

---

# **BASE DE DATOS EN J2EE**

**Eduard Lara**

# INDICE

---

1. Acceso a Access
2. Acceso a Mysql
3. Acceso a Oracle
4. Acceso a PostgreSQL
5. Practicas JSP con base de datos

# 1. ACCESO A ACCESS

## Paso 1. Acceso a un fichero access.mdb

The screenshot displays the Microsoft Access 2010 interface. The title bar reads 'Equipos - Microsoft Access (Error de activación de productos)'. The ribbon is set to 'Herramientas de tabla' with the 'Campos' group selected. The 'Equipos' table is open, showing 18 rows of data. To the right, a Windows Explorer window is open to 'Equipo > Disco local (C:) >', showing a list of files and folders including 'AdwCleaner', 'app', 'Archivos de programa', 'Archivos de programa (x86)', 'Brother', 'IDE', 'Intel', 'PerfLogs', 'TEMP', 'Usuarios', 'Windows', 'xampp', 'cine.accdb', and 'futbol.mdb'.

Codi	Equipo	Ciudad	Estadio	NomQuinLFP	NomCalenONLAE	NomQuinONLAE
1	Alaves	Vitoria	Mendizorroza	Alaves	Alaves	Alaves
2	Athletic de Bilbao	Bilbao	San Mames	Athletic	Ath.Bilbao	Ath.Bilbao
3	Barcelona	Barcelona	Camp Nou	Barcelona	Barcelona	Barcelona
4	Celta de Vigo	Vigo	Balaidos	Celta	R.Celta	R.Celta
5	Deportivo de La Coruna	La Corunya	Riazor	Deportivo	Dptvo.Corunya	Dptvo.Corunya
6	Espanyol	Barcelona	Olimpico de Montjuic	Espanyol	Espanyol	Espanyol
7	Las Palmas	Las Palmas de G.C.	Insular	Las Palmas	Las Palmas	Las Palmas
8	Malaga	Malaga	La Rosaleda	Malaga	Malaga	Malaga
9	Mallorca	Palma de Mallorca	Son Moix	Mallorca	R.Mallorca	R.Mallorca
10	Numancia	Soria	Los Pajaritos	Numancia	Numancia	Numancia
11	Osasuna	Pamplona	El Sadar	Osasuna	At.Osasuna	At.Osasuna
12	Real Oviedo	Oviedo	Carlos Tartiere	Oviedo	R.Oviedo	R.Oviedo
13	Racing de Santander	Santander	El Sardinero	Racing	Rac.Santander	Racing Santander
14	Rayo Vallecano	Madrid	Teresa Rivero	Rayo	Rayo Vallecano	Rayo Vallecano
15	Real Madrid	Madrid	Santiago Bernabeu	R. Madrid	R.Madrid	R.Madrid
16	Real Sociedad	San Sebastian	Anoeta	R. Sociedad	R.Sociedad	R.Sociedad
17	Valencia	Valencia	Mestalla	Valencia	Valencia	Valencia
18	Valladolid	Valladolid	Jose Zorrilla	Valladolid	R.Valladolid	R.Valladolid

# 1. ACCESO A ACCESS

**Paso 2.** El driver para acceder a los datos del fichero access mdb a través de Java sólo funciona para el JDK 1.6 y 1.7.

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <%
      String url = "jdbc:odbc;;DRIVER=Microsoft Access Driver (*.mdb, *.accdb);DBQ=futbol.mdb; PWD= ";
      Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
      Connection cnn = DriverManager.getConnection(url);
      Statement stm = cnn.createStatement();
      String sql = "SLEECT * FROM EQUIPOS";
      ResultSet res = stm.executeQuery(sql);
      out.print("</table>");
      while (res.next())
        out.print("<tr><td>" + res.getObject(1) + "<td>" + res.getObject(2) + "<td>" + res.getObject(3));

