**Java Assignment No.9**

/\*

    Ques : Create below table in database.

Student (Roll\_No int, Name String, City varchar, Grade Char,

Marks number)

Write a java program to perform below operations also use

exception handling to handle different exceptions (like database

connection error, query error etc. as per your choice)

a) Insert below data in above table.

1. Atul, Sangli, A, 90.50

2. Sangram, Sangli, B, 70.25

3. Satya, Mumbai, B, 61.36

4. Jaydeep, Pune, B, 60.95

5. Prashant, Sangli, C, 55.26

6. Abhi, Pune, C, 55.84

    Name : Sandesh shivaji shinde

    PRN : 23620006

\*/

import java.sql.\*;

public class Ques\_1 {

    // Database connection parameters

    static final *String* JDBC\_DRIVER = "com.mysql.cj.jdbc.Driver";

    static final *String* DB\_URL = "jdbc:mysql://localhost:3306/mydb"; // Change mydb to your database name

    static final *String* USER = "root";

    static final *String* PASS = "root";

    public static *void* main(*String*[] *args*) {

*Connection* conn = null;

*Statement* stmt = null;

        try {

            // Register JDBC driver

            Class.forName(JDBC\_DRIVER);

            // Open a connection

            System.out.println("Connecting to database...");

            conn = DriverManager.getConnection(DB\_URL, USER, PASS);

            // Create a statement

            stmt = conn.createStatement();

            // Create Students table

*String* createTableSQL = "CREATE TABLE IF NOT EXISTS Students ("

                    + "Roll\_No INT PRIMARY KEY,"

                    + "Name VARCHAR(255),"

                    + "City VARCHAR(255),"

                    + "Grade CHAR(1),"

                    + "Marks DECIMAL(5,2)"

                    + ")";

            stmt.executeUpdate(createTableSQL);

            // Insert data into the table

*String*[] insertData = {

                    "INSERT INTO Students VALUES (1, 'Atul', 'Sangli', 'A', 90.50)",

                    "INSERT INTO Students VALUES (2, 'Sangram', 'Sangli', 'B', 70.25)",

                    "INSERT INTO Students VALUES (3, 'Satya', 'Mumbai', 'B', 61.36)",

                    "INSERT INTO Students VALUES (4, 'Jaydeep', 'Pune', 'B', 60.95)",

                    "INSERT INTO Students VALUES (5, 'Prashant', 'Sangli', 'C', 55.26)",

                    "INSERT INTO Students VALUES (6, 'Abhi', 'Pune', 'C', 55.84)"

            };

            for (*String* sql : insertData) {

                stmt.executeUpdate(sql);

            }

            System.out.println("Data inserted successfully.");

            // Delete record for Roll\_No 5

*String* deleteRecordSQL = "DELETE FROM Students WHERE Roll\_No = 5";

            stmt.executeUpdate(deleteRecordSQL);

            System.out.println("Record with Roll\_No 5 deleted successfully.");

            // Update city from Sangli to Pune

*String* updateCitySQL = "UPDATE Students SET City = 'Pune' WHERE City = 'Sangli'";

            stmt.executeUpdate(updateCitySQL);

            System.out.println("City updated successfully.");

            // Display names of students having marks greater than 60

*String* displayNamesSQL = "SELECT Name FROM Students WHERE Marks > 60";

*ResultSet* rs = stmt.executeQuery(displayNamesSQL);

            System.out.println("Names of students with marks greater than 60:");

            while (rs.next()) {

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

            }

            // Display students according to their marks (Descending order)

*String* displayByMarksSQL = "SELECT \* FROM Students ORDER BY Marks DESC";

            rs = stmt.executeQuery(displayByMarksSQL);

            System.out.println("\nStudents sorted by marks (Descending order):");

            while (rs.next()) {

                System.out.println(rs.getInt("Roll\_No") + "\t" +

                                   rs.getString("Name") + "\t" +

                                   rs.getString("City") + "\t" +

                                   rs.getString("Grade") + "\t" +

                                   rs.getDouble("Marks"));

            }

            // Clean-up environment

            rs.close();

            stmt.close();

            conn.close();

        } catch (*SQLException* *se*) {

            // Handle errors for JDBC

            se.printStackTrace();

        } catch (*Exception* *e*) {

            // Handle errors for Class.forName

            e.printStackTrace();

        } finally {

            // Finally block used to close resources

            try {

                if (stmt != null) stmt.close();

            } catch (*SQLException* *se2*) {

            } // nothing we can do

            try {

                if (conn != null) conn.close();

