sizing:border-box;-moz-box-sizing:border-box;box-sizing:border-box}\*:before,\*:after{-webkit-box-sizing:border-box;-moz-box-sizing:border-box;box-sizing:border-box}html{font-size:10px;-webkit-tap-highlight-color:rgba(0,0,0,0)}body{font-family:"Helvetica Neue",Helvetica,Arial,sans-serif;font-size:14px;line-height:1.42857143;color:#333;background-color:#fff}input,button,select,textarea{font-family:inherit;font-size:inherit;line-height:inherit}a{color:#337ab7;text-decoration:none}a:hover,a:focus{color:#23527c;text-decoration:underline}a:focus{outline:thin dotted;outline:5px auto -webkit-focus-ring-color;outline-offset:-2px}figure{margin:0}img{vertical-align:middle}.img-responsive{display:block;max-width:100%;height:auto}.img-rounded{border-radius:6px}.img-thumbnail{padding:4px;line-height:1.42857143;background-color:#fff;border:1px solid #ddd;border-radius:4px;-webkit-transition:all .2s ease-in-out;-o-transition:all .2s ease-in-out;transition:all .2s ease-in-out;display:inline-block;max-width:100%;height:auto}.img-circle{border-radius:50%}hr{margin-top:20px;margin-bottom:20px;border:0;border-top:1px solid #eee}.sr-only{position:absolute;width:1px;height:1px;margin:-1px;padding:0;overflow:hidden;clip:rect(0, 0, 0, 0);border:0}.sr-only-focusable:active,.sr-only-focusable:focus{position:static;width:auto;height:auto;margin:0;overflow:visible;clip:auto}[role="button"]{cursor:pointer}.container{margin-right:auto;margin-left:auto;padding-left:15px;padding-right:15px}@media (min-width:768px){.container{width:750px}}@media (min-width:992px){.container{width:970px}}@media (min-width:1200px){.container{width:1170px}}.container-fluid{margin-right:auto;margin-left:auto;padding-left:15px;padding-right:15px}.row{margin-left:-15px;margin-right:-15px}.col-xs-1, .col-sm-1, .col-md-1, .col-lg-1, .col-xs-2, .col-sm-2, .col-md-2, .col-lg-2, .col-xs-3, .col-sm-3, .col-md-3, .col-lg-3, .col-xs-4, .col-sm-4, .col-md-4, .col-lg-4, .col-xs-5, .col-sm-5, .col-md-5, .col-lg-5, .col-xs-6, .col-sm-6, .col-md-6, .col-lg-6, .col-xs-7, .col-sm-7, .col-md-7, .col-lg-7, .col-xs-8, .col-sm-8, .col-md-8, .col-lg-8, .col-xs-9, .col-sm-9, .col-md-9, .col-lg-9, .col-xs-10, .col-sm-10, .col-md-10, .col-lg-10, .col-xs-11, .col-sm-11, .col-md-11, .col-lg-11, .col-xs-12, .col-sm-12, .col-md-12, .col-lg-12{position:relative;min-height:1px;padding-left:15px;padding-right:15px}.col-xs-1, .col-xs-2, .col-xs-3, .col-xs-4, .col-xs-5, .col-xs-6, .col-xs-7, .col-xs-8, .col-xs-9, .col-xs-10, .col-xs-11, .col-xs-12{float:left}.col-xs-12{width:100%}.col-xs-11{width:91.66666667%}.col-xs-10{width:83.33333333%}.col-xs-9{width:75%}.col-xs-8{width:66.66666667%}.col-xs-7{width:58.33333333%}.col-xs-6{width:50%}.col-xs-5{width:41.66666667%}.col-xs-4{width:33.33333333%}.col-xs-3{width:25%}.col-xs-2{width:16.66666667%}.col-xs-1{width:8.33333333%}.col-xs-pull-12{right:100%}.col-xs-pull-11{right:91.66666667%}.col-xs-pull-10{right:83.33333333%}.col-xs-pull-9{right:75%}.col-xs-pull-8{right:66.66666667%}.col-xs-pull-7{right:58.33333333%}.col-xs-pull-6{right:50%}.col-xs-pull-5{right:41.66666667%}.col-xs-pull-4{right:33.33333333%}.col-xs-pull-3{right:25%}.col-xs-pull-2{right:16.66666667%}.col-xs-pull-1{right:8.33333333%}.col-xs-pull-0{right:auto}.col-xs-push-12{left:100%}.col-xs-push-11{left:91.66666667%}.col-xs-push-10{left:83.33333333%}.col-xs-push-9{left:75%}.col-xs-push-8{left:66.66666667%}.col-xs-push-7{left:58.33333333%}.col-xs-push-6{left:50%}.col-xs-push-5{left:41.66666667%}.col-xs-push-4{left:33.33333333%}.col-xs-push-3{left:25%}.col-xs-push-2{left:16.66666667%}.col-xs-push-1{left:8.33333333%}.col-xs-push-0{left:auto}.col-xs-offset-12{margin-left:100%}.col-xs-offset-11{margin-left:91.66666667%}.col-xs-offset-10{margin-left:83.33333333%}.col-xs-offset-9{margin-left:75%}.col-xs-offset-8{margin-left:66.66666667%}.col-xs-offset-7{margin-left:58.33333333%}.col-xs-offset-6{margin-left:50%}.col-xs-offset-5{margin-left:41.66666667%}.col-xs-offset-4{margin-left:33.33333333%}.col-xs-offset-3{margin-left:25%}.col-xs-offset-2{margin-left:16.66666667%}.col-xs-offset-1{margin-left:8.33333333%}.col-xs-offset-0{margin-left:0}@media (min-width:768px){.col-sm-1, .col-sm-2, .col-sm-3, .col-sm-4, .col-sm-5, .col-sm-6, .col-sm-7, .col-sm-8, .col-sm-9, .col-sm-10, .col-sm-11, .col-sm-12{float:left}.col-sm-12{width:100%}.col-sm-11{width:91.66666667%}.col-sm-10{width:83.33333333%}.col-sm-9{width:75%}.col-sm-8{width:66.66666667%}.col-sm-7{width:58.33333333%}.col-sm-6{width:50%}.col-sm-5{width:41.66666667%}.col-sm-4{width:33.33333333%}.col-sm-3{width:25%}.col-sm-2{width:16.66666667%}.col-sm-1{width:8.33333333%}.col-sm-pull-12{right:100%}.col-sm-pull-11{right:91.66666667%}.col-sm-pull-10{right:83.33333333%}.col-sm-pull-9{right:75%}.col-sm-pull-8{right:66.66666667%}.col-sm-pull-7{right:58.33333333%}.col-sm-pull-6{right:50%}.col-sm-pull-5{right:41.66666667%}.col-sm-pull-4{right:33.33333333%}.col-sm-pull-3{right:25%}.col-sm-pull-2{right:16.66666667%}.col-sm-pull-1{right:8.33333333%}.col-sm-pull-0{right:auto}.col-sm-push-12{left:100%}.col-sm-push-11{left:91.66666667%}.col-sm-push-10{left:83.33333333%}.col-sm-push-9{left:75%}.col-sm-push-8{left:66.66666667%}.col-sm-push-7{left:58.33333333%}.col-sm-push-6{left:50%}.col-sm-push-5{left:41.66666667%}.col-sm-push-4{left:33.33333333%}.col-sm-push-3{left:25%}.col-sm-push-2{left:16.66666667%}.col-sm-push-1{left:8.33333333%}.col-sm-push-0{left:auto}.col-sm-offset-12{margin-left:100%}.col-sm-offset-11{margin-left:91.66666667%}.col-sm-offset-10{margin-left:83.33333333%}.col-sm-offset-9{margin-left:75%}.col-sm-offset-8{margin-left:66.66666667%}.col-sm-offset-7{margin-left:58.33333333%}.col-sm-offset-6{margin-left:50%}.col-sm-offset-5{margin-left:41.66666667%}.col-sm-offset-4{margin-left:33.33333333%}.col-sm-offset-3{margin-left:25%}.col-sm-offset-2{margin-left:16.66666667%}.col-sm-offset-1{margin-left:8.33333333%}.col-sm-offset-0{margin-left:0}}@media (min-width:992px){.col-md-1, .col-md-2, .col-md-3, .col-md-4, .col-md-5, .col-md-6, .col-md-7, .col-md-8, .col-md-9, .col-md-10, .col-md-11, .col-md-12{float:left}.col-md-12{width:100%}.col-md-11{width:91.66666667%}.col-md-10{width:83.33333333%}.col-md-9{width:75%}.col-md-8{width:66.66666667%}.col-md-7{width:58.33333333%}.col-md-6{width:50%}.col-md-5{width:41.66666667%}.col-md-4{width:33.33333333%}.col-md-3{width:25%}.col-md-2{width:16.66666667%}.col-md-1{width:8.33333333%}.col-md-pull-12{right:100%}.col-md-pull-11{right:91.66666667%}.col-md-pull-10{right:83.33333333%}.col-md-pull-9{right:75%}.col-md-pull-8{right:66.66666667%}.col-md-pull-7{right:58.33333333%}.col-md-pull-6{right:50%}.col-md-pull-5{right:41.66666667%}.col-md-pull-4{right:33.33333333%}.col-md-pull-3{right:25%}.col-md-pull-2{right:16.66666667%}.col-md-pull-1{right:8.33333333%}.col-md-pull-0{right:auto}.col-md-push-12{left:100%}.col-md-push-11{left:91.66666667%}.col-md-push-10{left:83.33333333%}.col-md-push-9{left:75%}.col-md-push-8{left:66.66666667%}.col-md-push-7{left:58.33333333%}.col-md-push-6{left:50%}.col-md-push-5{left:41.66666667%}.col-md-push-4{left:33.33333333%}.col-md-push-3{left:25%}.col-md-push-2{left:16.66666667%}.col-md-push-1{left:8.33333333%}.col-md-push-0{left:auto}.col-md-offset-12{margin-left:100%}.col-md-offset-11{margin-left:91.66666667%}.col-md-offset-10{margin-left:83.33333333%}.col-md-offset-9{margin-left:75%}.col-md-offset-8{margin-left:66.66666667%}.col-md-offset-7{margin-left:58.33333333%}.col-md-offset-6{margin-left:50%}.col-md-offset-5{margin-left:41.66666667%}.col-md-offset-4{margin-left:33.33333333%}.col-md-offset-3{margin-left:25%}.col-md-offset-2{margin-left:16.66666667%}.col-md-offset-1{margin-left:8.33333333%}.col-md-offset-0{margin-left:0}}@media (min-width:1200px){.col-lg-1, .col-lg-2, .col-lg-3, .col-lg-4, .col-lg-5, .col-lg-6, .col-lg-7, .col-lg-8, .col-lg-9, .col-lg-10, .col-lg-11, .col-lg-12{float:left}.col-lg-12{width:100%}.col-lg-11{width:91.66666667%}.col-lg-10{width:83.33333333%}.col-lg-9{width:75%}.col-lg-8{width:66.66666667%}.col-lg-7{width:58.33333333%}.col-lg-6{width:50%}.col-lg-5{width:41.66666667%}.col-lg-4{width:33.33333333%}.col-lg-3{width:25%}.col-lg-2{width:16.66666667%}.col-lg-1{width:8.33333333%}.col-lg-pull-12{right:100%}.col-lg-pull-11{right:91.66666667%}.col-lg-pull-10{right:83.33333333%}.col-lg-pull-9{right:75%}.col-lg-pull-8{right:66.66666667%}.col-lg-pull-7{right:58.33333333%}.col-lg-pull-6{right:50%}.col-lg-pull-5{right:41.66666667%}.col-lg-pull-4{right:33.33333333%}.col-lg-pull-3{right:25%}.col-lg-pull-2{right:16.66666667%}.col-lg-pull-1{right:8.33333333%}.col-lg-pull-0{right:auto}.col-lg-push-12{left:100%}.col-lg-push-11{left:91.66666667%}.col-lg-push-10{left:83.33333333%}.col-lg-push-9{left:75%}.col-lg-push-8{left:66.66666667%}.col-lg-push-7{left:58.33333333%}.col-lg-push-6{left:50%}.col-lg-push-5{left:41.66666667%}.col-lg-push-4{left:33.33333333%}.col-lg-push-3{left:25%}.col-lg-push-2{left:16.66666667%}.col-lg-push-1{left:8.33333333%}.col-lg-push-0{left:auto}.col-lg-offset-12{margin-left:100%}.col-lg-offset-11{margin-left:91.66666667%}.col-lg-offset-10{margin-left:83.33333333%}.col-lg-offset-9{margin-left:75%}.col-lg-offset-8{margin-left:66.66666667%}.col-lg-offset-7{margin-left:58.33333333%}.col-lg-offset-6{margin-left:50%}.col-lg-offset-5{margin-left:41.66666667%}.col-lg-offset-4{margin-left:33.33333333%}.col-lg-offset-3{margin-left:25%}.col-lg-offset-2{margin-left:16.66666667%}.col-lg-offset-1{margin-left:8.33333333%}.col-lg-offset-0{margin-left:0}}.btn{display:inline-block;margin-bottom:0;font-weight:normal;text-align:center;vertical-align:middle;-ms-touch-action:manipulation;touch-action:manipulation;cursor:pointer;background-image:none;border:1px solid transparent;white-space:nowrap;padding:6px 12px;font-size:14px;line-height:1.42857143;border-radius:4px;-webkit-user-select:none;-moz-user-select:none;-ms-user-select:none;user-select:none}.btn:focus,.btn:active:focus,.btn.active:focus,.btn.focus,.btn:active.focus,.btn.active.focus{outline:thin dotted;outline:5px auto -webkit-focus-ring-color;outline-offset:-2px}.btn:hover,.btn:focus,.btn.focus{color:#333;text-decoration:none}.btn:active,.btn.active{outline:0;background-image:none;-webkit-box-shadow:inset 0 3px 5px rgba(0,0,0,0.125);box-shadow:inset 0 3px 5px rgba(0,0,0,0.125)}.btn.disabled,.btn[disabled],fieldset[disabled] .btn{cursor:not-allowed;opacity:.65;filter:alpha(opacity=65);-webkit-box-shadow:none;box-shadow:none}a.btn.disabled,fieldset[disabled] a.btn{pointer-events:none}.btn-default{color:#333;background-color:#fff;border-color:#ccc}.btn-default:focus,.btn-default.focus{color:#333;background-color:#e6e6e6;border-color:#8c8c8c}.btn-default:hover{color:#333;background-color:#e6e6e6;border-color:#adadad}.btn-default:active,.btn-default.active,.open>.dropdown-toggle.btn-default{color:#333;background-color:#e6e6e6;border-color:#adadad}.btn-default:active:hover,.btn-default.active:hover,.open>.dropdown-toggle.btn-default:hover,.btn-default:active:focus,.btn-default.active:focus,.open>.dropdown-toggle.btn-default:focus,.btn-default:active.focus,.btn-default.active.focus,.open>.dropdown-toggle.btn-default.focus{color:#333;background-color:#d4d4d4;border-color:#8c8c8c}.btn-default:active,.btn-default.active,.open>.dropdown-toggle.btn-default{background-image:none}.btn-default.disabled:hover,.btn-default[disabled]:hover,fieldset[disabled] .btn-default:hover,.btn-default.disabled:focus,.btn-default[disabled]:focus,fieldset[disabled] .btn-default:focus,.btn-default.disabled.focus,.btn-default[disabled].focus,fieldset[disabled] .btn-default.focus{background-color:#fff;border-color:#ccc}.btn-default .badge{color:#fff;background-color:#333}.btn-primary{color:#fff;background-color:#337ab7;border-color:#2e6da4}.btn-primary:focus,.btn-primary.focus{color:#fff;background-color:#286090;border-color:#122b40}.btn-primary:hover{color:#fff;background-color:#286090;border-color:#204d74}.btn-primary:active,.btn-primary.active,.open>.dropdown-toggle.btn-primary{color:#fff;background-color:#286090;border-color:#204d74}.btn-primary:active:hover,.btn-primary.active:hover,.open>.dropdown-toggle.btn-primary:hover,.btn-primary:active:focus,.btn-primary.active:focus,.open>.dropdown-toggle.btn-primary:focus,.btn-primary:active.focus,.btn-primary.active.focus,.open>.dropdown-toggle.btn-primary.focus{color:#fff;background-color:#204d74;border-color:#122b40}.btn-primary:active,.btn-primary.active,.open>.dropdown-toggle.btn-primary{background-image:none}.btn-primary.disabled:hover,.btn-primary[disabled]:hover,fieldset[disabled] .btn-primary:hover,.btn-primary.disabled:focus,.btn-primary[disabled]:focus,fieldset[disabled] .btn-primary:focus,.btn-primary.disabled.focus,.btn-primary[disabled].focus,fieldset[disabled] .btn-primary.focus{background-color:#337ab7;border-color:#2e6da4}.btn-primary .badge{color:#337ab7;background-color:#fff}.btn-success{color:#fff;background-color:#5cb85c;border-color:#4cae4c}.btn-success:focus,.btn-success.focus{color:#fff;background-color:#449d44;border-color:#255625}.btn-success:hover{color:#fff;background-color:#449d44;border-color:#398439}.btn-success:active,.btn-success.active,.open>.dropdown-toggle.btn-success{color:#fff;background-color:#449d44;border-color:#398439}.btn-success:active:hover,.btn-success.active:hover,.open>.dropdown-toggle.btn-success:hover,.btn-success:active:focus,.btn-success.active:focus,.open>.dropdown-toggle.btn-success:focus,.btn-success:active.focus,.btn-success.active.focus,.open>.dropdown-toggle.btn-success.focus{color:#fff;background-color:#398439;border-color:#255625}.btn-success:active,.btn-success.active,.open>.dropdown-toggle.btn-success{background-image:none}.btn-success.disabled:hover,.btn-success[disabled]:hover,fieldset[disabled] .btn-success:hover,.btn-success.disabled:focus,.btn-success[disabled]:focus,fieldset[disabled] .btn-success:focus,.btn-success.disabled.focus,.btn-success[disabled].focus,fieldset[disabled] .btn-success.focus{background-color:#5cb85c;border-color:#4cae4c}.btn-success .badge{color:#5cb85c;background-color:#fff}.btn-info{color:#fff;background-color:#5bc0de;border-color:#46b8da}.btn-info:focus,.btn-info.focus{color:#fff;background-color:#31b0d5;border-color:#1b6d85}.btn-info:hover{color:#fff;background-color:#31b0d5;border-color:#269abc}.btn-info:active,.btn-info.active,.open>.dropdown-toggle.btn-info{color:#fff;background-color:#31b0d5;border-color:#269abc}.btn-info:active:hover,.btn-info.active:hover,.open>.dropdown-toggle.btn-info:hover,.btn-info:active:focus,.btn-info.active:focus,.open>.dropdown-toggle.btn-info:focus,.btn-info:active.focus,.btn-info.active.focus,.open>.dropdown-toggle.btn-info.focus{color:#fff;background-color:#269abc;border-color:#1b6d85}.btn-info:active,.btn-info.active,.open>.dropdown-toggle.btn-info{background-image:none}.btn-info.disabled:hover,.btn-info[disabled]:hover,fieldset[disabled] .btn-info:hover,.btn-info.disabled:focus,.btn-info[disabled]:focus,fieldset[disabled] .btn-info:focus,.btn-info.disabled.focus,.btn-info[disabled].focus,fieldset[disabled] .btn-info.focus{background-color:#5bc0de;border-color:#46b8da}.btn-info .badge{color:#5bc0de;background-color:#fff}.btn-warning{color:#fff;background-color:#f0ad4e;border-color:#eea236}.btn-warning:focus,.btn-warning.focus{color:#fff;background-color:#ec971f;border-color:#985f0d}.btn-warning:hover{color:#fff;background-color:#ec971f;border-color:#d58512}.btn-warning:active,.btn-warning.active,.open>.dropdown-toggle.btn-warning{color:#fff;background-color:#ec971f;border-color:#d58512}.btn-warning:active:hover,.btn-warning.active:hover,.open>.dropdown-toggle.btn-warning:hover,.btn-warning:active:focus,.btn-warning.active:focus,.open>.dropdown-toggle.btn-warning:focus,.btn-warning:active.focus,.btn-warning.active.focus,.open>.dropdown-toggle.btn-warning.focus{color:#fff;background-color:#d58512;border-color:#985f0d}.btn-warning:active,.btn-warning.active,.open>.dropdown-toggle.btn-warning{background-image:none}.btn-warning.disabled:hover,.btn-warning[disabled]:hover,fieldset[disabled] .btn-warning:hover,.btn-warning.disabled:focus,.btn-warning[disabled]:focus,fieldset[disabled] .btn-warning:focus,.btn-warning.disabled.focus,.btn-warning[disabled].focus,fieldset[disabled] .btn-warning.focus{background-color:#f0ad4e;border-color:#eea236}.btn-warning .badge{color:#f0ad4e;background-color:#fff}.btn-danger{color:#fff;background-color:#d9534f;border-color:#d43f3a}.btn-danger:focus,.btn-danger.focus{color:#fff;background-color:#c9302c;border-color:#761c19}.btn-danger:hover{color:#fff;background-color:#c9302c;border-color:#ac2925}.btn-danger:active,.btn-danger.active,.open>.dropdown-toggle.btn-danger{color:#fff;background-color:#c9302c;border-color:#ac2925}.btn-danger:active:hover,.btn-danger.active:hover,.open>.dropdown-toggle.btn-danger:hover,.btn-danger:active:focus,.btn-danger.active:focus,.open>.dropdown-toggle.btn-danger:focus,.btn-danger:active.focus,.btn-danger.active.focus,.open>.dropdown-toggle.btn-danger.focus{color:#fff;background-color:#ac2925;border-color:#761c19}.btn-danger:active,.btn-danger.active,.open>.dropdown-toggle.btn-danger{background-image:none}.btn-danger.disabled:hover,.btn-danger[disabled]:hover,fieldset[disabled] .btn-danger:hover,.btn-danger.disabled:focus,.btn-danger[disabled]:focus,fieldset[disabled] .btn-danger:focus,.btn-danger.disabled.focus,.btn-danger[disabled].focus,fieldset[disabled] .btn-danger.focus{background-color:#d9534f;border-color:#d43f3a}.btn-danger .badge{color:#d9534f;background-color:#fff}.btn-link{color:#337ab7;font-weight:normal;border-radius:0}.btn-link,.btn-link:active,.btn-link.active,.btn-link[disabled],fieldset[disabled] .btn-link{background-color:transparent;-webkit-box-shadow:none;box-shadow:none}.btn-link,.btn-link:hover,.btn-link:focus,.btn-link:active{border-color:transparent}.btn-link:hover,.btn-link:focus{color:#23527c;text-decoration:underline;background-color:transparent}.btn-link[disabled]:hover,fieldset[disabled] .btn-link:hover,.btn-link[disabled]:focus,fieldset[disabled] .btn-link:focus{color:#777;text-decoration:none}.btn-lg{padding:10px 16px;font-size:18px;line-height:1.3333333;border-radius:6px}.btn-sm{padding:5px 10px;font-size:12px;line-height:1.5;border-radius:3px}.btn-xs{padding:1px 5px;font-size:12px;line-height:1.5;border-radius:3px}.btn-block{display:block;width:100%}.btn-block+.btn-block{margin-top:5px}input[type="submit"].btn-block,input[type="reset"].btn-block,input[type="button"].btn-block{width:100%}.clearfix:before,.clearfix:after,.container:before,.container:after,.container-fluid:before,.container-fluid:after,.row:before,.row:after{content:" ";display:table}.clearfix:after,.container:after,.container-fluid:after,.row:after{clear:both}.center-block{display:block;margin-left:auto;margin-right:auto}.pull-right{float:right !important}.pull-left{float:left !important}.hide{display:none !important}.show{display:block !important}.invisible{visibility:hidden}.text-hide{font:0/0 a;color:transparent;text-shadow:none;background-color:transparent;border:0}.hidden{display:none !important}.affix{position:fixed}

## .\mobile\www\css\bootstrap\textunderscore buttons.css

1:.btn {  
 2: display: inline-block;  
 3: margin-bottom: 0;  
 4: font-weight: normal;  
 5: text-align: center;  
 6: vertical-align: middle;  
 7: -ms-touch-action: manipulation;  
 8: touch-action: manipulation;  
 9: cursor: pointer;  
 10: background-image: none;  
 11: border: 1px solid transparent;  
 12: white-space: nowrap;  
 13: padding: 6px 12px;  
 14: font-size: 14px;  
 15: line-height: 1.42857143;  
 16: border-radius: 4px;  
 17: -webkit-user-select: none;  
 18: -moz-user-select: none;  
 19: -ms-user-select: none;  
 20: user-select: none;  
 21:}  
 22:.btn:focus,  
 23:.btn:active:focus,  
 24:.btn.active:focus,  
 25:.btn.focus,  
 26:.btn:active.focus,  
 27:.btn.active.focus {  
 28: outline: thin dotted;  
 29: outline: 5px auto -webkit-focus-ring-color;  
 30: outline-offset: -2px;  
 31:}  
 32:.btn:hover,  
 33:.btn:focus,  
 34:.btn.focus {  
 35: color: #333333;  
 36: text-decoration: none;  
 37:}  
 38:.btn:active,  
 39:.btn.active {  
 40: outline: 0;  
 41: background-image: none;  
 42: -webkit-box-shadow: inset 0 3px 5px rgba(0, 0, 0, 0.125);  
 43: box-shadow: inset 0 3px 5px rgba(0, 0, 0, 0.125);  
 44:}  
 45:.btn.disabled,  
 46:.btn[disabled],  
 47:fieldset[disabled] .btn {  
 48: cursor: not-allowed;  
 49: opacity: 0.65;  
 50: filter: alpha(opacity=65);  
 51: -webkit-box-shadow: none;  
 52: box-shadow: none;  
 53:}  
 54:a.btn.disabled,  
 55:fieldset[disabled] a.btn {  
 56: pointer-events: none;  
 57:}  
 58:.btn-default {  
 59: color: #333333;  
 60: background-color: #ffffff;  
 61: border-color: #cccccc;  
 62:}  
 63:.btn-default:focus,  
 64:.btn-default.focus {  
 65: color: #333333;  
 66: background-color: #e6e6e6;  
 67: border-color: #8c8c8c;  
 68:}  
 69:.btn-default:hover {  
 70: color: #333333;  
 71: background-color: #e6e6e6;  
 72: border-color: #adadad;  
 73:}  
 74:.btn-default:active,  
 75:.btn-default.active,  
 76:.open > .dropdown-toggle.btn-default {  
 77: color: #333333;  
 78: background-color: #e6e6e6;  
 79: border-color: #adadad;  
 80:}  
 81:.btn-default:active:hover,  
 82:.btn-default.active:hover,  
 83:.open > .dropdown-toggle.btn-default:hover,  
 84:.btn-default:active:focus,  
 85:.btn-default.active:focus,  
 86:.open > .dropdown-toggle.btn-default:focus,  
 87:.btn-default:active.focus,  
 88:.btn-default.active.focus,  
 89:.open > .dropdown-toggle.btn-default.focus {  
 90: color: #333333;  
 91: background-color: #d4d4d4;  
 92: border-color: #8c8c8c;  
 93:}  
 94:.btn-default:active,  
 95:.btn-default.active,  
 96:.open > .dropdown-toggle.btn-default {  
 97: background-image: none;  
 98:}  
 99:.btn-default.disabled:hover,  
 100:.btn-default[disabled]:hover,  
 101:fieldset[disabled] .btn-default:hover,  
 102:.btn-default.disabled:focus,  
 103:.btn-default[disabled]:focus,  
 104:fieldset[disabled] .btn-default:focus,  
 105:.btn-default.disabled.focus,  
 106:.btn-default[disabled].focus,  
 107:fieldset[disabled] .btn-default.focus {  
 108: background-color: #ffffff;  
 109: border-color: #cccccc;  
 110:}  
 111:.btn-default .badge {  
 112: color: #ffffff;  
 113: background-color: #333333;  
 114:}  
 115:.btn-primary {  
 116: color: #ffffff;  
 117: background-color: #337ab7;  
 118: border-color: #2e6da4;  
 119:}  
 120:.btn-primary:focus,  
 121:.btn-primary.focus {  
 122: color: #ffffff;  
 123: background-color: #286090;  
 124: border-color: #122b40;  
 125:}  
 126:.btn-primary:hover {  
 127: color: #ffffff;  
 128: background-color: #286090;  
 129: border-color: #204d74;  
 130:}  
 131:.btn-primary:active,  
 132:.btn-primary.active,  
 133:.open > .dropdown-toggle.btn-primary {  
 134: color: #ffffff;  
 135: background-color: #286090;  
 136: border-color: #204d74;  
 137:}  
 138:.btn-primary:active:hover,  
 139:.btn-primary.active:hover,  
 140:.open > .dropdown-toggle.btn-primary:hover,  
 141:.btn-primary:active:focus,  
 142:.btn-primary.active:focus,  
 143:.open > .dropdown-toggle.btn-primary:focus,  
 144:.btn-primary:active.focus,  
 145:.btn-primary.active.focus,  
 146:.open > .dropdown-toggle.btn-primary.focus {  
 147: color: #ffffff;  
 148: background-color: #204d74;  
 149: border-color: #122b40;  
 150:}  
 151:.btn-primary:active,  
 152:.btn-primary.active,  
 153:.open > .dropdown-toggle.btn-primary {  
 154: background-image: none;  
 155:}  
 156:.btn-primary.disabled:hover,  
 157:.btn-primary[disabled]:hover,  
 158:fieldset[disabled] .btn-primary:hover,  
 159:.btn-primary.disabled:focus,  
 160:.btn-primary[disabled]:focus,  
 161:fieldset[disabled] .btn-primary:focus,  
 162:.btn-primary.disabled.focus,  
 163:.btn-primary[disabled].focus,  
 164:fieldset[disabled] .btn-primary.focus {  
 165: background-color: #337ab7;  
 166: border-color: #2e6da4;  
 167:}  
 168:.btn-primary .badge {  
 169: color: #337ab7;  
 170: background-color: #ffffff;  
 171:}  
 172:.btn-success {  
 173: color: #ffffff;  
 174: background-color: #5cb85c;  
 175: border-color: #4cae4c;  
 176:}  
 177:.btn-success:focus,  
 178:.btn-success.focus {  
 179: color: #ffffff;  
 180: background-color: #449d44;  
 181: border-color: #255625;  
 182:}  
 183:.btn-success:hover {  
 184: color: #ffffff;  
 185: background-color: #449d44;  
 186: border-color: #398439;  
 187:}  
 188:.btn-success:active,  
 189:.btn-success.active,  
 190:.open > .dropdown-toggle.btn-success {  
 191: color: #ffffff;  
 192: background-color: #449d44;  
 193: border-color: #398439;  
 194:}  
 195:.btn-success:active:hover,  
 196:.btn-success.active:hover,  
 197:.open > .dropdown-toggle.btn-success:hover,  
 198:.btn-success:active:focus,  
 199:.btn-success.active:focus,  
 200:.open > .dropdown-toggle.btn-success:focus,  
 201:.btn-success:active.focus,  
 202:.btn-success.active.focus,  
 203:.open > .dropdown-toggle.btn-success.focus {  
 204: color: #ffffff;  
 205: background-color: #398439;  
 206: border-color: #255625;  
 207:}  
 208:.btn-success:active,  
 209:.btn-success.active,  
 210:.open > .dropdown-toggle.btn-success {  
 211: background-image: none;  
 212:}  
 213:.btn-success.disabled:hover,  
 214:.btn-success[disabled]:hover,  
 215:fieldset[disabled] .btn-success:hover,  
 216:.btn-success.disabled:focus,  
 217:.btn-success[disabled]:focus,  
 218:fieldset[disabled] .btn-success:focus,  
 219:.btn-success.disabled.focus,  
 220:.btn-success[disabled].focus,  
 221:fieldset[disabled] .btn-success.focus {  
 222: background-color: #5cb85c;  
 223: border-color: #4cae4c;  
 224:}  
 225:.btn-success .badge {  
 226: color: #5cb85c;  
 227: background-color: #ffffff;  
 228:}  
 229:.btn-info {  
 230: color: #ffffff;  
 231: background-color: #5bc0de;  
 232: border-color: #46b8da;  
 233:}  
 234:.btn-info:focus,  
 235:.btn-info.focus {  
 236: color: #ffffff;  
 237: background-color: #31b0d5;  
 238: border-color: #1b6d85;  
 239:}  
 240:.btn-info:hover {  
 241: color: #ffffff;  
 242: background-color: #31b0d5;  
 243: border-color: #269abc;  
 244:}  
 245:.btn-info:active,  
 246:.btn-info.active,  
 247:.open > .dropdown-toggle.btn-info {  
 248: color: #ffffff;  
 249: background-color: #31b0d5;  
 250: border-color: #269abc;  
 251:}  
 252:.btn-info:active:hover,  
 253:.btn-info.active:hover,  
 254:.open > .dropdown-toggle.btn-info:hover,  
 255:.btn-info:active:focus,  
 256:.btn-info.active:focus,  
 257:.open > .dropdown-toggle.btn-info:focus,  
 258:.btn-info:active.focus,  
 259:.btn-info.active.focus,  
 260:.open > .dropdown-toggle.btn-info.focus {  
 261: color: #ffffff;  
 262: background-color: #269abc;  
 263: border-color: #1b6d85;  
 264:}  
 265:.btn-info:active,  
 266:.btn-info.active,  
 267:.open > .dropdown-toggle.btn-info {  
 268: background-image: none;  
 269:}  
 270:.btn-info.disabled:hover,  
 271:.btn-info[disabled]:hover,  
 272:fieldset[disabled] .btn-info:hover,  
 273:.btn-info.disabled:focus,  
 274:.btn-info[disabled]:focus,  
 275:fieldset[disabled] .btn-info:focus,  
 276:.btn-info.disabled.focus,  
 277:.btn-info[disabled].focus,  
 278:fieldset[disabled] .btn-info.focus {  
 279: background-color: #5bc0de;  
 280: border-color: #46b8da;  
 281:}  
 282:.btn-info .badge {  
 283: color: #5bc0de;  
 284: background-color: #ffffff;  
 285:}  
 286:.btn-warning {  
 287: color: #ffffff;  
 288: background-color: #f0ad4e;  
 289: border-color: #eea236;  
 290:}  
 291:.btn-warning:focus,  
 292:.btn-warning.focus {  
 293: color: #ffffff;  
 294: background-color: #ec971f;  
 295: border-color: #985f0d;  
 296:}  
 297:.btn-warning:hover {  
 298: color: #ffffff;  
 299: background-color: #ec971f;  
 300: border-color: #d58512;  
 301:}  
 302:.btn-warning:active,  
 303:.btn-warning.active,  
 304:.open > .dropdown-toggle.btn-warning {  
 305: color: #ffffff;  
 306: background-color: #ec971f;  
 307: border-color: #d58512;  
 308:}  
 309:.btn-warning:active:hover,  
 310:.btn-warning.active:hover,  
 311:.open > .dropdown-toggle.btn-warning:hover,  
 312:.btn-warning:active:focus,  
 313:.btn-warning.active:focus,  
 314:.open > .dropdown-toggle.btn-warning:focus,  
 315:.btn-warning:active.focus,  
 316:.btn-warning.active.focus,  
 317:.open > .dropdown-toggle.btn-warning.focus {  
 318: color: #ffffff;  
 319: background-color: #d58512;  
 320: border-color: #985f0d;  
 321:}  
 322:.btn-warning:active,  
 323:.btn-warning.active,  
 324:.open > .dropdown-toggle.btn-warning {  
 325: background-image: none;  
 326:}  
 327:.btn-warning.disabled:hover,  
 328:.btn-warning[disabled]:hover,  
 329:fieldset[disabled] .btn-warning:hover,  
 330:.btn-warning.disabled:focus,  
 331:.btn-warning[disabled]:focus,  
 332:fieldset[disabled] .btn-warning:focus,  
 333:.btn-warning.disabled.focus,  
 334:.btn-warning[disabled].focus,  
 335:fieldset[disabled] .btn-warning.focus {  
 336: background-color: #f0ad4e;  
 337: border-color: #eea236;  
 338:}  
 339:.btn-warning .badge {  
 340: color: #f0ad4e;  
 341: background-color: #ffffff;  
 342:}  
 343:.btn-danger {  
 344: color: #ffffff;  
 345: background-color: #d9534f;  
 346: border-color: #d43f3a;  
 347:}  
 348:.btn-danger:focus,  
 349:.btn-danger.focus {  
 350: color: #ffffff;  
 351: background-color: #c9302c;  
 352: border-color: #761c19;  
 353:}  
 354:.btn-danger:hover {  
 355: color: #ffffff;  
 356: background-color: #c9302c;  
 357: border-color: #ac2925;  
 358:}  
 359:.btn-danger:active,  
 360:.btn-danger.active,  
 361:.open > .dropdown-toggle.btn-danger {  
 362: color: #ffffff;  
 363: background-color: #c9302c;  
 364: border-color: #ac2925;  
 365:}  
 366:.btn-danger:active:hover,  
 367:.btn-danger.active:hover,  
 368:.open > .dropdown-toggle.btn-danger:hover,  
 369:.btn-danger:active:focus,  
 370:.btn-danger.active:focus,  
 371:.open > .dropdown-toggle.btn-danger:focus,  
 372:.btn-danger:active.focus,  
 373:.btn-danger.active.focus,  
 374:.open > .dropdown-toggle.btn-danger.focus {  
 375: color: #ffffff;  
 376: background-color: #ac2925;  
 377: border-color: #761c19;  
 378:}  
 379:.btn-danger:active,  
 380:.btn-danger.active,  
 381:.open > .dropdown-toggle.btn-danger {  
 382: background-image: none;  
 383:}  
 384:.btn-danger.disabled:hover,  
 385:.btn-danger[disabled]:hover,  
 386:fieldset[disabled] .btn-danger:hover,  
 387:.btn-danger.disabled:focus,  
 388:.btn-danger[disabled]:focus,  
 389:fieldset[disabled] .btn-danger:focus,  
 390:.btn-danger.disabled.focus,  
 391:.btn-danger[disabled].focus,  
 392:fieldset[disabled] .btn-danger.focus {  
 393: background-color: #d9534f;  
 394: border-color: #d43f3a;  
 395:}  
 396:.btn-danger .badge {  
 397: color: #d9534f;  
 398: background-color: #ffffff;  
 399:}  
 400:.btn-link {  
 401: color: #337ab7;  
 402: font-weight: normal;  
 403: border-radius: 0;  
 404:}  
 405:.btn-link,  
 406:.btn-link:active,  
 407:.btn-link.active,  
 408:.btn-link[disabled],  
 409:fieldset[disabled] .btn-link {  
 410: background-color: transparent;  
 411: -webkit-box-shadow: none;  
 412: box-shadow: none;  
 413:}  
 414:.btn-link,  
 415:.btn-link:hover,  
 416:.btn-link:focus,  
 417:.btn-link:active {  
 418: border-color: transparent;  
 419:}  
 420:.btn-link:hover,  
 421:.btn-link:focus {  
 422: color: #23527c;  
 423: text-decoration: underline;  
 424: background-color: transparent;  
 425:}  
 426:.btn-link[disabled]:hover,  
 427:fieldset[disabled] .btn-link:hover,  
 428:.btn-link[disabled]:focus,  
 429:fieldset[disabled] .btn-link:focus {  
 430: color: #777777;  
 431: text-decoration: none;  
 432:}  
 433:.btn-lg {  
 434: padding: 10px 16px;  
 435: font-size: 18px;  
 436: line-height: 1.3333333;  
 437: border-radius: 6px;  
 438:}  
 439:.btn-sm {  
 440: padding: 5px 10px;  
 441: font-size: 12px;  
 442: line-height: 1.5;  
 443: border-radius: 3px;  
 444:}  
 445:.btn-xs {  
 446: padding: 1px 5px;  
 447: font-size: 12px;  
 448: line-height: 1.5;  
 449: border-radius: 3px;  
 450:}  
 451:.btn-block {  
 452: display: block;  
 453: width: 100%;  
 454:}  
 455:.btn-block + .btn-block {  
 456: margin-top: 5px;  
 457:}

## .\mobile\www\css\bootstrap\textunderscore grid.css

1:.col-xs-1, .col-sm-1, .col-md-1, .col-lg-1, .col-xs-2, .col-sm-2, .col-md-2, .col-lg-2, .col-xs-3, .col-sm-3, .col-md-3, .col-lg-3, .col-xs-4, .col-sm-4, .col-md-4, .col-lg-4, .col-xs-5, .col-sm-5, .col-md-5, .col-lg-5, .col-xs-6, .col-sm-6, .col-md-6, .col-lg-6, .col-xs-7, .col-sm-7, .col-md-7, .col-lg-7, .col-xs-8, .col-sm-8, .col-md-8, .col-lg-8, .col-xs-9, .col-sm-9, .col-md-9, .col-lg-9, .col-xs-10, .col-sm-10, .col-md-10, .col-lg-10, .col-xs-11, .col-sm-11, .col-md-11, .col-lg-11, .col-xs-12, .col-sm-12, .col-md-12, .col-lg-12 {  
 2: position: relative;  
 3: min-height: 1px;  
 4: padding-left: 15px;  
 5: padding-right: 15px;  
 6:}  
 7:.col-xs-1, .col-xs-2, .col-xs-3, .col-xs-4, .col-xs-5, .col-xs-6, .col-xs-7, .col-xs-8, .col-xs-9, .col-xs-10, .col-xs-11, .col-xs-12 {  
 8: float: left;  
 9:}  
 10:.col-xs-12 {  
 11: width: 100%;  
 12:}  
 13:.col-xs-11 {  
 14: width: 91.66666667%;  
 15:}  
 16:.col-xs-10 {  
 17: width: 83.33333333%;  
 18:}  
 19:.col-xs-9 {  
 20: width: 75%;  
 21:}  
 22:.col-xs-8 {  
 23: width: 66.66666667%;  
 24:}  
 25:.col-xs-7 {  
 26: width: 58.33333333%;  
 27:}  
 28:.col-xs-6 {  
 29: width: 50%;  
 30:}  
 31:.col-xs-5 {  
 32: width: 41.66666667%;  
 33:}  
 34:.col-xs-4 {  
 35: width: 33.33333333%;  
 36:}  
 37:.col-xs-3 {  
 38: width: 25%;  
 39:}  
 40:.col-xs-2 {  
 41: width: 16.66666667%;  
 42:}  
 43:.col-xs-1 {  
 44: width: 8.33333333%;  
 45:}  
 46:.col-xs-pull-12 {  
 47: right: 100%;  
 48:}  
 49:.col-xs-pull-11 {  
 50: right: 91.66666667%;  
 51:}  
 52:.col-xs-pull-10 {  
 53: right: 83.33333333%;  
 54:}  
 55:.col-xs-pull-9 {  
 56: right: 75%;  
 57:}  
 58:.col-xs-pull-8 {  
 59: right: 66.66666667%;  
 60:}  
 61:.col-xs-pull-7 {  
 62: right: 58.33333333%;  
 63:}  
 64:.col-xs-pull-6 {  
 65: right: 50%;  
 66:}  
 67:.col-xs-pull-5 {  
 68: right: 41.66666667%;  
 69:}  
 70:.col-xs-pull-4 {  
 71: right: 33.33333333%;  
 72:}  
 73:.col-xs-pull-3 {  
 74: right: 25%;  
 75:}  
 76:.col-xs-pull-2 {  
 77: right: 16.66666667%;  
 78:}  
 79:.col-xs-pull-1 {  
 80: right: 8.33333333%;  
 81:}  
 82:.col-xs-pull-0 {  
 83: right: auto;  
 84:}  
 85:.col-xs-push-12 {  
 86: left: 100%;  
 87:}  
 88:.col-xs-push-11 {  
 89: left: 91.66666667%;  
 90:}  
 91:.col-xs-push-10 {  
 92: left: 83.33333333%;  
 93:}  
 94:.col-xs-push-9 {  
 95: left: 75%;  
 96:}  
 97:.col-xs-push-8 {  
 98: left: 66.66666667%;  
 99:}  
 100:.col-xs-push-7 {  
 101: left: 58.33333333%;  
 102:}  
 103:.col-xs-push-6 {  
 104: left: 50%;  
 105:}  
 106:.col-xs-push-5 {  
 107: left: 41.66666667%;  
 108:}  
 109:.col-xs-push-4 {  
 110: left: 33.33333333%;  
 111:}  
 112:.col-xs-push-3 {  
 113: left: 25%;  
 114:}  
 115:.col-xs-push-2 {  
 116: left: 16.66666667%;  
 117:}  
 118:.col-xs-push-1 {  
 119: left: 8.33333333%;  
 120:}  
 121:.col-xs-push-0 {  
 122: left: auto;  
 123:}  
 124:.col-xs-offset-12 {  
 125: margin-left: 100%;  
 126:}  
 127:.col-xs-offset-11 {  
 128: margin-left: 91.66666667%;  
 129:}  
 130:.col-xs-offset-10 {  
 131: margin-left: 83.33333333%;  
 132:}  
 133:.col-xs-offset-9 {  
 134: margin-left: 75%;  
 135:}  
 136:.col-xs-offset-8 {  
 137: margin-left: 66.66666667%;  
 138:}  
 139:.col-xs-offset-7 {  
 140: margin-left: 58.33333333%;  
 141:}  
 142:.col-xs-offset-6 {  
 143: margin-left: 50%;  
 144:}  
 145:.col-xs-offset-5 {  
 146: margin-left: 41.66666667%;  
 147:}  
 148:.col-xs-offset-4 {  
 149: margin-left: 33.33333333%;  
 150:}  
 151:.col-xs-offset-3 {  
 152: margin-left: 25%;  
 153:}  
 154:.col-xs-offset-2 {  
 155: margin-left: 16.66666667%;  
 156:}  
 157:.col-xs-offset-1 {  
 158: margin-left: 8.33333333%;  
 159:}  
 160:.col-xs-offset-0 {  
 161: margin-left: 0%;  
 162:}  
 163:@media (min-width: 768px) {  
 164: .col-sm-1, .col-sm-2, .col-sm-3, .col-sm-4, .col-sm-5, .col-sm-6, .col-sm-7, .col-sm-8, .col-sm-9, .col-sm-10, .col-sm-11, .col-sm-12 {  
 165: float: left;  
 166: }  
 167: .col-sm-12 {  
 168: width: 100%;  
 169: }  
 170: .col-sm-11 {  
 171: width: 91.66666667%;  
 172: }  
 173: .col-sm-10 {  
 174: width: 83.33333333%;  
 175: }  
 176: .col-sm-9 {  
 177: width: 75%;  
 178: }  
 179: .col-sm-8 {  
 180: width: 66.66666667%;  
 181: }  
 182: .col-sm-7 {  
 183: width: 58.33333333%;  
 184: }  
 185: .col-sm-6 {  
 186: width: 50%;  
 187: }  
 188: .col-sm-5 {  
 189: width: 41.66666667%;  
 190: }  
 191: .col-sm-4 {  
 192: width: 33.33333333%;  
 193: }  
 194: .col-sm-3 {  
 195: width: 25%;  
 196: }  
 197: .col-sm-2 {  
 198: width: 16.66666667%;  
 199: }  
 200: .col-sm-1 {  
 201: width: 8.33333333%;  
 202: }  
 203: .col-sm-pull-12 {  
 204: right: 100%;  
 205: }  
 206: .col-sm-pull-11 {  
 207: right: 91.66666667%;  
 208: }  
 209: .col-sm-pull-10 {  
 210: right: 83.33333333%;  
 211: }  
 212: .col-sm-pull-9 {  
 213: right: 75%;  
 214: }  
 215: .col-sm-pull-8 {  
 216: right: 66.66666667%;  
 217: }  
 218: .col-sm-pull-7 {  
 219: right: 58.33333333%;  
 220: }  
 221: .col-sm-pull-6 {  
 222: right: 50%;  
 223: }  
 224: .col-sm-pull-5 {  
 225: right: 41.66666667%;  
 226: }  
 227: .col-sm-pull-4 {  
 228: right: 33.33333333%;  
 229: }  
 230: .col-sm-pull-3 {  
 231: right: 25%;  
 232: }  
 233: .col-sm-pull-2 {  
 234: right: 16.66666667%;  
 235: }  
 236: .col-sm-pull-1 {  
 237: right: 8.33333333%;  
 238: }  
 239: .col-sm-pull-0 {  
 240: right: auto;  
 241: }  
 242: .col-sm-push-12 {  
 243: left: 100%;  
 244: }  
 245: .col-sm-push-11 {  
 246: left: 91.66666667%;  
 247: }  
 248: .col-sm-push-10 {  
 249: left: 83.33333333%;  
 250: }  
 251: .col-sm-push-9 {  
 252: left: 75%;  
 253: }  
 254: .col-sm-push-8 {  
 255: left: 66.66666667%;  
 256: }  
 257: .col-sm-push-7 {  
 258: left: 58.33333333%;  
 259: }  
 260: .col-sm-push-6 {  
 261: left: 50%;  
 262: }  
 263: .col-sm-push-5 {  
 264: left: 41.66666667%;  
 265: }  
 266: .col-sm-push-4 {  
 267: left: 33.33333333%;  
 268: }  
 269: .col-sm-push-3 {  
 270: left: 25%;  
 271: }  
 272: .col-sm-push-2 {  
 273: left: 16.66666667%;  
 274: }  
 275: .col-sm-push-1 {  
 276: left: 8.33333333%;  
 277: }  
 278: .col-sm-push-0 {  
 279: left: auto;  
 280: }  
 281: .col-sm-offset-12 {  
 282: margin-left: 100%;  
 283: }  
 284: .col-sm-offset-11 {  
 285: margin-left: 91.66666667%;  
 286: }  
 287: .col-sm-offset-10 {  
 288: margin-left: 83.33333333%;  
 289: }  
 290: .col-sm-offset-9 {  
 291: margin-left: 75%;  
 292: }  
 293: .col-sm-offset-8 {  
 294: margin-left: 66.66666667%;  
 295: }  
 296: .col-sm-offset-7 {  
 297: margin-left: 58.33333333%;  
 298: }  
 299: .col-sm-offset-6 {  
 300: margin-left: 50%;  
 301: }  
 302: .col-sm-offset-5 {  
 303: margin-left: 41.66666667%;  
 304: }  
 305: .col-sm-offset-4 {  
 306: margin-left: 33.33333333%;  
 307: }  
 308: .col-sm-offset-3 {  
 309: margin-left: 25%;  
 310: }  
 311: .col-sm-offset-2 {  
 312: margin-left: 16.66666667%;  
 313: }  
 314: .col-sm-offset-1 {  
 315: margin-left: 8.33333333%;  
 316: }  
 317: .col-sm-offset-0 {  
 318: margin-left: 0%;  
 319: }  
 320:}  
 321:@media (min-width: 992px) {  
 322: .col-md-1, .col-md-2, .col-md-3, .col-md-4, .col-md-5, .col-md-6, .col-md-7, .col-md-8, .col-md-9, .col-md-10, .col-md-11, .col-md-12 {  
 323: float: left;  
 324: }  
 325: .col-md-12 {  
 326: width: 100%;  
 327: }  
 328: .col-md-11 {  
 329: width: 91.66666667%;  
 330: }  
 331: .col-md-10 {  
 332: width: 83.33333333%;  
 333: }  
 334: .col-md-9 {  
 335: width: 75%;  
 336: }  
 337: .col-md-8 {  
 338: width: 66.66666667%;  
 339: }  
 340: .col-md-7 {  
 341: width: 58.33333333%;  
 342: }  
 343: .col-md-6 {  
 344: width: 50%;  
 345: }  
 346: .col-md-5 {  
 347: width: 41.66666667%;  
 348: }  
 349: .col-md-4 {  
 350: width: 33.33333333%;  
 351: }  
 352: .col-md-3 {  
 353: width: 25%;  
 354: }  
 355: .col-md-2 {  
 356: width: 16.66666667%;  
 357: }  
 358: .col-md-1 {  
 359: width: 8.33333333%;  
 360: }  
 361: .col-md-pull-12 {  
 362: right: 100%;  
 363: }  
 364: .col-md-pull-11 {  
 365: right: 91.66666667%;  
 366: }  
 367: .col-md-pull-10 {  
 368: right: 83.33333333%;  
 369: }  
 370: .col-md-pull-9 {  
 371: right: 75%;  
 372: }  
 373: .col-md-pull-8 {  
 374: right: 66.66666667%;  
 375: }  
 376: .col-md-pull-7 {  
 377: right: 58.33333333%;  
 378: }  
 379: .col-md-pull-6 {  
 380: right: 50%;  
 381: }  
 382: .col-md-pull-5 {  
 383: right: 41.66666667%;  
 384: }  
 385: .col-md-pull-4 {  
 386: right: 33.33333333%;  
 387: }  
 388: .col-md-pull-3 {  
 389: right: 25%;  
 390: }  
 391: .col-md-pull-2 {  
 392: right: 16.66666667%;  
 393: }  
 394: .col-md-pull-1 {  
 395: right: 8.33333333%;  
 396: }  
 397: .col-md-pull-0 {  
 398: right: auto;  
 399: }  
 400: .col-md-push-12 {  
 401: left: 100%;  
 402: }  
 403: .col-md-push-11 {  
 404: left: 91.66666667%;  
 405: }  
 406: .col-md-push-10 {  
 407: left: 83.33333333%;  
 408: }  
 409: .col-md-push-9 {  
 410: left: 75%;  
 411: }  
 412: .col-md-push-8 {  
 413: left: 66.66666667%;  
 414: }  
 415: .col-md-push-7 {  
 416: left: 58.33333333%;  
 417: }  
 418: .col-md-push-6 {  
 419: left: 50%;  
 420: }  
 421: .col-md-push-5 {  
 422: left: 41.66666667%;  
 423: }  
 424: .col-md-push-4 {  
 425: left: 33.33333333%;  
 426: }  
 427: .col-md-push-3 {  
 428: left: 25%;  
 429: }  
 430: .col-md-push-2 {  
 431: left: 16.66666667%;  
 432: }  
 433: .col-md-push-1 {  
 434: left: 8.33333333%;  
 435: }  
 436: .col-md-push-0 {  
 437: left: auto;  
 438: }  
 439: .col-md-offset-12 {  
 440: margin-left: 100%;  
 441: }  
 442: .col-md-offset-11 {  
 443: margin-left: 91.66666667%;  
 444: }  
 445: .col-md-offset-10 {  
 446: margin-left: 83.33333333%;  
 447: }  
 448: .col-md-offset-9 {  
 449: margin-left: 75%;  
 450: }  
 451: .col-md-offset-8 {  
 452: margin-left: 66.66666667%;  
 453: }  
 454: .col-md-offset-7 {  
 455: margin-left: 58.33333333%;  
 456: }  
 457: .col-md-offset-6 {  
 458: margin-left: 50%;  
 459: }  
 460: .col-md-offset-5 {  
 461: margin-left: 41.66666667%;  
 462: }  
 463: .col-md-offset-4 {  
 464: margin-left: 33.33333333%;  
 465: }  
 466: .col-md-offset-3 {  
 467: margin-left: 25%;  
 468: }  
 469: .col-md-offset-2 {  
 470: margin-left: 16.66666667%;  
 471: }  
 472: .col-md-offset-1 {  
 473: margin-left: 8.33333333%;  
 474: }  
 475: .col-md-offset-0 {  
 476: margin-left: 0%;  
 477: }  
 478:}  
 479:@media (min-width: 1200px) {  
 480: .col-lg-1, .col-lg-2, .col-lg-3, .col-lg-4, .col-lg-5, .col-lg-6, .col-lg-7, .col-lg-8, .col-lg-9, .col-lg-10, .col-lg-11, .col-lg-12 {  
 481: float: left;  
 482: }  
 483: .col-lg-12 {  
 484: width: 100%;  
 485: }  
 486: .col-lg-11 {  
 487: width: 91.66666667%;  
 488: }  
 489: .col-lg-10 {  
 490: width: 83.33333333%;  
 491: }  
 492: .col-lg-9 {  
 493: width: 75%;  
 494: }  
 495: .col-lg-8 {  
 496: width: 66.66666667%;  
 497: }  
 498: .col-lg-7 {  
 499: width: 58.33333333%;  
 500: }  
 501: .col-lg-6 {  
 502: width: 50%;  
 503: }  
 504: .col-lg-5 {  
 505: width: 41.66666667%;  
 506: }  
 507: .col-lg-4 {  
 508: width: 33.33333333%;  
 509: }  
 510: .col-lg-3 {  
 511: width: 25%;  
 512: }  
 513: .col-lg-2 {  
 514: width: 16.66666667%;  
 515: }  
 516: .col-lg-1 {  
 517: width: 8.33333333%;  
 518: }  
 519: .col-lg-pull-12 {  
 520: right: 100%;  
 521: }  
 522: .col-lg-pull-11 {  
 523: right: 91.66666667%;  
 524: }  
 525: .col-lg-pull-10 {  
 526: right: 83.33333333%;  
 527: }  
 528: .col-lg-pull-9 {  
 529: right: 75%;  
 530: }  
 531: .col-lg-pull-8 {  
 532: right: 66.66666667%;  
 533: }  
 534: .col-lg-pull-7 {  
 535: right: 58.33333333%;  
 536: }  
 537: .col-lg-pull-6 {  
 538: right: 50%;  
 539: }  
 540: .col-lg-pull-5 {  
 541: right: 41.66666667%;  
 542: }  
 543: .col-lg-pull-4 {  
 544: right: 33.33333333%;  
 545: }  
 546: .col-lg-pull-3 {  
 547: right: 25%;  
 548: }  
 549: .col-lg-pull-2 {  
 550: right: 16.66666667%;  
 551: }  
 552: .col-lg-pull-1 {  
 553: right: 8.33333333%;  
 554: }  
 555: .col-lg-pull-0 {  
 556: right: auto;  
 557: }  
 558: .col-lg-push-12 {  
 559: left: 100%;  
 560: }  
 561: .col-lg-push-11 {  
 562: left: 91.66666667%;  
 563: }  
 564: .col-lg-push-10 {  
 565: left: 83.33333333%;  
 566: }  
 567: .col-lg-push-9 {  
 568: left: 75%;  
 569: }  
 570: .col-lg-push-8 {  
 571: left: 66.66666667%;  
 572: }  
 573: .col-lg-push-7 {  
 574: left: 58.33333333%;  
 575: }  
 576: .col-lg-push-6 {  
 577: left: 50%;  
 578: }  
 579: .col-lg-push-5 {  
 580: left: 41.66666667%;  
 581: }  
 582: .col-lg-push-4 {  
 583: left: 33.33333333%;  
 584: }  
 585: .col-lg-push-3 {  
 586: left: 25%;  
 587: }  
 588: .col-lg-push-2 {  
 589: left: 16.66666667%;  
 590: }  
 591: .col-lg-push-1 {  
 592: left: 8.33333333%;  
 593: }  
 594: .col-lg-push-0 {  
 595: left: auto;  
 596: }  
 597: .col-lg-offset-12 {  
 598: margin-left: 100%;  
 599: }  
 600: .col-lg-offset-11 {  
 601: margin-left: 91.66666667%;  
 602: }  
 603: .col-lg-offset-10 {  
 604: margin-left: 83.33333333%;  
 605: }  
 606: .col-lg-offset-9 {  
 607: margin-left: 75%;  
 608: }  
 609: .col-lg-offset-8 {  
 610: margin-left: 66.66666667%;  
 611: }  
 612: .col-lg-offset-7 {  
 613: margin-left: 58.33333333%;  
 614: }  
 615: .col-lg-offset-6 {  
 616: margin-left: 50%;  
 617: }  
 618: .col-lg-offset-5 {  
 619: margin-left: 41.66666667%;  
 620: }  
 621: .col-lg-offset-4 {  
 622: margin-left: 33.33333333%;  
 623: }  
 624: .col-lg-offset-3 {  
 625: margin-left: 25%;  
 626: }  
 627: .col-lg-offset-2 {  
 628: margin-left: 16.66666667%;  
 629: }  
 630: .col-lg-offset-1 {  
 631: margin-left: 8.33333333%;  
 632: }  
 633: .col-lg-offset-0 {  
 634: margin-left: 0%;  
 635: }  
 636:}  
 637:

## .\mobile\www\css\font-awesome.min.css

1:/\*!  
 2: \* Font Awesome 4.6.1 by @davegandy - http://fontawesome.io - @fontawesome  
 3: \* License - http://fontawesome.io/license (Font: SIL OFL 1.1, CSS: MIT License)  
 4: \*/@font-face{font-family:'FontAwesome';src:url('../fonts/fontawesome-webfont.eot?v=4.6.1');src:url('../fonts/fontawesome-webfont.eot?#iefix&v=4.6.1') format('embedded-opentype'),url('../fonts/fontawesome-webfont.woff2?v=4.6.1') format('woff2'),url('../fonts/fontawesome-webfont.woff?v=4.6.1') format('woff'),url('../fonts/fontawesome-webfont.ttf?v=4.6.1') format('truetype'),url('../fonts/fontawesome-webfont.svg?v=4.6.1#fontawesomeregular') format('svg');font-weight:normal;font-style:normal}.fa{display:inline-block;font:normal normal normal 14px/1 FontAwesome;font-size:inherit;text-rendering:auto;-webkit-font-smoothing:antialiased;-moz-osx-font-smoothing:grayscale}.fa-lg{font-size:1.33333333em;line-height:.75em;vertical-align:-15%}.fa-2x{font-size:2em}.fa-3x{font-size:3em}.fa-4x{font-size:4em}.fa-5x{font-size:5em}.fa-fw{width:1.28571429em;text-align:center}.fa-ul{padding-left:0;margin-left:2.14285714em;list-style-type:none}.fa-ul>li{position:relative}.fa-li{position:absolute;left:-2.14285714em;width:2.14285714em;top:.14285714em;text-align:center}.fa-li.fa-lg{left:-1.85714286em}.fa-border{padding:.2em .25em .15em;border:solid .08em #eee;border-radius:.1em}.fa-pull-left{float:left}.fa-pull-right{float:right}.fa.fa-pull-left{margin-right:.3em}.fa.fa-pull-right{margin-left:.3em}.pull-right{float:right}.pull-left{float:left}.fa.pull-left{margin-right:.3em}.fa.pull-right{margin-left:.3em}.fa-spin{-webkit-animation:fa-spin 2s infinite linear;animation:fa-spin 2s infinite linear}.fa-pulse{-webkit-animation:fa-spin 1s infinite steps(8);animation:fa-spin 1s infinite steps(8)}@-webkit-keyframes fa-spin{0%{-webkit-transform:rotate(0deg);transform:rotate(0deg)}100%{-webkit-transform:rotate(359deg);transform:rotate(359deg)}}@keyframes fa-spin{0%{-webkit-transform:rotate(0deg);transform:rotate(0deg)}100%{-webkit-transform:rotate(359deg);transform:rotate(359deg)}}.fa-rotate-90{-ms-filter:"progid:DXImageTransform.Microsoft.BasicImage(rotation=1)";-webkit-transform:rotate(90deg);-ms-transform:rotate(90deg);transform:rotate(90deg)}.fa-rotate-180{-ms-filter:"progid:DXImageTransform.Microsoft.BasicImage(rotation=2)";-webkit-transform:rotate(180deg);-ms-transform:rotate(180deg);transform:rotate(180deg)}.fa-rotate-270{-ms-filter:"progid:DXImageTransform.Microsoft.BasicImage(rotation=3)";-webkit-transform:rotate(270deg);-ms-transform:rotate(270deg);transform:rotate(270deg)}.fa-flip-horizontal{-ms-filter:"progid:DXImageTransform.Microsoft.BasicImage(rotation=0, mirror=1)";-webkit-transform:scale(-1, 1);-ms-transform:scale(-1, 1);transform:scale(-1, 1)}.fa-flip-vertical{-ms-filter:"progid:DXImageTransform.Microsoft.BasicImage(rotation=2, mirror=1)";-webkit-transform:scale(1, -1);-ms-transform:scale(1, -1);transform:scale(1, -1)}:root .fa-rotate-90,:root .fa-rotate-180,:root .fa-rotate-270,:root .fa-flip-horizontal,:root .fa-flip-vertical{filter:none}.fa-stack{position:relative;display:inline-block;width:2em;height:2em;line-height:2em;vertical-align:middle}.fa-stack-1x,.fa-stack-2x{position:absolute;left:0;width:100%;text-align:center}.fa-stack-1x{line-height:inherit}.fa-stack-2x{font-size:2em}.fa-inverse{color:#fff}.fa-glass:before{content:"\f000"}.fa-music:before{content:"\f001"}.fa-search:before{content:"\f002"}.fa-envelope-o:before{content:"\f003"}.fa-heart:before{content:"\f004"}.fa-star:before{content:"\f005"}.fa-star-o:before{content:"\f006"}.fa-user:before{content:"\f007"}.fa-film:before{content:"\f008"}.fa-th-large:before{content:"\f009"}.fa-th:before{content:"\f00a"}.fa-th-list:before{content:"\f00b"}.fa-check:before{content:"\f00c"}.fa-remove:before,.fa-close:before,.fa-times:before{content:"\f00d"}.fa-search-plus:before{content:"\f00e"}.fa-search-minus:before{content:"\f010"}.fa-power-off:before{content:"\f011"}.fa-signal:before{content:"\f012"}.fa-gear:before,.fa-cog:before{content:"\f013"}.fa-trash-o:before{content:"\f014"}.fa-home:before{content:"\f015"}.fa-file-o:before{content:"\f016"}.fa-clock-o:before{content:"\f017"}.fa-road:before{content:"\f018"}.fa-download:before{content:"\f019"}.fa-arrow-circle-o-down:before{content:"\f01a"}.fa-arrow-circle-o-up:before{content:"\f01b"}.fa-inbox:before{content:"\f01c"}.fa-play-circle-o:before{content:"\f01d"}.fa-rotate-right:before,.fa-repeat:before{content:"\f01e"}.fa-refresh:before{content:"\f021"}.fa-list-alt:before{content:"\f022"}.fa-lock:before{content:"\f023"}.fa-flag:before{content:"\f024"}.fa-headphones:before{content:"\f025"}.fa-volume-off:before{content:"\f026"}.fa-volume-down:before{content:"\f027"}.fa-volume-up:before{content:"\f028"}.fa-qrcode:before{content:"\f029"}.fa-barcode:before{content:"\f02a"}.fa-tag:before{content:"\f02b"}.fa-tags:before{content:"\f02c"}.fa-book:before{content:"\f02d"}.fa-bookmark:before{content:"\f02e"}.fa-print:before{content:"\f02f"}.fa-camera:before{content:"\f030"}.fa-font:before{content:"\f031"}.fa-bold:before{content:"\f032"}.fa-italic:before{content:"\f033"}.fa-text-height:before{content:"\f034"}.fa-text-width:before{content:"\f035"}.fa-align-left:before{content:"\f036"}.fa-align-center:before{content:"\f037"}.fa-align-right:before{content:"\f038"}.fa-align-justify:before{content:"\f039"}.fa-list:before{content:"\f03a"}.fa-dedent:before,.fa-outdent:before{content:"\f03b"}.fa-indent:before{content:"\f03c"}.fa-video-camera:before{content:"\f03d"}.fa-photo:before,.fa-image:before,.fa-picture-o:before{content:"\f03e"}.fa-pencil:before{content:"\f040"}.fa-map-marker:before{content:"\f041"}.fa-adjust:before{content:"\f042"}.fa-tint:before{content:"\f043"}.fa-edit:before,.fa-pencil-square-o:before{content:"\f044"}.fa-share-square-o:before{content:"\f045"}.fa-check-square-o:before{content:"\f046"}.fa-arrows:before{content:"\f047"}.fa-step-backward:before{content:"\f048"}.fa-fast-backward:before{content:"\f049"}.fa-backward:before{content:"\f04a"}.fa-play:before{content:"\f04b"}.fa-pause:before{content:"\f04c"}.fa-stop:before{content:"\f04d"}.fa-forward:before{content:"\f04e"}.fa-fast-forward:before{content:"\f050"}.fa-step-forward:before{content:"\f051"}.fa-eject:before{content:"\f052"}.fa-chevron-left:before{content:"\f053"}.fa-chevron-right:before{content:"\f054"}.fa-plus-circle:before{content:"\f055"}.fa-minus-circle:before{content:"\f056"}.fa-times-circle:before{content:"\f057"}.fa-check-circle:before{content:"\f058"}.fa-question-circle:before{content:"\f059"}.fa-info-circle:before{content:"\f05a"}.fa-crosshairs:before{content:"\f05b"}.fa-times-circle-o:before{content:"\f05c"}.fa-check-circle-o:before{content:"\f05d"}.fa-ban:before{content:"\f05e"}.fa-arrow-left:before{content:"\f060"}.fa-arrow-right:before{content:"\f061"}.fa-arrow-up:before{content:"\f062"}.fa-arrow-down:before{content:"\f063"}.fa-mail-forward:before,.fa-share:before{content:"\f064"}.fa-expand:before{content:"\f065"}.fa-compress:before{content:"\f066"}.fa-plus:before{content:"\f067"}.fa-minus:before{content:"\f068"}.fa-asterisk:before{content:"\f069"}.fa-exclamation-circle:before{content:"\f06a"}.fa-gift:before{content:"\f06b"}.fa-leaf:before{content:"\f06c"}.fa-fire:before{content:"\f06d"}.fa-eye:before{content:"\f06e"}.fa-eye-slash:before{content:"\f070"}.fa-warning:before,.fa-exclamation-triangle:before{content:"\f071"}.fa-plane:before{content:"\f072"}.fa-calendar:before{content:"\f073"}.fa-random:before{content:"\f074"}.fa-comment:before{content:"\f075"}.fa-magnet:before{content:"\f076"}.fa-chevron-up:before{content:"\f077"}.fa-chevron-down:before{content:"\f078"}.fa-retweet:before{content:"\f079"}.fa-shopping-cart:before{content:"\f07a"}.fa-folder:before{content:"\f07b"}.fa-folder-open:before{content:"\f07c"}.fa-arrows-v:before{content:"\f07d"}.fa-arrows-h:before{content:"\f07e"}.fa-bar-chart-o:before,.fa-bar-chart:before{content:"\f080"}.fa-twitter-square:before{content:"\f081"}.fa-facebook-square:before{content:"\f082"}.fa-camera-retro:before{content:"\f083"}.fa-key:before{content:"\f084"}.fa-gears:before,.fa-cogs:before{content:"\f085"}.fa-comments:before{content:"\f086"}.fa-thumbs-o-up:before{content:"\f087"}.fa-thumbs-o-down:before{content:"\f088"}.fa-star-half:before{content:"\f089"}.fa-heart-o:before{content:"\f08a"}.fa-sign-out:before{content:"\f08b"}.fa-linkedin-square:before{content:"\f08c"}.fa-thumb-tack:before{content:"\f08d"}.fa-external-link:before{content:"\f08e"}.fa-sign-in:before{content:"\f090"}.fa-trophy:before{content:"\f091"}.fa-github-square:before{content:"\f092"}.fa-upload:before{content:"\f093"}.fa-lemon-o:before{content:"\f094"}.fa-phone:before{content:"\f095"}.fa-square-o:before{content:"\f096"}.fa-bookmark-o:before{content:"\f097"}.fa-phone-square:before{content:"\f098"}.fa-twitter:before{content:"\f099"}.fa-facebook-f:before,.fa-facebook:before{content:"\f09a"}.fa-github:before{content:"\f09b"}.fa-unlock:before{content:"\f09c"}.fa-credit-card:before{content:"\f09d"}.fa-feed:before,.fa-rss:before{content:"\f09e"}.fa-hdd-o:before{content:"\f0a0"}.fa-bullhorn:before{content:"\f0a1"}.fa-bell:before{content:"\f0f3"}.fa-certificate:before{content:"\f0a3"}.fa-hand-o-right:before{content:"\f0a4"}.fa-hand-o-left:before{content:"\f0a5"}.fa-hand-o-up:before{content:"\f0a6"}.fa-hand-o-down:before{content:"\f0a7"}.fa-arrow-circle-left:before{content:"\f0a8"}.fa-arrow-circle-right:before{content:"\f0a9"}.fa-arrow-circle-up:before{content:"\f0aa"}.fa-arrow-circle-down:before{content:"\f0ab"}.fa-globe:before{content:"\f0ac"}.fa-wrench:before{content:"\f0ad"}.fa-tasks:before{content:"\f0ae"}.fa-filter:before{content:"\f0b0"}.fa-briefcase:before{content:"\f0b1"}.fa-arrows-alt:before{content:"\f0b2"}.fa-group:before,.fa-users:before{content:"\f0c0"}.fa-chain:before,.fa-link:before{content:"\f0c1"}.fa-cloud:before{content:"\f0c2"}.fa-flask:before{content:"\f0c3"}.fa-cut:before,.fa-scissors:before{content:"\f0c4"}.fa-copy:before,.fa-files-o:before{content:"\f0c5"}.fa-paperclip:before{content:"\f0c6"}.fa-save:before,.fa-floppy-o:before{content:"\f0c7"}.fa-square:before{content:"\f0c8"}.fa-navicon:before,.fa-reorder:before,.fa-bars:before{content:"\f0c9"}.fa-list-ul:before{content:"\f0ca"}.fa-list-ol:before{content:"\f0cb"}.fa-strikethrough:before{content:"\f0cc"}.fa-underline:before{content:"\f0cd"}.fa-table:before{content:"\f0ce"}.fa-magic:before{content:"\f0d0"}.fa-truck:before{content:"\f0d1"}.fa-pinterest:before{content:"\f0d2"}.fa-pinterest-square:before{content:"\f0d3"}.fa-google-plus-square:before{content:"\f0d4"}.fa-google-plus:before{content:"\f0d5"}.fa-money:before{content:"\f0d6"}.fa-caret-down:before{content:"\f0d7"}.fa-caret-up:before{content:"\f0d8"}.fa-caret-left:before{content:"\f0d9"}.fa-caret-right:before{content:"\f0da"}.fa-columns:before{content:"\f0db"}.fa-unsorted:before,.fa-sort:before{content:"\f0dc"}.fa-sort-down:before,.fa-sort-desc:before{content:"\f0dd"}.fa-sort-up:before,.fa-sort-asc:before{content:"\f0de"}.fa-envelope:before{content:"\f0e0"}.fa-linkedin:before{content:"\f0e1"}.fa-rotate-left:before,.fa-undo:before{content:"\f0e2"}.fa-legal:before,.fa-gavel:before{content:"\f0e3"}.fa-dashboard:before,.fa-tachometer:before{content:"\f0e4"}.fa-comment-o:before{content:"\f0e5"}.fa-comments-o:before{content:"\f0e6"}.fa-flash:before,.fa-bolt:before{content:"\f0e7"}.fa-sitemap:before{content:"\f0e8"}.fa-umbrella:before{content:"\f0e9"}.fa-paste:before,.fa-clipboard:before{content:"\f0ea"}.fa-lightbulb-o:before{content:"\f0eb"}.fa-exchange:before{content:"\f0ec"}.fa-cloud-download:before{content:"\f0ed"}.fa-cloud-upload:before{content:"\f0ee"}.fa-user-md:before{content:"\f0f0"}.fa-stethoscope:before{content:"\f0f1"}.fa-suitcase:before{content:"\f0f2"}.fa-bell-o:before{content:"\f0a2"}.fa-coffee:before{content:"\f0f4"}.fa-cutlery:before{content:"\f0f5"}.fa-file-text-o:before{content:"\f0f6"}.fa-building-o:before{content:"\f0f7"}.fa-hospital-o:before{content:"\f0f8"}.fa-ambulance:before{content:"\f0f9"}.fa-medkit:before{content:"\f0fa"}.fa-fighter-jet:before{content:"\f0fb"}.fa-beer:before{content:"\f0fc"}.fa-h-square:before{content:"\f0fd"}.fa-plus-square:before{content:"\f0fe"}.fa-angle-double-left:before{content:"\f100"}.fa-angle-double-right:before{content:"\f101"}.fa-angle-double-up:before{content:"\f102"}.fa-angle-double-down:before{content:"\f103"}.fa-angle-left:before{content:"\f104"}.fa-angle-right:before{content:"\f105"}.fa-angle-up:before{content:"\f106"}.fa-angle-down:before{content:"\f107"}.fa-desktop:before{content:"\f108"}.fa-laptop:before{content:"\f109"}.fa-tablet:before{content:"\f10a"}.fa-mobile-phone:before,.fa-mobile:before{content:"\f10b"}.fa-circle-o:before{content:"\f10c"}.fa-quote-left:before{content:"\f10d"}.fa-quote-right:before{content:"\f10e"}.fa-spinner:before{content:"\f110"}.fa-circle:before{content:"\f111"}.fa-mail-reply:before,.fa-reply:before{content:"\f112"}.fa-github-alt:before{content:"\f113"}.fa-folder-o:before{content:"\f114"}.fa-folder-open-o:before{content:"\f115"}.fa-smile-o:before{content:"\f118"}.fa-frown-o:before{content:"\f119"}.fa-meh-o:before{content:"\f11a"}.fa-gamepad:before{content:"\f11b"}.fa-keyboard-o:before{content:"\f11c"}.fa-flag-o:before{content:"\f11d"}.fa-flag-checkered:before{content:"\f11e"}.fa-terminal:before{content:"\f120"}.fa-code:before{content:"\f121"}.fa-mail-reply-all:before,.fa-reply-all:before{content:"\f122"}.fa-star-half-empty:before,.fa-star-half-full:before,.fa-star-half-o:before{content:"\f123"}.fa-location-arrow:before{content:"\f124"}.fa-crop:before{content:"\f125"}.fa-code-fork:before{content:"\f126"}.fa-unlink:before,.fa-chain-broken:before{content:"\f127"}.fa-question:before{content:"\f128"}.fa-info:before{content:"\f129"}.fa-exclamation:before{content:"\f12a"}.fa-superscript:before{content:"\f12b"}.fa-subscript:before{content:"\f12c"}.fa-eraser:before{content:"\f12d"}.fa-puzzle-piece:before{content:"\f12e"}.fa-microphone:before{content:"\f130"}.fa-microphone-slash:before{content:"\f131"}.fa-shield:before{content:"\f132"}.fa-calendar-o:before{content:"\f133"}.fa-fire-extinguisher:before{content:"\f134"}.fa-rocket:before{content:"\f135"}.fa-maxcdn:before{content:"\f136"}.fa-chevron-circle-left:before{content:"\f137"}.fa-chevron-circle-right:before{content:"\f138"}.fa-chevron-circle-up:before{content:"\f139"}.fa-chevron-circle-down:before{content:"\f13a"}.fa-html5:before{content:"\f13b"}.fa-css3:before{content:"\f13c"}.fa-anchor:before{content:"\f13d"}.fa-unlock-alt:before{content:"\f13e"}.fa-bullseye:before{content:"\f140"}.fa-ellipsis-h:before{content:"\f141"}.fa-ellipsis-v:before{content:"\f142"}.fa-rss-square:before{content:"\f143"}.fa-play-circle:before{content:"\f144"}.fa-ticket:before{content:"\f145"}.fa-minus-square:before{content:"\f146"}.fa-minus-square-o:before{content:"\f147"}.fa-level-up:before{content:"\f148"}.fa-level-down:before{content:"\f149"}.fa-check-square:before{content:"\f14a"}.fa-pencil-square:before{content:"\f14b"}.fa-external-link-square:before{content:"\f14c"}.fa-share-square:before{content:"\f14d"}.fa-compass:before{content:"\f14e"}.fa-toggle-down:before,.fa-caret-square-o-down:before{content:"\f150"}.fa-toggle-up:before,.fa-caret-square-o-up:before{content:"\f151"}.fa-toggle-right:before,.fa-caret-square-o-right:before{content:"\f152"}.fa-euro:before,.fa-eur:before{content:"\f153"}.fa-gbp:before{content:"\f154"}.fa-dollar:before,.fa-usd:before{content:"\f155"}.fa-rupee:before,.fa-inr:before{content:"\f156"}.fa-cny:before,.fa-rmb:before,.fa-yen:before,.fa-jpy:before{content:"\f157"}.fa-ruble:before,.fa-rouble:before,.fa-rub:before{content:"\f158"}.fa-won:before,.fa-krw:before{content:"\f159"}.fa-bitcoin:before,.fa-btc:before{content:"\f15a"}.fa-file:before{content:"\f15b"}.fa-file-text:before{content:"\f15c"}.fa-sort-alpha-asc:before{content:"\f15d"}.fa-sort-alpha-desc:before{content:"\f15e"}.fa-sort-amount-asc:before{content:"\f160"}.fa-sort-amount-desc:before{content:"\f161"}.fa-sort-numeric-asc:before{content:"\f162"}.fa-sort-numeric-desc:before{content:"\f163"}.fa-thumbs-up:before{content:"\f164"}.fa-thumbs-down:before{content:"\f165"}.fa-youtube-square:before{content:"\f166"}.fa-youtube:before{content:"\f167"}.fa-xing:before{content:"\f168"}.fa-xing-square:before{content:"\f169"}.fa-youtube-play:before{content:"\f16a"}.fa-dropbox:before{content:"\f16b"}.fa-stack-overflow:before{content:"\f16c"}.fa-instagram:before{content:"\f16d"}.fa-flickr:before{content:"\f16e"}.fa-adn:before{content:"\f170"}.fa-bitbucket:before{content:"\f171"}.fa-bitbucket-square:before{content:"\f172"}.fa-tumblr:before{content:"\f173"}.fa-tumblr-square:before{content:"\f174"}.fa-long-arrow-down:before{content:"\f175"}.fa-long-arrow-up:before{content:"\f176"}.fa-long-arrow-left:before{content:"\f177"}.fa-long-arrow-right:before{content:"\f178"}.fa-apple:before{content:"\f179"}.fa-windows:before{content:"\f17a"}.fa-android:before{content:"\f17b"}.fa-linux:before{content:"\f17c"}.fa-dribbble:before{content:"\f17d"}.fa-skype:before{content:"\f17e"}.fa-foursquare:before{content:"\f180"}.fa-trello:before{content:"\f181"}.fa-female:before{content:"\f182"}.fa-male:before{content:"\f183"}.fa-gittip:before,.fa-gratipay:before{content:"\f184"}.fa-sun-o:before{content:"\f185"}.fa-moon-o:before{content:"\f186"}.fa-archive:before{content:"\f187"}.fa-bug:before{content:"\f188"}.fa-vk:before{content:"\f189"}.fa-weibo:before{content:"\f18a"}.fa-renren:before{content:"\f18b"}.fa-pagelines:before{content:"\f18c"}.fa-stack-exchange:before{content:"\f18d"}.fa-arrow-circle-o-right:before{content:"\f18e"}.fa-arrow-circle-o-left:before{content:"\f190"}.fa-toggle-left:before,.fa-caret-square-o-left:before{content:"\f191"}.fa-dot-circle-o:before{content:"\f192"}.fa-wheelchair:before{content:"\f193"}.fa-vimeo-square:before{content:"\f194"}.fa-turkish-lira:before,.fa-try:before{content:"\f195"}.fa-plus-square-o:before{content:"\f196"}.fa-space-shuttle:before{content:"\f197"}.fa-slack:before{content:"\f198"}.fa-envelope-square:before{content:"\f199"}.fa-wordpress:before{content:"\f19a"}.fa-openid:before{content:"\f19b"}.fa-institution:before,.fa-bank:before,.fa-university:before{content:"\f19c"}.fa-mortar-board:before,.fa-graduation-cap:before{content:"\f19d"}.fa-yahoo:before{content:"\f19e"}.fa-google:before{content:"\f1a0"}.fa-reddit:before{content:"\f1a1"}.fa-reddit-square:before{content:"\f1a2"}.fa-stumbleupon-circle:before{content:"\f1a3"}.fa-stumbleupon:before{content:"\f1a4"}.fa-delicious:before{content:"\f1a5"}.fa-digg:before{content:"\f1a6"}.fa-pied-piper:before{content:"\f1a7"}.fa-pied-piper-alt:before{content:"\f1a8"}.fa-drupal:before{content:"\f1a9"}.fa-joomla:before{content:"\f1aa"}.fa-language:before{content:"\f1ab"}.fa-fax:before{content:"\f1ac"}.fa-building:before{content:"\f1ad"}.fa-child:before{content:"\f1ae"}.fa-paw:before{content:"\f1b0"}.fa-spoon:before{content:"\f1b1"}.fa-cube:before{content:"\f1b2"}.fa-cubes:before{content:"\f1b3"}.fa-behance:before{content:"\f1b4"}.fa-behance-square:before{content:"\f1b5"}.fa-steam:before{content:"\f1b6"}.fa-steam-square:before{content:"\f1b7"}.fa-recycle:before{content:"\f1b8"}.fa-automobile:before,.fa-car:before{content:"\f1b9"}.fa-cab:before,.fa-taxi:before{content:"\f1ba"}.fa-tree:before{content:"\f1bb"}.fa-spotify:before{content:"\f1bc"}.fa-deviantart:before{content:"\f1bd"}.fa-soundcloud:before{content:"\f1be"}.fa-database:before{content:"\f1c0"}.fa-file-pdf-o:before{content:"\f1c1"}.fa-file-word-o:before{content:"\f1c2"}.fa-file-excel-o:before{content:"\f1c3"}.fa-file-powerpoint-o:before{content:"\f1c4"}.fa-file-photo-o:before,.fa-file-picture-o:before,.fa-file-image-o:before{content:"\f1c5"}.fa-file-zip-o:before,.fa-file-archive-o:before{content:"\f1c6"}.fa-file-sound-o:before,.fa-file-audio-o:before{content:"\f1c7"}.fa-file-movie-o:before,.fa-file-video-o:before{content:"\f1c8"}.fa-file-code-o:before{content:"\f1c9"}.fa-vine:before{content:"\f1ca"}.fa-codepen:before{content:"\f1cb"}.fa-jsfiddle:before{content:"\f1cc"}.fa-life-bouy:before,.fa-life-buoy:before,.fa-life-saver:before,.fa-support:before,.fa-life-ring:before{content:"\f1cd"}.fa-circle-o-notch:before{content:"\f1ce"}.fa-ra:before,.fa-rebel:before{content:"\f1d0"}.fa-ge:before,.fa-empire:before{content:"\f1d1"}.fa-git-square:before{content:"\f1d2"}.fa-git:before{content:"\f1d3"}.fa-y-combinator-square:before,.fa-yc-square:before,.fa-hacker-news:before{content:"\f1d4"}.fa-tencent-weibo:before{content:"\f1d5"}.fa-qq:before{content:"\f1d6"}.fa-wechat:before,.fa-weixin:before{content:"\f1d7"}.fa-send:before,.fa-paper-plane:before{content:"\f1d8"}.fa-send-o:before,.fa-paper-plane-o:before{content:"\f1d9"}.fa-history:before{content:"\f1da"}.fa-circle-thin:before{content:"\f1db"}.fa-header:before{content:"\f1dc"}.fa-paragraph:before{content:"\f1dd"}.fa-sliders:before{content:"\f1de"}.fa-share-alt:before{content:"\f1e0"}.fa-share-alt-square:before{content:"\f1e1"}.fa-bomb:before{content:"\f1e2"}.fa-soccer-ball-o:before,.fa-futbol-o:before{content:"\f1e3"}.fa-tty:before{content:"\f1e4"}.fa-binoculars:before{content:"\f1e5"}.fa-plug:before{content:"\f1e6"}.fa-slideshare:before{content:"\f1e7"}.fa-twitch:before{content:"\f1e8"}.fa-yelp:before{content:"\f1e9"}.fa-newspaper-o:before{content:"\f1ea"}.fa-wifi:before{content:"\f1eb"}.fa-calculator:before{content:"\f1ec"}.fa-paypal:before{content:"\f1ed"}.fa-google-wallet:before{content:"\f1ee"}.fa-cc-visa:before{content:"\f1f0"}.fa-cc-mastercard:before{content:"\f1f1"}.fa-cc-discover:before{content:"\f1f2"}.fa-cc-amex:before{content:"\f1f3"}.fa-cc-paypal:before{content:"\f1f4"}.fa-cc-stripe:before{content:"\f1f5"}.fa-bell-slash:before{content:"\f1f6"}.fa-bell-slash-o:before{content:"\f1f7"}.fa-trash:before{content:"\f1f8"}.fa-copyright:before{content:"\f1f9"}.fa-at:before{content:"\f1fa"}.fa-eyedropper:before{content:"\f1fb"}.fa-paint-brush:before{content:"\f1fc"}.fa-birthday-cake:before{content:"\f1fd"}.fa-area-chart:before{content:"\f1fe"}.fa-pie-chart:before{content:"\f200"}.fa-line-chart:before{content:"\f201"}.fa-lastfm:before{content:"\f202"}.fa-lastfm-square:before{content:"\f203"}.fa-toggle-off:before{content:"\f204"}.fa-toggle-on:before{content:"\f205"}.fa-bicycle:before{content:"\f206"}.fa-bus:before{content:"\f207"}.fa-ioxhost:before{content:"\f208"}.fa-angellist:before{content:"\f209"}.fa-cc:before{content:"\f20a"}.fa-shekel:before,.fa-sheqel:before,.fa-ils:before{content:"\f20b"}.fa-meanpath:before{content:"\f20c"}.fa-buysellads:before{content:"\f20d"}.fa-connectdevelop:before{content:"\f20e"}.fa-dashcube:before{content:"\f210"}.fa-forumbee:before{content:"\f211"}.fa-leanpub:before{content:"\f212"}.fa-sellsy:before{content:"\f213"}.fa-shirtsinbulk:before{content:"\f214"}.fa-simplybuilt:before{content:"\f215"}.fa-skyatlas:before{content:"\f216"}.fa-cart-plus:before{content:"\f217"}.fa-cart-arrow-down:before{content:"\f218"}.fa-diamond:before{content:"\f219"}.fa-ship:before{content:"\f21a"}.fa-user-secret:before{content:"\f21b"}.fa-motorcycle:before{content:"\f21c"}.fa-street-view:before{content:"\f21d"}.fa-heartbeat:before{content:"\f21e"}.fa-venus:before{content:"\f221"}.fa-mars:before{content:"\f222"}.fa-mercury:before{content:"\f223"}.fa-intersex:before,.fa-transgender:before{content:"\f224"}.fa-transgender-alt:before{content:"\f225"}.fa-venus-double:before{content:"\f226"}.fa-mars-double:before{content:"\f227"}.fa-venus-mars:before{content:"\f228"}.fa-mars-stroke:before{content:"\f229"}.fa-mars-stroke-v:before{content:"\f22a"}.fa-mars-stroke-h:before{content:"\f22b"}.fa-neuter:before{content:"\f22c"}.fa-genderless:before{content:"\f22d"}.fa-facebook-official:before{content:"\f230"}.fa-pinterest-p:before{content:"\f231"}.fa-whatsapp:before{content:"\f232"}.fa-server:before{content:"\f233"}.fa-user-plus:before{content:"\f234"}.fa-user-times:before{content:"\f235"}.fa-hotel:before,.fa-bed:before{content:"\f236"}.fa-viacoin:before{content:"\f237"}.fa-train:before{content:"\f238"}.fa-subway:before{content:"\f239"}.fa-medium:before{content:"\f23a"}.fa-yc:before,.fa-y-combinator:before{content:"\f23b"}.fa-optin-monster:before{content:"\f23c"}.fa-opencart:before{content:"\f23d"}.fa-expeditedssl:before{content:"\f23e"}.fa-battery-4:before,.fa-battery-full:before{content:"\f240"}.fa-battery-3:before,.fa-battery-three-quarters:before{content:"\f241"}.fa-battery-2:before,.fa-battery-half:before{content:"\f242"}.fa-battery-1:before,.fa-battery-quarter:before{content:"\f243"}.fa-battery-0:before,.fa-battery-empty:before{content:"\f244"}.fa-mouse-pointer:before{content:"\f245"}.fa-i-cursor:before{content:"\f246"}.fa-object-group:before{content:"\f247"}.fa-object-ungroup:before{content:"\f248"}.fa-sticky-note:before{content:"\f249"}.fa-sticky-note-o:before{content:"\f24a"}.fa-cc-jcb:before{content:"\f24b"}.fa-cc-diners-club:before{content:"\f24c"}.fa-clone:before{content:"\f24d"}.fa-balance-scale:before{content:"\f24e"}.fa-hourglass-o:before{content:"\f250"}.fa-hourglass-1:before,.fa-hourglass-start:before{content:"\f251"}.fa-hourglass-2:before,.fa-hourglass-half:before{content:"\f252"}.fa-hourglass-3:before,.fa-hourglass-end:before{content:"\f253"}.fa-hourglass:before{content:"\f254"}.fa-hand-grab-o:before,.fa-hand-rock-o:before{content:"\f255"}.fa-hand-stop-o:before,.fa-hand-paper-o:before{content:"\f256"}.fa-hand-scissors-o:before{content:"\f257"}.fa-hand-lizard-o:before{content:"\f258"}.fa-hand-spock-o:before{content:"\f259"}.fa-hand-pointer-o:before{content:"\f25a"}.fa-hand-peace-o:before{content:"\f25b"}.fa-trademark:before{content:"\f25c"}.fa-registered:before{content:"\f25d"}.fa-creative-commons:before{content:"\f25e"}.fa-gg:before{content:"\f260"}.fa-gg-circle:before{content:"\f261"}.fa-tripadvisor:before{content:"\f262"}.fa-odnoklassniki:before{content:"\f263"}.fa-odnoklassniki-square:before{content:"\f264"}.fa-get-pocket:before{content:"\f265"}.fa-wikipedia-w:before{content:"\f266"}.fa-safari:before{content:"\f267"}.fa-chrome:before{content:"\f268"}.fa-firefox:before{content:"\f269"}.fa-opera:before{content:"\f26a"}.fa-internet-explorer:before{content:"\f26b"}.fa-tv:before,.fa-television:before{content:"\f26c"}.fa-contao:before{content:"\f26d"}.fa-500px:before{content:"\f26e"}.fa-amazon:before{content:"\f270"}.fa-calendar-plus-o:before{content:"\f271"}.fa-calendar-minus-o:before{content:"\f272"}.fa-calendar-times-o:before{content:"\f273"}.fa-calendar-check-o:before{content:"\f274"}.fa-industry:before{content:"\f275"}.fa-map-pin:before{content:"\f276"}.fa-map-signs:before{content:"\f277"}.fa-map-o:before{content:"\f278"}.fa-map:before{content:"\f279"}.fa-commenting:before{content:"\f27a"}.fa-commenting-o:before{content:"\f27b"}.fa-houzz:before{content:"\f27c"}.fa-vimeo:before{content:"\f27d"}.fa-black-tie:before{content:"\f27e"}.fa-fonticons:before{content:"\f280"}.fa-reddit-alien:before{content:"\f281"}.fa-edge:before{content:"\f282"}.fa-credit-card-alt:before{content:"\f283"}.fa-codiepie:before{content:"\f284"}.fa-modx:before{content:"\f285"}.fa-fort-awesome:before{content:"\f286"}.fa-usb:before{content:"\f287"}.fa-product-hunt:before{content:"\f288"}.fa-mixcloud:before{content:"\f289"}.fa-scribd:before{content:"\f28a"}.fa-pause-circle:before{content:"\f28b"}.fa-pause-circle-o:before{content:"\f28c"}.fa-stop-circle:before{content:"\f28d"}.fa-stop-circle-o:before{content:"\f28e"}.fa-shopping-bag:before{content:"\f290"}.fa-shopping-basket:before{content:"\f291"}.fa-hashtag:before{content:"\f292"}.fa-bluetooth:before{content:"\f293"}.fa-bluetooth-b:before{content:"\f294"}.fa-percent:before{content:"\f295"}.fa-gitlab:before{content:"\f296"}.fa-wpbeginner:before{content:"\f297"}.fa-wpforms:before{content:"\f298"}.fa-envira:before{content:"\f299"}.fa-universal-access:before{content:"\f29a"}.fa-wheelchair-alt:before{content:"\f29b"}.fa-question-circle-o:before{content:"\f29c"}.fa-blind:before{content:"\f29d"}.fa-audio-description:before{content:"\f29e"}.fa-volume-control-phone:before{content:"\f2a0"}.fa-braille:before{content:"\f2a1"}.fa-assistive-listening-systems:before{content:"\f2a2"}.fa-asl-interpreting:before,.fa-american-sign-language-interpreting:before{content:"\f2a3"}.fa-deafness:before,.fa-hard-of-hearing:before,.fa-deaf:before{content:"\f2a4"}.fa-glide:before{content:"\f2a5"}.fa-glide-g:before{content:"\f2a6"}.fa-signing:before,.fa-sign-language:before{content:"\f2a7"}.fa-low-vision:before{content:"\f2a8"}.fa-viadeo:before{content:"\f2a9"}.fa-viadeo-square:before{content:"\f2aa"}.fa-snapchat:before{content:"\f2ab"}.fa-snapchat-ghost:before{content:"\f2ac"}.fa-snapchat-square:before{content:"\f2ad"}.sr-only{position:absolute;width:1px;height:1px;padding:0;margin:-1px;overflow:hidden;clip:rect(0, 0, 0, 0);border:0}.sr-only-focusable:active,.sr-only-focusable:focus{position:static;width:auto;height:auto;margin:0;overflow:visible;clip:auto}