      out.print("</table>");
    %>
  </body>
</html>
```

# 1. ACCESO A ACCESS

**Paso 3.** Para el JDK 8, hay una solución proporcionada por un tercero, que es el driver Ucan Access: <http://ucanaccess.sourceforge.net/site.html>

The screenshot shows the UCanAccess website. At the top, there is a navigation bar with links: Home, Getting Started, JDBC Client Tools, Change Log, Links/Licensing, and About. Below the navigation bar, there is a green button labeled "Download UCanAccess 4.0.2" and a red mouse cursor icon with the text "U Can Donate Click here".

The main content area is titled "UCanAccess" and describes it as an open-source Java JDBC driver implementation for Microsoft Access databases. It mentions that it is a pure Java implementation and runs on both Windows and non-Windows operating systems.

Below the description, there is a list of features:

- Jackcess as MS Access input/output library (Web site: <http://jackcess.sourceforge.net>)
- HSQLDB as synchronized DBMS (Web site: <http://hsqldb.org/>).

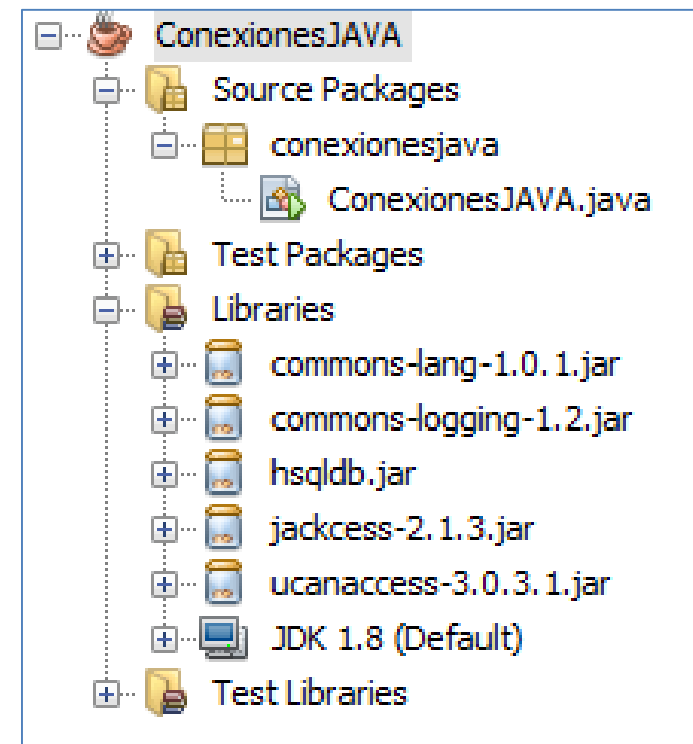
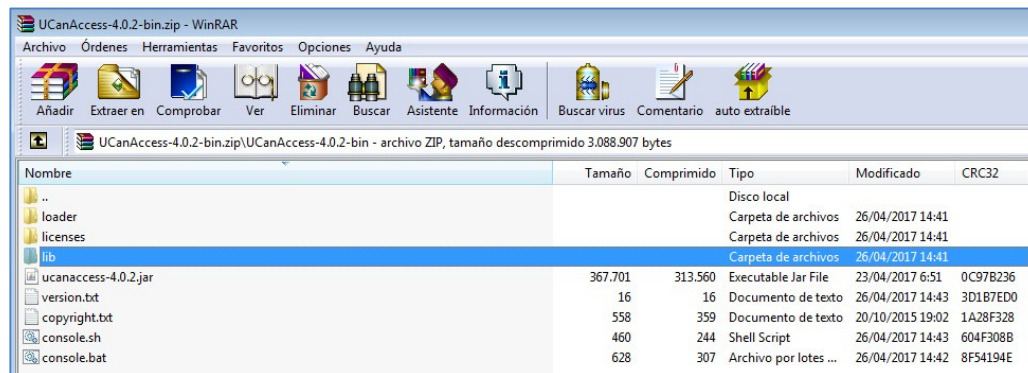
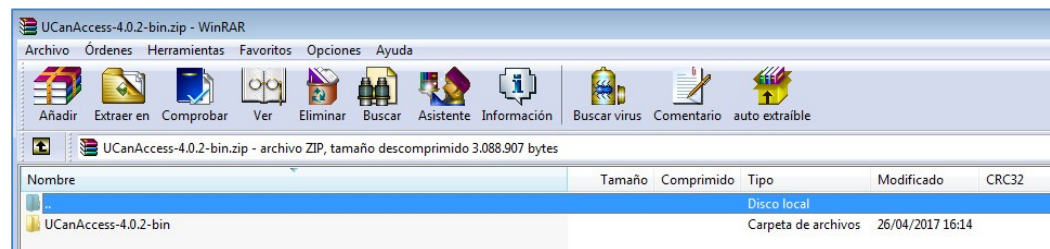
Below the features, there is a list of supported Access formats: 2000, 2002/2003, 2007, 2010/2013/2016 databases. (Access 2000, 2002, 2003, 2007, 2010, 2013, 2016). It also lists supported SQL statements: SELECT, INSERT, UPDATE, DELETE, and transactions and savepoints. Other features include Access data types: YESNO, BYTE, INTEGER, LONG, SINGLE, DOUBLE, NUMERIC, CURRENCY, and concurrent access from multiple application server threads.

Below the features, there is a table showing the latest version of the driver:

Name	Modified	Size	Downloads / Week
UCanAccess-4.0.2-bin.zip	2017-04-26	2.7 MB	713
UCanAccess-4.0.2-src.zip	2017-04-26	699.2 kB	28
README.txt	2017-04-26	29.2 kB	22

# 1. ACCESO A ACCESS

**Paso 4.** Descargamos el driver UcanAccess y añadimos las 5 librerías jar existentes a nuestro proyecto netbeans.



# 1. ACCESO A ACCESS

**Paso 5.** Este sería el código Java para acceder a las tablas de un fichero Access mdb o accdb:

```
public static void main(String[] args) {
    try {
        Class.forName("net.ucanaccess.jdbc.UcanaccessDriver");
        String url = "jdbc:ucanaccess://C:/Users/eduardo/Documents/NetBeansProjects/ConexionesJAVA/Northwind.mdb";
        Connection conn=DriverManager.getConnection(url);
        DatabaseMetaData pepe=conn.getMetaData();

        Statement s = conn.createStatement();
        String selTable = "SELECT Customers.CustomerID, Customers.CompanyName, Customers.ContactName, "
            + "Customers.ContactTitle FROM Customers";

        s.execute(selTable);
        ResultSet rs = s.getResultSet();
        while((rs!=null) && (rs.next()))
        {
            System.out.println(rs.getString(1) + " : " + rs.getString(2) + " : " + rs.getString(3) + " : " + rs.getString(4));
        }

