Create jdbc connection using PrepareStatement and use Placeholder ‘ ?’ for taking input from user.

```java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.Scanner;

public class CreateDB {

    public static void main(String[] args) throws SQLException, ClassNotFoundException {
        Connection con = null;
        PreparedStatement pstmt = null;
        Scanner sc = new Scanner(System.in);

            // Getting the database name from user input
            System.out.println("Enter the name of the database you want to create: ");
            String dbName = sc.nextLine();

            // Establishing the connection
            con = DriverManager.getConnection("jdbc:mysql://localhost:3306/", "root", "root");

            // SQL query with a placeholder for the database name
            String sql = "CREATE DATABASE IF NOT EXISTS " + dbName;  // Placeholder can't be used here

            // Creating a PreparedStatement object (though Statement is better here)
            pstmt = con.prepareStatement(sql);

            // Executing the query
            pstmt.executeUpdate();
            System.out.println("Database '" + dbName + "' created successfully...");

            sc.close();
    }
}
```

Problems Javadoc Declaration Console

```
<terminated> CreateDB (1) [Java Application] C:\Users\wasim\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_22.0.2.v20240802-1626\jre\b
Enter the name of the database you want to create:
mydb2
Database 'mydb2' created successfully...
```

```java
    public static void main(String[] args) throws SQLException
    {

        // Establishing the connection
        Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/mydbms", "root", "root");

        // Creating a Statement object
        Statement stmt = con.createStatement();

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter Your Table name : ");
        String table=sc.nextLine();


        System.out.println("How many columns do you want to add?: ");
        int n = sc.nextInt();
        sc.nextLine();  // Consume newline left by nextInt()

        String sql = "CREATE TABLE "+table+" (\n";

        for (int i = 1; i <= n; i++) {
            System.out.println("Enter column name (e.g., columnName): ");
            String col = sc.nextLine();

            System.out.println("Enter data type (e.g., INT, VARCHAR(50)): ");
            String data = sc.nextLine();

            System.out.println("Do you want to add any key constraint (yes/no)?");
            String s = sc.nextLine();
            String key = "";

            if (s.equalsIgnoreCase("yes")) {
                System.out.println("Enter the key constraint (e.g., PRIMARY KEY, AUTO_INCREMENT, UNIQUE): ");
                key = sc.nextLine();
            }
            if(i<n)
                sql=sql+col+" "+data+" "+key+",\n";
            else
                sql=sql+col+" "+data+" "+key+");";
        }
        System.out.println("Generated SQL query:");
        stmt.executeUpdate(sql);

        System.out.println("Table created successfully...");

        con.close();


    }
```

```
    // SQL query with placeholders
    String sql = "INSERT INTO EMP (empid,ename, job, mgr, hiredate, sal, comm, deptno) VALUES (?,?, ?, ?, ?, ?, ?, ?)";

    // Creating a PreparedStatement object
    pstmt = con.prepareStatement(sql);



    // Getting input from the user
    System.out.println("Enter employee id: ");
    int empid = sc.nextInt();
    sc.nextLine();  // Consume newline left by nextInt()

    System.out.println("Enter employee name: ");
    String ename = sc.nextLine();

    System.out.println("Enter employee job: ");
    String job = sc.nextLine();

    System.out.println("Enter manager name: ");
    String mgr = sc.nextLine();

    System.out.println("Enter hire date (YYYY-MM-DD): ");
    String hiredate = sc.nextLine();

    System.out.println("Enter salary: ");
    int sal = sc.nextInt();

    System.out.println("Enter commission: ");
    int comm = sc.nextInt();

    System.out.println("Enter department number: ");
    int deptno = sc.nextInt();

    // Setting values for placeholders
    pstmt.setInt(1, empid);
    pstmt.setString(2, ename);
    pstmt.setString(3, job);
    pstmt.setString(4, mgr);
    pstmt.setString(5, hiredate);
    pstmt.setInt(6, sal);
    pstmt.setInt(7, comm);
    pstmt.setInt(8, deptno);

    // Executing the query
    int rowsInserted = pstmt.executeUpdate();
    if (rowsInserted > 0) {
        System.out.println("A new employee was inserted successfully!");
    }

    sc.close();
```

```
// Establishing the database connection