## .\mobile\www\css\full\textunderscore bootstrap.min.css

1:@charset "UTF-8";/\*!  
 2: \* Bootstrap v4.0.0-alpha (http://getbootstrap.com)  
 3: \* Copyright 2011-2015 Twitter, Inc.  
 4: \* Licensed under MIT (https://github.com/twbs/bootstrap/blob/master/LICENSE)  
 5: \*//\*! normalize.css v3.0.3 | MIT License | github.com/necolas/normalize.css \*/img,legend{border:0}dl,ol,p,pre,ul{margin-top:0}address,dl,ol,p,ul{margin-bottom:1rem}b,dt,optgroup,strong{font-weight:700}caption,th{text-align:left}fieldset,legend,td,th{padding:0}pre,textarea{overflow:auto}.btn-group>.btn-group,.btn-toolbar .btn-group,.btn-toolbar .input-group,.col-xs-1,.col-xs-10,.col-xs-11,.col-xs-12,.col-xs-2,.col-xs-3,.col-xs-4,.col-xs-5,.col-xs-6,.col-xs-7,.col-xs-8,.col-xs-9,.dropdown-menu,.table-reflow thead,.table-reflow tr{float:left}.btn,.c-indicator{-webkit-user-select:none;-moz-user-select:none;-ms-user-select:none}.btn-toolbar:after,.clearfix:after,.container-fluid:after,.container:after,.dl-horizontal:after,.dropdown-item,.modal-footer:after,.modal-header:after,.nav-tabs:after,.navbar:after,.pager:after,.row:after{clear:both}article,aside,details,figcaption,figure,footer,header,hgroup,main,menu,nav,section,summary{display:block}audio,canvas,progress,video{display:inline-block;vertical-align:baseline}audio:not([controls]){display:none;height:0}[hidden],template{display:none}a{background-color:transparent}a:active,a:hover{outline:0}abbr[title]{border-bottom:1px dotted}dfn{font-style:italic}h1{margin:.67em 0;font-size:2em}mark{color:#000;background:#ff0}small{font-size:80%}sub,sup{position:relative;font-size:75%;line-height:0;vertical-align:baseline}sup{top:-.5em}sub{bottom:-.25em}img{vertical-align:middle}svg:not(:root){overflow:hidden}hr{height:0;-webkit-box-sizing:content-box;box-sizing:content-box}code,kbd,pre,samp{font-size:1em}button,input,optgroup,select,textarea{margin:0;font:inherit;color:inherit}address,legend{line-height:inherit}button{overflow:visible}button,select{text-transform:none}button,html input[type=button],input[type=reset],input[type=submit]{-webkit-appearance:button;cursor:pointer}button[disabled],html input[disabled]{cursor:default}button::-moz-focus-inner,input::-moz-focus-inner{padding:0;border:0}input{line-height:normal}input[type=checkbox],input[type=radio]{-webkit-box-sizing:border-box;box-sizing:border-box;padding:0}input[type=number]::-webkit-inner-spin-button,input[type=number]::-webkit-outer-spin-button{height:auto}input[type=search]{-webkit-box-sizing:content-box;box-sizing:content-box;-webkit-appearance:none}input[type=search]::-webkit-search-cancel-button,input[type=search]::-webkit-search-decoration{-webkit-appearance:none}textarea{resize:vertical}table{border-spacing:0;border-collapse:collapse}@media print{blockquote,img,pre,tr{page-break-inside:avoid}\*,:after,:before{text-shadow:none!important;-webkit-box-shadow:none!important;box-shadow:none!important}a,a:visited{text-decoration:underline}abbr[title]:after{content:" (" attr(title) ")"}blockquote,pre{border:1px solid #999}thead{display:table-header-group}img{max-width:100%!important}h2,h3,p{orphans:3;widows:3}h2,h3{page-break-after:avoid}.navbar{display:none}.btn>.caret,.dropup>.btn>.caret{border-top-color:#000!important}.label{border:1px solid #000}.table{border-collapse:collapse!important}.table td,.table th{background-color:#fff!important}.table-bordered td,.table-bordered th{border:1px solid #ddd!important}}html{font-family:sans-serif;-webkit-text-size-adjust:100%;-ms-text-size-adjust:100%;-webkit-box-sizing:border-box;box-sizing:border-box;font-size:16px;-webkit-tap-highlight-color:transparent}\*,:after,:before{-webkit-box-sizing:inherit;box-sizing:inherit}@-ms-viewport{width:device-width}@viewport{width:device-width}body{margin:0;font-family:"Helvetica Neue",Helvetica,Arial,sans-serif;font-size:1rem;line-height:1.5;color:#373a3c;background-color:#fff}h1,h2,h3,h4,h5,h6{margin-top:0;margin-bottom:.5rem}abbr[data-original-title],abbr[title]{cursor:help;border-bottom:1px dotted #818a91}address{font-style:normal}ol ol,ol ul,ul ol,ul ul{margin-bottom:0}dd,label{margin-bottom:.5rem}dd{margin-left:0}blockquote,figure{margin:0 0 1rem}a{color:#0275d8;text-decoration:none}a:focus,a:hover{color:#014c8c;text-decoration:underline}a:focus{outline:dotted thin;outline:-webkit-focus-ring-color auto 5px;outline-offset:-2px}[role=button]{cursor:pointer}table{background-color:transparent}caption{padding-top:.75rem;padding-bottom:.75rem;color:#818a91;caption-side:bottom}label{display:inline-block}button,input,select,textarea{margin:0;line-height:inherit}fieldset{min-width:0;margin:0;border:0}legend{display:block;width:100%;margin-bottom:.5rem;font-size:1.5rem}.list-inline>li,output{display:inline-block}.h1,.h2,.h3,.h4,.h5,.h6,h1,h2,h3,h4,h5,h6{font-family:inherit;font-weight:500;line-height:1.1;color:inherit;margin-bottom:.5rem}.display-1,.display-2,.display-3,.display-4,.lead{font-weight:300}.blockquote,hr{margin-bottom:1rem}.h1,h1{font-size:2.5rem}.h2,h2{font-size:2rem}.h3,h3{font-size:1.75rem}.h4,h4{font-size:1.5rem}.h5,h5{font-size:1.25rem}.h6,h6{font-size:1rem}.lead{font-size:1.25rem}.display-1{font-size:3.5rem}.display-2{font-size:4.5rem}.display-3{font-size:5.5rem}.display-4{font-size:6rem}hr{margin-top:1rem;border:0;border-top:.0625rem solid rgba(0,0,0,.1)}.small,small{font-size:80%;font-weight:400}.mark,mark{padding:.2em;background-color:#fcf8e3}.list-inline,.list-unstyled{padding-left:0;list-style:none}.list-inline{margin-left:-5px}.list-inline>li{padding-right:5px;padding-left:5px}.dl-horizontal{margin-right:-1.875rem;margin-left:-1.875rem}.container,.container-fluid{margin-right:auto;margin-left:auto}.dl-horizontal:after,.dl-horizontal:before{display:table;content:" "}.initialism{font-size:90%;text-transform:uppercase}.blockquote{padding:.5rem 1rem;font-size:1.25rem;border-left:.25rem solid #eceeef}.blockquote ol:last-child,.blockquote p:last-child,.blockquote ul:last-child{margin-bottom:0}.blockquote footer{display:block;font-size:80%;line-height:1.5;color:#818a91}.blockquote footer:before{content:"\2014 \00A0"}.blockquote-reverse{padding-right:1rem;padding-left:0;text-align:right;border-right:.25rem solid #eceeef;border-left:0}.blockquote-reverse footer:before{content:""}.blockquote-reverse footer:after{content:"\00A0 \2014"}.figure{display:inline-block}.figure>img{margin-bottom:.5rem;line-height:1}.table,pre{margin-bottom:1rem}.figure-caption{font-size:90%;color:#818a91}.carousel-inner>.carousel-item>a>img,.carousel-inner>.carousel-item>img,.figure>img,.img-responsive{display:block;max-width:100%;height:auto}.img-rounded{border-radius:.3rem}.img-thumbnail{display:inline-block;max-width:100%;height:auto;padding:.25rem;line-height:1.5;background-color:#fff;border:1px solid #ddd;border-radius:.25rem;-webkit-transition:all .2s ease-in-out;-o-transition:all .2s ease-in-out;transition:all .2s ease-in-out}code,kbd{padding:.2rem .4rem;font-size:90%}.img-circle{border-radius:50%}code,kbd,pre,samp{font-family:Menlo,Monaco,Consolas,"Courier New",monospace}code{color:#bd4147;background-color:#f7f7f9;border-radius:.25rem}kbd{color:#fff;background-color:#333;border-radius:.2rem}kbd kbd{padding:0;font-size:100%;font-weight:700}pre{display:block;font-size:90%;line-height:1.5;color:#373a3c}.container-fluid:after,.container-fluid:before,.container:after,.container:before,.row:after,.row:before{display:table;content:" "}pre code{padding:0;font-size:inherit;color:inherit;background-color:transparent;border-radius:0}.container,.container-fluid{padding-right:.9375rem;padding-left:.9375rem}.pre-scrollable{max-height:340px;overflow-y:scroll}.row{margin-right:-.9375rem;margin-left:-.9375rem}.col-lg-1,.col-lg-10,.col-lg-11,.col-lg-12,.col-lg-2,.col-lg-3,.col-lg-4,.col-lg-5,.col-lg-6,.col-lg-7,.col-lg-8,.col-lg-9,.col-md-1,.col-md-10,.col-md-11,.col-md-12,.col-md-2,.col-md-3,.col-md-4,.col-md-5,.col-md-6,.col-md-7,.col-md-8,.col-md-9,.col-sm-1,.col-sm-10,.col-sm-11,.col-sm-12,.col-sm-2,.col-sm-3,.col-sm-4,.col-sm-5,.col-sm-6,.col-sm-7,.col-sm-8,.col-sm-9,.col-xl-1,.col-xl-10,.col-xl-11,.col-xl-12,.col-xl-2,.col-xl-3,.col-xl-4,.col-xl-5,.col-xl-6,.col-xl-7,.col-xl-8,.col-xl-9,.col-xs-1,.col-xs-10,.col-xs-11,.col-xs-12,.col-xs-2,.col-xs-3,.col-xs-4,.col-xs-5,.col-xs-6,.col-xs-7,.col-xs-8,.col-xs-9{position:relative;min-height:1px;padding-right:.9375rem;padding-left:.9375rem}.col-xs-1{width:8.333333%}.col-xs-2{width:16.666667%}.col-xs-3{width:25%}.col-xs-4{width:33.333333%}.col-xs-5{width:41.666667%}.col-xs-6{width:50%}.col-xs-7{width:58.333333%}.col-xs-8{width:66.666667%}.col-xs-9{width:75%}.col-xs-10{width:83.333333%}.col-xs-11{width:91.666667%}.col-xs-12{width:100%}.col-xs-pull-0{right:auto}.col-xs-pull-1{right:8.333333%}.col-xs-pull-2{right:16.666667%}.col-xs-pull-3{right:25%}.col-xs-pull-4{right:33.333333%}.col-xs-pull-5{right:41.666667%}.col-xs-pull-6{right:50%}.col-xs-pull-7{right:58.333333%}.col-xs-pull-8{right:66.666667%}.col-xs-pull-9{right:75%}.col-xs-pull-10{right:83.333333%}.col-xs-pull-11{right:91.666667%}.col-xs-pull-12{right:100%}.col-xs-push-0{left:auto}.col-xs-push-1{left:8.333333%}.col-xs-push-2{left:16.666667%}.col-xs-push-3{left:25%}.col-xs-push-4{left:33.333333%}.col-xs-push-5{left:41.666667%}.col-xs-push-6{left:50%}.col-xs-push-7{left:58.333333%}.col-xs-push-8{left:66.666667%}.col-xs-push-9{left:75%}.col-xs-push-10{left:83.333333%}.col-xs-push-11{left:91.666667%}.col-xs-push-12{left:100%}.col-xs-offset-0{margin-left:0}.col-xs-offset-1{margin-left:8.333333%}.col-xs-offset-2{margin-left:16.666667%}.col-xs-offset-3{margin-left:25%}.col-xs-offset-4{margin-left:33.333333%}.col-xs-offset-5{margin-left:41.666667%}.col-xs-offset-6{margin-left:50%}.col-xs-offset-7{margin-left:58.333333%}.col-xs-offset-8{margin-left:66.666667%}.col-xs-offset-9{margin-left:75%}.col-xs-offset-10{margin-left:83.333333%}.col-xs-offset-11{margin-left:91.666667%}.col-xs-offset-12{margin-left:100%}@media (min-width:34em){.container{max-width:34rem}.col-sm-1,.col-sm-10,.col-sm-11,.col-sm-12,.col-sm-2,.col-sm-3,.col-sm-4,.col-sm-5,.col-sm-6,.col-sm-7,.col-sm-8,.col-sm-9{float:left}.col-sm-1{width:8.333333%}.col-sm-2{width:16.666667%}.col-sm-3{width:25%}.col-sm-4{width:33.333333%}.col-sm-5{width:41.666667%}.col-sm-6{width:50%}.col-sm-7{width:58.333333%}.col-sm-8{width:66.666667%}.col-sm-9{width:75%}.col-sm-10{width:83.333333%}.col-sm-11{width:91.666667%}.col-sm-12{width:100%}.col-sm-pull-0{right:auto}.col-sm-pull-1{right:8.333333%}.col-sm-pull-2{right:16.666667%}.col-sm-pull-3{right:25%}.col-sm-pull-4{right:33.333333%}.col-sm-pull-5{right:41.666667%}.col-sm-pull-6{right:50%}.col-sm-pull-7{right:58.333333%}.col-sm-pull-8{right:66.666667%}.col-sm-pull-9{right:75%}.col-sm-pull-10{right:83.333333%}.col-sm-pull-11{right:91.666667%}.col-sm-pull-12{right:100%}.col-sm-push-0{left:auto}.col-sm-push-1{left:8.333333%}.col-sm-push-2{left:16.666667%}.col-sm-push-3{left:25%}.col-sm-push-4{left:33.333333%}.col-sm-push-5{left:41.666667%}.col-sm-push-6{left:50%}.col-sm-push-7{left:58.333333%}.col-sm-push-8{left:66.666667%}.col-sm-push-9{left:75%}.col-sm-push-10{left:83.333333%}.col-sm-push-11{left:91.666667%}.col-sm-push-12{left:100%}.col-sm-offset-0{margin-left:0}.col-sm-offset-1{margin-left:8.333333%}.col-sm-offset-2{margin-left:16.666667%}.col-sm-offset-3{margin-left:25%}.col-sm-offset-4{margin-left:33.333333%}.col-sm-offset-5{margin-left:41.666667%}.col-sm-offset-6{margin-left:50%}.col-sm-offset-7{margin-left:58.333333%}.col-sm-offset-8{margin-left:66.666667%}.col-sm-offset-9{margin-left:75%}.col-sm-offset-10{margin-left:83.333333%}.col-sm-offset-11{margin-left:91.666667%}.col-sm-offset-12{margin-left:100%}}@media (min-width:48em){.container{max-width:45rem}.col-md-1,.col-md-10,.col-md-11,.col-md-12,.col-md-2,.col-md-3,.col-md-4,.col-md-5,.col-md-6,.col-md-7,.col-md-8,.col-md-9{float:left}.col-md-1{width:8.333333%}.col-md-2{width:16.666667%}.col-md-3{width:25%}.col-md-4{width:33.333333%}.col-md-5{width:41.666667%}.col-md-6{width:50%}.col-md-7{width:58.333333%}.col-md-8{width:66.666667%}.col-md-9{width:75%}.col-md-10{width:83.333333%}.col-md-11{width:91.666667%}.col-md-12{width:100%}.col-md-pull-0{right:auto}.col-md-pull-1{right:8.333333%}.col-md-pull-2{right:16.666667%}.col-md-pull-3{right:25%}.col-md-pull-4{right:33.333333%}.col-md-pull-5{right:41.666667%}.col-md-pull-6{right:50%}.col-md-pull-7{right:58.333333%}.col-md-pull-8{right:66.666667%}.col-md-pull-9{right:75%}.col-md-pull-10{right:83.333333%}.col-md-pull-11{right:91.666667%}.col-md-pull-12{right:100%}.col-md-push-0{left:auto}.col-md-push-1{left:8.333333%}.col-md-push-2{left:16.666667%}.col-md-push-3{left:25%}.col-md-push-4{left:33.333333%}.col-md-push-5{left:41.666667%}.col-md-push-6{left:50%}.col-md-push-7{left:58.333333%}.col-md-push-8{left:66.666667%}.col-md-push-9{left:75%}.col-md-push-10{left:83.333333%}.col-md-push-11{left:91.666667%}.col-md-push-12{left:100%}.col-md-offset-0{margin-left:0}.col-md-offset-1{margin-left:8.333333%}.col-md-offset-2{margin-left:16.666667%}.col-md-offset-3{margin-left:25%}.col-md-offset-4{margin-left:33.333333%}.col-md-offset-5{margin-left:41.666667%}.col-md-offset-6{margin-left:50%}.col-md-offset-7{margin-left:58.333333%}.col-md-offset-8{margin-left:66.666667%}.col-md-offset-9{margin-left:75%}.col-md-offset-10{margin-left:83.333333%}.col-md-offset-11{margin-left:91.666667%}.col-md-offset-12{margin-left:100%}}@media (min-width:62em){.container{max-width:60rem}.col-lg-1,.col-lg-10,.col-lg-11,.col-lg-12,.col-lg-2,.col-lg-3,.col-lg-4,.col-lg-5,.col-lg-6,.col-lg-7,.col-lg-8,.col-lg-9{float:left}.col-lg-1{width:8.333333%}.col-lg-2{width:16.666667%}.col-lg-3{width:25%}.col-lg-4{width:33.333333%}.col-lg-5{width:41.666667%}.col-lg-6{width:50%}.col-lg-7{width:58.333333%}.col-lg-8{width:66.666667%}.col-lg-9{width:75%}.col-lg-10{width:83.333333%}.col-lg-11{width:91.666667%}.col-lg-12{width:100%}.col-lg-pull-0{right:auto}.col-lg-pull-1{right:8.333333%}.col-lg-pull-2{right:16.666667%}.col-lg-pull-3{right:25%}.col-lg-pull-4{right:33.333333%}.col-lg-pull-5{right:41.666667%}.col-lg-pull-6{right:50%}.col-lg-pull-7{right:58.333333%}.col-lg-pull-8{right:66.666667%}.col-lg-pull-9{right:75%}.col-lg-pull-10{right:83.333333%}.col-lg-pull-11{right:91.666667%}.col-lg-pull-12{right:100%}.col-lg-push-0{left:auto}.col-lg-push-1{left:8.333333%}.col-lg-push-2{left:16.666667%}.col-lg-push-3{left:25%}.col-lg-push-4{left:33.333333%}.col-lg-push-5{left:41.666667%}.col-lg-push-6{left:50%}.col-lg-push-7{left:58.333333%}.col-lg-push-8{left:66.666667%}.col-lg-push-9{left:75%}.col-lg-push-10{left:83.333333%}.col-lg-push-11{left:91.666667%}.col-lg-push-12{left:100%}.col-lg-offset-0{margin-left:0}.col-lg-offset-1{margin-left:8.333333%}.col-lg-offset-2{margin-left:16.666667%}.col-lg-offset-3{margin-left:25%}.col-lg-offset-4{margin-left:33.333333%}.col-lg-offset-5{margin-left:41.666667%}.col-lg-offset-6{margin-left:50%}.col-lg-offset-7{margin-left:58.333333%}.col-lg-offset-8{margin-left:66.666667%}.col-lg-offset-9{margin-left:75%}.col-lg-offset-10{margin-left:83.333333%}.col-lg-offset-11{margin-left:91.666667%}.col-lg-offset-12{margin-left:100%}}@media (min-width:75em){.container{max-width:72.25rem}.col-xl-1,.col-xl-10,.col-xl-11,.col-xl-12,.col-xl-2,.col-xl-3,.col-xl-4,.col-xl-5,.col-xl-6,.col-xl-7,.col-xl-8,.col-xl-9{float:left}.col-xl-1{width:8.333333%}.col-xl-2{width:16.666667%}.col-xl-3{width:25%}.col-xl-4{width:33.333333%}.col-xl-5{width:41.666667%}.col-xl-6{width:50%}.col-xl-7{width:58.333333%}.col-xl-8{width:66.666667%}.col-xl-9{width:75%}.col-xl-10{width:83.333333%}.col-xl-11{width:91.666667%}.col-xl-12{width:100%}.col-xl-pull-0{right:auto}.col-xl-pull-1{right:8.333333%}.col-xl-pull-2{right:16.666667%}.col-xl-pull-3{right:25%}.col-xl-pull-4{right:33.333333%}.col-xl-pull-5{right:41.666667%}.col-xl-pull-6{right:50%}.col-xl-pull-7{right:58.333333%}.col-xl-pull-8{right:66.666667%}.col-xl-pull-9{right:75%}.col-xl-pull-10{right:83.333333%}.col-xl-pull-11{right:91.666667%}.col-xl-pull-12{right:100%}.col-xl-push-0{left:auto}.col-xl-push-1{left:8.333333%}.col-xl-push-2{left:16.666667%}.col-xl-push-3{left:25%}.col-xl-push-4{left:33.333333%}.col-xl-push-5{left:41.666667%}.col-xl-push-6{left:50%}.col-xl-push-7{left:58.333333%}.col-xl-push-8{left:66.666667%}.col-xl-push-9{left:75%}.col-xl-push-10{left:83.333333%}.col-xl-push-11{left:91.666667%}.col-xl-push-12{left:100%}.col-xl-offset-0{margin-left:0}.col-xl-offset-1{margin-left:8.333333%}.col-xl-offset-2{margin-left:16.666667%}.col-xl-offset-3{margin-left:25%}.col-xl-offset-4{margin-left:33.333333%}.col-xl-offset-5{margin-left:41.666667%}.col-xl-offset-6{margin-left:50%}.col-xl-offset-7{margin-left:58.333333%}.col-xl-offset-8{margin-left:66.666667%}.col-xl-offset-9{margin-left:75%}.col-xl-offset-10{margin-left:83.333333%}.col-xl-offset-11{margin-left:91.666667%}.col-xl-offset-12{margin-left:100%}}.table{width:100%;max-width:100%}.table td,.table th{padding:.75rem;line-height:1.5;vertical-align:top;border-top:1px solid #eceeef}.table thead th{vertical-align:bottom;border-bottom:2px solid #eceeef}.table tbody+tbody{border-top:2px solid #eceeef}.table .table{background-color:#fff}.table-sm td,.table-sm th{padding:.3rem}.table-bordered,.table-bordered td,.table-bordered th{border:1px solid #eceeef}.table-bordered thead td,.table-bordered thead th{border-bottom-width:2px}.table-striped tbody tr:nth-of-type(odd){background-color:#f9f9f9}.table-active,.table-active>td,.table-active>th,.table-hover tbody tr:hover{background-color:#f5f5f5}.table-hover .table-active:hover,.table-hover .table-active:hover>td,.table-hover .table-active:hover>th{background-color:#e8e8e8}.table-success,.table-success>td,.table-success>th{background-color:#dff0d8}.table-hover .table-success:hover,.table-hover .table-success:hover>td,.table-hover .table-success:hover>th{background-color:#d0e9c6}.table-info,.table-info>td,.table-info>th{background-color:#d9edf7}.table-hover .table-info:hover,.table-hover .table-info:hover>td,.table-hover .table-info:hover>th{background-color:#c4e3f3}.table-warning,.table-warning>td,.table-warning>th{background-color:#fcf8e3}.table-hover .table-warning:hover,.table-hover .table-warning:hover>td,.table-hover .table-warning:hover>th{background-color:#faf2cc}.table-danger,.table-danger>td,.table-danger>th{background-color:#f2dede}.table-hover .table-danger:hover,.table-hover .table-danger:hover>td,.table-hover .table-danger:hover>th{background-color:#ebcccc}.table-responsive{display:block;width:100%;overflow-x:auto}.collapsing,.dropdown-divider,.embed-responsive,.modal,.modal-open,.navbar-divider{overflow:hidden}.thead-inverse th{color:#fff;background-color:#373a3c}.thead-default th{color:#55595c;background-color:#eceeef}.table-inverse{color:#eceeef;background-color:#373a3c}.table-inverse.table-bordered{border:0}.table-inverse td,.table-inverse th,.table-inverse thead th{border-color:#55595c}.table-reflow tbody{display:block;white-space:nowrap}.table-reflow td,.table-reflow th{border-top:1px solid #eceeef;border-left:1px solid #eceeef}.table-reflow td:last-child,.table-reflow th:last-child{border-right:1px solid #eceeef}.table-reflow tbody:last-child tr:last-child td,.table-reflow tbody:last-child tr:last-child th,.table-reflow tfoot:last-child tr:last-child td,.table-reflow tfoot:last-child tr:last-child th,.table-reflow thead:last-child tr:last-child td,.table-reflow thead:last-child tr:last-child th{border-bottom:1px solid #eceeef}.table-reflow tr td,.table-reflow tr th{display:block!important;border:1px solid #eceeef}.form-control,.form-control-file,.form-control-range{display:block}.form-control{width:100%;padding:.375rem .75rem;font-size:1rem;line-height:1.5;color:#55595c;background-color:#fff;background-image:none;border:.0625rem solid #ccc;border-radius:.25rem}.form-control::-ms-expand{background-color:transparent;border:0}.form-control:focus{border-color:#66afe9;outline:0}.form-control::-webkit-input-placeholder{color:#999;opacity:1}.form-control::-moz-placeholder{color:#999;opacity:1}.form-control:-ms-input-placeholder{color:#999;opacity:1}.form-control::placeholder{color:#999;opacity:1}.has-success .checkbox,.has-success .checkbox-inline,.has-success .control-label,.has-success .form-control-feedback,.has-success .help-block,.has-success .radio,.has-success .radio-inline,.has-success.checkbox label,.has-success.checkbox-inline label,.has-success.radio label,.has-success.radio-inline label{color:#5cb85c}.form-control:disabled,.form-control[readonly],fieldset[disabled] .form-control{background-color:#eceeef;opacity:1}.form-control[disabled],fieldset[disabled] .form-control{cursor:not-allowed}.form-control-label{padding:.4375rem .75rem;margin-bottom:0}@media screen and (-webkit-min-device-pixel-ratio:0){input[type=date].form-control,input[type=time].form-control,input[type=datetime-local].form-control,input[type=month].form-control{line-height:2.375rem}.input-group-sm input[type=date].form-control,.input-group-sm input[type=time].form-control,.input-group-sm input[type=datetime-local].form-control,.input-group-sm input[type=month].form-control,input[type=date].input-sm,input[type=time].input-sm,input[type=datetime-local].input-sm,input[type=month].input-sm{line-height:1.95rem}.input-group-lg input[type=date].form-control,.input-group-lg input[type=time].form-control,.input-group-lg input[type=datetime-local].form-control,.input-group-lg input[type=month].form-control,input[type=date].input-lg,input[type=time].input-lg,input[type=datetime-local].input-lg,input[type=month].input-lg{line-height:3.291667rem}}.form-control-static{min-height:2.375rem;padding-top:.4375rem;padding-bottom:.4375rem;margin-bottom:0}.form-control-static.form-control-lg,.form-control-static.form-control-sm,.input-group-lg>.form-control-static.form-control,.input-group-lg>.form-control-static.input-group-addon,.input-group-lg>.input-group-btn>.form-control-static.btn,.input-group-sm>.form-control-static.form-control,.input-group-sm>.form-control-static.input-group-addon,.input-group-sm>.input-group-btn>.form-control-static.btn{padding-right:0;padding-left:0}.form-control-sm,.input-group-sm>.form-control,.input-group-sm>.input-group-addon,.input-group-sm>.input-group-btn>.btn{padding:.275rem .75rem;font-size:.85rem;line-height:1.5;border-radius:.2rem}.form-control-lg,.input-group-lg>.form-control,.input-group-lg>.input-group-addon,.input-group-lg>.input-group-btn>.btn{padding:.75rem 1.25rem;font-size:1.25rem;line-height:1.333333;border-radius:.3rem}.form-group{margin-bottom:15px}.checkbox,.radio{position:relative;display:block;margin-bottom:.75rem}.checkbox label,.checkbox-inline,.radio label,.radio-inline{padding-left:1.25rem;margin-bottom:0;cursor:pointer;font-weight:400}.checkbox label input:only-child,.radio label input:only-child{position:static}.checkbox input[type=checkbox],.checkbox-inline input[type=checkbox],.radio input[type=radio],.radio-inline input[type=radio]{position:absolute;margin-top:.25rem;margin-left:-1.25rem}.collapsing,.dropdown,.dropup{position:relative}.checkbox+.checkbox,.radio+.radio{margin-top:-.25rem}.checkbox-inline,.radio-inline{position:relative;display:inline-block;vertical-align:middle}.checkbox-inline+.checkbox-inline,.radio-inline+.radio-inline{margin-top:0;margin-left:.75rem}fieldset[disabled] input[type=checkbox],fieldset[disabled] input[type=radio],input[type=checkbox].disabled,input[type=checkbox]:disabled,input[type=radio].disabled,input[type=radio]:disabled{cursor:not-allowed}.checkbox-inline.disabled,.checkbox.disabled label,.radio-inline.disabled,.radio.disabled label,fieldset[disabled] .checkbox label,fieldset[disabled] .checkbox-inline,fieldset[disabled] .radio label,fieldset[disabled] .radio-inline{cursor:not-allowed}.form-control-error,.form-control-success,.form-control-warning{padding-right:2.25rem;background-repeat:no-repeat;background-position:center right .59375rem;-webkit-background-size:1.54375rem 1.54375rem;background-size:1.54375rem 1.54375rem}.has-success .form-control{border-color:#5cb85c}.has-success .input-group-addon{color:#5cb85c;background-color:#eaf6ea;border-color:#5cb85c}.has-warning .checkbox,.has-warning .checkbox-inline,.has-warning .control-label,.has-warning .form-control-feedback,.has-warning .help-block,.has-warning .radio,.has-warning .radio-inline,.has-warning.checkbox label,.has-warning.checkbox-inline label,.has-warning.radio label,.has-warning.radio-inline label{color:#f0ad4e}.has-success .form-control-success{background-image:url(data:image/svg+xml;base64,)}.has-warning .form-control{border-color:#f0ad4e}.has-warning .input-group-addon{color:#f0ad4e;background-color:#fff;border-color:#f0ad4e}.has-error .checkbox,.has-error .checkbox-inline,.has-error .control-label,.has-error .form-control-feedback,.has-error .help-block,.has-error .radio,.has-error .radio-inline,.has-error.checkbox label,.has-error.checkbox-inline label,.has-error.radio label,.has-error.radio-inline label{color:#d9534f}.has-warning .form-control-warning{background-image:url(data:image/svg+xml;base64,)}.has-error .form-control{border-color:#d9534f}.has-error .input-group-addon{color:#d9534f;background-color:#fdf7f7;border-color:#d9534f}.has-error .form-control-error{background-image:url(data:image/svg+xml;base64,)}.btn-danger-outline,.btn-info-outline,.btn-info.active,.btn-info:active,.btn-primary-outline,.btn-primary.active,.btn-primary:active,.btn-secondary-outline,.btn-secondary.active,.btn-secondary:active,.btn-success-outline,.btn-success.active,.btn-success:active,.btn-warning-outline,.btn-warning.active,.btn-warning:active,.btn.active,.btn:active,.open>.btn-info.dropdown-toggle,.open>.btn-primary.dropdown-toggle,.open>.btn-secondary.dropdown-toggle,.open>.btn-success.dropdown-toggle,.open>.btn-warning.dropdown-toggle{background-image:none}@media (min-width:34em){.form-inline .form-control-static,.form-inline .form-group{display:inline-block}.form-inline .control-label,.form-inline .form-group{margin-bottom:0;vertical-align:middle}.form-inline .form-control{display:inline-block;width:auto;vertical-align:middle}.form-inline .input-group{display:inline-table;vertical-align:middle}.form-inline .input-group .form-control,.form-inline .input-group .input-group-addon,.form-inline .input-group .input-group-btn{width:auto}.form-inline .input-group>.form-control{width:100%}.form-inline .checkbox,.form-inline .radio{display:inline-block;margin-top:0;margin-bottom:0;vertical-align:middle}.form-inline .checkbox label,.form-inline .radio label{padding-left:0}.form-inline .checkbox input[type=checkbox],.form-inline .radio input[type=radio]{position:relative;margin-left:0}.form-inline .has-feedback .form-control-feedback{top:0}}.btn-block,input[type=button].btn-block,input[type=reset].btn-block,input[type=submit].btn-block{width:100%}.btn{display:inline-block;padding:.375rem 1rem;font-size:1rem;font-weight:400;line-height:1.5;text-align:center;white-space:nowrap;vertical-align:middle;-ms-touch-action:manipulation;touch-action:manipulation;cursor:pointer;user-select:none;border:.0625rem solid transparent;border-radius:.25rem}.btn.active.focus,.btn.active:focus,.btn.focus,.btn:active.focus,.btn:active:focus,.btn:focus{outline:dotted thin;outline:-webkit-focus-ring-color auto 5px;outline-offset:-2px}.btn.focus,.btn:focus,.btn:hover{text-decoration:none}.btn.active,.btn:active{outline:0}.btn.disabled,.btn:disabled,fieldset[disabled] .btn{cursor:not-allowed;opacity:.65}a.btn.disaabled,fieldset[disabled] a.btn{pointer-events:none}.btn-primary{color:#fff;background-color:#0275d8;border-color:#0275d8}.btn-primary.active,.btn-primary.focus,.btn-primary:active,.btn-primary:focus,.btn-primary:hover,.open>.btn-primary.dropdown-toggle{color:#fff;background-color:#025aa5;border-color:#01549b}.btn-primary.disabled.focus,.btn-primary.disabled:focus,.btn-primary:disabled.focus,.btn-primary:disabled:focus,fieldset[disabled] .btn-primary.focus,fieldset[disabled] .btn-primary:focus{background-color:#0275d8;border-color:#0275d8}.btn-primary.disabled:hover,.btn-primary:disabled:hover,fieldset[disabled] .btn-primary:hover{background-color:#0275d8;border-color:#0275d8}.btn-secondary{color:#373a3c;background-color:#fff;border-color:#ccc}.btn-secondary.active,.btn-secondary.focus,.btn-secondary:active,.btn-secondary:focus,.btn-secondary:hover,.open>.btn-secondary.dropdown-toggle{color:#373a3c;background-color:#e6e6e6;border-color:#adadad}.btn-secondary.disabled.focus,.btn-secondary.disabled:focus,.btn-secondary:disabled.focus,.btn-secondary:disabled:focus,fieldset[disabled] .btn-secondary.focus,fieldset[disabled] .btn-secondary:focus{background-color:#fff;border-color:#ccc}.btn-secondary.disabled:hover,.btn-secondary:disabled:hover,fieldset[disabled] .btn-secondary:hover{background-color:#fff;border-color:#ccc}.btn-info{color:#fff;background-color:#5bc0de;border-color:#5bc0de}.btn-info.active,.btn-info.focus,.btn-info:active,.btn-info:focus,.btn-info:hover,.open>.btn-info.dropdown-toggle{color:#fff;background-color:#31b0d5;border-color:#2aabd2}.btn-info.disabled.focus,.btn-info.disabled:focus,.btn-info:disabled.focus,.btn-info:disabled:focus,fieldset[disabled] .btn-info.focus,fieldset[disabled] .btn-info:focus{background-color:#5bc0de;border-color:#5bc0de}.btn-info.disabled:hover,.btn-info:disabled:hover,fieldset[disabled] .btn-info:hover{background-color:#5bc0de;border-color:#5bc0de}.btn-success{color:#fff;background-color:#5cb85c;border-color:#5cb85c}.btn-success.active,.btn-success.focus,.btn-success:active,.btn-success:focus,.btn-success:hover,.open>.btn-success.dropdown-toggle{color:#fff;background-color:#449d44;border-color:#419641}.btn-success.disabled.focus,.btn-success.disabled:focus,.btn-success:disabled.focus,.btn-success:disabled:focus,fieldset[disabled] .btn-success.focus,fieldset[disabled] .btn-success:focus{background-color:#5cb85c;border-color:#5cb85c}.btn-success.disabled:hover,.btn-success:disabled:hover,fieldset[disabled] .btn-success:hover{background-color:#5cb85c;border-color:#5cb85c}.btn-warning{color:#fff;background-color:#f0ad4e;border-color:#f0ad4e}.btn-warning.active,.btn-warning.focus,.btn-warning:active,.btn-warning:focus,.btn-warning:hover,.open>.btn-warning.dropdown-toggle{color:#fff;background-color:#ec971f;border-color:#eb9316}.btn-warning.disabled.focus,.btn-warning.disabled:focus,.btn-warning:disabled.focus,.btn-warning:disabled:focus,fieldset[disabled] .btn-warning.focus,fieldset[disabled] .btn-warning:focus{background-color:#f0ad4e;border-color:#f0ad4e}.btn-warning.disabled:hover,.btn-warning:disabled:hover,fieldset[disabled] .btn-warning:hover{background-color:#f0ad4e;border-color:#f0ad4e}.btn-danger{color:#fff;background-color:#d9534f;border-color:#d9534f}.btn-danger.active,.btn-danger.focus,.btn-danger:active,.btn-danger:focus,.btn-danger:hover,.open>.btn-danger.dropdown-toggle{color:#fff;background-color:#c9302c;border-color:#c12e2a}.btn-danger.active,.btn-danger:active,.open>.btn-danger.dropdown-toggle{background-image:none}.btn-danger.disabled.focus,.btn-danger.disabled:focus,.btn-danger:disabled.focus,.btn-danger:disabled:focus,fieldset[disabled] .btn-danger.focus,fieldset[disabled] .btn-danger:focus{background-color:#d9534f;border-color:#d9534f}.btn-danger.disabled:hover,.btn-danger:disabled:hover,fieldset[disabled] .btn-danger:hover{background-color:#d9534f;border-color:#d9534f}.btn-primary-outline{color:#0275d8;background-color:transparent;border-color:#0275d8}.btn-primary-outline.active,.btn-primary-outline.focus,.btn-primary-outline:active,.btn-primary-outline:focus,.btn-primary-outline:hover,.open>.btn-primary-outline.dropdown-toggle{color:#fff;background-color:#0275d8;border-color:#0275d8}.btn-primary-outline.disabled.focus,.btn-primary-outline.disabled:focus,.btn-primary-outline:disabled.focus,.btn-primary-outline:disabled:focus,fieldset[disabled] .btn-primary-outline.focus,fieldset[disabled] .btn-primary-outline:focus{border-color:#43a7fd}.btn-primary-outline.disabled:hover,.btn-primary-outline:disabled:hover,fieldset[disabled] .btn-primary-outline:hover{border-color:#43a7fd}.btn-secondary-outline{color:#ccc;background-color:transparent;border-color:#ccc}.btn-secondary-outline.active,.btn-secondary-outline.focus,.btn-secondary-outline:active,.btn-secondary-outline:focus,.btn-secondary-outline:hover,.open>.btn-secondary-outline.dropdown-toggle{color:#fff;background-color:#ccc;border-color:#ccc}.btn-secondary-outline.disabled.focus,.btn-secondary-outline.disabled:focus,.btn-secondary-outline:disabled.focus,.btn-secondary-outline:disabled:focus,fieldset[disabled] .btn-secondary-outline.focus,fieldset[disabled] .btn-secondary-outline:focus{border-color:#fff}.btn-secondary-outline.disabled:hover,.btn-secondary-outline:disabled:hover,fieldset[disabled] .btn-secondary-outline:hover{border-color:#fff}.btn-info-outline{color:#5bc0de;background-color:transparent;border-color:#5bc0de}.btn-info-outline.active,.btn-info-outline.focus,.btn-info-outline:active,.btn-info-outline:focus,.btn-info-outline:hover,.open>.btn-info-outline.dropdown-toggle{color:#fff;background-color:#5bc0de;border-color:#5bc0de}.btn-info-outline.disabled.focus,.btn-info-outline.disabled:focus,.btn-info-outline:disabled.focus,.btn-info-outline:disabled:focus,fieldset[disabled] .btn-info-outline.focus,fieldset[disabled] .btn-info-outline:focus{border-color:#b0e1ef}.btn-info-outline.disabled:hover,.btn-info-outline:disabled:hover,fieldset[disabled] .btn-info-outline:hover{border-color:#b0e1ef}.btn-success-outline{color:#5cb85c;background-color:transparent;border-color:#5cb85c}.btn-success-outline.active,.btn-success-outline.focus,.btn-success-outline:active,.btn-success-outline:focus,.btn-success-outline:hover,.open>.btn-success-outline.dropdown-toggle{color:#fff;background-color:#5cb85c;border-color:#5cb85c}.btn-success-outline.disabled.focus,.btn-success-outline.disabled:focus,.btn-success-outline:disabled.focus,.btn-success-outline:disabled:focus,fieldset[disabled] .btn-success-outline.focus,fieldset[disabled] .btn-success-outline:focus{border-color:#a3d7a3}.btn-success-outline.disabled:hover,.btn-success-outline:disabled:hover,fieldset[disabled] .btn-success-outline:hover{border-color:#a3d7a3}.btn-warning-outline{color:#f0ad4e;background-color:transparent;border-color:#f0ad4e}.btn-warning-outline.active,.btn-warning-outline.focus,.btn-warning-outline:active,.btn-warning-outline:focus,.btn-warning-outline:hover,.open>.btn-warning-outline.dropdown-toggle{color:#fff;background-color:#f0ad4e;border-color:#f0ad4e}.btn-warning-outline.disabled.focus,.btn-warning-outline.disabled:focus,.btn-warning-outline:disabled.focus,.btn-warning-outline:disabled:focus,fieldset[disabled] .btn-warning-outline.focus,fieldset[disabled] .btn-warning-outline:focus{border-color:#f8d9ac}.btn-warning-outline.disabled:hover,.btn-warning-outline:disabled:hover,fieldset[disabled] .btn-warning-outline:hover{border-color:#f8d9ac}.btn-danger-outline{color:#d9534f;background-color:transparent;border-color:#d9534f}.btn-danger-outline.active,.btn-danger-outline.focus,.btn-danger-outline:active,.btn-danger-outline:focus,.btn-danger-outline:hover,.open>.btn-danger-outline.dropdown-toggle{color:#fff;background-color:#d9534f;border-color:#d9534f}.btn-danger-outline.disabled.focus,.btn-danger-outline.disabled:focus,.btn-danger-outline:disabled.focus,.btn-danger-outline:disabled:focus,fieldset[disabled] .btn-danger-outline.focus,fieldset[disabled] .btn-danger-outline:focus{border-color:#eba5a3}.btn-danger-outline.disabled:hover,.btn-danger-outline:disabled:hover,fieldset[disabled] .btn-danger-outline:hover{border-color:#eba5a3}.btn-link{font-weight:400;color:#0275d8;border-radius:0}.btn-link,.btn-link.active,.btn-link:active,.btn-link:disabled,fieldset[disabled] .btn-link{background-color:transparent}.btn-link,.btn-link:active,.btn-link:focus,.btn-link:hover{border-color:transparent}.btn-link:focus,.btn-link:hover{color:#014c8c;text-decoration:underline;background-color:transparent}.btn-link:disabled:focus,.btn-link:disabled:hover,fieldset[disabled] .btn-link:focus,fieldset[disabled] .btn-link:hover{color:#818a91;text-decoration:none}.btn-group-lg>.btn,.btn-lg{padding:.75rem 1.25rem;font-size:1.25rem;line-height:1.333333;border-radius:.3rem}.btn-group-sm>.btn,.btn-sm{padding:.25rem .75rem;font-size:.85rem;line-height:1.5;border-radius:.2rem}.btn-block{display:block}.btn-block+.btn-block{margin-top:5px}.fade{opacity:0;-webkit-transition:opacity .15s linear;-o-transition:opacity .15s linear;transition:opacity .15s linear}.fade.in{opacity:1}.collapse{display:none}.collapse.in{display:block}.collapsing{height:0;-webkit-transition-timing-function:ease;-o-transition-timing-function:ease;transition-timing-function:ease;-webkit-transition-duration:.35s;-o-transition-duration:.35s;transition-duration:.35s;-webkit-transition-property:height;-o-transition-property:height;transition-property:height}.dropdown-toggle:after{display:inline-block;width:0;height:0;margin-left:.25rem;vertical-align:middle;content:"";border-top:.3em solid;border-right:.3em solid transparent;border-left:.3em solid transparent}.dropdown-toggle:focus{outline:0}.dropdown-menu{position:absolute;top:100%;left:0;z-index:1000;display:none;min-width:160px;padding:5px 0;margin:2px 0 0;font-size:1rem;text-align:left;list-style:none;background-color:#fff;-webkit-background-clip:padding-box;background-clip:padding-box;border:1px solid rgba(0,0,0,.15);border-radius:.25rem}.dropdown-header,.dropdown-item{display:block;padding:3px 20px;line-height:1.5;white-space:nowrap}.dropdown-divider{height:1px;margin:.5rem 0;background-color:#e5e5e5}.dropdown-item{width:100%;font-weight:400;color:#373a3c;text-align:inherit;background:0 0;border:0}.c-indicator,.label,.pager{text-align:center}.dropdown-item:focus,.dropdown-item:hover{color:#2b2d2f;text-decoration:none;background-color:#f5f5f5}.dropdown-item.active,.dropdown-item.active:focus,.dropdown-item.active:hover{color:#fff;text-decoration:none;background-color:#0275d8;outline:0}.dropdown-item.disabled,.dropdown-item.disabled:focus,.dropdown-item.disabled:hover{color:#818a91}.dropdown-item.disabled:focus,.dropdown-item.disabled:hover{text-decoration:none;cursor:not-allowed;background-color:transparent;background-image:none;filter:"progid:DXImageTransform.Microsoft.gradient(enabled = false)"}.c-input,.file{cursor:pointer}.open>.dropdown-menu{display:block}.open>a{outline:0}.dropdown-menu-right{right:0;left:auto}.dropdown-menu-left{right:auto;left:0}.dropdown-header{font-size:.85rem;color:#818a91}.dropdown-backdrop{position:fixed;top:0;right:0;bottom:0;left:0;z-index:990}.btn-group-vertical>.btn.active,.btn-group-vertical>.btn:active,.btn-group-vertical>.btn:focus,.btn-group-vertical>.btn:hover,.btn-group>.btn.active,.btn-group>.btn:active,.btn-group>.btn:focus,.btn-group>.btn:hover,.input-group-btn>.btn:active,.input-group-btn>.btn:focus,.input-group-btn>.btn:hover{z-index:2}.pull-right>.dropdown-menu{right:0;left:auto}.dropup .caret,.navbar-fixed-bottom .dropdown .caret{content:"";border-top:0;border-bottom:.3em solid}.dropup .dropdown-menu,.navbar-fixed-bottom .dropdown .dropdown-menu{top:auto;bottom:100%;margin-bottom:2px}.btn-group,.btn-group-vertical{position:relative;display:inline-block;vertical-align:middle}.btn-group-vertical>.btn,.btn-group>.btn{position:relative;float:left}.btn-group .btn+.btn,.btn-group .btn+.btn-group,.btn-group .btn-group+.btn,.btn-group .btn-group+.btn-group{margin-left:-1px}.btn-toolbar{margin-left:-5px}.btn-toolbar:after,.btn-toolbar:before{display:table;content:" "}.btn-toolbar>.btn,.btn-toolbar>.btn-group,.btn-toolbar>.input-group{margin-left:5px}.btn .caret,.btn-group>.btn:first-child{margin-left:0}.btn-group>.btn:not(:first-child):not(:last-child):not(.dropdown-toggle){border-radius:0}.btn-group>.btn:first-child:not(:last-child):not(.dropdown-toggle){border-top-right-radius:0;border-bottom-right-radius:0}.btn-group>.btn:last-child:not(:first-child),.btn-group>.dropdown-toggle:not(:first-child){border-top-left-radius:0;border-bottom-left-radius:0}.btn-group>.btn-group:not(:first-child):not(:last-child)>.btn{border-radius:0}.btn-group>.btn-group:first-child:not(:last-child)>.btn:last-child,.btn-group>.btn-group:first-child:not(:last-child)>.dropdown-toggle{border-top-right-radius:0;border-bottom-right-radius:0}.btn-group>.btn-group:last-child:not(:first-child)>.btn:first-child{border-top-left-radius:0;border-bottom-left-radius:0}.btn-group .dropdown-toggle:active,.btn-group.open .dropdown-toggle{outline:0}.btn-group>.btn+.dropdown-toggle{padding-right:8px;padding-left:8px}.btn-group-lg.btn-group>.btn+.dropdown-toggle,.btn-group>.btn-lg+.dropdown-toggle{padding-right:12px;padding-left:12px}.btn-group-lg>.btn .caret,.btn-lg .caret{border-width:.3em .3em 0}.dropup .btn-group-lg>.btn .caret,.dropup .btn-lg .caret{border-width:0 .3em .3em}.btn-group-vertical>.btn,.btn-group-vertical>.btn-group,.btn-group-vertical>.btn-group>.btn{display:block;float:none;width:100%;max-width:100%}.btn-group-vertical>.btn-group:after,.btn-group-vertical>.btn-group:before{display:table;content:" "}.btn-group-vertical>.btn-group:after{clear:both}.btn-group-vertical>.btn-group>.btn{float:none}.btn-group-vertical>.btn+.btn,.btn-group-vertical>.btn+.btn-group,.btn-group-vertical>.btn-group+.btn,.btn-group-vertical>.btn-group+.btn-group{margin-top:-1px;margin-left:0}.btn-group-vertical>.btn:not(:first-child):not(:last-child){border-radius:0}.btn-group-vertical>.btn:first-child:not(:last-child){border-top-right-radius:.25rem;border-bottom-right-radius:0;border-bottom-left-radius:0}.btn-group-vertical>.btn:last-child:not(:first-child){border-top-left-radius:0;border-top-right-radius:0;border-bottom-left-radius:.25rem}.btn-group-vertical>.btn-group:not(:first-child):not(:last-child)>.btn{border-radius:0}.btn-group-vertical>.btn-group:first-child:not(:last-child)>.btn:last-child,.btn-group-vertical>.btn-group:first-child:not(:last-child)>.dropdown-toggle{border-bottom-right-radius:0;border-bottom-left-radius:0}.btn-group-vertical>.btn-group:last-child:not(:first-child)>.btn:first-child{border-top-left-radius:0;border-top-right-radius:0}[data-toggle=buttons]>.btn input[type=checkbox],[data-toggle=buttons]>.btn input[type=radio],[data-toggle=buttons]>.btn-group>.btn input[type=checkbox],[data-toggle=buttons]>.btn-group>.btn input[type=radio]{position:absolute;clip:rect(0,0,0,0);pointer-events:none}.c-input,.input-group,.input-group-btn,.input-group-btn>.btn{position:relative}.input-group{display:table;border-collapse:separate}.input-group .form-control{position:relative;z-index:2;float:left;width:100%;margin-bottom:0}.input-group .form-control,.input-group-addon,.input-group-btn{display:table-cell}.input-group .form-control:not(:first-child):not(:last-child),.input-group-addon:not(:first-child):not(:last-child),.input-group-btn:not(:first-child):not(:last-child){border-radius:0}.input-group-addon,.input-group-btn{width:1%;white-space:nowrap;vertical-align:middle}.input-group-addon{padding:.375rem .75rem;font-size:1rem;font-weight:400;line-height:1;color:#55595c;text-align:center;background-color:#eceeef;border:1px solid #ccc;border-radius:.25rem}.alert-link,.close,.label{font-weight:700}.input-group-addon.form-control-sm,.input-group-sm>.input-group-addon,.input-group-sm>.input-group-btn>.input-group-addon.btn{padding:.275rem .75rem;font-size:.85rem;border-radius:.2rem}.input-group-addon.form-control-lg,.input-group-lg>.input-group-addon,.input-group-lg>.input-group-btn>.input-group-addon.btn{padding:1.25rem;font-size:1.25rem;border-radius:.3rem}.input-group-addon input[type=checkbox],.input-group-addon input[type=radio]{margin-top:0}.input-group .form-control:first-child,.input-group-addon:first-child,.input-group-btn:first-child>.btn,.input-group-btn:first-child>.btn-group>.btn,.input-group-btn:first-child>.dropdown-toggle,.input-group-btn:last-child>.btn-group:not(:last-child)>.btn,.input-group-btn:last-child>.btn:not(:last-child):not(.dropdown-toggle){border-top-right-radius:0;border-bottom-right-radius:0}.input-group-addon:first-child{border-right:0}.input-group .form-control:last-child,.input-group-addon:last-child,.input-group-btn:first-child>.btn-group:not(:first-child)>.btn,.input-group-btn:first-child>.btn:not(:first-child),.input-group-btn:last-child>.btn,.input-group-btn:last-child>.btn-group>.btn,.input-group-btn:last-child>.dropdown-toggle{border-top-left-radius:0;border-bottom-left-radius:0}.input-group-addon:last-child{border-left:0}.input-group-btn{font-size:0;white-space:nowrap}.input-group-btn>.btn+.btn{margin-left:-1px}.input-group-btn:first-child>.btn,.input-group-btn:first-child>.btn-group{margin-right:-1px}.input-group-btn:last-child>.btn,.input-group-btn:last-child>.btn-group{z-index:2;margin-left:-1px}.c-input{display:inline;padding-left:1.5rem;color:#555}.c-input>input{position:absolute;z-index:-1;opacity:0}.c-input>input:checked~.c-indicator{color:#fff;background-color:#0074d9}.c-input>input:active~.c-indicator{color:#fff;background-color:#84c6ff}.c-input+.c-input{margin-left:1rem}.c-indicator{position:absolute;top:0;left:0;display:block;width:1rem;height:1rem;font-size:65%;line-height:1rem;color:#eee;user-select:none;background-color:#eee;background-repeat:no-repeat;background-position:center center;-webkit-background-size:50% 50%;background-size:50% 50%}.c-checkbox .c-indicator{border-radius:.25rem}.c-checkbox input:checked~.c-indicator{background-image:url(data:image/svg+xml;base64,)}.c-checkbox input:indeterminate~.c-indicator{background-color:#0074d9;background-image:url(data:image/svg+xml;base64,)}.c-radio .c-indicator{border-radius:50%}.c-radio input:checked~.c-indicator{background-image:url(data:image/svg+xml;base64,)}.c-inputs-stacked .c-input{display:inline}.c-inputs-stacked .c-input:after{display:block;margin-bottom:.25rem;content:""}.c-select,.file{display:inline-block}.c-inputs-stacked .c-input+.c-input{margin-left:0}.c-select{max-width:100%;-webkit-appearance:none;padding:.375rem 1.75rem .375rem .75rem;padding-right:.75rem\9;vertical-align:middle;background:url() right .75rem center no-repeat #fff;background-image:none\9;-webkit-background-size:8px 10px;background-size:8px 10px;border:1px solid #ccc;-moz-appearance:none;appearance:none}.c-select:focus{border-color:#51a7e8;outline:0;-webkit-box-shadow:inset 0 1px 2px rgba(0,0,0,.075),0 0 5px rgba(81,167,232,.5);box-shadow:inset 0 1px 2px rgba(0,0,0,.075),0 0 5px rgba(81,167,232,.5)}.c-select::-ms-expand{opacity:0}.c-select-sm{padding-top:3px;padding-bottom:3px;font-size:12px}.c-select-sm:not([multiple]){height:26px;min-height:26px}.file{position:relative;height:2.5rem}.file-custom,.file-custom:before{position:absolute;height:2.5rem;padding:.5rem 1rem;line-height:1.5;color:#555}.file input{min-width:14rem;margin:0;filter:alpha(opacity=0);opacity:0}.file-custom{top:0;right:0;left:0;z-index:5;-webkit-user-select:none;-moz-user-select:none;-ms-user-select:none;user-select:none;background-color:#fff;border:.075rem solid #ddd;border-radius:.25rem;-webkit-box-shadow:inset 0 .2rem .4rem rgba(0,0,0,.05);box-shadow:inset 0 .2rem .4rem rgba(0,0,0,.05)}.file-custom:after{content:"Choose file..."}.file-custom:before{top:-.075rem;right:-.075rem;bottom:-.075rem;z-index:6;display:block;content:"Browse";background-color:#eee;border:.075rem solid #ddd;border-radius:0 .25rem .25rem 0}.file input:focus~.file-custom{-webkit-box-shadow:0 0 0 .075rem #fff,0 0 0 .2rem #0074d9;box-shadow:0 0 0 .075rem #fff,0 0 0 .2rem #0074d9}.nav{padding-left:0;margin-bottom:0;list-style:none}.nav-link{display:inline-block}.nav-link:focus,.nav-link:hover{text-decoration:none}.nav-link.disabled{color:#818a91}.nav-link.disabled,.nav-link.disabled:focus,.nav-link.disabled:hover{color:#818a91;cursor:not-allowed;background-color:transparent}.nav-inline .nav-link+.nav-link{margin-left:1rem}.nav-pills .nav-item+.nav-item,.nav-tabs .nav-item+.nav-item{margin-left:.2rem}.nav-tabs{border-bottom:1px solid #ddd}.nav-tabs:after,.nav-tabs:before{display:table;content:" "}.nav-tabs .nav-item{float:left;margin-bottom:-1px}.nav-tabs .nav-link{display:block;padding:.5em 1em;border:1px solid transparent;border-radius:.25rem .25rem 0 0}.nav-tabs .nav-link:focus,.nav-tabs .nav-link:hover{border-color:#eceeef #eceeef #ddd}.nav-tabs .nav-link.disabled,.nav-tabs .nav-link.disabled:focus,.nav-tabs .nav-link.disabled:hover{color:#818a91;background-color:transparent;border-color:transparent}.nav-tabs .nav-item.open .nav-link,.nav-tabs .nav-item.open .nav-link:focus,.nav-tabs .nav-item.open .nav-link:hover,.nav-tabs .nav-link.active,.nav-tabs .nav-link.active:focus,.nav-tabs .nav-link.active:hover{color:#55595c;background-color:#fff;border-color:#ddd #ddd transparent}.nav-pills .nav-item{float:left}.nav-pills .nav-link{display:block;padding:.5em 1em;border-radius:.25rem}.nav-pills .nav-item.open .nav-link,.nav-pills .nav-item.open .nav-link:focus,.nav-pills .nav-item.open .nav-link:hover,.nav-pills .nav-link.active,.nav-pills .nav-link.active:focus,.nav-pills .nav-link.active:hover{color:#fff;cursor:default;background-color:#0275d8}.nav-stacked .nav-item{display:block;float:none}.nav-stacked .nav-item+.nav-item{margin-top:.2rem;margin-left:0}.navbar-divider,.navbar-nav .nav-item+.nav-item,.navbar-nav .nav-link+.nav-link{margin-left:1rem}.tab-content>.tab-pane{display:none}.tab-content>.active{display:block}.nav-tabs .dropdown-menu{margin-top:-1px;border-top-left-radius:0;border-top-right-radius:0}.navbar{position:relative;padding:.5rem 1rem}.navbar:after,.navbar:before{display:table;content:" "}.navbar-static-top{z-index:1000}.navbar-fixed-bottom,.navbar-fixed-top{position:fixed;right:0;left:0;z-index:1030;margin-bottom:0}.card,.card-title{margin-bottom:.75rem}.navbar-fixed-top{top:0}.navbar-fixed-bottom{bottom:0}.navbar-sticky-top{position:-webkit-sticky;position:sticky;top:0;z-index:1030;width:100%}@media (min-width:34em){.navbar{border-radius:.25rem}.navbar-fixed-bottom,.navbar-fixed-top,.navbar-static-top,.navbar-sticky-top{border-radius:0}}.navbar-brand{float:left;padding-top:.25rem;padding-bottom:.25rem;margin-right:1rem;font-size:1.25rem}.navbar-brand:focus,.navbar-brand:hover{text-decoration:none}.navbar-brand>img{display:block}.navbar-divider{float:left;width:1px;padding-top:.425rem;padding-bottom:.425rem;margin-right:1rem}.navbar-divider:before{content:'\00a0'}.navbar-toggler{padding:.5rem .75rem;font-size:1.25rem;line-height:1;background:0 0;border:.0625rem solid transparent;border-radius:.25rem}.navbar-toggler:focus,.navbar-toggler:hover{text-decoration:none}@media (min-width:34em){.navbar-toggleable-xs{display:block!important}}@media (min-width:48em){.navbar-toggleable-sm{display:block!important}}.navbar-nav .nav-item{float:left}.navbar-nav .nav-link{display:block;padding-top:.425rem;padding-bottom:.425rem}.navbar-light .navbar-brand,.navbar-light .navbar-brand:focus,.navbar-light .navbar-brand:hover{color:rgba(0,0,0,.8)}.navbar-light .navbar-nav .nav-link{color:rgba(0,0,0,.3)}.navbar-light .navbar-nav .nav-link:focus,.navbar-light .navbar-nav .nav-link:hover{color:rgba(0,0,0,.6)}.navbar-light .navbar-nav .active>.nav-link,.navbar-light .navbar-nav .active>.nav-link:focus,.navbar-light .navbar-nav .active>.nav-link:hover,.navbar-light .navbar-nav .nav-link.active,.navbar-light .navbar-nav .nav-link.active:focus,.navbar-light .navbar-nav .nav-link.active:hover,.navbar-light .navbar-nav .nav-link.open,.navbar-light .navbar-nav .nav-link.open:focus,.navbar-light .navbar-nav .nav-link.open:hover,.navbar-light .navbar-nav .open>.nav-link,.navbar-light .navbar-nav .open>.nav-link:focus,.navbar-light .navbar-nav .open>.nav-link:hover{color:rgba(0,0,0,.8)}.navbar-light .navbar-divider{background-color:rgba(0,0,0,.075)}.navbar-dark .navbar-brand,.navbar-dark .navbar-brand:focus,.navbar-dark .navbar-brand:hover{color:#fff}.navbar-dark .navbar-nav .nav-link{color:rgba(255,255,255,.5)}.navbar-dark .navbar-nav .nav-link:focus,.navbar-dark .navbar-nav .nav-link:hover{color:rgba(255,255,255,.75)}.card-inverse .card-blockquote,.card-inverse .card-footer,.card-inverse .card-header,.card-inverse .card-title,.navbar-dark .navbar-nav .active>.nav-link,.navbar-dark .navbar-nav .active>.nav-link:focus,.navbar-dark .navbar-nav .active>.nav-link:hover,.navbar-dark .navbar-nav .nav-link.active,.navbar-dark .navbar-nav .nav-link.active:focus,.navbar-dark .navbar-nav .nav-link.active:hover,.navbar-dark .navbar-nav .nav-link.open,.navbar-dark .navbar-nav .nav-link.open:focus,.navbar-dark .navbar-nav .nav-link.open:hover,.navbar-dark .navbar-nav .open>.nav-link,.navbar-dark .navbar-nav .open>.nav-link:focus,.navbar-dark .navbar-nav .open>.nav-link:hover{color:#fff}.navbar-dark .navbar-divider{background-color:rgba(255,255,255,.075)}.card{position:relative;border:.0625rem solid #e5e5e5;border-radius:.25rem}.card-block{padding:1.25rem}.card-footer,.card-header{padding:.75rem 1.25rem;background-color:#f5f5f5}.card-title{margin-top:0}.card-subtitle{margin-top:-.375rem;margin-bottom:0}.card-text:last-child{margin-bottom:0}.card-link:hover{text-decoration:none}.card-link+.card-link{margin-left:1.25rem}.card>.list-group:first-child .list-group-item:first-child{border-radius:.25rem .25rem 0 0}.card>.list-group:last-child .list-group-item:last-child{border-radius:0 0 .25rem .25rem}.card-header{border-bottom:.0625rem solid #e5e5e5}.card-header:first-child{border-radius:.1875rem .1875rem 0 0}.card-footer{border-top:.0625rem solid #e5e5e5}.card-footer:last-child{border-radius:0 0 .1875rem .1875rem}.card-primary{background-color:#0275d8;border-color:#0275d8}.card-success{background-color:#5cb85c;border-color:#5cb85c}.card-info{background-color:#5bc0de;border-color:#5bc0de}.card-warning{background-color:#f0ad4e;border-color:#f0ad4e}.card-danger{background-color:#d9534f;border-color:#d9534f}.card-inverse .card-footer,.card-inverse .card-header{border-bottom:.075rem solid rgba(255,255,255,.2)}.card-inverse .card-blockquote>footer,.card-inverse .card-link,.card-inverse .card-text{color:rgba(255,255,255,.65)}.card-inverse .card-link:focus,.card-inverse .card-link:hover{color:#fff}.card-blockquote{padding:0;margin-bottom:0;border-left:0}.card-img{border-radius:.25rem}.card-img-overlay{position:absolute;top:0;right:0;bottom:0;left:0;padding:1.25rem}.card-img-top{border-radius:.25rem .25rem 0 0}.card-img-bottom{border-radius:0 0 .25rem .25rem}.card-deck{display:table;table-layout:fixed;border-spacing:1.25rem 0}.card-deck .card{display:table-cell;width:1%;vertical-align:top}.card-columns .card,.progress{width:100%}.card-deck-wrapper{margin-right:-1.25rem;margin-left:-1.25rem}.card-group{display:table;width:100%;table-layout:fixed}.card-group .card{display:table-cell;vertical-align:top}.breadcrumb>li,.card-columns .card,.pagination{display:inline-block}.card-group .card+.card{margin-left:0;border-left:0}.card-group .card:first-child .card-img-top{border-top-right-radius:0}.card-group .card:first-child .card-img-bottom{border-bottom-right-radius:0}.card-group .card:last-child .card-img-top{border-top-left-radius:0}.card-group .card:last-child .card-img-bottom{border-bottom-left-radius:0}.card-group .card:not(:first-child):not(:last-child){border-radius:0}.card-group .card:not(:first-child):not(:last-child) .card-img-bottom,.card-group .card:not(:first-child):not(:last-child) .card-img-top{border-radius:0}.breadcrumb,.pagination{border-radius:.25rem;margin-bottom:1rem}.card-columns{-webkit-column-count:3;-moz-column-count:3;column-count:3;-webkit-column-gap:1.25rem;-moz-column-gap:1.25rem;column-gap:1.25rem}.breadcrumb{padding:.75rem 1rem;list-style:none;background-color:#eceeef}.breadcrumb>li+li:before{padding-right:.5rem;padding-left:.5rem;color:#818a91;content:"/ "}.breadcrumb>.active{color:#818a91}.pagination{padding-left:0;margin-top:1rem}.pagination>li{display:inline}.pagination>li>a,.pagination>li>span{position:relative;float:left;padding:.5rem .75rem;margin-left:-1px;line-height:1.5;color:#0275d8;text-decoration:none;background-color:#fff;border:1px solid #ddd}.pagination>li:first-child>a,.pagination>li:first-child>span{margin-left:0;border-top-left-radius:.25rem;border-bottom-left-radius:.25rem}.pagination>li:last-child>a,.pagination>li:last-child>span{border-top-right-radius:.25rem;border-bottom-right-radius:.25rem}.pagination>li>a:focus,.pagination>li>a:hover,.pagination>li>span:focus,.pagination>li>span:hover{color:#014c8c;background-color:#eceeef;border-color:#ddd}.pagination>.active>a,.pagination>.active>a:focus,.pagination>.active>a:hover,.pagination>.active>span,.pagination>.active>span:focus,.pagination>.active>span:hover{z-index:2;color:#fff;cursor:default;background-color:#0275d8;border-color:#0275d8}.pagination>.disabled>a,.pagination>.disabled>a:focus,.pagination>.disabled>a:hover,.pagination>.disabled>span,.pagination>.disabled>span:focus,.pagination>.disabled>span:hover{color:#818a91;cursor:not-allowed;background-color:#fff;border-color:#ddd}.pagination-lg>li>a,.pagination-lg>li>span{padding:.75rem 1.5rem;font-size:1.25rem;line-height:1.333333}.pagination-lg>li:first-child>a,.pagination-lg>li:first-child>span{border-top-left-radius:.3rem;border-bottom-left-radius:.3rem}.pagination-lg>li:last-child>a,.pagination-lg>li:last-child>span{border-top-right-radius:.3rem;border-bottom-right-radius:.3rem}.pagination-sm>li>a,.pagination-sm>li>span{padding:.275rem .75rem;font-size:.85rem;line-height:1.5}.pagination-sm>li:first-child>a,.pagination-sm>li:first-child>span{border-top-left-radius:.2rem;border-bottom-left-radius:.2rem}.pagination-sm>li:last-child>a,.pagination-sm>li:last-child>span{border-top-right-radius:.2rem;border-bottom-right-radius:.2rem}.pager{padding-left:0;margin-top:1rem;margin-bottom:1rem;list-style:none}.pager:after,.pager:before{display:table;content:" "}.pager li{display:inline}.pager li>a,.pager li>span{display:inline-block;padding:5px 14px;background-color:#fff;border:1px solid #ddd;border-radius:15px}.pager li>a:focus,.pager li>a:hover{text-decoration:none;background-color:#eceeef}.pager .disabled>a,.pager .disabled>a:focus,.pager .disabled>a:hover,.pager .disabled>span{color:#818a91;cursor:not-allowed;background-color:#fff}.pager-next>a,.pager-next>span{float:right}.pager-prev>a,.pager-prev>span{float:left}.label{display:inline-block;padding:.25em .4em;font-size:75%;line-height:1;color:#fff;white-space:nowrap;vertical-align:baseline;border-radius:.25rem}.label:empty{display:none}.btn .label{position:relative;top:-1px}a.label:focus,a.label:hover{color:#fff;text-decoration:none;cursor:pointer}.label-pill{padding-right:.6em;padding-left:.6em;border-radius:1rem}.label-default{background-color:#818a91}.label-default[href]:focus,.label-default[href]:hover{background-color:#687077}.label-primary{background-color:#0275d8}.label-primary[href]:focus,.label-primary[href]:hover{background-color:#025aa5}.label-success{background-color:#5cb85c}.label-success[href]:focus,.label-success[href]:hover{background-color:#449d44}.label-info{background-color:#5bc0de}.label-info[href]:focus,.label-info[href]:hover{background-color:#31b0d5}.label-warning{background-color:#f0ad4e}.label-warning[href]:focus,.label-warning[href]:hover{background-color:#ec971f}.label-danger{background-color:#d9534f}.label-danger[href]:focus,.label-danger[href]:hover{background-color:#c9302c}.jumbotron{padding:2rem 1rem;margin-bottom:2rem;background-color:#eceeef;border-radius:.3rem}.jumbotron-hr{border-top-color:#d0d5d8}@media (min-width:34em){.jumbotron{padding:4rem 2rem}}.jumbotron-fluid{padding-right:0;padding-left:0;border-radius:0}.alert{padding:15px;margin-bottom:1rem;border:1px solid transparent;border-radius:.25rem}.alert>p,.alert>ul{margin-bottom:0}.alert>p+p{margin-top:5px}.alert-heading{margin-top:0;color:inherit}.alert-dismissible{padding-right:35px}.alert-dismissible .close{position:relative;top:-2px;right:-21px;color:inherit}.alert-success{color:#3c763d;background-color:#dff0d8;border-color:#d0e9c6}.alert-success hr{border-top-color:#c1e2b3}.alert-success .alert-link{color:#2b542c}.alert-info{color:#31708f;background-color:#d9edf7;border-color:#bcdff1}.alert-info hr{border-top-color:#a6d5ec}.alert-info .alert-link{color:#245269}.alert-warning{color:#8a6d3b;background-color:#fcf8e3;border-color:#faf2cc}.alert-warning hr{border-top-color:#f7ecb5}.alert-warning .alert-link{color:#66512c}.alert-danger{color:#a94442;background-color:#f2dede;border-color:#ebcccc}.alert-danger hr{border-top-color:#e4b9b9}.alert-danger .alert-link{color:#843534}@-webkit-keyframes progress-bar-stripes{from{background-position:1rem 0}to{background-position:0 0}}@-o-keyframes progress-bar-stripes{from{background-position:1rem 0}to{background-position:0 0}}@keyframes progress-bar-stripes{from{background-position:1rem 0}to{background-position:0 0}}.progress{display:block;height:1rem;margin-bottom:1rem}.progress[value]{-webkit-appearance:none;color:#0074d9;border:0;-moz-appearance:none;appearance:none}.progress[value]::-webkit-progress-bar{background-color:#eee;border-radius:.25rem}.progress[value]::-webkit-progress-value::before{content:attr(value)}.progress[value]::-webkit-progress-value{background-color:#0074d9;border-top-left-radius:.25rem;border-bottom-left-radius:.25rem}.progress[value="100"]::-webkit-progress-value{border-top-right-radius:.25rem;border-bottom-right-radius:.25rem}@media screen and (min-width:0 \0){.progress{background-color:#eee;border-radius:.25rem}.progress-bar{display:inline-block;height:1rem;text-indent:-999rem;background-color:#0074d9;border-top-left-radius:.25rem;border-bottom-left-radius:.25rem}.progress[width^="0"]{min-width:2rem;color:#818a91;background-color:transparent;background-image:none}.progress[width="100%"]{border-top-right-radius:.25rem;border-bottom-right-radius:.25rem}}.progress-striped[value]::-webkit-progress-value{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);-webkit-background-size:1rem 1rem;background-size:1rem 1rem}.progress-striped[value]::-moz-progress-bar{background-image:linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-size:1rem 1rem}.progress-animated[value]::-webkit-progress-value{-webkit-animation:progress-bar-stripes 2s linear infinite;animation:progress-bar-stripes 2s linear infinite}.progress-animated[value]::-moz-progress-bar{animation:progress-bar-stripes 2s linear infinite}.progress-success[value]::-webkit-progress-value{background-color:#5cb85c}.progress-success[value]::-moz-progress-bar{background-color:#5cb85c}@media screen and (min-width:0 \0){.progress-bar-striped{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15) 25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15) 75%,transparent 75%,transparent);-webkit-background-size:1rem 1rem;background-size:1rem 1rem}.progress-animated .progress-bar-striped{-webkit-animation:progress-bar-stripes 2s linear infinite;-o-animation:progress-bar-stripes 2s linear infinite;animation:progress-bar-stripes 2s linear infinite}.progress-success .progress-bar{background-color:#5cb85c}}.progress-info[value]::-webkit-progress-value{background-color:#5bc0de}.progress-info[value]::-moz-progress-bar{background-color:#5bc0de}@media screen and (min-width:0 \0){.progress-info .progress-bar{background-color:#5bc0de}}.progress-warning[value]::-webkit-progress-value{background-color:#f0ad4e}.progress-warning[value]::-moz-progress-bar{background-color:#f0ad4e}@media screen and (min-width:0 \0){.progress-warning .progress-bar{background-color:#f0ad4e}}.progress-danger[value]::-webkit-progress-value{background-color:#d9534f}.progress-danger[value]::-moz-progress-bar{background-color:#d9534f}@media screen and (min-width:0 \0){.progress-danger .progress-bar{background-color:#d9534f}}.media{margin-top:15px}.media:first-child{margin-top:0}.media,.media-body{overflow:hidden;zoom:1}.media-body{width:10000px}.media-body,.media-left,.media-right{display:table-cell;vertical-align:top}.media-middle{vertical-align:middle}.media-bottom{vertical-align:bottom}.media-object{display:block}.media-object.img-thumbnail{max-width:none}.media-right{padding-left:10px}.media-left{padding-right:10px}.media-heading{margin-top:0;margin-bottom:5px}.media-list{padding-left:0;list-style:none}.list-group{padding-left:0;margin-bottom:0}.list-group-item{position:relative;display:block;padding:.75rem 1.25rem;margin-bottom:-.0625rem;background-color:#fff;border:.0625rem solid #ddd}.list-group-item:first-child{border-top-left-radius:.25rem;border-top-right-radius:.25rem}.list-group-item:last-child{margin-bottom:0;border-bottom-right-radius:.25rem;border-bottom-left-radius:.25rem}.list-group-flush .list-group-item{border-width:.0625rem 0;border-radius:0}a.list-group-item,button.list-group-item{width:100%;color:#555;text-align:inherit}a.list-group-item .list-group-item-heading,button.list-group-item .list-group-item-heading{color:#333}a.list-group-item:focus,a.list-group-item:hover,button.list-group-item:focus,button.list-group-item:hover{color:#555;text-decoration:none;background-color:#f5f5f5}.list-group-item.disabled,.list-group-item.disabled:focus,.list-group-item.disabled:hover{color:#818a91;cursor:not-allowed;background-color:#eceeef}.list-group-item.disabled .list-group-item-heading,.list-group-item.disabled:focus .list-group-item-heading,.list-group-item.disabled:hover .list-group-item-heading{color:inherit}.list-group-item.disabled .list-group-item-text,.list-group-item.disabled:focus .list-group-item-text,.list-group-item.disabled:hover .list-group-item-text{color:#818a91}.list-group-item.active,.list-group-item.active:focus,.list-group-item.active:hover{z-index:2;color:#fff;background-color:#0275d8;border-color:#0275d8}.list-group-item.active .list-group-item-heading,.list-group-item.active .list-group-item-heading>.small,.list-group-item.active .list-group-item-heading>small,.list-group-item.active:focus .list-group-item-heading,.list-group-item.active:focus .list-group-item-heading>.small,.list-group-item.active:focus .list-group-item-heading>small,.list-group-item.active:hover .list-group-item-heading,.list-group-item.active:hover .list-group-item-heading>.small,.list-group-item.active:hover .list-group-item-heading>small{color:inherit}.list-group-item.active .list-group-item-text,.list-group-item.active:focus .list-group-item-text,.list-group-item.active:hover .list-group-item-text{color:#a8d6fe}.list-group-item-state{color:#a94442;background-color:#f2dede}a.list-group-item-state,button.list-group-item-state{color:#a94442}a.list-group-item-state .list-group-item-heading,button.list-group-item-state .list-group-item-heading{color:inherit}a.list-group-item-state:focus,a.list-group-item-state:hover,button.list-group-item-state:focus,button.list-group-item-state:hover{color:#a94442;background-color:#ebcccc}a.list-group-item-state.active,a.list-group-item-state.active:focus,a.list-group-item-state.active:hover,button.list-group-item-state.active,button.list-group-item-state.active:focus,button.list-group-item-state.active:hover{color:#fff;background-color:#a94442;border-color:#a94442}.list-group-item-heading{margin-top:0;margin-bottom:5px}.list-group-item-text{margin-bottom:0;line-height:1.3}.embed-responsive{position:relative;display:block;height:0;padding:0}.embed-responsive .embed-responsive-item,.embed-responsive embed,.embed-responsive iframe,.embed-responsive object,.embed-responsive video{position:absolute;top:0;bottom:0;left:0;width:100%;height:100%;border:0}.embed-responsive-21by9{padding-bottom:42.857143%}.embed-responsive-16by9{padding-bottom:56.25%}.embed-responsive-4by3{padding-bottom:75%}.close{float:right;font-size:1.5rem;line-height:1;color:#000;text-shadow:0 1px 0 #fff;opacity:.2}.close:focus,.close:hover{color:#000;text-decoration:none;cursor:pointer;opacity:.5}button.close{-webkit-appearance:none;padding:0;cursor:pointer;background:0 0;border:0}.modal-content,.popover{-webkit-background-clip:padding-box}.modal{position:fixed;top:0;right:0;bottom:0;left:0;z-index:1050;display:none;-webkit-overflow-scrolling:touch;outline:0}.modal-footer:after,.modal-footer:before,.modal-header:after,.modal-header:before{display:table;content:" "}.modal.fade .modal-dialog{-webkit-transition:-webkit-transform .3s ease-out;-o-transition:-o-transform .3s ease-out;transition:transform .3s ease-out;-webkit-transform:translate(0,-25%);-ms-transform:translate(0,-25%);-o-transform:translate(0,-25%);transform:translate(0,-25%)}.modal.in .modal-dialog{-webkit-transform:translate(0,0);-ms-transform:translate(0,0);-o-transform:translate(0,0);transform:translate(0,0)}.modal-open .modal{overflow-x:hidden;overflow-y:auto}.modal-dialog{position:relative;width:auto;margin:10px}.modal-content{position:relative;background-color:#fff;background-clip:padding-box;border:1px solid rgba(0,0,0,.2);border-radius:.3rem;outline:0}.modal-backdrop{position:fixed;top:0;right:0;bottom:0;left:0;z-index:1040;background-color:#000}.modal-backdrop.fade{opacity:0}.modal-backdrop.in{opacity:.5}.modal-header{padding:15px;border-bottom:1px solid #e5e5e5}.modal-header .close{margin-top:-2px}.modal-title{margin:0;line-height:1.5}.modal-body{position:relative;padding:15px}.modal-footer{padding:15px;text-align:right;border-top:1px solid #e5e5e5}.popover,.tooltip{font-family:"Helvetica Neue",Helvetica,Arial,sans-serif;font-size:.85rem;font-style:normal;font-weight:400;line-height:1.5;text-align:left;text-shadow:none;text-transform:none;letter-spacing:normal;word-break:normal;word-spacing:normal;word-wrap:normal;white-space:normal;line-break:auto;text-decoration:none}.popover,.tooltip{position:absolute;display:block}.modal-footer .btn+.btn{margin-bottom:0;margin-left:5px}.modal-footer .btn-group .btn+.btn{margin-left:-1px}.modal-footer .btn-block+.btn-block{margin-left:0}.modal-scrollbar-measure{position:absolute;top:-9999px;width:50px;height:50px;overflow:scroll}@media (min-width:34em){.modal-dialog{width:600px;margin:30px auto}.modal-sm{width:300px}}@media (min-width:48em){.modal-lg{width:900px}}.tooltip{z-index:1070;text-align:start;opacity:0}.tooltip.in{opacity:.9}.tooltip.bs-tether-element-attached-bottom,.tooltip.tooltip-top{padding:5px 0;margin-top:-3px}.tooltip.bs-tether-element-attached-bottom .tooltip-arrow,.tooltip.tooltip-top .tooltip-arrow{bottom:0;left:50%;margin-left:-5px;border-width:5px 5px 0;border-top-color:#000}.tooltip.bs-tether-element-attached-left,.tooltip.tooltip-right{padding:0 5px;margin-left:3px}.tooltip.bs-tether-element-attached-left .tooltip-arrow,.tooltip.tooltip-right .tooltip-arrow{top:50%;left:0;margin-top:-5px;border-width:5px 5px 5px 0;border-right-color:#000}.tooltip.bs-tether-element-attached-top,.tooltip.tooltip-bottom{padding:5px 0;margin-top:3px}.tooltip.bs-tether-element-attached-top .tooltip-arrow,.tooltip.tooltip-bottom .tooltip-arrow{top:0;left:50%;margin-left:-5px;border-width:0 5px 5px;border-bottom-color:#000}.tooltip.bs-tether-element-attached-right,.tooltip.tooltip-left{padding:0 5px;margin-left:-3px}.tooltip.bs-tether-element-attached-right .tooltip-arrow,.tooltip.tooltip-left .tooltip-arrow{top:50%;right:0;margin-top:-5px;border-width:5px 0 5px 5px;border-left-color:#000}.tooltip-inner{max-width:200px;padding:3px 8px;color:#fff;text-align:center;background-color:#000;border-radius:.25rem}.tooltip-arrow{position:absolute;width:0;height:0;border-color:transparent;border-style:solid}.popover{top:0;left:0;z-index:1060;max-width:276px;padding:1px;text-align:start;background-color:#fff;background-clip:padding-box;border:1px solid rgba(0,0,0,.2);border-radius:.3rem}.carousel-caption,.carousel-control{color:#fff;text-align:center;text-shadow:0 1px 2px rgba(0,0,0,.6)}.text-nowrap,.text-truncate{white-space:nowrap}.popover.bs-tether-element-attached-bottom,.popover.popover-top{margin-top:-10px}.popover.bs-tether-element-attached-bottom .popover-arrow,.popover.popover-top .popover-arrow{bottom:-11px;left:50%;margin-left:-11px;border-top-color:rgba(0,0,0,.25);border-bottom-width:0}.popover.bs-tether-element-attached-bottom .popover-arrow:after,.popover.popover-top .popover-arrow:after{bottom:1px;margin-left:-10px;content:"";border-top-color:#fff;border-bottom-width:0}.popover.bs-tether-element-attached-left,.popover.popover-right{margin-left:10px}.popover.bs-tether-element-attached-left .popover-arrow,.popover.popover-right .popover-arrow{top:50%;left:-11px;margin-top:-11px;border-right-color:rgba(0,0,0,.25);border-left-width:0}.popover.bs-tether-element-attached-left .popover-arrow:after,.popover.popover-right .popover-arrow:after{bottom:-10px;left:1px;content:"";border-right-color:#fff;border-left-width:0}.popover.bs-tether-element-attached-top,.popover.popover-bottom{margin-top:10px}.popover.bs-tether-element-attached-top .popover-arrow,.popover.popover-bottom .popover-arrow{top:-11px;left:50%;margin-left:-11px;border-top-width:0;border-bottom-color:rgba(0,0,0,.25)}.popover.bs-tether-element-attached-top .popover-arrow:after,.popover.popover-bottom .popover-arrow:after{top:1px;margin-left:-10px;content:"";border-top-width:0;border-bottom-color:#fff}.popover.bs-tether-element-attached-right,.popover.popover-left{margin-left:-10px}.popover.bs-tether-element-attached-right .popover-arrow,.popover.popover-left .popover-arrow{top:50%;right:-11px;margin-top:-11px;border-right-width:0;border-left-color:rgba(0,0,0,.25)}.popover.bs-tether-element-attached-right .popover-arrow:after,.popover.popover-left .popover-arrow:after{right:1px;bottom:-10px;content:"";border-right-width:0;border-left-color:#fff}.popover-title{padding:8px 14px;margin:0;font-size:1rem;background-color:#f7f7f7;border-bottom:1px solid #ebebeb;border-radius:-.7rem -.7rem 0 0}.popover-content{padding:9px 14px}.popover-arrow,.popover-arrow:after{position:absolute;display:block;width:0;height:0;border-color:transparent;border-style:solid}.carousel,.carousel-inner{position:relative}.popover-arrow{border-width:11px}.popover-arrow:after{content:"";border-width:10px}.carousel-inner{width:100%;overflow:hidden}.carousel-inner>.carousel-item{position:relative;display:none;-webkit-transition:.6s ease-in-out left;-o-transition:.6s ease-in-out left;transition:.6s ease-in-out left}.carousel-inner>.carousel-item>a>img,.carousel-inner>.carousel-item>img{line-height:1}@media all and (transform-3d),(-webkit-transform-3d){.carousel-inner>.carousel-item{-webkit-transition:-webkit-transform .6s ease-in-out;-o-transition:-o-transform .6s ease-in-out;transition:transform .6s ease-in-out;-webkit-backface-visibility:hidden;backface-visibility:hidden;-webkit-perspective:1000px;perspective:1000px}.carousel-inner>.carousel-item.active.right,.carousel-inner>.carousel-item.next{left:0;-webkit-transform:translate3d(100%,0,0);transform:translate3d(100%,0,0)}.carousel-inner>.carousel-item.active.left,.carousel-inner>.carousel-item.prev{left:0;-webkit-transform:translate3d(-100%,0,0);transform:translate3d(-100%,0,0)}.carousel-inner>.carousel-item.active,.carousel-inner>.carousel-item.next.left,.carousel-inner>.carousel-item.prev.right{left:0;-webkit-transform:translate3d(0,0,0);transform:translate3d(0,0,0)}}.carousel-inner>.active,.carousel-inner>.next,.carousel-inner>.prev{display:block}.carousel-inner>.active{left:0}.carousel-inner>.next,.carousel-inner>.prev{position:absolute;top:0;width:100%}.carousel-inner>.next{left:100%}.carousel-inner>.prev{left:-100%}.carousel-inner>.next.left,.carousel-inner>.prev.right{left:0}.carousel-inner>.active.left{left:-100%}.carousel-inner>.active.right{left:100%}.carousel-control{position:absolute;top:0;bottom:0;left:0;width:15%;font-size:20px;opacity:.5}.carousel-control.left{background-image:-webkit-gradient(linear,left top,right top,from(rgba(0,0,0,.5)),to(rgba(0,0,0,.0001)));background-image:-webkit-linear-gradient(left,rgba(0,0,0,.5) 0,rgba(0,0,0,.0001) 100%);background-image:-o-linear-gradient(left,rgba(0,0,0,.5) 0,rgba(0,0,0,.0001) 100%);background-image:linear-gradient(to right,rgba(0,0,0,.5) 0,rgba(0,0,0,.0001) 100%);filter:progid:DXImageTransform.Microsoft.gradient(startColorstr='#80000000', endColorstr='#00000000', GradientType=1);background-repeat:repeat-x}.carousel-control.right{right:0;left:auto;background-image:-webkit-gradient(linear,left top,right top,from(rgba(0,0,0,.0001)),to(rgba(0,0,0,.5)));background-image:-webkit-linear-gradient(left,rgba(0,0,0,.0001) 0,rgba(0,0,0,.5) 100%);background-image:-o-linear-gradient(left,rgba(0,0,0,.0001) 0,rgba(0,0,0,.5) 100%);background-image:linear-gradient(to right,rgba(0,0,0,.0001) 0,rgba(0,0,0,.5) 100%);filter:progid:DXImageTransform.Microsoft.gradient(startColorstr='#00000000', endColorstr='#80000000', GradientType=1);background-repeat:repeat-x}.carousel-control:focus,.carousel-control:hover{color:#fff;text-decoration:none;outline:0;opacity:.9}.carousel-control .icon-next,.carousel-control .icon-prev{position:absolute;top:50%;z-index:5;display:inline-block;width:20px;height:20px;margin-top:-10px;font-family:serif;line-height:1}.carousel-control .icon-prev{left:50%;margin-left:-10px}.carousel-control .icon-next{right:50%;margin-right:-10px}.carousel-control .icon-prev:before{content:"\2039"}.carousel-control .icon-next:before{content:"\203a"}.carousel-indicators{position:absolute;bottom:10px;left:50%;z-index:15;width:60%;padding-left:0;margin-left:-30%;text-align:center;list-style:none}.carousel-indicators li{display:inline-block;width:10px;height:10px;margin:1px;text-indent:-999px;cursor:pointer;background-color:transparent;border:1px solid #fff;border-radius:10px}.carousel-indicators .active{width:12px;height:12px;margin:0;background-color:#fff}.carousel-caption{position:absolute;right:15%;bottom:20px;left:15%;z-index:10;padding-top:20px;padding-bottom:20px}.carousel-caption .btn,.text-hide{text-shadow:none}@media (min-width:34em){.carousel-control .icon-next,.carousel-control .icon-prev{width:30px;height:30px;margin-top:-15px;font-size:30px}.carousel-control .icon-prev{margin-left:-15px}.carousel-control .icon-next{margin-right:-15px}.carousel-caption{right:20%;left:20%;padding-bottom:30px}.carousel-indicators{bottom:20px}}.clearfix:after,.clearfix:before{display:table;content:" "}.center-block{display:block;margin-right:auto;margin-left:auto}.hidden-xl-down,.hidden-xs-up,.visible-print-block,[hidden]{display:none!important}.pull-right{float:right!important}.pull-left{float:left!important}.sr-only{position:absolute;width:1px;height:1px;padding:0;margin:-1px;overflow:hidden;clip:rect(0,0,0,0);border:0}.p-r-0,.p-x-0{padding-right:0!important}.p-l-0,.p-x-0{padding-left:0!important}.p-t-0,.p-y-0{padding-top:0!important}.p-b-0,.p-y-0{padding-bottom:0!important}.sr-only-focusable:active,.sr-only-focusable:focus{position:static;width:auto;height:auto;margin:0;overflow:visible;clip:auto}.m-r-0,.m-x-0{margin-right:0!important}.m-l-0,.m-x-0{margin-left:0!important}.m-t-0,.m-y-0{margin-top:0!important}.m-b-0,.m-y-0{margin-bottom:0!important}.invisible{visibility:hidden}.text-hide{font:"0/0" a;color:transparent;background-color:transparent;border:0}.text-left{text-align:left}.text-right{text-align:right}.text-center{text-align:center}.text-justify{text-align:justify}.text-truncate{overflow:hidden;text-overflow:ellipsis}.text-xs-left{text-align:left}.text-xs-right{text-align:right}.text-xs-center{text-align:center}.text-lowercase{text-transform:lowercase}.text-uppercase{text-transform:uppercase}.text-capitalize{text-transform:capitalize}.text-muted{color:#818a91}.text-primary{color:#0275d8}a.text-primary:focus,a.text-primary:hover{color:#025aa5}.text-success{color:#5cb85c}a.text-success:focus,a.text-success:hover{color:#449d44}.text-info{color:#5bc0de}a.text-info:focus,a.text-info:hover{color:#31b0d5}.text-warning{color:#f0ad4e}a.text-warning:focus,a.text-warning:hover{color:#ec971f}.text-danger{color:#d9534f}a.text-danger:focus,a.text-danger:hover{color:#c9302c}.bg-inverse{color:#eceeef;background-color:#373a3c}.bg-faded{background-color:#f7f7f9}.bg-primary{color:#fff;background-color:#0275d8}a.bg-primary:focus,a.bg-primary:hover{background-color:#025aa5}.bg-success{color:#fff;background-color:#5cb85c}a.bg-success:focus,a.bg-success:hover{background-color:#449d44}.bg-info{color:#fff;background-color:#5bc0de}a.bg-info:focus,a.bg-info:hover{background-color:#31b0d5}.bg-warning{color:#fff;background-color:#f0ad4e}a.bg-warning:focus,a.bg-warning:hover{background-color:#ec971f}.bg-danger{color:#fff;background-color:#d9534f}a.bg-danger:focus,a.bg-danger:hover{background-color:#c9302c}.m-a-0{margin:0!important}.m-r,.m-x{margin-right:1rem!important}.m-l,.m-x{margin-left:1rem!important}.m-t,.m-y{margin-top:1rem!important}.m-b,.m-y{margin-bottom:1rem!important}.m-a{margin:1rem!important}.m-t-md,.m-y-md{margin-top:1.5rem!important}.m-b-md,.m-y-md{margin-bottom:1.5rem!important}.m-x-auto{margin-right:auto!important;margin-left:auto!important}.m-r-md,.m-x-md{margin-right:1.5rem!important}.m-l-md,.m-x-md{margin-left:1.5rem!important}.m-a-md{margin:1.5rem!important}.m-r-lg,.m-x-lg{margin-right:3rem!important}.m-l-lg,.m-x-lg{margin-left:3rem!important}.m-t-lg,.m-y-lg{margin-top:3rem!important}.m-b-lg,.m-y-lg{margin-bottom:3rem!important}.m-a-lg{margin:3rem!important}.p-a-0{padding:0!important}.p-r,.p-x{padding-right:1rem!important}.p-l,.p-x{padding-left:1rem!important}.p-t,.p-y{padding-top:1rem!important}.p-b,.p-y{padding-bottom:1rem!important}.p-a{padding:1rem!important}.p-r-md,.p-x-md{padding-right:1.5rem!important}.p-l-md,.p-x-md{padding-left:1.5rem!important}.p-t-md,.p-y-md{padding-top:1.5rem!important}.p-b-md,.p-y-md{padding-bottom:1.5rem!important}.p-a-md{padding:1.5rem!important}.p-r-lg,.p-x-lg{padding-right:3rem!important}.p-l-lg,.p-x-lg{padding-left:3rem!important}.p-t-lg,.p-y-lg{padding-top:3rem!important}.p-b-lg,.p-y-lg{padding-bottom:3rem!important}.p-a-lg{padding:3rem!important}.pos-f-t{position:fixed;top:0;right:0;left:0;z-index:1030}@media (max-width:33.9em){.hidden-xs-down{display:none!important}}@media (min-width:34em){.text-sm-left{text-align:left}.text-sm-right{text-align:right}.text-sm-center{text-align:center}.hidden-sm-up{display:none!important}}@media (max-width:47.9em){.hidden-sm-down{display:none!important}}@media (min-width:48em){.text-md-left{text-align:left}.text-md-right{text-align:right}.text-md-center{text-align:center}.hidden-md-up{display:none!important}}@media (max-width:61.9em){.hidden-md-down{display:none!important}}@media (min-width:62em){.text-lg-left{text-align:left}.text-lg-right{text-align:right}.text-lg-center{text-align:center}.hidden-lg-up{display:none!important}}@media (max-width:74.9em){.hidden-lg-down{display:none!important}}@media (min-width:75em){.text-xl-left{text-align:left}.text-xl-right{text-align:right}.text-xl-center{text-align:center}.hidden-xl-up{display:none!important}}@media print{.visible-print-block{display:block!important}}.visible-print-inline{display:none!important}@media print{.visible-print-inline{display:inline!important}}.visible-print-inline-block{display:none!important}@media print{.visible-print-inline-block{display:inline-block!important}.hidden-print .hidden-print{display:none!important}}  
 6:/\*# sourceMappingURL=bootstrap.min.css.map \*/

