WEEK-6

Aim:

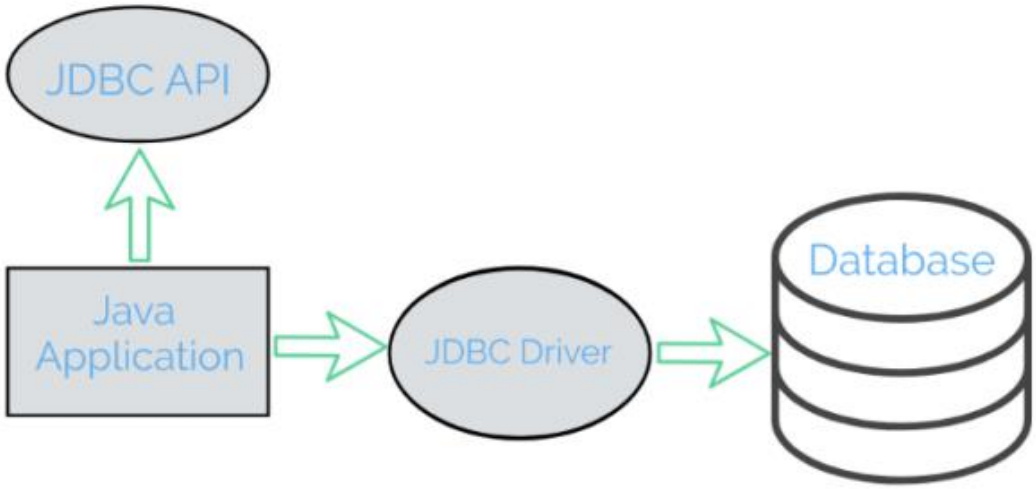
Create necessary tables for the application chosen using JDBC and establish database connectivity.

Scenario 1: Establish the connectivity using JDBC drivers.
Scenario 2: Use create and select statements.
Scenario 3: Use insert, update and delete queries.

Java Database Connectivity (JDBC):

JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is part of JavaSE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database. There are four types of JDBC drivers.

i) JDBC-ODBC Bridge Driver
ii) Native Driver
iii) Network Protocol Driver
iv) Thin Driver



6.1 Java Database Connectivity with MySQL:

To connect Java application with the MySQL database, we need to follow 4 following steps.

1. **Driver class:** The driver class for the MySQL database is **com.mysql.jdbc.Driver**.
2. **Connection URL:** The connection URL for the mysql database is **jdbc:mysql://localhost:3306/sonoo** where **jdbc** is the **API**, mysql is the database, localhost is the server name on which mysql is running, we may also use IP address, 3306 is the port number and sonoo is the database name.
3. **Username**: The default username for the mysql database is **root**
4. **Password**: It is the password given by the user at the time of installing the mysql database.

After the installation of MySQL, we have to select a database in which we will work.

The list of databases can be displayed using the **show database;** command.

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sakila             |
| sys                |
| world              |
+--------------------+
6 rows in set (0.02 sec)
```

To select a database use the **use databasename;** command. Here world database is in use.

```
mysql> USE world;
Database changed
```

To view the list of tables in the world database use the command **show tables;**

```
mysql> show tables;
+-----------------+
| Tables_in_world |
+-----------------+
| city            |
| country         |
| countrylanguage |
+-----------------+
3 rows in set (0.01 sec)
```

6.2 Setup for JDBC using VS code:

Step-1: First we need to check whether the JDK is installed or not if not, we need to install.

Step-2: After installing JDK open the visual studio code and create the new java project by following steps

- Open command palette and search for java project
- Select No Build tools and choose the location where you have to save the application

Step-3: After creating the Java project add **mysqlconnector.jar** file to the reference library (that you can find out at the Java project that was appear at the left bottom of the vs code).

i) PROGRAM for creating a table using create and select statement.

```
import java.sql.*;

public class App {
    public static void main(String[] args) throws Exception {
        Connection connect = null;
        Class.forName("com.mysql.cj.jdbc.Driver");
        connect =
DriverManager.getConnection("jdbc:mysql://localhost:3306/world", "root",
"Manvi@09022001");
        Statement statement =  connect.createStatement();
  ResultSet resultSet = statement.executeQuery("select * from student");

        while(resultSet.next()){
            System.out.println("Student ID: "+resultSet.getString(1));
```

```
            System.out.println("Student Name: "+resultSet.getString(2));
            System.out.println("Student Department: "+resultSet.getString(3));
            System.out.println();
        }
        resultSet.close();
        statement.close();
        connect.close();
    }
}
```

OUTPUT:

```
mysql> show tables;
+-----------------+
| Tables_in_world |
+-----------------+
| city            |
| country         |
| countrylanguage |
| student         |
+-----------------+
4 rows in set (0.00 sec)
```

ii) PROGRAM using insert statement.

