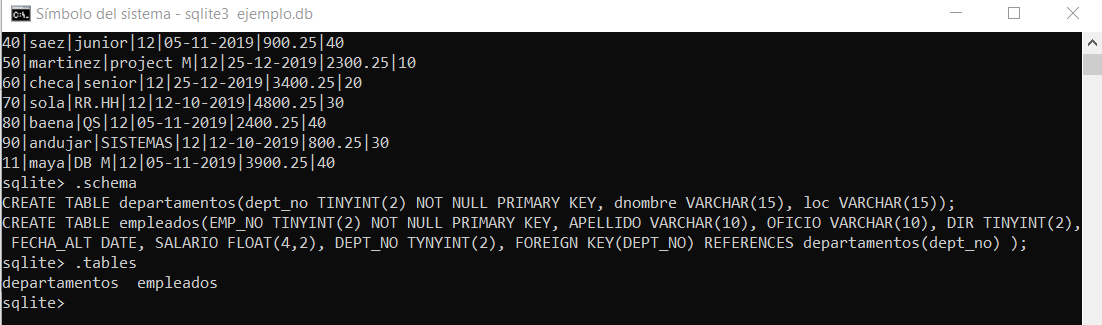
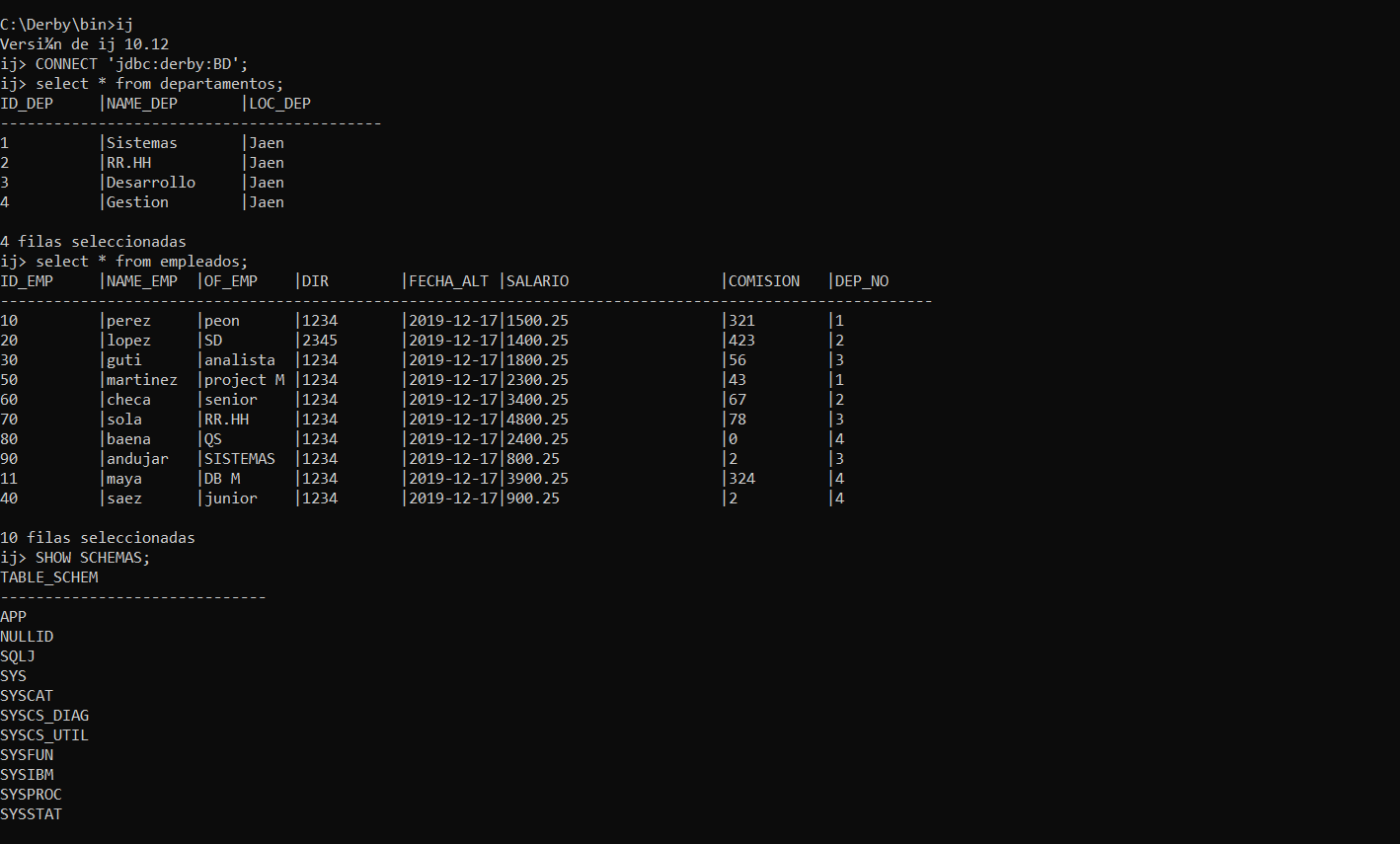
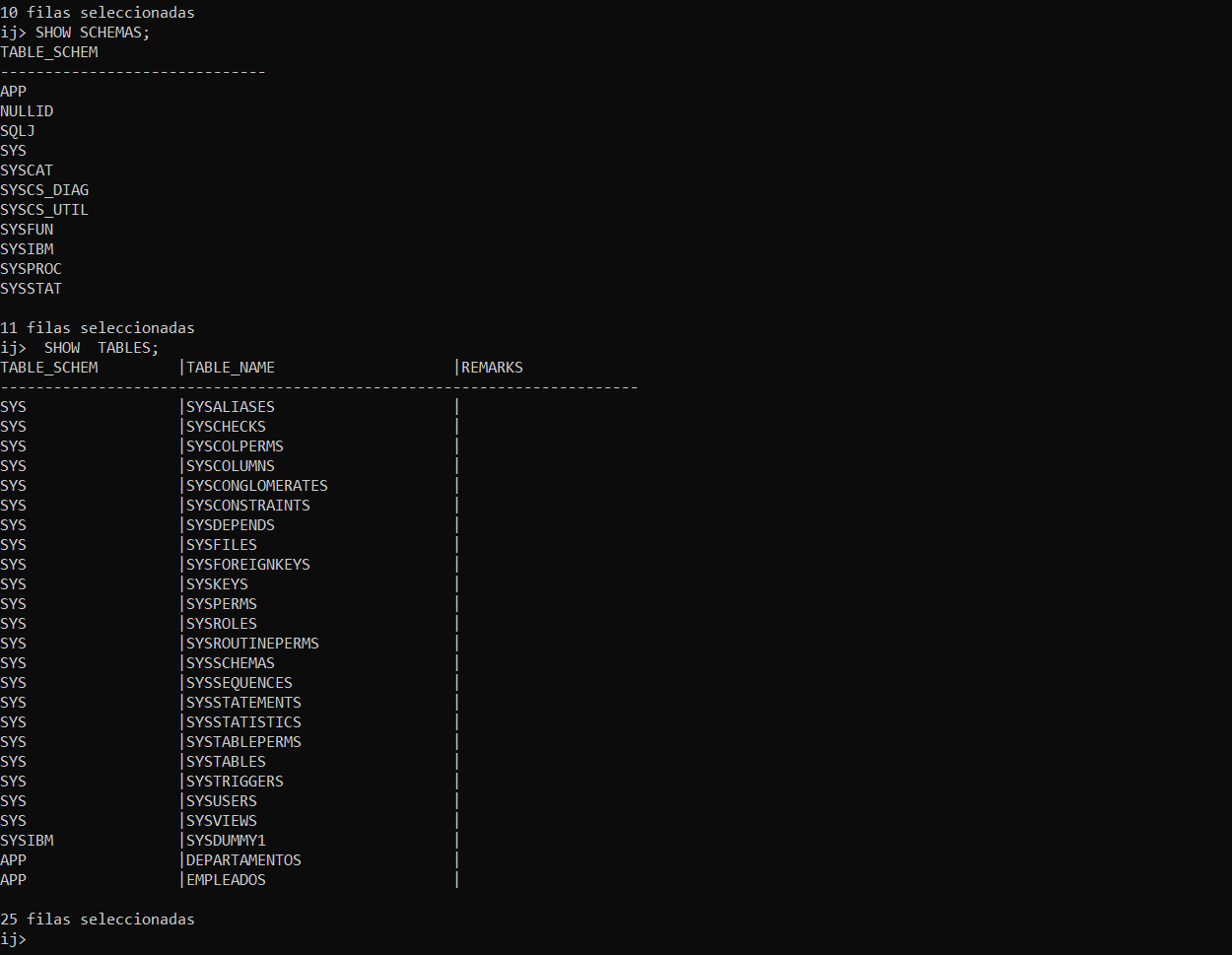
**EJERCICIOS TEMA 2**

Ejercicio 1:

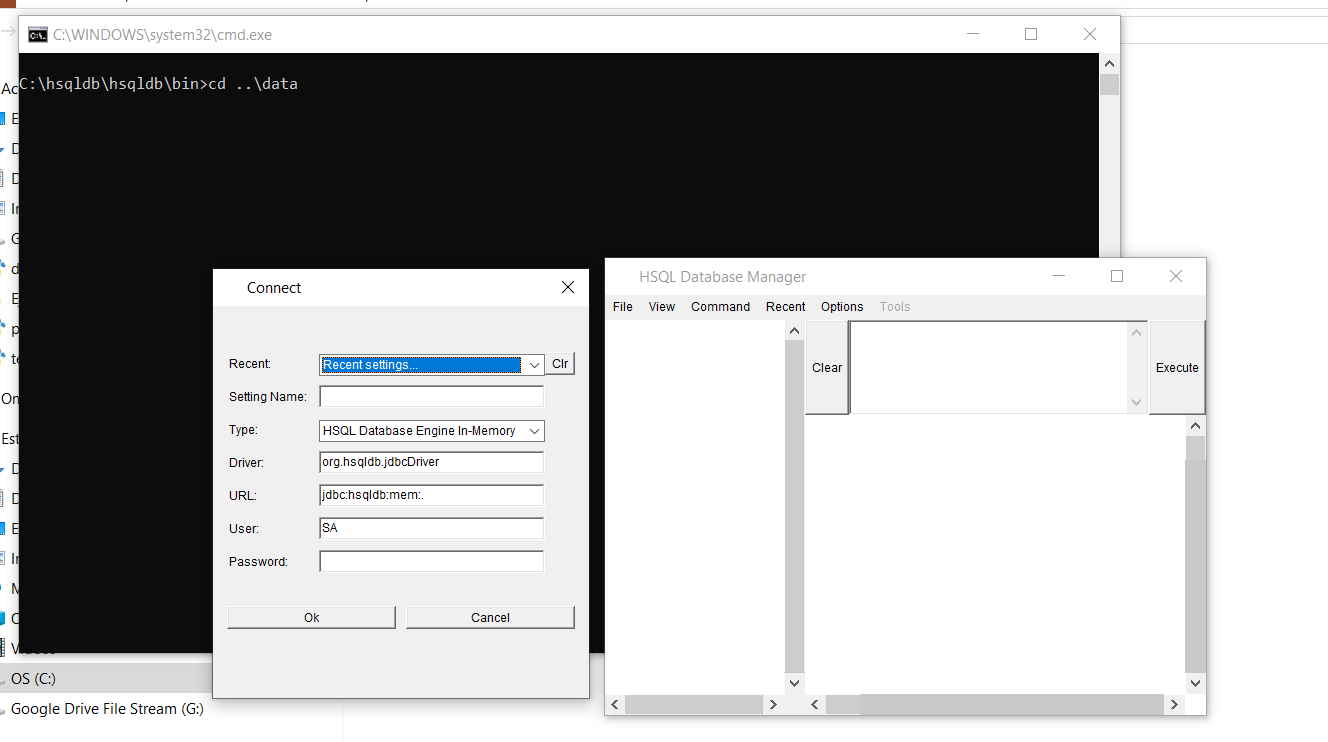


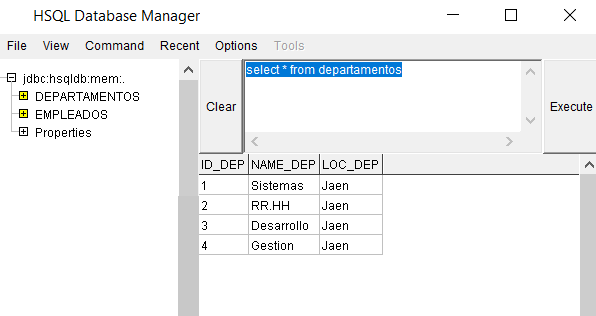
Ejercicio 2 :