## .\mobile\www\css\index.css

1:/\*  
 2: \* Licensed to the Apache Software Foundation (ASF) under one  
 3: \* or more contributor license agreements. See the NOTICE file  
 4: \* distributed with this work for additional information  
 5: \* regarding copyright ownership. The ASF licenses this file  
 6: \* to you under the Apache License, Version 2.0 (the  
 7: \* "License"); you may not use this file except in compliance  
 8: \* with the License. You may obtain a copy of the License at  
 9: \*  
 10: \* http://www.apache.org/licenses/LICENSE-2.0  
 11: \*  
 12: \* Unless required by applicable law or agreed to in writing,  
 13: \* software distributed under the License is distributed on an  
 14: \* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY  
 15: \* KIND, either express or implied. See the License for the  
 16: \* specific language governing permissions and limitations  
 17: \* under the License.  
 18: \*/  
 19:\* {  
 20: -webkit-tap-highlight-color: rgba(0,0,0,0); /\* make transparent link selection, adjust last value opacity 0 to 1.0 \*/  
 21:}  
 22:  
 23:body {  
 24: -webkit-touch-callout: none; /\* prevent callout to copy image, etc when tap to hold \*/  
 25: -webkit-text-size-adjust: none; /\* prevent webkit from resizing text to fit \*/  
 26: -webkit-user-select: none; /\* prevent copy paste, to allow, change 'none' to 'text' \*/  
 27: background-color:#E4E4E4;  
 28: background-image:linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);  
 29: background-image:-webkit-linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);  
 30: background-image:-ms-linear-gradient(top, #A7A7A7 0%, #E4E4E4 51%);  
 31: background-image:-webkit-gradient(  
 32: linear,  
 33: left top,  
 34: left bottom,  
 35: color-stop(0, #A7A7A7),  
 36: color-stop(0.51, #E4E4E4)  
 37: );  
 38: background-attachment:fixed;  
 39: font-family:'HelveticaNeue-Light', 'HelveticaNeue', Helvetica, Arial, sans-serif;  
 40: font-size:12px;  
 41: height:100%;  
 42: margin:0px;  
 43: padding:0px;  
 44: text-transform:uppercase;  
 45: width:100%;  
 46:}  
 47:  
 48:/\* Portrait layout (default) \*/  
 49:.app {  
 50: background:url(../img/logo.png) no-repeat center top; /\* 170px x 200px \*/  
 51: position:absolute; /\* position in the center of the screen \*/  
 52: left:50%;  
 53: top:50%;  
 54: height:50px; /\* text area height \*/  
 55: width:225px; /\* text area width \*/  
 56: text-align:center;  
 57: padding:180px 0px 0px 0px; /\* image height is 200px (bottom 20px are overlapped with text) \*/  
 58: margin:-115px 0px 0px -112px; /\* offset vertical: half of image height and text area height \*/  
 59: /\* offset horizontal: half of text area width \*/  
 60:}  
 61:  
 62:/\* Landscape layout (with min-width) \*/  
 63:@media screen and (min-aspect-ratio: 1/1) and (min-width:400px) {  
 64: .app {  
 65: background-position:left center;  
 66: padding:75px 0px 75px 170px; /\* padding-top + padding-bottom + text area = image height \*/  
 67: margin:-90px 0px 0px -198px; /\* offset vertical: half of image height \*/  
 68: /\* offset horizontal: half of image width and text area width \*/  
 69: }  
 70:}  
 71:  
 72:h1 {  
 73: font-size:24px;  
 74: font-weight:normal;  
 75: margin:0px;  
 76: overflow:visible;  
 77: padding:0px;  
 78: text-align:center;  
 79:}  
 80:  
 81:.event {  
 82: border-radius:4px;  
 83: -webkit-border-radius:4px;  
 84: color:#FFFFFF;  
 85: font-size:12px;  
 86: margin:0px 30px;  
 87: padding:2px 0px;  
 88:}  
 89:  
 90:.event.listening {  
 91: background-color:#333333;  
 92: display:block;  
 93:}  
 94:  
 95:.event.received {  
 96: background-color:#4B946A;  
 97: display:none;  
 98:}  
 99:  
 100:@keyframes fade {  
 101: from { opacity: 1.0; }  
 102: 50% { opacity: 0.4; }  
 103: to { opacity: 1.0; }  
 104:}  
 105:   
 106:@-webkit-keyframes fade {  
 107: from { opacity: 1.0; }  
 108: 50% { opacity: 0.4; }  
 109: to { opacity: 1.0; }  
 110:}  
 111:   
 112:.blink {  
 113: animation:fade 3000ms infinite;  
 114: -webkit-animation:fade 3000ms infinite;  
 115:}

## .\mobile\www\css\jquery-confirm.css

1:/\*!  
 2: \* jquery-confirm v2.5.1 (http://craftpip.github.io/jquery-confirm/)  
 3: \* Author: boniface pereira  
 4: \* Website: www.craftpip.com  
 5: \* Contact: hey@craftpip.com  
 6: \*  
 7: \* Copyright 2013-2016 jquery-confirm  
 8: \* Licensed under MIT (https://github.com/craftpip/jquery-confirm/blob/master/LICENSE)  
 9: \*/  
 10:@-webkit-keyframes jconfirm-rotate {  
 11: from {  
 12: -webkit-transform: rotate(0deg);  
 13: transform: rotate(0deg);  
 14: }  
 15: to {  
 16: -webkit-transform: rotate(360deg);  
 17: transform: rotate(360deg);  
 18: }  
 19:}  
 20:@keyframes jconfirm-rotate {  
 21: from {  
 22: -webkit-transform: rotate(0deg);  
 23: transform: rotate(0deg);  
 24: }  
 25: to {  
 26: -webkit-transform: rotate(360deg);  
 27: transform: rotate(360deg);  
 28: }  
 29:}  
 30:body.jconfirm-noscroll {  
 31: overflow: hidden !important;  
 32:}  
 33:.jconfirm {  
 34: position: fixed;  
 35: top: 0;  
 36: left: 0;  
 37: right: 0;  
 38: bottom: 0;  
 39: z-index: 99999999;  
 40: font-family: inherit;  
 41: overflow: hidden;  
 42:}  
 43:.jconfirm .jconfirm-bg {  
 44: position: fixed;  
 45: top: 0;  
 46: left: 0;  
 47: right: 0;  
 48: bottom: 0;  
 49: opacity: 0;  
 50: -webkit-transition: all .4s;  
 51: transition: all .4s;  
 52:}  
 53:.jconfirm .jconfirm-bg.seen {  
 54: opacity: 1;  
 55:}  
 56:.jconfirm .jconfirm-scrollpane {  
 57: position: fixed;  
 58: top: 0;  
 59: left: 0;  
 60: right: 0;  
 61: bottom: 0;  
 62: overflow-y: auto;  
 63: -webkit-perspective: 500px;  
 64: perspective: 500px;  
 65: -webkit-perspective-origin: center;  
 66: perspective-origin: center;  
 67:}  
 68:.jconfirm .jconfirm-box {  
 69: background: white;  
 70: border-radius: 4px;  
 71: position: relative;  
 72: outline: none;  
 73: padding: 15px 15px 0;  
 74:}  
 75:.jconfirm .jconfirm-box div.closeIcon {  
 76: height: 20px;  
 77: width: 20px;  
 78: position: absolute;  
 79: top: 5px;  
 80: right: 5px;  
 81: cursor: pointer;  
 82: opacity: .6;  
 83: text-align: center;  
 84: -webkit-transition: opacity 0.1s ease-in;  
 85: transition: opacity 0.1s ease-in;  
 86: display: none;  
 87: font-size: 27px;  
 88: line-height: 14px;  
 89:}  
 90:.jconfirm .jconfirm-box div.closeIcon .fa {  
 91: font-size: 16px;  
 92:}  
 93:.jconfirm .jconfirm-box div.closeIcon .glyphicon {  
 94: font-size: 16px;  
 95:}  
 96:.jconfirm .jconfirm-box div.closeIcon .zmdi {  
 97: font-size: 16px;  
 98:}  
 99:.jconfirm .jconfirm-box div.closeIcon:hover {  
 100: opacity: 1;  
 101:}  
 102:.jconfirm .jconfirm-box div.title-c {  
 103: display: block;  
 104: font-size: 25px;  
 105: line-height: 20px;  
 106:}  
 107:.jconfirm .jconfirm-box div.title-c .icon-c {  
 108: font-size: inherit;  
 109: padding-bottom: 15px;  
 110: display: inline-block;  
 111: margin-right: 8px;  
 112: vertical-align: middle;  
 113:}  
 114:.jconfirm .jconfirm-box div.title-c .icon-c i {  
 115: vertical-align: middle;  
 116:}  
 117:.jconfirm .jconfirm-box div.title-c .icon-c:empty {  
 118: display: none;  
 119:}  
 120:.jconfirm .jconfirm-box div.title-c .title {  
 121: font-size: inherit;  
 122: font-family: inherit;  
 123: display: inline-block;  
 124: vertical-align: middle;  
 125: padding-bottom: 15px;  
 126:}  
 127:.jconfirm .jconfirm-box div.title-c .title:empty {  
 128: display: none;  
 129:}  
 130:.jconfirm .jconfirm-box div.content-pane {  
 131: margin-bottom: 15px;  
 132: height: auto;  
 133: -webkit-transition: height 0.4s ease-in;  
 134: transition: height 0.4s ease-in;  
 135: display: inline-block;  
 136: width: 100%;  
 137: position: relative;  
 138:}  
 139:.jconfirm .jconfirm-box div.content-pane .content {  
 140: font-size: 20px;  
 141: position: absolute;  
 142: top: 10px;  
 143: left: 0;  
 144: -webkit-transition: all 0.2s ease-in;  
 145: transition: all 0.2s ease-in;  
 146: right: 0;  
 147:}  
 148:.jconfirm .jconfirm-box div.content-pane .content img {  
 149: width: 100%;  
 150: height: auto;  
 151:}  
 152:.jconfirm .jconfirm-box div.content-pane .content:empty {  
 153: display: none;  
 154:}  
 155:.jconfirm .jconfirm-box div.content-pane .content:empty.loading {  
 156: height: 40px;  
 157: position: relative;  
 158: opacity: 0.6;  
 159: display: block;  
 160:}  
 161:.jconfirm .jconfirm-box div.content-pane .content:empty.loading:before {  
 162: content: '';  
 163: height: 20px;  
 164: width: 20px;  
 165: border: solid 2px transparent;  
 166: position: absolute;  
 167: left: 50%;  
 168: margin-left: -10px;  
 169: border-radius: 50%;  
 170: -webkit-animation: jconfirm-rotate 1s infinite linear;  
 171: animation: jconfirm-rotate 1s infinite linear;  
 172: border-bottom-color: #aaa;  
 173: top: 50%;  
 174: margin-top: -10px;  
 175:}  
 176:.jconfirm .jconfirm-box div.content-pane .content:empty.loading:after {  
 177: content: '';  
 178: position: absolute;  
 179: left: 50%;  
 180: margin-left: -15px;  
 181:}  
 182:.jconfirm .jconfirm-box .buttons {  
 183: padding-bottom: 15px;  
 184:}  
 185:.jconfirm .jconfirm-box .buttons button + button {  
 186: margin-left: 5px;  
 187:}  
 188:.jconfirm .jquery-clear {  
 189: clear: both;  
 190:}  
 191:.jconfirm.rtl {  
 192: direction: rtl;  
 193:}  
 194:.jconfirm.rtl div.closeIcon {  
 195: left: 12px;  
 196: right: auto;  
 197:}  
 198:.jconfirm.jconfirm-white .jconfirm-bg {  
 199: background-color: rgba(0, 0, 0, 0.2);  
 200:}  
 201:.jconfirm.jconfirm-white .jconfirm-box {  
 202: box-shadow: 0 2px 6px rgba(0, 0, 0, 0.2);  
 203: border-radius: 5px;  
 204:}  
 205:.jconfirm.jconfirm-white .jconfirm-box .buttons {  
 206: float: right;  
 207:}  
 208:.jconfirm.jconfirm-white .jconfirm-box .buttons button {  
 209: border: none;  
 210: background-image: none;  
 211: text-transform: uppercase;  
 212: font-size: 16px;  
 213: font-weight: bold;  
 214: text-shadow: none;  
 215: -webkit-transition: background .1s;  
 216: transition: background .1s;  
 217: color: white;  
 218: padding: 20px;  
 219:}  
 220:.jconfirm.jconfirm-white .jconfirm-box .buttons button.btn-default {  
 221: box-shadow: none;  
 222: color: #333;  
 223:}  
 224:.jconfirm.jconfirm-white .jconfirm-box .buttons button.btn-default:hover {  
 225: background: #ddd;  
 226:}  
 227:.jconfirm.jconfirm-black .jconfirm-bg {  
 228: background-color: rgba(0, 0, 0, 0.5);  
 229:}  
 230:.jconfirm.jconfirm-black .jconfirm-box {  
 231: box-shadow: 0 2px 6px rgba(0, 0, 0, 0.2);  
 232: background: #444;  
 233: border-radius: 5px;  
 234: color: white;  
 235:}  
 236:.jconfirm.jconfirm-black .jconfirm-box .buttons {  
 237: float: right;  
 238:}  
 239:.jconfirm.jconfirm-black .jconfirm-box .buttons button {  
 240: border: none;  
 241: background-image: none;  
 242: text-transform: uppercase;  
 243: font-size: 16px;  
 244: font-weight: bold;  
 245: text-shadow: none;  
 246: -webkit-transition: background .1s;  
 247: transition: background .1s;  
 248: color: white;  
 249:}  
 250:.jconfirm.jconfirm-black .jconfirm-box .buttons button.btn-default {  
 251: box-shadow: none;  
 252: color: #fff;  
 253: background: none;  
 254:}  
 255:.jconfirm.jconfirm-black .jconfirm-box .buttons button.btn-default:hover {  
 256: background: #666;  
 257:}  
 258:.jconfirm .jconfirm-box.hilight {  
 259: -webkit-animation: hilight 0.82s cubic-bezier(0.36, 0.07, 0.19, 0.97) both;  
 260: animation: hilight 0.82s cubic-bezier(0.36, 0.07, 0.19, 0.97) both;  
 261: -webkit-transform: translate3d(0, 0, 0);  
 262: transform: translate3d(0, 0, 0);  
 263:}  
 264:@-webkit-keyframes hilight {  
 265: 10%,  
 266: 90% {  
 267: -webkit-transform: translate3d(-2px, 0, 0);  
 268: transform: translate3d(-2px, 0, 0);  
 269: }  
 270: 20%,  
 271: 80% {  
 272: -webkit-transform: translate3d(4px, 0, 0);  
 273: transform: translate3d(4px, 0, 0);  
 274: }  
 275: 30%,  
 276: 50%,  
 277: 70% {  
 278: -webkit-transform: translate3d(-8px, 0, 0);  
 279: transform: translate3d(-8px, 0, 0);  
 280: }  
 281: 40%,  
 282: 60% {  
 283: -webkit-transform: translate3d(8px, 0, 0);  
 284: transform: translate3d(8px, 0, 0);  
 285: }  
 286:}  
 287:@keyframes hilight {  
 288: 10%,  
 289: 90% {  
 290: -webkit-transform: translate3d(-2px, 0, 0);  
 291: transform: translate3d(-2px, 0, 0);  
 292: }  
 293: 20%,  
 294: 80% {  
 295: -webkit-transform: translate3d(4px, 0, 0);  
 296: transform: translate3d(4px, 0, 0);  
 297: }  
 298: 30%,  
 299: 50%,  
 300: 70% {  
 301: -webkit-transform: translate3d(-8px, 0, 0);  
 302: transform: translate3d(-8px, 0, 0);  
 303: }  
 304: 40%,  
 305: 60% {  
 306: -webkit-transform: translate3d(8px, 0, 0);  
 307: transform: translate3d(8px, 0, 0);  
 308: }  
 309:}  
 310:/\*Transition rules\*/  
 311:.jconfirm {  
 312: -webkit-perspective: 400px;  
 313: perspective: 400px;  
 314:}  
 315:.jconfirm .jconfirm-box {  
 316: opacity: 1;  
 317: -webkit-transition-property: -webkit-transform, opacity, box-shadow;  
 318: transition-property: transform, opacity, box-shadow;  
 319:}  
 320:.jconfirm .jconfirm-box.anim-top,  
 321:.jconfirm .jconfirm-box.anim-left,  
 322:.jconfirm .jconfirm-box.anim-right,  
 323:.jconfirm .jconfirm-box.anim-bottom,  
 324:.jconfirm .jconfirm-box.anim-opacity,  
 325:.jconfirm .jconfirm-box.anim-zoom,  
 326:.jconfirm .jconfirm-box.anim-scale,  
 327:.jconfirm .jconfirm-box.anim-none,  
 328:.jconfirm .jconfirm-box.anim-rotate,  
 329:.jconfirm .jconfirm-box.anim-rotatex,  
 330:.jconfirm .jconfirm-box.anim-rotatey,  
 331:.jconfirm .jconfirm-box.anim-scaley,  
 332:.jconfirm .jconfirm-box.anim-scalex {  
 333: opacity: 0;  
 334:}  
 335:.jconfirm .jconfirm-box.anim-rotate {  
 336: -webkit-transform: rotate(90deg);  
 337: -ms-transform: rotate(90deg);  
 338: transform: rotate(90deg);  
 339:}  
 340:.jconfirm .jconfirm-box.anim-rotatex {  
 341: -webkit-transform: rotateX(90deg);  
 342: transform: rotateX(90deg);  
 343: -webkit-transform-origin: center;  
 344: -ms-transform-origin: center;  
 345: transform-origin: center;  
 346:}  
 347:.jconfirm .jconfirm-box.anim-rotatexr {  
 348: -webkit-transform: rotateX(-90deg);  
 349: transform: rotateX(-90deg);  
 350: -webkit-transform-origin: center;  
 351: -ms-transform-origin: center;  
 352: transform-origin: center;  
 353:}  
 354:.jconfirm .jconfirm-box.anim-rotatey {  
 355: -webkit-transform: rotatey(90deg);  
 356: -ms-transform: rotatey(90deg);  
 357: transform: rotatey(90deg);  
 358: -webkit-transform-origin: center;  
 359: -ms-transform-origin: center;  
 360: transform-origin: center;  
 361:}  
 362:.jconfirm .jconfirm-box.anim-rotateyr {  
 363: -webkit-transform: rotatey(-90deg);  
 364: -ms-transform: rotatey(-90deg);  
 365: transform: rotatey(-90deg);  
 366: -webkit-transform-origin: center;  
 367: -ms-transform-origin: center;  
 368: transform-origin: center;  
 369:}  
 370:.jconfirm .jconfirm-box.anim-scaley {  
 371: -webkit-transform: scaley(1.5);  
 372: -ms-transform: scaley(1.5);  
 373: transform: scaley(1.5);  
 374: -webkit-transform-origin: center;  
 375: -ms-transform-origin: center;  
 376: transform-origin: center;  
 377:}  
 378:.jconfirm .jconfirm-box.anim-scalex {  
 379: -webkit-transform: scalex(1.5);  
 380: -ms-transform: scalex(1.5);  
 381: transform: scalex(1.5);  
 382: -webkit-transform-origin: center;  
 383: -ms-transform-origin: center;  
 384: transform-origin: center;  
 385:}  
 386:.jconfirm .jconfirm-box.anim-top {  
 387: -webkit-transform: translate(0px, -100px);  
 388: -ms-transform: translate(0px, -100px);  
 389: transform: translate(0px, -100px);  
 390:}  
 391:.jconfirm .jconfirm-box.anim-left {  
 392: -webkit-transform: translate(-100px, 0px);  
 393: -ms-transform: translate(-100px, 0px);  
 394: transform: translate(-100px, 0px);  
 395:}  
 396:.jconfirm .jconfirm-box.anim-right {  
 397: -webkit-transform: translate(100px, 0px);  
 398: -ms-transform: translate(100px, 0px);  
 399: transform: translate(100px, 0px);  
 400:}  
 401:.jconfirm .jconfirm-box.anim-bottom {  
 402: -webkit-transform: translate(0px, 100px);  
 403: -ms-transform: translate(0px, 100px);  
 404: transform: translate(0px, 100px);  
 405:}  
 406:.jconfirm .jconfirm-box.anim-zoom {  
 407: -webkit-transform: scale(1.2);  
 408: -ms-transform: scale(1.2);  
 409: transform: scale(1.2);  
 410:}  
 411:.jconfirm .jconfirm-box.anim-scale {  
 412: -webkit-transform: scale(0.5);  
 413: -ms-transform: scale(0.5);  
 414: transform: scale(0.5);  
 415:}  
 416:.jconfirm .jconfirm-box.anim-none {  
 417: display: none;  
 418:}  
 419:.jconfirm.jconfirm-supervan .jconfirm-bg {  
 420: background-color: rgba(54, 70, 93, 0.95);  
 421:}  
 422:.jconfirm.jconfirm-supervan .jconfirm-box {  
 423: background-color: transparent;  
 424:}  
 425:.jconfirm.jconfirm-supervan .jconfirm-box div.closeIcon {  
 426: color: white;  
 427:}  
 428:.jconfirm.jconfirm-supervan .jconfirm-box div.title-c {  
 429: text-align: center;  
 430: color: white;  
 431: font-size: 28px;  
 432: font-weight: normal;  
 433:}  
 434:.jconfirm.jconfirm-supervan .jconfirm-box div.title-c > \* {  
 435: padding-bottom: 25px;  
 436:}  
 437:.jconfirm.jconfirm-supervan .jconfirm-box div.content-pane {  
 438: margin-bottom: 25px;  
 439:   
 440:}  
 441:.jconfirm.jconfirm-supervan .jconfirm-box div.content {  
 442: text-align: center;  
 443: color: white;  
 444:}  
 445:.jconfirm.jconfirm-supervan .jconfirm-box .buttons {  
 446: text-align: center;  
 447:}  
 448:.jconfirm.jconfirm-supervan .jconfirm-box .buttons button {  
 449: font-size: 16px;  
 450: border-radius: 2px;  
 451: background: #303f53;  
 452: text-shadow: none;  
 453: border: none;  
 454: color: white;  
 455: padding: 10px;  
 456: min-width: 100px;  
 457:}  
 458:.jconfirm.jconfirm-material .jconfirm-bg {  
 459: background-color: rgba(0, 0, 0, 0.67);  
 460:}  
 461:.jconfirm.jconfirm-material .jconfirm-box {  
 462: background-color: white;  
 463: box-shadow: 0 7px 8px -4px rgba(0, 0, 0, 0.2), 0 13px 19px 2px rgba(0, 0, 0, 0.14), 0 5px 24px 4px rgba(0, 0, 0, 0.12);  
 464: padding: 30px 25px 10px 25px;  
 465:}  
 466:.jconfirm.jconfirm-material .jconfirm-box div.closeIcon {  
 467: color: rgba(0, 0, 0, 0.87);  
 468:}  
 469:.jconfirm.jconfirm-material .jconfirm-box div.title-c {  
 470: color: rgba(0, 0, 0, 0.87);  
 471: font-size: 22px;  
 472: font-weight: bold;  
 473:}  
 474:.jconfirm.jconfirm-material .jconfirm-box div.content {  
 475: text-align: left;  
 476: color: rgba(0, 0, 0, 0.87);  
 477:}  
 478:.jconfirm.jconfirm-material .jconfirm-box .buttons {  
 479: text-align: right;  
 480:}  
 481:.jconfirm.jconfirm-material .jconfirm-box .buttons button {  
 482: text-transform: uppercase;  
 483: font-weight: 500;  
 484:}  
 485:.jconfirm.jconfirm-bootstrap .jconfirm-bg {  
 486: background-color: rgba(0, 0, 0, 0.21);  
 487:}  
 488:.jconfirm.jconfirm-bootstrap .jconfirm-box {  
 489: background-color: white;  
 490: box-shadow: 0 3px 8px 0px rgba(0, 0, 0, 0.2);  
 491: border: solid 1px rgba(0, 0, 0, 0.4);  
 492: padding: 15px 0 0;  
 493:}  
 494:.jconfirm.jconfirm-bootstrap .jconfirm-box div.closeIcon {  
 495: color: rgba(0, 0, 0, 0.87);  
 496:}  
 497:.jconfirm.jconfirm-bootstrap .jconfirm-box div.title-c {  
 498: color: rgba(0, 0, 0, 0.87);  
 499: font-size: 22px;  
 500: font-weight: bold;  
 501: padding-left: 15px;  
 502: padding-right: 15px;  
 503:}  
 504:.jconfirm.jconfirm-bootstrap .jconfirm-box div.content {  
 505: text-align: left;  
 506: color: rgba(0, 0, 0, 0.87);  
 507: padding: 0px 15px;  
 508:}  
 509:.jconfirm.jconfirm-bootstrap .jconfirm-box .buttons {  
 510: text-align: right;  
 511: padding: 0px 0 0px;  
 512: margin: -5px 0 0px;  
 513: border-top: solid 1px #ddd;  
 514: overflow: hidden;  
 515: border-radius: 0 0 4px 4px;  
 516:}  
 517:.jconfirm.jconfirm-bootstrap .jconfirm-box .buttons button {  
 518: font-weight: 500;  
 519: border-radius: 0px;  
 520: margin: 0;  
 521: border-left: solid 1px #ddd;  
 522:}

## .\mobile\www\css\login.css

1:#page\_login {  
 2: background: #fff;  
 3: background-image: url("../img/login/ipad\_splash.png");  
 4: background-repeat: no-repeat;  
 5: height: auto;  
 6:}  
 7:  
 8:#page\_login .ui-panel-wrapper {  
 9: background-color: transparent;  
 10:}  
 11:  
 12:#page\_login section {  
 13: width: auto;  
 14:}  
 15:  
 16:#page\_login header {  
 17: background: #244a7d;  
 18: border: none;  
 19:}  
 20:  
 21:#page\_login footer {  
 22: background: #244a7d;  
 23: border: none;  
 24:}  
 25:  
 26:#page\_login .principal {  
 27: width: 100%;  
 28:}  
 29:  
 30:#page\_login /\*.ui-block-a\*/ .formulario {  
 31: margin-top: 7em;  
 32: float: right;  
 33: margin-right: 3.2em;  
 34: padding: 20px;  
 35: background-color: rgba(36, 74, 125, 0.4);  
 36:}  
 37:  
 38:/\*#page\_login .ui-block-b {  
 39:   
 40:}\*/  
 41:  
 42:#page\_login /\*.ui-field-contain\*/ .controles {  
 43: width: 90%;  
 44:}  
 45:  
 46:#page\_login /\*.ui-btn\*/ .botao{  
 47: background-image: linear-gradient(to right, #244a7d, #4167ad 130%);  
 48: border: 0 none;  
 49: border-radius: 3px;  
 50: color: #fff;  
 51: font-size: 0.9em;  
 52: height: auto;  
 53: padding: 15px 0;  
 54: text-align: center;  
 55: text-shadow: none;  
 56:}  
 57:  
 58:#page\_login /\*.ui-field-contain\*/ .controles label {  
 59: font-weight: bold;  
 60: color: #244a7d;  
 61: text-shadow: none;  
 62:}  
 63:  
 64:#page\_login .terceira{  
 65: width: 90%;  
 66:}  
 67:  
 68:#page\_login .terceira fieldset{  
 69: float: right;  
 70: margin-right: 50px;  
 71: display: block;  
 72:}  
 73:  
 74:#dataHoraLogin{  
 75: font-size: 2em;  
 76: font-weight: bold;  
 77: text-align: center;  
 78: width: 100%;  
 79: display: block;  
 80:}

## .\mobile\www\css\reset.css

1:/\* http://meyerweb.com/eric/tools/css/reset/   
 2: v2.0 | 20110126  
 3: License: none (public domain)  
 4:\*/  
 5:html, body, div, span, applet, object, iframe, h1, h2, h3, h4, h5, h6, p, blockquote, pre, a, abbr, acronym, address, big, cite,  
 6: code, del, dfn, em, img, ins, kbd, q, s, samp, small, strike, strong, sub, sup, tt, var, b, u, i, center, dl, dt, dd, ol, ul, li,  
 7: fieldset, form, label, legend, table, caption, tbody, tfoot, thead, tr, th, td, article, aside, canvas, details, embed, figure,  
 8: figcaption, footer, header, hgroup, menu, nav, output, ruby, section, summary, time, mark, audio, video {  
 9: margin: 0;  
 10: padding: 0;  
 11: border: 0;  
 12: font-size: 100%;  
 13: font: inherit;  
 14: vertical-align: baseline;  
 15:}  
 16:/\* HTML5 display-role reset for older browsers \*/  
 17:article, aside, details, figcaption, figure, footer, header, hgroup, menu, nav, section {  
 18: display: block;  
 19:}  
 20:  
 21:body {  
 22: line-height: 1;  
 23:}  
 24:  
 25:ol, ul {  
 26: list-style: none;  
 27:}  
 28:  
 29:blockquote, q {  
 30: quotes: none;  
 31:}  
 32:  
 33:blockquote:before, blockquote:after, q:before, q:after {  
 34: content: '';  
 35: content: none;  
 36:}  
 37:  
 38:table {  
 39: border-collapse: collapse;  
 40: border-spacing: 0;  
 41:}

