

EJERCICIO RESUELTO

Módulo: Programación

Gestión BD de Contactos

Descripción:

En la presente unidad hemos visto como conectarnos con distintos SGBD como puede ser Oracle y como realizar sobre el operaciones de DML.

Vamos a crear una aplicación que nos permita gestionar una base de datos que nos gestionará nuestra agenda de contactos.

Por lo tanto en este ejercicio vamos a realizar las siguientes tareas:

- Crear una BD en Oracle de Contactos
- Realizar una clase que conecte con el SGBD Oracle.
- Realizar una clase que construya un menú con las siguientes opciones:
 - Mostrar todos los Contactos
 - Crear un nuevo Contacto
 - Actualizar la información de un Contacto
 - Eliminar un Contacto
 - Buscar un contacto nombre o apellidos.
- Para cada contacto necesitamos almacenar:
 - Nombre
 - Apellidos
 - DNI
 - Teléfono
 - Email

Objetivos:

- Usar sentencias SQL para el DML (Data Manipulation Lenguaje).
- Establecer la conexión.
- Realizar una aplicación completa para la gestión de una BD.

Recursos:

- Acceso a Internet.

- Netbeans.
- JDBC.
- Driver Oracle.

Resolución:

Para este ejercicio, tendremos nuestra base de datos Oracle en una máquina virtual. La sentencia SQL para la creación de la tabla es la siguiente:

```
create table contactos(  
    dni varchar2(9) primary key,  
    nombre varchar2(30),  
    apellidos varchar2(50),  
    telefono number(9),  
    email varchar2(100)  
);
```

Después creamos el proyecto y añadimos el jar "ojdbc7.jar". Para poder comunicarnos con la base de datos Oracle.

Para conectarse a la base de datos se usara la función

```
DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());  
this.conn = DriverManager.getConnection(  
    "jdbc:oracle:thin:@172.16.90.99:1521:orcl",  
    "sys as SYSDBA",  
    "oracle"  
);
```

El primer parámetro de la función getConnection es la conexión. La parte "jdbc:Oracle:thin:" siempre debe ir, ya que indica el driver que estamos usando después va "@172.16.90.99" esa es la IP de la maquina virtual, si tuviéramos Oracle en nuestra propia máquina deberíamos poner localhost. Después va el puerto de Oracle que por defecto es 1521 y después el SID de la conexión. Los últimos dos parámetros son el usuario de la conexión y su contraseña.

La clase para interactuar con la base de datos quedaría como sigue:

```

package gestion_bd_contactos;

import java.sql.*;

public class Database {
    private Connection conn;

    public Database() {
        try{
            DriverManager.registerDriver(new oracle.jdbc.driver.OracleDriver());
            this.conn = DriverManager.getConnection(
                "jdbc:oracle:thin:@172.16.90.99:1521:orcl",
                "sys as SYSDBA",
                "oracle"
            );
            System.out.println("\n-----Conexión "
                + "establecida-----");
        }catch(SQLException e){
            System.err.println("Error al conectarse a la "
                + "base de datos");
        }
    }

    public void close() {
        try{
            conn.close();
        }catch(SQLException e){
            System.err.println("Error al cerrar la conexión");
        }
    }

    public void showAll() {
        System.out.println("-----Mostrando todos"
            + " los contactos-----");
        String q = "SELECT * from contactos";
        executeSelect(q);
    }

    public void addContact(String dni, String nombre,
        String apellidos, int telefono, String email){
        System.out.println("-----Añadiendo el contacto "
            + "con matricula "+dni+"-----");
        String query = "INSERT INTO contactos values ('"
            + dni+"', '" + nombre+"', '"
            + apellidos+"', '" + telefono+"', '"
            + email+"')";
        executeQuery(query, "insertado");
    }

    public void updateContact(String dni, String nombre,
        String apellidos, int telefono, String email){
        System.out.println("-----Modificando el contacto"
            + "con dni "+dni+"-----");
        String query = "update contactos set "
            + "nombre='"+nombre+"', "+ "apellidos='"+apellidos+"', "
            + "telefono='"+telefono+"', email='"+email+"'"
            + "where dni='"+dni+"' ";
        executeQuery(query, "actualizado");
    }

    public void deleteContact(String dni){
        System.out.println("-----Eliminado el contacto"
            + "con dni "+dni+"-----");
        String query = "delete from contactos "
            + "where dni='"+dni+"'";
        executeQuery(query, "eliminado");
    }
}

```

```

public void searchContact(String dni){
    System.out.println("-----Mostrando el contacto"
        + "con dni "+dni+"-----");
    String q = "SELECT * from contactos where dni='"+dni+"'";
    executeSelect(q);
}

private void executeQuery(String q, String action){
    try{
        Statement stm = conn.createStatement();
        int n = stm.executeUpdate(q);

        System.out.println("\nSe han "+action+" "+n+" contactos");

    }catch(SQLException e){
        e.printStackTrace();
        System.err.println("\nNo se pudo realizar la operacion");
    }
}

private void executeSelect(String q){
    try{
        ResultSet result = conn.createStatement().executeQuery(q);
        System.out.println("\nContactos encontrados en la BD: ");
        while(result.next()){
            System.out.println("\nDni: "+result.getString(1)+
                "\nNombre: "+result.getString(2)+
                "\nApellidos: "+result.getString(3)+
                "\nTelefono: "+result.getDouble(4)+
                "\nEmail: "+result.getString(5));
        }
    }catch(SQLException e){
        System.err.println("No se pudo realizar la consulta");
    }
}
}

```

Después, este sería el main:

```

package gestion_bd_contactos;

public class Gestion_BD_Contactos {
    public static void main(String[] args) {
        Database db = new Database();
        db.addContact("12345678A", "Nombre1", "Apellidos1",
            612345678, "email1@correo.es");
        db.addContact("23456789B", "Nombre2", "Apellidos2",
            617845678, "email2@correo.es");
        db.addContact("01234567C", "Nombre3", "Apellidos3",
            612965678, "email3@correo.es");

        db.showAll();

        db.updateContact("12345678A", "Nombre1", "Apellidos1",
            612345678, "emailnuevo@correo.es");

        db.searchContact("12345678A");

        db.deleteContact("12345678A");

        db.close();
    }
}

```

Y el resultado obtenido sería el siguiente:

```

run:

-----Conexión establecida-----
-----Añadiendo el contacto con matricula 12345678A-----

Se han insertado 1 contactos
-----Añadiendo el contacto con matricula 23456789B-----

Se han insertado 1 contactos
-----Añadiendo el contacto con matricula 01234567C-----

Se han insertado 1 contactos
-----Mostrando todos los contactos-----

Contactos encontrados en la BD:

Dni: 01234567C
Nombre: Nombre3
Apellidos: Apellidos3
Telefono: 6.12965678E8
Email: email3@correo.es

Dni: 12345678A
Nombre: Nombre1
Apellidos: Apellidos1
Telefono: 6.12345678E8
Email: email1@correo.es

```

```
Dni: 23456789B
Nombre: Nombre2
Apellidos: Apellidos2
Telefono: 6.17845678E8
Email: email2@correo.es
-----Modificando el contactocon dni 12345678A-----

Se han actualizado 1 contactos
-----Mostrando el contactocon dni 12345678A-----

Contactos encontrados en la BD:

Dni: 12345678A
Nombre: Nombrel
Apellidos: Apellidos1
Telefono: 6.12345678E8
Email: emailnuevo@correo.es
-----Eliminado el contactocon dni 12345678A-----

Se han eliminado 1 contactos

Se han eliminado 2 contactos
```