Java JDBC (CRUD)

domingo, 21 de enero de 2024 10:27 p. m.

```java
// SELECT CON JDBC
import java.sql.*;

public class Main {
    public static void main(String[] args) {
        // #1 -> Declarar variables Connection, Statement y ResultSet
        Connection connection = null;
        Statement statement = null;
        ResultSet resultSet = null;

        try {
            // #2 -> Establecer la conexión con la base de datos
            connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/project","root","Bred0420@@");
            System.out.println("Conexion exitosa <3");

            // #3 -> Crear objeto Statement
            statement = connection.createStatement();

            // #4 ->  Ejecutar consulta SQL
            resultSet = statement.executeQuery("SELECT * FROM employees");

            // #5 -> Procesamos los datos generados por la consulta SQL
            while (resultSet.next()) {
                System.out.println(resultSet.getString("first_name") + "," + resultSet.getString("last_name"));
            }

        } catch (SQLException e) {
            e.printStackTrace();
            System.out.println("Algo salio mal");
        }
    }
}
```

```java
// CREATE CON JDBC
import java.sql.*;

public class Main {
    public static void main(String[] args) {
        // #1 -> Declarar variables Connection, PreparedStatement y ResultSet
        Connection connection = null;
        PreparedStatement statement = null;

        try {
            // #2 -> Establecer la conexión con la base de datos
            connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/project","root","Bred0420@@");
            System.out.println("Conexion exitosa <3");

            //#3 -> Definir consulta SQL
            String sql = ("INSERT INTO employees (first_name, last_name) VALUES (?, ?)");
            // #4 -> Crear objeto PrepareStatement y pasarle la consulta SQL como argumento
            statement = connection.prepareStatement(sql);
            // #5 -> Asignarle los valores a la consulta SQL
            statement.setString(1, "Marlly");
            statement.setString(2, "Guido");

            // #6 -> Ejecución del query SQL (INSERT)
            int rowsAffected = statement.executeUpdate();

            // #7 -> Validar si el query se ejecutó correctamente
            if (rowsAffected > 0) {
                System.out.println("Se ha creado un nuevo empleado");
            }

        } catch (SQLException e) {
            e.printStackTrace();
            System.out.println("Algo salio mal");
        }
    }
}
```

```java
// UPDATE CON JDBC
import java.sql.*;

public class Main {
    public static void main(String[] args) {
        // #1 -> Declarar variables Connection, Statement y ResultSet
        Connection connection = null;
        Statement statement = null;
        ResultSet resultSet = null;

        try {
            // #2 -> Establecer la conexión con la base de datos
            connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/project","root","Bred0420@@");
            System.out.println("Conexion exitosa <3");

            // #3 -> Crear Objeto Statement
            statement = connection.createStatement();

            // #4 -> Ejecución del Query SQL (UPDATE)
            int rowsAffected = statement.executeUpdate("UPDATE employees " + "SET last_name = 'Montoya' " +
                    "WHERE first_name = 'Marlly'");

            // #5 -> Crear Objeto resultSet para mostrar los cambios
            resultSet = statement.executeQuery("SELECT * FROM employees ORDER BY first_name");

            // #6 -> Mostrar los cambios
            while (resultSet.next()) {
                System.out.println((resultSet.getString("first_name")) + ", " + (resultSet.getString("last_name")));
            }

        } catch (SQLException e) {
            e.printStackTrace();
            System.out.println("Algo salio mal");
        }
    }
}
```

```java
// DELETE CON JDBC
import java.sql.*;

public class Main {
    public static void main(String[] args) {
        // #1 -> Declarar variables Connection, PreparedStatement
        Connection connection = null;
        PreparedStatement statement = null;

        try {
            // #2 -> Establecer la conexión con la base de datos
            connection = DriverManager.getConnection("jdbc:mysql://localhost:3306/project","root","Bred0420@@");
            System.out.println("Conexion exitosa <3");

            // #3 -> Definir consulta SQL
            String sql = ("DELETE FROM employees WHERE first_name = (?)");
            // #4 -> Creamos objeto preparedStatement y le asignamos la consulta como argumento
            statement = connection.prepareStatement(sql);
            // #4 -> Definimos los valores de la consulta SQL
            statement.setString(1, "Marlly");

            // #5 Ejecutar el query
            int rowsAffected = statement.executeUpdate();

            if (rowsAffected > 0){
                System.out.println("Se ha eliminado el empleado correctamente");
            }

        } catch (SQLException e) {
            e.printStackTrace();
            System.out.println("Algo salio mal");
        }
    }
}
```