## .\mobile\www\css\style.css

1:input.invalido{  
 2: background-image: url(../icons/SVG/error.svg);  
 3: background-size: 20px 20px;  
 4: background-repeat: no-repeat;  
 5: background-position: left;  
 6: background-position-x: 5px;  
 7:}  
 8:  
 9:.noshadow {  
 10: -moz-box-shadow: none !important;  
 11: -webkit-box-shadow: none !important;  
 12: box-shadow: none !important;  
 13:}  
 14:  
 15:.center-btn {  
 16: margin: 0 auto !important;  
 17:}  
 18:  
 19:.wo-margin-btn {  
 20: margin: 0 !important;  
 21:}  
 22:  
 23:.middle-btn:after {  
 24: top: 50% !important;  
 25: margin-top: -11px;  
 26:}  
 27:  
 28:.small-btn {  
 29: background: #059 !important;  
 30:}  
 31:  
 32:.ui-icon-myMute:after {  
 33: background-image: url("../icons/SVG/mute.svg");  
 34:}  
 35:  
 36:.ui-icon-myRoundArrow:after {  
 37: background-image: url("../icons/SVG/round-arrow-1.svg");  
 38:}  
 39:  
 40:.ui-content h1, h2, h3 {  
 41: display: inline;  
 42:}  
 43:  
 44:.ui-input-text, .ui-input-search {  
 45: margin: 0.75em 0;  
 46:}  
 47:  
 48:.center {  
 49: text-align: center;  
 50:}  
 51:  
 52:.button\_separator {  
 53: margin-bottom: 10%;  
 54:}  
 55:  
 56:.menu\_separator {  
 57: margin-bottom: 30%;  
 58:}  
 59:  
 60:footer span.versao {  
 61: display: inline-block;  
 62: text-align: left;  
 63: width: 19%;  
 64:}  
 65:  
 66:footer span.data\_atual {  
 67: display: inline-block;  
 68: text-align: center;  
 69: width: 19%;  
 70:}  
 71:  
 72:  
 73:footer span.posto {  
 74: display: inline-block;  
 75: text-align: center;  
 76: width: 19%;  
 77:}  
 78:  
 79:footer span.sentido {  
 80: display: inline-block;  
 81: text-align: center;  
 82: width: 19%;  
 83:}  
 84:  
 85:footer span.ipad {  
 86: display: inline-block;  
 87: float: right;  
 88: text-align: right;  
 89: width: 19%;  
 90:}  
 91:  
 92:div.ui-block-b>h2 {  
 93: font-weight: bold;  
 94: font-size: x-large;  
 95:}  
 96:  
 97:section.ui-content>fieldset {  
 98: margin-bottom: 2%;  
 99: min-height: 60px;  
 100: line-height: 44px;  
 101:}  
 102:  
 103:section.ui-content>fieldset>legend.ui-block-a, section.ui-content>div>fieldset>legend.ui-block-a  
 104: {  
 105: width: 32%;  
 106:}  
 107:  
 108:section.ui-content>fieldset>:not(fieldset).ui-block-b, section.ui-content>div>fieldset>:not(fieldset).ui-block-b  
 109: {  
 110: width: 68%;  
 111:}  
 112:  
 113:legend, label {  
 114: text-align: right;  
 115:}  
 116:  
 117:section.ui-content fieldset>legend, section.ui-content fieldset label[class\*="ui-block"]  
 118: {  
 119: display: inline;  
 120: font-weight: bold;  
 121: font-size: x-large;  
 122: line-height: 60px;  
 123: padding-right: .5em;  
 124:}  
 125:  
 126:section.ui-content>fieldset>div.ui-block-b {  
 127: text-align: left;  
 128:}  
 129:  
 130:section.ui-content>fieldset.ui-grid-c>.ui-block-a {  
 131: text-align: right;  
 132: width: 30%;  
 133:}  
 134:  
 135:section.ui-content>fieldset.ui-grid-c>.ui-block-b {  
 136: text-align: left;  
 137: width: 20%;  
 138:}  
 139:  
 140:section.ui-content>fieldset.ui-grid-c>.ui-block-c {  
 141: text-align: center;  
 142: width: 20%;  
 143:}  
 144:  
 145:section.ui-content>fieldset.ui-grid-c>.ui-block-d {  
 146: text-align: left;  
 147: width: 30%;  
 148:}  
 149:  
 150:nav.avancar {  
 151: width: 100%;  
 152: position: fixed;  
 153: bottom: 3%;  
 154: right: 1%;  
 155:}  
 156:  
 157:nav.avancar>div.ui-block-a {  
 158: width: 85%;  
 159:}  
 160:  
 161:nav.avancar>div.ui-block-b {  
 162: width: 15%;  
 163:}  
 164:  
 165:.ui-grid-b>.sel\_pais {  
 166: width: 30%;  
 167:}  
 168:  
 169:.ui-grid-b>.sel\_uf {  
 170: width: 20%;  
 171:}  
 172:  
 173:.ui-grid-b>.sel\_municipio {  
 174: width: 50%;  
 175:}  
 176:  
 177:#sel\_tipo\_veiculo\_1 .classe\_veiculo, #sel\_tipo\_veiculo\_2 .classe\_veiculo  
 178: {  
 179: height: 100%;  
 180:}  
 181:  
 182:#section\_v\_simples {  
 183: width: 45%;  
 184:}  
 185:  
 186:#section\_v\_semireboques {  
 187: width: 30%;  
 188:}  
 189:  
 190:#section\_v\_combinados {  
 191: width: 40%;  
 192:}  
 193:  
 194:#section\_v\_reboques {  
 195: width: 35%;  
 196:}  
 197:  
 198:.troca\_classe\_veiculo {  
 199: width: 15%;  
 200: display: inline-block;  
 201: vertical-align: middle;  
 202: text-align: center;  
 203: padding: 0;  
 204: margin: 0% 3%;  
 205: line-height: 640px;  
 206:}  
 207:  
 208:.troca\_classe\_veiculo .ui-grid-a {  
 209: float: left;  
 210:}  
 211:  
 212:.troca\_classe\_veiculo .ui-grid-c {  
 213: float: right;  
 214:}  
 215:  
 216:.troca\_classe\_veiculo button {  
 217: margin-top: 340px;  
 218:}  
 219:  
 220:.classe\_veiculo {  
 221: display: inline-block;  
 222: vertical-align: top;  
 223: padding: 0;  
 224: margin: 0;  
 225: max-height: 640px;  
 226:}  
 227:  
 228:.classe\_veiculo h2 {  
 229: text-align: center;  
 230: display: block;  
 231: margin: 0px;  
 232: font-weight: bold;  
 233: font-size: large;  
 234: height: 1.5em;  
 235:}  
 236:  
 237:.tipos\_veiculos {  
 238: display: inline-block;  
 239: border: solid;  
 240: text-align: center;  
 241: width: 90%;  
 242: padding: 0% 0%;  
 243: margin: 0% 5%;  
 244: min-height: 600px;  
 245: max-height: 600px;  
 246: background-color: #EEE;  
 247: border-radius: 2%;  
 248:}  
 249:  
 250:.tipos\_veiculos ol {  
 251: display: inherit;  
 252: list-style: none;  
 253: text-align: center;  
 254: width: 100%;  
 255: padding: .2em 0em;  
 256: margin: 0%;  
 257:}  
 258:  
 259:#section\_v\_simples li {  
 260: line-height: 70px;  
 261:}  
 262:  
 263:#section\_v\_semireboques li, #section\_v\_reboques li {  
 264: line-height: 100px;  
 265:}  
 266:  
 267:#section\_v\_combinados li {  
 268: line-height: 120px;  
 269:}  
 270:  
 271:.tipos\_veiculos ol li {  
 272: display: inline;  
 273: vertical-align: middle;  
 274: margin: 0px 10px;  
 275:}  
 276:  
 277:.img\_sel\_veiculo {  
 278: vertical-align: middle;  
 279: max-width: 70%;  
 280:}  
 281:  
 282:.v-simples {  
 283: max-width: 35%;  
 284:}  
 285:  
 286:#m.v-simples { /\* moto \*/  
 287: max-width: 30%;  
 288:}  
 289:  
 290:.v-onibus {  
 291: max-width: 60%;  
 292:}  
 293:  
 294:.v-carga {  
 295: max-width: 42%;  
 296:}  
 297:  
 298:#c3.v-carga { /\* carga: 4CD \*/  
 299: max-width: 40%;  
 300:}  
 301:  
 302:#c4.v-carga { /\* carga: 4CD \*/  
 303: max-width: 48%;  
 304:}  
 305:  
 306:.v-semireboque, .v-combinado, .v-reboque {  
 307: max-width: 90%;  
 308:}  
 309:  
 310:#confirmar\_veiculo>div {  
 311: padding: 1.5em 0%;  
 312:}  
 313:  
 314:#confirmar\_veiculo.ui-grid-b>div:nth-of-type(3n), #confirmar\_veiculo.ui-grid-b>div:nth-of-type(3n+1)  
 315: {  
 316: width: 10%;  
 317:}  
 318:  
 319:#confirmar\_veiculo.ui-grid-b>div:nth-of-type(3n+2) {  
 320: width: 80%;  
 321:}  
 322:  
 323:#confirmar\_veiculo.ui-grid-b nav>div:nth-of-type(1) {  
 324: padding-left: 10%;  
 325: padding-right: 3%;  
 326:}  
 327:  
 328:#confirmar\_veiculo.ui-grid-b nav>div:nth-of-type(2) {  
 329: padding-left: 3%;  
 330: padding-right: 10%;  
 331:}  
 332:  
 333:#confirmar\_veiculo section button {  
 334: background-color: #EEE;  
 335:}  
 336:  
 337:#confirmar\_veiculo section button.ui-btn-icon-left:after {  
 338: left: 5em;  
 339:}  
 340:  
 341:.input\_placa {  
 342: text-transform: uppercase;  
 343:}  
 344:  
 345:ol.tipos\_carrocerias {  
 346: display: inline-block;  
 347: list-style: none;  
 348: text-align: center;  
 349: width: 100%;  
 350: padding: 0%;  
 351: margin: 0%;  
 352: background-color: white;  
 353: border-radius: 1.5em;  
 354:}  
 355:  
 356:ol.tipos\_carrocerias>div {  
 357: max-width: 33%;  
 358: width: 32%;  
 359: height: 100%;  
 360: vertical-align: middle;  
 361: display: inherit;  
 362: text-align: inherit;  
 363: padding: inherit;  
 364: margin: inherit;  
 365: display: inherit;  
 366:}  
 367:  
 368:ol.tipos\_carrocerias>div:nth-of-type(2n) {  
 369: max-width: 31%;  
 370: width: 31%;  
 371:}  
 372:  
 373:ol.tipos\_carrocerias li {  
 374: display: block;  
 375: vertical-align: middle;  
 376: max-width: 100%;  
 377: padding: 0px 7%;  
 378: border-radius: 15px;  
 379:}  
 380:  
 381:ol.tipos\_carrocerias div:nth-child(2n+1) li:nth-child(2n) {  
 382: background-color: #DDD;  
 383:}  
 384:  
 385:ol.tipos\_carrocerias div:nth-child(2n) li:nth-child(2n+1) {  
 386: background-color: #DDD;  
 387:}  
 388:  
 389:ol.tipos\_carrocerias div:nth-child(1) li {  
 390: margin: 7% 2%;  
 391:}  
 392:  
 393:ol.tipos\_carrocerias div:nth-child(2) li {  
 394: margin: 0% 2%;  
 395: padding: 0% 12%;  
 396:}  
 397:  
 398:ol.tipos\_carrocerias div:nth-child(3) li {  
 399: margin: 0% 2%;  
 400: padding: 0% 12%;  
 401:}  
 402:  
 403:.img\_carroceria {  
 404: display: inline-block;  
 405: text-align: center;  
 406: max-width: 100%;  
 407:}  
 408:  
 409:.img\_carroceria ~ span {  
 410: display: block;  
 411: text-align: center;  
 412: max-width: 100%;  
 413: color: black;  
 414:}  
 415:  
 416:#grupo\_placa\_dados\_simples, #grupo\_placa\_unica\_simples>.ui-block-b,  
 417: #grupo\_placa\_dados\_onibus, #grupo\_placa\_unica\_onibus>.ui-block-b,  
 418: #grupo\_placa\_dados\_carga, #grupo\_placa\_unica\_carga>.ui-block-b {  
 419: width: 15%;  
 420:}  
 421:  
 422:div.ui-block-b>.placa\_prefixo {  
 423: width: 40%;  
 424:}  
 425:  
 426:div.ui-block-b>.placa\_sufixo {  
 427: width: 60%;  
 428:}  
 429:  
 430:div.ui-block-b>.placa\_sufixo>.ui-block-a {  
 431: display: inline;  
 432: text-align: center;  
 433: width: 10%;  
 434: padding-top: .4em;  
 435:}  
 436:  
 437:div.ui-block-b>.placa\_sufixo>.ui-block-b {  
 438: width: 90%;  
 439: display: inline-block;  
 440:}  
 441:  
 442:div.ui-block-b>.placa\_prefixo input, div.ui-block-b>.placa\_sufixo input,  
 443: #grupo\_placa\_unica\_simples>.ui-block-b input,  
 444: #grupo\_placa\_unica\_onibus>.ui-block-b input, #grupo\_placa\_unica\_carga>.ui-block-b input  
 445: {  
 446: width: 100%;  
 447: text-align: center;  
 448:}  
 449:  
 450:#grupo\_pessoas\_ambos>fieldset.ui-block-a {  
 451: width: 45%;  
 452:}  
 453:  
 454:#grupo\_pessoas\_ambos>fieldset.ui-block-b {  
 455: width: 55%;  
 456:}  
 457:  
 458:#grupo\_pessoas\_ambos>fieldset>.ui-block-a {  
 459: width: 71%;  
 460:}  
 461:  
 462:#grupo\_pessoas\_ambos>fieldset>.ui-block-b {  
 463: width: 27%;  
 464:}  
 465:  
 466:#grupo\_placa\_vermelha\_carga fieldset.ui-block-a,  
 467: #grupo\_carga\_perigosa\_carga fieldset.ui-block-a {  
 468: width: 25%;  
 469:}  
 470:  
 471:#grupo\_placa\_vermelha\_carga fieldset.ui-block-b,  
 472: #grupo\_carga\_perigosa\_carga fieldset.ui-block-b {  
 473: width: 75%;  
 474:}  
 475:  
 476:#grupo\_peso\_carga>.ui-block-b>.ui-block-a {  
 477: width: 15%;  
 478:}  
 479:  
 480:#caracterizacao\_viagem2\_carga legend.fonte\_menor {  
 481: font-size: 1.3868em;  
 482:}  
 483:  
 484:#caracterizacao\_viagem2\_carga section.ui-content>fieldset {  
 485: margin-bottom: 0%;  
 486: min-height: 1.9em;  
 487:}  
 488:  
 489:#caracterizacao\_viagem2\_carga .ui-controlgroup {  
 490: margin: 0;  
 491:}  
 492:  
 493:#caracterizacao\_viagem2\_carga .ui-input-text {  
 494: margin: 0.2em 0;  
 495:}  
 496:  
 497:#caracterizacao\_viagem2\_carga section.ui-content fieldset>legend {  
 498: line-height: 50px;  
 499:}  
 500:  
 501:#grupo\_valor\_frete\_carga>div.ui-block-b, #grupo\_valor\_nota\_carga>div.ui-block-b  
 502: {  
 503: width: 25%;  
 504:}  
 505:  
 506:#grupo\_sabe\_embarque\_carga>.ui-grid-a>.ui-block-a,  
 507: #grupo\_sabe\_desembarque\_carga>.ui-grid-a>.ui-block-a {  
 508: width: 18%;  
 509:}  
 510:  
 511:#grupo\_sabe\_embarque\_carga>.ui-grid-a>.ui-block-b,  
 512: #grupo\_sabe\_desembarque\_carga>.ui-grid-a>.ui-block-b {  
 513: width: 82%;  
 514:}  
 515:  
 516:#grupo\_sabe\_embarque\_carga>.ui-grid-a>.ui-grid-a>.ui-block-a,  
 517: #grupo\_sabe\_desembarque\_carga>.ui-grid-a>.ui-grid-a>.ui-block-a {  
 518: width: 62%;  
 519:}  
 520:  
 521:#grupo\_sabe\_embarque\_carga>.ui-grid-a>.ui-grid-a>.ui-block-b,  
 522: #grupo\_sabe\_desembarque\_carga>.ui-grid-a>.ui-grid-a>.ui-block-b {  
 523: width: 38%;  
 524:}  
 525:  
 526:#grupo\_sabe\_embarque\_carga>.ui-grid-a>.ui-grid-a>.ui-grid-a>.ui-block-a,  
 527: #grupo\_sabe\_desembarque\_carga>.ui-grid-a>.ui-grid-a>.ui-grid-a>.ui-block-a  
 528: {  
 529: width: 24%;  
 530:}  
 531:  
 532:#grupo\_sabe\_embarque\_carga>.ui-grid-a>.ui-grid-a>.ui-grid-a>.ui-block-b,  
 533: #grupo\_sabe\_desembarque\_carga>.ui-grid-a>.ui-grid-a>.ui-grid-a>.ui-block-b  
 534: {  
 535: width: 76%;  
 536:}  
 537:  
 538:#caracterizacao\_viagem2\_carga #grupo\_municipios\_parada\_carga>legend {  
 539: line-height: 1.2em;  
 540: width: 34%;  
 541:}  
 542:  
 543:#caracterizacao\_viagem2\_carga section.ui-content fieldset.embarque\_desembarque>legend  
 544: {  
 545: line-height: 1.2em;  
 546:}  
 547:  
 548:#caracterizacao\_viagem2\_carga fieldset.embarque\_desembarque>legend {  
 549: width: 21%;  
 550: padding-right: 0em;  
 551:}  
 552:  
 553:#caracterizacao\_viagem2\_carga fieldset.embarque\_desembarque>div {  
 554: width: 79%;  
 555:}  
 556:  
 557:#caracterizacao\_viagem2\_carga #grupo\_municipios\_parada\_carga>div {  
 558: width: 66%;  
 559:}  
 560:  
 561:#grupo\_municipios\_parada\_carga>.ui-grid-a>.ui-block-a {  
 562: width: 22%;  
 563:}  
 564:  
 565:#grupo\_municipios\_parada\_carga>.ui-grid-a>.ui-block-b {  
 566: width: 78%;  
 567:}  
 568:  
 569:#grupo\_municipios\_parada\_carga>.ui-grid-a>.ui-grid-solo>.ui-grid-a>.ui-block-a  
 570: {  
 571: width: 20%;  
 572:}  
 573:  
 574:#grupo\_municipios\_parada\_carga>.ui-grid-a>.ui-grid-solo>.ui-grid-a>.ui-block-b  
 575: {  
 576: width: 80%;  
 577:}  
 578:  
 579:.escondido {  
 580: display: none;  
 581:}  
 582:  
 583:#sumario\_tabela td{  
 584: color: black;  
 585: text-align: center;  
 586:}  
 587:  
 588:#sumario\_tab\_cancelados td{  
 589: color: black;  
 590: text-align: center;  
 591:}  
 592:  
 593:/\* bloqueando copiar no ipad \*/  
 594:\* {  
 595: -webkit-user-select: none;  
 596: -khtml-user-select: none;  
 597: -moz-user-select: none;  
 598: -ms-user-select: none;  
 599: user-select: none;  
 600:}  
 601:/\* permitindo acessar os inputs\*/  
 602:input {  
 603: -webkit-user-select: auto !important;  
 604: -khtml-user-select: auto !important;  
 605: -moz-user-select: auto !important;  
 606: -ms-user-select: auto !important;  
 607: user-select: auto !important;  
 608:}  
 609:  
 610:/\* desbloqueando os inputtext no ipad  
 611:input[type=text] {  
 612: -webkit-user-select: auto;  
 613:}\*/  
 614:  
 615:div.display\_block{  
 616: text-align: center;  
 617:}  
 618:  
 619:h1.display\_value{  
 620: font-weight: bolder;  
 621: font-size: xx-large;  
 622:}  
 623:  
 624:div#display\_data\_hora\_block{  
 625: margin-top: 1%;  
 626: margin-bottom: 2%;  
 627:}  
 628:  
 629:button#menu\_nova\_pesquisa{  
 630: height: 100px;  
 631: font-size: larger;  
 632:}  
 633:  
 634:.avisoTreinamento{  
 635: height: 120px;  
 636: background-color: crimson;  
 637: font-size: xx-large;  
 638: text-align: center;  
 639:}  
 640:  
 641:.qrCode\_group{  
 642: width: 41% !important;  
 643:}  
 644:  
 645:label[for=dataHoraLogin]{  
 646: font-size: 0.9em;  
 647:}

## .\mobile\platforms\ios\www\views\pergunta\textunderscore extra.html

1:<section data-role="page" id="pergunta\_extra" data-transition="slide" data-theme="b">  
 2: <!-- header -->  
 3: <header data-role="header">  
 4: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 5: <h1>Pergunta Extra</h1>  
 6: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 7: </header>  
 8:  
 9: <!--content -->  
 10: <section role="main" class="ui-content">  
 11:   
 12: <!-- Posto 19 e 149 -->  
 13: <fieldset class="escondido" id="grupo\_pergunta\_extra\_p1">  
 14: <legend style="line-height:30px">Se a rodovia BR-251 entre Unaí/MG e Montes Claros/MG estivesse pavimentada, você a usaria para a sua rota?</legend>  
 15: <div>  
 16: <select name="select\_pergunta\_extra\_p1" id="select\_pergunta\_extra\_p1" data-theme="a">  
 17: </select>  
 18: </div>  
 19: </fieldset>  
 20:  
 21: <!-- Posto 149 -->  
 22: <fieldset class="escondido" id="grupo\_pergunta\_extra\_p2">  
 23: <legend style="line-height:30px">Se a rodovia BR-457 entre Vianópolis/GO e Cristalina/GO estivesse pavimentada, você a usaria para a sua rota?</legend>  
 24: <div>  
 25: <select name="select\_pergunta\_extra\_p2" id="select\_pergunta\_extra\_p2" data-theme="a">  
 26: </select>  
 27: </div>  
 28: </fieldset>  
 29:  
 30: <!-- Posto 40 -->  
 31: <fieldset class="escondido" id="grupo\_pergunta\_extra\_p3">  
 32: <legend style="line-height:30px">Se a Rodovia BR-146 entre Araxá/MG e Passos/MG estivesse pavimentada, você a usaria para a sua rota?</legend>  
 33: <div>  
 34: <select name="select\_pergunta\_extra\_p3" id="select\_pergunta\_extra\_p3" data-theme="a">  
 35: </select>  
 36: </div>  
 37: </fieldset>  
 38:  
 39: <nav class="ui-grid-a avancar escondido" id="grupo\_pergunta\_extra\_avancar">  
 40: <div class="ui-block-a">&nbsp;</div>  
 41: <div class="ui-block-b">  
 42: <button class="ui-btn ui-btn-a ui-corner-all ui-icon-arrow-r ui-btn-icon-right" id="pergunta\_extra\_avancar">Finalizar</button>  
 43: </div>  
 44: </nav>  
 45:  
 46: </section>  
 47:  
 48: <!-- footer -->  
 49: <footer data-role='footer' data-position='fixed'>   
 50: <span class='versao'>Versão: <span id='versao'></span></span>   
 51: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 52: <span class='posto'>Posto: <span id='posto'></span></span>   
 53: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 54: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 55: </footer>  
 56:  
 57:</section>

## .\mobile\www\index.html

1:<!DOCTYPE html>  
 2:<html>  
 3:<head>  
 4:<!--  
 5: Customize this policy to fit your own app's needs. For more guidance, see:  
 6: https://github.com/apache/cordova-plugin-whitelist/blob/master/README.md#content-security-policy  
 7: Some notes:  
 8: \* gap: is required only on iOS (when using UIWebView) and is needed for JS->native communication  
 9: \* https://ssl.gstatic.com is required only on Android and is needed for TalkBack to function properly  
 10: \* Disables use of inline scripts in order to mitigate risk of XSS vulnerabilities. To change this:  
 11: \* Enable inline JS: add 'unsafe-inline' to default-src  
 12: -->  
 13:<meta http-equiv="Content-Security-Policy"  
 14: content="default-src 'self' data: gap: https://ssl.gstatic.com 'unsafe-eval'; style-src 'self' 'unsafe-inline'; media-src \*">  
 15:<meta charset="UTF-8">  
 16:<meta name="format-detection" content="telephone=no">  
 17:<meta name="msapplication-tap-highlight" content="no">  
 18:<meta name="viewport" content="user-scalable=no, initial-scale=1, minimum-scale=1, width=device-width, target-densitydpi=device-dpi">  
 19:<link rel="stylesheet" href="css/reset.css" />  
 20:<link rel="stylesheet" href="js/libs/jquery-mobile/jquery.mobile-1.4.5.min.css" />  
 21:<link rel="stylesheet" href="js/libs/dataTable/jquery.dataTables.min.css" />  
 22:<link rel="stylesheet" href="css/jquery-confirm.css" />  
 23:<link rel="stylesheet" href="css/bootstrap\_grid.css" />  
 24:<link rel="stylesheet" href="css/bootstrap\_buttons.css" />  
 25:<link rel="stylesheet" href="css/font-awesome.min.css" />  
 26:<link rel="stylesheet" href="css/style.css" />  
 27:<link rel="stylesheet" href="css/login.css" />  
 28:  
 29:<title>Contagem de Tráfego</title>  
 30:<link rel="shortcut icon" href="">  
 31:</head>  
 32:<body>  
 33:  
 34: <!-- panel: menu -->  
 35: <section data-role="panel" id="panel\_menu" data-display="push">  
 36: <!-- todos os botões devem exigir confirmação de senha para continuar com a operação -->  
 37: <button class="ui-btn ui-corner-all ui-btn-a ui-icon-delete ui-btn-icon-left button\_separator menu\_separator" id="btn\_sair">Sair</button>  
 38: <!-- <button class="ui-btn ui-btn-a ui-corner-all button\_separator" id="duplica\_log" name="duplica\_log">Exportar log de operação</button> -->  
 39: <button class="ui-btn ui-btn-a ui-corner-all button\_separator" id="duplica\_db" name="duplica\_db">Exportar Banco de Dados</button>  
 40: <!-- <button class="ui-btn ui-btn-a ui-corner-all button\_separator menu\_separator" id="exporta\_db\_to\_json" name="exporta\_db\_to\_json">Exportar JSON</button> -->  
 41: <button class="escondido ui-btn ui-btn-a ui-corner-all button\_separator" id="menu\_enviar\_dados" name="menu\_enviar\_dados">Enviar dados <br />ainda não enviados</button>  
 42: <button class="escondido ui-btn ui-btn-a ui-corner-all button\_separator" id="menu\_enviar\_todos\_dados" name="menu\_enviar\_todos\_dados">Enviar todos os dados</button>  
 43: </section>  
 44:  
 45: <!-- page: login -->  
 46: <section data-role="page" id="page\_login" data-theme="b">  
 47:  
 48: <!-- header -->  
 49: <header data-role="header"></header>  
 50:  
 51: <!-- content -->  
 52: <section role="main" class="ui-content">  
 53: <div class="ui-grid-a principal">  
 54: <div class="ui-block-a formulario">  
 55: <div class="ui-field-contain controles">  
 56: <label for="dataHoraLogin">Data e hora:</label>  
 57: <span id="dataHoraLogin">00/00/0000 - 00:00</span>  
 58: </div>  
 59: <div class="ui-field-contain controles">  
 60: <label for="usuario">Usuário:</label>  
 61: <input id="usuario" type="text" name="usuario" data-theme="a" />  
 62: </div>  
 63: <div class="ui-field-contain controles">  
 64: <label for="senha">Senha:</label>  
 65: <input id="senha" type="password" name="senha" data-theme="a" />  
 66: </div>  
 67: <div class="ui-grid-a terceira" >  
 68: <div class="ui-block-a">  
 69: <fieldset id='agregador' data-role="controlgroup" data-type="horizontal" data-theme="a">  
 70: <input type="radio" name="sentido" id="rdo\_sentido\_ab" value="AB">  
 71: <label for="rdo\_sentido\_ab">AB</label>  
 72: <input type="radio" name="sentido" id="rdo\_sentido\_ba" value="BA">  
 73: <label for="rdo\_sentido\_ba">BA</label>  
 74: </fieldset>  
 75: </div>  
 76:   
 77: <div class="ui-block-b">  
 78: <button class="botao ui-btn ui-corner-all ui-btn-a ui-btn-icon-left ui-icon-check" id="entrar" name="entrar">Entrar</button>  
 79: </div>  
 80: </div>  
 81:   
 82: </div>  
 83: <div class="ui-block-b"></div>  
 84: </div>  
 85: </section>  
 86:  
 87: <!-- footer -->  
 88: <footer data-role="footer" data-position="fixed">  
 89: <span class="versao">Versão: <span id="versao"></span></span> <span class="ipad">id-iPad: <span id="ipadID"></span></span>  
 90: </footer>  
 91:  
 92: </section>  
 93:  
 94:</body>  
 95:<script type="text/javascript" src="js/libs/jquery/jquery.js"></script>  
 96:<script type="text/javascript" src="js/libs/jquery/jquery.inputmask.bundle.min.js"></script>  
 97:<script type="text/javascript" src="js/libs/jquery/jquery.maskMoney.min.js"></script>  
 98:<script type="text/javascript" src="js/libs/jquery-mobile/jquery.mobile-1.4.5.min.js"></script>  
 99:<script type="text/javascript" src="js/libs/dataTable/jquery.dataTables.min.js"></script>  
 100:<script type="text/javascript" src="js/libs/jquery-confirm.js"></script>  
 101:<script type="text/javascript" src="js/libs/moment.min.js"></script>  
 102:<script type="text/javascript" src="js/myDialogs.js"></script>  
 103:<script type="text/javascript" src="cordova.js"></script>  
 104:<script type="text/javascript" src="js/myDb.js"></script>  
 105:<script type="text/javascript" src="js/paises.js"></script>  
 106:<script type="text/javascript" src="js/carga\_perigosa.js"></script>  
 107:<script type="text/javascript" src="js/produtos.js"></script>  
 108:<script type="text/javascript" src="js/propriedadeVeiculo.js"></script>  
 109:<script type="text/javascript" src="js/combustivel.js"></script>  
 110:<script type="text/javascript" src="js/motivoEscolhaRota.js"></script>  
 111:<script type="text/javascript" src="js/motivoViagem.js"></script>  
 112:<script type="text/javascript" src="js/util.js"></script>  
 113:<script type="text/javascript" src="js/datas.js"></script>  
 114:  
 115:<script type="text/javascript" src="js/controllers.js"></script>  
 116:<script type="text/javascript" src="js/logger.js"></script>  
 117:<script type="text/javascript" src="js/jsonWriter.js"></script>  
 118:<script type="text/javascript" src="js/ipadID.js"></script>  
 119:<script type="text/javascript" src="js/logins.js"></script>  
 120:<script type="text/javascript" src="js/index.js"></script>  
 121:</html>

## .\mobile\www\views\confirmar\textunderscore veiculo.html

1:<section data-role="page" id="confirmar\_veiculo" data-transition="slide" data-theme="a">  
 2:   
 3: <!-- header -->  
 4: <header data-role="header">  
 5: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 6: <h1>1. Selecione um tipo de veículo</h1>  
 7: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 8: </header>  
 9:  
 10: <!-- content -->  
 11: <section role="main" class="ui-content">  
 12:   
 13: <div id="confirmar\_veiculo" class="ui-grid-b ">  
 14: <div class="ui-block-a">&nbsp;</div>  
 15: <div class="ui-block-b center">  
 16: <h2>Confirma o veículo selecionado?</h2>  
 17: </div>  
 18: <div class="ui-block-c">&nbsp;</div>  
 19:  
 20: <div class="ui-block-a">&nbsp;</div>  
 21: <div class="ui-block-b center">  
 22: <img id="img\_confirmar\_veiculo"/>  
 23: </div>  
 24: <div class="ui-block-c">&nbsp;</div>  
 25:  
 26: <div class="ui-block-a">&nbsp;</div>  
 27: <div class="ui-block-b">  
 28: <nav class="ui-grid-a">  
 29: <div class="ui-block-a">  
 30: <button class="ui-btn ui-shadow ui-corner-all ui-btn-a ui-icon-check ui-btn-icon-left" id="conf\_veic\_sim" name="conf\_veic\_sim">Sim</button>  
 31: </div>  
 32: <div class="ui-block-b">  
 33: <button class="ui-btn ui-shadow ui-corner-all ui-btn-a ui-icon-delete ui-btn-icon-left" id="conf\_veic\_nao" name="conf\_veic\_nao">Não</button>  
 34: </div>  
 35: </nav>  
 36: </div>  
 37: <div class="ui-block-c">&nbsp;</div>  
 38: </div>  
 39:   
 40: </section>  
 41:  
 42: <!-- sem rodapé -->  
 43: <footer data-role='footer' data-position='fixed'>   
 44: <span class='versao'>Versão: <span id='versao'></span></span>   
 45: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 46: <span class='posto'>Posto: <span id='posto'></span></span>   
 47: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 48: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 49: </footer>  
 50:</section>

## .\mobile\www\views\menu.html

1:<section data-role="page" id="menu" data-transition="slide" data-theme="b">  
 2:  
 3: <!-- header -->  
 4: <header data-role="header">  
 5: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 6: <h1>Menu principal</h1>  
 7: </header>  
 8:  
 9: <!-- content -->  
 10: <section role="main" class="ui-content">  
 11: <div class="ui-grid-b">  
 12: <div class="ui-block-a display\_block">  
 13: <h1 id="display\_posto" class="display\_value">POSTO</h1><br>  
 14: <span class="display\_desc">Posto</span>  
 15: </div>  
 16: <div class="ui-block-b display\_block">  
 17: <h1 id="display\_idIpad" class="display\_value">000</h1><br>  
 18: <span class="display\_desc">ID Ipad</span>  
 19: </div>  
 20: <div class="ui-block-c display\_block">  
 21: <h1 id="display\_sentido" class="display\_value">ZZ</h1><br>  
 22: <span class="display\_desc">Sentido</span>  
 23: </div>  
 24: </div>  
 25:   
 26: <div class="ui-grid-solo" id="display\_data\_hora\_block">  
 27: <div class="ui-block-a display\_block">  
 28: <h1 id="display\_data\_hora" class="display\_value">00/00/0000 00:00</h1><br>  
 29: <span class="display\_desc">Data e hora</span>  
 30: </div>  
 31: </div>  
 32:  
 33: <div id='banner\_supra' style="height: 120px"></div>  
 34:   
 35: <div class="ui-grid-b">  
 36: <div class="ui-block-a">&nbsp;</div>  
 37: <div class="ui-block-b">  
 38: <button class="ui-btn ui-btn-a ui-corner-all" id="menu\_nova\_pesquisa" name="menu\_nova\_pesquisa">Iniciar nova pesquisa</button>  
 39: </div>  
 40: <div class="ui-block-c">&nbsp;</div>  
 41: </div>  
 42:   
 43: <div style="height: 180px"></div>  
 44:   
 45: <div class="ui-grid-d">  
 46: <div class="ui-block-a">&nbsp;</div>  
 47: <div class="ui-block-b">&nbsp;</div>  
 48: <div class="ui-block-c">&nbsp;</div>  
 49: <!--<div class="ui-block-d"><button class="ui-btn ui-btn-a ui-corner-all" id="menu\_trocar\_sentido" name="menu\_trocar\_sentido">Trocar Sentido</button></div>-->  
 50: <div class="ui-block-d">&nbsp;</div>  
 51: <div class="ui-block-e"><button class="ui-btn ui-btn-a ui-corner-all" id="menu\_sumario" name="menu\_sumario">Sumário</button></div>  
 52: </div>  
 53:   
 54:<!-- <div class="ui-grid-b">  
 55: <div class="ui-block-a">&nbsp;</div>  
 56: <div class="ui-block-b">  
 57: <button class="ui-btn ui-btn-a ui-corner-all" id="menu\_sumario" name="menu\_sumario">Sumário</button>  
 58: <button class="ui-btn ui-btn-a ui-corner-all" id="testaBugOld" name="testaBugOld">Testa Bug Old troca página</button><br>  
 59: <button class="ui-btn ui-btn-a ui-corner-all" id="testaBugNew" name="testaBugOld">Testa Bug New troca página</button><br>  
 60: </div>  
 61: <div class="ui-block-c">&nbsp;</div>  
 62: </div>-->  
 63:   
 64: </section>  
 65:  
 66: <!-- footer -->  
 67: <footer data-role='footer' data-position='fixed'>   
 68: <span class='versao'>Versão: <span id='versao'></span></span>   
 69: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 70: <span class='posto'>Posto: <span id='posto'></span></span>   
 71: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 72: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 73: </footer>  
 74:  
 75:</section>

## .\mobile\www\views\pergunta\textunderscore extra.html

1:<section data-role="page" id="pergunta\_extra" data-transition="slide" data-theme="b">  
 2: <!-- header -->  
 3: <header data-role="header">  
 4: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 5: <h1>Pergunta Extra</h1>  
 6: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 7: </header>  
 8:  
 9: <!--content -->  
 10: <section role="main" class="ui-content">  
 11:   
 12: <!-- Posto 19 e 149 -->  
 13: <fieldset class="escondido" id="grupo\_pergunta\_extra\_p1">  
 14: <legend style="line-height:30px">Se a rodovia BR-251 entre Unaí/MG e Montes Claros/MG estivesse pavimentada, você a usaria para a sua rota?</legend>  
 15: <div>  
 16: <select name="select\_pergunta\_extra\_p1" id="select\_pergunta\_extra\_p1" data-theme="a">  
 17: </select>  
 18: </div>  
 19: </fieldset>  
 20:  
 21: <!-- Posto 149 -->  
 22: <fieldset class="escondido" id="grupo\_pergunta\_extra\_p2">  
 23: <legend style="line-height:30px">Se a rodovia BR-457 entre Vianópolis/GO e Cristalina/GO estivesse pavimentada, você a usaria para a sua rota?</legend>  
 24: <div>  
 25: <select name="select\_pergunta\_extra\_p2" id="select\_pergunta\_extra\_p2" data-theme="a">  
 26: </select>  
 27: </div>  
 28: </fieldset>  
 29:  
 30: <!-- Posto 40 -->  
 31: <fieldset class="escondido" id="grupo\_pergunta\_extra\_p3">  
 32: <legend style="line-height:30px">Se a Rodovia BR-146 entre Araxá/MG e Passos/MG estivesse pavimentada, você a usaria para a sua rota?</legend>  
 33: <div>  
 34: <select name="select\_pergunta\_extra\_p3" id="select\_pergunta\_extra\_p3" data-theme="a">  
 35: </select>  
 36: </div>  
 37: </fieldset>  
 38:  
 39: <nav class="ui-grid-a avancar escondido" id="grupo\_pergunta\_extra\_avancar">  
 40: <div class="ui-block-a">&nbsp;</div>  
 41: <div class="ui-block-b">  
 42: <button class="ui-btn ui-btn-a ui-corner-all ui-icon-arrow-r ui-btn-icon-right" id="pergunta\_extra\_avancar">Finalizar</button>  
 43: </div>  
 44: </nav>  
 45:  
 46: </section>  
 47:  
 48: <!-- footer -->  
 49: <footer data-role='footer' data-position='fixed'>   
 50: <span class='versao'>Versão: <span id='versao'></span></span>   
 51: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 52: <span class='posto'>Posto: <span id='posto'></span></span>   
 53: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 54: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 55: </footer>  
 56:  
 57:</section>

## .\mobile\www\views\selecionar\textunderscore tipo.html

1:<section data-role="page" id="selecionar\_tipo" data-transition="slide" data-theme="a">  
 2:  
 3: <!-- header -->  
 4: <header data-role="header">  
 5: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 6: <h1>1. Selecione um tipo de veículo</h1>  
 7: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 8: </header>  
 9:  
 10: <!-- content -->  
 11: <section role="main" class="ui-content" id="sel\_tipo\_veiculo\_1">  
 12:  
 13: <section class="ui-grid-a classe\_veiculo" id="section\_v\_simples">  
 14: <h2>Veículos Simples</h2>  
 15: <nav class="tipos\_veiculos">  
 16: <ol>  
 17: <li><img src="../img/tipoVeiculo/1\_simples/p1\_01.png" class="img\_sel\_veiculo v-simples" id="p1\_01" limitePessoas="7" /></li>  
 18: <li><img src="../img/tipoVeiculo/1\_simples/p1\_02.png" class="img\_sel\_veiculo v-simples" id="p1\_02" limitePessoas="15" /></li>  
 19: <li><img src="../img/tipoVeiculo/1\_simples/p3.png" class="img\_sel\_veiculo v-simples" id="p3" limitePessoas="5" /></li>  
 20: <li><img src="../img/tipoVeiculo/1\_simples/p2.png" class="img\_sel\_veiculo v-simples" id="p2" limitePessoas="21" /></li>  
 21: <li><img src="../img/tipoVeiculo/1\_simples/m.png" class="img\_sel\_veiculo v-simples" id="m" limitePessoas="2" /></li>  
 22: <li><img src="../img/tipoVeiculo/1\_simples/o1.png" class="img\_sel\_veiculo v-onibus" id="o1" limitePessoas="99" /></li>  
 23: <li><img src="../img/tipoVeiculo/1\_simples/o2.png" class="img\_sel\_veiculo v-onibus" id="o2" limitePessoas="99" /></li>  
 24: <li><img src="../img/tipoVeiculo/1\_simples/o3.png" class="img\_sel\_veiculo v-onibus" id="o3" limitePessoas="99" /></li>  
 25: <li><img src="../img/tipoVeiculo/1\_simples/c1.png" class="img\_sel\_veiculo v-carga" id="c1" /></li>  
 26: <li><img src="../img/tipoVeiculo/1\_simples/c2.png" class="img\_sel\_veiculo v-carga" id="c2" /></li>  
 27: <li><img src="../img/tipoVeiculo/1\_simples/c3.png" class="img\_sel\_veiculo v-carga" id="c3" /></li>  
 28: <li><img src="../img/tipoVeiculo/1\_simples/c4.png" class="img\_sel\_veiculo v-carga" id="c4" /></li>  
 29: <!-- <li><img src="../img/tipoVeiculo/1\_simples/c5.png" class="img\_sel\_veiculo v-carga" id="c5" /></li> -->  
 30: </ol>  
 31: </nav>  
 32: </section>  
 33:  
 34: <section class="ui-grid-b classe\_veiculo" id="section\_v\_semireboques">  
 35: <h2>Semi-reboques</h2>  
 36: <nav class="tipos\_veiculos">  
 37: <ol>  
 38: <li><img src="../img/tipoVeiculo/2\_semireboques/s1.png" class="img\_sel\_veiculo v-semireboque" id="s1" /></li>  
 39: <li><img src="../img/tipoVeiculo/2\_semireboques/s2.png" class="img\_sel\_veiculo v-semireboque" id="s2" /></li>  
 40: <li><img src="../img/tipoVeiculo/2\_semireboques/s3.png" class="img\_sel\_veiculo v-semireboque" id="s3" /></li>  
 41: <li><img src="../img/tipoVeiculo/2\_semireboques/s4.png" class="img\_sel\_veiculo v-semireboque" id="s4" /></li>  
 42: <li><img src="../img/tipoVeiculo/2\_semireboques/s5.png" class="img\_sel\_veiculo v-semireboque" id="s5" /></li>  
 43: <li><img src="../img/tipoVeiculo/2\_semireboques/s6.png" class="img\_sel\_veiculo v-semireboque" id="s6" /></li>  
 44: </ol>  
 45: </nav>  
 46: </section>  
 47:  
 48: <section class="ui-grid-c troca\_classe\_veiculo">  
 49: <!--<img id="selTipoVeiculo\_1" src="../icons/SVG/right-arrow.svg" class="switch\_classe\_veiculo" />-->  
 50: <button id="selTipoVeiculo\_1" class="ui-btn ui-btn-a ui-btn-icon-top ui-icon-arrow-r switch\_classe\_veiculo">  
 51: Combinados<br />e Reboques  
 52: </button>  
 53: </section>  
 54: </section>  
 55:  
 56: <section role="main" class="ui-content escondido" id="sel\_tipo\_veiculo\_2">  
 57: <section class="ui-grid-a troca\_classe\_veiculo">  
 58: <!--<img id="selTipoVeiculo\_2" src="../icons/SVG/left-arrow.svg" class="switch\_classe\_veiculo" />-->  
 59: <button id="selTipoVeiculo\_2" class="ui-btn ui-btn-a ui-btn-icon-top ui-icon-arrow-l switch\_classe\_veiculo">  
 60: Simples e<br />Semi-reboques  
 61: </button>  
 62: </section>  
 63: <section class="ui-grid-b classe\_veiculo" id="section\_v\_combinados">  
 64: <h2>Combinados</h2>  
 65: <nav class="tipos\_veiculos">  
 66: <ol>  
 67: <li><img src="../img/tipoVeiculo/3\_combinados/se1.png" class="img\_sel\_veiculo v-combinado" id="se1" /></li>  
 68: <li><img src="../img/tipoVeiculo/3\_combinados/se2.png" class="img\_sel\_veiculo v-combinado" id="se2" /></li>  
 69: <li><img src="../img/tipoVeiculo/3\_combinados/se3.png" class="img\_sel\_veiculo v-combinado" id="se3" /></li>  
 70: <li><img src="../img/tipoVeiculo/3\_combinados/se4.png" class="img\_sel\_veiculo v-combinado" id="se4" /></li>  
 71: <li><img src="../img/tipoVeiculo/3\_combinados/se5.png" class="img\_sel\_veiculo v-combinado" id="se5" /></li>  
 72: </ol>  
 73: </nav>  
 74: </section>  
 75:  
 76: <section class="ui-grid-c classe\_veiculo" id="section\_v\_reboques">  
 77: <h2>Reboques</h2>  
 78: <nav class="tipos\_veiculos">  
 79: <ol>  
 80: <li><img src="../img/tipoVeiculo/4\_reboques/r1.png" class="img\_sel\_veiculo v-reboque" id="r1" /></li>  
 81: <li><img src="../img/tipoVeiculo/4\_reboques/r2.png" class="img\_sel\_veiculo v-reboque" id="r2" /></li>  
 82: <li><img src="../img/tipoVeiculo/4\_reboques/r3.png" class="img\_sel\_veiculo v-reboque" id="r3" /></li>  
 83: <li><img src="../img/tipoVeiculo/4\_reboques/r4.png" class="img\_sel\_veiculo v-reboque" id="r4" /></li>  
 84: <li><img src="../img/tipoVeiculo/4\_reboques/r5.png" class="img\_sel\_veiculo v-reboque" id="r5" /></li>  
 85: <li><img src="../img/tipoVeiculo/4\_reboques/r6.png" class="img\_sel\_veiculo v-reboque" id="r6" /></li>  
 86: </ol>  
 87: </nav>  
 88: </section>  
 89:  
 90: </section>  
 91:  
 92: <footer data-role='footer' data-position='fixed'>   
 93: <span class='versao'>Versão: <span id='versao'></span></span>   
 94: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 95: <span class='posto'>Posto: <span id='posto'></span></span>   
 96: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 97: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 98: </footer>  
 99:</section>

## .\mobile\www\views\sumario.html

1:<section data-role="page" id="sumario" data-transition="slide" data-theme="b">  
 2:  
 3: <!-- header -->  
 4: <header data-role="header">  
 5: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 6: <h1>Sumário</h1>  
 7: </header>  
 8:  
 9: <!-- content -->  
 10: <section role="main" class="ui-content">  
 11:  
 12: <!-- Tabela com quantidade de pesquisas, média, soma, máximo e mínimo de cada dia -->  
 13: <table id="sumario\_tabela" class="display, cell-border" width="100%"></table>  
 14:   
 15: <br><br>  
 16: <p><output name="sumario\_ult\_pesq" id="sumario\_ult\_pesq"></output></p>  
 17: <br><br>  
 18:  
 19: <!-- Tabela com dados de pesquisas canceladas -->  
 20: <table id="sumario\_tab\_cancelados" class="display, cell-border" width="100%"></table>  
 21:  
 22: <br><br>  
 23: <!-- Retorna ao Menu -->  
 24: <div class="ui-grid-b">  
 25: <div class="ui-block-a">&nbsp;</div>  
 26: <div class="ui-block-b">  
 27: <button class="ui-btn ui-btn-a ui-corner-all" id="sumario\_retornar\_menu" name="sumario\_retornar\_menu">Retornar ao Menu</button>  
 28: </div>  
 29: <div class="ui-block-c">&nbsp;</div>  
 30: </div>  
 31:  
 32: </section>  
 33:  
 34: <!-- footer -->  
 35: <footer data-role='footer' data-position='fixed'>   
 36: <span class='versao'>Versão: <span id='versao'></span></span>   
 37: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 38: <span class='posto'>Posto: <span id='posto'></span></span>   
 39: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 40: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 41: </footer>  
 42:  
 43:</section>

## .\mobile\www\views\carga\caracterizacao\textunderscore carga.html

1:<section data-role="page" id="caracterizacao\_carga" data-transition="slide" data-theme="b">  
 2:  
 3: <!-- header -->  
 4: <header data-role="header">  
 5: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 6: <h1>3. Caracterização do veículo</h1>  
 7: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 8: </header>  
 9:  
 10: <!--content -->  
 11: <section role="main" class="ui-content">  
 12:  
 13: <fieldset class="ui-grid-a" id="grupo\_ano\_carga">  
 14: <legend class="ui-block-a">Ano de fabricação:</legend>  
 15: <div class="ui-block-b">  
 16: <select name="ano\_carga" id="ano\_carga" data-theme="a">  
 17: </select>  
 18: </div>  
 19: </fieldset>  
 20:   
 21: <fieldset class="ui-grid-a escondido" id="grupo\_propriedade\_carga">  
 22: <legend class="ui-block-a">Propriedade:</legend>  
 23: <div class="ui-block-b">  
 24: <select name="propriedade\_carga" id="propriedade\_carga" data-theme="a">  
 25: </select>  
 26: </div>  
 27: </fieldset>  
 28:  
 29: <fieldset class="ui-grid-a escondido" id="grupo\_agregado\_carga">  
 30: <legend class="ui-block-a">Agregado:</legend>  
 31: <div class="ui-block-b">  
 32: <select name="agregado\_carga" id="agregado\_carga" data-theme="a">  
 33: </select>  
 34: </div>  
 35: </fieldset>  
 36:  
 37: <nav class="ui-grid-a avancar escondido" id="grupo\_caracterizacao\_carga\_avancar">  
 38: <div class="ui-block-a">&nbsp;</div>  
 39: <div class="ui-block-b">  
 40: <button class="ui-btn ui-corner-all ui-icon-arrow-r ui-btn-icon-right ui-btn-a" id="caracterizacao\_carga\_avancar">Avançar</button>  
 41: </div>  
 42: </nav>  
 43:  
 44: </section>  
 45:  
 46: <!-- footer -->  
 47: <footer data-role='footer' data-position='fixed'>   
 48: <span class='versao'>Versão: <span id='versao'></span></span>   
 49: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 50: <span class='posto'>Posto: <span id='posto'></span></span>   
 51: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 52: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 53: </footer>  
 54:  
 55:</section>

## .\mobile\www\views\carga\caracterizacao\textunderscore viagem2\textunderscore carga.html

1:<section data-role="page" id="caracterizacao\_viagem2\_carga" data-transition="slide" data-theme="b">  
 2:  
 3: <!-- header -->  
 4: <header data-role="header">  
 5: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 6: <h1>5. Caracterização da viagem (Continuação)</h1>  
 7: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 8: </header>  
 9:  
 10: <!--content -->  
 11: <section role="main" class="ui-content">  
 12:  
 13: <fieldset class="ui-grid-a" id="grupo\_possui\_carga">  
 14: <legend class="ui-block-a">Possui carga:</legend>  
 15: <div data-role="controlgroup" data-theme="a" data-type="horizontal" class="ui-block-b">  
 16: <input type="radio" name="possui\_carga" id="possui\_carga\_sim" value="sim" />  
 17: <label for="possui\_carga\_sim">Sim</label>  
 18: <input type="radio" name="possui\_carga" id="possui\_carga\_nao" value="nao" />  
 19: <label for="possui\_carga\_nao">Não</label>  
 20: </div>  
 21: </fieldset>  
 22:  
 23: <!-- POSSUI CARGA: SIM -->  
 24: <div class="escondido" id="div\_possui\_carga\_sim">  
 25: <fieldset class="ui-grid-a" id="grupo\_produto\_carga">  
 26: <legend class="ui-block-a">Tipo de carga:</legend>  
 27: <div class="ui-block-b">  
 28: <input data-type='search' type="text" name="produto\_carga" id="produto\_carga" data-theme="a" readonly="readonly">  
 29: </div>  
 30: </fieldset>  
 31:  
 32: <fieldset class="ui-grid-a escondido" id="grupo\_peso\_carga">  
 33: <legend class="ui-block-a">Peso da carga:</legend>  
 34: <div class="ui-block-b ui-grid-a">  
 35: <div class="ui-block-a">  
 36: <input id="peso\_carga" type="text" pattern="\d\*" mask="9{+}[,9]" mask-decimal-separator="," name="peso\_carga" value=""  
 37: data-theme="a" />  
 38: </div>  
 39: <div class="ui-block-b" data-role="controlgroup" data-theme="a" data-type="horizontal">  
 40: <input type="radio" name="unidade\_peso\_carga" id="unidade\_peso\_carga\_ton" value="ton" checked="checked" />  
 41: <label for="unidade\_peso\_carga\_ton">Ton</label>  
 42: <input type="radio" name="unidade\_peso\_carga" id="unidade\_peso\_carga\_kg" value="kg" />  
 43: <label for="unidade\_peso\_carga\_kg">Kg</label>  
 44: </div>  
 45: </div>  
 46: </fieldset>  
 47:  
 48: <fieldset class="ui-grid-a escondido" id="grupo\_valor\_frete\_carga">  
 49: <legend class="ui-block-a">Valor do frete:</legend>  
 50: <div class="ui-block-b">  
 51: <input id="valor\_frete\_carga" type="text" pattern="\d\*" typeMask="money-BRL" name="valor\_frete\_carga" value="" data-theme="a" />  
 52: </div>  
 53: </fieldset>  
 54:  
 55: <fieldset class="ui-grid-a escondido" id="grupo\_valor\_nota\_carga">  
 56: <legend class="ui-block-a">Valor da carga/nota fiscal:</legend>  
 57: <div class="ui-block-b">  
 58: <input id="valor\_nota\_carga" type="text" pattern="\d\*" typeMask="money-BRL" name="valor\_nota\_carga" value="" data-theme="a" />  
 59: </div>  
 60: </fieldset>  
 61:  
 62: <!-- EMBARQUE DA CARGA -->  
 63: <fieldset class="ui-grid-a escondido embarque\_desembarque" id="grupo\_sabe\_embarque\_carga">  
 64: <legend class="ui-block-a">Local embarque da carga:</legend>  
 65: <div class="ui-block-b ui-grid-a">  
 66: <div class="ui-block-a">  
 67: <input id="embarque\_carga\_nao\_sei" type="checkbox" name="embarque\_carga\_nao\_sei" data-theme="a" />  
 68: <label for="embarque\_carga\_nao\_sei">Não sabe informar</label>  
 69: </div>  
 70: <div class="ui-block-b ui-grid-a" id="div\_sabe\_embarque\_sim">  
 71: <div class="ui-block-a ui-grid-a">  
 72: <div class="ui-block-a" id="grupo\_embarque\_uf\_carga">  
 73: <select name="embarque\_uf\_carga" id="embarque\_uf\_carga" data-theme="a">  
 74: </select>  
 75: </div>  
 76: <div class="ui-block-b sel\_municipio escondido" id="grupo\_embarque\_mun\_carga">  
 77: <input data-type='search' type="text" name="embarque\_mun\_carga" id="embarque\_mun\_carga" data-theme="a" readonly="readonly">  
 78: </div>  
 79: </div>  
 80: <div class="ui-block-b escondido" id="grupo\_embarque\_local\_carga">  
 81: <select name="embarque\_local\_carga" id="embarque\_local\_carga" data-theme="a">  
 82: </select>  
 83: </div>  
 84: </div>  
 85: </div>  
 86: </fieldset>  
 87:  
 88: <!-- DESEMBARQUE DA CARGA -->  
 89: <fieldset class="ui-grid-a escondido embarque\_desembarque" id="grupo\_sabe\_desembarque\_carga">  
 90: <legend class="ui-block-a fonte\_menor">Local desembarque da carga:</legend>  
 91: <div class="ui-block-b ui-grid-a">  
 92: <div class="ui-block-a">  
 93: <input id="desembarque\_carga\_nao\_sei" type="checkbox" name="desembarque\_carga\_nao\_sei" data-theme="a" />  
 94: <label for="desembarque\_carga\_nao\_sei">Não sabe informar</label>  
 95: </div>  
 96: <div class="ui-block-b ui-grid-a" id="div\_sabe\_desembarque\_sim">  
 97: <div class="ui-block-a ui-grid-a">  
 98: <div class="ui-block-a" id="grupo\_desembarque\_uf\_carga">  
 99: <select name="desembarque\_uf\_carga" id="desembarque\_uf\_carga" data-theme="a">  
 100: </select>  
 101: </div>  
 102: <div class="ui-block-b sel\_municipio escondido" id="grupo\_desembarque\_mun\_carga">  
 103: <input data-type='search' type="text" name="desembarque\_mun\_carga" id="desembarque\_mun\_carga" data-theme="a"  
 104: readonly="readonly">  
 105: </div>  
 106: </div>  
 107: <div class="ui-block-b escondido" id="grupo\_desembarque\_local\_carga">  
 108: <select name="desembarque\_local\_carga" id="desembarque\_local\_carga" data-theme="a">  
 109: </select>  
 110: </div>  
 111: </div>  
 112: </div>  
 113: </fieldset>  
 114: </div>  
 115:  
 116: <!-- POSSUI CARGA: NAO -->  
 117: <div class="escondido" id="div\_possui\_carga\_nao">  
 118: <fieldset class="ui-grid-a" id="grupo\_carga\_anterior\_carga">  
 119: <legend class="ui-block-a">Carga anterior:</legend>  
 120: <div data-role="controlgroup" data-theme="a" data-type="horizontal" class="ui-block-b">  
 121: <input type="radio" name="carga\_anterior\_carga" id="carga\_anterior\_carga\_sim" value="sim" />  
 122: <label for="carga\_anterior\_carga\_sim">Sim</label>  
 123: <input type="radio" name="carga\_anterior\_carga" id="carga\_anterior\_carga\_nao" value="nao" />  
 124: <label for="carga\_anterior\_carga\_nao">Não</label>  
 125: </div>  
 126: </fieldset>  
 127: <fieldset class="ui-grid-a escondido" id="grupo\_produto\_anterior\_carga">  
 128: <legend class="ui-block-a">Tipo de carga:</legend>  
 129: <div class="ui-block-b">  
 130: <input data-type='search' type="text" name="produto\_anterior\_carga" id="produto\_anterior\_carga" data-theme="a"  
 131: readonly="readonly">  
 132: </div>  
 133: </fieldset>  
 134:  
 135: <fieldset class="ui-grid-a escondido" id="grupo\_indo\_pegar\_carga">  
 136: <legend class="ui-block-a">Indo pegar uma carga:</legend>  
 137: <div data-role="controlgroup" data-theme="a" data-type="horizontal" class="ui-block-b">  
 138: <input type="radio" name="indo\_pegar\_carga" id="indo\_pegar\_carga\_sim" value="sim" />  
 139: <label for="indo\_pegar\_carga\_sim">Sim</label>  
 140: <input type="radio" name="indo\_pegar\_carga" id="indo\_pegar\_carga\_nao" value="nao" />  
 141: <label for="indo\_pegar\_carga\_nao">Não</label>  
 142: </div>  
 143: </fieldset>  
 144: </div>  
 145:  
 146: <fieldset class="ui-grid-a escondido" id="grupo\_parada\_especial\_carga">  
 147: <legend class="ui-block-a">Utiliza parada especial:</legend>  
 148: <div data-role="controlgroup" data-theme="a" data-type="horizontal" class="ui-block-b">  
 149: <input type="radio" name="parada\_especial\_carga" id="parada\_especial\_carga\_sim" value="sim" />  
 150: <label for="parada\_especial\_carga\_sim">Sim</label>  
 151: <input type="radio" name="parada\_especial\_carga" id="parada\_especial\_carga\_nao" value="nao" />  
 152: <label for="parada\_especial\_carga\_nao">Não</label>  
 153: </div>  
 154: </fieldset>  
 155:  
 156: <fieldset class="ui-grid-a escondido" id="grupo\_municipios\_parada\_carga">  
 157: <legend class="ui-block-a fonte\_menor">Na sua rota de viagem, quais seriam os melhores municípios para que houvesse um  
 158: local de parada obrigatória de descanso:</legend>  
 159: <div class="ui-block-b ui-grid-a">  
 160: <div class="ui-block-a">  
 161: <input id="municipios\_parada\_nao\_sei\_carga" type="checkbox" name="municipios\_parada\_nao\_sei\_carga" data-theme="a" />  
 162: <label for="municipios\_parada\_nao\_sei\_carga">Não sabe informar</label>  
 163: </div>  
 164: <div class="ui-block-b ui-grid-solo" id="div\_municipios\_parada\_carga">  
 165: <div class="ui-block-a ui-grid-a">  
 166: <div class="ui-block-a" id="grupo\_municipios\_parada\_uf1\_carga">  
 167: <select name="municipios\_parada\_uf1\_carga" id="municipios\_parada\_uf1\_carga" data-theme="a">  
 168: </select>  
 169: </div>  
 170: <div class="ui-block-b sel\_municipio escondido" id="grupo\_municipios\_parada\_mun1\_carga">  
 171: <input data-type='search' type="text" name="municipios\_parada\_mun1\_carga" id="municipios\_parada\_mun1\_carga" data-theme="a"  
 172: readonly="readonly">  
 173: </div>  
 174: </div>  
 175: <div class="ui-block-a ui-grid-a">  
 176: <div class="ui-block-a" id="grupo\_municipios\_parada\_uf2\_carga">  
 177: <select name="municipios\_parada\_uf2\_carga" id="municipios\_parada\_uf2\_carga" data-theme="a">  
 178: </select>  
 179: </div>  
 180: <div class="ui-block-b sel\_municipio escondido" id="grupo\_municipios\_parada\_mun2\_carga">  
 181: <input data-type='search' type="text" name="municipios\_parada\_mun2\_carga" id="municipios\_parada\_mun2\_carga" data-theme="a"  
 182: readonly="readonly">  
 183: </div>  
 184: </div>  
 185: </div>  
 186: </div>  
 187: </fieldset>  
 188:  
 189: <nav class="ui-grid-a avancar escondido" style="position: relative" id="grupo\_caracterizacao\_viagem2\_carga\_avancar">  
 190: <div class="ui-block-a">&nbsp;</div>  
 191: <div class="ui-block-b">  
 192: <button class="ui-btn ui-corner-all ui-icon-arrow-r ui-btn-icon-right ui-btn-a" id="caracterizacao\_viagem2\_carga\_avancar">Finalizar</button>  
 193: </div>  
 194: </nav>  
 195: </section>  
 196:  
 197: <!-- footer -->  
 198: <footer data-role='footer' data-position='fixed'>   
 199: <span class='versao'>Versão: <span id='versao'></span></span>   
 200: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 201: <span class='posto'>Posto: <span id='posto'></span></span>   
 202: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 203: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 204: </footer>  
 205:  
 206:</section>