        s.close();
        conn.close();
    }
    catch(Exception ex)
    {
        ex.printStackTrace();
    }
}
```

## 2. ACCESO A MYSQL

**Paso 1.** El driver de acceso a mysql desde Java se encuentra en la url:  
<http://dev.mysql.com/downloads/connector/j/5.1.html>

Please report any bugs or inconsistencies you observe to our [Bugs Database](#).  
Thank you for your support!


**Generally Available (GA) Releases** **Development Releases**

### Connector/J 5.1.43

Select Operating System:

Platform Independent ▼

<b>Platform Independent (Architecture Independent), Compressed TAR Archive</b> (mysql-connector-java-5.1.43.tar.gz)	5.1.43
<b>Platform Independent (Architecture Independent), ZIP Archive</b> (mysql-connector-java-5.1.43.zip)	5.1.43

 We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the packages you



## 2. ACCESO A MYSQL

**Paso 2.** Descargamos la versión zip del driver (Plataforma Independiente):

### Begin Your Download

mysql-connector-java-5.1.43.tar.gz

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system
- Comment in the MySQL Documentation

Login »

using my Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle account, you can sign up for a free account by clicking the Sign Up link and following the instructions.

[No thanks, just start my download.](#)

Abriendo mysql-connector-java-5.1.43.tar.gz

Ha elegido abrir:



mysql-connector-java-5.1.43.tar.gz

que es: Archivo WinRAR (3,3 MB)

de: <https://cdn.mysql.com>

¿Qué debería hacer Firefox con este archivo?



Abrir con

WinRAR archiver (predeterminada)



Guardar archivo



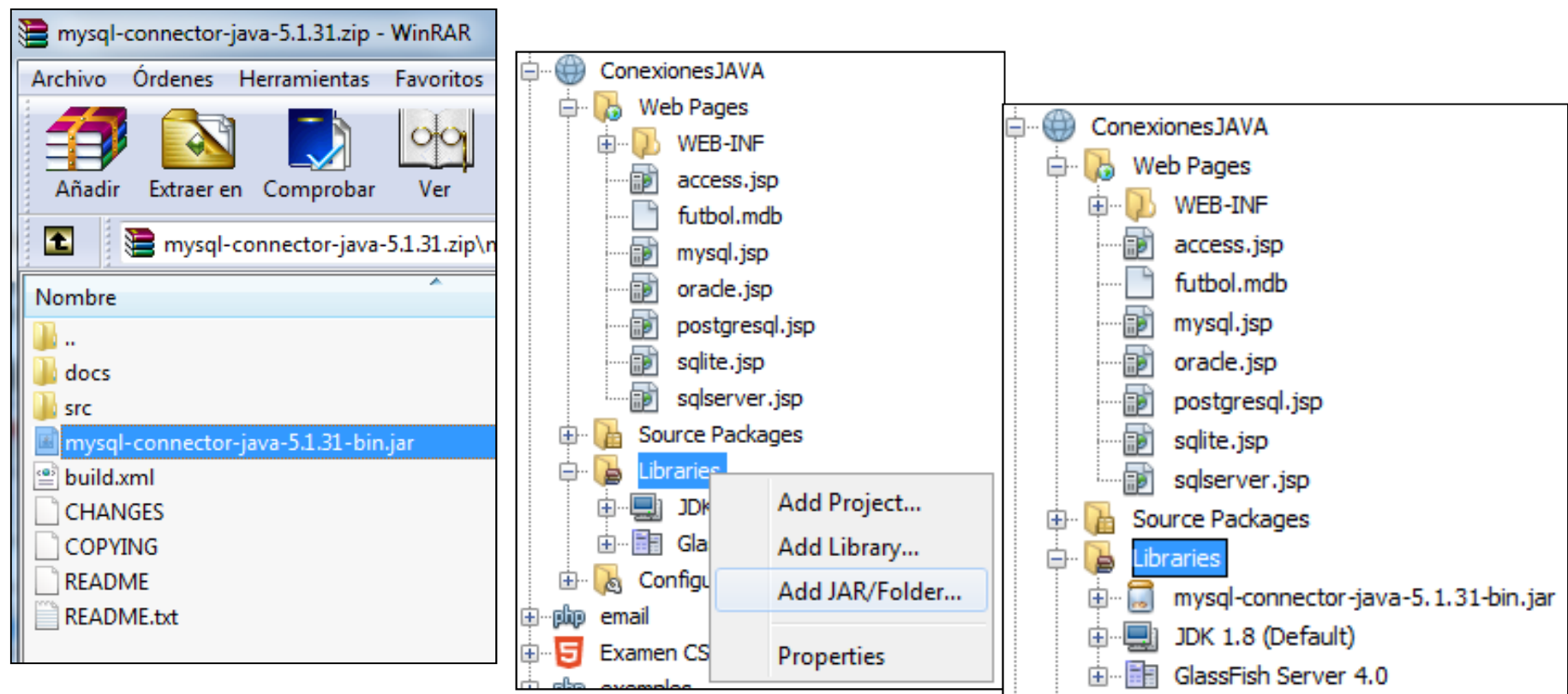
Hacer esto automáticamente para estos archivos a partir de ahora.

Aceptar

Cancelar

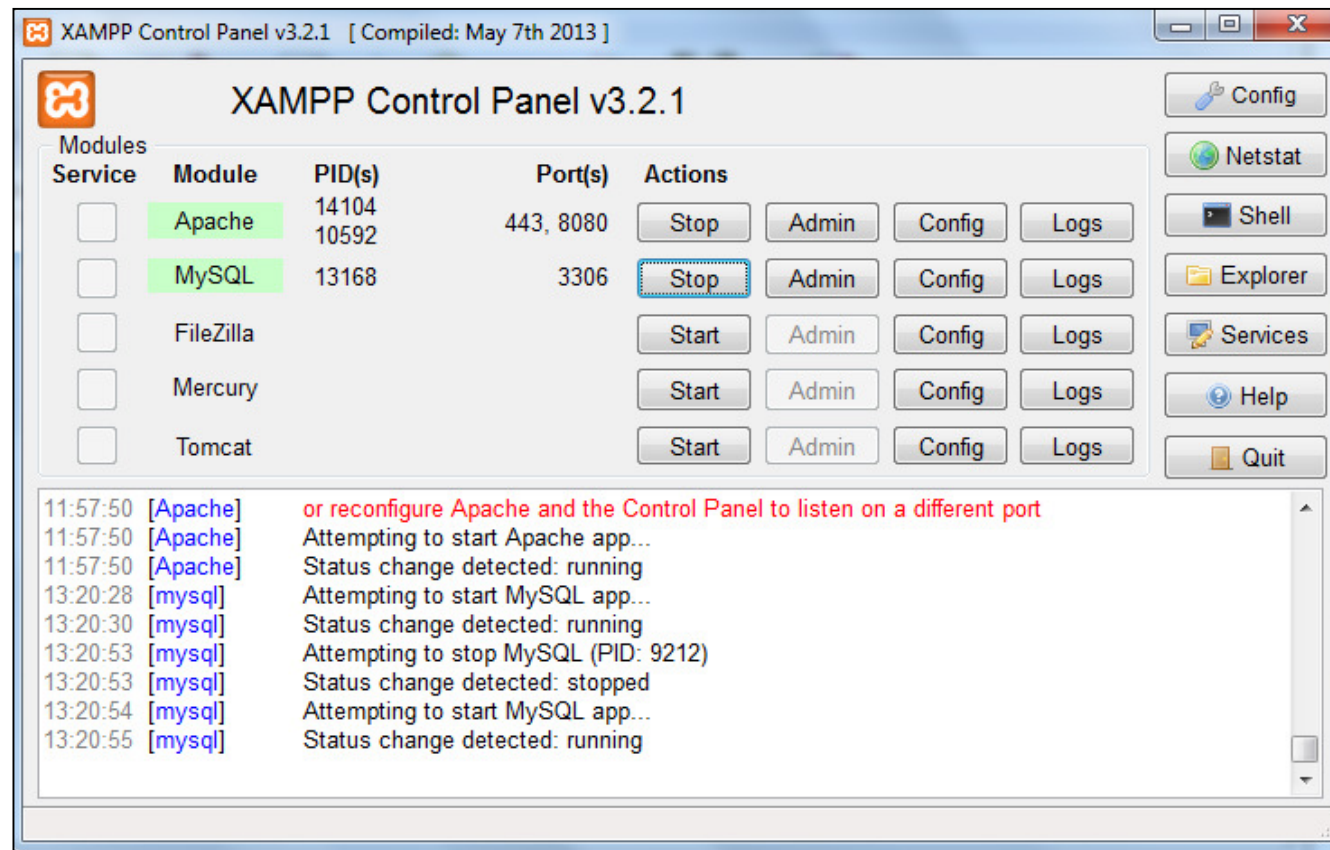
## 2. ACCESO A MYSQL

**Paso 3.** Descomprimos el fichero zip, extraemos el driver jar, y lo agregamos a nuestro proyecto, haciendo click botón derecho sobre Libraries en nuestro proyecto Netbeans.



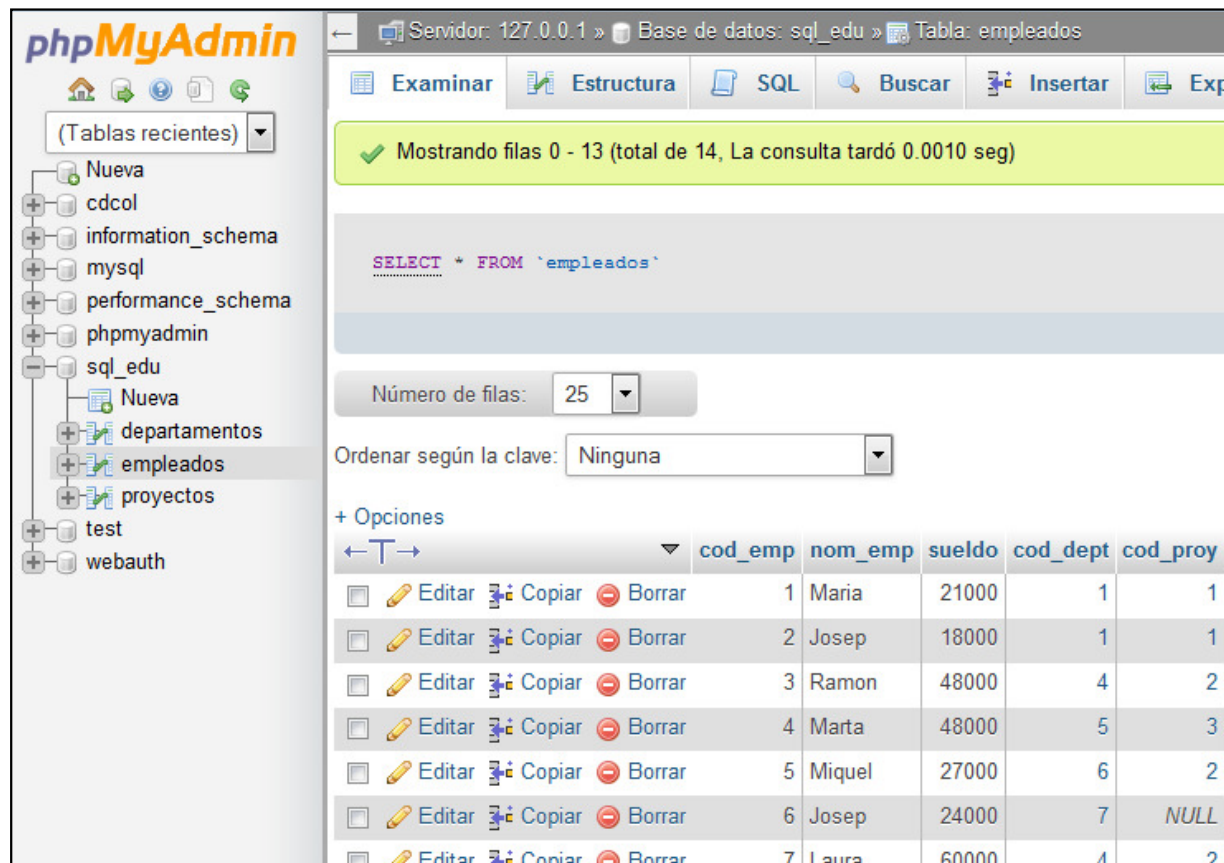
