**DataBase**

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Main;

import com.mysql.cj.jdbc.MysqlDataSource;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.ResultSetMetaData;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

public class DataBase {

/\*\*

\* @return the photo

\*/

public static String getPhotoPatn() {

return photoPath;

}

/\*\*

\* @param aMySQL the MySQL to set

\*/

public static void setMySQL(String aMySQL) {

MySQL = aMySQL;

}

/\*\*

\* @return the MySQL

\*/

public static String getMySQL() {

return MySQL;

}

/\*\*

\* @return the namePerson

\*/

public static String getNamePerson() {

return namePerson;

}

/\*\*

\* @param aNamePerson the namePerson to set

\*/

public static void setNamePerson(String aNamePerson) {

namePerson = aNamePerson;

}

/\*\*

\* @return the array

\*/

public static String[] getArray() {

return array;

}

private String tbl = "users";

private Statement s = null;

SettingsForm o = new SettingsForm();

private String addName = "Такого нема";

public ArrayList<String> list = new ArrayList<>();

public static String[] array;

private int idBranch = 0;

private static String nameBranch = "";

private static String namePerson = "";

private static String MySQL = "";

private static String photoPath = "";

/\*\*

\* @return the addName

\*/

public String getAddName() {

return addName;

}

/\*\*

\* @param addName the addName to set

\*/

public void setAddName(String addName) {

this.addName = addName;

}

/\*\*

\* @return the tbl\_users

\*/

public String getTbl\_users() {

return tbl\_users;

}

private String server = o.getServer();

private String port = o.getPort();

**q**

@Override

protected void finalize() {

if (conn != null) {

try {

conn.close();

} catch (Exception e) {

}

}

}

/\*\*

\* // метод поверне об’єкт НАБІР ЗАПИСІВ який потiм буде використано для

\* виведення в таблицю на формі

\*

\* @param table Ім'я таблиці

\* @return Xrez Результат

\*/

public ResultSet GetDataSet(String table) {

getCon();

ResultSet Xrez = null;

try {

if (s == null) {

s = (Statement) conn.createStatement();

}

if (table == "Door") {

Xrez = s.executeQuery("SELECT name, appointment, type, style, coating, size, material " + "FROM " + table + " ORDER BY name ASC");

} else {

Xrez = s.executeQuery("SELECT name " + "FROM " + table + " ORDER BY name ASC");

}

return Xrez;

} catch (SQLException ex) {

Logger.getLogger(DataB.class.getName()).log(Level.SEVERE, null, ex);

return Xrez;

}

}

/\*\*

\* Пошук

\*

\* @param table ім'я тамблиці

\* @param name що шукати

\* @return

\*/

public ResultSet Search(String table, String name) {

getCon();

ResultSet Xrez = null;

try {

if (s == null) {

s = (Statement) conn.createStatement();

}

if(table=="Door")

{

Xrez = s.executeQuery("SELECT name, appointment, type, style, coating, size, material " + "FROM " + table + " WHERE name LIKE '%" + name + "%' "

+ " OR appointment LIKE '%" + name + "%'"

+ " OR type LIKE '%" + name + "%'"

+ " OR style LIKE '%" + name + "%'"

+ " OR coating LIKE '%" + name + "%'"

+ " OR size LIKE '%" + name + "%'"

+ " OR material LIKE '%" + name + "%'"

+ "ORDER BY name ASC;");

}

else

{

Xrez = s.executeQuery("SELECT name " + "FROM " + table + " WHERE name LIKE '%" + name + "%' "

+ "ORDER BY name ASC;");

}

return Xrez;

} catch (SQLException ex) {

Logger.getLogger(DataB.class.getName()).log(Level.SEVERE, null, ex);

return Xrez;

}

}

/\*\*

\*

\* @param nameTbl ім'я таблиці

\* @param name ім'я дверей

\* @param appoint призначення

\* @param type тип

\* @param style стиль

\* @param coating покриття

\* @param size розміп

\* @param materia матеріал

\*/

public void InsertData(String nameTbl, String name, String appoint, String type, String style, String coating, String size, String materia) {

try {

//Connect();

getCon();

String mySQLquery = "";

mySQLquery

= "INSERT INTO `" + nameTbl + "` "

+ "(`name`,`appointment`,`type`,`style`,`coating`,`size`,`material`) "

+ "VALUES (?,?,?,?,?,?,?);";

PreparedStatement X = (PreparedStatement) conn.prepareStatement(mySQLquery);

X.setString(1, name);

X.setString(2, appoint);

X.setString(3, type);

X.setString(4, style);

X.setString(5, coating);

X.setString(6, size);

X.setString(7, materia);

X.execute();

} catch (Exception e) {

System.out.println(e.fillInStackTrace());

}

}

/\*\*

\* Видалення

\*

\* @param id Код

\* @param Table Ім'я таблиці

\* @param nameID Що видалити

\*/

public void del(String Table, String pole, String name) {

// Connect();

getCon();

String mySQLquery = "";

try {

mySQLquery

= "DELETE FROM `" + Table + "` "

+ "WHERE `" + pole + "`= '" + name + "';";

conn.prepareStatement(mySQLquery);

PreparedStatement X = (PreparedStatement) conn.prepareStatement(mySQLquery);

X.execute();

} catch (Exception e) {

System.out.println(e.fillInStackTrace() + "\n" + mySQLquery);

}

}

/\*\*

\*

\* @param nameTbl ім'я таблиці

\* @param name ім'я дверей

\* @param appoint призначення

\* @param type тип

\* @param style стиль

\* @param coating покриття

\* @param size розміп

\* @param materia матеріал

\*/

public void Update(String nameTbl, String name, String appoint, String type, String style, String coating, String size, String materia) {

try {

getCon();

String mySQLquery = "";

mySQLquery

= "UPDATE `" + nameTbl + "` "

+ "SET name=?, appointment=?, type=?, style=?, coating=?, size=?, material=? WHERE `name` = '" + frmShow.getNames() + "';";

PreparedStatement X = (PreparedStatement) conn.prepareStatement(mySQLquery);

X.setString(1, name);

X.setString(2, appoint);

X.setString(3, type);

X.setString(4, style);

X.setString(5, coating);

X.setString(6, size);

X.setString(7, materia);

X.execute();

} catch (Exception e) {

System.out.println(e.fillInStackTrace());

}

}

/\*\*

\* Отримання інфи для заповнення випадаючого списку

\*

\* @param code Текстовий запит

\* @return

\*/

public ResultSet GetParam(String code) {

//Connect();

getCon();

list.clear();

array = null;

ResultSet Xrez = null;

try {

if (s == null) {

try {

s = (Statement) conn.createStatement();

} catch (Exception ex) {

}

}

Xrez = s.executeQuery(code);

return Xrez;

} catch (SQLException ex) {

Logger.getLogger(DataB.class.getName()).log(Level.SEVERE, null, ex);

return Xrez;

}

}

/\*\*

\*

\* @param code Повернення повтору

\* @return

\*/

public ResultSet GetInd(String code) {

// Connect();

getCon();

ResultSet Xrez = null;

try {

if (s == null) {

s = (Statement) conn.createStatement();

}

Xrez = s.executeQuery(code);

while (Xrez.next()) {

String name = Xrez.getString(1);

setAddName(name);

}

return Xrez;

} catch (SQLException ex) {

Logger.getLogger(DataB.class.getName()).log(Level.SEVERE, null, ex);

return Xrez;

}

}

/\*\*

\* Отримання повторі

\*

\* @param nameCol ім'я стовпця

\* @param value значення

\* @param table таблиця

\* @return

\*/

public ResultSet GetRep(String nameCol, String value, String table) {

// Connect();

getCon();

ResultSet Xrez = null;

try {

if (s == null) {

s = (Statement) conn.createStatement();

}

Xrez = s.executeQuery("SELECT " + nameCol + " FROM " + table + " WHERE " + nameCol + " = '" + value + "'");

while (Xrez.next()) {

String name = Xrez.getString(1);

setAddName(name);

}

return Xrez;

} catch (SQLException ex) {

Logger.getLogger(DataB.class.getName()).log(Level.SEVERE, null, ex);

return Xrez;

}

}

/\*\*

\* Вставка фільтрів

\*

\* @param table таблиця

\* @param nameCol ім'я стовпя

\* @param value значення

\*/

public void InsertDataFilter(String table, String nameCol, String value) {

try {

//Connect();

getCon();

String mySQLquery = "";

mySQLquery

= "INSERT INTO `" + table + "` "

+ "(`" + nameCol + "`) "

+ "VALUES (?);";

PreparedStatement X = (PreparedStatement) conn.prepareStatement(mySQLquery);

X.setString(1, value);

X.execute();

} catch (Exception e) {

System.out.println(e.fillInStackTrace());

}

}

/\*\*

\* Оновлення фільтра

\*

\* @param table таблиця

\* @param nameCol ім'я стовпця

\* @param value значення

\*/

public void UpdateFilter(String table, String nameCol, String value) {

try {

getCon();