## .\mobile\www\views\carga\caracterizacao\textunderscore viagem\textunderscore carga.html

1:<section data-role="page" id="caracterizacao\_viagem\_carga" data-transition="slide" data-theme="b">  
 2:  
 3: <!-- header -->  
 4: <header data-role="header">  
 5: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 6: <h1>4. Caracterização da viagem</h1>  
 7: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 8: </header>  
 9:  
 10: <!--content -->  
 11: <section role="main" class="ui-content">  
 12:  
 13: <fieldset class="ui-grid-a" id="grupo\_origem\_carga">  
 14: <legend class="ui-block-a">Origem da viagem:</legend>  
 15: <div class="ui-block-b ui-grid-b" id="grupo\_origem\_carga">  
 16: <div class="ui-block-a sel\_pais" id="grupo\_origem\_pais\_carga">  
 17: <select name="origem\_pais\_carga" id="origem\_pais\_carga" data-theme="a">  
 18: </select>  
 19: </div>  
 20: <div class="ui-block-b sel\_uf" id="grupo\_origem\_uf\_carga">  
 21: <select name="origem\_uf\_carga" id="origem\_uf\_carga" data-theme="a">  
 22: </select>  
 23: </div>  
 24: <div class="ui-block-c sel\_municipio escondido" id="grupo\_origem\_municipio\_carga">  
 25: <input data-type='search' type="text" name="origem\_municipio\_carga" id="origem\_municipio\_carga" data-theme="a"  
 26: readonly="readonly">  
 27: </div>  
 28: </div>  
 29: </fieldset>  
 30:  
 31: <fieldset class="ui-grid-a escondido" id="grupo\_destino\_carga">  
 32: <legend class="ui-block-a">Destino da viagem:</legend>  
 33: <div class="ui-block-b ui-grid-b" id="grupo\_destino\_carga">  
 34: <div class="ui-block-a sel\_pais" id="grupo\_destino\_pais\_carga">  
 35: <select name="destino\_pais\_carga" id="destino\_pais\_carga" data-theme="a">  
 36: </select>  
 37: </div>  
 38: <div class="ui-block-b sel\_uf" id="grupo\_destino\_uf\_carga">  
 39: <select name="destino\_uf\_carga" id="destino\_uf\_carga" data-theme="a">  
 40: </select>  
 41: </div>  
 42: <div class="ui-block-c sel\_municipio escondido" id="grupo\_destino\_municipio\_carga">  
 43: <input data-type='search' type="text" name="destino\_municipio\_carga" id="destino\_municipio\_carga" data-theme="a"  
 44: readonly="readonly">  
 45: </div>  
 46: </div>  
 47: </fieldset>  
 48:  
 49: <fieldset class="ui-grid-a escondido" id="grupo\_frequencia\_carga">  
 50: <legend class="ui-block-a">Frequência da viagem:</legend>  
 51: <div class="ui-block-b ui-grid-b">  
 52: <div class="ui-block-a">  
 53: <input id="frequencia\_num\_carga" type="text" pattern="\d\*" typeMask="integer" name="frequencia\_num\_carga" value="" min="1"  
 54: data-theme="a" />  
 55: </div>  
 56: <label class="ui-block-b center" for="frequencia\_sel\_carga">por</label>  
 57: <div class="ui-block-c">  
 58: <select name="frequencia\_sel\_carga" id="frequencia\_sel\_carga" data-theme="a">  
 59: </select>  
 60: </div>  
 61: </div>  
 62: </fieldset>  
 63:  
 64: <fieldset class="ui-grid-a escondido" id="grupo\_motivo\_rota\_carga">  
 65: <legend class="ui-block-a">Motivo da escolha da rota:</legend>  
 66: <div class="ui-block-b">  
 67: <select name="motivo\_rota\_carga" id="motivo\_rota\_carga" data-theme="a">  
 68: </select>  
 69: </div>  
 70: </fieldset>  
 71:  
 72: <nav class="ui-grid-a avancar escondido" id="grupo\_caracterizacao\_viagem\_carga\_avancar">  
 73: <div class="ui-block-a">&nbsp;</div>  
 74: <div class="ui-block-b">  
 75: <button class="ui-btn ui-corner-all ui-icon-arrow-r ui-btn-icon-right ui-btn-a" id="caracterizacao\_viagem\_carga\_avancar">Avançar</button>  
 76: </div>  
 77: </nav>  
 78:  
 79: </section>  
 80:  
 81: <!-- footer -->  
 82: <footer data-role='footer' data-position='fixed'>   
 83: <span class='versao'>Versão: <span id='versao'></span></span>   
 84: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 85: <span class='posto'>Posto: <span id='posto'></span></span>   
 86: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 87: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 88: </footer>  
 89:  
 90:</section>

## .\mobile\www\views\carga\identificacao\textunderscore visual.html

1:<section data-role="page" id="identificacao\_visual\_carga" data-transition="slide" data-theme="b">  
 2: <!-- header -->  
 3: <header data-role="header">  
 4: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 5: <h1>2. Identificação visual do veículo</h1>  
 6: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 7: </header>  
 8:  
 9: <!--content -->  
 10: <section role="main" class="ui-content">  
 11:  
 12: <fieldset class="ui-grid-a" id="grupo\_tipo\_carroceria\_carga">  
 13: <legend class="ui-block-a">Tipo de carroceria:</legend>  
 14: <div data-role="controlgroup" data-theme="a" data-type="horizontal" class="ui-block-b">  
 15: <input type="radio" id="tipo\_carroceria\_carga" name="tipo\_carroceria\_carga" value="" />  
 16: <label for="tipo\_carroceria\_carga">Selecione</label>  
 17: </div>  
 18: </fieldset>  
 19:  
 20: <section class="ui-grid-a" id="grupo\_tipo\_carroceria\_imagens\_carga">  
 21: <ol class="tipos\_carrocerias">  
 22: <div>  
 23: <li><img src="../../img/carroceria/aberta.png" class="img\_carroceria" alt="Aberta" id="carroceria\_1" /></li>  
 24: <li><img src="../../img/carroceria/cegonha.png" class="img\_carroceria" alt="Cegonha" id="carroceria\_7" /></li>  
 25: <li><img src="../../img/carroceria/graneleiro.png" class="img\_carroceria" alt="Graneleiro" id="carroceria\_9" /></li>  
 26: <li><img src="../../img/carroceria/porta\_conteiner.png" class="img\_carroceria tipo\_porta\_conteiner"  
 27: alt="Porta-Conteiner" id="carroceria\_10" /></li>  
 28: </div>  
 29: <div>  
 30: <li><img src="../../img/carroceria/bau\_simples.png" class="img\_carroceria" alt="Baú Simples" id="carroceria\_4" /></li>  
 31: <li><img src="../../img/carroceria/bau\_frigorifico.png" class="img\_carroceria" alt="Baú Frigorífico" id="carroceria\_3" /></li>  
 32: <li><img src="../../img/carroceria/silo.png" class="img\_carroceria" alt="Silo" id="carroceria\_12" /></li>  
 33: <li><img src="../../img/carroceria/sider.png" class="img\_carroceria" alt="Sider" id="carroceria\_11" /></li>  
 34: <li><img src="../../img/carroceria/tanque.png" class="img\_carroceria" alt="Tanque" id="carroceria\_13" /></li>  
 35: </div>  
 36: <div>  
 37: <li><img src="../../img/carroceria/basculante.png" class="img\_carroceria" alt="Basculante" id="carroceria\_2" /></li>  
 38: <li><img src="../../img/carroceria/bebidas.png" class="img\_carroceria" alt="Bebidas" id="carroceria\_5" /></li>  
 39: <li><img src="../../img/carroceria/canavieiro.png" class="img\_carroceria" alt="Canavieiro" id="carroceria\_6" /></li>  
 40: <li><img src="../../img/carroceria/florestal.png" class="img\_carroceria" alt="Florestal" id="carroceria\_8" /></li>  
 41: <li><img src="../../img/carroceria/outros.png" class="img\_carroceria" alt="Outros" id="carroceria\_14" /></li>  
 42: </div>  
 43: </ol>  
 44: </section>  
 45:  
 46: <fieldset class="ui-grid-a escondido" id="grupo\_tipo\_conteiner\_carga">  
 47: <legend class="ui-block-a">Porta-Conteiner:</legend>  
 48: <div class="ui-block-b">  
 49: <select name="tipo\_conteiner\_carga" data-theme="a" id="tipo\_conteiner\_carga">  
 50: </select>  
 51: </div>  
 52: </fieldset>  
 53:  
 54: <fieldset class="ui-grid-a escondido" id="grupo\_placa\_estrangeira\_carga">  
 55: <legend class="ui-block-a">Placa estrangeira:</legend>  
 56: <div data-role="controlgroup" data-theme="a" data-type="horizontal" class="ui-block-b">  
 57: <input type="radio" name="placa\_estrangeira\_carga" id="placa\_estrangeira\_carga\_sim" value="sim" />  
 58: <label for="placa\_estrangeira\_carga\_sim">Sim</label>  
 59: <input type="radio" name="placa\_estrangeira\_carga" id="placa\_estrangeira\_carga\_nao" value="nao" />  
 60: <label for="placa\_estrangeira\_carga\_nao">Não</label>  
 61: </div>  
 62: </fieldset>  
 63:  
 64: <fieldset class="ui-grid-a escondido" id="grupo\_pais\_carga">  
 65: <legend class="ui-block-a">País:</legend>  
 66: <div class="ui-block-b">  
 67: <select name="pais\_carga" data-theme="a" id="pais\_carga">  
 68: </select>  
 69: </div>  
 70: </fieldset>  
 71:  
 72: <!-- Placa estrangeira -->  
 73: <fieldset class="ui-grid-a escondido" id="grupo\_placa\_unica\_carga">  
 74: <legend class="ui-block-a">Placa do veículo:</legend>  
 75: <div class="ui-block-b">  
 76: <input id="placa\_unica\_carga" class="input\_placa" type="text" name="placa\_unica\_carga" value="" data-theme="a" />  
 77: </div>  
 78: </fieldset>  
 79:  
 80: <fieldset class="ui-grid-a escondido" id="grupo\_placa\_carga">  
 81: <legend class="ui-block-a">Placa do veículo:</legend>  
 82: <div class="ui-block-b ui-grid-a" id="grupo\_placa\_dados\_carga">  
 83: <div class="ui-block-a placa\_prefixo">  
 84: <input id="placa\_letras\_carga" class="input\_placa" type="text" name="placa\_letras\_carga" value="" data-theme="a"  
 85: maxlength="3" mask="a{3}" />  
 86: </div>  
 87: <div class="ui-block-b ui-grid-a placa\_sufixo escondido" id="grupo\_placa\_numeros\_carga">  
 88: <span class="ui-block-a">-</span>  
 89: <div class="ui-block-b">  
 90: <input id="placa\_numeros\_carga" type="text" pattern="\d\*" name="placa\_numeros\_carga" value="" data-theme="a" maxlength="4"  
 91: mask="9{4}" />  
 92: </div>  
 93: </div>  
 94: </div>  
 95: <fieldset class="ui-block-b ui-grid-a escondido" id="grupo\_placa\_vermelha\_carga">  
 96: <legend class="ui-block-a">Placa vermelha:</legend>  
 97: <div class="ui-block-b ui-grid-a" id="grupo\_placa\_vermelha\_dados\_carga">  
 98: <fieldset data-role="controlgroup" data-theme="a" data-type="horizontal">  
 99: <input type="radio" name="placa\_vermelha\_carga" class="ui-block-a" id="placa\_vermelha\_carga\_sim" value="sim">  
 100: <label for="placa\_vermelha\_carga\_sim">Sim</label>  
 101: <input type="radio" name="placa\_vermelha\_carga" class="ui-block-b" id="placa\_vermelha\_carga\_nao" value="nao">  
 102: <label for="placa\_vermelha\_carga\_nao">Não</label>  
 103: </fieldset>  
 104: </div>  
 105: </fieldset>  
 106: </fieldset>  
 107:  
 108:  
 109: <fieldset class="ui-grid-a escondido" id="grupo\_qr\_code\_carga">  
 110: <legend class="ui-block-a">Possui QR Code:</legend>  
 111: <div class="ui-block-b ui-grid-a" id="grupo\_qr\_code\_dados\_carga">  
 112: <fieldset data-role="controlgroup" data-theme="a" class="ui-block-a qrCode\_group" data-type="horizontal">  
 113: <input type="radio" name="qr\_code\_carga" id="qr\_code\_carga\_sim" value="sim">  
 114: <label for="qr\_code\_carga\_sim">Sim</label>  
 115: <input type="radio" name="qr\_code\_carga" id="qr\_code\_carga\_nao" value="nao">  
 116: <label for="qr\_code\_carga\_nao">Não</label>  
 117: </fieldset>  
 118: <fieldset class="ui-block-b ui-grid-b escondido" id="grupo\_placa\_vermelha\_rntrc\_carga">  
 119: <legend class="ui-block-a">RNTRC:</legend>  
 120: <div class="ui-block-b" id="grupo\_placa\_vermelha\_rntrc\_sel\_carga">  
 121: <select name="placa\_vermelha\_rntrc\_sel\_carga" id="placa\_vermelha\_rntrc\_sel\_carga" data-theme="a">  
 122: </select>  
 123: </div>  
 124: <div class="ui-block-c escondido" id="grupo\_placa\_vermelha\_rntrc\_num\_carga">  
 125: <input id="placa\_vermelha\_rntrc\_num\_carga" type="text" pattern="\d\*" name="placa\_vermelha\_rntrc\_num\_carga" value=""  
 126: data-theme="a" mask="9{9}" />  
 127: </div>  
 128: </fieldset>  
 129: </div>  
 130: </fieldset>  
 131:  
 132: <fieldset class="ui-grid-a escondido" id="grupo\_carga\_perigosa\_carga">  
 133: <legend class="ui-block-a">Carga perigosa:</legend>  
 134: <div class="ui-block-b ui-grid-a" id="grupo\_carga\_perigosa\_dados\_carga">  
 135: <fieldset data-role="controlgroup" data-theme="a" class="ui-block-a" data-type="horizontal">  
 136: <input type="radio" name="carga\_perigosa\_carga" id="carga\_perigosa\_carga\_sim" value="sim">  
 137: <label for="carga\_perigosa\_carga\_sim">Sim</label>  
 138: <input type="radio" name="carga\_perigosa\_carga" id="carga\_perigosa\_carga\_nao" value="nao">  
 139: <label for="carga\_perigosa\_carga\_nao">Não</label>  
 140: </fieldset>  
 141: <fieldset class="ui-block-b ui-grid-a escondido" id="grupo\_carga\_perigosa\_numeros\_carga">  
 142: <div class="ui-block-a" id="grupo\_carga\_perigosa\_risco\_carga">  
 143: <input id="carga\_perigosa\_risco\_carga" type="text" name="carga\_perigosa\_risco\_carga" placeholder="Número de Risco"  
 144: data-theme="a" maxlength="4" mask="9{\*}" />  
 145: </div>  
 146: <div class="ui-block-b escondido" id="grupo\_carga\_perigosa\_onu\_carga">  
 147: <input id="carga\_perigosa\_onu\_carga" type="text" pattern="\d\*" name="carga\_perigosa\_onu\_carga" placeholder="Número da ONU"  
 148: data-theme="a" maxlength="4" mask="9{\*}"/>  
 149: </div>  
 150: </fieldset>  
 151: </div>  
 152: </fieldset>  
 153:  
 154: <nav class="ui-grid-a avancar escondido" id="grupo\_identificacao\_visual\_carga\_avancar">  
 155: <div class="ui-block-a">&nbsp;</div>  
 156: <div class="ui-block-b">  
 157: <button class="ui-btn ui-btn-a ui-corner-all ui-icon-arrow-r ui-btn-icon-right" id="identificacao\_visual\_carga\_avancar">Avançar</button>  
 158: </div>  
 159: </nav>  
 160:  
 161: </section>  
 162:  
 163: <!-- footer -->  
 164: <footer data-role='footer' data-position='fixed'>   
 165: <span class='versao'>Versão: <span id='versao'></span></span>   
 166: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 167: <span class='posto'>Posto: <span id='posto'></span></span>   
 168: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 169: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 170: </footer>  
 171:  
 172:</section>

## .\mobile\www\views\onibus\caracterizacao\textunderscore onibus.html

1:<section data-role="page" id="caracterizacao\_onibus" data-transition="slide" data-theme="b">  
 2:  
 3: <!-- header -->  
 4: <header data-role="header">  
 5: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 6: <h1>3. Caracterização do veículo</h1>  
 7: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 8: </header>  
 9:  
 10: <!--content -->  
 11: <section role="main" class="ui-content">  
 12:  
 13: <fieldset class="ui-grid-a" id="grupo\_ano\_onibus">  
 14: <legend class="ui-block-a">Ano de fabricação:</legend>  
 15: <div class="ui-block-b">  
 16: <select name="ano\_onibus" id="ano\_onibus" data-theme="a">  
 17: </select>  
 18: </div>  
 19: </fieldset>  
 20:  
 21: <fieldset class="ui-grid-a escondido" id="grupo\_propriedade\_onibus">  
 22: <legend class="ui-block-a">Propriedade:</legend>  
 23: <div class="ui-block-b">  
 24: <select name="propriedade\_onibus" id="propriedade\_onibus" data-theme="a">  
 25: </select>  
 26: </div>  
 27: </fieldset>  
 28:  
 29: <nav class="ui-grid-a avancar escondido" id="grupo\_caracterizacao\_onibus\_avancar">  
 30: <div class="ui-block-a">&nbsp;</div>  
 31: <div class="ui-block-b">  
 32: <button class="ui-btn ui-corner-all ui-icon-arrow-r ui-btn-icon-right ui-btn-a" id="caracterizacao\_onibus\_avancar">Avançar</button>  
 33: </div>  
 34: </nav>  
 35:  
 36: </section>  
 37:  
 38: <!-- footer -->  
 39: <footer data-role='footer' data-position='fixed'>   
 40: <span class='versao'>Versão: <span id='versao'></span></span>   
 41: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 42: <span class='posto'>Posto: <span id='posto'></span></span>   
 43: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 44: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 45: </footer>  
 46:  
 47:</section>

## .\mobile\www\views\onibus\caracterizacao\textunderscore viagem\textunderscore onibus.html

1:<section data-role="page" id="caracterizacao\_viagem\_onibus" data-transition="slide" data-theme="b">  
 2:  
 3: <!-- header -->  
 4: <header data-role="header">  
 5: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 6: <h1>4. Caracterização da viagem</h1>  
 7: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 8: </header>  
 9:  
 10: <!--content -->  
 11: <section role="main" class="ui-content">  
 12:  
 13: <fieldset class="ui-grid-a" id="grupo\_origem\_onibus">  
 14: <legend class="ui-block-a">Origem da viagem:</legend>  
 15: <div class="ui-block-b ui-grid-b" id="grupo\_origem\_onibus">  
 16: <div class="ui-block-a sel\_pais" id="grupo\_origem\_pais\_onibus">  
 17: <select name="origem\_pais\_onibus" id="origem\_pais\_onibus" data-theme="a">  
 18: </select>  
 19: </div>  
 20: <div class="ui-block-b sel\_uf" id="grupo\_origem\_uf\_onibus">  
 21: <select name="origem\_uf\_onibus" id="origem\_uf\_onibus" data-theme="a">  
 22: </select>  
 23: </div>  
 24: <div class="ui-block-c sel\_municipio escondido" id="grupo\_origem\_municipio\_onibus">  
 25: <input data-type='search' type="text" name="origem\_municipio\_onibus" id="origem\_municipio\_onibus" data-theme="a"  
 26: readonly="readonly">  
 27: </div>  
 28: </div>  
 29: </fieldset>  
 30:  
 31: <fieldset class="ui-grid-a escondido" id="grupo\_destino\_onibus">  
 32: <legend class="ui-block-a">Destino da viagem:</legend>  
 33: <div class="ui-block-b ui-grid-b" id="grupo\_destino\_onibus">  
 34: <div class="ui-block-a sel\_pais" id="grupo\_destino\_pais\_onibus">  
 35: <select name="destino\_pais\_onibus" id="destino\_pais\_onibus" data-theme="a">  
 36: </select>  
 37: </div>  
 38: <div class="ui-block-b sel\_uf" id="grupo\_destino\_uf\_onibus">  
 39: <select name="destino\_uf\_onibus" id="destino\_uf\_onibus" data-theme="a">  
 40: </select>  
 41: </div>  
 42: <div class="ui-block-c sel\_municipio escondido" id="grupo\_destino\_municipio\_onibus">  
 43: <input data-type='search' type="text" name="destino\_municipio\_onibus" id="destino\_municipio\_onibus" data-theme="a"  
 44: readonly="readonly">  
 45: </div>  
 46: </div>  
 47: </fieldset>  
 48:  
 49: <fieldset class="ui-grid-a escondido" id="grupo\_frequencia\_onibus">  
 50: <legend class="ui-block-a">Frequência da viagem:</legend>  
 51: <div class="ui-block-b ui-grid-b">  
 52: <div class="ui-block-a">  
 53: <input id="frequencia\_num\_onibus" type="text" pattern="\d\*" typeMask="integer" name="frequencia\_num\_onibus" value="" min="1"  
 54: data-theme="a" />  
 55: </div>  
 56: <label class="ui-block-b center" for="frequencia\_sel\_onibus">por</label>  
 57: <div class="ui-block-c">  
 58: <select name="frequencia\_sel\_onibus" id="frequencia\_sel\_onibus" data-theme="a">  
 59: </select>  
 60: </div>  
 61: </div>  
 62: </fieldset>  
 63:  
 64: <fieldset class="ui-grid-a escondido" id="grupo\_tipo\_viagem\_onibus">  
 65: <legend class="ui-block-a">Tipo de viagem/serviço:</legend>  
 66: <div class="ui-block-b">  
 67: <select name="tipo\_viagem\_onibus" id="tipo\_viagem\_onibus" data-theme="a">  
 68: </select>  
 69: </div>  
 70: </fieldset>  
 71:  
 72: <fieldset class="ui-grid-a escondido" id="grupo\_pessoas\_onibus">  
 73: <legend class="ui-block-a">Pessoas no veículo:</legend>  
 74: <div class="ui-block-b ui-grid-b">  
 75: <div class="ui-block-a">  
 76: <input id="pessoas\_onibus" type="text" pattern="\d\*" typeMask="integer" name="pessoas\_onibus" value="" data-theme="a" />  
 77: </div>  
 78: <div class="ui-block-b"></div>  
 79: <div class="ui-block-c"></div>  
 80: </div>  
 81: </fieldset>  
 82:  
 83: <nav class="ui-grid-a avancar escondido" id="grupo\_caracterizacao\_viagem\_onibus\_avancar">  
 84: <div class="ui-block-a">&nbsp;</div>  
 85: <div class="ui-block-b">  
 86: <button class="ui-btn ui-corner-all ui-icon-arrow-r ui-btn-icon-right ui-btn-a" id="caracterizacao\_viagem\_onibus\_avancar">Finalizar</button>  
 87: </div>  
 88: </nav>  
 89:  
 90: </section>  
 91:  
 92: <!-- footer -->  
 93: <footer data-role='footer' data-position='fixed'>   
 94: <span class='versao'>Versão: <span id='versao'></span></span>   
 95: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 96: <span class='posto'>Posto: <span id='posto'></span></span>   
 97: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 98: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 99: </footer>  
 100:  
 101:</section>

## .\mobile\www\views\onibus\identificacao\textunderscore visual.html

1:<section data-role="page" id="identificacao\_visual\_onibus" data-transition="slide" data-theme="b">  
 2: <!-- header -->  
 3: <header data-role="header">  
 4: <a href="#panel\_menu" data-icon="gear" class="ui-btn-left">Controles</a>  
 5: <h1>2. Identificação visual do veículo</h1>  
 6: <button id="btn\_cancelar" name="btn\_cancelar" data-icon="delete" class="ui-btn-right">Cancelar</button>  
 7: </header>  
 8:  
 9: <!--content -->  
 10: <section role="main" class="ui-content">  
 11:  
 12: <fieldset class="ui-grid-a" id="grupo\_placa\_estrangeira\_onibus">  
 13: <legend class="ui-block-a">Placa estrangeira:</legend>  
 14: <div data-role="controlgroup" data-theme="a" data-type="horizontal" class="ui-block-b">  
 15: <input type="radio" name="placa\_estrangeira\_onibus" id="placa\_estrangeira\_onibus\_sim" value="sim" />  
 16: <label for="placa\_estrangeira\_onibus\_sim">Sim</label>  
 17: <input type="radio" name="placa\_estrangeira\_onibus" id="placa\_estrangeira\_onibus\_nao" value="nao" />  
 18: <label for="placa\_estrangeira\_onibus\_nao">Não</label>  
 19: </div>  
 20: </fieldset>  
 21:  
 22: <fieldset class="ui-grid-a escondido" id="grupo\_pais\_onibus">  
 23: <legend class="ui-block-a">País:</legend>  
 24: <div class="ui-block-b">  
 25: <select name="pais\_onibus" data-theme="a" id="pais\_onibus">  
 26: </select>  
 27: </div>  
 28: </fieldset>  
 29:  
 30: <fieldset class="ui-grid-a escondido" id="grupo\_placa\_onibus">  
 31: <legend class="ui-block-a">Placa do veículo:</legend>  
 32: <div class="ui-block-b ui-grid-a" id="grupo\_placa\_dados\_onibus">  
 33: <div class="ui-block-a placa\_prefixo">  
 34: <input id="placa\_letras\_onibus" class="input\_placa" type="text" name="placa\_letras\_onibus" value="" data-theme="a"  
 35: maxlength="3" mask="a{3}" />  
 36: </div>  
 37: <div class="ui-block-b ui-grid-a placa\_sufixo escondido" id="grupo\_placa\_numeros\_onibus">  
 38: <span class="ui-block-a">-</span>  
 39: <div class="ui-block-b">  
 40: <input id="placa\_numeros\_onibus" type="text" pattern="\d\*" name="placa\_numeros\_onibus" value="" data-theme="a"  
 41: maxlength="4" mask="9{4}" />  
 42: </div>  
 43: </div>  
 44: </div>  
 45: </fieldset>  
 46:  
 47: <fieldset class="ui-grid-a escondido" id="grupo\_placa\_unica\_onibus">  
 48: <legend class="ui-block-a">Placa do veículo:</legend>  
 49: <div class="ui-block-b">  
 50: <input id="placa\_unica\_onibus" class="input\_placa" type="text" name="placa\_unica\_onibus" value="" data-theme="a" />  
 51: </div>  
 52: </fieldset>  
 53:  
 54: <nav class="ui-grid-a avancar escondido" id="grupo\_identificacao\_visual\_onibus\_avancar">  
 55: <div class="ui-block-a">&nbsp;</div>  
 56: <div class="ui-block-b">  
 57: <button class="ui-btn ui-btn-a ui-corner-all ui-icon-arrow-r ui-btn-icon-right" id="identificacao\_visual\_onibus\_avancar">Avançar</button>  
 58: </div>  
 59: </nav>  
 60:  
 61: </section>  
 62:  
 63: <!-- footer -->  
 64: <footer data-role='footer' data-position='fixed'>   
 65: <span class='versao'>Versão: <span id='versao'></span></span>   
 66: <span class='data\_atual'>Data: <span id='data\_atual'></span></span>   
 67: <span class='posto'>Posto: <span id='posto'></span></span>   
 68: <span class='sentido'>Sentido: <span id='sentido'></span></span>   
 69: <span class='ipad'>id-iPad: <span id='ipadID'></span></span>   
 70: </footer>  
 71:  
 72:</section>

## .\document.py

1:# -\*- coding: utf-8 -\*-  
 2:"""  
 3:Created on Wed May 30 16:21:28 2018  
 4:  
 5:@author: Geraldo  
 6:"""  
 7:import os  
 8:  
 9:# começa definindo as constantes  
 10:  
 11:rootdir = "."  
 12:  
 13:NUMERA = True  
 14:  
 15:HEADER = """  
 16:\\documentclass{article}  
 17:\\usepackage[utf8]{inputenc}  
 18:\\usepackage[T1]{fontenc}  
 19:\\usepackage{lmodern}  
 20:  
 21:  
 22:\\begin{document}  
 23:"""  
 24:  
 25:FOOTER = """  
 26:\\end{document}  
 27:"""  
 28:FILE\_HEADER = """  
 29:\\subsection{%s}  
 30:  
 31:\\begin{verbatim}  
 32:  
 33:"""  
 34:FILE\_FOOTER = """  
 35:\\end{verbatim}  
 36:"""  
 37:  
 38:  
 39:# define os arquivos que vão ser usados  
 40:  
 41:arquivos = [ "java", "js", "json" , "css", "html", "py","xml","properties",  
 42: "txt"]  
 43:nome = { "java" : "Java",  
 44: "js" : "JavaScript",  
 45: "json" : "JSON",  
 46: "css" : "Cascade Style Sheets (CSS)",  
 47: "html" : "HTML",  
 48: "py" : "Python",  
 49: "xml" : "XML",  
 50: "properties" : "properties",  
 51: "txt": "txt"  
 52: }  
 53:  
 54:EXCLUDE = ["node\_modules","doc","lib","doxy","resources","images","simple",  
 55: ".settings",".external","concetrador\logs","concentrador\store",  
 56: "ignore","plugins"]  
 57:  
 58:  
 59:# Como queremos ordenar por tipo de arquivo  
 60:  
 61:def doublein(string,lista):  
 62: for string1 in lista:  
 63: if string1 in string:  
 64: return True  
 65: return False  
 66:  
 67:# problemas no Latex do pandoc  
 68:def trata(nome):  
 69: x=nome.replace("\_","\\textunderscore ")  
 70: return x.replace("\\","\\backslash ")  
 71:  
 72:  
 73:  
 74:with open('output.latex','w',encoding="utf8") as fout:  
 75: fout.writelines(HEADER)  
 76: for extensao in arquivos:  
 77: print("EXTENSAO: ",extensao)  
 78: for root, subFolders, files in os.walk(rootdir):  
 79:   
 80: for name in files:  
 81:   
 82: completo = os.path.join(root,name)  
 83: \_,extatual = os.path.splitext(completo)  
 84:   
 85: if doublein(root,EXCLUDE):  
 86: continue  
 87:   
 88: if extatual == "."+extensao :  
 89: fout.writelines(FILE\_HEADER % trata(completo))  
 90: print(trata(completo))  
 91: try:  
 92: linha = 0  
 93: with open(completo, 'r',encoding="utf8") as fin:  
 94: for lines in fin:  
 95: # defende de UTF-8 errados  
 96: line=bytes(lines,'utf-8').decode('utf-8','ignore')  
 97: linha+=1  
 98: if NUMERA:  
 99: line="%4d:" % linha + line  
 100: fout.writelines(line)  
 101: except:  
 102: print("arquivo de entrada: ",completo)  
 103: raise  
 104: fout.writelines(FILE\_FOOTER)  
 105: fout.writelines(FOOTER)  
 106:   
 107:  
 108:   
 109:

## .\concentrador\src\logins\verifica\textunderscore fase\textunderscore 4.py

1:# -\*- coding: utf-8 -\*-  
 2:"""  
 3:Created on Sun Sep 10 18:14:51 2017  
 4:  
 5:@author: mangeli  
 6:usar python 3  
 7:"""  
 8:  
 9:import json, csv  
 10:  
 11:with open("users\_fase2\_old.json", 'r') as f:  
 12: logins = json.load(f)  
 13:   
 14:f = open("POSTOS\_FASE\_4.csv")  
 15:postosFase4 = csv.reader(f, delimiter='\t')  
 16:  
 17:#testa se os postos tem senha  
 18:#for row in postosFase4:  
 19:# achei = False  
 20:# for login in logins['logins']:  
 21:# if int(login['usr']) == int(row[0]):  
 22:# achei = True  
 23:# if not achei:  
 24:# print 'posto', row[0], 'nao tem senha'  
 25:  
 26:''' todos os postos da fase 4 já estão no sistema '''  
 27:  
 28:  
 29:  
 30:#modifica a descrição para a constante na planilha  
 31:for row in postosFase4:  
 32: for login in logins['logins']:  
 33: if int(login['usr']) == int(row[0]):  
 34: print (row[0], row[14], "\n", login['usr'], login['trecho'])  
 35: print ('\n============')  
 36: login['trecho'] = row[14]  
 37:  
 38:f.close() #fecha planilha  
 39:  
 40:with open("users.json", 'w') as f:  
 41: json.dump(logins,f,indent=4, ensure\_ascii=False)  
 42:

## .\concentrador\build.xml

1:<?xml version="1.0" encoding="UTF-8"?>  
 2:<!-- You may freely edit this file. See commented blocks below for -->  
 3:<!-- some examples of how to customize the build. -->  
 4:<!-- (If you delete it and reopen the project it will be recreated.) -->  
 5:<!-- By default, only the Clean and Build commands use this build script. -->  
 6:<!-- Commands such as Run, Debug, and Test only use this build script if -->  
 7:<!-- the Compile on Save feature is turned off for the project. -->  
 8:<!-- You can turn off the Compile on Save (or Deploy on Save) setting -->  
 9:<!-- in the project's Project Properties dialog box.-->  
 10:<project name="concentrador" default="default" basedir=".">  
 11: <description>Builds, tests, and runs the project concentrador.</description>  
 12: <import file="nbproject/build-impl.xml"/>  
 13: <!--  
 14:  
 15: There exist several targets which are by default empty and which can be   
 16: used for execution of your tasks. These targets are usually executed   
 17: before and after some main targets. They are:   
 18:  
 19: -pre-init: called before initialization of project properties  
 20: -post-init: called after initialization of project properties  
 21: -pre-compile: called before javac compilation  
 22: -post-compile: called after javac compilation  
 23: -pre-compile-single: called before javac compilation of single file  
 24: -post-compile-single: called after javac compilation of single file  
 25: -pre-compile-test: called before javac compilation of JUnit tests  
 26: -post-compile-test: called after javac compilation of JUnit tests  
 27: -pre-compile-test-single: called before javac compilation of single JUnit test  
 28: -post-compile-test-single: called after javac compilation of single JUunit test  
 29: -pre-jar: called before JAR building  
 30: -post-jar: called after JAR building  
 31: -post-clean: called after cleaning build products  
 32:  
 33: (Targets beginning with '-' are not intended to be called on their own.)  
 34:  
 35: Example of inserting an obfuscator after compilation could look like this:  
 36:  
 37: <target name="-post-compile">  
 38: <obfuscate>  
 39: <fileset dir="${build.classes.dir}"/>  
 40: </obfuscate>  
 41: </target>  
 42:  
 43: For list of available properties check the imported   
 44: nbproject/build-impl.xml file.   
 45:  
 46:  
 47: Another way to customize the build is by overriding existing main targets.  
 48: The targets of interest are:   
 49:  
 50: -init-macrodef-javac: defines macro for javac compilation  
 51: -init-macrodef-junit: defines macro for junit execution  
 52: -init-macrodef-debug: defines macro for class debugging  
 53: -init-macrodef-java: defines macro for class execution  
 54: -do-jar: JAR building  
 55: run: execution of project   
 56: -javadoc-build: Javadoc generation  
 57: test-report: JUnit report generation  
 58:  
 59: An example of overriding the target for project execution could look like this:  
 60:  
 61: <target name="run" depends="concentrador-impl.jar">  
 62: <exec dir="bin" executable="launcher.exe">  
 63: <arg file="${dist.jar}"/>  
 64: </exec>  
 65: </target>  
 66:  
 67: Notice that the overridden target depends on the jar target and not only on   
 68: the compile target as the regular run target does. Again, for a list of available   
 69: properties which you can use, check the target you are overriding in the  
 70: nbproject/build-impl.xml file.   
 71:  
 72: -->  
 73: <target name="package-for-store" depends="jar">  
 74: <property name="store.jar.name" value="concentrador"/>  
 75: <property name="store.dir" value="store"/>  
 76: <property name="store.jar" value="${store.dir}/${store.jar.name}.jar"/>  
 77: <echo message="Packaging ${application.title} into a single JAR at ${store.jar}"/>  
 78: <delete dir="${store.dir}"/>  
 79: <mkdir dir="${store.dir}"/>  
 80: <jar destfile="${store.dir}/temp\_final.jar" filesetmanifest="skip">  
 81: <zipgroupfileset dir="dist" includes="\*.jar"/>  
 82: <zipgroupfileset dir="dist/lib" includes="\*.jar"/>  
 83: <manifest>  
 84: <attribute name="Main-Class" value="${main.class}"/>  
 85: </manifest>  
 86: </jar>  
 87: <zip destfile="${store.jar}">  
 88: <zipfileset src="${store.dir}/temp\_final.jar"  
 89: excludes="META-INF/\*.SF, META-INF/\*.DSA, META-INF/\*.RSA"/>  
 90: </zip>  
 91: <delete file="${store.dir}/temp\_final.jar"/>  
 92:</target>  
 93:</project>

