

# Day 23

---

## JDBC

- Java DataBase Connectivity( JDBC )
- Set of rules are called specification/Standard. In Java it is called interface.
- Specification = [ Collection of Abstract classes & Interfaces ]
- JDBC is specification defined by Sun Microsystems.
  - Vendor : SUN/ORACLE
  - Implementor : DB Vendors[ Oracle, MySQL, PostGres, etc ]
  - User : Java Application developer
- Employee -> name, empid, salary, dept, desig, joinDate --> Record
- Collection of data/record is called database.
- A S/W which is used to manage large data / database is called database management system[ DBMS ].
- Any DBMS, which implements E.F Codd's rule is called relational DBMS.
- If we want to access & process data from relational database management system then we should use JDBC.
- To use JDBC we should import java.sql package.
- Interfaces:
  - Driver
  - Connection
  - Statement
  - PreparedStatement
  - CallableStatement
  - ResultSet
  - Blob
  - Clob
  - NClob
  - DatabaseMetaData
  - ParameterMetaData
  - ResultSetMetaData
- Classes:
  - Date
  - Time
  - Timestamp
  - DriverManager
  - DriverPropertyInfo
  - Types
  - SQLPermission
- Exception:
  - SQLException

## JDBC Version:

1. JDK 1.8 : JDBC 4.2 API
2. JDK 1.7 : JDBC 4.1 API
3. JDK 1.6 : JDBC 4.0 API
4. JDK 1.4 : JDBC 3.0 API
5. JDK 1.2 : JDBC 2.0 API
6. JDK 1.1 JDBC 1.0 API

## JDBC Driver

- Driver is a program which is responsible for converting Java request into SQL request and SQL response into Java response.
- Types
  1. TYPE - I
    - It is also called as Bridge Driver
    - Example : `sun.jdbc.odbc.JdbcOdbcDriver`
    - It is obsolete Driver.
  2. TYPE - II
    - It is called Native API Driver
    - Example : ORACLE OCI Driver
    - It is platform dependant driver
  3. TYPE - III
    - It is also called as Network protocol driver
    - Example : RMI web logic driver
    - It is pure java driver.
  4. TYPE - IV
    - It is called Pure Java Driver / Database protocol driver
    - Example : `com.mysql.jdbc.Driver`

## JDBC Connection Information

- Database Server :MySQL 8.0.18
- User : sandeep
- Password : sandeep
- Database : home\_db
- URL : `jdbc:mysql://127.0.0.1:3306/home_db;` ; `jdbc:mysql://localhost:3306/home_db;`
- Driver : `com.mysql.cj.jdbc.Driver`
- Connector : `mysql-connector-java-8.0.18.jar`

## Steps to connect Java application to the database.

- Step 0:
  - Include JDBC connector into Runtime classpath/build path
- Step 1:
  - Load and register JDBC Driver
- Step 2:

- Establish Connection using users credential
- Step 3:
  - Create Statement/PreparedStatement/CallableStatement to execute query
- Step 4:
  - Prepare and execute query
- Step 5:
  - Close resources

### **Plain Old Java Object( POJO )**

- It is also called as DTO / VO / BO / Entity
- Rules:
  1. It must be packaged public class
  2. It should contain default constructor
  3. For columns it should contain private fields
  4. For every private field it should contain getter and setter method( camel case convention)
  5. It should not contain B.L. method but it can contain toString, hashCode, equals method.
- For ORM we should define POJO class.

### **Data Access Object( DAO ) Layer**

- If we want to separate BL from Data manipulation logic then we should define DAO.
- Rules to define DAO
  1. It must be packaged public class.
  2. It should contain default constructor
  3. It must contain CRUD operation
- C : Create/ Insert
- R : Read/Select
- U : Update
- D : Delete

sandeepkulange@gmail.com sandeepkulange@sunbeaminfo.com 9527325202

Java Head First - Kathy Seira Java Complete Reference

Core Java Volume I and II Java Certification : Khalid Mughal

Java Docs & Oracle Tutorial Jenkov & JornelDev