String mySQLquery = "";

mySQLquery

= "UPDATE `" + table + "` "

+ "SET name=? WHERE `name` = '" + frmShow.getNames() + "';";

PreparedStatement X = (PreparedStatement) conn.prepareStatement(mySQLquery);

X.setString(1, value);

X.execute();

} catch (Exception e) {

System.out.println(e.fillInStackTrace());

}

}

}

MoveMouseListener

package Main;

import java.awt.\*;

import java.awt.event.MouseEvent;

import java.awt.event.MouseListener;

import java.awt.event.MouseMotionListener;

import javax.swing.\*;

public class MoveMouseListener implements MouseListener, MouseMotionListener {

JComponent target;

JFrame frame;

Point start\_drag;

Point start\_loc;

public MoveMouseListener(JComponent target, JFrame frame) {

this.target = target;

this.frame = frame;

}

public static JFrame getFrame(Container target) {

if (target instanceof JFrame) {

return (JFrame) target;

}

return getFrame(target.getParent());

}

public void mouseClicked(MouseEvent e) {}

public void mouseEntered(MouseEvent e) {}

public void mouseExited(MouseEvent e) {}

public void mousePressed(MouseEvent e) {

this.start\_drag = this.getScreenLocation(e);

this.start\_loc = this.getFrame(this.target).getLocation( );

}

public void mouseReleased(MouseEvent e) {}

public void mouseMoved(MouseEvent e) {}

public void mouseDragged(MouseEvent e) {

Point current = this.getScreenLocation(e);

Point offset = new Point(

(int)current.getX( )-(int)start\_drag.getX( ),

(int)current.getY( )-(int)start\_drag.getY( ));

JFrame frame = this.getFrame(target);

Point new\_location = new Point(

(int)(this.start\_loc.getX( )+offset.getX( )),

(int)(this.start\_loc.getY( )+offset.getY( )));

frame.setLocation(new\_location);

}

Point getScreenLocation(MouseEvent e) {

Point cursor = e.getPoint( );

Point target\_location = this.target.getLocationOnScreen( );

return new Point(

(int)(target\_location.getX( )+cursor.getX( )),

(int)(target\_location.getY( )+cursor.getY( )));

}

}

Main

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Main;

import java.awt.Color;

import java.io.IOException;

import java.sql.Connection;

import javax.swing.JOptionPane;

/\*\*

\*

\* @author ximik

\*/

public class Main extends javax.swing.JFrame {

/\*\*

\* @param status the status to set

\*/

public static void setStatus(int statuss) {

status = statuss;

}

/\*\*

\* Creates new form StartForm

\*/

private static int status = 0;

private static DataB a = new DataB();

static private Connection conn = null;

public Main() {

initComponents();

MoveMouseListener mml = new MoveMouseListener(pnlHead, this);

pnlHead.addMouseListener(mml);

pnlHead.addMouseMotionListener(mml);

this.setLocationRelativeTo(null);

try

{

Settings s = new Settings();

frmShow f = new frmShow();

}

catch (Exception ex)

{

}

PlugIn();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

pnlHead = new javax.swing.JPanel();

btnExit = new javax.swing.JButton();

jLabel3 = new javax.swing.JLabel();

jLabel1 = new javax.swing.JLabel();

jButton2 = new javax.swing.JButton();

btnUpdate = new javax.swing.JButton();

jButton1 = new javax.swing.JButton();

pnlActivv = new javax.swing.JPanel();

jLabel2 = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setBackground(new java.awt.Color(255, 0, 51));

setUndecorated(true);

addWindowListener(new java.awt.event.WindowAdapter() {

public void windowOpened(java.awt.event.WindowEvent evt) {

formWindowOpened(evt);

}

});

jPanel1.setBackground(new java.awt.Color(143, 93, 100));

jPanel1.setForeground(new java.awt.Color(153, 0, 51));

pnlHead.setBackground(new java.awt.Color(143, 36, 72));

btnExit.setBackground(new java.awt.Color(143, 36, 72));

btnExit.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/close.png"))); // NOI18N

btnExit.setBorder(null);

btnExit.setBorderPainted(false);

btnExit.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnExitActionPerformed(evt);

}

});

jLabel3.setFont(new java.awt.Font("Times New Roman", 0, 30)); // NOI18N

jLabel3.setForeground(new java.awt.Color(255, 255, 255));

jLabel3.setText("Галерея дверей");

javax.swing.GroupLayout pnlHeadLayout = new javax.swing.GroupLayout(pnlHead);

pnlHead.setLayout(pnlHeadLayout);

pnlHeadLayout.setHorizontalGroup(

pnlHeadLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlHeadLayout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(btnExit, javax.swing.GroupLayout.PREFERRED\_SIZE, 45, javax.swing.GroupLayout.PREFERRED\_SIZE))

);

pnlHeadLayout.setVerticalGroup(

pnlHeadLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(btnExit, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT\_SIZE, 43, Short.MAX\_VALUE)

);

jLabel1.setFont(new java.awt.Font("Times New Roman", 0, 36)); // NOI18N

jLabel1.setForeground(new java.awt.Color(255, 255, 255));

jLabel1.setText("Розробник: Магей Юрій");

jButton2.setBackground(new java.awt.Color(143, 36, 72));

jButton2.setFont(new java.awt.Font("Times New Roman", 0, 33)); // NOI18N

jButton2.setForeground(new java.awt.Color(255, 255, 255));

jButton2.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/cogwheel.png"))); // NOI18N

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

btnUpdate.setBackground(new java.awt.Color(143, 36, 72));

btnUpdate.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N

btnUpdate.setForeground(new java.awt.Color(255, 255, 255));

btnUpdate.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/update-2.png"))); // NOI18N

btnUpdate.setText("Повторити");

btnUpdate.setBorder(null);

btnUpdate.setBorderPainted(false);

btnUpdate.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnUpdateActionPerformed(evt);

}

});

jButton1.setBackground(new java.awt.Color(143, 36, 72));

jButton1.setFont(new java.awt.Font("Times New Roman", 0, 48)); // NOI18N

jButton1.setForeground(new java.awt.Color(255, 255, 255));

jButton1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/door.png"))); // NOI18N

jButton1.setText("Почати");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

javax.swing.GroupLayout pnlActivvLayout = new javax.swing.GroupLayout(pnlActivv);

pnlActivv.setLayout(pnlActivvLayout);

pnlActivvLayout.setHorizontalGroup(

pnlActivvLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGap(0, 45, Short.MAX\_VALUE)

);

pnlActivvLayout.setVerticalGroup(

pnlActivvLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGap(0, 0, Short.MAX\_VALUE)

);

jLabel2.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jLabel2.setForeground(new java.awt.Color(255, 255, 255));

jLabel2.setText("Стан підключення");

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(pnlHead, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel1)

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(jLabel2)

.addGap(24, 24, 24)

.addComponent(pnlActivv, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(btnUpdate, javax.swing.GroupLayout.PREFERRED\_SIZE, 137, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jButton2, javax.swing.GroupLayout.PREFERRED\_SIZE, 80, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton1)

.addGap(140, 140, 140))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(pnlHead, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 42, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(pnlActivv, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(btnUpdate, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton2, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jLabel1))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

pack();

}// </editor-fold>

private void btnExitActionPerformed(java.awt.event.ActionEvent evt) {

System.exit(0);

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

if(a.getCon()==null)

{

}

else

{

frmShow main = new frmShow();

main.setVisible(true);

this.dispose();

}

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

if (status == 0) {

Settings s = new Settings();

try {

s.Read();

} catch (IOException ex) {

}

s.setVisible(true);

setStatus(1);

}

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

try {

PlugIn();

} catch (Exception ex) {

JOptionPane.showMessageDialog(this, "Помилка підключення");

// btnUpdate.setVisible(true);

}

}

private void btnUpdateActionPerformed(java.awt.event.ActionEvent evt) {

try {

// Set o = new Set();

PlugIn();

} catch (Exception ex) {

JOptionPane.showMessageDialog(this, "Помилка підключення");

btnUpdate.setVisible(true);

}

}

/\*\*

\* Підключення

\*/

static void PlugIn() {

Settings o = new Settings();

Main.conn=null;

if (Main.conn == null) {

a.finalize();

a = null;

a = new DataB();

conn = a.Connect();

if (conn == null) {

pnlActivv.setBackground(Color.red);

btnUpdate.setVisible(true);

} else {

pnlActivv.setBackground(Color.green);

btnUpdate.setVisible(false);

// frmView.GetData("works");

}

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Main.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Main.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Main.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Main.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new Main().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton btnExit;

public static javax.swing.JButton btnUpdate;

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JLabel jLabel1;

public static javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JPanel jPanel1;

public static javax.swing.JPanel pnlActivv;

private javax.swing.JPanel pnlHead;

// End of variables declaration

}

AddP

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Main;

import java.awt.event.KeyEvent;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.text.NumberFormat;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

