Generics: Type safe code can be used to implement generics.

Syntax: List <emp> list= new ArrayList<emp> (); [-> It will collect only employee type objects].

List <Integer> list= new ArrayList<Integer> (); [-> It will collect only Integer class type objects].

Ex: list.add(10);

List.add(“Sandeep”) -> Error

List <Object> list= new ArrayList<Object> (); [-> It will collect only different type of objects].

What is JDBC?

We are connection our application with database to collect the information from database is called java database connectivity.

Driver

JVM

Database

1. type1: JDBC ODBC Bridge Driver -> It is provided by SUN Micro Systems. (Bad Performance)
2. type2: Partly native partly java Driver (It is better than type 1). [ Half code in native language and Half code in java]
3. type3: Net Protocol Drive (It uses third)
4. type4: Pure java Drivers 0r thin driver (Performance was Excellent) [It is purely developed in java 99% of industries]

DDL (Data Definition Language): Create, Alter, Drop.

DML (Data Manipulation Language): Insert, Update, Delete.

DQL (Data Query Language): Select.

2)

Com.mysql.jdbc.Driver -------------🡪 SQL Vendor.

Oracle.jdbc.diver.OracleDriver. ---🡪 Oracle Driver.

Oci.Driver -> Type 2

**Creating and loading the JDBC:**

1. **Load the Driver -> Register Driver JVM**
2. **Establishing the connectivity [ Host , Port, URL]**

**Jdbc:mysql://Localhost: 3306: Schematype.**

1. **Create the statement.**
2. **Execute.**
3. **Handle the result.**
4. ->Class.forName (“ com.mysql.jdbc.Driver”) -> Name of the class and it will be creating the object and -> Register the driver to driver manger.
5. Connection con=DriverManger.getConnection(“”);

**(3,4,5) Static, Dynamic and stored are different types of queries:**

* **Static ->** We use Statement. ( con.createStatment();) .

If there is no need of sending any conditions.

For example, Select \* from emp;

boolean s= stmt.execute();

* **Dynamic->**PreparedStatment (con.prepareStatment();)

int k= stmt.executeUpdate();

* **Stored->** CallabillStatment. (con.preapreCall();)

ResultSet rs= stmt.executeQuery();

Downloading MYSQL:

1. Mysql.com.
2. After that click link “mysql workbench”.
3. Select windows Operating system and click on the windows “MSI Installer”.
4. After that download sql queries for that click on the link “MYSql Connector”.
5. Click on this link **“**[**Connector/J**](http://dev.mysql.com/downloads/connector/j/)**:”** Standardized database driver for Java platforms and development.
6. Download “ZIP. file” .

**Transactions:**

1. Atomicity: All the blocks are none of them need to be update is called atomicity. (Transactional Batch) -> prepareStatment();
2. Concurrency: Parallel we can perform two tasks at a same time.
3. Isolation: Read only, Read- write, dirty read, still data.
4. Durability: Make it permanent -> statement ();

**Without seeing the name of table we can know the names,values, data……….. by using “ MetaData”**

ResultSetMetaData rsmd=rs.getMetaData();

getMetaColumnSize();

getMetaColumnDatatype();

getMetaColumnCount();

* Stored procedure and function both are used when we need repeatedly executed.

**EXAMPLE:**

CallabaleStatment cs=Con.prepareCall(“{ call find cube (?,?,?)}”);

cs.registerOutParameter(2, java.sql.Types.Integer);

**Procedure:**

Create procedure procedure\_name (IN x int, out res int, INOUT res1 int )

Begin

Set res=x\*x\*X;

Set res1=res\*x;

End;

**Function:**

Create FUNCTION function\_name (int p ) return res;

Begin

Declare res x;

Set res=x\*x\*X;

Set res1=res\*x;

End;