## 2. ACCESO A MYSQL

**Paso 4.** En el caso de la base de datos mysql, debemos activar previamente el servicio desde el control panel de xampp. Se levanta en el puerto 3306.



## 2. ACCESO A MYSQL

**Paso 5.** Desde Phpmyadmin vemos cual es la base de datos (sql\_edu) y la tabla que atacaremos (por ejemplo empleados):

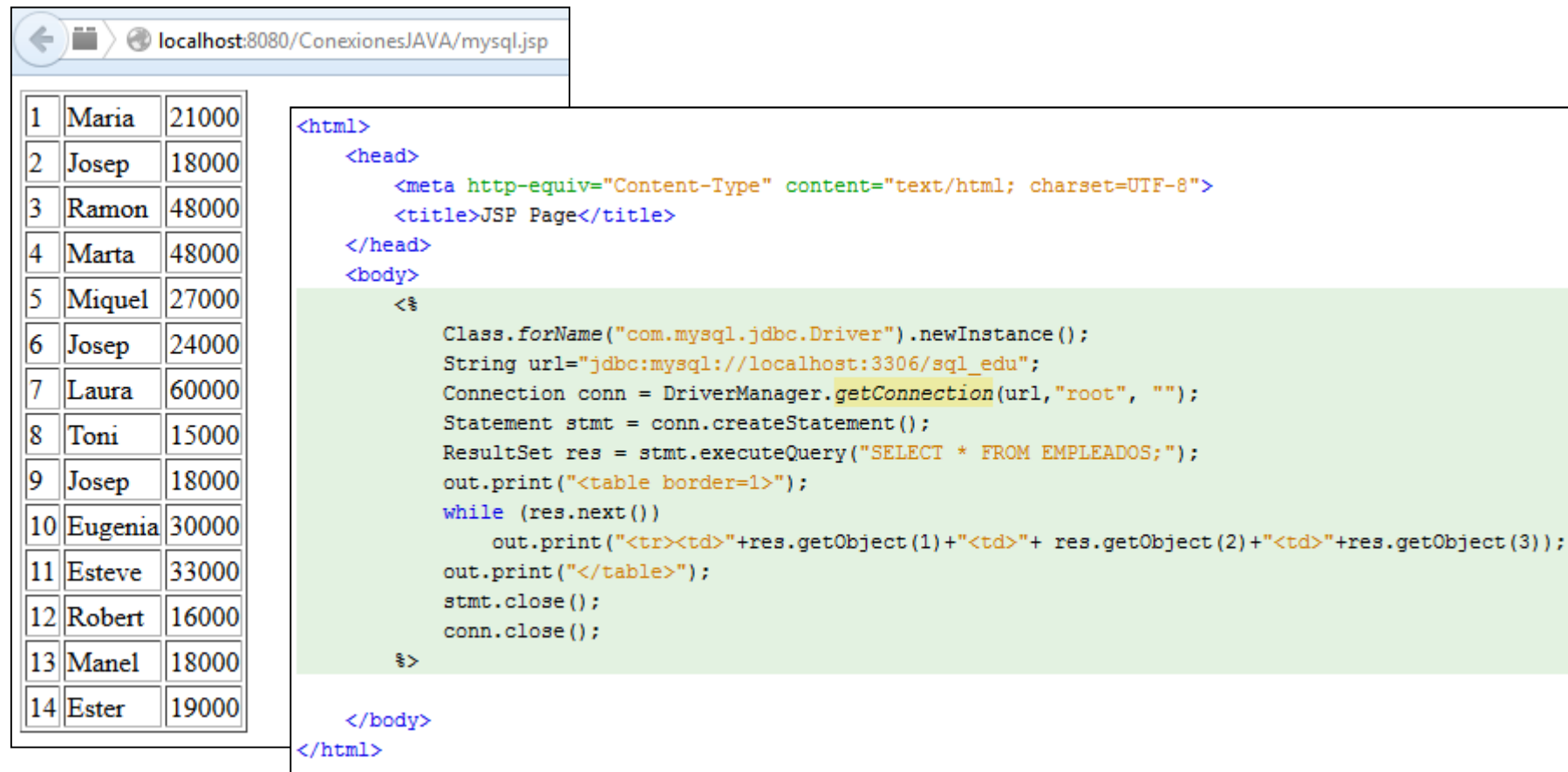


The screenshot shows the phpMyAdmin interface. On the left, the database structure tree is visible, with 'sql\_edu' expanded to show tables: 'departamentos', 'empleados', and 'proyectos'. The 'empleados' table is selected. The main panel shows the 'Examinar' (Browse) tab. A green status bar indicates 'Mostrando filas 0 - 13 (total de 14, La consulta tardó 0.0010 seg)'. Below this, the SQL query 'SELECT \* FROM `empleados`' is entered. The 'Número de filas' (Number of rows) is set to 25, and 'Ordenar según la clave' (Order by key) is set to 'Ninguna' (None). The '+ Opciones' (Options) section is expanded, showing a table with 5 columns: 'cod\_emp', 'nom\_emp', 'sueldo', 'cod\_dept', and 'cod\_proy'. The table contains 7 rows of data.

	cod_emp	nom_emp	sueldo	cod_dept	cod_proy
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	Maria	21000	1	1
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	2	Josep	18000	1	1
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	3	Ramon	48000	4	2
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	4	Marta	48000	5	3
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	5	Miquel	27000	6	2
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	6	Josep	24000	7	NULL
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	7	Laura	60000	4	2

## 2. ACCESO A MYSQL

**Paso 6.** El código de acceso a mysql desde Java es el siguiente:



The screenshot displays a web browser window with the address bar showing `localhost:8080/ConexionesJAVA/mysql.jsp`. The browser content is divided into two main sections. On the left, there is a table with 14 rows of employee data. On the right, there is a code editor showing the JSP code that generates this table.

1	Maria	21000
2	Josep	18000
3	Ramon	48000
4	Marta	48000
5	Miquel	27000
6	Josep	24000
7	Laura	60000
8	Toni	15000
9	Josep	18000
10	Eugenia	30000
11	Esteve	33000
12	Robert	16000
13	Manel	18000
14	Ester	19000