/\*\*

\*

\* @author ximik

\*/

public class AddP extends javax.swing.JFrame {

/\*\*

\* @return the variantStart

\*/

public static int getVariantStart() {

return variantStart;

}

/\*\*

\* @param aVariantStart the variantStart to set

\*/

public static void setVariantStart(int aVariantStart) {

variantStart = aVariantStart;

}

DataB a = new DataB();

ResultSet X = null;

private static int variantStart = 0;

/\*\*

\* Creates new form frmAdd

\*/

public AddP() {

initComponents();

MoveMouseListener mml = new MoveMouseListener(pnlHead, this);

pnlHead.addMouseListener(mml);

pnlHead.addMouseMotionListener(mml);

this.setLocationRelativeTo(null);

if (variantStart == 0) {

lblTitle.setText("Додати");

getJCM();

} else if (variantStart == 1) //Редагування

{

lblTitle.setText("Редагувати");

getJCM();

X = null;

X = a.GetParam("SELECT \* FROM Door WHERE name = " + "'" + frmShow.getNames() + "';");

try {

while (X.next()) {

String name = X.getString(2);

String app = X.getString(3);

String type = X.getString(4);

String style = X.getString(5);

String coating = X.getString(6);

String size = X.getString(7);

String mt = X.getString(8);

txtName.setText(name);

txtApp.setText(app);

jcmType.setSelectedItem(type);

jcmStyle.setSelectedItem(style);

jcmCoating.setSelectedItem(coating);

txtSize.setText(size);

jcmMaterial.setSelectedItem(mt);

}

} catch (SQLException ex) {

Logger.getLogger(AddP.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

void getJCM() {

X = null;

jcmMaterial.removeAllItems();

X = a.GetParam("SELECT name FROM Material;");

try {

while (X.next()) {

String g = X.getString(1);

jcmMaterial.addItem(g);

}

} catch (SQLException ex) {

Logger.getLogger(AddP.class.getName()).log(Level.SEVERE, null, ex);

}

X = null;

jcmCoating.removeAllItems();

X = a.GetParam("SELECT name FROM Coating;");

try {

while (X.next()) {

String g = X.getString(1);

jcmCoating.addItem(g);

}

} catch (SQLException ex) {

Logger.getLogger(AddP.class.getName()).log(Level.SEVERE, null, ex);

}

X = null;

jcmStyle.removeAllItems();

X = a.GetParam("SELECT name FROM Style;");

try {

while (X.next()) {

String g = X.getString(1);

jcmStyle.addItem(g);

}

} catch (SQLException ex) {

Logger.getLogger(AddP.class.getName()).log(Level.SEVERE, null, ex);

}

X = null;

jcmType.removeAllItems();

X = a.GetParam("SELECT name FROM Type;");

try {

while (X.next()) {

String g = X.getString(1);

jcmType.addItem(g);

}

} catch (SQLException ex) {

Logger.getLogger(AddP.class.getName()).log(Level.SEVERE, null, ex);

}

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

pnlMain = new javax.swing.JPanel();

pnlHead = new javax.swing.JPanel();

lblTitle = new javax.swing.JLabel();

btnExit = new javax.swing.JButton();

txtName = new javax.swing.JTextField();

jLabel9 = new javax.swing.JLabel();

btnSave = new javax.swing.JButton();

jLabel17 = new javax.swing.JLabel();

txtApp = new javax.swing.JTextField();

jLabel18 = new javax.swing.JLabel();

jLabel19 = new javax.swing.JLabel();

jLabel20 = new javax.swing.JLabel();

jLabel21 = new javax.swing.JLabel();

txtSize = new javax.swing.JTextField();

jLabel22 = new javax.swing.JLabel();

jcmMaterial = new javax.swing.JComboBox<>();

jcmStyle = new javax.swing.JComboBox<>();

jcmType = new javax.swing.JComboBox<>();

jcmCoating = new javax.swing.JComboBox<>();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setUndecorated(true);

pnlMain.setBackground(new java.awt.Color(143, 93, 100));

pnlMain.setForeground(new java.awt.Color(102, 204, 255));

pnlHead.setBackground(new java.awt.Color(143, 36, 72));

lblTitle.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

lblTitle.setForeground(new java.awt.Color(255, 255, 255));

lblTitle.setText("Додати");

btnExit.setBackground(new java.awt.Color(143, 36, 72));

btnExit.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/close.png"))); // NOI18N

btnExit.setBorder(null);

btnExit.setBorderPainted(false);

btnExit.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnExitActionPerformed(evt);

}

});

javax.swing.GroupLayout pnlHeadLayout = new javax.swing.GroupLayout(pnlHead);

pnlHead.setLayout(pnlHeadLayout);

pnlHeadLayout.setHorizontalGroup(

pnlHeadLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlHeadLayout.createSequentialGroup()

.addContainerGap()

.addComponent(lblTitle)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(btnExit, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE))

);

pnlHeadLayout.setVerticalGroup(

pnlHeadLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlHeadLayout.createSequentialGroup()

.addContainerGap()

.addComponent(lblTitle, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addContainerGap())

.addComponent(btnExit, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

txtName.setBackground(new java.awt.Color(143, 93, 100));

txtName.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

txtName.setForeground(new java.awt.Color(255, 255, 255));

txtName.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyPressed(java.awt.event.KeyEvent evt) {

txtNameKeyPressed(evt);

}

public void keyTyped(java.awt.event.KeyEvent evt) {

txtNameKeyTyped(evt);

}

});

jLabel9.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jLabel9.setForeground(new java.awt.Color(255, 255, 255));

jLabel9.setText("Назва");

btnSave.setBackground(new java.awt.Color(143, 36, 72));

btnSave.setFont(new java.awt.Font("Times New Roman", 0, 28)); // NOI18N

btnSave.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/diskette.png"))); // NOI18N

btnSave.setText("Зберегти");

btnSave.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnSaveActionPerformed(evt);

}

});

jLabel17.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jLabel17.setForeground(new java.awt.Color(255, 255, 255));

jLabel17.setText("Призначення");

txtApp.setBackground(new java.awt.Color(143, 93, 100));

txtApp.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

txtApp.setForeground(new java.awt.Color(255, 255, 255));

txtApp.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyTyped(java.awt.event.KeyEvent evt) {

txtAppKeyTyped(evt);

}

});

jLabel18.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jLabel18.setForeground(new java.awt.Color(255, 255, 255));

jLabel18.setText("Тип");

jLabel19.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jLabel19.setForeground(new java.awt.Color(255, 255, 255));

jLabel19.setText("Стиль");

jLabel20.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jLabel20.setForeground(new java.awt.Color(255, 255, 255));

jLabel20.setText("Покриття");

jLabel21.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jLabel21.setForeground(new java.awt.Color(255, 255, 255));

jLabel21.setText("Розмір");

txtSize.setBackground(new java.awt.Color(143, 93, 100));

txtSize.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

txtSize.setForeground(new java.awt.Color(255, 255, 255));

txtSize.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyTyped(java.awt.event.KeyEvent evt) {

txtSizeKeyTyped(evt);

}

});

jLabel22.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jLabel22.setForeground(new java.awt.Color(255, 255, 255));

jLabel22.setText("Матеріал");

jcmMaterial.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jcmStyle.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jcmType.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jcmCoating.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

javax.swing.GroupLayout pnlMainLayout = new javax.swing.GroupLayout(pnlMain);

pnlMain.setLayout(pnlMainLayout);

pnlMainLayout.setHorizontalGroup(

pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(pnlHead, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGroup(pnlMainLayout.createSequentialGroup()

.addContainerGap()

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlMainLayout.createSequentialGroup()

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel17)

.addComponent(jLabel18))

.addGap(47, 47, 47)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(txtApp)

.addComponent(jcmType, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlMainLayout.createSequentialGroup()

.addGap(0, 0, Short.MAX\_VALUE)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlMainLayout.createSequentialGroup()

.addComponent(jLabel9)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(txtName, javax.swing.GroupLayout.PREFERRED\_SIZE, 350, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlMainLayout.createSequentialGroup()

.addComponent(btnSave, javax.swing.GroupLayout.PREFERRED\_SIZE, 268, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(50, 50, 50))))

.addGroup(pnlMainLayout.createSequentialGroup()

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel21)

.addComponent(jLabel20)

.addComponent(jLabel22)

.addComponent(jLabel19))

.addGap(85, 85, 85)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jcmStyle, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jcmMaterial, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(txtSize)

.addComponent(jcmCoating, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))))

.addContainerGap())

);

pnlMainLayout.setVerticalGroup(

pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlMainLayout.createSequentialGroup()

.addComponent(pnlHead, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(txtName, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel9, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGap(18, 18, 18)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel17, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(txtApp, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jLabel18, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jcmType, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(24, 24, 24)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel19, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jcmStyle, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(15, 15, 15)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel20, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jcmCoating, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(txtSize, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel21, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel22, javax.swing.GroupLayout.PREFERRED\_SIZE, 29, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jcmMaterial, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(btnSave)