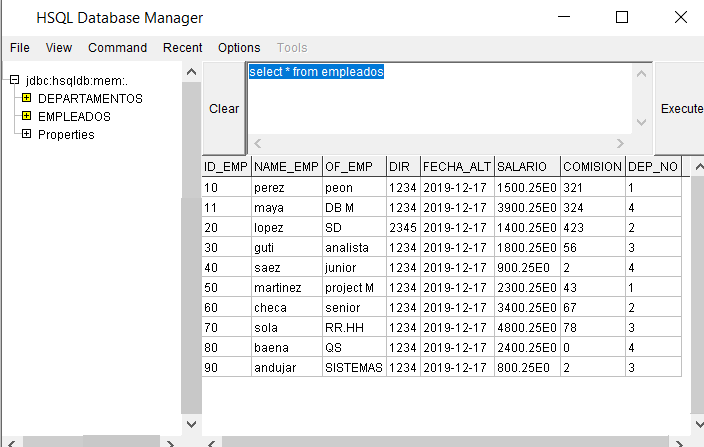




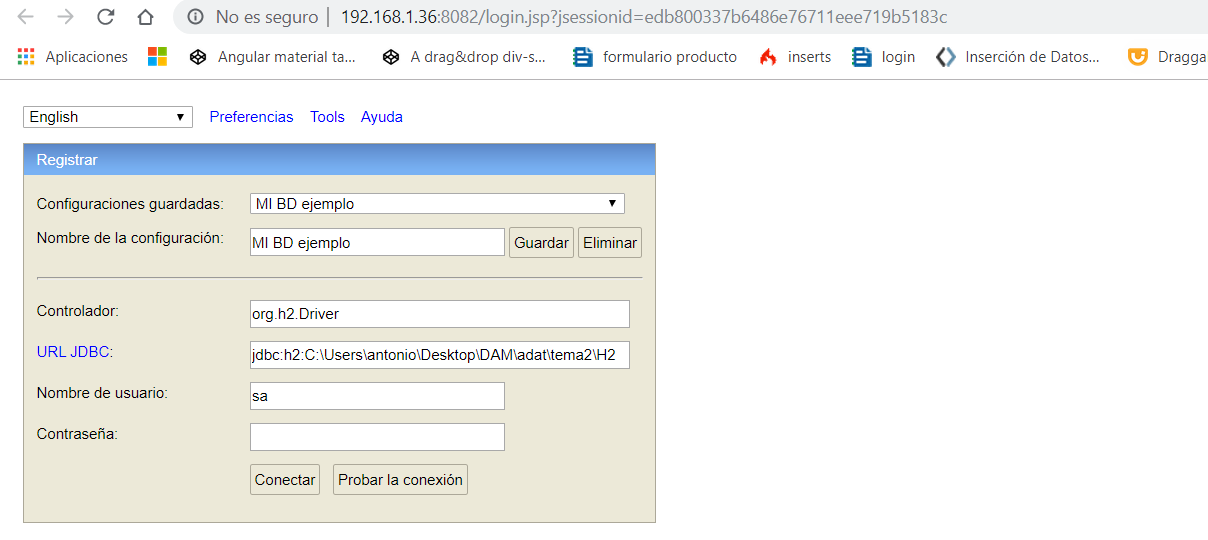
Ejercicio 3:

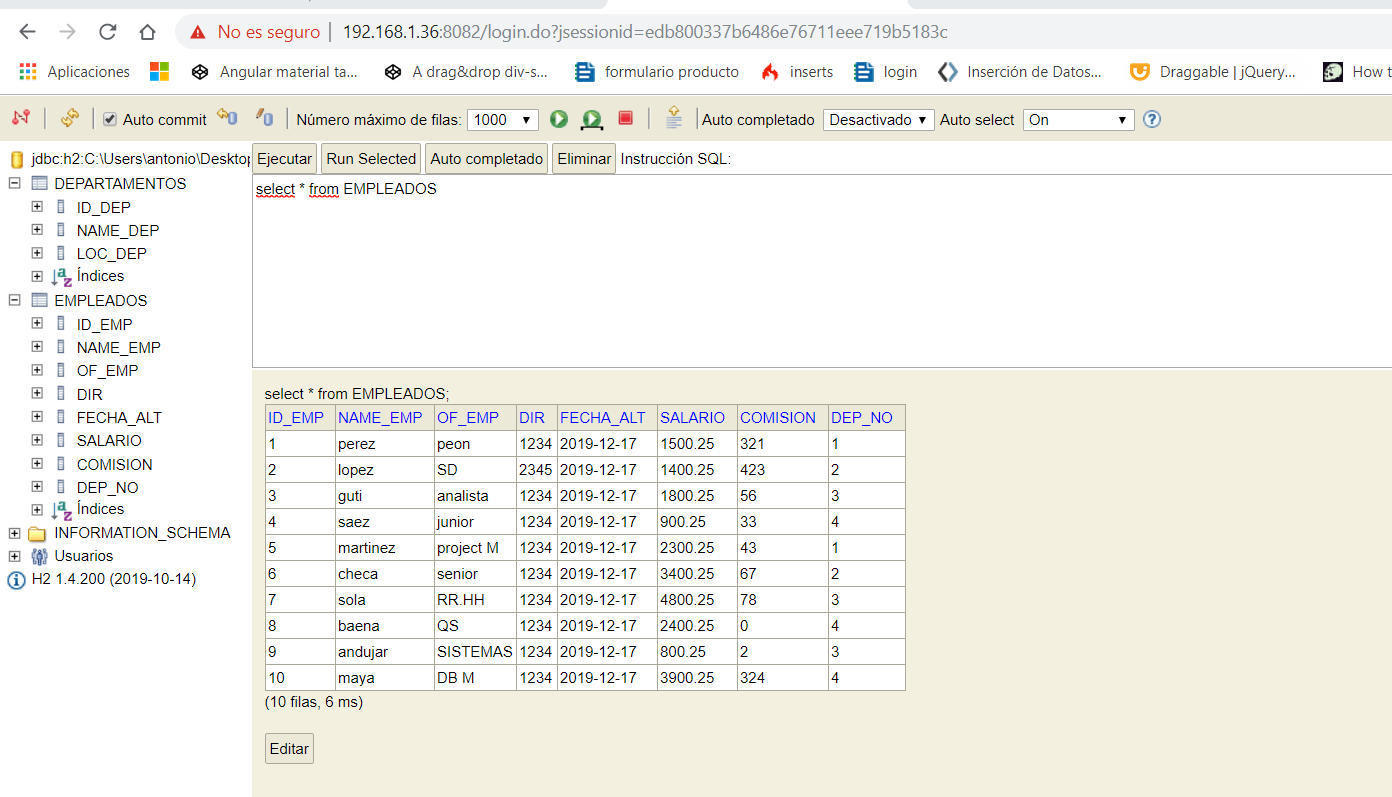
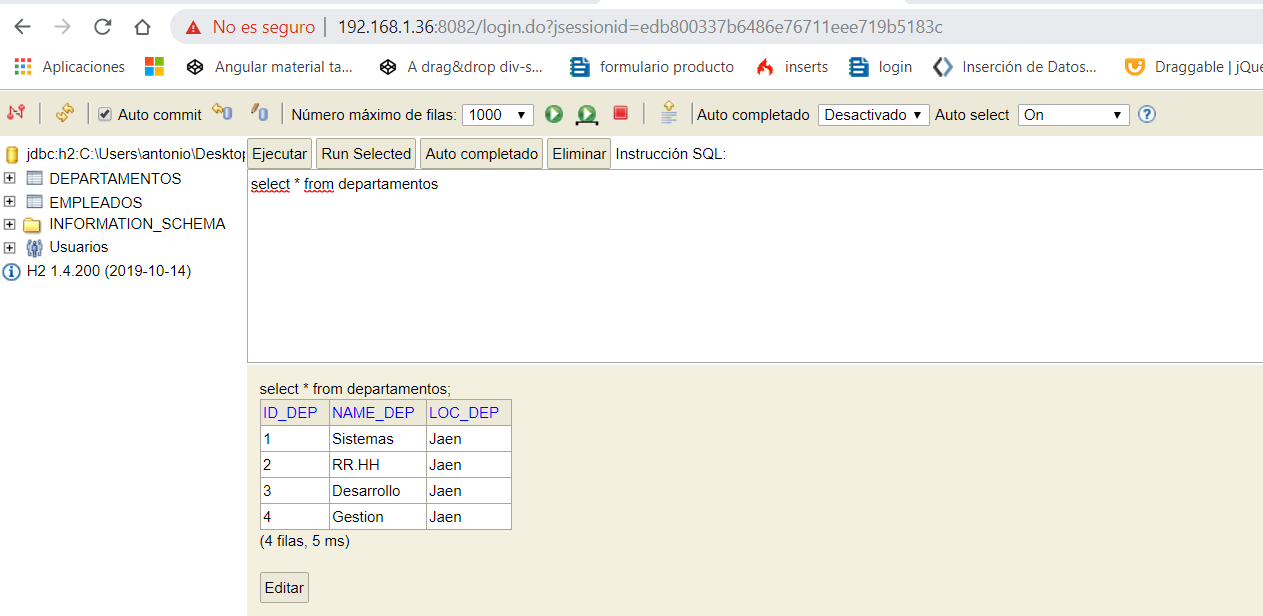