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/mydbms", "root", "root");
Scanner sc = new Scanner(System.in);
String sql = "";

// Prompting user to choose what to update
System.out.println("Enter 1 to update ID, 2 to update Name, 3 to update Age");
int choice = sc.nextInt();
sc.nextLine();  // Consume newline

// Defining SQL queries based on user choice
if (choice == 1) {
    System.out.println("Enter current ID: ");
    int currentId = sc.nextInt();
    System.out.println("Enter new ID: ");
    int newId = sc.nextInt();

    sql = "UPDATE STUDENT SET ID = ? WHERE ID = ?";
    PreparedStatement ps = con.prepareStatement(sql);
    ps.setInt(1, newId);
    ps.setInt(2, currentId);

    int rowsUpdated = ps.executeUpdate();
    System.out.println("Updated " + rowsUpdated + " row(s) with new ID = " + newId);

} else if (choice == 2) {
    System.out.println("Enter ID of the student whose name you want to update: ");
    int id = sc.nextInt();
    sc.nextLine();  // Consume newline
    System.out.println("Enter new name: ");
    String newName = sc.nextLine();

    sql = "UPDATE STUDENT SET NAME = ? WHERE ID = ?";
    PreparedStatement ps = con.prepareStatement(sql);
    ps.setString(1, newName);
    ps.setInt(2, id);

    int rowsUpdated = ps.executeUpdate();
    System.out.println("Updated " + rowsUpdated + " row(s) with new name = " + newName);

} else if (choice == 3) {
    System.out.println("Enter ID of the student whose age you want to update: ");
    int id = sc.nextInt();
    System.out.println("Enter new age: ");
    int newAge = sc.nextInt();

    sql = "UPDATE STUDENT SET AGE = ? WHERE ID = ?";
    PreparedStatement ps = con.prepareStatement(sql);
    ps.setInt(1, newAge);
    ps.setInt(2, id);
```

```java
public class DeleteData {

    public static void main(String[] args) throws SQLException {

        // Establishing connection to the database
        Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/mydbms", "root", "root");
        Scanner sc = new Scanner(System.in);
        String sql = "";

        // Prompt user for delete option
        System.out.println("Delete by id press 1 \nDelete by name press 2");
        int n = sc.nextInt();
        sc.nextLine();  // Consume the newline

        // Choose deletion by ID or name
        if (n == 1) {
            System.out.println("Enter the id: ");
            int id = sc.nextInt();
            sql = "DELETE FROM STUDENT WHERE ID = ?";

            // Creating PreparedStatement and setting the ID parameter
            PreparedStatement ps = con.prepareStatement(sql);
            ps.setInt(1, id);

            // Executing the delete operation
            int rowsAffected = ps.executeUpdate();
            System.out.println("Deleted " + rowsAffected + " row(s) with ID = " + id);

        } else if (n == 2) {
            System.out.println("Enter the name: ");
            String name = sc.nextLine();
            sql = "DELETE FROM STUDENT WHERE NAME = ?";

            // Creating PreparedStatement and setting the name parameter
            PreparedStatement ps = con.prepareStatement(sql);
            ps.setString(1, name);

            // Executing the delete operation
            int rowsAffected = ps.executeUpdate();
            System.out.println("Deleted " + rowsAffected + " row(s) with name = '" + name + "'");
        } else {
            System.out.println("Invalid option selected.");
        }

        // Closing resources
        sc.close();
        con.close();
    }
}
```

```java
public class DisplayData {

    public static void main(String[] args) throws SQLException, ClassNotFoundException {
        Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/mydbms", "root", "root");
        Statement stmt = con.createStatement();

        // SQL query to retrieve data from the Students table
        String sql = "SELECT * FROM Student";

        // Executing the query and getting the result set
        ResultSet rs = stmt.executeQuery(sql);

        System.out.println(" id |      name      | age | marks");

        while(rs.next())
        {
            System.out.println("------------------------------------");
            int id = rs.getInt("id");
            String name= rs.getString("name");
            int age = rs.getInt("age");
            int marks = rs.getInt("marks");

            int size=15-name.length();

            String s="  "+id+" |"+name;
            while(size-- >0)
            {
                s=s+" ";
            }


            System.out.println(s+" | "+age+"  | "+marks);

        }

        con.close();
    }
```