.addContainerGap())

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(pnlMain, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(pnlMain, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

pack();

}// </editor-fold>

private void btnExitActionPerformed(java.awt.event.ActionEvent evt) {

variantStart = 0;

frmShow.setStatus(0);

dispose();

}

private void txtNameKeyPressed(java.awt.event.KeyEvent evt) {

}

private void txtNameKeyTyped(java.awt.event.KeyEvent evt) {

if (txtName.getText().length() >= 200) {// limit to 3 characters

evt.consume();

}

}

private void btnSaveActionPerformed(java.awt.event.ActionEvent evt) {

if (txtName.getText().length() > 0 && txtApp.getText().length() > 0 && txtSize.getText().length() > 0) {

DataB d = new DataB();

d.GetInd("SELECT name FROM Door WHERE `name` = '" + txtName.getText() + "';");

if (variantStart == 0) {

if (d.getAddName() == "Такого нема") {

d.InsertData("Door", txtName.getText(), txtApp.getText(), jcmType.getSelectedItem() + "",

jcmStyle.getSelectedItem() + "", jcmCoating.getSelectedItem() + "", txtSize.getText(), jcmMaterial.getSelectedItem().toString());

JOptionPane.showMessageDialog(rootPane, "Успішно");

frmShow.setStatus(0);

dispose();

frmShow.GetData("Door");

} else {

JOptionPane.showMessageDialog(rootPane, "Двері існують, повторіть спробу", "Помилка", 2);

}

} else if (variantStart == 1) {

if (d.getAddName() == "Такого нема" || txtName.getText().equals(frmShow.getNames()) == true) {

d.Update("Door", txtName.getText(), txtApp.getText(), jcmType.getSelectedItem() + "",

jcmStyle.getSelectedItem() + "", jcmCoating.getSelectedItem() + "", txtSize.getText(), jcmMaterial.getSelectedItem().toString());

JOptionPane.showMessageDialog(rootPane, "Успішно");

frmShow.setStatus(0);

frmShow.GetData("Door");

dispose();

} else {

JOptionPane.showMessageDialog(rootPane, "Двері існують, повторіть спробу", "Помилка", 2);

}

}

} else {

JOptionPane.showMessageDialog(rootPane, "Заповніть всі поля", "Помилка", 2);

}

frmShow.setNames("");

}

private void txtAppKeyTyped(java.awt.event.KeyEvent evt) {

// char c = evt.getKeyChar();

// if ((!Character.isDigit(c)

// || (c == KeyEvent.VK\_BACK\_SPACE)

// || (c == KeyEvent.VK\_DELETE)

// || (txtApp.getText().length() >= 7))) {

// evt.consume();

// }

}

private void txtSizeKeyTyped(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(AddP.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(AddP.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(AddP.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(AddP.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new AddP().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton btnExit;

private javax.swing.JButton btnSave;

private javax.swing.JLabel jLabel17;

private javax.swing.JLabel jLabel18;

private javax.swing.JLabel jLabel19;

private javax.swing.JLabel jLabel20;

private javax.swing.JLabel jLabel21;

private javax.swing.JLabel jLabel22;

private javax.swing.JLabel jLabel9;

private javax.swing.JComboBox<String> jcmCoating;

private javax.swing.JComboBox<String> jcmMaterial;

private javax.swing.JComboBox<String> jcmStyle;

private javax.swing.JComboBox<String> jcmType;

private javax.swing.JLabel lblTitle;

private javax.swing.JPanel pnlHead;

private javax.swing.JPanel pnlMain;

private javax.swing.JTextField txtApp;

private javax.swing.JTextField txtName;

private javax.swing.JTextField txtSize;

// End of variables declaration

}

Settings

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Main;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileWriter;

import java.io.IOException;

import java.util.ArrayList;

import java.util.Scanner;

import javax.swing.JOptionPane;

/\*\*

\*

\* @author ximik

\*/

public class Settings extends javax.swing.JFrame {

/\*\*

\* @param dbname the dbname to set

\*/

public void setDbname(String dbname) {

this.dbname = dbname;

}

/\*get + set

/\*\*

\* @return the server

\*/

public String getServer() {

return server;

}

/\*\*

\* @param server the server to set

\*/

public void setServer(String server) {

this.server = server;

}

/\*\*

\* @return the port

\*/

public String getPort() {

return port;

}

/\*\*

\* @param port the port to set

\*/

public void setPort(String port) {

this.port = port;

}

/\*\*

\* @return the user

\*/

public String getUser() {

return user;

}

/\*\*

\* @param user the user to set

\*/

public void setUser(String user) {

this.user = user;

}

/\*\*

\* @return the pass

\*/

public String getPass() {

return pass;

}

/\*\*

\* @param pass the pass to set

\*/

public void setPass(String pass) {

this.pass = pass;

}

/\*\*

\* @return the dbname

\*/

public String getDbname() {

return dbname;

}

////////////

private int delID = 0;

ArrayList<String> list = new ArrayList<>();

public static String[] array;

private String server ;

private String port ;

private String user ;

private String pass ;

private String dbname;

/\*\*

\* Creates new form Set

\*/

public Settings() {

initComponents();

txtServer.setHorizontalAlignment(txtServer.CENTER);

txtBD.setHorizontalAlignment(txtBD.CENTER);

txtPort.setHorizontalAlignment(txtPort.CENTER);

txtUser.setHorizontalAlignment(txtUser.CENTER);

txtPass.setHorizontalAlignment(txtPass.CENTER);

setLocationRelativeTo(null);

MoveMouseListener mml = new MoveMouseListener(pnlHead, this);

pnlHead.addMouseListener(mml);

pnlHead.addMouseMotionListener(mml);

btnSave.setEnabled(false);

server = "";

port = "";

user = "";

pass = "";

dbname = "";

this.setAlwaysOnTop(true);

try {

Read();

} catch (IOException ex) {

java.util.logging.Logger.getLogger(Settings.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

txtServer.setEnabled(false);

txtBD.setEnabled(false);

txtPort.setEnabled(false);

txtUser.setEnabled(false);

txtPass.setEnabled(false);

}

public void Read() throws IOException {

try (Scanner scan = new Scanner(new File("UserSettings.txt"))) {

while (scan.hasNextLine()) {

list.add(scan.nextLine());

}

} catch (FileNotFoundException e) {

System.out.println("Файл налаштувань пошкоджено");

}

try {

array = list.toArray(new String[0]);

txtServer.setText(array[0]);

txtPort.setText(array[1]);

txtUser.setText(array[2]);

txtPass.setText(array[3]);

txtBD.setText(array[4]);

setServer(array[0]);

setPort(array[1]);

setUser(array[2]);

setPass(array[3]);

setDbname(array[4]);

} catch (Exception ex) {

JOptionPane.showMessageDialog(rootPane, "Помилка зчитування");

FileWriter nFile = new FileWriter("UserSettings.txt");

nFile.write("127.0.0.1\n");

nFile.write("3306\n");

nFile.write("root\n");

nFile.write("\n");

nFile.write("Paterns");

nFile.close();

}

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

pnlContent = new javax.swing.JPanel();

pnlHead = new javax.swing.JPanel();

jLabel7 = new javax.swing.JLabel();

btnExit = new javax.swing.JButton();

btnSave = new javax.swing.JButton();

jButton1 = new javax.swing.JButton();

jLabel5 = new javax.swing.JLabel();

txtBD = new javax.swing.JTextField();

txtPass = new javax.swing.JTextField();

jLabel4 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

txtUser = new javax.swing.JTextField();

txtPort = new javax.swing.JTextField();

jLabel2 = new javax.swing.JLabel();

jLabel1 = new javax.swing.JLabel();

txtServer = new javax.swing.JTextField();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setUndecorated(true);

pnlContent.setBackground(new java.awt.Color(143, 93, 100));

pnlHead.setBackground(new java.awt.Color(143, 36, 72));

jLabel7.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

jLabel7.setForeground(new java.awt.Color(255, 255, 255));

jLabel7.setText("Налаштування");

btnExit.setBackground(new java.awt.Color(143, 36, 72));

btnExit.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/close.png"))); // NOI18N

btnExit.setBorder(null);

btnExit.setBorderPainted(false);

btnExit.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnExitActionPerformed(evt);

}

});

javax.swing.GroupLayout pnlHeadLayout = new javax.swing.GroupLayout(pnlHead);

pnlHead.setLayout(pnlHeadLayout);

pnlHeadLayout.setHorizontalGroup(

pnlHeadLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlHeadLayout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel7)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(btnExit, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE))

);

pnlHeadLayout.setVerticalGroup(

pnlHeadLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlHeadLayout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel7, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addContainerGap())

.addComponent(btnExit, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

btnSave.setBackground(new java.awt.Color(143, 36, 72));

btnSave.setFont(new java.awt.Font("Times New Roman", 1, 14)); // NOI18N

btnSave.setForeground(new java.awt.Color(255, 255, 255));

btnSave.setText("Зберегти");

btnSave.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnSaveActionPerformed(evt);

