

PRÁCTICA 9 – VISUALIZADOR DE LA ESTRUCTURA DE UNA BASE DE DATOS MYSQL

En algunas aplicaciones es necesario que el usuario introduzca información que no debe ser vista por otras personas como son las claves de acceso. En este caso es importante que cada carácter que teclee el usuario se convierta en otro, normalmente un asterisco. Swing proporciona la clase `JPasswordField` que realiza esta tarea de forma automática. La apariencia es la misma que un campo de texto pero los caracteres se sustituyen por un carácter (se puede configurar) a medida que se escribe.

También suele ser necesario en algunas aplicaciones, seleccionar de una lista que se le presenta al usuario, una o más opciones. En este caso la clase `JList` de Swing es la que implementa esta funcionalidad. Destacar que a diferencia de otras clases que dispone Swing para que los usuarios puedan seleccionar de entre una lista de opciones, la clase `JList` permite crear, mostrar y modificar el esquema de selección de opciones de forma dinámica. En el caso de la clase `JComboBox` o los `JCheckBox`, las posibles opciones deben ser conocidas en el momento de realizar la codificación del programa.

El objetivo de esta práctica es hacer uso de las clases de Swing anteriores, así como otras estudiadas en el curso, para implementar una aplicación que nos permita mostrar la estructura de tablas y campos de una base de datos MySQL. Para acceso a la base de datos se utilizará el driver correspondiente, así como las clases que incorpora el paquete `java.sql`. Como en prácticas anteriores en el apartado “Notas de implementación” se darán ejemplos de uso de estas clases.

La funcionalidad que deberá tener la aplicación a desarrollar en esta práctica será la siguiente:

- Disponer de un cuadro de texto y otro de clave donde el usuario introduzca el usuario y clave que le permita acceder a la base de datos.
- Realizar la conexión con la base de datos utilizando el usuario y clave introducidos así como de informar de cualquier error que se pueda producir.
- Mostrar en una lista las tablas que componen la base de datos.
- Permitir mediante el uso de *toggle buttons* que el usuario decida qué modo de selección (simple, por intervalo o por múltiples intervalos) utilizar para elegir las tablas.
- Permitir al usuario deseleccionar toda la selección.
- Mostrar en una lista los campos de las tablas seleccionadas, anteponiendo el nombre de la tabla separada por un punto, es decir, tabla.campo.

Notas de implementación

El acceso de una aplicación Java a una base de datos se realiza mediante la API JDBC. Esta API hace uso de un driver específico para cada base de datos (Oracle, MySQL, PostgreSQL, ...) que proporciona la conexión con dicha base de datos para realizar las operaciones habituales sobre la misma.

A continuación se da un ejemplo de código de acceso a una base de datos MySQL y de la recuperación de las tablas y campos que compone cada tabla.

```
...

import java.sql.*;
import java.util.logging.*;

...

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

Connection con = DriverManager.getConnection("jdbc:mysql://SERVIDOR/BASE_DE_DATOS?useSSL=true",
        "USUARIO",
        "CLAVE");

DatabaseMetaData md = con.getMetaData();
String[] types = {"TABLE"};
ResultSet rs = md.getTables(null, null, "%", types);

while (rs.next()) {
    String nombreTabla = rs.getString("TABLE_NAME");
    System.out.println("Tabla: " + nombreTabla);

    ResultSet rs2 = md.getColumns(null, null, nombreTabla, null);
    while (rs2.next()) {
        String nombreCampo = rs2.getString("COLUMN_NAME");
        System.out.println("    Campo: " + nombreCampo);
    }
}

con.close();

...
```

Para esta práctica los datos de la conexión son los siguientes. La conexión solo se puede realizar desde un ordenador conectado a la red de la ULPGC.

- Servidor: **mozart.dis.ulpgc.es**
- Base de datos: **DIU_BD**
- Usuario: **estudiante-DIU**
- Clave: **DIU-aed56-noi**

Documentación a entregar

Cada grupo entregará a través del campus virtual de la asignatura un fichero comprimido que contendrá lo siguiente:

- El proyecto Netbeans que implementa la aplicación desarrollada.
- La memoria en formato pdf en español o inglés que constará de:
 - Una portada indicando el nombre de la asignatura, curso y nombre de los componentes del grupo.
 - Una descripción de las actividades que el alumno ha realizado, así como captura de pantalla de la aplicación que deberá incluir una etiqueta con el nombre de los componentes del grupo.
 - La memoria deberá estar correctamente justificada y sin faltas de ortografía.
 - La memoria debe incluir una autoevaluación de la interfaz desarrollada de acuerdo a los principios de Schneiderman y Plaisant (ver Tema 1).