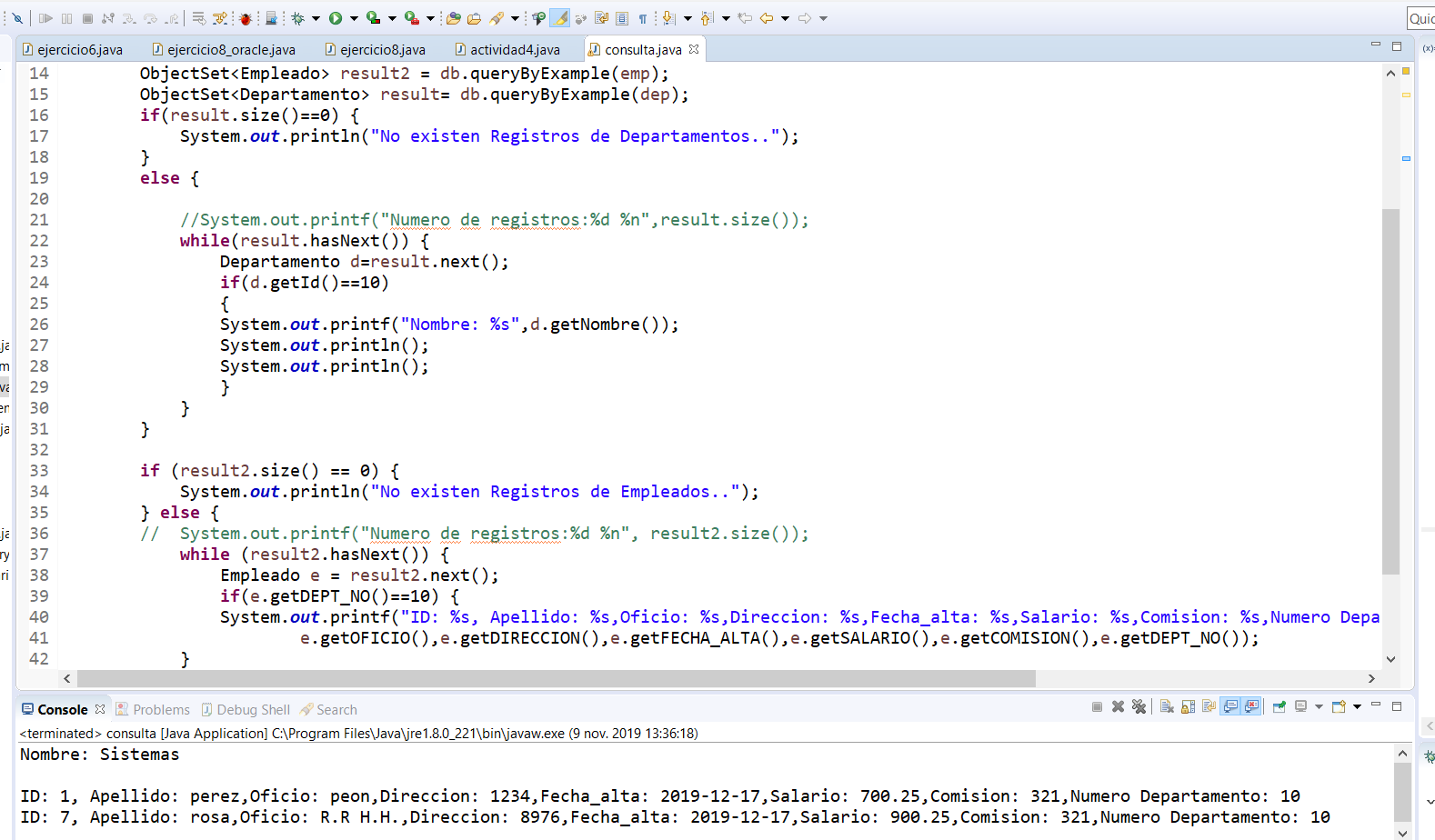


Ejercicio 4:





Ejercicio 5:



// código

package actividad2\_5;

import com.db4o.Db4oEmbedded;

import com.db4o.ObjectContainer;

import com.db4o.ObjectSet;

import com.sun.corba.se.spi.orbutil.fsm.Guard.Result;

public class consulta {

static String BDPer= "DBEmpleado.yap";

public static void main(String[] args) {

// TODO Auto-generated method stub

ObjectContainer db= Db4oEmbedded.openFile(Db4oEmbedded.newConfiguration(),BDPer);

Departamento dep= new Departamento(0,null,null);

Empleado emp=new Empleado(0, null, null, 0, null, 0.0, 0, 0);

ObjectSet<Empleado> result2 = db.queryByExample(emp);

ObjectSet<Departamento> result= db.queryByExample(dep);

if(result.size()==0) {

System.out.println("No existen Registros de Departamentos..");

}

else {

//System.out.printf("Numero de registros:%d %n",result.size());

while(result.hasNext()) {

Departamento d=result.next();

if(d.getId()==10)

{

System.out.printf("Nombre: %s",d.getNombre());

System.out.println();

System.out.println();

}

}

}

if (result2.size() == 0) {

System.out.println("No existen Registros de Empleados..");

} else {

// System.out.printf("Numero de registros:%d %n", result2.size());

while (result2.hasNext()) {

Empleado e = result2.next();

if(e.getDEPT\_NO()==10) {

System.out.printf("ID: %s, Apellido: %s,Oficio: %s,Direccion: %s,Fecha\_alta: %s,Salario: %s,Comision: %s,Numero Departamento: %s %n", e.getId(), e.getApellido(),

e.getOFICIO(),e.getDIRECCION(),e.getFECHA\_ALTA(),e.getSALARIO(),e.getCOMISION(),e.getDEPT\_NO());

}

}

db.close();

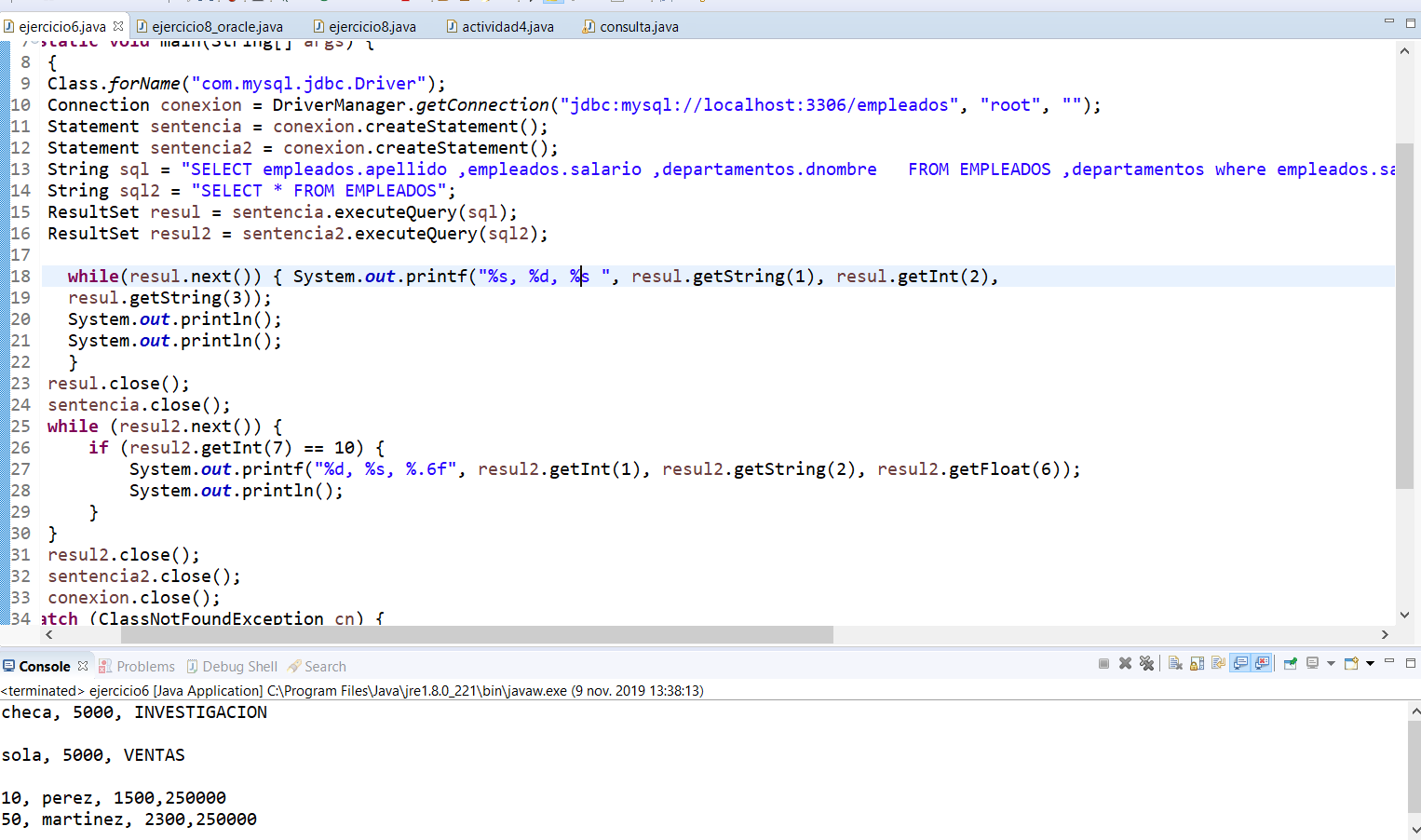
}

}

}

//

Ejercicio 6:



//código

**package** actividad2\_6;

**import** java.sql.\*;

**public** **class** ejercicio6 {

**public** **static** **void** main(String[] args) {

**try** {

Class.*forName*("com.mysql.jdbc.Driver");

Connection conexion = DriverManager.*getConnection*("jdbc:mysql://localhost:3306/empleados", "root", "");

Statement sentencia = conexion.createStatement();

Statement sentencia2 = conexion.createStatement();

String sql = "SELECT empleados.apellido ,empleados.salario ,departamentos.dnombre FROM EMPLEADOS ,departamentos where empleados.salario =(SELECT MAX(empleados.salario) FROM empleados) and departamentos.dept\_no=empleados.dept\_no;";

String sql2 = "SELECT \* FROM EMPLEADOS";

ResultSet resul = sentencia.executeQuery(sql);

ResultSet resul2 = sentencia2.executeQuery(sql2);

**while**(resul.next()) { System.***out***.printf("%s, %d, %s ", resul.getString(1), resul.getInt(2),

resul.getString(3));

System.***out***.println();

System.***out***.println();

}

resul.close();

sentencia.close();

**while** (resul2.next()) {

**if** (resul2.getInt(7) == 10) {

System.***out***.printf("%d, %s, %.6f", resul2.getInt(1), resul2.getString(2), resul2.getFloat(6));

System.***out***.println();

}

}

resul2.close();

sentencia2.close();

conexion.close();

} **catch** (ClassNotFoundException cn) {

cn.printStackTrace();

} **catch** (SQLException e) {

e.printStackTrace();

}

}

}

//

Ejercicio 7

//código

**package** actividad2\_7;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.sql.Statement;

**import** java.sql.\*;

**public** **class** ejercicio7 {

**public** **static** **void** main(String[] args) {

**try** {

Class.*forName*("sun.jdbc.odbc.JdbcOdbcDriver");

Connection conexion = DriverManager.*getConnection*("jdbc:odbc:Mysql-odbc://localhost:3306/empleados", "root", "");

Statement sentencia = conexion.createStatement();

Statement sentencia2 = conexion.createStatement();

String sql = "SELECT empleados.apellido ,empleados.salario ,departamentos.dnombre FROM EMPLEADOS ,departamentos where empleados.salario =(SELECT MAX(empleados.salario) FROM empleados) and departamentos.dept\_no=empleados.dept\_no;";

String sql2 = "SELECT \* FROM EMPLEADOS";

ResultSet resul = sentencia.executeQuery(sql);

ResultSet resul2 = sentencia2.executeQuery(sql2);

**while** (resul.next()) {

System.***out***.printf("%s, %d, %s ", resul.getString(1), resul.getInt(2), resul.getString(3));

System.***out***.println();

System.***out***.println();

}

resul.close();

sentencia.close();

**while** (resul2.next()) {

**if** (resul2.getInt(7) == 10) {

System.***out***.printf("%d, %s, %.6f", resul2.getInt(1), resul2.getString(2), resul2.getFloat(6));

System.***out***.println();

}

}

resul2.close();

sentencia2.close();

conexion.close();

} **catch** (ClassNotFoundException cn) {

cn.printStackTrace();

} **catch** (SQLException e) {

e.printStackTrace();