}

});

jButton1.setBackground(new java.awt.Color(143, 36, 72));

jButton1.setFont(new java.awt.Font("Times New Roman", 1, 14)); // NOI18N

jButton1.setForeground(new java.awt.Color(255, 255, 255));

jButton1.setText("Змінити дані");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jLabel5.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N

jLabel5.setForeground(new java.awt.Color(255, 255, 255));

jLabel5.setText("База данних");

txtBD.setBackground(new java.awt.Color(143, 93, 100));

txtBD.setFont(new java.awt.Font("Times New Roman", 0, 14)); // NOI18N

txtBD.setForeground(new java.awt.Color(255, 255, 255));

txtBD.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

txtBDActionPerformed(evt);

}

});

txtPass.setBackground(new java.awt.Color(143, 93, 100));

txtPass.setFont(new java.awt.Font("Times New Roman", 0, 14)); // NOI18N

txtPass.setForeground(new java.awt.Color(255, 255, 255));

jLabel4.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N

jLabel4.setForeground(new java.awt.Color(255, 255, 255));

jLabel4.setText("Пароль");

jLabel4.setName(""); // NOI18N

jLabel3.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N

jLabel3.setForeground(new java.awt.Color(255, 255, 255));

jLabel3.setText("Ім'я користувача");

txtUser.setBackground(new java.awt.Color(143, 93, 100));

txtUser.setFont(new java.awt.Font("Times New Roman", 0, 14)); // NOI18N

txtUser.setForeground(new java.awt.Color(255, 255, 255));

txtPort.setBackground(new java.awt.Color(143, 93, 100));

txtPort.setFont(new java.awt.Font("Times New Roman", 0, 14)); // NOI18N

txtPort.setForeground(new java.awt.Color(255, 255, 255));

txtPort.setToolTipText("");

txtPort.setCaretColor(new java.awt.Color(69, 73, 74));

jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N

jLabel2.setForeground(new java.awt.Color(255, 255, 255));

jLabel2.setText("Порт");

jLabel1.setFont(new java.awt.Font("Times New Roman", 1, 18)); // NOI18N

jLabel1.setForeground(new java.awt.Color(255, 255, 255));

jLabel1.setText("Сервер");

txtServer.setBackground(new java.awt.Color(143, 93, 100));

txtServer.setFont(new java.awt.Font("Times New Roman", 0, 14)); // NOI18N

txtServer.setForeground(new java.awt.Color(255, 255, 255));

txtServer.setToolTipText("");

txtServer.setCaretColor(new java.awt.Color(69, 73, 74));

javax.swing.GroupLayout pnlContentLayout = new javax.swing.GroupLayout(pnlContent);

pnlContent.setLayout(pnlContentLayout);

pnlContentLayout.setHorizontalGroup(

pnlContentLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(pnlHead, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGroup(pnlContentLayout.createSequentialGroup()

.addContainerGap()

.addGroup(pnlContentLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlContentLayout.createSequentialGroup()

.addGroup(pnlContentLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(txtServer, javax.swing.GroupLayout.PREFERRED\_SIZE, 304, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGroup(pnlContentLayout.createSequentialGroup()

.addGap(130, 130, 130)

.addComponent(jLabel1))

.addGroup(pnlContentLayout.createSequentialGroup()

.addGap(138, 138, 138)

.addComponent(jLabel2))

.addGroup(pnlContentLayout.createSequentialGroup()

.addGap(130, 130, 130)

.addComponent(jLabel4))

.addGroup(pnlContentLayout.createSequentialGroup()

.addComponent(jButton1)

.addGap(56, 56, 56)

.addComponent(btnSave, javax.swing.GroupLayout.PREFERRED\_SIZE, 135, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlContentLayout.createSequentialGroup()

.addGap(0, 0, Short.MAX\_VALUE)

.addGroup(pnlContentLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlContentLayout.createSequentialGroup()

.addComponent(txtPort, javax.swing.GroupLayout.PREFERRED\_SIZE, 304, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlContentLayout.createSequentialGroup()

.addComponent(jLabel3)

.addGap(73, 73, 73))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlContentLayout.createSequentialGroup()

.addGroup(pnlContentLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(txtUser, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED\_SIZE, 304, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(txtPass, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED\_SIZE, 303, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addContainerGap())

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlContentLayout.createSequentialGroup()

.addComponent(jLabel5)

.addGap(99, 99, 99))))))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlContentLayout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(txtBD, javax.swing.GroupLayout.PREFERRED\_SIZE, 303, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

pnlContentLayout.setVerticalGroup(

pnlContentLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlContentLayout.createSequentialGroup()

.addComponent(pnlHead, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(txtServer, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(txtPort, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(txtUser, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel4)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(txtPass, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel5)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(txtBD, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addGroup(pnlContentLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jButton1)

.addComponent(btnSave))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(pnlContent, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(pnlContent, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

pack();

}// </editor-fold>

private void btnSaveActionPerformed(java.awt.event.ActionEvent evt) {

try {

if (txtServer.getText().trim().length() <= 0 || txtPort.getText().trim().length() <= 0

|| txtUser.getText().trim().length() <= 0 || txtBD.getText().trim().length() <= 0) {

JOptionPane.showMessageDialog(rootPane, "Помилка");

} else {

FileWriter nFile = new FileWriter("UserSettings.txt");

nFile.write(txtServer.getText() + "\n");

nFile.write(txtPort.getText() + "\n");

nFile.write(txtUser.getText() + "\n");

nFile.write(txtPass.getText() + "\n");

nFile.write(txtBD.getText() + "\n");

nFile.close();

btnSave.setEnabled(false);

txtServer.setEnabled(false);

txtBD.setEnabled(false);

txtPort.setEnabled(false);

txtUser.setEnabled(false);

txtPass.setEnabled(false);

}

Main.PlugIn();

} catch (IOException ex) {

// Logger.getLogger(Open.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

btnSave.setEnabled(true);

txtServer.setEnabled(true);

txtBD.setEnabled(true);

txtPort.setEnabled(true);

txtUser.setEnabled(true);

txtPass.setEnabled(true);

}

private void txtBDActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void btnExitActionPerformed(java.awt.event.ActionEvent evt) {

Main.setStatus(0);

dispose();

Main.setStatus(0);

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Settings.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Settings.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Settings.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Settings.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new Settings().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton btnExit;

private javax.swing.JButton btnSave;

private javax.swing.JButton jButton1;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel7;

private javax.swing.JPanel pnlContent;

private javax.swing.JPanel pnlHead;

private javax.swing.JTextField txtBD;

private javax.swing.JTextField txtPass;

public javax.swing.JTextField txtPort;

public javax.swing.JTextField txtServer;

private javax.swing.JTextField txtUser;

// End of variables declaration

}

frmShow

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package Main;

import java.awt.Color;

import java.awt.Component;

import java.awt.Font;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.ResultSetMetaData;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

import javax.swing.JTable;

import javax.swing.JTextArea;

import javax.swing.table.TableCellRenderer;

import javax.swing.table.TableColumnModel;

/\*\*

\*

\* @author ximik

\*/

public class frmShow extends javax.swing.JFrame {

/\*\*

\* @return the name

\*/

public static String getNames() {

return names;

}

/\*\*

\* @param aName the name to set

\*/

public static void setNames(String aName) {

names = aName;

}

/\*\*

\* @return the type

\*/

public static String getTypes() {

return types;

}

/\*\*

\* @param aType the type to set

\*/

public static void setType(String aType) {

types = aType;

}

/\*\*

\* @return the minInfo

\*/

public static String getMinInfo() {

return minInfo;

}

/\*\*

\* @param aMinInfo the minInfo to set

\*/

public static void setMinInfo(String aMinInfo) {

minInfo = aMinInfo;

}

/\*\*

\* @return the status

\*/

public static int getStatus() {

return status;

}

/\*\*

\* @param status the status to set

\*/

public static void setStatus(int statu) {

status = statu;

}

private Connection conn = null;

private DataB a = new DataB();

static ResultSet X = null;

static int status = 0;

private static String names = "";

private static String types = "";

private static String nameT = "";

private static String minInfo = "";

Settings s = new Settings();

String[] YesNo = new String[]{"Так", "Ні"};

/\*\*

\* Creates new form frmView

\*/

public frmShow() {

initComponents();

MoveMouseListener m = new MoveMouseListener(pnlHead, this);

pnlHead.addMouseListener(m);

pnlHead.addMouseMotionListener(m);

this.setLocationRelativeTo(null);

tbl.setDefaultEditor(Object.class, null);

editColor(1);

GetData("Door");