## .\concentrador\nbproject\build-impl.xml

1:<?xml version="1.0" encoding="UTF-8"?>  
 2:<!--  
 3:\*\*\* GENERATED FROM project.xml - DO NOT EDIT \*\*\*  
 4:\*\*\* EDIT ../build.xml INSTEAD \*\*\*  
 5:  
 6:For the purpose of easier reading the script  
 7:is divided into following sections:  
 8:  
 9: - initialization  
 10: - compilation  
 11: - jar  
 12: - execution  
 13: - debugging  
 14: - javadoc  
 15: - test compilation  
 16: - test execution  
 17: - test debugging  
 18: - applet  
 19: - cleanup  
 20:  
 21: -->  
 22:<project xmlns:j2seproject1="http://www.netbeans.org/ns/j2se-project/1" xmlns:j2seproject3="http://www.netbeans.org/ns/j2se-project/3" xmlns:jaxrpc="http://www.netbeans.org/ns/j2se-project/jax-rpc" basedir=".." default="default" name="concentrador-impl">  
 23: <fail message="Please build using Ant 1.8.0 or higher.">  
 24: <condition>  
 25: <not>  
 26: <antversion atleast="1.8.0"/>  
 27: </not>  
 28: </condition>  
 29: </fail>  
 30: <target depends="test,jar,javadoc" description="Build and test whole project." name="default"/>  
 31: <!--   
 32: ======================  
 33: INITIALIZATION SECTION   
 34: ======================  
 35: -->  
 36: <target name="-pre-init">  
 37: <!-- Empty placeholder for easier customization. -->  
 38: <!-- You can override this target in the ../build.xml file. -->  
 39: </target>  
 40: <target depends="-pre-init" name="-init-private">  
 41: <property file="nbproject/private/config.properties"/>  
 42: <property file="nbproject/private/configs/${config}.properties"/>  
 43: <property file="nbproject/private/private.properties"/>  
 44: </target>  
 45: <target depends="-pre-init,-init-private" name="-init-user">  
 46: <property file="${user.properties.file}"/>  
 47: <!-- The two properties below are usually overridden -->  
 48: <!-- by the active platform. Just a fallback. -->  
 49: <property name="default.javac.source" value="1.4"/>  
 50: <property name="default.javac.target" value="1.4"/>  
 51: </target>  
 52: <target depends="-pre-init,-init-private,-init-user" name="-init-project">  
 53: <property file="nbproject/configs/${config}.properties"/>  
 54: <property file="nbproject/project.properties"/>  
 55: </target>  
 56: <target depends="-pre-init,-init-private,-init-user,-init-project,-init-macrodef-property" name="-do-init">  
 57: <property name="platform.java" value="${java.home}/bin/java"/>  
 58: <available file="${manifest.file}" property="manifest.available"/>  
 59: <condition property="splashscreen.available">  
 60: <and>  
 61: <not>  
 62: <equals arg1="${application.splash}" arg2="" trim="true"/>  
 63: </not>  
 64: <available file="${application.splash}"/>  
 65: </and>  
 66: </condition>  
 67: <condition property="main.class.available">  
 68: <and>  
 69: <isset property="main.class"/>  
 70: <not>  
 71: <equals arg1="${main.class}" arg2="" trim="true"/>  
 72: </not>  
 73: </and>  
 74: </condition>  
 75: <condition property="profile.available">  
 76: <and>  
 77: <isset property="javac.profile"/>  
 78: <length length="0" string="${javac.profile}" when="greater"/>  
 79: <matches pattern="1\.[89](\..\*)?" string="${javac.source}"/>  
 80: </and>  
 81: </condition>  
 82: <condition property="do.archive">  
 83: <or>  
 84: <not>  
 85: <istrue value="${jar.archive.disabled}"/>  
 86: </not>  
 87: <istrue value="${not.archive.disabled}"/>  
 88: </or>  
 89: </condition>  
 90: <condition property="do.mkdist">  
 91: <and>  
 92: <isset property="do.archive"/>  
 93: <isset property="libs.CopyLibs.classpath"/>  
 94: <not>  
 95: <istrue value="${mkdist.disabled}"/>  
 96: </not>  
 97: </and>  
 98: </condition>  
 99: <condition property="do.archive+manifest.available">  
 100: <and>  
 101: <isset property="manifest.available"/>  
 102: <istrue value="${do.archive}"/>  
 103: </and>  
 104: </condition>  
 105: <condition property="do.archive+main.class.available">  
 106: <and>  
 107: <isset property="main.class.available"/>  
 108: <istrue value="${do.archive}"/>  
 109: </and>  
 110: </condition>  
 111: <condition property="do.archive+splashscreen.available">  
 112: <and>  
 113: <isset property="splashscreen.available"/>  
 114: <istrue value="${do.archive}"/>  
 115: </and>  
 116: </condition>  
 117: <condition property="do.archive+profile.available">  
 118: <and>  
 119: <isset property="profile.available"/>  
 120: <istrue value="${do.archive}"/>  
 121: </and>  
 122: </condition>  
 123: <condition property="have.tests">  
 124: <or>  
 125: <available file="${test.src.dir}"/>  
 126: </or>  
 127: </condition>  
 128: <condition property="have.sources">  
 129: <or>  
 130: <available file="${src.dir}"/>  
 131: <available file="${src.resources.dir}"/>  
 132: </or>  
 133: </condition>  
 134: <condition property="netbeans.home+have.tests">  
 135: <and>  
 136: <isset property="netbeans.home"/>  
 137: <isset property="have.tests"/>  
 138: </and>  
 139: </condition>  
 140: <condition property="no.javadoc.preview">  
 141: <and>  
 142: <isset property="javadoc.preview"/>  
 143: <isfalse value="${javadoc.preview}"/>  
 144: </and>  
 145: </condition>  
 146: <property name="run.jvmargs" value=""/>  
 147: <property name="run.jvmargs.ide" value=""/>  
 148: <property name="javac.compilerargs" value=""/>  
 149: <property name="work.dir" value="${basedir}"/>  
 150: <condition property="no.deps">  
 151: <and>  
 152: <istrue value="${no.dependencies}"/>  
 153: </and>  
 154: </condition>  
 155: <property name="javac.debug" value="true"/>  
 156: <property name="javadoc.preview" value="true"/>  
 157: <property name="application.args" value=""/>  
 158: <property name="source.encoding" value="${file.encoding}"/>  
 159: <property name="runtime.encoding" value="${source.encoding}"/>  
 160: <condition property="javadoc.encoding.used" value="${javadoc.encoding}">  
 161: <and>  
 162: <isset property="javadoc.encoding"/>  
 163: <not>  
 164: <equals arg1="${javadoc.encoding}" arg2=""/>  
 165: </not>  
 166: </and>  
 167: </condition>  
 168: <property name="javadoc.encoding.used" value="${source.encoding}"/>  
 169: <property name="includes" value="\*\*"/>  
 170: <property name="excludes" value=""/>  
 171: <property name="do.depend" value="false"/>  
 172: <condition property="do.depend.true">  
 173: <istrue value="${do.depend}"/>  
 174: </condition>  
 175: <path id="endorsed.classpath.path" path="${endorsed.classpath}"/>  
 176: <condition else="" property="endorsed.classpath.cmd.line.arg" value="-Xbootclasspath/p:'${toString:endorsed.classpath.path}'">  
 177: <and>  
 178: <isset property="endorsed.classpath"/>  
 179: <not>  
 180: <equals arg1="${endorsed.classpath}" arg2="" trim="true"/>  
 181: </not>  
 182: </and>  
 183: </condition>  
 184: <condition else="" property="javac.profile.cmd.line.arg" value="-profile ${javac.profile}">  
 185: <isset property="profile.available"/>  
 186: </condition>  
 187: <condition else="false" property="jdkBug6558476">  
 188: <and>  
 189: <matches pattern="1\.[56]" string="${java.specification.version}"/>  
 190: <not>  
 191: <os family="unix"/>  
 192: </not>  
 193: </and>  
 194: </condition>  
 195: <condition else="false" property="javac.fork">  
 196: <or>  
 197: <istrue value="${jdkBug6558476}"/>  
 198: <istrue value="${javac.external.vm}"/>  
 199: </or>  
 200: </condition>  
 201: <property name="jar.index" value="false"/>  
 202: <property name="jar.index.metainf" value="${jar.index}"/>  
 203: <property name="copylibs.rebase" value="true"/>  
 204: <available file="${meta.inf.dir}/persistence.xml" property="has.persistence.xml"/>  
 205: <condition property="junit.available">  
 206: <or>  
 207: <available classname="org.junit.Test" classpath="${run.test.classpath}"/>  
 208: <available classname="junit.framework.Test" classpath="${run.test.classpath}"/>  
 209: </or>  
 210: </condition>  
 211: <condition property="testng.available">  
 212: <available classname="org.testng.annotations.Test" classpath="${run.test.classpath}"/>  
 213: </condition>  
 214: <condition property="junit+testng.available">  
 215: <and>  
 216: <istrue value="${junit.available}"/>  
 217: <istrue value="${testng.available}"/>  
 218: </and>  
 219: </condition>  
 220: <condition else="testng" property="testng.mode" value="mixed">  
 221: <istrue value="${junit+testng.available}"/>  
 222: </condition>  
 223: <condition else="" property="testng.debug.mode" value="-mixed">  
 224: <istrue value="${junit+testng.available}"/>  
 225: </condition>  
 226: <property name="java.failonerror" value="true"/>  
 227: </target>  
 228: <target name="-post-init">  
 229: <!-- Empty placeholder for easier customization. -->  
 230: <!-- You can override this target in the ../build.xml file. -->  
 231: </target>  
 232: <target depends="-pre-init,-init-private,-init-user,-init-project,-do-init" name="-init-check">  
 233: <fail unless="src.dir">Must set src.dir</fail>  
 234: <fail unless="src.resources.dir">Must set src.resources.dir</fail>  
 235: <fail unless="test.src.dir">Must set test.src.dir</fail>  
 236: <fail unless="build.dir">Must set build.dir</fail>  
 237: <fail unless="dist.dir">Must set dist.dir</fail>  
 238: <fail unless="build.classes.dir">Must set build.classes.dir</fail>  
 239: <fail unless="dist.javadoc.dir">Must set dist.javadoc.dir</fail>  
 240: <fail unless="build.test.classes.dir">Must set build.test.classes.dir</fail>  
 241: <fail unless="build.test.results.dir">Must set build.test.results.dir</fail>  
 242: <fail unless="build.classes.excludes">Must set build.classes.excludes</fail>  
 243: <fail unless="dist.jar">Must set dist.jar</fail>  
 244: </target>  
 245: <target name="-init-macrodef-property">  
 246: <macrodef name="property" uri="http://www.netbeans.org/ns/j2se-project/1">  
 247: <attribute name="name"/>  
 248: <attribute name="value"/>  
 249: <sequential>  
 250: <property name="@{name}" value="${@{value}}"/>  
 251: </sequential>  
 252: </macrodef>  
 253: </target>  
 254: <target depends="-init-ap-cmdline-properties" if="ap.supported.internal" name="-init-macrodef-javac-with-processors">  
 255: <macrodef name="javac" uri="http://www.netbeans.org/ns/j2se-project/3">  
 256: <attribute default="${src.dir}:${src.resources.dir}" name="srcdir"/>  
 257: <attribute default="${build.classes.dir}" name="destdir"/>  
 258: <attribute default="${javac.classpath}" name="classpath"/>  
 259: <attribute default="${javac.processorpath}" name="processorpath"/>  
 260: <attribute default="${build.generated.sources.dir}/ap-source-output" name="apgeneratedsrcdir"/>  
 261: <attribute default="${includes}" name="includes"/>  
 262: <attribute default="${excludes}" name="excludes"/>  
 263: <attribute default="${javac.debug}" name="debug"/>  
 264: <attribute default="${empty.dir}" name="sourcepath"/>  
 265: <attribute default="${empty.dir}" name="gensrcdir"/>  
 266: <element name="customize" optional="true"/>  
 267: <sequential>  
 268: <property location="${build.dir}/empty" name="empty.dir"/>  
 269: <mkdir dir="${empty.dir}"/>  
 270: <mkdir dir="@{apgeneratedsrcdir}"/>  
 271: <javac debug="@{debug}" deprecation="${javac.deprecation}" destdir="@{destdir}" encoding="${source.encoding}" excludes="@{excludes}" fork="${javac.fork}" includeantruntime="false" includes="@{includes}" source="${javac.source}" sourcepath="@{sourcepath}" srcdir="@{srcdir}" target="${javac.target}" tempdir="${java.io.tmpdir}">  
 272: <src>  
 273: <dirset dir="@{gensrcdir}" erroronmissingdir="false">  
 274: <include name="\*"/>  
 275: </dirset>  
 276: </src>  
 277: <classpath>  
 278: <path path="@{classpath}"/>  
 279: </classpath>  
 280: <compilerarg line="${endorsed.classpath.cmd.line.arg}"/>  
 281: <compilerarg line="${javac.profile.cmd.line.arg}"/>  
 282: <compilerarg line="${javac.compilerargs}"/>  
 283: <compilerarg value="-processorpath"/>  
 284: <compilerarg path="@{processorpath}:${empty.dir}"/>  
 285: <compilerarg line="${ap.processors.internal}"/>  
 286: <compilerarg line="${annotation.processing.processor.options}"/>  
 287: <compilerarg value="-s"/>  
 288: <compilerarg path="@{apgeneratedsrcdir}"/>  
 289: <compilerarg line="${ap.proc.none.internal}"/>  
 290: <customize/>  
 291: </javac>  
 292: </sequential>  
 293: </macrodef>  
 294: </target>  
 295: <target depends="-init-ap-cmdline-properties" name="-init-macrodef-javac-without-processors" unless="ap.supported.internal">  
 296: <macrodef name="javac" uri="http://www.netbeans.org/ns/j2se-project/3">  
 297: <attribute default="${src.dir}:${src.resources.dir}" name="srcdir"/>  
 298: <attribute default="${build.classes.dir}" name="destdir"/>  
 299: <attribute default="${javac.classpath}" name="classpath"/>  
 300: <attribute default="${javac.processorpath}" name="processorpath"/>  
 301: <attribute default="${build.generated.sources.dir}/ap-source-output" name="apgeneratedsrcdir"/>  
 302: <attribute default="${includes}" name="includes"/>  
 303: <attribute default="${excludes}" name="excludes"/>  
 304: <attribute default="${javac.debug}" name="debug"/>  
 305: <attribute default="${empty.dir}" name="sourcepath"/>  
 306: <attribute default="${empty.dir}" name="gensrcdir"/>  
 307: <element name="customize" optional="true"/>  
 308: <sequential>  
 309: <property location="${build.dir}/empty" name="empty.dir"/>  
 310: <mkdir dir="${empty.dir}"/>  
 311: <javac debug="@{debug}" deprecation="${javac.deprecation}" destdir="@{destdir}" encoding="${source.encoding}" excludes="@{excludes}" fork="${javac.fork}" includeantruntime="false" includes="@{includes}" source="${javac.source}" sourcepath="@{sourcepath}" srcdir="@{srcdir}" target="${javac.target}" tempdir="${java.io.tmpdir}">  
 312: <src>  
 313: <dirset dir="@{gensrcdir}" erroronmissingdir="false">  
 314: <include name="\*"/>  
 315: </dirset>  
 316: </src>  
 317: <classpath>  
 318: <path path="@{classpath}"/>  
 319: </classpath>  
 320: <compilerarg line="${endorsed.classpath.cmd.line.arg}"/>  
 321: <compilerarg line="${javac.profile.cmd.line.arg}"/>  
 322: <compilerarg line="${javac.compilerargs}"/>  
 323: <customize/>  
 324: </javac>  
 325: </sequential>  
 326: </macrodef>  
 327: </target>  
 328: <target depends="-init-macrodef-javac-with-processors,-init-macrodef-javac-without-processors" name="-init-macrodef-javac">  
 329: <macrodef name="depend" uri="http://www.netbeans.org/ns/j2se-project/3">  
 330: <attribute default="${src.dir}:${src.resources.dir}" name="srcdir"/>  
 331: <attribute default="${build.classes.dir}" name="destdir"/>  
 332: <attribute default="${javac.classpath}" name="classpath"/>  
 333: <sequential>  
 334: <depend cache="${build.dir}/depcache" destdir="@{destdir}" excludes="${excludes}" includes="${includes}" srcdir="@{srcdir}">  
 335: <classpath>  
 336: <path path="@{classpath}"/>  
 337: </classpath>  
 338: </depend>  
 339: </sequential>  
 340: </macrodef>  
 341: <macrodef name="force-recompile" uri="http://www.netbeans.org/ns/j2se-project/3">  
 342: <attribute default="${build.classes.dir}" name="destdir"/>  
 343: <sequential>  
 344: <fail unless="javac.includes">Must set javac.includes</fail>  
 345: <pathconvert pathsep="${line.separator}" property="javac.includes.binary">  
 346: <path>  
 347: <filelist dir="@{destdir}" files="${javac.includes}"/>  
 348: </path>  
 349: <globmapper from="\*.java" to="\*.class"/>  
 350: </pathconvert>  
 351: <tempfile deleteonexit="true" property="javac.includesfile.binary"/>  
 352: <echo file="${javac.includesfile.binary}" message="${javac.includes.binary}"/>  
 353: <delete>  
 354: <files includesfile="${javac.includesfile.binary}"/>  
 355: </delete>  
 356: <delete>  
 357: <fileset file="${javac.includesfile.binary}"/>  
 358: </delete>  
 359: </sequential>  
 360: </macrodef>  
 361: </target>  
 362: <target if="${junit.available}" name="-init-macrodef-junit-init">  
 363: <condition else="false" property="nb.junit.batch" value="true">  
 364: <and>  
 365: <istrue value="${junit.available}"/>  
 366: <not>  
 367: <isset property="test.method"/>  
 368: </not>  
 369: </and>  
 370: </condition>  
 371: <condition else="false" property="nb.junit.single" value="true">  
 372: <and>  
 373: <istrue value="${junit.available}"/>  
 374: <isset property="test.method"/>  
 375: </and>  
 376: </condition>  
 377: </target>  
 378: <target name="-init-test-properties">  
 379: <property name="test.binaryincludes" value="&lt;nothing&gt;"/>  
 380: <property name="test.binarytestincludes" value=""/>  
 381: <property name="test.binaryexcludes" value=""/>  
 382: </target>  
 383: <target if="${nb.junit.single}" name="-init-macrodef-junit-single" unless="${nb.junit.batch}">  
 384: <macrodef name="junit" uri="http://www.netbeans.org/ns/j2se-project/3">  
 385: <attribute default="${includes}" name="includes"/>  
 386: <attribute default="${excludes}" name="excludes"/>  
 387: <attribute default="\*\*" name="testincludes"/>  
 388: <attribute default="" name="testmethods"/>  
 389: <element name="customize" optional="true"/>  
 390: <sequential>  
 391: <property name="junit.forkmode" value="perTest"/>  
 392: <junit dir="${work.dir}" errorproperty="tests.failed" failureproperty="tests.failed" fork="true" forkmode="${junit.forkmode}" showoutput="true" tempdir="${build.dir}">  
 393: <test methods="@{testmethods}" name="@{testincludes}" todir="${build.test.results.dir}"/>  
 394: <syspropertyset>  
 395: <propertyref prefix="test-sys-prop."/>  
 396: <mapper from="test-sys-prop.\*" to="\*" type="glob"/>  
 397: </syspropertyset>  
 398: <formatter type="brief" usefile="false"/>  
 399: <formatter type="xml"/>  
 400: <jvmarg value="-ea"/>  
 401: <customize/>  
 402: </junit>  
 403: </sequential>  
 404: </macrodef>  
 405: </target>  
 406: <target depends="-init-test-properties" if="${nb.junit.batch}" name="-init-macrodef-junit-batch" unless="${nb.junit.single}">  
 407: <macrodef name="junit" uri="http://www.netbeans.org/ns/j2se-project/3">  
 408: <attribute default="${includes}" name="includes"/>  
 409: <attribute default="${excludes}" name="excludes"/>  
 410: <attribute default="\*\*" name="testincludes"/>  
 411: <attribute default="" name="testmethods"/>  
 412: <element name="customize" optional="true"/>  
 413: <sequential>  
 414: <property name="junit.forkmode" value="perTest"/>  
 415: <junit dir="${work.dir}" errorproperty="tests.failed" failureproperty="tests.failed" fork="true" forkmode="${junit.forkmode}" showoutput="true" tempdir="${build.dir}">  
 416: <batchtest todir="${build.test.results.dir}">  
 417: <fileset dir="${test.src.dir}" excludes="@{excludes},${excludes}" includes="@{includes}">  
 418: <filename name="@{testincludes}"/>  
 419: </fileset>  
 420: <fileset dir="${build.test.classes.dir}" excludes="@{excludes},${excludes},${test.binaryexcludes}" includes="${test.binaryincludes}">  
 421: <filename name="${test.binarytestincludes}"/>  
 422: </fileset>  
 423: </batchtest>  
 424: <syspropertyset>  
 425: <propertyref prefix="test-sys-prop."/>  
 426: <mapper from="test-sys-prop.\*" to="\*" type="glob"/>  
 427: </syspropertyset>  
 428: <formatter type="brief" usefile="false"/>  
 429: <formatter type="xml"/>  
 430: <jvmarg value="-ea"/>  
 431: <customize/>  
 432: </junit>  
 433: </sequential>  
 434: </macrodef>  
 435: </target>  
 436: <target depends="-init-macrodef-junit-init,-init-macrodef-junit-single, -init-macrodef-junit-batch" if="${junit.available}" name="-init-macrodef-junit"/>  
 437: <target if="${testng.available}" name="-init-macrodef-testng">  
 438: <macrodef name="testng" uri="http://www.netbeans.org/ns/j2se-project/3">  
 439: <attribute default="${includes}" name="includes"/>  
 440: <attribute default="${excludes}" name="excludes"/>  
 441: <attribute default="\*\*" name="testincludes"/>  
 442: <attribute default="" name="testmethods"/>  
 443: <element name="customize" optional="true"/>  
 444: <sequential>  
 445: <condition else="" property="testng.methods.arg" value="@{testincludes}.@{testmethods}">  
 446: <isset property="test.method"/>  
 447: </condition>  
 448: <union id="test.set">  
 449: <fileset dir="${test.src.dir}" excludes="@{excludes},\*\*/\*.xml,${excludes}" includes="@{includes}">  
 450: <filename name="@{testincludes}"/>  
 451: </fileset>  
 452: </union>  
 453: <taskdef classname="org.testng.TestNGAntTask" classpath="${run.test.classpath}" name="testng"/>  
 454: <testng classfilesetref="test.set" failureProperty="tests.failed" listeners="org.testng.reporters.VerboseReporter" methods="${testng.methods.arg}" mode="${testng.mode}" outputdir="${build.test.results.dir}" suitename="concentrador" testname="TestNG tests" workingDir="${work.dir}">  
 455: <xmlfileset dir="${build.test.classes.dir}" includes="@{testincludes}"/>  
 456: <propertyset>  
 457: <propertyref prefix="test-sys-prop."/>  
 458: <mapper from="test-sys-prop.\*" to="\*" type="glob"/>  
 459: </propertyset>  
 460: <customize/>  
 461: </testng>  
 462: </sequential>  
 463: </macrodef>  
 464: </target>  
 465: <target name="-init-macrodef-test-impl">  
 466: <macrodef name="test-impl" uri="http://www.netbeans.org/ns/j2se-project/3">  
 467: <attribute default="${includes}" name="includes"/>  
 468: <attribute default="${excludes}" name="excludes"/>  
 469: <attribute default="\*\*" name="testincludes"/>  
 470: <attribute default="" name="testmethods"/>  
 471: <element implicit="true" name="customize" optional="true"/>  
 472: <sequential>  
 473: <echo>No tests executed.</echo>  
 474: </sequential>  
 475: </macrodef>  
 476: </target>  
 477: <target depends="-init-macrodef-junit" if="${junit.available}" name="-init-macrodef-junit-impl">  
 478: <macrodef name="test-impl" uri="http://www.netbeans.org/ns/j2se-project/3">  
 479: <attribute default="${includes}" name="includes"/>  
 480: <attribute default="${excludes}" name="excludes"/>  
 481: <attribute default="\*\*" name="testincludes"/>  
 482: <attribute default="" name="testmethods"/>  
 483: <element implicit="true" name="customize" optional="true"/>  
 484: <sequential>  
 485: <j2seproject3:junit excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">  
 486: <customize/>  
 487: </j2seproject3:junit>  
 488: </sequential>  
 489: </macrodef>  
 490: </target>  
 491: <target depends="-init-macrodef-testng" if="${testng.available}" name="-init-macrodef-testng-impl">  
 492: <macrodef name="test-impl" uri="http://www.netbeans.org/ns/j2se-project/3">  
 493: <attribute default="${includes}" name="includes"/>  
 494: <attribute default="${excludes}" name="excludes"/>  
 495: <attribute default="\*\*" name="testincludes"/>  
 496: <attribute default="" name="testmethods"/>  
 497: <element implicit="true" name="customize" optional="true"/>  
 498: <sequential>  
 499: <j2seproject3:testng excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">  
 500: <customize/>  
 501: </j2seproject3:testng>  
 502: </sequential>  
 503: </macrodef>  
 504: </target>  
 505: <target depends="-init-macrodef-test-impl,-init-macrodef-junit-impl,-init-macrodef-testng-impl" name="-init-macrodef-test">  
 506: <macrodef name="test" uri="http://www.netbeans.org/ns/j2se-project/3">  
 507: <attribute default="${includes}" name="includes"/>  
 508: <attribute default="${excludes}" name="excludes"/>  
 509: <attribute default="\*\*" name="testincludes"/>  
 510: <attribute default="" name="testmethods"/>  
 511: <sequential>  
 512: <j2seproject3:test-impl excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">  
 513: <customize>  
 514: <classpath>  
 515: <path path="${run.test.classpath}"/>  
 516: </classpath>  
 517: <jvmarg line="${endorsed.classpath.cmd.line.arg}"/>  
 518: <jvmarg line="${run.jvmargs}"/>  
 519: <jvmarg line="${run.jvmargs.ide}"/>  
 520: </customize>  
 521: </j2seproject3:test-impl>  
 522: </sequential>  
 523: </macrodef>  
 524: </target>  
 525: <target if="${junit.available}" name="-init-macrodef-junit-debug" unless="${nb.junit.batch}">  
 526: <macrodef name="junit-debug" uri="http://www.netbeans.org/ns/j2se-project/3">  
 527: <attribute default="${includes}" name="includes"/>  
 528: <attribute default="${excludes}" name="excludes"/>  
 529: <attribute default="\*\*" name="testincludes"/>  
 530: <attribute default="" name="testmethods"/>  
 531: <element name="customize" optional="true"/>  
 532: <sequential>  
 533: <property name="junit.forkmode" value="perTest"/>  
 534: <junit dir="${work.dir}" errorproperty="tests.failed" failureproperty="tests.failed" fork="true" forkmode="${junit.forkmode}" showoutput="true" tempdir="${build.dir}">  
 535: <test methods="@{testmethods}" name="@{testincludes}" todir="${build.test.results.dir}"/>  
 536: <syspropertyset>  
 537: <propertyref prefix="test-sys-prop."/>  
 538: <mapper from="test-sys-prop.\*" to="\*" type="glob"/>  
 539: </syspropertyset>  
 540: <formatter type="brief" usefile="false"/>  
 541: <formatter type="xml"/>  
 542: <jvmarg value="-ea"/>  
 543: <jvmarg line="${debug-args-line}"/>  
 544: <jvmarg value="-Xrunjdwp:transport=${debug-transport},address=${jpda.address}"/>  
 545: <customize/>  
 546: </junit>  
 547: </sequential>  
 548: </macrodef>  
 549: </target>  
 550: <target depends="-init-test-properties" if="${nb.junit.batch}" name="-init-macrodef-junit-debug-batch">  
 551: <macrodef name="junit-debug" uri="http://www.netbeans.org/ns/j2se-project/3">  
 552: <attribute default="${includes}" name="includes"/>  
 553: <attribute default="${excludes}" name="excludes"/>  
 554: <attribute default="\*\*" name="testincludes"/>  
 555: <attribute default="" name="testmethods"/>  
 556: <element name="customize" optional="true"/>  
 557: <sequential>  
 558: <property name="junit.forkmode" value="perTest"/>  
 559: <junit dir="${work.dir}" errorproperty="tests.failed" failureproperty="tests.failed" fork="true" forkmode="${junit.forkmode}" showoutput="true" tempdir="${build.dir}">  
 560: <batchtest todir="${build.test.results.dir}">  
 561: <fileset dir="${test.src.dir}" excludes="@{excludes},${excludes}" includes="@{includes}">  
 562: <filename name="@{testincludes}"/>  
 563: </fileset>  
 564: <fileset dir="${build.test.classes.dir}" excludes="@{excludes},${excludes},${test.binaryexcludes}" includes="${test.binaryincludes}">  
 565: <filename name="${test.binarytestincludes}"/>  
 566: </fileset>  
 567: </batchtest>  
 568: <syspropertyset>  
 569: <propertyref prefix="test-sys-prop."/>  
 570: <mapper from="test-sys-prop.\*" to="\*" type="glob"/>  
 571: </syspropertyset>  
 572: <formatter type="brief" usefile="false"/>  
 573: <formatter type="xml"/>  
 574: <jvmarg value="-ea"/>  
 575: <jvmarg line="${debug-args-line}"/>  
 576: <jvmarg value="-Xrunjdwp:transport=${debug-transport},address=${jpda.address}"/>  
 577: <customize/>  
 578: </junit>  
 579: </sequential>  
 580: </macrodef>  
 581: </target>  
 582: <target depends="-init-macrodef-junit-debug,-init-macrodef-junit-debug-batch" if="${junit.available}" name="-init-macrodef-junit-debug-impl">  
 583: <macrodef name="test-debug-impl" uri="http://www.netbeans.org/ns/j2se-project/3">  
 584: <attribute default="${includes}" name="includes"/>  
 585: <attribute default="${excludes}" name="excludes"/>  
 586: <attribute default="\*\*" name="testincludes"/>  
 587: <attribute default="" name="testmethods"/>  
 588: <element implicit="true" name="customize" optional="true"/>  
 589: <sequential>  
 590: <j2seproject3:junit-debug excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">  
 591: <customize/>  
 592: </j2seproject3:junit-debug>  
 593: </sequential>  
 594: </macrodef>  
 595: </target>  
 596: <target if="${testng.available}" name="-init-macrodef-testng-debug">  
 597: <macrodef name="testng-debug" uri="http://www.netbeans.org/ns/j2se-project/3">  
 598: <attribute default="${main.class}" name="testClass"/>  
 599: <attribute default="" name="testMethod"/>  
 600: <element name="customize2" optional="true"/>  
 601: <sequential>  
 602: <condition else="-testclass @{testClass}" property="test.class.or.method" value="-methods @{testClass}.@{testMethod}">  
 603: <isset property="test.method"/>  
 604: </condition>  
 605: <condition else="-suitename concentrador -testname @{testClass} ${test.class.or.method}" property="testng.cmd.args" value="@{testClass}">  
 606: <matches pattern=".\*\.xml" string="@{testClass}"/>  
 607: </condition>  
 608: <delete dir="${build.test.results.dir}" quiet="true"/>  
 609: <mkdir dir="${build.test.results.dir}"/>  
 610: <j2seproject3:debug classname="org.testng.TestNG" classpath="${debug.test.classpath}">  
 611: <customize>  
 612: <customize2/>  
 613: <jvmarg value="-ea"/>  
 614: <arg line="${testng.debug.mode}"/>  
 615: <arg line="-d ${build.test.results.dir}"/>  
 616: <arg line="-listener org.testng.reporters.VerboseReporter"/>  
 617: <arg line="${testng.cmd.args}"/>  
 618: </customize>  
 619: </j2seproject3:debug>  
 620: </sequential>  
 621: </macrodef>  
 622: </target>  
 623: <target depends="-init-macrodef-testng-debug" if="${testng.available}" name="-init-macrodef-testng-debug-impl">  
 624: <macrodef name="testng-debug-impl" uri="http://www.netbeans.org/ns/j2se-project/3">  
 625: <attribute default="${main.class}" name="testClass"/>  
 626: <attribute default="" name="testMethod"/>  
 627: <element implicit="true" name="customize2" optional="true"/>  
 628: <sequential>  
 629: <j2seproject3:testng-debug testClass="@{testClass}" testMethod="@{testMethod}">  
 630: <customize2/>  
 631: </j2seproject3:testng-debug>  
 632: </sequential>  
 633: </macrodef>  
 634: </target>  
 635: <target depends="-init-macrodef-junit-debug-impl" if="${junit.available}" name="-init-macrodef-test-debug-junit">  
 636: <macrodef name="test-debug" uri="http://www.netbeans.org/ns/j2se-project/3">  
 637: <attribute default="${includes}" name="includes"/>  
 638: <attribute default="${excludes}" name="excludes"/>  
 639: <attribute default="\*\*" name="testincludes"/>  
 640: <attribute default="" name="testmethods"/>  
 641: <attribute default="${main.class}" name="testClass"/>  
 642: <attribute default="" name="testMethod"/>  
 643: <sequential>  
 644: <j2seproject3:test-debug-impl excludes="@{excludes}" includes="@{includes}" testincludes="@{testincludes}" testmethods="@{testmethods}">  
 645: <customize>  
 646: <classpath>  
 647: <path path="${run.test.classpath}"/>  
 648: </classpath>  
 649: <jvmarg line="${endorsed.classpath.cmd.line.arg}"/>  
 650: <jvmarg line="${run.jvmargs}"/>  
 651: <jvmarg line="${run.jvmargs.ide}"/>  
 652: </customize>  
 653: </j2seproject3:test-debug-impl>  
 654: </sequential>  
 655: </macrodef>  
 656: </target>  
 657: <target depends="-init-macrodef-testng-debug-impl" if="${testng.available}" name="-init-macrodef-test-debug-testng">  
 658: <macrodef name="test-debug" uri="http://www.netbeans.org/ns/j2se-project/3">  
 659: <attribute default="${includes}" name="includes"/>  
 660: <attribute default="${excludes}" name="excludes"/>  
 661: <attribute default="\*\*" name="testincludes"/>  
 662: <attribute default="" name="testmethods"/>  
 663: <attribute default="${main.class}" name="testClass"/>  
 664: <attribute default="" name="testMethod"/>  
 665: <sequential>  
 666: <j2seproject3:testng-debug-impl testClass="@{testClass}" testMethod="@{testMethod}">  
 667: <customize2>  
 668: <syspropertyset>  
 669: <propertyref prefix="test-sys-prop."/>  
 670: <mapper from="test-sys-prop.\*" to="\*" type="glob"/>  
 671: </syspropertyset>  
 672: </customize2>  
 673: </j2seproject3:testng-debug-impl>  
 674: </sequential>  
 675: </macrodef>  
 676: </target>  
 677: <target depends="-init-macrodef-test-debug-junit,-init-macrodef-test-debug-testng" name="-init-macrodef-test-debug"/>  
 678: <!--  
 679: pre NB7.2 profiling section; consider it deprecated  
 680: -->  
 681: <target depends="-profile-pre-init, init, -profile-post-init, -profile-init-macrodef-profile, -profile-init-check" if="profiler.info.jvmargs.agent" name="profile-init"/>  
 682: <target if="profiler.info.jvmargs.agent" name="-profile-pre-init">  
 683: <!-- Empty placeholder for easier customization. -->  
 684: <!-- You can override this target in the ../build.xml file. -->  
 685: </target>  
 686: <target if="profiler.info.jvmargs.agent" name="-profile-post-init">  
 687: <!-- Empty placeholder for easier customization. -->  
 688: <!-- You can override this target in the ../build.xml file. -->  
 689: </target>  
 690: <target if="profiler.info.jvmargs.agent" name="-profile-init-macrodef-profile">  
 691: <macrodef name="resolve">  
 692: <attribute name="name"/>  
 693: <attribute name="value"/>  
 694: <sequential>  
 695: <property name="@{name}" value="${env.@{value}}"/>  
 696: </sequential>  
 697: </macrodef>  
 698: <macrodef name="profile">  
 699: <attribute default="${main.class}" name="classname"/>  
 700: <element name="customize" optional="true"/>  
 701: <sequential>  
 702: <property environment="env"/>  
 703: <resolve name="profiler.current.path" value="${profiler.info.pathvar}"/>  
 704: <java classname="@{classname}" dir="${profiler.info.dir}" failonerror="${java.failonerror}" fork="true" jvm="${profiler.info.jvm}">  
 705: <jvmarg line="${endorsed.classpath.cmd.line.arg}"/>  
 706: <jvmarg value="${profiler.info.jvmargs.agent}"/>  
 707: <jvmarg line="${profiler.info.jvmargs}"/>  
 708: <env key="${profiler.info.pathvar}" path="${profiler.info.agentpath}:${profiler.current.path}"/>  
 709: <arg line="${application.args}"/>  
 710: <classpath>  
 711: <path path="${run.classpath}"/>  
 712: </classpath>  
 713: <syspropertyset>  
 714: <propertyref prefix="run-sys-prop."/>  
 715: <mapper from="run-sys-prop.\*" to="\*" type="glob"/>  
 716: </syspropertyset>  
 717: <customize/>  
 718: </java>  
 719: </sequential>  
 720: </macrodef>  
 721: </target>  
 722: <target depends="-profile-pre-init, init, -profile-post-init, -profile-init-macrodef-profile" if="profiler.info.jvmargs.agent" name="-profile-init-check">  
 723: <fail unless="profiler.info.jvm">Must set JVM to use for profiling in profiler.info.jvm</fail>  
 724: <fail unless="profiler.info.jvmargs.agent">Must set profiler agent JVM arguments in profiler.info.jvmargs.agent</fail>  
 725: </target>  
 726: <!--  
 727: end of pre NB7.2 profiling section  
 728: -->  
 729: <target depends="-init-debug-args" name="-init-macrodef-nbjpda">  
 730: <macrodef name="nbjpdastart" uri="http://www.netbeans.org/ns/j2se-project/1">  
 731: <attribute default="${main.class}" name="name"/>  
 732: <attribute default="${debug.classpath}" name="classpath"/>  
 733: <attribute default="" name="stopclassname"/>  
 734: <sequential>  
 735: <nbjpdastart addressproperty="jpda.address" name="@{name}" stopclassname="@{stopclassname}" transport="${debug-transport}">  
 736: <classpath>  
 737: <path path="@{classpath}"/>  
 738: </classpath>  
 739: </nbjpdastart>  
 740: </sequential>  
 741: </macrodef>  
 742: <macrodef name="nbjpdareload" uri="http://www.netbeans.org/ns/j2se-project/1">  
 743: <attribute default="${build.classes.dir}" name="dir"/>  
 744: <sequential>  
 745: <nbjpdareload>  
 746: <fileset dir="@{dir}" includes="${fix.classes}">  
 747: <include name="${fix.includes}\*.class"/>  
 748: </fileset>  
 749: </nbjpdareload>  
 750: </sequential>  
 751: </macrodef>  
 752: </target>  
 753: <target name="-init-debug-args">  
 754: <property name="version-output" value="java version &quot;${ant.java.version}"/>  
 755: <condition property="have-jdk-older-than-1.4">  
 756: <or>  
 757: <contains string="${version-output}" substring="java version &quot;1.0"/>  
 758: <contains string="${version-output}" substring="java version &quot;1.1"/>  
 759: <contains string="${version-output}" substring="java version &quot;1.2"/>  
 760: <contains string="${version-output}" substring="java version &quot;1.3"/>  
 761: </or>  
 762: </condition>  
 763: <condition else="-Xdebug" property="debug-args-line" value="-Xdebug -Xnoagent -Djava.compiler=none">  
 764: <istrue value="${have-jdk-older-than-1.4}"/>  
 765: </condition>  
 766: <condition else="dt\_socket" property="debug-transport-by-os" value="dt\_shmem">  
 767: <os family="windows"/>  
 768: </condition>  
 769: <condition else="${debug-transport-by-os}" property="debug-transport" value="${debug.transport}">  
 770: <isset property="debug.transport"/>  
 771: </condition>  
 772: </target>  
 773: <target depends="-init-debug-args" name="-init-macrodef-debug">  
 774: <macrodef name="debug" uri="http://www.netbeans.org/ns/j2se-project/3">  
 775: <attribute default="${main.class}" name="classname"/>  
 776: <attribute default="${debug.classpath}" name="classpath"/>  
 777: <element name="customize" optional="true"/>  
 778: <sequential>  
 779: <java classname="@{classname}" dir="${work.dir}" failonerror="${java.failonerror}" fork="true">  
 780: <jvmarg line="${endorsed.classpath.cmd.line.arg}"/>  
 781: <jvmarg line="${debug-args-line}"/>  
 782: <jvmarg value="-Xrunjdwp:transport=${debug-transport},address=${jpda.address}"/>  
 783: <jvmarg value="-Dfile.encoding=${runtime.encoding}"/>  
 784: <redirector errorencoding="${runtime.encoding}" inputencoding="${runtime.encoding}" outputencoding="${runtime.encoding}"/>  
 785: <jvmarg line="${run.jvmargs}"/>  
 786: <jvmarg line="${run.jvmargs.ide}"/>  
 787: <classpath>  
 788: <path path="@{classpath}"/>  
 789: </classpath>  
 790: <syspropertyset>  
 791: <propertyref prefix="run-sys-prop."/>  
 792: <mapper from="run-sys-prop.\*" to="\*" type="glob"/>  
 793: </syspropertyset>  
 794: <customize/>  
 795: </java>  
 796: </sequential>  
 797: </macrodef>  
 798: </target>  
 799: <target name="-init-macrodef-java">  
 800: <macrodef name="java" uri="http://www.netbeans.org/ns/j2se-project/1">  
 801: <attribute default="${main.class}" name="classname"/>  
 802: <attribute default="${run.classpath}" name="classpath"/>  
 803: <attribute default="jvm" name="jvm"/>  
 804: <element name="customize" optional="true"/>  
 805: <sequential>  
 806: <java classname="@{classname}" dir="${work.dir}" failonerror="${java.failonerror}" fork="true">  
 807: <jvmarg line="${endorsed.classpath.cmd.line.arg}"/>  
 808: <jvmarg value="-Dfile.encoding=${runtime.encoding}"/>  
 809: <redirector errorencoding="${runtime.encoding}" inputencoding="${runtime.encoding}" outputencoding="${runtime.encoding}"/>  
 810: <jvmarg line="${run.jvmargs}"/>  
 811: <jvmarg line="${run.jvmargs.ide}"/>  
 812: <classpath>  
 813: <path path="@{classpath}"/>  
 814: </classpath>  
 815: <syspropertyset>  
 816: <propertyref prefix="run-sys-prop."/>  
 817: <mapper from="run-sys-prop.\*" to="\*" type="glob"/>  
 818: </syspropertyset>  
 819: <customize/>  
 820: </java>  
 821: </sequential>  
 822: </macrodef>  
 823: </target>  
 824: <target name="-init-macrodef-copylibs">  
 825: <macrodef name="copylibs" uri="http://www.netbeans.org/ns/j2se-project/3">  
 826: <attribute default="${manifest.file}" name="manifest"/>  
 827: <element name="customize" optional="true"/>  
 828: <sequential>  
 829: <property location="${build.classes.dir}" name="build.classes.dir.resolved"/>  
 830: <pathconvert property="run.classpath.without.build.classes.dir">  
 831: <path path="${run.classpath}"/>  
 832: <map from="${build.classes.dir.resolved}" to=""/>  
 833: </pathconvert>  
 834: <pathconvert pathsep=" " property="jar.classpath">  
 835: <path path="${run.classpath.without.build.classes.dir}"/>  
 836: <chainedmapper>  
 837: <flattenmapper/>  
 838: <filtermapper>  
 839: <replacestring from=" " to="%20"/>  
 840: </filtermapper>  
 841: <globmapper from="\*" to="lib/\*"/>  
 842: </chainedmapper>  
 843: </pathconvert>  
 844: <taskdef classname="org.netbeans.modules.java.j2seproject.copylibstask.CopyLibs" classpath="${libs.CopyLibs.classpath}" name="copylibs"/>  
 845: <copylibs compress="${jar.compress}" excludeFromCopy="${copylibs.excludes}" index="${jar.index}" indexMetaInf="${jar.index.metainf}" jarfile="${dist.jar}" manifest="@{manifest}" rebase="${copylibs.rebase}" runtimeclasspath="${run.classpath.without.build.classes.dir}">  
 846: <fileset dir="${build.classes.dir}" excludes="${dist.archive.excludes}"/>  
 847: <manifest>  
 848: <attribute name="Class-Path" value="${jar.classpath}"/>  
 849: <customize/>  
 850: </manifest>  
 851: </copylibs>  
 852: </sequential>  
 853: </macrodef>  
 854: </target>  
 855: <target name="-init-presetdef-jar">  
 856: <presetdef name="jar" uri="http://www.netbeans.org/ns/j2se-project/1">  
 857: <jar compress="${jar.compress}" index="${jar.index}" jarfile="${dist.jar}">  
 858: <j2seproject1:fileset dir="${build.classes.dir}" excludes="${dist.archive.excludes}"/>  
 859: </jar>  
 860: </presetdef>  
 861: </target>  
 862: <target name="-init-ap-cmdline-properties">  
 863: <property name="annotation.processing.enabled" value="true"/>  
 864: <property name="annotation.processing.processors.list" value=""/>  
 865: <property name="annotation.processing.processor.options" value=""/>  
 866: <property name="annotation.processing.run.all.processors" value="true"/>  
 867: <property name="javac.processorpath" value="${javac.classpath}"/>  
 868: <property name="javac.test.processorpath" value="${javac.test.classpath}"/>  
 869: <condition property="ap.supported.internal" value="true">  
 870: <not>  
 871: <matches pattern="1\.[0-5](\..\*)?" string="${javac.source}"/>  
 872: </not>  
 873: </condition>  
 874: </target>  
 875: <target depends="-init-ap-cmdline-properties" if="ap.supported.internal" name="-init-ap-cmdline-supported">  
 876: <condition else="" property="ap.processors.internal" value="-processor ${annotation.processing.processors.list}">  
 877: <isfalse value="${annotation.processing.run.all.processors}"/>  
 878: </condition>  
 879: <condition else="" property="ap.proc.none.internal" value="-proc:none">  
 880: <isfalse value="${annotation.processing.enabled}"/>  
 881: </condition>  
 882: </target>  
 883: <target depends="-init-ap-cmdline-properties,-init-ap-cmdline-supported" name="-init-ap-cmdline">  
 884: <property name="ap.cmd.line.internal" value=""/>  
 885: </target>  
 886: <target depends="-pre-init,-init-private,-init-user,-init-project,-do-init,-post-init,-init-check,-init-macrodef-property,-init-macrodef-javac,-init-macrodef-test,-init-macrodef-test-debug,-init-macrodef-nbjpda,-init-macrodef-debug,-init-macrodef-java,-init-presetdef-jar,-init-ap-cmdline" name="init"/>  
 887: <!--  
 888: ===================  
 889: COMPILATION SECTION  
 890: ===================  
 891: -->  
 892: <target name="-deps-jar-init" unless="built-jar.properties">  
 893: <property location="${build.dir}/built-jar.properties" name="built-jar.properties"/>  
 894: <delete file="${built-jar.properties}" quiet="true"/>  
 895: </target>  
 896: <target if="already.built.jar.${basedir}" name="-warn-already-built-jar">  
 897: <echo level="warn" message="Cycle detected: concentrador was already built"/>  
 898: </target>  
 899: <target depends="init,-deps-jar-init" name="deps-jar" unless="no.deps">  
 900: <mkdir dir="${build.dir}"/>  
 901: <touch file="${built-jar.properties}" verbose="false"/>  
 902: <property file="${built-jar.properties}" prefix="already.built.jar."/>  
 903: <antcall target="-warn-already-built-jar"/>  
 904: <propertyfile file="${built-jar.properties}">  
 905: <entry key="${basedir}" value=""/>  
 906: </propertyfile>  
 907: </target>  
 908: <target depends="init,-check-automatic-build,-clean-after-automatic-build" name="-verify-automatic-build"/>  
 909: <target depends="init" name="-check-automatic-build">  
 910: <available file="${build.classes.dir}/.netbeans\_automatic\_build" property="netbeans.automatic.build"/>  
 911: </target>  
 912: <target depends="init" if="netbeans.automatic.build" name="-clean-after-automatic-build">  
 913: <antcall target="clean"/>  
 914: </target>  
 915: <target depends="init,deps-jar" name="-pre-pre-compile">  
 916: <mkdir dir="${build.classes.dir}"/>  
 917: </target>  
 918: <target name="-pre-compile">  
 919: <!-- Empty placeholder for easier customization. -->  
 920: <!-- You can override this target in the ../build.xml file. -->  
 921: </target>  
 922: <target if="do.depend.true" name="-compile-depend">  
 923: <pathconvert property="build.generated.subdirs">  
 924: <dirset dir="${build.generated.sources.dir}" erroronmissingdir="false">  
 925: <include name="\*"/>  
 926: </dirset>  
 927: </pathconvert>  
 928: <j2seproject3:depend srcdir="${src.dir}:${src.resources.dir}:${build.generated.subdirs}"/>  
 929: </target>  
 930: <target depends="init,deps-jar,-pre-pre-compile,-pre-compile, -copy-persistence-xml,-compile-depend" if="have.sources" name="-do-compile">  
 931: <j2seproject3:javac gensrcdir="${build.generated.sources.dir}"/>  
 932: <copy todir="${build.classes.dir}">  
 933: <fileset dir="${src.dir}" excludes="${build.classes.excludes},${excludes}" includes="${includes}"/>  
 934: <fileset dir="${src.resources.dir}" excludes="${build.classes.excludes},${excludes}" includes="${includes}"/>  
 935: </copy>  
 936: </target>  
 937: <target if="has.persistence.xml" name="-copy-persistence-xml">  
 938: <mkdir dir="${build.classes.dir}/META-INF"/>  
 939: <copy todir="${build.classes.dir}/META-INF">  
 940: <fileset dir="${meta.inf.dir}" includes="persistence.xml orm.xml"/>  
 941: </copy>  
 942: </target>  
 943: <target name="-post-compile">  
 944: <!-- Empty placeholder for easier customization. -->  
 945: <!-- You can override this target in the ../build.xml file. -->  
 946: </target>  
 947: <target depends="init,deps-jar,-verify-automatic-build,-pre-pre-compile,-pre-compile,-do-compile,-post-compile" description="Compile project." name="compile"/>  
 948: <target name="-pre-compile-single">  
 949: <!-- Empty placeholder for easier customization. -->  
 950: <!-- You can override this target in the ../build.xml file. -->  
 951: </target>  
 952: <target depends="init,deps-jar,-pre-pre-compile" name="-do-compile-single">  
 953: <fail unless="javac.includes">Must select some files in the IDE or set javac.includes</fail>  
 954: <j2seproject3:force-recompile/>  
 955: <j2seproject3:javac excludes="" gensrcdir="${build.generated.sources.dir}" includes="${javac.includes}" sourcepath="${src.dir}:${src.resources.dir}"/>  
 956: </target>  
 957: <target name="-post-compile-single">  
 958: <!-- Empty placeholder for easier customization. -->  
 959: <!-- You can override this target in the ../build.xml file. -->  
 960: </target>  
 961: <target depends="init,deps-jar,-verify-automatic-build,-pre-pre-compile,-pre-compile-single,-do-compile-single,-post-compile-single" name="compile-single"/>  
 962: <!--  
 963: ====================  
 964: JAR BUILDING SECTION  
 965: ====================  
 966: -->  
 967: <target depends="init" name="-pre-pre-jar">  
 968: <dirname file="${dist.jar}" property="dist.jar.dir"/>  
 969: <mkdir dir="${dist.jar.dir}"/>  
 970: </target>  
 971: <target name="-pre-jar">  
 972: <!-- Empty placeholder for easier customization. -->  
 973: <!-- You can override this target in the ../build.xml file. -->  
 974: </target>  
 975: <target depends="init" if="do.archive" name="-do-jar-create-manifest" unless="manifest.available">  
 976: <tempfile deleteonexit="true" destdir="${build.dir}" property="tmp.manifest.file"/>  
 977: <touch file="${tmp.manifest.file}" verbose="false"/>  
 978: </target>  
 979: <target depends="init" if="do.archive+manifest.available" name="-do-jar-copy-manifest">  
 980: <tempfile deleteonexit="true" destdir="${build.dir}" property="tmp.manifest.file"/>  
 981: <copy file="${manifest.file}" tofile="${tmp.manifest.file}"/>  
 982: </target>  
 983: <target depends="init,-do-jar-create-manifest,-do-jar-copy-manifest" if="do.archive+main.class.available" name="-do-jar-set-mainclass">  
 984: <manifest file="${tmp.manifest.file}" mode="update">  
 985: <attribute name="Main-Class" value="${main.class}"/>  
 986: </manifest>  
 987: </target>  
 988: <target depends="init,-do-jar-create-manifest,-do-jar-copy-manifest" if="do.archive+profile.available" name="-do-jar-set-profile">  
 989: <manifest file="${tmp.manifest.file}" mode="update">  
 990: <attribute name="Profile" value="${javac.profile}"/>  
 991: </manifest>  
 992: </target>  
 993: <target depends="init,-do-jar-create-manifest,-do-jar-copy-manifest" if="do.archive+splashscreen.available" name="-do-jar-set-splashscreen">  
 994: <basename file="${application.splash}" property="splashscreen.basename"/>  
 995: <mkdir dir="${build.classes.dir}/META-INF"/>  
 996: <copy failonerror="false" file="${application.splash}" todir="${build.classes.dir}/META-INF"/>  
 997: <manifest file="${tmp.manifest.file}" mode="update">  
 998: <attribute name="SplashScreen-Image" value="META-INF/${splashscreen.basename}"/>  
 999: </manifest>  
1000: </target>  
1001: <target depends="init,-init-macrodef-copylibs,compile,-pre-pre-jar,-pre-jar,-do-jar-create-manifest,-do-jar-copy-manifest,-do-jar-set-mainclass,-do-jar-set-profile,-do-jar-set-splashscreen" if="do.mkdist" name="-do-jar-copylibs">  
1002: <j2seproject3:copylibs manifest="${tmp.manifest.file}"/>  
1003: <echo level="info">To run this application from the command line without Ant, try:</echo>  
1004: <property location="${dist.jar}" name="dist.jar.resolved"/>  
1005: <echo level="info">java -jar "${dist.jar.resolved}"</echo>  
1006: </target>  
1007: <target depends="init,compile,-pre-pre-jar,-pre-jar,-do-jar-create-manifest,-do-jar-copy-manifest,-do-jar-set-mainclass,-do-jar-set-profile,-do-jar-set-splashscreen" if="do.archive" name="-do-jar-jar" unless="do.mkdist">  
1008: <j2seproject1:jar manifest="${tmp.manifest.file}"/>  
1009: <property location="${build.classes.dir}" name="build.classes.dir.resolved"/>  
1010: <property location="${dist.jar}" name="dist.jar.resolved"/>  
1011: <pathconvert property="run.classpath.with.dist.jar">  
1012: <path path="${run.classpath}"/>  
1013: <map from="${build.classes.dir.resolved}" to="${dist.jar.resolved}"/>  
1014: </pathconvert>  
1015: <condition else="" property="jar.usage.message" value="To run this application from the command line without Ant, try:${line.separator}${platform.java} -cp ${run.classpath.with.dist.jar} ${main.class}">  
1016: <isset property="main.class.available"/>  
1017: </condition>  
1018: <condition else="debug" property="jar.usage.level" value="info">  
1019: <isset property="main.class.available"/>  
1020: </condition>  
1021: <echo level="${jar.usage.level}" message="${jar.usage.message}"/>  
1022: </target>  
1023: <target depends="-do-jar-copylibs" if="do.archive" name="-do-jar-delete-manifest">  
1024: <delete>  
1025: <fileset file="${tmp.manifest.file}"/>  
1026: </delete>  
1027: </target>  
1028: <target depends="init,compile,-pre-pre-jar,-pre-jar,-do-jar-create-manifest,-do-jar-copy-manifest,-do-jar-set-mainclass,-do-jar-set-profile,-do-jar-set-splashscreen,-do-jar-jar,-do-jar-delete-manifest" name="-do-jar-without-libraries"/>  
1029: <target depends="init,compile,-pre-pre-jar,-pre-jar,-do-jar-create-manifest,-do-jar-copy-manifest,-do-jar-set-mainclass,-do-jar-set-profile,-do-jar-set-splashscreen,-do-jar-copylibs,-do-jar-delete-manifest" name="-do-jar-with-libraries"/>  
1030: <target name="-post-jar">  
1031: <!-- Empty placeholder for easier customization. -->  
1032: <!-- You can override this target in the ../build.xml file. -->  
1033: </target>  
1034: <target depends="init,compile,-pre-jar,-do-jar-without-libraries,-do-jar-with-libraries,-post-jar" name="-do-jar"/>  
1035: <target depends="init,compile,-pre-jar,-do-jar,-post-jar" description="Build JAR." name="jar"/>  
1036: <!--  
1037: =================  
1038: EXECUTION SECTION  
1039: =================  
1040: -->  
1041: <target depends="init,compile" description="Run a main class." name="run">  
1042: <j2seproject1:java>  
1043: <customize>  
1044: <arg line="${application.args}"/>  
1045: </customize>  
1046: </j2seproject1:java>  
1047: </target>  
1048: <target name="-do-not-recompile">  
1049: <property name="javac.includes.binary" value=""/>  
1050: </target>  
1051: <target depends="init,compile-single" name="run-single">  
1052: <fail unless="run.class">Must select one file in the IDE or set run.class</fail>  
1053: <j2seproject1:java classname="${run.class}"/>  
1054: </target>  
1055: <target depends="init,compile-test-single" name="run-test-with-main">  
1056: <fail unless="run.class">Must select one file in the IDE or set run.class</fail>  
1057: <j2seproject1:java classname="${run.class}" classpath="${run.test.classpath}"/>  
1058: </target>  
1059: <!--  
1060: =================  
1061: DEBUGGING SECTION  
1062: =================  
1063: -->  
1064: <target depends="init" if="netbeans.home" name="-debug-start-debugger">  
1065: <j2seproject1:nbjpdastart name="${debug.class}"/>  
1066: </target>  
1067: <target depends="init" if="netbeans.home" name="-debug-start-debugger-main-test">  
1068: <j2seproject1:nbjpdastart classpath="${debug.test.classpath}" name="${debug.class}"/>  
1069: </target>  
1070: <target depends="init,compile" name="-debug-start-debuggee">  
1071: <j2seproject3:debug>  
1072: <customize>  
1073: <arg line="${application.args}"/>  
1074: </customize>  
1075: </j2seproject3:debug>  
1076: </target>  
1077: <target depends="init,compile,-debug-start-debugger,-debug-start-debuggee" description="Debug project in IDE." if="netbeans.home" name="debug"/>  
1078: <target depends="init" if="netbeans.home" name="-debug-start-debugger-stepinto">  
1079: <j2seproject1:nbjpdastart stopclassname="${main.class}"/>  
1080: </target>  
1081: <target depends="init,compile,-debug-start-debugger-stepinto,-debug-start-debuggee" if="netbeans.home" name="debug-stepinto"/>  
1082: <target depends="init,compile-single" if="netbeans.home" name="-debug-start-debuggee-single">  
1083: <fail unless="debug.class">Must select one file in the IDE or set debug.class</fail>  
1084: <j2seproject3:debug classname="${debug.class}"/>  
1085: </target>  
1086: <target depends="init,compile-single,-debug-start-debugger,-debug-start-debuggee-single" if="netbeans.home" name="debug-single"/>  
1087: <target depends="init,compile-test-single" if="netbeans.home" name="-debug-start-debuggee-main-test">  
1088: <fail unless="debug.class">Must select one file in the IDE or set debug.class</fail>  
1089: <j2seproject3:debug classname="${debug.class}" classpath="${debug.test.classpath}"/>  
1090: </target>  
1091: <target depends="init,compile-test-single,-debug-start-debugger-main-test,-debug-start-debuggee-main-test" if="netbeans.home" name="debug-test-with-main"/>  
1092: <target depends="init" name="-pre-debug-fix">  
1093: <fail unless="fix.includes">Must set fix.includes</fail>  
1094: <property name="javac.includes" value="${fix.includes}.java"/>  
1095: </target>  
1096: <target depends="init,-pre-debug-fix,compile-single" if="netbeans.home" name="-do-debug-fix">  
1097: <j2seproject1:nbjpdareload/>  
1098: </target>  
1099: <target depends="init,-pre-debug-fix,-do-debug-fix" if="netbeans.home" name="debug-fix"/>  
1100: <!--  
1101: =================  
1102: PROFILING SECTION  
1103: =================  
1104: -->  
1105: <!--  
1106: pre NB7.2 profiler integration  
1107: -->  
1108: <target depends="profile-init,compile" description="Profile a project in the IDE." if="profiler.info.jvmargs.agent" name="-profile-pre72">  
1109: <fail unless="netbeans.home">This target only works when run from inside the NetBeans IDE.</fail>  
1110: <nbprofiledirect>  
1111: <classpath>  
1112: <path path="${run.classpath}"/>  
1113: </classpath>  
1114: </nbprofiledirect>  
1115: <profile/>  
1116: </target>  
1117: <target depends="profile-init,compile-single" description="Profile a selected class in the IDE." if="profiler.info.jvmargs.agent" name="-profile-single-pre72">  
1118: <fail unless="profile.class">Must select one file in the IDE or set profile.class</fail>  
1119: <fail unless="netbeans.home">This target only works when run from inside the NetBeans IDE.</fail>  
1120: <nbprofiledirect>  
1121: <classpath>  
1122: <path path="${run.classpath}"/>  
1123: </classpath>  
1124: </nbprofiledirect>  
1125: <profile classname="${profile.class}"/>  
1126: </target>  
1127: <target depends="profile-init,compile-single" if="profiler.info.jvmargs.agent" name="-profile-applet-pre72">  
1128: <fail unless="netbeans.home">This target only works when run from inside the NetBeans IDE.</fail>  
1129: <nbprofiledirect>  
1130: <classpath>  
1131: <path path="${run.classpath}"/>  
1132: </classpath>  
1133: </nbprofiledirect>  
1134: <profile classname="sun.applet.AppletViewer">  
1135: <customize>  
1136: <arg value="${applet.url}"/>  
1137: </customize>  
1138: </profile>  
1139: </target>  
1140: <target depends="profile-init,compile-test-single" if="profiler.info.jvmargs.agent" name="-profile-test-single-pre72">  
1141: <fail unless="netbeans.home">This target only works when run from inside the NetBeans IDE.</fail>  
1142: <nbprofiledirect>  
1143: <classpath>  
1144: <path path="${run.test.classpath}"/>  
1145: </classpath>  
1146: </nbprofiledirect>  
1147: <junit dir="${profiler.info.dir}" errorproperty="tests.failed" failureproperty="tests.failed" fork="true" jvm="${profiler.info.jvm}" showoutput="true">  
1148: <env key="${profiler.info.pathvar}" path="${profiler.info.agentpath}:${profiler.current.path}"/>  
1149: <jvmarg value="${profiler.info.jvmargs.agent}"/>  
1150: <jvmarg line="${profiler.info.jvmargs}"/>  
1151: <test name="${profile.class}"/>  
1152: <classpath>  
1153: <path path="${run.test.classpath}"/>  
1154: </classpath>  
1155: <syspropertyset>  
1156: <propertyref prefix="test-sys-prop."/>  
1157: <mapper from="test-sys-prop.\*" to="\*" type="glob"/>  
1158: </syspropertyset>  
1159: <formatter type="brief" usefile="false"/>  
1160: <formatter type="xml"/>  
1161: </junit>  
1162: </target>  
1163: <!--  
1164: end of pre NB72 profiling section  
1165: -->  
1166: <target if="netbeans.home" name="-profile-check">  
1167: <condition property="profiler.configured">  
1168: <or>  
1169: <contains casesensitive="true" string="${run.jvmargs.ide}" substring="-agentpath:"/>  
1170: <contains casesensitive="true" string="${run.jvmargs.ide}" substring="-javaagent:"/>  
1171: </or>  
1172: </condition>  
1173: </target>  
1174: <target depends="-profile-check,-profile-pre72" description="Profile a project in the IDE." if="profiler.configured" name="profile" unless="profiler.info.jvmargs.agent">  
1175: <startprofiler/>  
1176: <antcall target="run"/>  
1177: </target>  
1178: <target depends="-profile-check,-profile-single-pre72" description="Profile a selected class in the IDE." if="profiler.configured" name="profile-single" unless="profiler.info.jvmargs.agent">  
1179: <fail unless="run.class">Must select one file in the IDE or set run.class</fail>  
1180: <startprofiler/>  
1181: <antcall target="run-single"/>  
1182: </target>  
1183: <target depends="-profile-test-single-pre72" description="Profile a selected test in the IDE." name="profile-test-single"/>  
1184: <target depends="-profile-check" description="Profile a selected test in the IDE." if="profiler.configured" name="profile-test" unless="profiler.info.jvmargs">  
1185: <fail unless="test.includes">Must select some files in the IDE or set test.includes</fail>  
1186: <startprofiler/>  
1187: <antcall target="test-single"/>  
1188: </target>  
1189: <target depends="-profile-check" description="Profile a selected class in the IDE." if="profiler.configured" name="profile-test-with-main">  
1190: <fail unless="run.class">Must select one file in the IDE or set run.class</fail>  
1191: <startprofiler/>  
1192: <antcal target="run-test-with-main"/>  
1193: </target>  
1194: <target depends="-profile-check,-profile-applet-pre72" if="profiler.configured" name="profile-applet" unless="profiler.info.jvmargs.agent">  
1195: <fail unless="applet.url">Must select one file in the IDE or set applet.url</fail>  
1196: <startprofiler/>  
1197: <antcall target="run-applet"/>  
1198: </target>  
1199: <!--  
1200: ===============  
1201: JAVADOC SECTION  
1202: ===============  
1203: -->  
1204: <target depends="init" if="have.sources" name="-javadoc-build">  
1205: <mkdir dir="${dist.javadoc.dir}"/>  
1206: <condition else="" property="javadoc.endorsed.classpath.cmd.line.arg" value="-J${endorsed.classpath.cmd.line.arg}">  
1207: <and>  
1208: <isset property="endorsed.classpath.cmd.line.arg"/>  
1209: <not>  
1210: <equals arg1="${endorsed.classpath.cmd.line.arg}" arg2=""/>  
1211: </not>  
1212: </and>  
1213: </condition>  
1214: <condition else="" property="bug5101868workaround" value="\*.java">  
1215: <matches pattern="1\.[56](\..\*)?" string="${java.version}"/>  
1216: </condition>  
1217: <javadoc additionalparam="-J-Dfile.encoding=${file.encoding} ${javadoc.additionalparam}" author="${javadoc.author}" charset="UTF-8" destdir="${dist.javadoc.dir}" docencoding="UTF-8" encoding="${javadoc.encoding.used}" failonerror="true" noindex="${javadoc.noindex}" nonavbar="${javadoc.nonavbar}" notree="${javadoc.notree}" private="${javadoc.private}" source="${javac.source}" splitindex="${javadoc.splitindex}" use="${javadoc.use}" useexternalfile="true" version="${javadoc.version}" windowtitle="${javadoc.windowtitle}">  
1218: <classpath>  
1219: <path path="${javac.classpath}"/>  
1220: </classpath>  
1221: <fileset dir="${src.dir}" excludes="${bug5101868workaround},${excludes}" includes="${includes}">  
1222: <filename name="\*\*/\*.java"/>  
1223: </fileset>  
1224: <fileset dir="${src.resources.dir}" excludes="${bug5101868workaround},${excludes}" includes="${includes}">  
1225: <filename name="\*\*/\*.java"/>  
1226: </fileset>  
1227: <fileset dir="${build.generated.sources.dir}" erroronmissingdir="false">  
1228: <include name="\*\*/\*.java"/>  
1229: <exclude name="\*.java"/>  
1230: </fileset>  
1231: <arg line="${javadoc.endorsed.classpath.cmd.line.arg}"/>  
1232: </javadoc>  
1233: <copy todir="${dist.javadoc.dir}">  
1234: <fileset dir="${src.dir}" excludes="${excludes}" includes="${includes}">  
1235: <filename name="\*\*/doc-files/\*\*"/>  
1236: </fileset>  
1237: <fileset dir="${src.resources.dir}" excludes="${excludes}" includes="${includes}">  
1238: <filename name="\*\*/doc-files/\*\*"/>  
1239: </fileset>  
1240: <fileset dir="${build.generated.sources.dir}" erroronmissingdir="false">  
1241: <include name="\*\*/doc-files/\*\*"/>  
1242: </fileset>  
1243: </copy>  
1244: </target>  
1245: <target depends="init,-javadoc-build" if="netbeans.home" name="-javadoc-browse" unless="no.javadoc.preview">  
1246: <nbbrowse file="${dist.javadoc.dir}/index.html"/>  
1247: </target>  
1248: <target depends="init,-javadoc-build,-javadoc-browse" description="Build Javadoc." name="javadoc"/>  
1249: <!--  
1250: =========================  
1251: TEST COMPILATION SECTION  
1252: =========================  
1253: -->  
1254: <target depends="init,compile" if="have.tests" name="-pre-pre-compile-test">  
1255: <mkdir dir="${build.test.classes.dir}"/>  
1256: </target>  
1257: <target name="-pre-compile-test">  
1258: <!-- Empty placeholder for easier customization. -->  
1259: <!-- You can override this target in the ../build.xml file. -->  
1260: </target>  
1261: <target if="do.depend.true" name="-compile-test-depend">  
1262: <j2seproject3:depend classpath="${javac.test.classpath}" destdir="${build.test.classes.dir}" srcdir="${test.src.dir}"/>  
1263: </target>  
1264: <target depends="init,deps-jar,compile,-pre-pre-compile-test,-pre-compile-test,-compile-test-depend" if="have.tests" name="-do-compile-test">  
1265: <j2seproject3:javac apgeneratedsrcdir="${build.test.classes.dir}" classpath="${javac.test.classpath}" debug="true" destdir="${build.test.classes.dir}" processorpath="${javac.test.processorpath}" srcdir="${test.src.dir}"/>  
1266: <copy todir="${build.test.classes.dir}">  
1267: <fileset dir="${test.src.dir}" excludes="${build.classes.excludes},${excludes}" includes="${includes}"/>  
1268: </copy>  
1269: </target>  
1270: <target name="-post-compile-test">  
1271: <!-- Empty placeholder for easier customization. -->  
1272: <!-- You can override this target in the ../build.xml file. -->  
1273: </target>  
1274: <target depends="init,compile,-pre-pre-compile-test,-pre-compile-test,-do-compile-test,-post-compile-test" name="compile-test"/>  
1275: <target name="-pre-compile-test-single">  
1276: <!-- Empty placeholder for easier customization. -->  
1277: <!-- You can override this target in the ../build.xml file. -->  
1278: </target>  
1279: <target depends="init,deps-jar,compile,-pre-pre-compile-test,-pre-compile-test-single" if="have.tests" name="-do-compile-test-single">  
1280: <fail unless="javac.includes">Must select some files in the IDE or set javac.includes</fail>  
1281: <j2seproject3:force-recompile destdir="${build.test.classes.dir}"/>  
1282: <j2seproject3:javac apgeneratedsrcdir="${build.test.classes.dir}" classpath="${javac.test.classpath}" debug="true" destdir="${build.test.classes.dir}" excludes="" includes="${javac.includes}" processorpath="${javac.test.processorpath}" sourcepath="${test.src.dir}" srcdir="${test.src.dir}"/>  
1283: <copy todir="${build.test.classes.dir}">  
1284: <fileset dir="${test.src.dir}" excludes="${build.classes.excludes},${excludes}" includes="${includes}"/>  
1285: </copy>  
1286: </target>  
1287: <target name="-post-compile-test-single">  
1288: <!-- Empty placeholder for easier customization. -->  
1289: <!-- You can override this target in the ../build.xml file. -->  
1290: </target>  
1291: <target depends="init,compile,-pre-pre-compile-test,-pre-compile-test-single,-do-compile-test-single,-post-compile-test-single" name="compile-test-single"/>  
1292: <!--  
1293: =======================  
1294: TEST EXECUTION SECTION  
1295: =======================  
1296: -->  
1297: <target depends="init" if="have.tests" name="-pre-test-run">  
1298: <mkdir dir="${build.test.results.dir}"/>  
1299: </target>  
1300: <target depends="init,compile-test,-pre-test-run" if="have.tests" name="-do-test-run">  
1301: <j2seproject3:test includes="${includes}" testincludes="\*\*/\*Test.java"/>  
1302: </target>  
1303: <target depends="init,compile-test,-pre-test-run,-do-test-run" if="have.tests" name="-post-test-run">  
1304: <fail if="tests.failed" unless="ignore.failing.tests">Some tests failed; see details above.</fail>  
1305: </target>  
1306: <target depends="init" if="have.tests" name="test-report"/>  
1307: <target depends="init" if="netbeans.home+have.tests" name="-test-browse"/>  
1308: <target depends="init,compile-test,-pre-test-run,-do-test-run,test-report,-post-test-run,-test-browse" description="Run unit tests." name="test"/>  
1309: <target depends="init" if="have.tests" name="-pre-test-run-single">  
1310: <mkdir dir="${build.test.results.dir}"/>  
1311: </target>  
1312: <target depends="init,compile-test-single,-pre-test-run-single" if="have.tests" name="-do-test-run-single">  
1313: <fail unless="test.includes">Must select some files in the IDE or set test.includes</fail>  
1314: <j2seproject3:test excludes="" includes="${test.includes}" testincludes="${test.includes}"/>  
1315: </target>  
1316: <target depends="init,compile-test-single,-pre-test-run-single,-do-test-run-single" if="have.tests" name="-post-test-run-single">  
1317: <fail if="tests.failed" unless="ignore.failing.tests">Some tests failed; see details above.</fail>  
1318: </target>  
1319: <target depends="init,compile-test-single,-pre-test-run-single,-do-test-run-single,-post-test-run-single" description="Run single unit test." name="test-single"/>  
1320: <target depends="init,compile-test-single,-pre-test-run-single" if="have.tests" name="-do-test-run-single-method">  
1321: <fail unless="test.class">Must select some files in the IDE or set test.class</fail>  
1322: <fail unless="test.method">Must select some method in the IDE or set test.method</fail>  
1323: <j2seproject3:test excludes="" includes="${javac.includes}" testincludes="${test.class}" testmethods="${test.method}"/>  
1324: </target>  
1325: <target depends="init,compile-test-single,-pre-test-run-single,-do-test-run-single-method" if="have.tests" name="-post-test-run-single-method">  
1326: <fail if="tests.failed" unless="ignore.failing.tests">Some tests failed; see details above.</fail>  
1327: </target>  
1328: <target depends="init,compile-test-single,-pre-test-run-single,-do-test-run-single-method,-post-test-run-single-method" description="Run single unit test." name="test-single-method"/>  
1329: <!--  
1330: =======================  
1331: TEST DEBUGGING SECTION  
1332: =======================  
1333: -->  
1334: <target depends="init,compile-test-single,-pre-test-run-single" if="have.tests" name="-debug-start-debuggee-test">  
1335: <fail unless="test.class">Must select one file in the IDE or set test.class</fail>  
1336: <j2seproject3:test-debug excludes="" includes="${javac.includes}" testClass="${test.class}" testincludes="${javac.includes}"/>  
1337: </target>  
1338: <target depends="init,compile-test-single,-pre-test-run-single" if="have.tests" name="-debug-start-debuggee-test-method">  
1339: <fail unless="test.class">Must select one file in the IDE or set test.class</fail>  
1340: <fail unless="test.method">Must select some method in the IDE or set test.method</fail>  
1341: <j2seproject3:test-debug excludes="" includes="${javac.includes}" testClass="${test.class}" testMethod="${test.method}" testincludes="${test.class}" testmethods="${test.method}"/>  
1342: </target>  
1343: <target depends="init,compile-test" if="netbeans.home+have.tests" name="-debug-start-debugger-test">  
1344: <j2seproject1:nbjpdastart classpath="${debug.test.classpath}" name="${test.class}"/>  
1345: </target>  
1346: <target depends="init,compile-test-single,-debug-start-debugger-test,-debug-start-debuggee-test" name="debug-test"/>  
1347: <target depends="init,compile-test-single,-debug-start-debugger-test,-debug-start-debuggee-test-method" name="debug-test-method"/>  
1348: <target depends="init,-pre-debug-fix,compile-test-single" if="netbeans.home" name="-do-debug-fix-test">  
1349: <j2seproject1:nbjpdareload dir="${build.test.classes.dir}"/>  
1350: </target>  
1351: <target depends="init,-pre-debug-fix,-do-debug-fix-test" if="netbeans.home" name="debug-fix-test"/>  
1352: <!--  
1353: =========================  
1354: APPLET EXECUTION SECTION  
1355: =========================  
1356: -->  
1357: <target depends="init,compile-single" name="run-applet">  
1358: <fail unless="applet.url">Must select one file in the IDE or set applet.url</fail>  
1359: <j2seproject1:java classname="sun.applet.AppletViewer">  
1360: <customize>  
1361: <arg value="${applet.url}"/>  
1362: </customize>  
1363: </j2seproject1:java>  
1364: </target>  
1365: <!--  
1366: =========================  
1367: APPLET DEBUGGING SECTION  
1368: =========================  
1369: -->  
1370: <target depends="init,compile-single" if="netbeans.home" name="-debug-start-debuggee-applet">  
1371: <fail unless="applet.url">Must select one file in the IDE or set applet.url</fail>  
1372: <j2seproject3:debug classname="sun.applet.AppletViewer">  
1373: <customize>  
1374: <arg value="${applet.url}"/>  
1375: </customize>  
1376: </j2seproject3:debug>  
1377: </target>  
1378: <target depends="init,compile-single,-debug-start-debugger,-debug-start-debuggee-applet" if="netbeans.home" name="debug-applet"/>  
1379: <!--  
1380: ===============  
1381: CLEANUP SECTION  
1382: ===============  
1383: -->  
1384: <target name="-deps-clean-init" unless="built-clean.properties">  
1385: <property location="${build.dir}/built-clean.properties" name="built-clean.properties"/>  
1386: <delete file="${built-clean.properties}" quiet="true"/>  
1387: </target>  
1388: <target if="already.built.clean.${basedir}" name="-warn-already-built-clean">  
1389: <echo level="warn" message="Cycle detected: concentrador was already built"/>  
1390: </target>  
1391: <target depends="init,-deps-clean-init" name="deps-clean" unless="no.deps">  
1392: <mkdir dir="${build.dir}"/>  
1393: <touch file="${built-clean.properties}" verbose="false"/>  
1394: <property file="${built-clean.properties}" prefix="already.built.clean."/>  
1395: <antcall target="-warn-already-built-clean"/>  
1396: <propertyfile file="${built-clean.properties}">  
1397: <entry key="${basedir}" value=""/>  
1398: </propertyfile>  
1399: </target>  
1400: <target depends="init" name="-do-clean">  
1401: <delete dir="${build.dir}"/>  
1402: <delete dir="${dist.dir}" followsymlinks="false" includeemptydirs="true"/>  
1403: </target>  
1404: <target name="-post-clean">  
1405: <!-- Empty placeholder for easier customization. -->  
1406: <!-- You can override this target in the ../build.xml file. -->  
1407: </target>  
1408: <target depends="init,deps-clean,-do-clean,-post-clean" description="Clean build products." name="clean"/>  
1409: <target name="-check-call-dep">  
1410: <property file="${call.built.properties}" prefix="already.built."/>  
1411: <condition property="should.call.dep">  
1412: <and>  
1413: <not>  
1414: <isset property="already.built.${call.subproject}"/>  
1415: </not>  
1416: <available file="${call.script}"/>  
1417: </and>  
1418: </condition>  
1419: </target>  
1420: <target depends="-check-call-dep" if="should.call.dep" name="-maybe-call-dep">  
1421: <ant antfile="${call.script}" inheritall="false" target="${call.target}">  
1422: <propertyset>  
1423: <propertyref prefix="transfer."/>  
1424: <mapper from="transfer.\*" to="\*" type="glob"/>  
1425: </propertyset>  
1426: </ant>  
1427: </target>  
1428:</project>