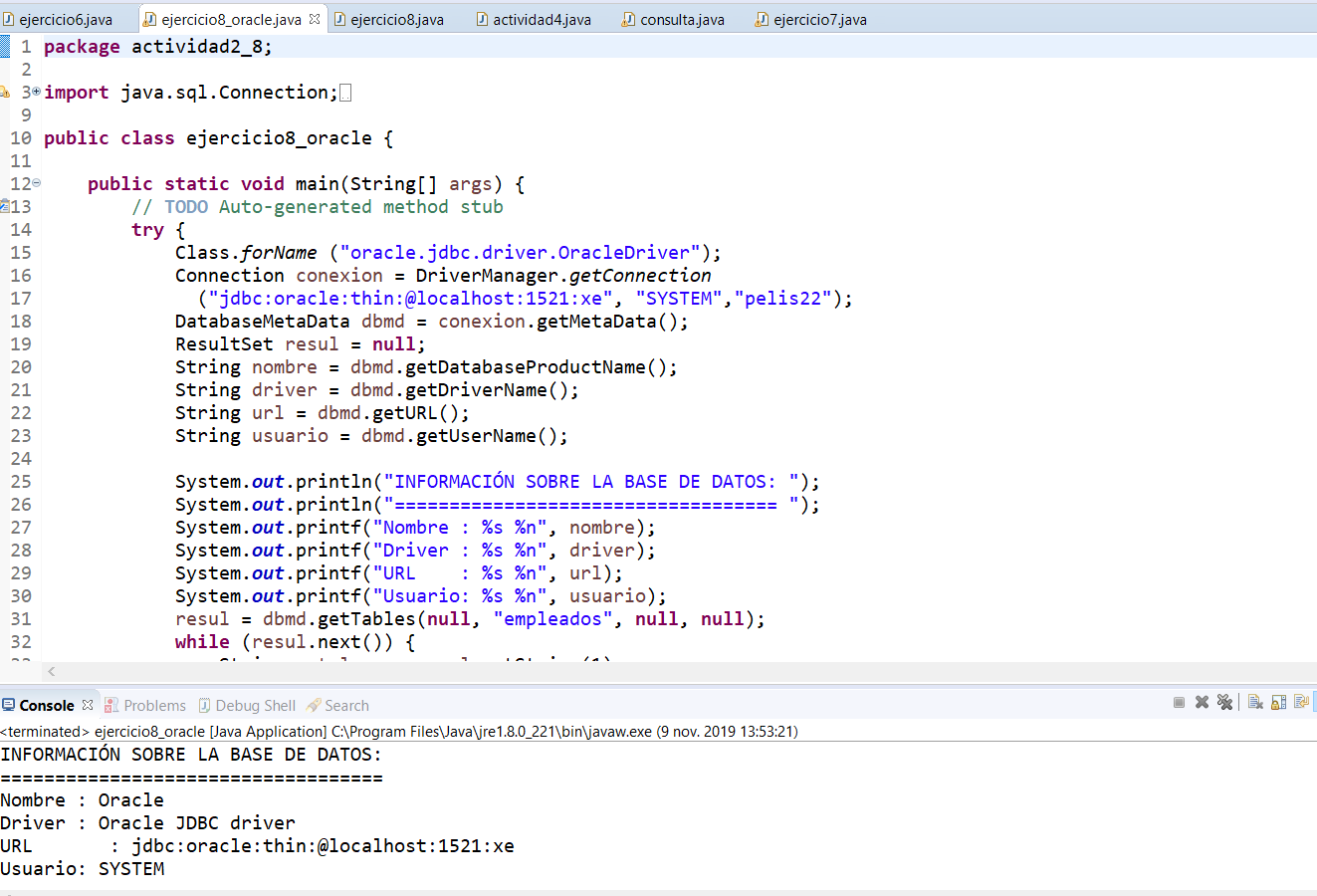
}

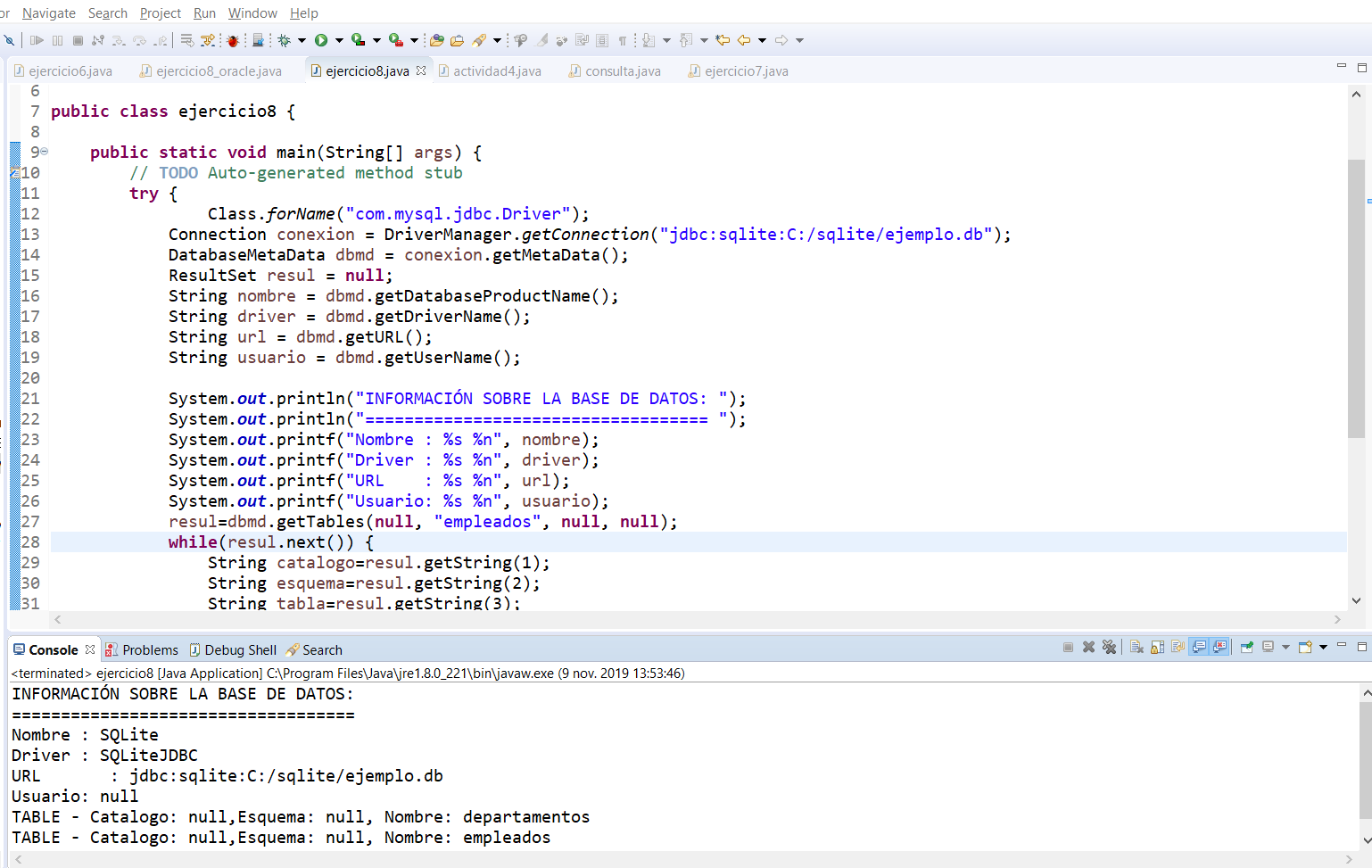
}

}

//

Ejercicio 8





//código

package actividad2\_8;

import java.sql.Connection;

import java.sql.DatabaseMetaData;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.\*;

public class ejercicio8\_oracle {

public static void main(String[] args) {

// TODO Auto-generated method stub

try {

Class.forName ("oracle.jdbc.driver.OracleDriver");

Connection conexion = DriverManager.getConnection

("jdbc:oracle:thin:@localhost:1521:xe", "SYSTEM","pelis22");

DatabaseMetaData dbmd = conexion.getMetaData();

ResultSet resul = null;

String nombre = dbmd.getDatabaseProductName();

String driver = dbmd.getDriverName();

String url = dbmd.getURL();

String usuario = dbmd.getUserName();

System.out.println("INFORMACIÓN SOBRE LA BASE DE DATOS: ");

System.out.println("=================================== ");

System.out.printf("Nombre : %s %n", nombre);

System.out.printf("Driver : %s %n", driver);

System.out.printf("URL : %s %n", url);

System.out.printf("Usuario: %s %n", usuario);

resul = dbmd.getTables(null, "empleados", null, null);

while (resul.next()) {

String catalogo = resul.getString(1);

String esquema = resul.getString(2);

String tabla = resul.getString(3);

String tipo = resul.getString(4);

System.out.printf("%s - Catalogo: %s,Esquema: %s, Nombre: %s %n", tipo, catalogo, esquema, tabla);

}

conexion.close();

} catch (ClassNotFoundException cn) {

cn.printStackTrace();

} catch (SQLException e) {

e.printStackTrace();

}

}

}

//

//código

**package** actividad2\_8;

**import** java.sql.\*;

**public** **class** ejercicio8 {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

**try** {

Class.*forName*("com.mysql.jdbc.Driver");

Connection conexion = DriverManager.*getConnection*("jdbc:sqlite:C:/sqlite/ejemplo.db");

DatabaseMetaData dbmd = conexion.getMetaData();

ResultSet resul = **null**;

String nombre = dbmd.getDatabaseProductName();

String driver = dbmd.getDriverName();

String url = dbmd.getURL();

String usuario = dbmd.getUserName();

System.***out***.println("INFORMACIÓN SOBRE LA BASE DE DATOS: ");

System.***out***.println("=================================== ");

System.***out***.printf("Nombre : %s %n", nombre);

System.***out***.printf("Driver : %s %n", driver);

System.***out***.printf("URL : %s %n", url);

System.***out***.printf("Usuario: %s %n", usuario);

resul=dbmd.getTables(**null**, "empleados", **null**, **null**);

**while**(resul.next()) {

String catalogo=resul.getString(1);

String esquema=resul.getString(2);

String tabla=resul.getString(3);

String tipo=resul.getString(4);

System.***out***.printf("%s - Catalogo: %s,Esquema: %s, Nombre: %s %n",tipo, catalogo, esquema ,tabla);

}

conexion.close();

} **catch** (ClassNotFoundException cn) {

cn.printStackTrace();

} **catch** (SQLException e) {

e.printStackTrace();