```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <%
      Class.forName("com.mysql.jdbc.Driver").newInstance();
      String url="jdbc:mysql://localhost:3306/sql_edu";
      Connection conn = DriverManager.getConnection(url,"root", "");
      Statement stmt = conn.createStatement();
      ResultSet res = stmt.executeQuery("SELECT * FROM EMPLEADOS;");
      out.print("<table border=1>");
      while (res.next())
        out.print("<tr><td>"+res.getObject(1)+"<td>"+ res.getObject(2)+"<td>"+res.getObject(3));
      out.print("</table>");
      stmt.close();
      conn.close();
    %>

  </body>
</html>
```

### 3. ACCESO A ORACLE

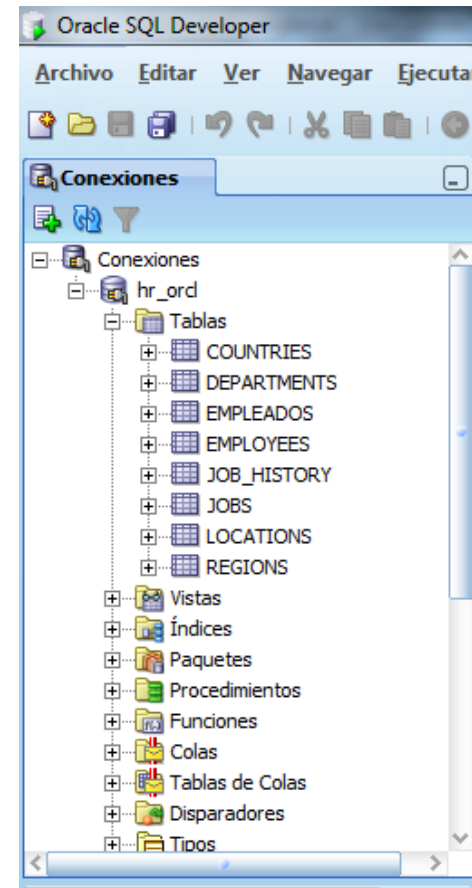
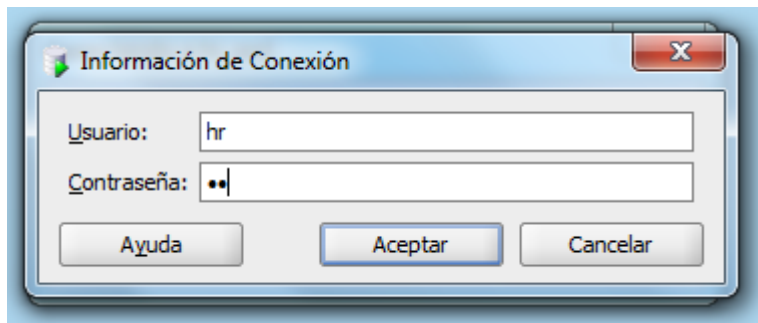
---

**Paso 1.** Para acceder a la base de datos de Oracle, primero tenemos que comprobar que está iniciada. A continuación se muestran los servicios mínimos de Oracle que tendrían que estar iniciados en services.msc

 Oracle ORCL VSS Writer Service		Manual	Sistema local
 OracleDBConsoleorcl	Iniciado	Automático	Sistema local
 OracleJobSchedulerORCL		Deshabilitado	Sistema local
 OracleMTSRecoveryService		Automático	Sistema local
 OracleOraDb11g_home1ClrAgent		Manual	Sistema local
 OracleOraDb11g_home1TNSListener	Iniciado	Automático	Sistema local
 OracleServiceORCL	Iniciado	Automático	Sistema local

### 3. ACCESO A ORACLE

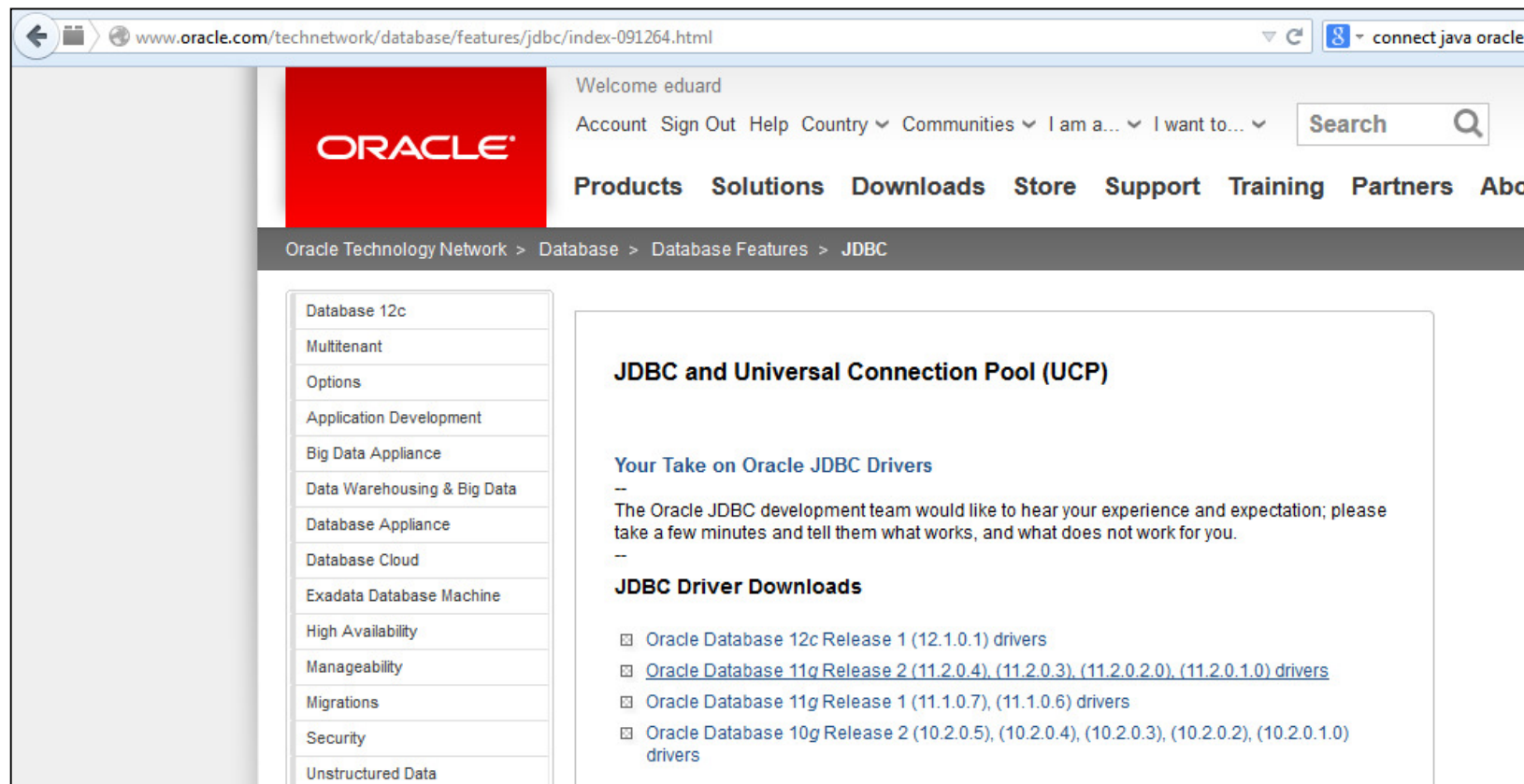
**Paso 2.** El segundo punto importante para estar seguros del buen funcionamiento de la base de datos Oracle es que nos pudiéramos conectar a ella a través del SQL Developer





# 3. ACCESO A ORACLE

**Paso 3.** Para acceder desde Java a una instancia de la base de datos Oracle, debemos descargar el Oracle JDBC driver version 11gR2





# 3. ACCESO A ORACLE

**Paso 4.** Descargamos la versión para usar con JDK 1.6:

**Oracle Database 11g Release 2 JDBC Drivers**

You must accept the [OTN License Agreement](#) to download this software.  
☒ Accept License Agreement | ☐ Decline License Agreement

**Oracle Database 11g Release 2 (11.2.0.4) JDBC Drivers**

**SimpleFAN**  
simplefan.jar (20,365 bytes)

**JDBC Thin for All Platforms**  
JavaDoc (6,415,512 bytes)  
README

ojdbc5.jar (2,091,135 bytes) - Classes for use with JDK 1.5. It contains the JDBC driver classes, except classes for NLS support in Oracle Object and Collection types.

ojdbc5\_g.jar (3,424,145 bytes) - Same as ojdbc5.jar, except that classes were compiled with "javac -g" and contain tracing code.