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

pnlMain = new javax.swing.JPanel();

pnlHead = new javax.swing.JPanel();

btnExit = new javax.swing.JButton();

jLabel1 = new javax.swing.JLabel();

pnlButtton = new javax.swing.JPanel();

btnEdit = new javax.swing.JButton();

btnDel = new javax.swing.JButton();

btnAdd = new javax.swing.JButton();

jScrollPane1 = new javax.swing.JScrollPane();

tbl = new javax.swing.JTable();

txtSearch = new javax.swing.JTextField();

lblHead1 = new javax.swing.JLabel();

pnlTable = new javax.swing.JPanel();

btnDoor = new javax.swing.JButton();

btnType = new javax.swing.JButton();

btnMatherial = new javax.swing.JButton();

btnStyle = new javax.swing.JButton();

btnCoating = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setUndecorated(true);

addWindowListener(new java.awt.event.WindowAdapter() {

public void windowOpened(java.awt.event.WindowEvent evt) {

formWindowOpened(evt);

}

});

pnlMain.setBackground(new java.awt.Color(143, 93, 100));

pnlHead.setBackground(new java.awt.Color(143, 36, 72));

pnlHead.setForeground(new java.awt.Color(221, 47, 85));

btnExit.setBackground(new java.awt.Color(143, 36, 72));

btnExit.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/close.png"))); // NOI18N

btnExit.setBorder(null);

btnExit.setBorderPainted(false);

btnExit.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnExitActionPerformed(evt);

}

});

jLabel1.setFont(new java.awt.Font("Times New Roman", 0, 30)); // NOI18N

jLabel1.setForeground(new java.awt.Color(255, 255, 255));

jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

jLabel1.setText("Галерея дверей");

javax.swing.GroupLayout pnlHeadLayout = new javax.swing.GroupLayout(pnlHead);

pnlHead.setLayout(pnlHeadLayout);

pnlHeadLayout.setHorizontalGroup(

pnlHeadLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlHeadLayout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 1062, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 28, Short.MAX\_VALUE)

.addComponent(btnExit, javax.swing.GroupLayout.PREFERRED\_SIZE, 45, javax.swing.GroupLayout.PREFERRED\_SIZE))

);

pnlHeadLayout.setVerticalGroup(

pnlHeadLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(pnlHeadLayout.createSequentialGroup()

.addGap(0, 0, Short.MAX\_VALUE)

.addGroup(pnlHeadLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(btnExit, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, 44, Short.MAX\_VALUE)))

);

pnlButtton.setBackground(new java.awt.Color(143, 93, 100));

pnlButtton.setLayout(new java.awt.GridLayout(3, 1, 0, 30));

btnEdit.setBackground(new java.awt.Color(143, 36, 72));

btnEdit.setFont(new java.awt.Font("Times New Roman", 0, 28)); // NOI18N

btnEdit.setForeground(new java.awt.Color(255, 255, 255));

btnEdit.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/pencil-2.png"))); // NOI18N

btnEdit.setText("Редагувати");

btnEdit.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnEditActionPerformed(evt);

}

});

pnlButtton.add(btnEdit);

btnDel.setBackground(new java.awt.Color(143, 36, 72));

btnDel.setFont(new java.awt.Font("Times New Roman", 0, 28)); // NOI18N

btnDel.setForeground(new java.awt.Color(255, 255, 255));

btnDel.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/delete.png"))); // NOI18N

btnDel.setText("Видалити");

btnDel.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnDelActionPerformed(evt);

}

});

pnlButtton.add(btnDel);

btnAdd.setBackground(new java.awt.Color(143, 36, 72));

btnAdd.setFont(new java.awt.Font("Times New Roman", 0, 28)); // NOI18N

btnAdd.setForeground(new java.awt.Color(255, 255, 255));

btnAdd.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/add.png"))); // NOI18N

btnAdd.setText("Додати");

btnAdd.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnAddActionPerformed(evt);

}

});

pnlButtton.add(btnAdd);

tbl.setBackground(new java.awt.Color(143, 93, 100));

tbl.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N

tbl.setForeground(new java.awt.Color(0, 0, 0));

tbl.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

}

));

tbl.setToolTipText("Клацніть два рази для \nперегляду детальної інформації");

tbl.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

tbl.setGridColor(new java.awt.Color(102, 204, 255));

tbl.setRowHeight(40);

tbl.setSelectionBackground(new java.awt.Color(143, 36, 72));

tbl.setSelectionForeground(new java.awt.Color(255, 255, 255));

tbl.addMouseListener(new java.awt.event.MouseAdapter() {

public void mouseClicked(java.awt.event.MouseEvent evt) {

tblMouseClicked(evt);

}

public void mouseEntered(java.awt.event.MouseEvent evt) {

tblMouseEntered(evt);

}

public void mousePressed(java.awt.event.MouseEvent evt) {

tblMousePressed(evt);

}

});

jScrollPane1.setViewportView(tbl);

txtSearch.setFont(new java.awt.Font("Times New Roman", 0, 18)); // NOI18N

txtSearch.setForeground(new java.awt.Color(0, 0, 0));

txtSearch.setToolTipText("Пошук відбувається по всім полям\nВведіть запит та натисніть Enter");

txtSearch.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyPressed(java.awt.event.KeyEvent evt) {

txtSearchKeyPressed(evt);

}

public void keyTyped(java.awt.event.KeyEvent evt) {

txtSearchKeyTyped(evt);

}

});

lblHead1.setFont(new java.awt.Font("Times New Roman", 0, 24)); // NOI18N

lblHead1.setForeground(new java.awt.Color(255, 255, 255));

lblHead1.setText("Пошук");

pnlTable.setBackground(new java.awt.Color(143, 93, 100));

pnlTable.setLayout(new java.awt.GridLayout(1, 5, 10, 10));

btnDoor.setBackground(new java.awt.Color(143, 36, 72));

btnDoor.setFont(new java.awt.Font("Times New Roman", 0, 28)); // NOI18N

btnDoor.setForeground(new java.awt.Color(255, 255, 255));

btnDoor.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/door.png"))); // NOI18N

btnDoor.setText("Двері");

btnDoor.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnDoorActionPerformed(evt);

}

});

pnlTable.add(btnDoor);

btnType.setBackground(new java.awt.Color(143, 36, 72));

btnType.setFont(new java.awt.Font("Times New Roman", 0, 28)); // NOI18N

btnType.setForeground(new java.awt.Color(255, 255, 255));

btnType.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/geometric.png"))); // NOI18N

btnType.setText("Типи");

btnType.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnTypeActionPerformed(evt);

}

});

pnlTable.add(btnType);

btnMatherial.setBackground(new java.awt.Color(143, 36, 72));

btnMatherial.setFont(new java.awt.Font("Times New Roman", 0, 28)); // NOI18N

btnMatherial.setForeground(new java.awt.Color(255, 255, 255));

btnMatherial.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/membrane.png"))); // NOI18N

btnMatherial.setText("Матеріал");

btnMatherial.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnMatherialActionPerformed(evt);

}

});

pnlTable.add(btnMatherial);

btnStyle.setBackground(new java.awt.Color(143, 36, 72));

btnStyle.setFont(new java.awt.Font("Times New Roman", 0, 28)); // NOI18N

btnStyle.setForeground(new java.awt.Color(255, 255, 255));

btnStyle.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/books.png"))); // NOI18N

btnStyle.setText("Стилі");

btnStyle.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnStyleActionPerformed(evt);

}

});

pnlTable.add(btnStyle);

btnCoating.setBackground(new java.awt.Color(143, 36, 72));

btnCoating.setFont(new java.awt.Font("Times New Roman", 0, 28)); // NOI18N

btnCoating.setForeground(new java.awt.Color(255, 255, 255));

btnCoating.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Photo/rack.png"))); // NOI18N

btnCoating.setText("Покриття");

btnCoating.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

btnCoatingActionPerformed(evt);

}

});

pnlTable.add(btnCoating);

javax.swing.GroupLayout pnlMainLayout = new javax.swing.GroupLayout(pnlMain);

pnlMain.setLayout(pnlMainLayout);

pnlMainLayout.setHorizontalGroup(

pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(pnlHead, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGroup(pnlMainLayout.createSequentialGroup()

.addContainerGap()

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(pnlButtton, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(lblHead1)

.addComponent(txtSearch))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlMainLayout.createSequentialGroup()

.addGap(0, 0, Short.MAX\_VALUE)

.addComponent(pnlTable, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, pnlMainLayout.createSequentialGroup()

.addContainerGap(207, Short.MAX\_VALUE)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 928, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap()))

);

pnlMainLayout.setVerticalGroup(

pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlMainLayout.createSequentialGroup()

.addComponent(pnlHead, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(75, 75, 75)

.addComponent(lblHead1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(txtSearch, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(51, 51, 51)

.addComponent(pnlButtton, javax.swing.GroupLayout.PREFERRED\_SIZE, 262, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(59, 59, 59)

.addComponent(pnlTable, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(32, Short.MAX\_VALUE))

.addGroup(pnlMainLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlMainLayout.createSequentialGroup()

