14/06/2025

Saturday

**JDBC - CRUD operations**:

CRUD stands for:

* **C** – Create (Insert)
* **R** – Read (Select)
* **U** – Update
* **D** – Delete

**How JDBC Works Internally**

**1. Load the Driver Class**

This step loads the JDBC driver into memory.

Class.forName("oracle.jdbc.driver.OracleDriver");

This allows the DriverManager to recognize the Oracle driver.

**2. Create a Connection**

Establishes a connection to the database using the JDBC URL, username, and password.

Connection conn = DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:XE", "system", "password");

**3. Create a Statement / PreparedStatement**

Used to send SQL commands to the database.

Statement stmt = conn.createStatement();

// or

PreparedStatement pstmt = conn.prepareStatement("INSERT INTO students ...");

**4. Execute SQL Query**

* For SELECT:

ResultSet rs = stmt.executeQuery("SELECT \* FROM students");

* For INSERT, UPDATE, DELETE:

int rows = pstmt.executeUpdate();

**5. Process the Result (for SELECT)**

while (rs.next()) {

System.out.println(rs.getString("name"));

}

**6. Close the Connection**

Always close the connection when done.

conn.close();

**📘 JDBC CRUD Summary Table**

| **Operation** | **SQL Command** | **JDBC Method** |
| --- | --- | --- |
| **Create** | INSERT INTO | executeUpdate() |
| **Read** | SELECT \* FROM | executeQuery() |
| **Update** | UPDATE ... SET | executeUpdate() |
| **Delete** | DELETE FROM | executeUpdate() |

**Hibernate**:

* ORM (Object Relational Mapping) tool/framework
* It maps java classes to database tables
* Allows you to perform database operations using Java Objects instead of SQL queries.

ORM:

* It simplifies database operations by allowing java classes (objects) to be mapped to database tables.

Features of hibernate:

1. ORM support
2. Instead of complex SQL queries we can use HQL (Hibernate Query Language).
3. Automatic Table Creation in respective databases.
4. Database Independence : easily we can switch between databases