import java.sql.*; // importar las clases del API de JDBC

Clases Principales JDBC

- Connection: crea una conexión entre la aplicación y la base de datos.
- Statement: representa una sentencia SQL que ejecutará el servidor de base de datos.
- PreparedStatement: representa una sentencia SQL, que permite configurar o parametrizar.
- ResultSet: representa una tabla con el resultado tras ejecutar una sentencia SELECT.

1. Conexión

getConnection()
Connection con = DriverManager.getConnection("jdbc:mysql://<servidor>/<base de datos>, String Usuario, String password);
Connection con = DriverManager.getConnection("jdbc:mysql://localhost/Instituto", "Pepe", "12345");

close()
con.close();

2. Ejecución de consultas (SELECT)

String sql = "SELECT * FROM TABLA";
Statement sentencia = con.**createStatement**();

ResultSet executeQuery(String sql)
ResultSet rs = sentencia.executeQuery(sql);

3. Ejecución de consultas (INSERT, UPDATE o DELETE)

int executeUpdate(String sql) // Devuelve numero de filas afectadas por la sentencia
String sql = "DELETE FROM TABLA WHERE ….";
Statement sentencia = con.**createStatement**();
sentencia.executeUpdate(sql);

4. Clase ResulSet

- **boolean next():** mueve cursor a la siguiente fila, devuelve verdadero o falso si ha podido realizar el movimiento. // while (rs.next()) {……...}
- **String getString(String nombreCampo):** devuelve el valor del campo como una cadena.
- **int getInt(String nombreCampo):** devuelve el valor del campo como un entero.
  numero = rs.getInt("nombreCampo"); o numero = rs.getInt(1);
- **Double getDouble(String nombreCampo):** devuelve el valor del campo como un real.
- **Date getDate(String nombreCampo):** devuelve el valor del campo como una fecha.

5. Tipos de ResulSet

- **Statement createStatement (int tipoResultSet, int concurrencia)**
- El parámetro **tipoResultSet**
  - **ResultSet.TYPE_FOWARD_ONLY:** cursor solo se mueve hacia delante.
  - **ResultSet.TYPE_SCROLL_INSENSITIVE:** cursor se mueve hacia delante o atrás. El ResultSet es una copia de los datos almacenados de la bases de datos.
  - **ResultSet.TYPE_SCROLL_SENSITIVE:** cursor se mueve hacia delante o atrás. El ResultSet es una vista de los datos almacenados de la bases de datos.
- El parámetro **concurrencia**
  - **ResultSet.CONCUR_READ_ONLY:** los datos del ResultSet serán de solo lectura.
  - **ResultSet.CONCUR_UPDATABLE:** los datos del ResultSet modifican la base de datos.

6. Métodos para mover el cursor

- **boolean next():** avanza el cursor y activa la siguiente fila.
- **boolean previous():** retrocede el cursor y activa la fila anterior.
- **boolean first():** coloca el cursor en la primera fila, activándola.
- **boolean last():** coloca el cursor en la última fila, activándola.
- **boolean absolute(int numeroFila):** mueve el cursor a la enésima fila. Comienza en 1.
- **boolean relative(int cuantasFilas):** mueve el cursor tomando como base su posición actual.
- **void beforeFirst():** coloca el cursor justo delante de la primera fila.
- **void afterLast():** coloca el cursor justo después de la ultima fila.

7. Ubicación del cursor

- **boolean isBeforeFirst():** especifica si el cursor se encuentra delante de la primera fila.
- **boolean isAfterLast():** especifica si el cursor está colocado justo detrás de la última fila.
- **boolean isFirst():** devuelve true si el cursor está apuntando a la primera fila.
- **boolean isLast():** devuelve true si el cursor apunta a la última fila.

8. Sentencias parametrizadas

Ejecución de consultas (SELECT):

String sql = “SELECT * FROM TABLA WHERE CAMPO=?”;
PreparedStatement sentecia = con.**prepareStatement**(sql);
sentencia.**setInt**(int índiceParámetro, int valor);
ResultSet rs = sentencia.executeQuery();

Ejecución de consultas (INSERT, UPDATE o DELETE):

String sql = “DELETE FROM TABLA WHERE CAMPO=?”;
PreparedStatement sentencia = con.**prepareStatement** (sql);
sentencia.**setInt**(int índiceParámetro, int valor);
sentencia.executeUpdate();

Métodos para asignar parámetros:

- **void setString(int índiceParámetro, String valor)**
- **void setInt(int índiceParámetro, int valor)**
- **void setDouble(int índiceParámetro, double valor)**
- **void setBoolean(int índiceParámetro, boolean valor)**
- **void setDate(int índiceParámetro, Date valor)**
- **void setNull(int índiceParámetro, int tipoSQL)**