.addGap(53, 53, 53)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 496, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(81, Short.MAX\_VALUE)))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(pnlMain, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(pnlMain, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

);

pack();

}// </editor-fold>

private void formWindowOpened(java.awt.event.WindowEvent evt) {

try {

PlugIn();

} catch (Exception ex) {

JOptionPane.showMessageDialog(this, "Помилка підключення");

//btnUpdate.setVisible(true);

}

}

private void txtSearchKeyTyped(java.awt.event.KeyEvent evt) {

try {

Search();

} catch (Exception ex) {

}

}

private void txtSearchKeyPressed(java.awt.event.KeyEvent evt) {

Search();

}

private void tblMousePressed(java.awt.event.MouseEvent evt) {

setNames(tbl.getModel().getValueAt(tbl.getSelectedRow(), 0).toString());

}

private void tblMouseEntered(java.awt.event.MouseEvent evt) {

}

private void tblMouseClicked(java.awt.event.MouseEvent evt) {

setNames(tbl.getModel().getValueAt(tbl.getSelectedRow(), 0).toString());

}

private void btnDelActionPerformed(java.awt.event.ActionEvent evt) {

if (getNames() != "") {

if (nameT == "Door") {

int rez = JOptionPane.showOptionDialog(null,

"Видалити двері: " + getNames() + "?",

"Видалення",

JOptionPane.YES\_NO\_OPTION,

JOptionPane.INFORMATION\_MESSAGE,

null,

YesNo,

"default");

if (rez == 0) {

a.del("Door", "name", getNames());

JOptionPane.showMessageDialog(this, "Успішно", "Видалення", 1);

GetData("Door");

setNames("");

}

} else {

int rez = JOptionPane.showOptionDialog(null,

"Видалити об'єкт з іменем: " + getNames() + "?",

"Видалення",

JOptionPane.YES\_NO\_OPTION,

JOptionPane.INFORMATION\_MESSAGE,

null,

YesNo,

"default");

if (rez == 0) {

a.del(nameT, "name", getNames());

JOptionPane.showMessageDialog(this, "Успішно", "Видалення", 1);

GetData(nameT);

setNames("");

}

}

}

}

private void btnEditActionPerformed(java.awt.event.ActionEvent evt) {

if (status == 0) {

if (getNames() != "") {

if (nameT == "Door") {

AddP.setVariantStart(1);

AddP add = new AddP();

add.setVisible(true);

status = 1;

} else {

Edit();

}

}

}

}

void Edit() {

String z = "";

if (nameT == "Type") {

z = JOptionPane.showInputDialog(rootPane, "Введіть нову назву для типу: " + getNames(), "Додавання",

JOptionPane.QUESTION\_MESSAGE);

} else if (nameT == "Style") {

z = JOptionPane.showInputDialog(rootPane, "Введіть нову назву для стилю: " + getNames(), "Додавання",

JOptionPane.QUESTION\_MESSAGE);

} else if (nameT == "Coating") {

z = JOptionPane.showInputDialog(rootPane, "Введіть нову назву для покриття: " + getNames(), "Додавання",

JOptionPane.QUESTION\_MESSAGE);

} else if (nameT == "Material") {

z = JOptionPane.showInputDialog(rootPane, "Введіть нову назву для метеріалу: " + getNames(), "Додавання",

JOptionPane.QUESTION\_MESSAGE);

}

try {

if (z.trim().length() > 0 && z.trim().length() < 35) {

a.GetRep("name", z, nameT);

if (a.getAddName() == "Такого нема" || z.equals(getNames())) {

try {

a.UpdateFilter(nameT, "name", z);

a.setAddName("Нема");

GetData(nameT);

JOptionPane.showMessageDialog(rootPane, "Успішно", "Додавання", 1);

setNames("");

a.setAddName("Такого нема");

} catch (Exception ex) {

}

} else {

a.setAddName("Такого нема");

JOptionPane.showMessageDialog(rootPane, "Таке існує існує", "Помилка", 2);

Edit();

}

} else if (z.trim().length() > 35) {

JOptionPane.showMessageDialog(rootPane, "Введено більше ніж 30 символів, повторіть спробу", "Помилка", 2);

Edit();

}

} catch (Exception ex) {

}

}

private void btnAddActionPerformed(java.awt.event.ActionEvent evt) {

if (status == 0) {

if (nameT == "Door") {

AddP.setVariantStart(0);

AddP add = new AddP();

add.setVisible(true);

status = 1;

} else {

add();

}

}

}

void add() {

String z = "";

if (nameT == "Type") {

z = JOptionPane.showInputDialog(rootPane, "Введіть назву типу: " + getNames(), "Додавання",

JOptionPane.QUESTION\_MESSAGE);

} else if (nameT == "Style") {

z = JOptionPane.showInputDialog(rootPane, "Введіть назву стилю: " + getNames(), "Додавання",

JOptionPane.QUESTION\_MESSAGE);

} else if (nameT == "Coating") {

z = JOptionPane.showInputDialog(rootPane, "Введіть назву покриття: " + getNames(), "Додавання",

JOptionPane.QUESTION\_MESSAGE);

} else if (nameT == "Material") {

z = JOptionPane.showInputDialog(rootPane, "Введіть назву метеріалу: " + getNames(), "Додавання",

JOptionPane.QUESTION\_MESSAGE);

}

try {

if (z.trim().length() > 0 && z.trim().length() < 35) {

a.GetRep("name", z, nameT);

if (a.getAddName() == "Такого нема" || z.equals(getNames())) {

try {

a.InsertDataFilter(nameT, "name", z);

a.setAddName("Нема");

GetData(nameT);

JOptionPane.showMessageDialog(rootPane, "Успішно", "Додавання", 1);

setNames("");

a.setAddName("Такого нема");

} catch (Exception ex) {

}

} else {

a.setAddName("Такого нема");

JOptionPane.showMessageDialog(rootPane, "Таке існує існує", "Помилка", 2);

add();

}

} else if (z.trim().length() > 30) {

JOptionPane.showMessageDialog(rootPane, "Введено більше ніж 30 символів, повторіть спробу", "Помилка", 2);

add();

}

} catch (Exception ex) {

}

}

private void btnExitActionPerformed(java.awt.event.ActionEvent evt) {

System.exit(0);

}

private void btnDoorActionPerformed(java.awt.event.ActionEvent evt) {

editColor(1);

GetData("Door");

}

private void btnTypeActionPerformed(java.awt.event.ActionEvent evt) {

editColor(2);

GetData("Type");

}

private void btnStyleActionPerformed(java.awt.event.ActionEvent evt) {

editColor(4);

GetData("Style");

}

private void btnMatherialActionPerformed(java.awt.event.ActionEvent evt) {

editColor(3);

GetData("Material");

}

private void btnCoatingActionPerformed(java.awt.event.ActionEvent evt) {

editColor(5);

GetData("Coating");

}

void editColor(int choose) {

btnDoor.setBackground(new Color(143, 36, 72));

btnMatherial.setBackground(new Color(143, 36, 72));

btnStyle.setBackground(new Color(143, 36, 72));

btnCoating.setBackground(new Color(143, 36, 72));

btnType.setBackground(new Color(143, 36, 72));

setNames("");

txtSearch.setText("");

if (choose == 1) {

btnDoor.setBackground(new Color(255, 102, 102));

} else if (choose == 2) {

btnType.setBackground(new Color(255, 102, 102));

} else if (choose == 3) {

btnMatherial.setBackground(new Color(255, 102, 102));

} else if (choose == 4) {

btnStyle.setBackground(new Color(255, 102, 102));

} else if (choose == 5) {

btnCoating.setBackground(new Color(255, 102, 102));

}

}

/\*\*

\* Пошук

\*/

void Search() {

if (nameT == "Door") {

DataB db = new DataB();

X = null;

X = db.Search("Door", txtSearch.getText());

int i = 0, ii = 1;

try {

ResultSetMetaData colonki = X.getMetaData();

i = colonki.getColumnCount();

javax.swing.table.DefaultTableModel dtm = new javax.swing.table.DefaultTableModel();

dtm.setColumnCount(i);//

String nazvaKolonki[] = new String[i];

nazvaKolonki[0] = "Назва";

nazvaKolonki[1] = "Призначення";

nazvaKolonki[2] = "Тип";

nazvaKolonki[3] = "Стиль";

nazvaKolonki[4] = "Покриття";

nazvaKolonki[5] = "Розмір";

nazvaKolonki[6] = "Матеріал";

dtm.setColumnIdentifiers(nazvaKolonki);

dtm.setRowCount(0);//

while (X.next()) {

dtm.setRowCount(dtm.getRowCount() + 1);

for (int j = 1; j <= i; j++) {

String g = X.getString(j);

dtm.setValueAt(g, ii - 1, j - 1);//виводимо дані у модель

}

ii++;

}

tbl.setModel(dtm);

final TableColumnModel columnModel = tbl.getColumnModel();

for (int column = 0; column < tbl.getColumnCount(); column++) {

int width = 15; // Min width

for (int row = 0; row < tbl.getRowCount(); row++) {

TableCellRenderer renderer = tbl.getCellRenderer(row, column);