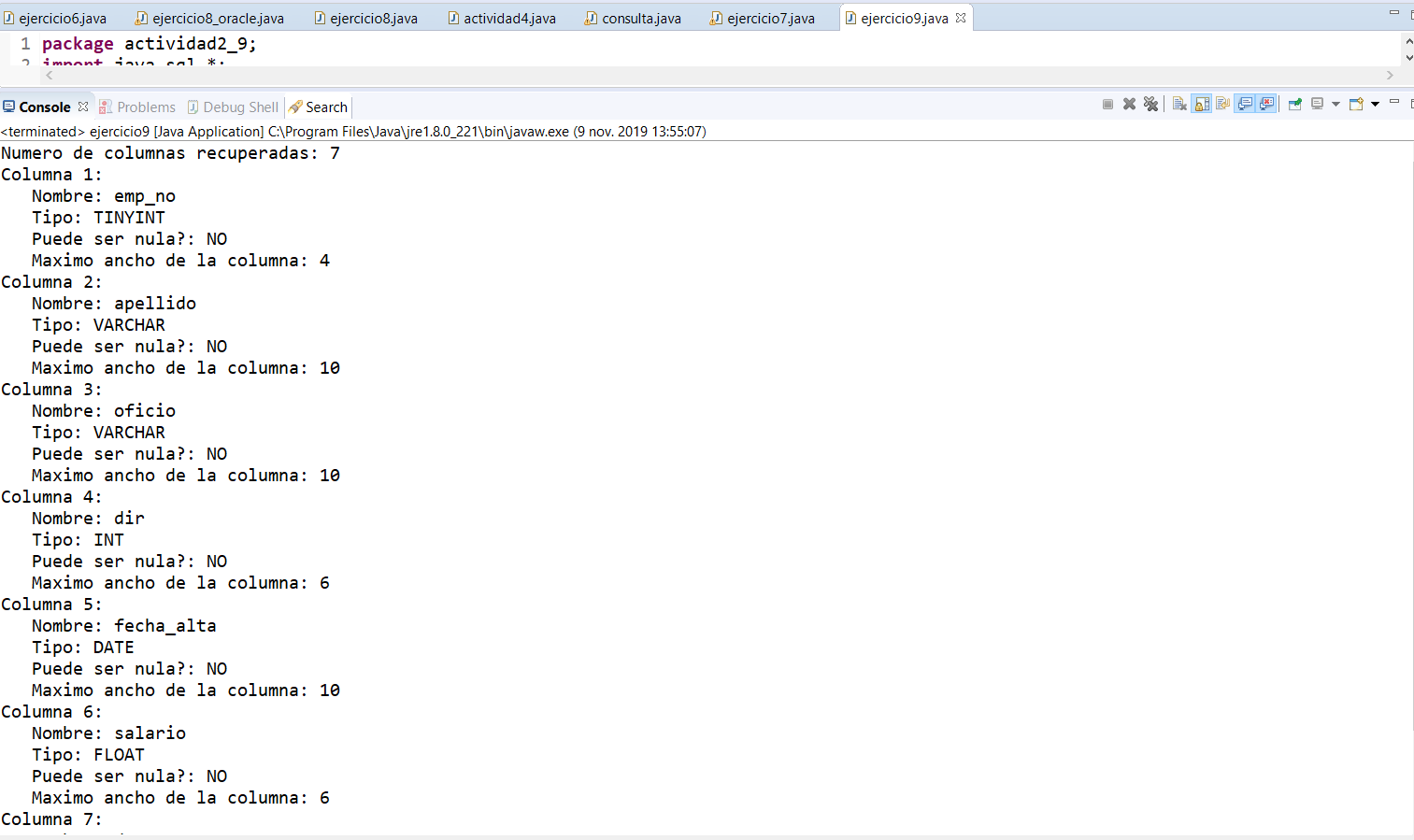
}

}

}

//

Ejercicio 9:



//código

**package** actividad2\_9;

**import** java.sql.\*;

**public** **class** ejercicio9 {

**public** **static** **void** main(String[] args) {

**try** {

// conexion a ORACLE

/\* Class.forName ("oracle.jdbc.driver.OracleDriver"); Connection

conexion = DriverManager.getConnection

("jdbc:oracle:thin:@localhost:1521:XE", "ejemplo", "ejemplo");

\*/

Class.*forName*("com.mysql.jdbc.Driver"); // Cargar el driver

Connection conexion = DriverManager.*getConnection*(

"jdbc:mysql://localhost:3306/empleados", "root", "");

Statement sentencia = conexion.createStatement();

ResultSet rs = sentencia

.executeQuery("SELECT \* FROM empleados");

ResultSetMetaData rsmd = rs.getMetaData();

**int** nColumnas = rsmd.getColumnCount();

String nula;

System.***out***.printf("Numero de columnas recuperadas: %d%n", nColumnas);

**for** (**int** i = 1; i <= nColumnas; i++) {

System.***out***.printf("Columna %d: %n ", i);

System.***out***.printf(" Nombre: %s %n Tipo: %s %n ",

rsmd.getColumnName(i), rsmd.getColumnTypeName(i));

**if** (rsmd.isNullable(i) == 0)

nula = "NO";

**else**

nula = "SI";

System.***out***.printf(" Puede ser nula?: %s %n ", nula);

System.***out***.printf(" Maximo ancho de la columna: %d %n",

rsmd.getColumnDisplaySize(i));

}// for

sentencia.close();

rs.close();

conexion.close();

} **catch** (ClassNotFoundException cn) {

cn.printStackTrace();

} **catch** (SQLException e) {

e.printStackTrace();

}

}

}

//

Ejercicio 10:

//código

package actividad10;

import java.sql.\*;

import java.util.Date;

import java.text.DateFormat;

import java.text.SimpleDateFormat;

public class ejercicio10a {

public static void main(String[] args) {

// TODO Auto-generated method stub

try {

Date date = new Date();

DateFormat fechaHora = new SimpleDateFormat("yyyy-MM-dd");

String empleado = args[0];

String apellido = args[1];

String oficio = args[2];

String dir = args[3];

String fecha = fechaHora.format(date);

String salario = args[4];

String departamento = args[5];

boolean direxiste = false;

boolean id = false;

boolean dep = false;

// conexion a ORACLE

/\*

\* Class.forName ("oracle.jdbc.driver.OracleDriver"); Connection conexion =

\* DriverManager.getConnection ("jdbc:oracle:thin:@localhost:1521:XE",

\* "ejemplo", "ejemplo");

\*/

Class.forName("com.mysql.jdbc.Driver"); // Cargar el driver

Connection conexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/empleados", "root", "");

//

Statement sentencia2 = conexion.createStatement();

String sql2 = "SELECT emp\_no, dir ,dept\_no FROM empleados";

Statement sentencia = conexion.createStatement();

ResultSet resul2 = sentencia2.executeQuery(sql2);

while (resul2.next()) {

if (resul2.getInt(1) == Integer.parseInt(empleado)) {

id = true;

} else if (resul2.getInt(2) == Integer.parseInt(dir)) {

direxiste = true;

} else if (resul2.getInt(3) == Integer.parseInt(departamento)) {

dep = true;

}

}

if (dep == false) {

System.out.println("el departamento no existe.");

} else if (direxiste == false) {

System.out.println("el director no existe.");

} else if (Double.parseDouble(salario) <= 0) {

System.out.println("salario incorrecto.");

} else if (id == true) {

System.out.println("el id del empleado ya existe.");

id = true;

} else if (id == false && direxiste == true && dep == true && Double.parseDouble(salario) >= 0 && apellido!=null && oficio!=null) {

String sql = String.format("insert into empleados values (%s, '%s', '%s', '%s', '%s', '%s', '%s')",

empleado, apellido, oficio, dir, fecha, salario, departamento);

int filas = sentencia.executeUpdate(sql);

// System.out.printf("Filas afectadas: %d %n", filas);

sentencia.close();

}

else if (apellido==null || oficio==null && apellido=="" || oficio=="") {

System.out.println("el apellido o el oficio estan vacios.");

id = true;

}

//

sentencia2.close();

conexion.close();

} catch (ClassNotFoundException cn) {

cn.printStackTrace();

} catch (SQLException e) {

e.printStackTrace();

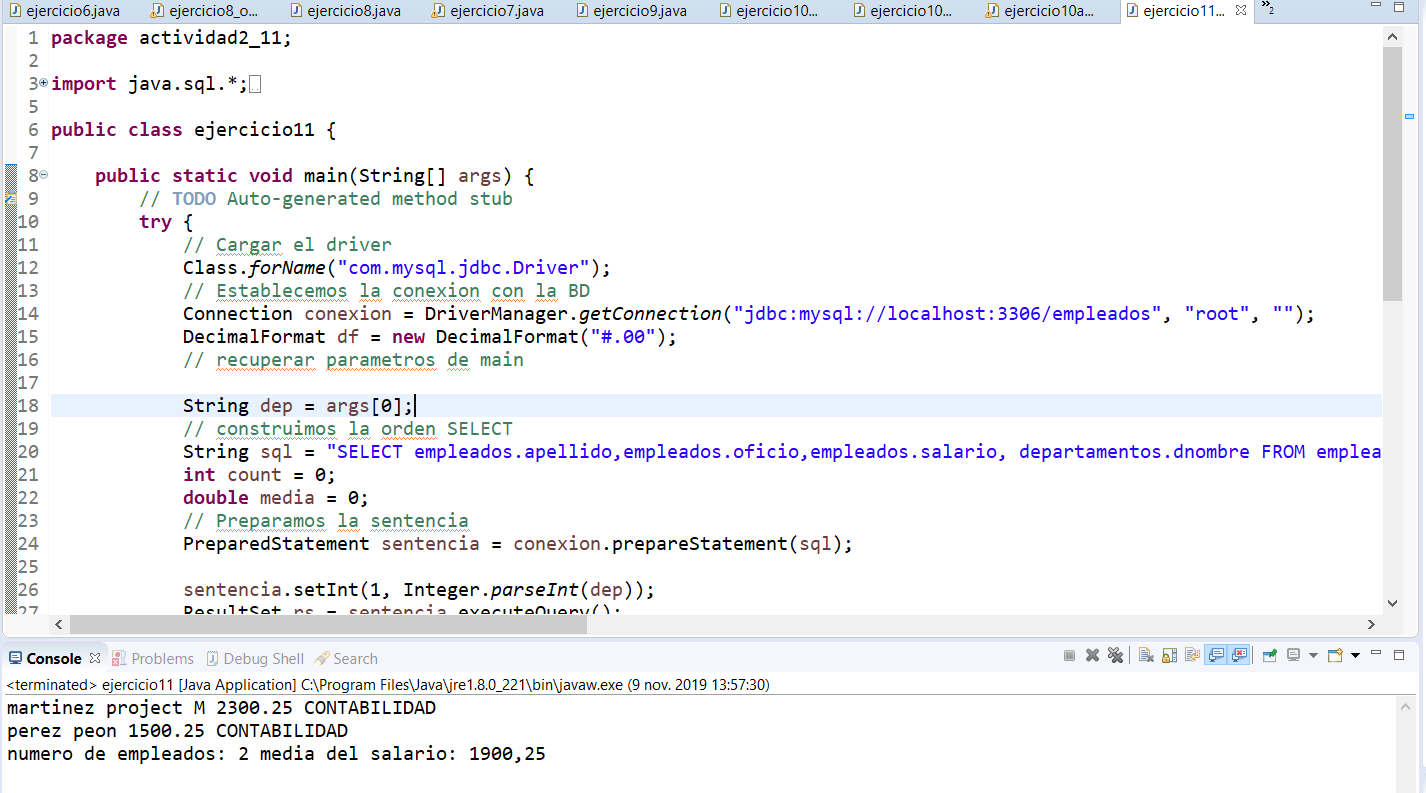
}

}

}

//

Ejercicio 11:

//código

package actividad2\_11;

import java.sql.\*;

import java.text.DecimalFormat;

public class ejercicio11 {

public static void main(String[] args) {

// TODO Auto-generated method stub

try {

// Cargar el driver

Class.forName("com.mysql.jdbc.Driver");

// Establecemos la conexion con la BD

Connection conexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/empleados", "root", "");

DecimalFormat df = new DecimalFormat("#.00");

// recuperar parametros de main

String dep = args[0];

// construimos la orden SELECT

String sql = "SELECT empleados.apellido,empleados.oficio,empleados.salario, departamentos.dnombre FROM empleados,departamentos WHERE empleados.dept\_no in(SELECT departamentos.dept\_no from departamentos where departamentos.dept\_no=?) AND departamentos.dept\_no=empleados.dept\_no ORDER BY 1";

int count = 0;

double media = 0;

// Preparamos la sentencia

PreparedStatement sentencia = conexion.prepareStatement(sql);

sentencia.setInt(1, Integer.parseInt(dep));

ResultSet rs = sentencia.executeQuery();

while (rs.next()) {

System.out.printf("%s %s %s %s ", rs.getString("apellido"), rs.getString("oficio"),

rs.getDouble("salario"), rs.getString("dnombre"));