```
import java.sql.*;

public class App {
    public static void main(String[] args) throws Exception {
        Connection connect = null;
        Class.forName("com.mysql.cj.jdbc.Driver");
        connect =
DriverManager.getConnection("jdbc:mysql://localhost:3306/world", "root",
"Manvi@09022001");
        Statement statement =  connect.createStatement();
        // insert
        statement.execute("create table student1 (s_id integer, s_name
varchar(20), s_dept varchar(5) )");
        statement.execute("insert into student1 values(01,'Arnav','cse')");
        statement.execute("insert into student1 values(02,'Shaurya','eee')");
        statement.execute("insert into student1 values(03,'Khushi','ece')");
        statement.execute("insert into student1 values(04,'Myrah','chem')");
ResultSet resultSet = statement.executeQuery("select * from student1");
        while(resultSet.next()){
            System.out.println("Student ID: "+resultSet.getString(1));
            System.out.println("Student Name: "+resultSet.getString(2));
            System.out.println("Student Department: "+resultSet.getString(3));
            System.out.println();
        }
        resultSet.close();
        statement.close();
        connect.close();
    }
}
```

OUTPUT:

```
Student ID: 1
Student Name: Arnav
Student Department: cse

Student ID: 2
Student Name: Shaurya

Student ID: 3
Student Name: Khushi
Student Department: ece

Student ID: 4
Student Name: Myrah
Student Department: chem
```

```
mysql> select *from student1;
+------+---------+--------+
| s_id | s_name  | s_dept |
+------+---------+--------+
|    1 | Arnav   | cse    |
|    2 | Shaurya | eee    |
|    3 | Khushi  | ece    |
|    4 | Myrah   | chem   |
+------+---------+--------+
4 rows in set (0.00 sec)
```

iii) PROGRAM using update statement.

```
import java.sql.*;

public class App {
    public static void main(String[] args) throws Exception {
        Connection connect = null;
        Class.forName("com.mysql.cj.jdbc.Driver");
        connect =
DriverManager.getConnection("jdbc:mysql://localhost:3306/world", "root",
"Manvi@09022001");
        Statement statement =  connect.createStatement();
// update....
        String sql = "update student set s_name='nerella' where s_id=2";
        statement.executeUpdate(sql);

ResultSet resultSet = statement.executeQuery("select * from student1");
        while(resultSet.next()){
            System.out.println("Student ID: "+resultSet.getString(1));
            System.out.println("Student Name: "+resultSet.getString(2));
            System.out.println("Student Department: "+resultSet.getString(3));
            System.out.println();
        }
        resultSet.close();
        statement.close();
        connect.close();
    }
}
```

OUTPUT:

```
Student ID: 1
Student Name: Arnav
Student Department: cse

Student ID: 2
Student Name: Manasi
Student Department: eee

Student ID: 3
Student Name: Khushi
Student Department: ece

Student ID: 4
Student Name: Myrah
Student Department: chem
```

```
mysql> select *from student1;
+------+--------+--------+
| s_id | s_name | s_dept |
+------+--------+--------+
|    1 | Arnav  | cse    |
|    2 | Manasi | eee    |
|    3 | Khushi | ece    |
|    4 | Myrah  | chem   |
+------+--------+--------+
4 rows in set (0.00 sec)
```

d) PROGRAM using delete statement

```
import java.sql.*;

public class App {
    public static void main(String[] args) throws Exception {
        Connection connect = null;
        Class.forName("com.mysql.cj.jdbc.Driver");
        connect =
DriverManager.getConnection("jdbc:mysql://localhost:3306/world", "root",
"Manvi@09022001");
        Statement statement =  connect.createStatement();
// delete...

        String sql="delete from student where s_id=2";
        statement.executeUpdate(sql);
  ResultSet resultSet = statement.executeQuery("select * from student1");

        while(resultSet.next()){
            System.out.println("Student ID: "+resultSet.getString(1));
            System.out.println("Student Name: "+resultSet.getString(2));
            System.out.println("Student Department: "+resultSet.getString(3));
            System.out.println();
        }
        resultSet.close();
        statement.close();
        connect.close();
    }
}
```

OUTPUT:

```
Student ID: 1
Student Name: Arnav
Student Department: cse

Student ID: 3
Student Name: Khushi
Student Department: ece

Student ID: 4
Student Name: Myrah
Student Department: chem
```

```
mysql> select *from student1;
+------+--------+--------+
| s_id | s_name | s_dept |
+------+--------+--------+
|    1 | Arnav  | cse    |
|    3 | Khushi | ece    |
|    4 | Myrah  | chem   |
+------+--------+--------+
3 rows in set (0.00 sec)
```