**ojdbc6.jar (2,739,616 bytes) - Classes for use with JDK 1.6. It contains the JDBC driver classes except classes for NLS support in Oracle Object and Collection types.**

ojdbc6\_g.jar (4,494,540 bytes) - Same as ojdbc6.jar except compiled with "javac -g" and contains tracing code.

ojdbc5dms.jar (2,609,739 bytes) - Same as ojdbc5.jar, except that it contains instrumentation to support DMS and limited java.util.logging calls.

ojdbc5dms\_g.jar (3,445,412 bytes) - Same as ojdbc5\_g.jar, except that it contains instrumentation to support DMS.

ojdbc6dms.jar (3,350,717 bytes) - Same as ojdbc6.jar, except that it contains instrumentation to support DMS and limited java.util.logging calls.

ojdbc6dms\_g.jar (4,517,780 bytes) - Same as ojdbc6\_g.jar except that it contains instrumentation to support DMS.

orai18n.jar (1,655,734 bytes) - NLS classes for use with JDK 1.5, and 1.6. It contains classes for NLS support in Oracle Object and Collection types. This jar file replaces the old nls\_charset jar/zip files.

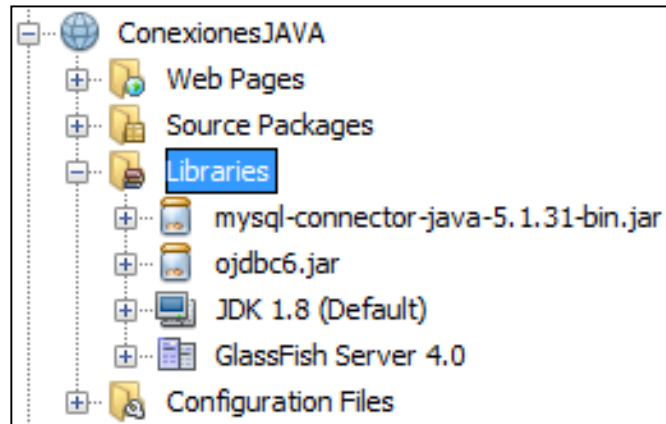
demo.zip (606,650 bytes) - contains sample JDBC programs.

ojdbc6.jar

### 3. ACCESO A ORACLE

---

**Paso 5.** Agregamos el driver a nuestro proyecto:



### 3. ACCESO A ORACLE

#### Paso 6. Código acceso a oracle desde Java:



The screenshot displays a web browser window with the address bar showing `localhost:8080/ConexionesJAVA/oracle.jsp`. The browser content is divided into two parts: a table of employee data on the left and a code editor on the right.






198	Donald	OConnell
199	Douglas	Grant
200	Jennifer	Whalen
201	Michael	Hartstein
202	Pat	Fay
203	Susan	Mavris
204	Hermann	Baer
205	Shelley	Higgins
206	William	Gietz
100	Steven	King
101	Neena	Kochhar
102	Lex	De Haan
103	Alexander	Hunold
104	Bruce	Ernst
105	David	Austin
106	Valli	Pataballa

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>JSP Page</title>
</head>
<body>
  <%
    Class.forName("oracle.jdbc.driver.OracleDriver");
    String url = "jdbc:oracle:thin:@//127.0.0.1:1521/orcl";
    Connection conn = DriverManager.getConnection(url,"hr","hr");
    Statement stmt = conn.createStatement();
    ResultSet res = stmt.executeQuery("SELECT * FROM EMPLOYEES");
    out.print("<table border=1>");
    while (res.next())
      out.print("<tr><td>"+res.getObject(1)+"<td>"+ res.getObject(2)+"<td>"+res.getObject(3));
    out.print("</table>");
    stmt.close();
    conn.close();
  %>
</body>
</html>
```