Component comp = tbl.prepareRenderer(renderer, row, column);

width = Math.max(comp.getPreferredSize().width + 1, width);

}

if (width > 300) {

width = 300;

}

columnModel.getColumn(column).setPreferredWidth(width);

}

tbl.getColumn("Назва").setCellRenderer(new myCellRenderer());

int columnNumber = tbl.getColumnModel().getColumnCount();

for (int c = 0; c < columnNumber; c++) {

tbl.getColumnModel().getColumn(c).setResizable(false);

}

} catch (Exception ex) {

Logger.getLogger(frmShow.class.getName()).log(Level.SEVERE, null, ex);

}

db.finalize();

} else {

DataB db = new DataB();

X = null;

X = db.Search(nameT, txtSearch.getText());

int i = 0, ii = 1;

try {

ResultSetMetaData colonki = X.getMetaData();

i = colonki.getColumnCount();

javax.swing.table.DefaultTableModel dtm = new javax.swing.table.DefaultTableModel();

dtm.setColumnCount(i);//

String nazvaKolonki[] = new String[i];

nazvaKolonki[0] = "Назва";

dtm.setColumnIdentifiers(nazvaKolonki);

dtm.setRowCount(0);//

while (X.next()) {

dtm.setRowCount(dtm.getRowCount() + 1);

for (int j = 1; j <= i; j++) {

String g = X.getString(j);

dtm.setValueAt(g, ii - 1, j - 1);//виводимо дані у модель

}

ii++;

}

tbl.setModel(dtm);

final TableColumnModel columnModel = tbl.getColumnModel();

for (int column = 0; column < tbl.getColumnCount(); column++) {

int width = 15; // Min width

for (int row = 0; row < tbl.getRowCount(); row++) {

TableCellRenderer renderer = tbl.getCellRenderer(row, column);

Component comp = tbl.prepareRenderer(renderer, row, column);

width = Math.max(comp.getPreferredSize().width + 1, width);

}

if (width > 300) {

width = 300;

}

columnModel.getColumn(column).setPreferredWidth(width);

}

tbl.getColumn("Назва").setCellRenderer(new myCellRenderer());

int columnNumber = tbl.getColumnModel().getColumnCount();

for (int c = 0; c < columnNumber; c++) {

tbl.getColumnModel().getColumn(c).setResizable(false);

}

} catch (Exception ex) {

Logger.getLogger(frmShow.class.getName()).log(Level.SEVERE, null, ex);

}

db.finalize();

}

}

void PlugIn() {

if (this.conn == null) {

a.finalize();

a = null;

a = new DataB();

conn = a.Connect();

if (conn == null) {

// btnUpdate.setVisible(true);

} else {

// btnUpdate.setVisible(false);

GetData("Door");

// JOptionPane.showMessageDialog(this, "Заєбісь");

}

}

}

/\*\*

\* Отримання даних

\*

\* @param nameT ім'я таблиці

\*/

public static void GetData(String nameT) {

frmShow.nameT = nameT;

DataB db = new DataB();

if (nameT == "Door") {

X = null;

X = db.GetDataSet(nameT);

int i = 0, ii = 1;

try {

ResultSetMetaData colonki = X.getMetaData();

i = colonki.getColumnCount();

javax.swing.table.DefaultTableModel dtm = new javax.swing.table.DefaultTableModel();

dtm.setColumnCount(i);//

String nazvaKolonki[] = new String[i];

nazvaKolonki[0] = "Назва";

nazvaKolonki[1] = "Призначення";

nazvaKolonki[2] = "Тип";

nazvaKolonki[3] = "Стиль";

nazvaKolonki[4] = "Покриття";

nazvaKolonki[5] = "Розмір";

nazvaKolonki[6] = "Матеріал";

dtm.setColumnIdentifiers(nazvaKolonki);

dtm.setRowCount(0);//

while (X.next()) {

dtm.setRowCount(dtm.getRowCount() + 1);

for (int j = 1; j <= i; j++) {

String g = X.getString(j);

dtm.setValueAt(g, ii - 1, j - 1);//виводимо дані у модель

}

ii++;

}

tbl.setModel(dtm);

final TableColumnModel columnModel = tbl.getColumnModel();

for (int column = 0; column < tbl.getColumnCount(); column++) {

int width = 15; // Min width

for (int row = 0; row < tbl.getRowCount(); row++) {

TableCellRenderer renderer = tbl.getCellRenderer(row, column);

Component comp = tbl.prepareRenderer(renderer, row, column);

width = Math.max(comp.getPreferredSize().width + 1, width);

}

if (width > 300) {

width = 300;

}

columnModel.getColumn(column).setPreferredWidth(width);

}

tbl.getColumn("Назва").setCellRenderer(new myCellRenderer());

tbl.getColumn("Призначення").setCellRenderer(new myCellRenderer());

tbl.getColumn("Тип").setCellRenderer(new myCellRenderer());

tbl.getColumn("Стиль").setCellRenderer(new myCellRenderer());

tbl.getColumn("Покриття").setCellRenderer(new myCellRenderer());

tbl.getColumn("Розмір").setCellRenderer(new myCellRenderer());

tbl.getColumn("Матеріал").setCellRenderer(new myCellRenderer());

int columnNumber = tbl.getColumnModel().getColumnCount();

for (int c = 0; c < columnNumber; c++) {

tbl.getColumnModel().getColumn(c).setResizable(false);

}

} catch (Exception ex) {

Logger.getLogger(frmShow.class.getName()).log(Level.SEVERE, null, ex);

}

} else {

X = null;

X = db.GetDataSet(nameT);

int i = 0, ii = 1;

try {

ResultSetMetaData colonki = X.getMetaData();

i = colonki.getColumnCount();

javax.swing.table.DefaultTableModel dtm = new javax.swing.table.DefaultTableModel();

dtm.setColumnCount(i);//

String nazvaKolonki[] = new String[i];

nazvaKolonki[0] = "Назва";

dtm.setColumnIdentifiers(nazvaKolonki);

dtm.setRowCount(0);//

while (X.next()) {

dtm.setRowCount(dtm.getRowCount() + 1);

for (int j = 1; j <= i; j++) {

String g = X.getString(j);

dtm.setValueAt(g, ii - 1, j - 1);//виводимо дані у модель

}

ii++;

}

tbl.setModel(dtm);

final TableColumnModel columnModel = tbl.getColumnModel();

for (int column = 0; column < tbl.getColumnCount(); column++) {

int width = 15; // Min width

for (int row = 0; row < tbl.getRowCount(); row++) {

TableCellRenderer renderer = tbl.getCellRenderer(row, column);

Component comp = tbl.prepareRenderer(renderer, row, column);

width = Math.max(comp.getPreferredSize().width + 1, width);

}

if (width > 300) {

width = 300;

}

columnModel.getColumn(column).setPreferredWidth(width);

}

tbl.getColumn("Назва").setCellRenderer(new myCellRenderer());

int columnNumber = tbl.getColumnModel().getColumnCount();

for (int c = 0; c < columnNumber; c++) {

tbl.getColumnModel().getColumn(c).setResizable(false);

}

} catch (Exception ex) {

Logger.getLogger(frmShow.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

private static class myCellRenderer extends JTextArea implements TableCellRenderer {

myCellRenderer() {

setLineWrap(true);

setWrapStyleWord(true);

}

public Component getTableCellRendererComponent(JTable table, Object value, boolean isSelected, boolean hasFocus, int row, int column) {

setText(value.toString());

setSize(table.getColumnModel().getColumn(column).getWidth(), getPreferredSize().height);

if (table.getRowHeight(row) != getPreferredSize().height) {

table.setRowHeight(row, getPreferredSize().height);

Font font = new Font("Times New Roman", Font.PLAIN, 18);

this.setFont(font);

}

if (!isSelected == true) {

this.setBackground(new Color(143, 93, 100));

this.setForeground(Color.BLACK);

} else {

this.setBackground(new Color(255, 102, 102));

this.setForeground(Color.white);

}

return this;

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(frmShow.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(frmShow.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(frmShow.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(frmShow.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new frmShow().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton btnAdd;

private javax.swing.JButton btnCoating;

private javax.swing.JButton btnDel;

private javax.swing.JButton btnDoor;

private javax.swing.JButton btnEdit;

private javax.swing.JButton btnExit;

private javax.swing.JButton btnMatherial;

private javax.swing.JButton btnStyle;

private javax.swing.JButton btnType;

private javax.swing.JLabel jLabel1;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JLabel lblHead1;

private javax.swing.JPanel pnlButtton;

private javax.swing.JPanel pnlHead;

private javax.swing.JPanel pnlMain;

private javax.swing.JPanel pnlTable;

public static javax.swing.JTable tbl;

private javax.swing.JTextField txtSearch;

// End of variables declaration

}