            } catch (*SQLException* *se*) {

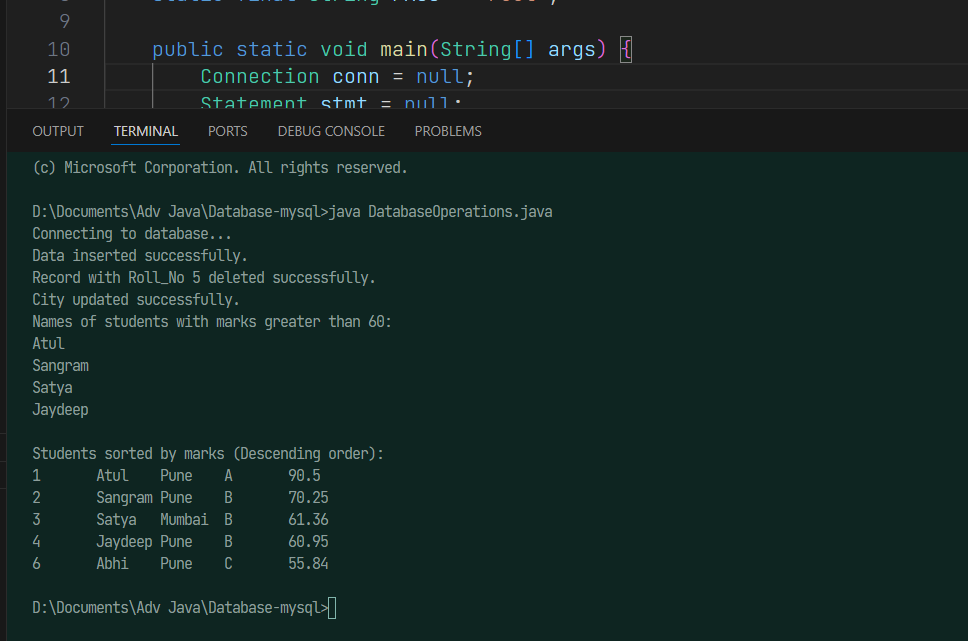
                se.printStackTrace();

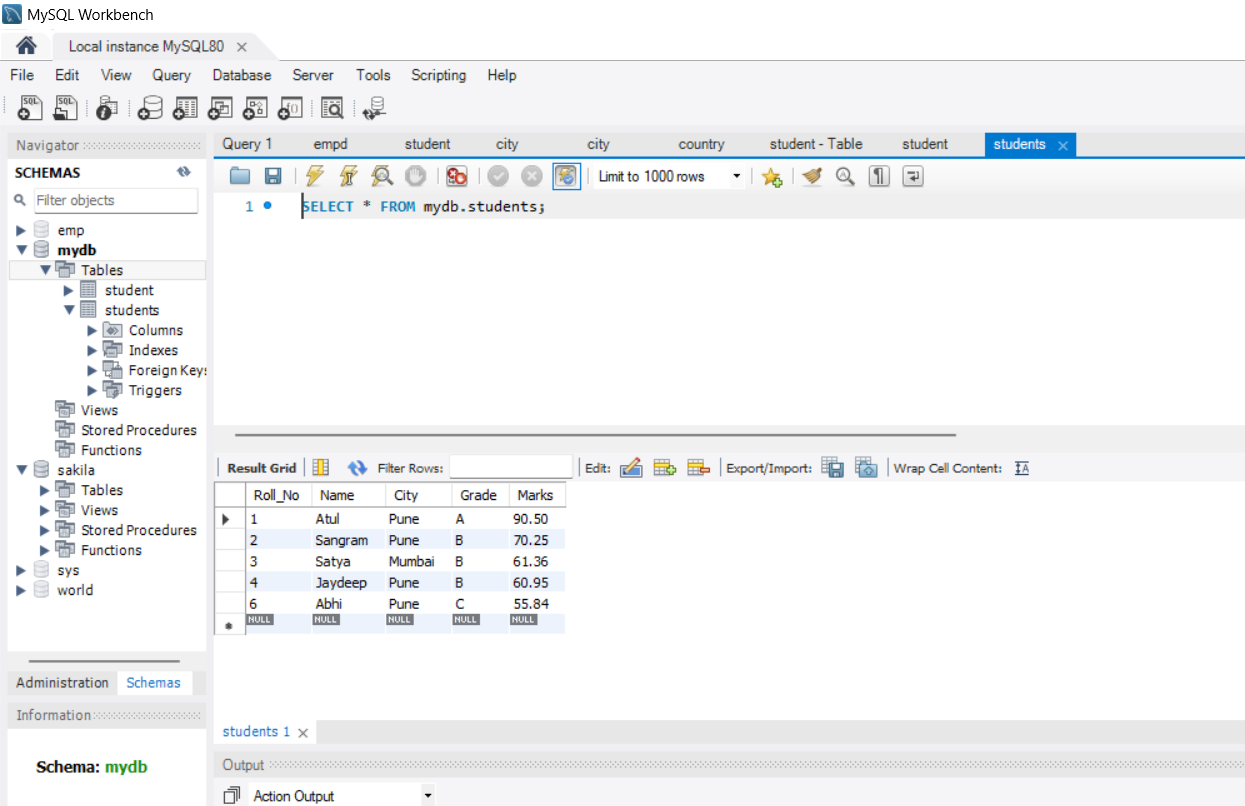
            } // end finally try

        } // end try

    }

}



****