## 4. ACCESO A POSTGRESQL

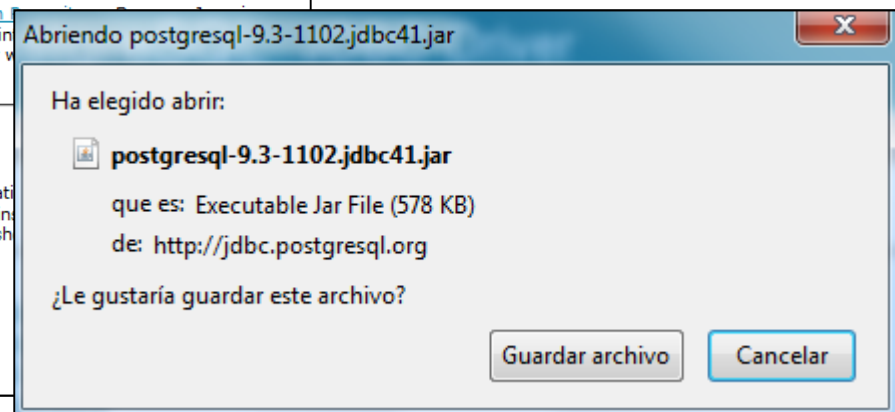
---

**Paso 1.** El asistente para instalar el servidor PostgreSQL habrá creado un servicio que estará iniciado y en tipo de inicio automático en services.msc:

 Parental Controls	Este servicio es un...		Manual
 Plug and Play	Habilita un equip...	Iniciado	Automático
 postgresql-9.3 - PostgreSQL Server 9.3	Proporciona alma...	Iniciado	Automático
 Programador de aplicaciones multimedia	Permite establece...	Iniciado	Automático
 Programador de tareas	Permite a un usua...	Iniciado	Automático

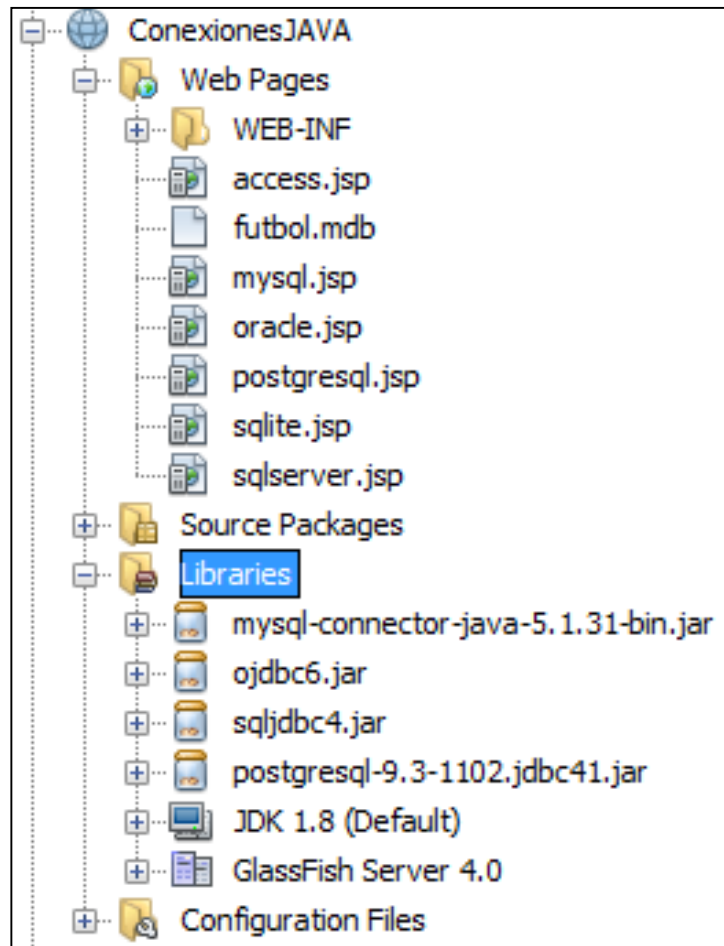
## 4. ACCESO A POSTGRESQL

**Paso 2.** Vamos a la url <http://jdbc.postgresql.org/download.html> para obtener el driver JDBC de PostgreSQL:



## 4. ACCESO A POSTGRESQL

**Paso 3.** Agregamos el driver JDBC para PostgreSql:



## 4. ACCESO A POSTGRESQL

**Paso 4.** El código de acceso a PostgreSql desde Java es el siguiente:



The screenshot shows a web browser window with the address bar displaying `localhost:8080/ConexionesJAVA/postgresql.jsp`. The browser content is split into two panes. The left pane shows the output of the JSP page, which is a table with 15 rows of employee data. The right pane shows the source code of the JSP page, which is a Java scriptlet that connects to a PostgreSQL database and retrieves the employee data.

