YASH SARANG - EXPERIMENT 12 - DBMS

Aim: Implement Database Connectivity (JDBC, ODBC)

Theory:

JDBC Architecture.

In the two tier model, a Java app talks directly to the data source. This requires a JDBC driver that can communicate with the particular data source. may be loaded on another machine to which the user is connected via network.

This is reffered to as a client/server config, with user's machine as the client, & the machine housing the data source as the server. The network can be an instant.

In the 3 tier model, commands are sent to a "middle-tier" of services, which then sends the commands to the data source. The data source processes the commands & sends the results back to the middle tier, which then sends them to the user

Until recently, the middle tier has often been written in languages such as C or C++ which offers fast performance. However with the introduction of optimizing compilers that translate Java bytecode into efficient machine specific code & technologies such as enterprise JavaBeans, the Java platform is fast becoming a standard platform of middle tier development.

Fundamental Steps in JDBC.

① Import JDBC packages
② Load & register the JDBC driver.
③ Open a connection to the database.
④ Create a statement object to perform a query.
⑤ Execute the statement object & return a query result set
⑥ Process the result set.
⑦ Close the result set & statement objects.
⑧ Close the connection.

```java
import java.sql.*;
public class jdbc {

    static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
    static final String DB_URL = "jdbc:mysql://localhost:3306/";
    static final String USER = "java";
    static final String PASS = "subo";
public static void main(String[] args) {
Connection conn = null;
Statement stmt = null;
try{

Class.forName("com.mysql.cj.jdbc.Driver");

System.out.println("Connecting to database...");
conn = DriverManager.getConnection(DB_URL, USER, PASS);

System.out.println("Creating database...");
stmt = conn.createStatement();

String sql1 = ("CREATE DATABASE STUDENTS1");
stmt.executeUpdate(sql1);
System.out.println("Database created successfully...");
}

catch(SQLException se){
se.printStackTrace();
}

catch(Exception e){
e.printStackTrace();
}

finally{
    try{
        if(stmt!=null)
        stmt.close();
        }

    catch(SQLException se2){
}
try{
    if(conn!=null)
    conn.close();
    }
    catch(SQLException se){
    se.printStackTrace();}
}

System.out.println("Goodbye!");
}
}
```

```
Connecting to database...
Creating database...
java.sql.SQLSyntaxErrorException: Access denied for user 'java'@'localhost' to database 'students1'
        at com.mysql.cj.jdbc.exceptions.SQLError.createSQLException(SQLError.java:120)
        at com.mysql.cj.jdbc.exceptions.SQLExceptionsMapping.translateException(SQLExceptionsMapping.java:122)
        at com.mysql.cj.jdbc.StatementImpl.executeUpdateInternal(StatementImpl.java:1340)
        at com.mysql.cj.jdbc.StatementImpl.executeLargeUpdate(StatementImpl.java:2089)
        at com.mysql.cj.jdbc.StatementImpl.executeUpdate(StatementImpl.java:1251)
        at jdbc.main(jdbc.java:22)
Goodbye!
```

```
Connecting to database...
Creating database...
Goodbye!
```