System.out.println();

count++;

media += rs.getDouble("salario");

}

if (count == 0) {

System.out.println(" el departamento no existe");

}

if (media > 0) {

media = media / count;

System.out.println("numero de empleados: " + count + " media del salario: " + df.format(media));

}

rs.close();// liberar recursos

sentencia.close();

conexion.close();

} catch (Exception e) {

e.printStackTrace();

}

}

}

//

Ejercicio 12:

//código:

**package** actividad2\_12;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.SQLException;

**import** com.mysql.jdbc.CallableStatement;

**public** **class** ejercicio12 {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

**try** {

Class.*forName*("oracle.jdbc.driver.OracleDriver");

Connection conexion = DriverManager.*getConnection*("jdbc:oracle:thin:@localhost:1521:xe", "SYSTEM",

"pelis22");

} **catch** (ClassNotFoundException cn) {

cn.printStackTrace();

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **static** **void** executeStoredProcedure(Connection conexion) **throws** SQLException {

conexion = DriverManager.*getConnection*("jdbc:oracle:thin:@localhost:1521:xe", "SYSTEM",

"pelis22");

**try**(CallableStatement cstmt = (CallableStatement) conexion.prepareCall("{call dbo.Getsalariomedio(?)}");) {

cstmt.getInt(1);

cstmt.registerOutParameter(1, java.sql.Types.***INTEGER***);

cstmt.execute();

System.***out***.println("Salario medio: " + cstmt.getInt(1));

}

}

}

//

//código :

**package** actividad2\_12;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.SQLException;

**import** com.mysql.jdbc.CallableStatement;

**public** **class** ejercicio12mysql {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

**try** {

Class.*forName*("com.mysql.jdbc.Driver"); // Cargar el driver

Connection conexion = DriverManager.*getConnection*("jdbc:mysql://localhost:3306/empleados", "root", "");

*executeStoredProcedure*(conexion);

} **catch** (ClassNotFoundException cn) {

cn.printStackTrace();

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **static** **void** executeStoredProcedure(Connection conexion) **throws** SQLException {

conexion = DriverManager.*getConnection*("jdbc:mysql://localhost:3306/empleados", "root", "");

**try** (CallableStatement cstmt = (CallableStatement) conexion

.prepareCall("{call dbo.Getsalariomedio(?)}");) {

cstmt.getInt(1);

cstmt.registerOutParameter(2, java.sql.Types.***INTEGER***);

cstmt.execute();

System.***out***.println("MANAGER ID: " + cstmt.getInt(2));

}

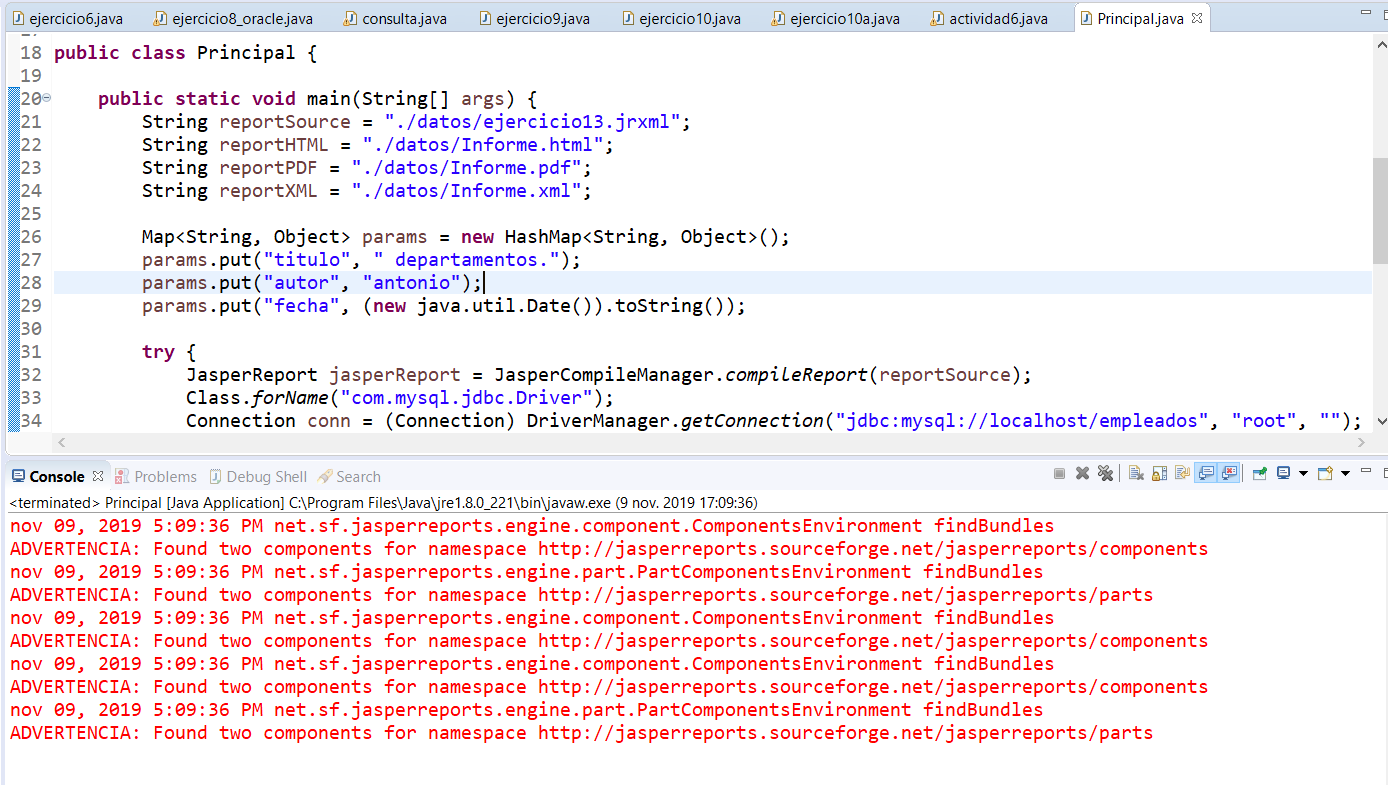
}

}

//

Ejercicio 13:





//código

**import** java.util.Map;

**import** com.mysql.jdbc.exceptions.jdbc4.CommunicationsException;

**import** com.mysql.jdbc.Connection;

**import** net.sf.jasperreports.engine.JRException;

**import** net.sf.jasperreports.engine.JasperCompileManager;

**import** net.sf.jasperreports.engine.JasperExportManager;

**import** net.sf.jasperreports.engine.JasperFillManager;

**import** net.sf.jasperreports.engine.JasperPrint;

**import** net.sf.jasperreports.engine.JasperReport;

**import** net.sf.jasperreports.view.JasperViewer;

**import** java.sql.DriverManager;

**import** java.sql.SQLException;

**import** java.util.HashMap;

**public** **class** Principal {

**public** **static** **void** main(String[] args) {

String reportSource = "./datos/ejercicio13.jrxml";

String reportHTML = "./datos/Informe.html";

String reportPDF = "./datos/Informe.pdf";

String reportXML = "./datos/Informe.xml";

Map<String, Object> params = **new** HashMap<String, Object>();

params.put("titulo", " departamentos.");

params.put("autor", "antonio");

params.put("fecha", (**new** java.util.Date()).toString());

**try** {

JasperReport jasperReport = JasperCompileManager.*compileReport*(reportSource);

Class.*forName*("com.mysql.jdbc.Driver");

Connection conn = (Connection) DriverManager.*getConnection*("jdbc:mysql://localhost/empleados", "root", "");

JasperPrint MiInforme = JasperFillManager.*fillReport*(jasperReport, params, conn);

JasperViewer.*viewReport*(MiInforme);

JasperExportManager.*exportReportToHtmlFile*(MiInforme, reportHTML);

JasperExportManager.*exportReportToPdfFile*(MiInforme, reportPDF);

JasperExportManager.*exportReportToXmlFile*(MiInforme, reportXML, **false**);

} **catch** (CommunicationsException c) {

System.***out***.println(" Error de comunicación con la BD. No está arrancada.");

} **catch** (ClassNotFoundException e) {

System.***out***.println(" Error driver. ");

} **catch** (SQLException e) {

System.***out***.println(" Error al ejecutar sentencia SQL ");

} **catch** (JRException ex) {

System.***out***.println(" Error Jasper.");

ex.printStackTrace();

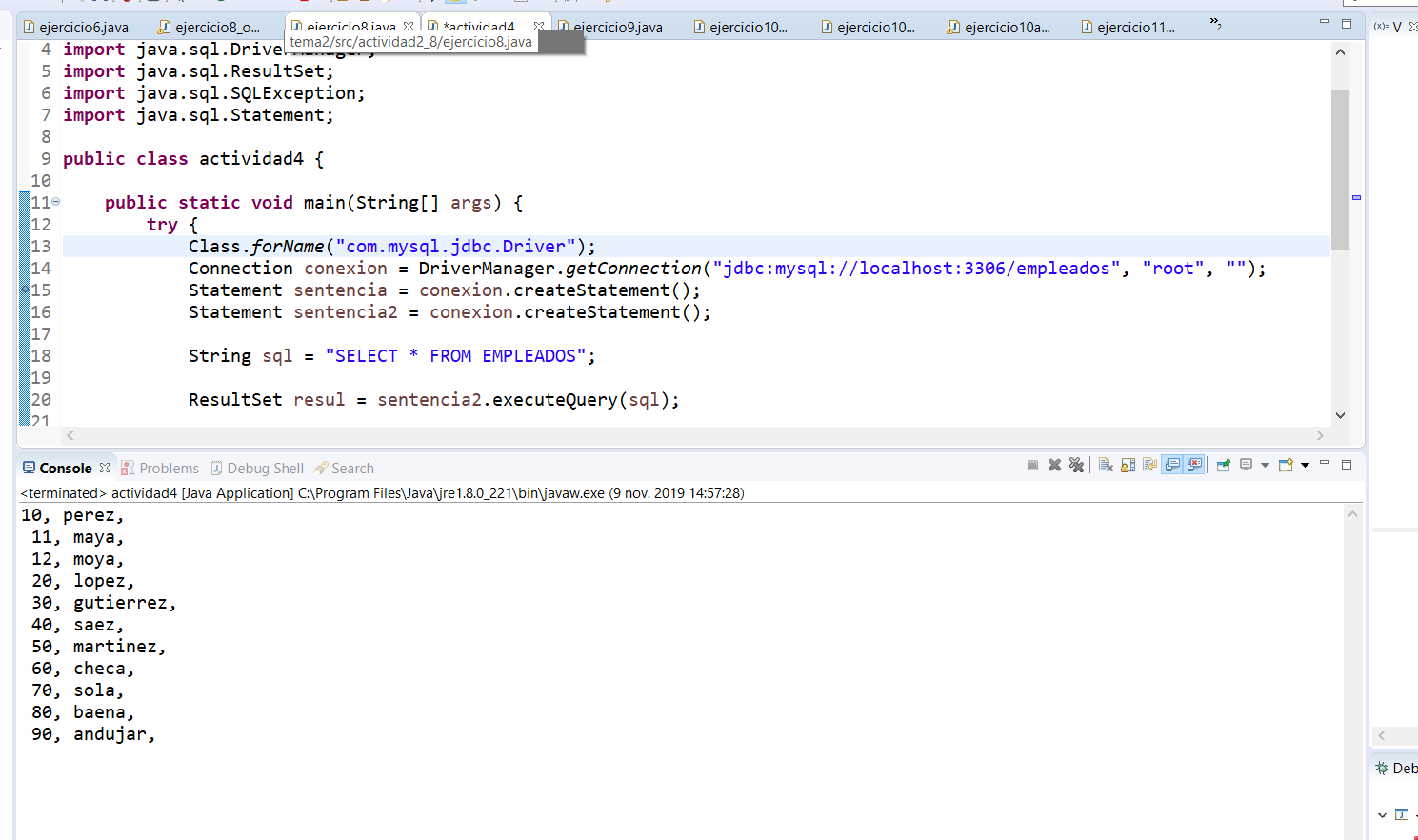
}

}

}

//

Ejercicio 4c:



//codigo

**package** comprueba;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.sql.Statement;

**public** **class** actividad4 {

**public** **static** **void** main(String[] args) {

**try** {

Class.*forName*("com.mysql.jdbc.Driver");

Connection conexion = DriverManager.*getConnection*("jdbc:mysql://localhost:3306/empleados", "root", "");

Statement sentencia = conexion.createStatement();

Statement sentencia2 = conexion.createStatement();

String sql = "SELECT \* FROM EMPLEADOS";

ResultSet resul = sentencia2.executeQuery(sql);

**while** (resul.next()) {

System.***out***.printf("%d, %s, %n ", resul.getInt("EMP\_NO"), resul.getString("APELLIDO"));

}

resul.close();

sentencia.close();

conexion.close();

} **catch** (ClassNotFoundException cn) {

cn.printStackTrace();

} **catch** (SQLException e) {

e.printStackTrace();

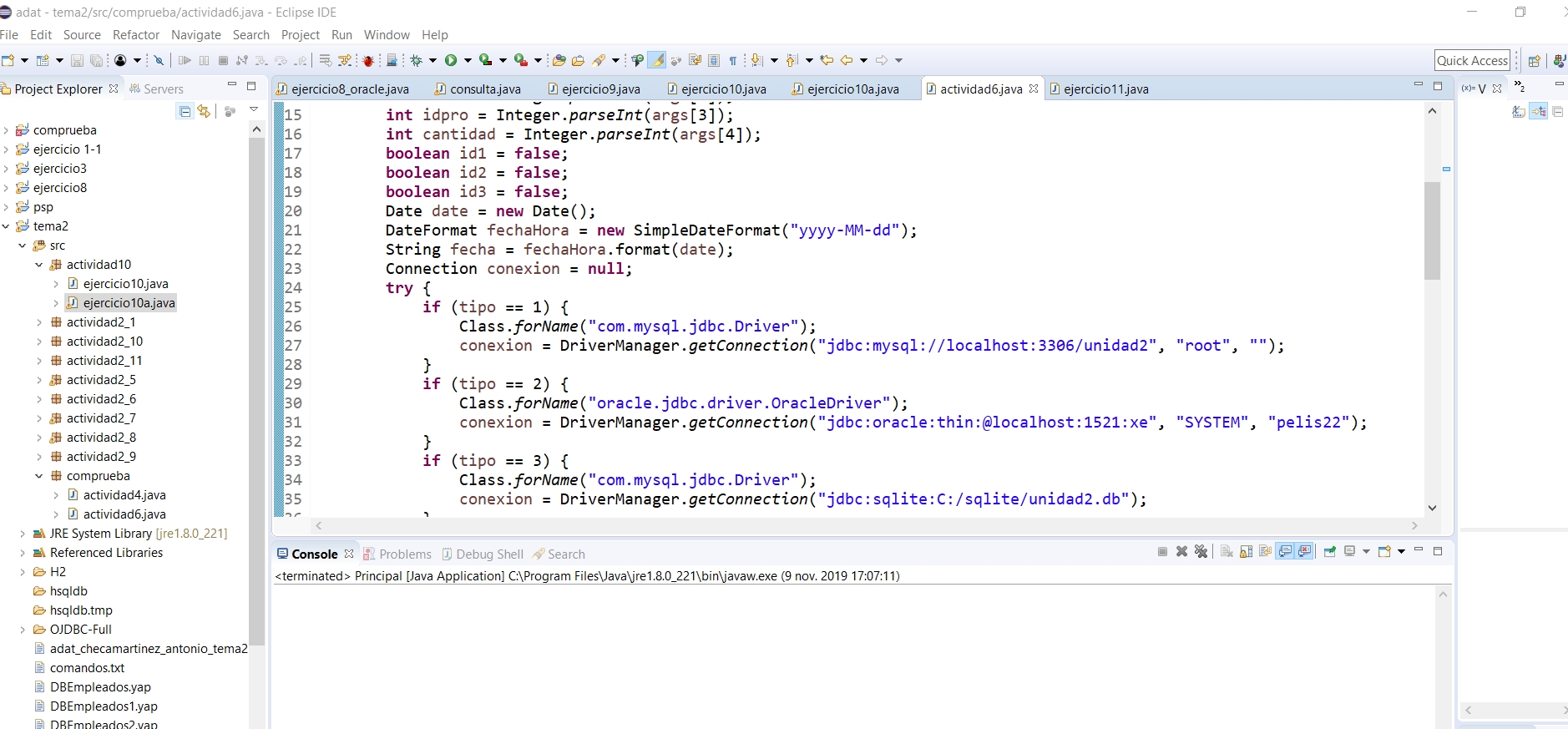
}

}

}

//

Ejercicio 6:



//código:

**package** comprueba;

**import** java.sql.\*;

**import** java.util.Date;

**import** java.text.DateFormat;

**import** java.text.SimpleDateFormat;

**public** **class** actividad6 {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

**int** tipo = Integer.*parseInt*(args[0]);

**int** id = Integer.*parseInt*(args[1]);

**int** idcli = Integer.*parseInt*(args[2]);

**int** idpro = Integer.*parseInt*(args[3]);

**int** cantidad = Integer.*parseInt*(args[4]);

**boolean** id1 = **false**;

**boolean** id2 = **false**;

**boolean** id3 = **false**;

Date date = **new** Date();

DateFormat fechaHora = **new** SimpleDateFormat("yyyy-MM-dd");

String fecha = fechaHora.format(date);

Connection conexion = **null**;

**try** {

**if** (tipo == 1) {

Class.*forName*("com.mysql.jdbc.Driver");

conexion = DriverManager.*getConnection*("jdbc:mysql://localhost:3306/unidad2", "root", "");

}

**if** (tipo == 2) {

Class.*forName*("oracle.jdbc.driver.OracleDriver");

conexion = DriverManager.*getConnection*("jdbc:oracle:thin:@localhost:1521:xe", "SYSTEM", "pelis22");

}

**if** (tipo == 3) {

Class.*forName*("com.mysql.jdbc.Driver");

conexion = DriverManager.*getConnection*("jdbc:sqlite:C:/sqlite/unidad2.db");

}

String sql1 = "SELECT id FROM productos";

String sql2 = "SELECT id FROM clientes";

String sql3 = "SELECT IDVENTA FROM ventas";

Statement sentencia = conexion.createStatement();

ResultSet resul = sentencia.executeQuery(sql1);

**while** (resul.next()) {

**if** (resul.getInt(1) == id) {

System.***out***.println("el id del producto ya existe.");

id1 = **true**;

}

}

ResultSet resul2 = sentencia.executeQuery(sql2);

**while** (resul2.next()) {

**if** (resul2.getInt(1) == id) {

System.***out***.println("el id del cliente ya existe.");

id2 = **true**;

}

}

ResultSet resul3 = sentencia.executeQuery(sql3);

**while** (resul3.next()) {

**if** (resul3.getInt(1) == id) {

System.***out***.println("el id de la venta ya existe.");

id3 = **true**;

}

}

**if**(cantidad<=0) {

System.***out***.println("la cantidad es incorrecta");

}

**if** (id1 == **false** && id2 == **false** && id3 == **false** && cantidad > 0) {

String sql = String.*format*("insert into ventas values (%i ,'%s', '%i', '%i', '%i',)", id, fecha, idcli,

idpro, cantidad);

**int** filas = sentencia.executeUpdate(sql);

System.***out***.printf("Filas afectadas: %d %n", filas);

}

sentencia.close();

conexion.close();

} **catch** (ClassNotFoundException cn) {

cn.printStackTrace();

} **catch** (SQLException e) {

e.printStackTrace();

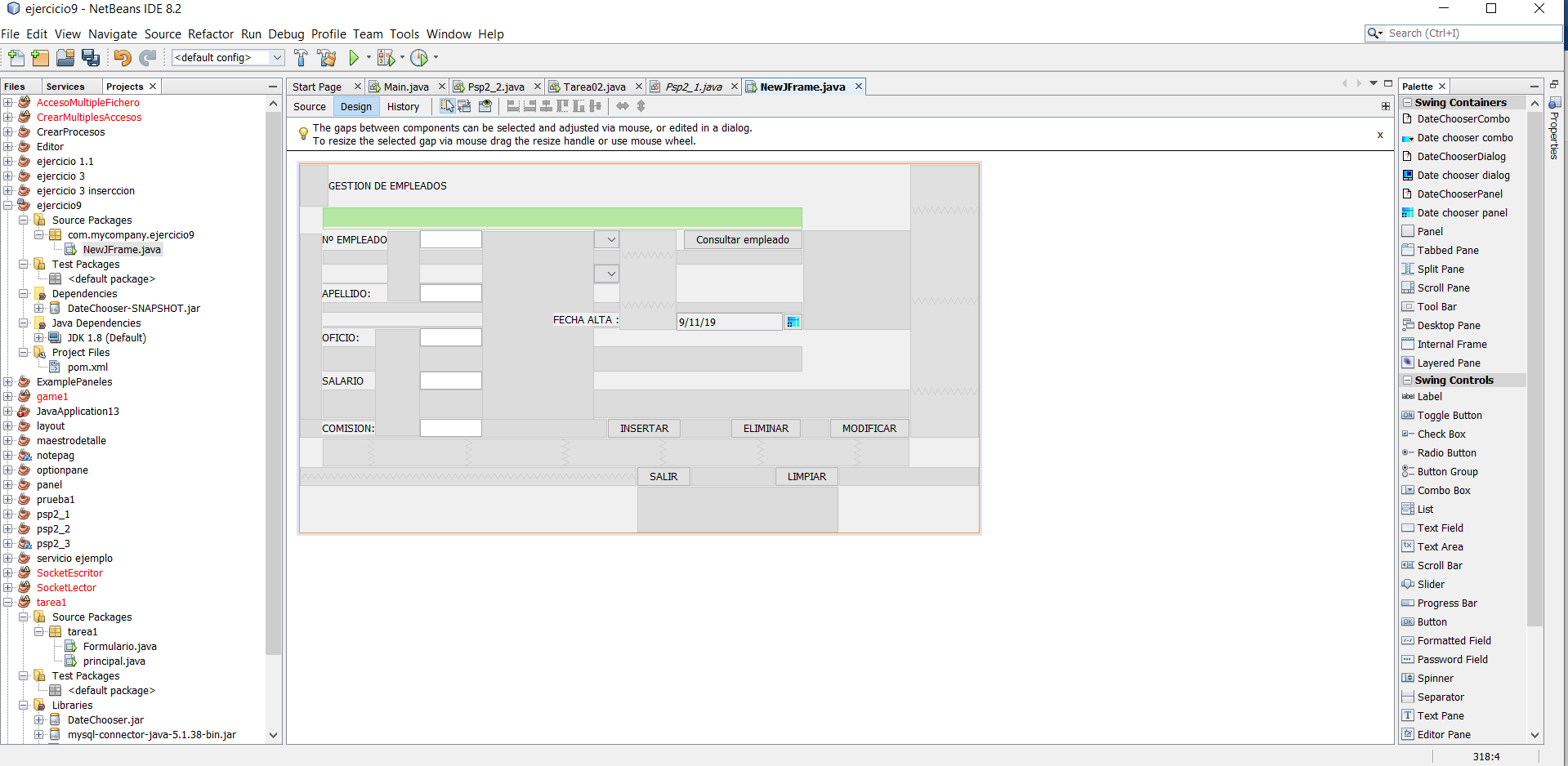
}

}

}

//

Ejercicio9:



//código

/\*

\* 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 com.mycompany.ejercicio9;

import javax.swing.JFrame;

import javax.swing.JOptionPane;

import java.sql.\*;

import java.util.Date;

import java.text.DateFormat;

import java.text.SimpleDateFormat;

/\*\*

\*

\* @author antonio

\*/

public class NewJFrame extends javax.swing.JFrame {

/\*\*

\* Creates new form NewJFrame

\*/

public NewJFrame() {

initComponents();

try {

Class.forName("com.mysql.jdbc.Driver");

Connection conexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/unidad2", "root", "");

String sql = "SELECT dept\_no,nombre from departamentos";

String sql2 = "SELECT dir from empleados";

PreparedStatement sentencia = conexion.prepareStatement(sql);

PreparedStatement sentencia2 = conexion.prepareStatement(sql2);

ResultSet rs = sentencia.executeQuery();

while (rs.next()) {

jComboBox1.addItem(rs.getInt("dnombre") + "|" + rs.getString("dnombre"));

}

ResultSet rs2 = sentencia2.executeQuery();

while (rs2.next()) {

jComboBox2.addItem(Integer.toString(rs2.getInt("dir")));

}

rs.close();// liberar recursos

sentencia.close();

rs2.close();// liberar recursos

sentencia2.close();

conexion.close();

} catch (ClassNotFoundException cn) {

cn.printStackTrace();

} catch (SQLException e) {

e.printStackTrace();

}

}

/\*\*

\* 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() {

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jLabel6 = new javax.swing.JLabel();

jLabel7 = new javax.swing.JLabel();

jComboBox1 = new javax.swing.JComboBox<>();

jTextField1 = new javax.swing.JTextField();

jTextField6 = new javax.swing.JTextField();

jTextField7 = new javax.swing.JTextField();

jTextField8 = new javax.swing.JTextField();

jTextField9 = new javax.swing.JTextField();

jComboBox2 = new javax.swing.JComboBox<>();

dateChooserCombo1 = new datechooser.beans.DateChooserCombo();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jButton4 = new javax.swing.JButton();

jButton5 = new javax.swing.JButton();

jButton6 = new javax.swing.JButton();

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

jLabel1.setText("APELLIDO:");

jLabel2.setText("OFICIO:");

jLabel3.setText("SALARIO");

jLabel4.setText("FECHA ALTA :");

jLabel5.setText("GESTION DE EMPLEADOS");

jLabel6.setText("COMISION:");

jLabel7.setText("Nº EMPLEADO");

dateChooserCombo1.setCurrentView(new datechooser.view.appearance.AppearancesList("Dali",

new datechooser.view.appearance.ViewAppearance("custom",

new datechooser.view.appearance.swing.SwingCellAppearance(new java.awt.Font("Tahoma", java.awt.Font.PLAIN, 13),

new java.awt.Color(0, 0, 0),

new java.awt.Color(0, 0, 255),

false,

true,

new datechooser.view.appearance.swing.ButtonPainter()),

new datechooser.view.appearance.swing.SwingCellAppearance(new java.awt.Font("Tahoma", java.awt.Font.PLAIN, 13),

new java.awt.Color(0, 0, 0),

new java.awt.Color(0, 0, 255),

true,

true,

new datechooser.view.appearance.swing.ButtonPainter()),

new datechooser.view.appearance.swing.SwingCellAppearance(new java.awt.Font("Tahoma", java.awt.Font.PLAIN, 13),

new java.awt.Color(0, 0, 255),

new java.awt.Color(0, 0, 255),

false,

true,

new datechooser.view.appearance.swing.ButtonPainter()),

new datechooser.view.appearance.swing.SwingCellAppearance(new java.awt.Font("Tahoma", java.awt.Font.PLAIN, 13),

new java.awt.Color(128, 128, 128),

new java.awt.Color(0, 0, 255),

false,

true,

new datechooser.view.appearance.swing.LabelPainter()),

new datechooser.view.appearance.swing.SwingCellAppearance(new java.awt.Font("Tahoma", java.awt.Font.PLAIN, 13),

new java.awt.Color(0, 0, 0),

new java.awt.Color(0, 0, 255),

false,

true,

new datechooser.view.appearance.swing.LabelPainter()),

new datechooser.view.appearance.swing.SwingCellAppearance(new java.awt.Font("Tahoma", java.awt.Font.PLAIN, 13),

new java.awt.Color(0, 0, 0),

new java.awt.Color(255, 0, 0),

false,

false,

new datechooser.view.appearance.swing.ButtonPainter()),

(datechooser.view.BackRenderer)null,

false,

true)));

dateChooserCombo1.setNothingAllowed(false);

try {

dateChooserCombo1.setDefaultPeriods(new datechooser.model.multiple.PeriodSet(new datechooser.model.multiple.Period(new java.util.GregorianCalendar(2019, 10, 9),

new java.util.GregorianCalendar(2019, 10, 9))));

} catch (datechooser.model.exeptions.IncompatibleDataExeption e1) {

e1.printStackTrace();

}

jButton1.setText("ELIMINAR");

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

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

jButton1ActionPerformed(evt);

}

});

jButton2.setText("MODIFICAR");

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

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

jButton2ActionPerformed(evt);

}

});

jButton3.setText("INSERTAR");

jButton3.addActionListener(new java.awt.event.ActionListener() {

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

jButton3ActionPerformed(evt);

}

});

jButton4.setText("LIMPIAR");

jButton4.addActionListener(new java.awt.event.ActionListener() {

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

jButton4ActionPerformed(evt);

}

});

jButton5.setText("SALIR");

jButton5.addActionListener(new java.awt.event.ActionListener() {

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

jButton5ActionPerformed(evt);

}

});

jButton6.setText("Consultar empleado");

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

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

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

.addGroup(layout.createSequentialGroup()

.addGap(27, 27, 27)

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

.addComponent(jLabel4)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)

.addGroup(layout.createSequentialGroup()

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

.addComponent(jLabel1)

.addComponent(jLabel7))

.addGap(40, 40, 40)

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

.addComponent(jTextField9, javax.swing.GroupLayout.DEFAULT\_SIZE, 76, Short.MAX\_VALUE)

.addComponent(jTextField1)))

.addGroup(javax.swing.GroupLayout.Alignment.LEADING, layout.createSequentialGroup()

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

.addComponent(jLabel6)

.addComponent(jLabel3)

.addComponent(jLabel2))

.addGap(55, 55, 55)

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

.addComponent(jTextField8)

.addComponent(jTextField6)

.addComponent(jTextField7))))

.addGap(137, 137, 137)

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

.addComponent(jComboBox1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jComboBox2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))))

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

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

.addComponent(dateChooserCombo1, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jButton6, javax.swing.GroupLayout.Alignment.TRAILING))

.addGap(215, 215, 215))

.addGroup(layout.createSequentialGroup()

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

.addGroup(layout.createSequentialGroup()

.addGap(35, 35, 35)

.addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED\_SIZE, 168, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(layout.createSequentialGroup()

.addGap(376, 376, 376)

.addComponent(jButton3)

.addGap(60, 60, 60)

.addComponent(jButton1)

.addGap(34, 34, 34)

.addComponent(jButton2)))

.addContainerGap(84, Short.MAX\_VALUE))

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

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

.addComponent(jButton5)

.addGap(102, 102, 102)

.addComponent(jButton4)

.addGap(171, 171, 171))

);

layout.setVerticalGroup(

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

.addGroup(layout.createSequentialGroup()

.addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED\_SIZE, 53, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(27, 27, 27)

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

.addComponent(jLabel7)

.addComponent(jComboBox1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jButton6))

.addGap(18, 18, 18)

.addComponent(jComboBox2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(2, 2, 2)

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

.addComponent(jLabel1)

.addComponent(jTextField9, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(13, 13, 13)

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

.addGroup(layout.createSequentialGroup()

.addComponent(jLabel4)

.addGap(3, 3, 3)

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

.addComponent(jLabel2)

.addComponent(jTextField8, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addComponent(dateChooserCombo1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(31, 31, 31)

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

.addComponent(jLabel3)

.addComponent(jTextField7, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(35, 35, 35)

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

.addComponent(jLabel6)

.addComponent(jTextField6, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jButton3)

.addComponent(jButton1)

.addComponent(jButton2))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 34, Short.MAX\_VALUE)

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

.addComponent(jButton4)

.addComponent(jButton5))

.addGap(56, 56, 56))

);

pack();

}// </editor-fold>

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

jTextField1.setText("");

jTextField6.setText("");

jTextField7.setText("");

jTextField8.setText("");

jTextField9.setText("");

// TODO add your handling code here:

}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

System.exit(0);

}

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

// TODO add your handling code here:

try {

Class.forName("com.mysql.jdbc.Driver");

Connection conexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/unidad2", "root", "");

String sql = "delete from departamentos where emp\_no=" + jTextField1.getText();

PreparedStatement sentencia = conexion.prepareStatement(sql);

PreparedStatement sentencia2 = conexion.prepareStatement(sql);

ResultSet rs = sentencia.executeQuery();

} catch (ClassNotFoundException cn) {

cn.printStackTrace();

} catch (SQLException e) {

e.printStackTrace();

}

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if (jTextField1.getText() == "" || jTextField6.getText() == "" || jTextField7.getText() == "" || jTextField8.getText() == "" || jTextField9.getText() == "") {

JOptionPane.showMessageDialog(null, "Todos los campos deben estar rellenos");

}

}

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

// TODO add your handling code here:

if (jTextField1.getText() == "" || jTextField6.getText() == "" || jTextField7.getText() == "" || jTextField8.getText() == "" || jTextField9.getText() == "") {

JOptionPane.showMessageDialog(null, "Todos los campos deben estar rellenos");

} else {

try {

Class.forName("com.mysql.jdbc.Driver");

Connection conexion = DriverManager.getConnection("jdbc:mysql://localhost:3306/unidad2", "root", "");

String sql = "update empleados set apellido = " + jTextField9.getText() + " , oficio=" + jTextField8.getText() + " ,salario=" + jTextField7.getText() + ", comision=" + jTextField9.getText() + " where emp\_no=" + jTextField1.getText();

PreparedStatement sentencia = conexion.prepareStatement(sql);

PreparedStatement sentencia2 = conexion.prepareStatement(sql);

ResultSet rs = sentencia.executeQuery();

} catch (ClassNotFoundException cn) {

cn.printStackTrace();

} catch (SQLException e) {

e.printStackTrace();

}

}

}

/\*\*

\* @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(NewJFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

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

} catch (IllegalAccessException ex) {

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

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

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

}

//</editor-fold>

/\* Create and display the form \*/

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

public void run() {

new NewJFrame().setVisible(true);

}

});

}

// Variables declaration - do not modify

private datechooser.beans.DateChooserCombo dateChooserCombo1;

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JButton jButton3;

private javax.swing.JButton jButton4;

private javax.swing.JButton jButton5;

private javax.swing.JButton jButton6;

private javax.swing.JComboBox<String> jComboBox1;

private javax.swing.JComboBox<String> jComboBox2;

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 jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField6;

private javax.swing.JTextField jTextField7;

private javax.swing.JTextField jTextField8;

private javax.swing.JTextField jTextField9;

// End of variables declaration

}

//