## .\concentrador\nbproject\project.xml

1:<?xml version="1.0" encoding="UTF-8"?>  
 2:<project xmlns="http://www.netbeans.org/ns/project/1">  
 3: <type>org.netbeans.modules.java.j2seproject</type>  
 4: <configuration>  
 5: <buildExtensions xmlns="http://www.netbeans.org/ns/ant-build-extender/1"/>  
 6: <data xmlns="http://www.netbeans.org/ns/j2se-project/3">  
 7: <name>concentrador</name>  
 8: <source-roots>  
 9: <root id="src.dir"/>  
 10: <root id="src.resources.dir"/>  
 11: </source-roots>  
 12: <test-roots>  
 13: <root id="test.src.dir"/>  
 14: </test-roots>  
 15: </data>  
 16: </configuration>  
 17:</project>

## .\mobile\config.xml

1:<?xml version='1.0' encoding='utf-8'?>  
 2:<widget id="br.ufrj.coppetec.contDNIT" version="2.6.1" xmlns="http://www.w3.org/ns/widgets" xmlns:cdv="http://cordova.apache.org/ns/1.0">  
 3: <name>Contagem Mobile</name>  
 4: <description>Software mobile para contagem de tráfego.</description>  
 5: <author email="dev@cordova.apache.org" href="http://cordova.io">Coppetec/UFRJ Team</author>  
 6: <content src="index.html" />  
 7: <access origin="\*" />  
 8: <allow-intent href="http://\*/\*" />  
 9: <allow-intent href="https://\*/\*" />  
 10: <allow-intent href="tel:\*" />  
 11: <allow-intent href="sms:\*" />  
 12: <allow-intent href="mailto:\*" />  
 13: <allow-intent href="geo:\*" />  
 14: <platform name="android">  
 15: <allow-intent href="market:\*" />  
 16: <preference name="SplashScreenDelay" value="5000" />  
 17: <preference name="SplashMaintainAspectRatio" value="false" />  
 18: <preference name="SplashShowOnlyFirstTime" value="false" />  
 19: <splash density="land-hdpi" src="www/res/splash\_screen.png" />  
 20: <splash density="land-ldpi" src="www/res/splash\_screen.png" />  
 21: <splash density="land-mdpi" src="www/res/splash\_screen.png" />  
 22: <splash density="land-xhdpi" src="www/res/splash\_screen.png" />  
 23: <preference name="AutoHideSplashScreen" value="true" />  
 24: </platform>  
 25: <platform name="ios">  
 26: <allow-intent href="itms:\*" />  
 27: <allow-intent href="itms-apps:\*" />  
 28: <splash height="768" src="www/res/splash\_screen.png" width="1024" />  
 29: <preference name="OverrideUserAgent" value="Mozilla/5.0 My Browser" />  
 30: <preference name="AppendUserAgent" value="My Browser" />  
 31: <preference name="target-device" value="tablet" />  
 32: <preference name="SuppressesLongPressGesture" value="true" />  
 33: <preference name="AutoHideSplashScreen" value="false" />  
 34: <preference name="BackupWebStorage" value="none" />  
 35: </platform>  
 36: <platform name="browser">  
 37: <preference name="SplashScreen" value="res/splash\_screen.png" />  
 38: <preference name="SplashScreenDelay" value="3000" />  
 39: <preference name="SplashScreenWidth" value="1024" />  
 40: <preference name="SplashScreenHeight" value="768" />  
 41: <preference name="ShowSplashScreen" value="true" />  
 42: <preference name="AutoHideSplashScreen" value="false" />  
 43: </platform>  
 44: <preference name="Orientation" value="landscape" />  
 45: <preference name="DisallowOverscroll" value="true" />  
 46: <preference name="FadeSplashScreen" value="false" />  
 47: <preference name="FullScreen" value="falses" />  
 48: <preference name="KeyboardDisplayRequiresUserAction" value="false" />  
 49: <preference name="EnableViewportScale" value="true" />  
 50: <engine name="ios" spec="^4.4.0" />  
 51: <plugin name="cordova-plugin-battery-status" spec="^1.2.4" />  
 52: <plugin name="cordova-plugin-cache-clear" spec="^1.3.7" />  
 53: <plugin name="cordova-plugin-device" spec="^1.1.6" />  
 54: <plugin name="cordova-plugin-dialogs" spec="^1.3.3" />  
 55: <plugin name="cordova-plugin-file" spec="^4.3.3" />  
 56: <plugin name="cordova-plugin-splashscreen" spec="^4.0.3" />  
 57: <plugin name="cordova-plugin-statusbar" spec="^2.2.3" />  
 58: <plugin name="cordova-plugin-whitelist" spec="1" />  
 59: <plugin name="cordova-sqlite-storage" spec="^2.0.4" />  
 60: <plugin name="io.phasr.cordova.plugin.itunesfilesharing" spec="~0.0.1" />  
 61:</widget>

## .\mobile\nbproject\build.xml

1:<?xml version="1.0" encoding="UTF-8"?>  
 2:<!--  
 3:DO NOT ALTER OR REMOVE COPYRIGHT NOTICES OR THIS HEADER.  
 4:  
 5:Copyright 1997-2012 Oracle and/or its affiliates. All rights reserved.  
 6:  
 7:Oracle and Java are registered trademarks of Oracle and/or its affiliates.  
 8:Other names may be trademarks of their respective owners.  
 9:  
 10:  
 11:The contents of this file are subject to the terms of either the GNU  
 12:General Public License Version 2 only ("GPL") or the Common  
 13:Development and Distribution License("CDDL") (collectively, the  
 14:"License"). You may not use this file except in compliance with the  
 15:License. You can obtain a copy of the License at  
 16:http://www.netbeans.org/cddl-gplv2.html  
 17:or nbbuild/licenses/CDDL-GPL-2-CP. See the License for the  
 18:specific language governing permissions and limitations under the  
 19:License. When distributing the software, include this License Header  
 20:Notice in each file and include the License file at  
 21:nbbuild/licenses/CDDL-GPL-2-CP. Oracle designates this  
 22:particular file as subject to the "Classpath" exception as provided  
 23:by Oracle in the GPL Version 2 section of the License file that  
 24:accompanied this code. If applicable, add the following below the  
 25:License Header, with the fields enclosed by brackets [] replaced by  
 26:your own identifying information:  
 27:"Portions Copyrighted [year] [name of copyright owner]"  
 28:  
 29:Contributor(s):  
 30:  
 31:The Original Software is NetBeans. The Initial Developer of the Original  
 32:Software is Sun Microsystems, Inc. Portions Copyright 1997-2006 Sun  
 33:Microsystems, Inc. All Rights Reserved.  
 34:  
 35:If you wish your version of this file to be governed by only the CDDL  
 36:or only the GPL Version 2, indicate your decision by adding  
 37:"[Contributor] elects to include this software in this distribution  
 38:under the [CDDL or GPL Version 2] license." If you do not indicate a  
 39:single choice of license, a recipient has the option to distribute  
 40:your version of this file under either the CDDL, the GPL Version 2 or  
 41:to extend the choice of license to its licensees as provided above.  
 42:However, if you add GPL Version 2 code and therefore, elected the GPL  
 43:Version 2 license, then the option applies only if the new code is  
 44:made subject to such option by the copyright holder.  
 45:-->  
 46:<!--  
 47: Generated file; DO NOT EDIT.  
 48:-->  
 49:<project name="mobile" basedir="..">  
 50: <property file="nbproject/configs/${config}.properties" />  
 51:  
 52: <scriptdef name="checkVersion" language="javascript">  
 53: <attribute name="first" />  
 54: <attribute name="property" />  
 55: <![CDATA[  
 56: var first = attributes.get("first");  
 57: if (first >= "3.0.0") {  
 58: project.setProperty(attributes.get("property"), true);  
 59: }  
 60: ]]>  
 61: </scriptdef>  
 62:   
 63: <scriptdef name="forDevice" language="javascript">  
 64: <![CDATA[  
 65: var dev = project.getProperty("device");  
 66: if (dev == "device") {  
 67: project.setProperty("build.for.device", true);  
 68: }  
 69: ]]>  
 70: </scriptdef>  
 71:  
 72: <target name="check-cordova-project">  
 73: <condition property="cordova.project">  
 74: <or>  
 75: <available file=".cordova"/>  
 76: <available file="hooks"/>  
 77: </or>  
 78: </condition>  
 79: </target>  
 80:   
 81: <target name="upgrade-to-cordova-project" depends="check-cordova-project,check-cordova-version" unless="cordova.project">  
 82: <echo level="info" message="${cordova.command} -d create ${java.io.tmpdir}/nb\_temp\_project com.coolappz.${project.name} ${project.name}" />  
 83: <delete dir="${java.io.tmpdir}/nb\_temp\_project"/>  
 84: <exec executable="${cordova.command}" resolveexecutable="true" searchpath="true" failonerror="true" >  
 85: <env key="${cordova.path.key}" path="${cordova.path.value}:${android.sdk.home}/tools:${android.sdk.home}/platform-tools:${jdk.home}/bin:${ant.home}/bin:${jdk.home}/bin"/>  
 86: <env key="JAVA\_HOME" path="${jdk.home}"/>  
 87: <arg value="-d"/>  
 88: <arg value="create"/>   
 89: <arg value="${java.io.tmpdir}/nb\_temp\_project"/>   
 90: <arg value="com.coolappz.${project.name}"/>  
 91: <arg value="${project.name}" />  
 92: </exec>  
 93: <copy todir="." overwrite="true" failonerror="false">  
 94: <fileset dir="${java.io.tmpdir}/nb\_temp\_project"/>  
 95: </copy>  
 96: <delete dir="${java.io.tmpdir}/nb\_temp\_project"/>  
 97: <delete dir="www"/>  
 98: <copy todir="www" failonerror="false" quiet="true" >  
 99: <fileset dir="${site.root}"/>  
 100: </copy>  
 101: </target>  
 102:  
 103: <target name="create-hello-world" depends="check-cordova-version">  
 104: <echo level="info" message="${cordova.command} -d create www\_nb\_temp com.coolappz.${project.name} ${project.name}" />  
 105: <exec executable="${cordova.command}" resolveexecutable="true" searchpath="true" failonerror="true">  
 106: <env key="${cordova.path.key}" path="${cordova.path.value}:${android.sdk.home}/tools:${android.sdk.home}/platform-tools:${jdk.home}/bin:${ant.home}/bin:${jdk.home}/bin"/>  
 107: <env key="JAVA\_HOME" path="${jdk.home}"/>  
 108: <arg value="-d"/>  
 109: <arg value="create"/>   
 110: <arg value="www\_nb\_temp"/>   
 111: <arg value="com.coolappz.${project.name}"/>  
 112: <arg value="${project.name}" />  
 113: </exec>  
 114: <delete dir="www"/>  
 115: <mkdir dir="www"/>  
 116: <move file="www\_nb\_temp/www" tofile="www"/>  
 117: <delete dir="www\_nb\_temp"/>  
 118: <delete file="www/config.xml"/>  
 119: </target>  
 120:   
 121: <taskdef   
 122: classname="org.netbeans.modules.cordova.updatetask.ReadConfigTask"   
 123: classpath="${update.task.jar}"  
 124: name="readconfig"/>  
 125:  
 126: <taskdef   
 127: classname="org.netbeans.modules.cordova.updatetask.PluginTask"   
 128: classpath="${update.task.jar}"  
 129: name="plugintask"/>  
 130:   
 131:   
 132: <target name="check-cordova-version">  
 133: <property environment="env"/>   
 134: <checkVersion first="${cordova.version}" property="cordova.ver.3"/>  
 135: <fail message="Cordova version 3 or greater required." unless="cordova.ver.3"/>  
 136: <readconfig/>  
 137: <forDevice/>  
 138: </target>  
 139:   
 140: <target name="check-android-template">  
 141: <available file="platforms/android" property="android.generated.exists"/>  
 142: </target>  
 143:  
 144: <target name="check-ios-template">  
 145: <available file="platforms/ios" property="ios.generated.exists"/>  
 146: </target>  
 147:   
 148: <target name="create-android" depends="check-android-template,check-cordova-version,upgrade-to-cordova-project" unless="android.generated.exists">  
 149: <echo level="info" message="${cordova.command} -d platform add android"/>  
 150: <exec executable="${cordova.command}" resolveexecutable="true" searchpath="true" failonerror="true">  
 151: <env key="${cordova.path.key}" path="${cordova.path.value}:${android.sdk.home}/tools:${android.sdk.home}/platform-tools:${jdk.home}/bin:${ant.home}/bin:${jdk.home}/bin"/>  
 152: <env key="JAVA\_HOME" path="${jdk.home}"/>  
 153: <arg value="-d"/>  
 154: <arg value="platform"/>   
 155: <arg value="add"/>   
 156: <arg value="android"/>  
 157: </exec>  
 158: </target>  
 159:   
 160: <target name="create-ios" depends="check-ios-template,check-cordova-version,upgrade-to-cordova-project" unless="ios.generated.exists">  
 161: <echo level="info" message="${cordova.command} -d platform add ios"/>  
 162: <exec executable="${cordova.command}" resolveexecutable="true" searchpath="true" dir="." failonerror="true">  
 163: <env key="${cordova.path.key}" path="${cordova.path.value}:${android.sdk.home}/tools:${android.sdk.home}/platform-tools:${jdk.home}/bin:${ant.home}/bin:${jdk.home}/bin"/>  
 164: <env key="JAVA\_HOME" path="${jdk.home}"/>  
 165: <arg value="-d"/>  
 166: <arg value="platform"/>   
 167: <arg value="add"/>   
 168: <arg value="ios"/>  
 169: </exec>  
 170: </target>  
 171:  
 172: <target name="rebuild-ios" depends="clean-ios,build-ios"/>  
 173:  
 174: <target name="build-ios" depends="create-ios,update-plugins,update-ios,build-ios-xcodebuild,build-ios-ipa"/>  
 175:  
 176: <target name="build-ios-xcodebuild">  
 177: <property name="path" location="platforms/ios/build"/>  
 178: <exec executable="xcodebuild" dir="platforms/ios" failonerror="true">  
 179: <arg value="-project"/>  
 180: <arg value="${project.name}.xcodeproj"/>  
 181: <arg value="ARCHS=${ios.build.arch}"/>   
 182: <arg value="-target"/>  
 183: <arg value="${project.name}"/>  
 184: <arg value="-configuration"/>   
 185: <arg value="Release"/>  
 186: <arg value="-sdk"/>   
 187: <arg value="${ios.build.sdk}" />  
 188: <arg value="build"/>  
 189: <arg value="CONFIGURATION\_BUILD\_DIR=${path}"/>  
 190: </exec>  
 191: </target>  
 192:   
 193: <target name="build-ios-ipa" if="build.for.device">  
 194: <exec executable="xcrun" dir="platforms/ios/build" failonerror="true">  
 195: <env key="CODESIGN\_ALLOCATE" value="/Applications/Xcode.app/Contents/Developer/Platforms/iPhoneOS.platform/Developer/usr/bin/codesign\_allocate" />  
 196: <arg value="-sdk"/>  
 197: <arg value="${ios.build.sdk}" />   
 198: <arg value="PackageApplication"/>  
 199: <arg value="-v"/>  
 200: <arg value="${project.name}.app"/>  
 201: <arg value="-o"/>  
 202: <arg value="${basedir}/platforms/ios/build/${project.name}.ipa"/>  
 203: <arg value="--sign"/>  
 204: <arg value="${ios.certificate.name}"/>  
 205: <arg value="--embed"/>  
 206: <arg value="${ios.provisioning.profile}"/>  
 207: </exec>  
 208: <available file="${basedir}/platforms/ios/build/${project.name}.ipa" property="ipa.found"/>  
 209: <fail unless="ipa.found" message="PackageApplication failed."/>  
 210: </target>  
 211:   
 212:  
 213: <target name="sim-ios" depends="build-ios,ios-run-device,ios-run-simulator">  
 214: </target>  
 215:   
 216: <target name="ios-run-device" if="build.for.device">  
 217: <echo>  
 218: Install "${basedir}/platforms/ios/build/${project.name}.ipa" through iTunes and run it.   
 219: </echo>   
 220: <exec executable="open" failonerror="true">  
 221: <arg value="${basedir}/platforms/ios/build/${project.name}.ipa"/>  
 222: </exec>  
 223: </target>  
 224:  
 225: <target name="ios-run-simulator" unless="build.for.device">  
 226: <exec executable="killall" dir="platforms/ios/build">  
 227: <arg value="launchd\_sim"/>  
 228: </exec>  
 229: <exec executable="${ios.sim.exec}" dir="platforms/ios/build">  
 230: <arg line="launch ${project.name}.app ${ios.device.args} --exit"/>  
 231: </exec>   
 232: </target>  
 233:   
 234: <target name="update-plugins">  
 235: <plugintask/>  
 236: </target>   
 237:   
 238: <target name="update-android">  
 239: <echo level="info" message="${cordova.command} prepare android"/>  
 240:  
 241: <exec executable="${cordova.command}" resolveexecutable="true" searchpath="true" failonerror="true">  
 242: <env key="${cordova.path.key}" path="${cordova.path.value}:${android.sdk.home}/tools:${android.sdk.home}/platform-tools:${jdk.home}/bin:${ant.home}/bin:${jdk.home}/bin"/>  
 243: <env key="JAVA\_HOME" path="${jdk.home}"/>  
 244: <arg value="prepare"/>  
 245: <arg value="android"/>  
 246: </exec>  
 247: </target>   
 248:   
 249:   
 250: <target name="update-ios">  
 251: <echo level="info" message="${cordova.command} prepare ios"/>  
 252: <exec executable="${cordova.command}" resolveexecutable="true" searchpath="true" failonerror="true">  
 253: <env key="${cordova.path.key}" path="${cordova.path.value}:${android.sdk.home}/tools:${android.sdk.home}/platform-tools:${jdk.home}/bin:${ant.home}/bin:${jdk.home}/bin"/>  
 254: <env key="JAVA\_HOME" path="${jdk.home}"/>  
 255: <arg value="prepare"/>  
 256: <arg value="ios"/>  
 257: </exec>  
 258: </target>   
 259:   
 260: <target name="rebuild-android" depends="clean-android,build-android"/>   
 261:   
 262: <target name="build-android" depends="create-android,update-plugins">  
 263: <echo level="info" message="${cordova.command} -d build android"/>  
 264: <exec executable="${cordova.command}" resolveexecutable="true" searchpath="true" failonerror="true">  
 265: <env key="${cordova.path.key}" path="${cordova.path.value}:${android.sdk.home}/tools:${android.sdk.home}/platform-tools:${jdk.home}/bin:${ant.home}/bin:${jdk.home}/bin"/>  
 266: <env key="JAVA\_HOME" path="${jdk.home}"/>  
 267: <arg value="-d"/>   
 268: <arg value="build"/>   
 269: <arg value="android"/>   
 270: </exec>  
 271: </target>   
 272:   
 273: <target name="sim-android" depends="create-android,update-plugins">  
 274: <echo level="info" message="${cordova.command} -d ${android.target.device.arg} android"/>  
 275: <exec executable="${cordova.command}" resolveexecutable="true" searchpath="true" failonerror="true">  
 276: <env key="${cordova.path.key}" path="${cordova.path.value}:${android.sdk.home}/tools:${android.sdk.home}/platform-tools:${jdk.home}/bin:${ant.home}/bin:${jdk.home}/bin"/>  
 277: <env key="JAVA\_HOME" path="${jdk.home}"/>  
 278: <arg value="-d"/>  
 279: <arg value="${android.target.device.arg}"/>  
 280: <arg value="android"/>   
 281: </exec>  
 282: </target>  
 283:   
 284: <target name="clean-android" depends="check-android-template" if="android.generated.exists">  
 285: <exec executable="./gradlew" dir="platforms/android">  
 286: <arg value="clean" />  
 287: </exec>  
 288: </target>  
 289:   
 290: <target name="clean-ios" depends="check-ios-template" if="ios.generated.exists">  
 291: <exec executable="platforms/ios/cordova/clean" />  
 292: </target>  
 293:</project>

## .\mobile\nbproject\project.xml

1:<?xml version="1.0" encoding="UTF-8"?>  
 2:<project xmlns="http://www.netbeans.org/ns/project/1">  
 3: <type>org.netbeans.modules.web.clientproject</type>  
 4: <configuration>  
 5: <data xmlns="http://www.netbeans.org/ns/clientside-project/1">  
 6: <name>mobile</name>  
 7: </data>  
 8: </configuration>  
 9:</project>

## .\mobile\nbproject\private\private.xml

1:<?xml version="1.0" encoding="UTF-8"?>  
 2:<project-private xmlns="http://www.netbeans.org/ns/project-private/1">  
 3: <editor-bookmarks xmlns="http://www.netbeans.org/ns/editor-bookmarks/2" lastBookmarkId="0"/>  
 4: <open-files xmlns="http://www.netbeans.org/ns/projectui-open-files/2">  
 5: <group>  
 6: <file>file:/Users/ludes/NetBeansProjects/ipaddnit/mobile/www/js/datas.js</file>  
 7: <file>file:/Users/ludes/NetBeansProjects/ipaddnit/mobile/www/js/index.js</file>  
 8: <file>file:/Users/ludes/NetBeansProjects/ipaddnit/mobile/www/js/logger.js</file>  
 9: </group>  
 10: </open-files>  
 11:</project-private>

## .\mobile\nbproject\private\retriever\catalog.xml

1:<?xml version="1.0" encoding="UTF-8" standalone="no"?>  
 2:<catalog xmlns="urn:oasis:names:tc:entity:xmlns:xml:catalog" prefer="system">  
 3: <system systemId="http://www.w3.org/ns/widgets" uri="www.w3.org/ns/widgets/index.xml"/>  
 4:</catalog>

## .\mobile\nbproject\private\retriever\www.w3.org\ns\widgets\index.xml

1:<?xml version="1.0" encoding="UTF-8"?>  
 2:<?xml-stylesheet type="text/css" href="http://www.w3.org/StyleSheets/TR/base"?>  
 3:<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
 4:"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">  
 5:<html lang="en" xml:lang="en" xmlns="http://www.w3.org/1999/xhtml">  
 6:<head>  
 7: <title>Widgets Namespace</title>  
 8: <link rel="stylesheet" type="text/css"  
 9: href="http://www.w3.org/StyleSheets/TR/base" />  
 10:</head>  
 11:<body>  
 12:<div class="head">  
 13:<p><a href="http://www.w3.org/"><img class="head"  
 14:src="http://www.w3.org/Icons/WWW/w3c\_home" alt="W3C" /></a></p>  
 15:</div>  
 16:<h1>Widgets namespace</h1>  
 17:  
 18:<p>The namespace name  
 19:<code>http://www.w3.org/ns/widgets</code> is intended  
 20:for use with the <a href="http://www.w3.org/TR/widgets/">Widget Packaging and XML Configuration</a> specification. </p>  
 21:<p>This page is maintained by the <a href="http://www.w3.org/2008/webapps/">Web Application Working Group</a> (<a href="http://www.w3.org/2008/webapps/charter/">charter</a>), part of the <a href="http://www.w3.org/2006/rwc/Activity">Rich Web Clients Activity</a> in the W3C <a href="http://www.w3.org/Interaction/">Interaction Domain</a>. Please send comments to <a href="mailto:public-webapps@w3.org">public-webapps@w3.org</a>, the W3C's public email list for issues related to Widgets. <a href="http://lists.w3.org/Archives/Public/public-webapps/">Archives</a> of the mailing list are available. </p>  
 22:<h2>Namespace change policy</h2>  
 23:<p>The <a href="http://www.w3.org/2008/webapps/">Web Application Working Group</a> reserves the right to  
 24: bring additional names from this namespace into active use by  
 25: providing definitions for them in new specifications or  
 26: subsequent editions of existing specifications. The Working  
 27: Group may also make modifications to the definitions of the  
 28: names already in use, although such changes will not normally  
 29: change the well-formedness or validity of existing documents  
 30: or significantly change their meaning.</p>  
 31:<p>All changes to the use of this namespace will be achieved by  
 32: the publication of documents governed by the W3C Process.</p>  
 33:  
 34:<p>For more information about  
 35: <abbr>XML</abbr> namespaces, please refer to  
 36: <cite><a href="http://www.w3.org/TR/REC-xml-names">Namespaces in  
 37:<abbr>XML</abbr></a></cite>.</p>  
 38:  
 39:<hr />  
 40:  
 41:<address>  
 42:<a href="http://datadriven.com.au">Marcos Caceres</a>, W3C Invited Expert  
 43:</address>  
 44:<p><small>Version: Mon 13 Jun 2011 12:20:29 BST</small></p>  
 45:  
 46:</body>  
 47:</html>

## .\mobile\platforms\ios\Contagem Mobile\config.xml

1:<?xml version='1.0' encoding='utf-8'?>  
 2:<widget id="br.ufrj.coppetec.contDNIT" version="2.4.2" xmlns="http://www.w3.org/ns/widgets" xmlns:cdv="http://cordova.apache.org/ns/1.0">  
 3: <feature name="LocalStorage">  
 4: <param name="ios-package" value="CDVLocalStorage" />  
 5: </feature>  
 6: <feature name="HandleOpenUrl">  
 7: <param name="ios-package" value="CDVHandleOpenURL" />  
 8: <param name="onload" value="true" />  
 9: </feature>  
 10: <feature name="IntentAndNavigationFilter">  
 11: <param name="ios-package" value="CDVIntentAndNavigationFilter" />  
 12: <param name="onload" value="true" />  
 13: </feature>  
 14: <feature name="GestureHandler">  
 15: <param name="ios-package" value="CDVGestureHandler" />  
 16: <param name="onload" value="true" />  
 17: </feature>  
 18: <feature name="Notification">  
 19: <param name="ios-package" value="CDVNotification" />  
 20: </feature>  
 21: <feature name="Device">  
 22: <param name="ios-package" value="CDVDevice" />  
 23: </feature>  
 24: <feature name="File">  
 25: <param name="ios-package" value="CDVFile" />  
 26: <param name="onload" value="true" />  
 27: </feature>  
 28: <feature name="SplashScreen">  
 29: <param name="ios-package" value="CDVSplashScreen" />  
 30: <param name="onload" value="true" />  
 31: </feature>  
 32: <feature name="StatusBar">  
 33: <param name="ios-package" value="CDVStatusBar" />  
 34: <param name="onload" value="true" />  
 35: </feature>  
 36: <feature name="SQLitePlugin">  
 37: <param name="ios-package" value="SQLitePlugin" />  
 38: </feature>  
 39: <feature name="Battery">  
 40: <param name="ios-package" value="CDVBattery" />  
 41: </feature>  
 42: <feature name="CacheClear">  
 43: <param name="ios-package" value="CacheClear" />  
 44: </feature>  
 45: <name>Contagem Mobile</name>  
 46: <description>Software mobile para contagem de tráfego.</description>  
 47: <author email="dev@cordova.apache.org" href="http://cordova.io">Coppetec/UFRJ Team</author>  
 48: <content src="index.html" />  
 49: <access origin="\*" />  
 50: <allow-intent href="http://\*/\*" />  
 51: <allow-intent href="https://\*/\*" />  
 52: <allow-intent href="tel:\*" />  
 53: <allow-intent href="sms:\*" />  
 54: <allow-intent href="mailto:\*" />  
 55: <allow-intent href="geo:\*" />  
 56: <allow-intent href="itms:\*" />  
 57: <allow-intent href="itms-apps:\*" />  
 58: <splash height="768" src="www/res/splash\_screen.png" width="1024" />  
 59: <preference name="AllowInlineMediaPlayback" value="false" />  
 60: <preference name="BackupWebStorage" value="none" />  
 61: <preference name="DisallowOverscroll" value="true" />  
 62: <preference name="EnableViewportScale" value="true" />  
 63: <preference name="KeyboardDisplayRequiresUserAction" value="false" />  
 64: <preference name="MediaPlaybackRequiresUserAction" value="false" />  
 65: <preference name="SuppressesIncrementalRendering" value="false" />  
 66: <preference name="SuppressesLongPressGesture" value="true" />  
 67: <preference name="Suppresses3DTouchGesture" value="false" />  
 68: <preference name="GapBetweenPages" value="0" />  
 69: <preference name="PageLength" value="0" />  
 70: <preference name="PaginationBreakingMode" value="page" />  
 71: <preference name="PaginationMode" value="unpaginated" />  
 72: <preference name="StatusBarOverlaysWebView" value="true" />  
 73: <preference name="StatusBarStyle" value="lightcontent" />  
 74: <preference name="Orientation" value="landscape" />  
 75: <preference name="FadeSplashScreen" value="false" />  
 76: <preference name="FullScreen" value="falses" />  
 77: <preference name="OverrideUserAgent" value="Mozilla/5.0 My Browser" />  
 78: <preference name="AppendUserAgent" value="My Browser" />  
 79: <preference name="target-device" value="tablet" />  
 80: <preference name="AutoHideSplashScreen" value="false" />  
 81:</widget>

## .\pnctcheck\pom.xml

1:<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 2: xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 3: <modelVersion>4.0.0</modelVersion>  
 4:  
 5: <groupId>br.ufrj.coppetec.pnct</groupId>  
 6: <artifactId>pnctcheck</artifactId>  
 7: <version>0.0.1-SNAPSHOT</version>  
 8: <packaging>jar</packaging>  
 9:  
 10: <name>pnctcheck</name>  
 11: <url>http://maven.apache.org</url>  
 12:  
 13: <properties>  
 14: <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
 15: <jdk.version>1.8</jdk.version>  
 16: <junit.version>4.12</junit.version>  
 17: <log4j.version>2.6.2</log4j.version>  
 18: <commons-compress.version>1.12</commons-compress.version>  
 19: <commons-io.version>2.5</commons-io.version>  
 20: </properties>  
 21:  
 22: <dependencies>  
 23: <dependency>  
 24: <groupId>junit</groupId>  
 25: <artifactId>junit</artifactId>  
 26: <version>${junit.version}</version>  
 27: <scope>test</scope>  
 28: </dependency>  
 29:  
 30: <dependency>  
 31: <groupId>org.apache.logging.log4j</groupId>  
 32: <artifactId>log4j-api</artifactId>  
 33: <version>${log4j.version}</version>  
 34: </dependency>  
 35: <dependency>  
 36: <groupId>org.apache.logging.log4j</groupId>  
 37: <artifactId>log4j-core</artifactId>  
 38: <version>${log4j.version}</version>  
 39: </dependency>  
 40:  
 41: <!-- https://mvnrepository.com/artifact/org.apache.commons/commons-compress -->  
 42: <dependency>  
 43: <groupId>org.apache.commons</groupId>  
 44: <artifactId>commons-compress</artifactId>  
 45: <version>${commons-compress.version}</version>  
 46: </dependency>  
 47:  
 48: <!-- https://mvnrepository.com/artifact/commons-io/commons-io -->  
 49: <dependency>  
 50: <groupId>commons-io</groupId>  
 51: <artifactId>commons-io</artifactId>  
 52: <version>${commons-io.version}</version>  
 53: </dependency>  
 54:  
 55: </dependencies>  
 56:</project>

## .\concentrador\nbproject\genfiles.properties

1:build.xml.data.CRC32=bf192b2e  
 2:build.xml.script.CRC32=c5184723  
 3:build.xml.stylesheet.CRC32=8064a381@1.79.1.48  
 4:# This file is used by a NetBeans-based IDE to track changes in generated files such as build-impl.xml.  
 5:# Do not edit this file. You may delete it but then the IDE will never regenerate such files for you.  
 6:nbproject/build-impl.xml.data.CRC32=922b2fe7  
 7:nbproject/build-impl.xml.script.CRC32=87d53657  
 8:nbproject/build-impl.xml.stylesheet.CRC32=05530350@1.79.1.48

## .\concentrador\nbproject\project.properties

1:annotation.processing.enabled=true  
 2:annotation.processing.enabled.in.editor=false  
 3:annotation.processing.processors.list=  
 4:annotation.processing.run.all.processors=true  
 5:annotation.processing.source.output=${build.generated.sources.dir}/ap-source-output  
 6:application.title=concentrador  
 7:application.vendor=ludes  
 8:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.expand-tabs=true  
 9:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.indent-shift-width=4  
 10:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.spaces-per-tab=4  
 11:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.tab-size=4  
 12:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.text-limit-width=130  
 13:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.text-line-wrap=none  
 14:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.usedProfile=project  
 15:auxiliary.org-netbeans-modules-editor-indent.text.x-java.CodeStyle.project.expand-tabs=false  
 16:auxiliary.org-netbeans-modules-editor-indent.text.x-java.CodeStyle.project.indent-shift-width=4  
 17:auxiliary.org-netbeans-modules-editor-indent.text.x-java.CodeStyle.project.spaces-per-tab=4  
 18:auxiliary.org-netbeans-modules-editor-indent.text.x-java.CodeStyle.project.tab-size=4  
 19:auxiliary.org-netbeans-modules-editor-indent.text.x-java.CodeStyle.project.text-limit-width=130  
 20:auxiliary.org-netbeans-modules-editor-indent.text.x-java.CodeStyle.project.text-line-wrap=none  
 21:build.classes.dir=${build.dir}/classes  
 22:build.classes.excludes=\*\*/\*.java,\*\*/\*.form  
 23:# This directory is removed when the project is cleaned:  
 24:build.dir=build  
 25:build.generated.dir=${build.dir}/generated  
 26:build.generated.sources.dir=${build.dir}/generated-sources  
 27:# Only compile against the classpath explicitly listed here:  
 28:build.sysclasspath=ignore  
 29:build.test.classes.dir=${build.dir}/test/classes  
 30:build.test.results.dir=${build.dir}/test/results  
 31:# Uncomment to specify the preferred debugger connection transport:  
 32:#debug.transport=dt\_socket  
 33:debug.classpath=\  
 34: ${run.classpath}  
 35:debug.test.classpath=\  
 36: ${run.test.classpath}  
 37:# Files in build.classes.dir which should be excluded from distribution jar  
 38:dist.archive.excludes=  
 39:# This directory is removed when the project is cleaned:  
 40:dist.dir=dist  
 41:dist.jar=${dist.dir}/concentrador.jar  
 42:dist.javadoc.dir=${dist.dir}/javadoc  
 43:endorsed.classpath=  
 44:excludes=  
 45:file.reference.commons-lang3-3.4.jar=lib/commons-lang3-3.4.jar  
 46:file.reference.hamcrest-core-1.3.jar=lib/hamcrest-core-1.3.jar  
 47:file.reference.json-20160212-javadoc.jar=lib/json-20160212-javadoc.jar  
 48:file.reference.json-20160212.jar=lib/json-20160212.jar  
 49:file.reference.junit-4.12.jar=lib/junit-4.12.jar  
 50:file.reference.log4j-api-2.6.2.jar=lib/log4j-api-2.6.2.jar  
 51:file.reference.log4j-core-2.6.2.jar=lib/log4j-core-2.6.2.jar  
 52:file.reference.sqlite-jdbc-3.8.11.2.jar=lib/sqlite-jdbc-3.8.11.2.jar  
 53:file.reference.swingx-all-1.6.4.jar=lib/swingx-all-1.6.4.jar  
 54:includes=\*\*  
 55:jar.compress=false  
 56:javac.classpath=\  
 57: ${file.reference.hamcrest-core-1.3.jar}:\  
 58: ${file.reference.json-20160212-javadoc.jar}:\  
 59: ${file.reference.json-20160212.jar}:\  
 60: ${file.reference.junit-4.12.jar}:\  
 61: ${file.reference.sqlite-jdbc-3.8.11.2.jar}:\  
 62: ${file.reference.log4j-api-2.6.2.jar}:\  
 63: ${file.reference.log4j-core-2.6.2.jar}:\  
 64: ${file.reference.swingx-all-1.6.4.jar}:\  
 65: ${file.reference.commons-lang3-3.4.jar}:\  
 66: ${libs.absolutelayout.classpath}  
 67:# Space-separated list of extra javac options  
 68:javac.compilerargs=  
 69:javac.deprecation=false  
 70:javac.external.vm=true  
 71:javac.processorpath=\  
 72: ${javac.classpath}  
 73:javac.source=1.8  
 74:javac.target=1.8  
 75:javac.test.classpath=\  
 76: ${javac.classpath}:\  
 77: ${build.classes.dir}  
 78:javac.test.processorpath=\  
 79: ${javac.test.classpath}  
 80:javadoc.additionalparam=  
 81:javadoc.author=false  
 82:javadoc.encoding=${source.encoding}  
 83:javadoc.noindex=false  
 84:javadoc.nonavbar=false  
 85:javadoc.notree=false  
 86:javadoc.private=false  
 87:javadoc.splitindex=true  
 88:javadoc.use=true  
 89:javadoc.version=false  
 90:javadoc.windowtitle=  
 91:main.class=br.ufrj.coppetec.concentrador.Concentrador  
 92:manifest.file=manifest.mf  
 93:meta.inf.dir=${src.dir}/META-INF  
 94:mkdist.disabled=false  
 95:platform.active=default\_platform  
 96:run.classpath=\  
 97: ${javac.classpath}:\  
 98: ${build.classes.dir}  
 99:# Space-separated list of JVM arguments used when running the project.  
 100:# You may also define separate properties like run-sys-prop.name=value instead of -Dname=value.  
 101:# To set system properties for unit tests define test-sys-prop.name=value:  
 102:run.jvmargs=  
 103:run.test.classpath=\  
 104: ${javac.test.classpath}:\  
 105: ${build.test.classes.dir}  
 106:source.encoding=UTF-8  
 107:src.dir=src  
 108:src.resources.dir=resources  
 109:test.src.dir=test

## .\concentrador\nbproject\configs\concentrador.properties

## .\mobile\nbproject\plugins.properties

1:# This is a list of plugins installed in your project  
 2:# You can delete or add new plugins  
 3:#  
 4:# Format is following:  
 5:# name.of.plugin=url\_of\_repository  
 6:#  
 7:# For more information see http://cordova.apache.org/blog/releases/2013/07/23/cordova-3.html  
 8:#  
 9:  
 10:# org.apache.cordova.device=https://git-wip-us.apache.org/repos/asf/cordova-plugin-device.git  
 11:# org.apache.cordova.network-information=https://git-wip-us.apache.org/repos/asf/cordova-plugin-network-information.git  
 12:# org.apache.cordova.battery-status=https://git-wip-us.apache.org/repos/asf/cordova-plugin-battery-status.git  
 13:# org.apache.cordova.device-motion=https://git-wip-us.apache.org/repos/asf/cordova-plugin-device-motion.git  
 14:# org.apache.cordova.device-orientation=https://git-wip-us.apache.org/repos/asf/cordova-plugin-device-orientation.git  
 15:# org.apache.cordova.geolocation=https://git-wip-us.apache.org/repos/asf/cordova-plugin-geolocation.git  
 16:# org.apache.cordova.camera=https://git-wip-us.apache.org/repos/asf/cordova-plugin-camera.git  
 17:# org.apache.cordova.media-capture=https://git-wip-us.apache.org/repos/asf/cordova-plugin-media-capture.git  
 18:# org.apache.cordova.media=https://git-wip-us.apache.org/repos/asf/cordova-plugin-media.git  
 19:# org.apache.cordova.file=https://git-wip-us.apache.org/repos/asf/cordova-plugin-file.git  
 20:# org.apache.cordova.file-transfer=https://git-wip-us.apache.org/repos/asf/cordova-plugin-file-transfer.git  
 21:# org.apache.cordova.dialogs=https://git-wip-us.apache.org/repos/asf/cordova-plugin-dialogs.git  
 22:# org.apache.cordova.vibration=https://git-wip-us.apache.org/repos/asf/cordova-plugin-vibration.git  
 23:# org.apache.cordova.contacts=https://git-wip-us.apache.org/repos/asf/cordova-plugin-contacts.git  
 24:# org.apache.cordova.globalization=https://git-wip-us.apache.org/repos/asf/cordova-plugin-globalization.git  
 25:# org.apache.cordova.splashscreen=https://git-wip-us.apache.org/repos/asf/cordova-plugin-splashscreen.git  
 26:# org.apache.cordova.console=https://git-wip-us.apache.org/repos/asf/cordova-plugin-console.git  
 27:

## .\mobile\nbproject\project.properties

1:auxiliary.org-netbeans-modules-cordova.cordova\_5f\_build\_5f\_script\_5f\_version=52  
 2:auxiliary.org-netbeans-modules-cordova.phonegap=false  
 3:auxiliary.org-netbeans-modules-javascript-nodejs.run\_2e\_enabled=false  
 4:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.expand-tabs=false  
 5:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.indent-shift-width=4  
 6:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.spaces-per-tab=4  
 7:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.tab-size=4  
 8:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.text-limit-width=130  
 9:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.project.text-line-wrap=none  
 10:auxiliary.org-netbeans-modules-editor-indent.CodeStyle.usedProfile=project  
 11:browser.autorefresh.Chrome.INTEGRATED=true  
 12:browser.highlightselection.Chrome.INTEGRATED=true  
 13:browser.run=true  
 14:file.reference.browser-www=platforms/browser/www  
 15:file.reference.mobile-www=www  
 16:files.encoding=UTF-8  
 17:site.root.folder=${file.reference.mobile-www}  
 18:start.file=index.html  
 19:web.context.root=/

## .\mobile\nbproject\configs\android.properties

1:device=emulator  
 2:display.name=Android Emulator  
 3:type=android

## .\mobile\nbproject\configs\android\textunderscore 1.properties

1:device=device  
 2:display.name=Android Device  
 3:type=android

## .\mobile\nbproject\configs\ios.properties

1:device=emulator  
 2:display.name=iPhone Simulator  
 3:ios.build.arch=i386  
 4:ios.build.sdk=iphonesimulator9.3  
 5:type=ios

## .\mobile\nbproject\configs\ios\textunderscore 1.properties

1:device=device  
 2:display.name=iPhone Device  
 3:ios.build.arch=armv6 armv7  
 4:ios.build.sdk=iphoneos6.0  
 5:type=ios

## .\mobile\nbproject\private\private.properties

1:browser=Chrome.INTEGRATED  
 2:external.project.url=  
 3:server=INTERNAL

## .\concentrador\src\logins\controle.txt

1:####################################  
 2:# controle #  
 3:# 1 - arquivo antigo #  
 4:# 2 - arquivo novo #  
 5:####################################  
 6:  
 7:=============  
 8:6 ENTR BR-101(A) - ENTR BR-101(B) (P/ALAGOINHAS)   
 9:6 ENTR BR-110(A) - ENTR BR-110(B)/BA-504 (P/ALAGOINHAS)   
 10:=============  
 11:11 JAGUARARI - ENTR BA-220 (SENHOR DO BONFIM)   
 12:11 JAGUARARI - ENTR BA-220 (SENHOR DO BONFIM)   
 13:=============  
 14:14 ENTR BR-226(B)/CE-275(B) (P/PEREIRO) - ENTR BR-404/434 (ICÓ)   
 15:14 ENTR BR-226(B)/CE-275(B) (P/PEREIRO) - ENTR BR-404/434 (ICÓ)   
 16:=============  
 17:15 ENTR ES-245(B) - ENTR ES-440   
 18:15 ENTR ES-245(B) - ENTR ES-440   
 19:=============  
 20:18 KM 15,3 - ENTR BR-484 (P/PONTE SOBRE RIO DOCE)   
 21:18 KM 15,3 - ENTR BR-484 (P/PONTE SOBRE RIO DOCE)   
 22:=============  
 23:20 ENTR BR-457(B)/GO-219 - ENTR GO-020(A)   
 24:20 ENTR BR-457(B)/GO-219 - ENTR GO-020(A)   
 25:=============  
 26:21 ENTR GO-338 (ABADIÂNIA - FIM TRAVESSIA URBANA) - ENTR BR-153(A) (P/ANÁPOLIS)   
 27:21 ENTR GO-338 (ABADIÂNIA - FIM TRAVESSIA URBANA) - ENTR BR-153(A) (P/ANÁPOLIS)   
 28:=============  
 29:23 ENTR BR-154/452(A)/483 - ENTR BR-452(B) (DIV GO/MG) (ITUMBIARA)   
 30:23 ENTR BR-154/452(A)/483 - ENTR BR-452(B) (DIV GO/MG) (ITUMBIARA)   
 31:=============  
 32:24 ENTR BR-060(B)/158(B)/GO-184(B) (P/ESTREITO) - ENTR GO-050(B)   
 33:24 ENTR BR-060(B)/158(B)/GO-184(B) (P/ESTREITO) - ENTR GO-050(B)   
 34:=============  
 35:25 FIM DA DUPLICAÇÃO - ENTR BR-402/MA-110 (BACABEIRA)   
 36:25 FIM DA DUPLICAÇÃO - ENTR BR-402/MA-110 (BACABEIRA)   
 37:=============  
 38:28 ENTR BR-251(A) (CANACÍ) - ENTR BR-135/251(B)/308/365 (MONTES CLAROS)   
 39:28 ENTR BR-251(A) (CANACÍ) - ENTR BR-135/251(B)/308/365 (MONTES CLAROS)   
 40:=============  
 41:30 ENTR MG-410 (P/PONTO DIAMANTE) - ENTR MG-181 (JOÃO PINHEIRO)   
 42:30 ENTR MG-410 (P/PONTO DIAMANTE) - ENTR MG-181 (JOÃO PINHEIRO)   
 43:=============  
 44:34 ENTR BR-365(B)/452(B) - RIO TIJUCO   
 45:34 ENTR BR-365(B)/452(B) - RIO TIJUCO   
 46:=============  
 47:35 ENTR BR-342(B)/418/MG-217 (RIB STO ANTÔNIO) (TEÓFILO OTONI) - ACESSO ITAMBACURI   
 48:35 ENTR BR-342(B)/418/MG-217 (RIB STO ANTÔNIO) (TEÓFILO OTONI) - ACESSO ITAMBACURI   
 49:=============  
 50:36 FIM PISTA DUPLA - DOM CORRÊA   
 51:36 FIM PISTA DUPLA - DOM CORRÊA   
 52:=============  
 53:39 ENTR BR-364(A)/262(A) (P/COMENDADOR GOMES) - ENTR BR-364(B)/262(B) (P/FRUTAL)   
 54:39 ENTR BR-364(A)/262(A) (P/COMENDADOR GOMES) - ENTR BR-364(B)/262(B) (P/FRUTAL)   
 55:=============  
 56:42 ACESSO CARMO DO PARANAIBA - ENTR MG-230   
 57:42 ACESSO CARMO DO PARANAIBA - ENTR MG-230   
 58:=============  
 59:50 ENTR BR-158(B)/MS-395 (P/BRASILÂNDIA) - ENTR MS-453/459 (P/ARAPUÁ)   
 60:50 ENTR BR-158(B)/MS-395 (P/BRASILÂNDIA) - ENTR MS-453/459 (P/ARAPUÁ)   
 61:=============  
 62:61 ENTR PE-450 (P/VERDEJANTE) - ENTR BR-116/361 (SALGUEIRO)   
 63:61 ENTR PE-450 (P/VERDEJANTE) - ENTR BR-116/361 (SALGUEIRO)   
 64:=============  
 65:96 ENTR SC-352(B) (P/TAIÓ) - SÃO CRISTOVÃO DO SUL   
 66:96 ENTR SC-352(B) (P/TAIÓ) - SÃO CRISTOVÃO DO SUL   
 67:=============  
 68:120 ENTR BR-262(B) - ENTR BR-050(B)   
 69:120 ENTR BR-262(B) - ENTR BR-464(B) (P/CONQUISTA)   
 70:=============  
 71:124 ENTR PE-270 (P/BUÍQUE) - ENTR BR-110 (CRUZEIRO DO NORDESTE)   
 72:124 ENTR PE-270 (P/BUÍQUE) - ENTR BR-110 (CRUZEIRO DO NORDESTE)   
 73:=============  
 74:125 ENTR PE-218 (P/BOM CONSELHO) - ENTR PE-187 (P/PALMEIRINA)   
 75:125 ENTR PE-218 (P/BOM CONSELHO) - ENTR PE-187 (P/PALMEIRINA)   
 76:=============  
 77:127 ENTR BA-493 (ITATIM) - ENTR BA-245   
 78:127 ENTR BA-493 (ITATIM) - ENTR BA-245   
 79:=============  
 80:130 DIV BA/MG - ENTR BR-251(A) (P/SALINAS)   
 81:130 DIV BA/MG - ENTR BR-251(A) (P/SALINAS)   
 82:=============  
 83:133 ENTR BR-235 - ENTR BA-220 (EUCLIDES DA CUNHA)   
 84:133 ENTR BR-235 - ENTR BA-220 (EUCLIDES DA CUNHA)   
 85:=============  
 86:135 ENTR CE-266 (SUCESSO) - ENTR BR-226/404 (CRATEÚS)   
 87:135 ENTR CE-266 (SUCESSO) - ENTR BR-226/404 (CRATEÚS)   
 88:=============  
 89:138 ENTR CE-456 (P/TARGINOS) - ENTR CE-257 (CANINDÉ)   
 90:138 ENTR CE-456 (P/TARGINOS) - ENTR CE-257 (CANINDÉ)   
 91:=============  
 92:139 ENTR CE-341 (CROATÁ) - ENTR CE-162 (SÃO LUÍS DO CURU)   
 93:139 ENTR CE-341 (CROATÁ) - ENTR CE-162 (SÃO LUÍS DO CURU)   
 94:=============  
 95:147 ENTR GO-114 (P/FLORES DE GOIÁS) - SANTA MARIA   
 96:147 ENTR GO-114 (P/FLORES DE GOIÁS) - SANTA MARIA   
 97:=============  
 98:152 ENTR BR-153(A)/GO-342(B) - ENTR BR-153(B)/GO-237 (URUAÇÚ)   
 99:152 ENTR BR-153(A)/GO-342(B) - ENTR BR-153(B)/GO-237 (URUAÇÚ)   
 100:=============  
 101:153 ENTR GO-210(B) (P/PANAMÁ) - ENTR BR-154/452(A)/483   
 102:153 ENTR GO-210(B) (P/PANAMÁ) - ENTR BR-154/452(A)/483   
 103:=============  
 104:157 ENTR MA-006(A) (COCALINHO) - ENTR MA-216 (BOM JARDIM)   
 105:157 ENTR MA-006(A) (COCALINHO) - ENTR MA-216 (BOM JARDIM)   
 106:=============  
 107:163 ENTR MA-132/270 (COLINAS) - ENTR MA-134 (PEIXE)   
 108:163 ENTR MA-132/270 (COLINAS) - ENTR MA-134 (PEIXE)   
 109:=============  
 110:165 ENTR MG-434 - ENTR MG-435 (P/CAETÉ)   
 111:165 ENTR MG-434 - ENTR MG-435 (P/CAETÉ)   
 112:=============  
 113:166 ENTR MG-275 (P/CARANDAÍ) - ACESSO ALTO DOCE (INÍCIO PISTA DUPLA)   
 114:166 ENTR MG-275 (P/CARANDAÍ) - ACESSO ALTO DOCE (INÍCIO PISTA DUPLA)   
 115:=============  
 116:178 ENTR BR-116(B) - ENTR MG-126 (BICAS)   
 117:178 ENTR BR-116(B) - ENTR MG-126 (BICAS)   
 118:=============  
 119:184 ENTR BR-462(B) - ENTR BR-262   
 120:184 ENTR BR-462(B) - ENTR BR-262   
 121:=============  
 122:185 ENTR BR-452 (P/UBERLÂNDIA) - ENTR BR-462 (PERDIZES)   
 123:185 ENTR BR-452 (P/UBERLÂNDIA) - ENTR BR-462 (PERDIZES)   
 124:=============  
 125:187 ENTR MG-354(B) - ENTR BR-265(A)   
 126:187 ENTR MG-354(B) - ENTR BR-265(A)   
 127:=============  
 128:190 ACESSO FORMIGA - ENTR MG-164 (CANDEIAS)   
 129:190 ACESSO FORMIGA - ENTR MG-164 (CANDEIAS)   
 130:=============  
 131:193 ACESSO GUIMARÂNIA - ENTR MG-188   
 132:193 ACESSO GUIMARÂNIA - ENTR MG-188   
 133:=============  
 134:194 ENTR MG-410 - PRESIDENTE OLEGÁRIO   
 135:194 ENTR MG-410 - PRESIDENTE OLEGÁRIO   
 136:=============  
 137:195 ENTR MG-060 (P/SÃO GONÇALO DO ABAETÉ) - ACESSO GALENA   
 138:195 ENTR MG-060 (P/SÃO GONÇALO DO ABAETÉ) - ACESSO GALENA   
 139:=============  
 140:196 ENTR BR-154(B)/461/464 (ITUIUTABA) - ACESSO GURINHATÃ   
 141:196 ENTR BR-154(B)/461/464 (ITUIUTABA) - ACESSO GURINHATÃ   
 142:=============  
 143:203 ACESSO CONCEIÇÃO DO RIO VERDE - ENTR BR-460   
 144:203 ACESSO CONCEIÇÃO DO RIO VERDE - ENTR BR-460   
 145:=============  
 146:218 BARRA DE SANTANA - ENTR PB-196 (P/RIACHO DE SANTO ANTÔNIO)   
 147:218 BARRA DE SANTANA - ENTR PB-196 (P/RIACHO DE SANTO ANTÔNIO)   
 148:=============  
 149:222 ENTR PE-590 (P/IPUBÍ) - ENTR BR-122 (OURICURÍ)   
 150:222 ENTR PE-590 (P/IPUBÍ) - ENTR BR-122 (OURICURÍ)   
 151:=============  
 152:225 RAJADA - ENTR BR-235(A)   
 153:225 RAJADA - ENTR BR-235(A)   
 154:=============  
 155:226 ENTR PI-394 - ENTR PI-140 (CANTO DO BURITI)   
 156:226 ENTR PI-394 - ENTR PI-140 (CANTO DO BURITI)   
 157:=============  
 158:228 ACESSO OESTE SOBRAL - ENTR BR-403(B)/CE-183   
 159:228 ACESSO OESTE SOBRAL - ENTR BR-403(B)/CE-183   
 160:=============  
 161:242 ENTR RJ-196 - ENTR RJ-182   
 162:242 ENTR RJ-196 - ENTR RJ-182   
 163:=============  
 164:246 ENTR BR-494 (ARANTINA) - AIURUOCA   
 165:246 ENTR BR-494 (ARANTINA) - AIURUOCA   
 166:=============  
 167:247 ENTR RJ-198(B) - ENTR RJ-200 (MONTE ALEGRE)   
 168:247 ENTR RJ-198(B) - ENTR RJ-200 (MONTE ALEGRE)   
 169:=============  
 170:248 ENTR RJ-202 - ENTR RJ-234 (P/PENEDO)   
 171:248 ENTR RJ-202 - ENTR RJ-234 (P/PENEDO)   
 172:=============  
 173:258 ENTR BR-421 (ARIQUEMES) - ENTR RO-459(A) (ALTO PARAISO)   
 174:258 ENTR BR-421 (ARIQUEMES) - ENTR RO-459(A) (ALTO PARAISO)   
 175:=============  
 176:261 ENTR RO-391 (POSTO GUAPORÉ) - MARCO RONDON   
 177:261 ENTR RO-391 (POSTO GUAPORÉ) - MARCO RONDON   
 178:=============  
 179:277 ENTR BA-392 (ANTAS) - ENTR BA-220 (CÍCERO DANTAS)   
 180:277 ENTR BA-392 (ANTAS) - ENTR BA-220 (CÍCERO DANTAS)   
 181:=============  
 182:289 ENTR BR-010(A) (PORTO FRANCO) - ENTR BR-010(B)/230(A)/MA-138 (ESTREITO)   
 183:289 ENTR BR-226(A)/230(B)/MA-138 (ESTREITO) - ENTR BR-226(B) (PORTO FRANCO)   
 184:=============  
 185:292 ENTR BR-020(A)/135(B)/BA-020 - ENTR BR-020(B)   
 186:292 ENTR BR-242(A) - ENTR BR-135(A)/242(B) (BARREIRAS)   
 187:=============