110	121	PONS
120	112	LASA
130	112	TEROL
150	121	PEREZ
160	111	AGUIRRE
180	110	PEREZ
190	121	VEIGA
210	100	GALVEZ
240	111	SANZ
250	100	ALBA
260	100	LOPEZ
270	112	GARCIA
280	130	FLOR
285	122	POLO
290	120	GIL

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <%
      Class.forName("org.postgresql.Driver");
      String url = "jdbc:postgresql://localhost:5432/bdroot";
      Connection conn = DriverManager.getConnection(url,"postgres", "cifo");
      Statement stmt = conn.createStatement();
      String queryString = "select * from empleados";
      ResultSet res = stmt.executeQuery(queryString);

      out.print("<table border=1>");
      while (res.next())
        out.print("<tr><td>" + res.getObject(1) + "<td>" + res.getObject(2) + "<td>" + res.getObject(3) );
      out.print("</table>");
      stmt.close();
      conn.close();
    %>
  </body>
</html>
```

## 5. PRACTICAS JSP CON BBDD

### 1. Recogida de parámetros e inserción en BBDD

---

Recogemos los parámetros de un formulario y los insertamos en la base de datos previo a su visualización

```
<body>
<%
    Connection con = null;
    Statement state = null;
    String nombre = request.getParameter("nombre");
    String apellido = request.getParameter("apellido");
    String edad = request.getParameter("edad");

    String DRIVER = "com.mysql.jdbc.Driver";
    Class.forName(DRIVER).newInstance();
    String url="jdbc:mysql://localhost:3306/prueba";
    con = DriverManager.getConnection(url, "root", "");

    state = con.createStatement();
    String s = "insert into usuarios values ('" + nombre+"', '" + apellido
    + "', '" + edad + "');";
    state.execute(s);

    out.println("Name: <b>" + nombre + "</b><br>");
    out.println("Apellido: <b>" + apellido + "</b><br>");
    out.println("Tus datos han sido enviados correctamente");

    state.close();
    con.close();
%>
<a href=http://localhost:8080/ConexionesJAVA/tabla.jsp>Ver tabla</a>
</body>
```



# 5. PRACTICAS JSP CON BBDD

## 2. Listado de valores

---

Mostramos en una tabla los valores anteriormente insertados:

```
<body>
<%
    Connection con = null;
    Statement state = null;
    ResultSet rs = null;

    Class.forName("com.mysql.jdbc.Driver").newInstance();
    String url="jdbc:mysql://localhost:3306/prueba";
    con = DriverManager.getConnection(url, "root", "");
    state = con.createStatement();
    String s = "Select * from usuarios";
    rs = state.executeQuery(s);
    out.println("<TABLE border=1>");
    out.println("<TR><TD>NOMBRE<TD>APELLIDO<TD>EDAD");

    while(rs.next())
        out.println("<TR><TD>" + rs.getString(1) + "<TD>" + rs.getString(2) + "<TD>" + rs.getString(3));

    out.println("</TABLE>");
    rs.close();
    state.close();
    con.close();
%>
</body>
```

# 5. PRACTICAS JSP CON BBDD

## 3. Mantenimiento completo

Juntamos las diferentes partes anteriores para mostrar un mantenimiento de la siguiente forma:

mantenimiento.jsp

Insert title here

http://localhost:8080/nuevo/mantenimiento.jsp

NOMBRE	APELLIDO	EDAD	
eduardo	benavides	18	<a href="#">Borrar entrada</a>
jose	buenas	23	<a href="#">Borrar entrada</a>
yolanda	daigual	26	<a href="#">Borrar entrada</a>
federico	bahamontes	30	<a href="#">Borrar entrada</a>
rajoy	aznar	44	<a href="#">Borrar entrada</a>
edu	navidad	1	<a href="#">Borrar entrada</a>

Nombre:  Apellido:  Edad:

# 5. PRACTICAS JSP CON BBDD

## 3. Mantenimiento completo

---

Juntamos las diferentes partes anteriores para mostrar un mantenimiento de la siguiente forma:

```
<% Connection con = null;
Statement state = null;
ResultSet rs = null;
String nombre=request.getParameter("nombre");
String apellido=request.getParameter("apellido");
String edad = request.getParameter("edad");
String accion=request.getParameter("accion");

try
{
    String DRIVER = "com.mysql.jdbc.Driver";
    Class.forName(DRIVER).newInstance();
    String url="jdbc:mysql://localhost:3306/prueba";
    con = DriverManager.getConnection(url, "root", "");
    state = con.createStatement();
```

Recogida de datos y  
apertura conexión a la  
base de datos

# 5. PRACTICAS JSP CON BBDD

## 3. Mantenimiento completo

Procesado del evento de borrado o inserción y visualización de los datos restantes:

```
if (accion!=null)
{
    if (accion.equals("borrado"))
    {
        String s1 = "delete from usuarios where nombre='"+nombre + "'";
        state.execute(s1);
    }
    if (accion.equals("insertar"))
    {
        String s = "insert into usuarios values ('"+nombre+"', '"+apellido+"', "+edad +")";
        state.execute(s);
    }
}
String p = "Select * from usuarios";
rs = state.executeQuery(p);
out.println("<TABLE border=1>");
out.println("<TR><TD>NOMBRE<TD>APELLIDO<TD>EDAD");

while(rs.next())
{
    out.println("<TR><TD>" + rs.getString(1) + "<TD>" + rs.getString(2) + "<TD>" + rs.getString(3) +
        "<TD><a href=http://localhost:8080/nuevo/mantenimiento.jsp?accion=borrado&nombre=" +
        rs.getString(1) + ">Borrar entrada</a>");
}
out.println("</TABLE>");
con.close();
```

## 5. PRACTICAS JSP CON BBDD

### 3. Mantenimiento completo

---

En la parte final poner el formulario de inserción de datos nuevos:

```
<form action=http://localhost:8080/nuevo/mantenimiento.jsp method=get>
  Nombre:<input type=text name=nombre>
  Apellido:<input type=text name=apellido>
  Edad:<input type=text name=edad>
  <input type=hidden name=accion value=insertar><br>
  <input type=submit>
  <input type=reset>
</form>
```