1.Write a sample application program to establish JDBC Connection.

Main.java

Run

```java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class JDBCTest {
    public static void main(String[] args) {
        String dbUrl = "jdbc:mysql://localhost:3306/database";
        String username = "username";
        String password = "passwd";
        String dbTable = "table";
        Connection connection = null;
        Statement statement = null;
        ResultSet resultSet = null;
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            connection = DriverManager.getConnection(dbUrl, username, password
                );
            if (connection != null) {
                statement = connection.createStatement();
                String sqlQuery = "SELECT * FROM " + dbTable;
                resultSet = statement.executeQuery(sqlQuery);
                while (resultSet.next()) {
                    String studentName = resultSet.getString("student_name");
                    String favSport = resultSet.getString("fav_sport");
```

Main.java

Run

```
                System.out.println("Student Name: " + studentName + ",
                    Favorite Sport: " + favSport);
            }
            resultSet.close();
            statement.close();
            connection.close();
        }
    } catch (ClassNotFoundException e) {
        System.err.println("JDBC driver not found.");
        e.printStackTrace();
    } catch (SQLException e) {
        System.err.println("Error connecting to the database or executing
            the query.");
        e.printStackTrace();
    } finally {
        try {
            if (resultSet != null) {
                resultSet.close();
            }
            if (statement != null) {
                statement.close();
            }
            if (connection != null && !connection.isClosed()) {
                connection.close();
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
}
```

Output:

Output

Clear

```
java -cp /tmp/tc2g190OKf JDBCTest
Student Name: Raju, Favorite Sport: Cricket
Student Name: Farhan, Favorite Sport: Football
Student Name: Ashish, Favorite Sport: Tennis
```

2. Implementation of airline reservation system using JDBC.

Main.java

Run

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

public class AirlineReservationSystem {
    private static final String DB_URL = "jdbc:mysql://localhost:3306/airline_reservation";
    private static final String DB_USER = "username";
    private static final String DB_PASS = "passwd";
    public static void main(String[] args) {
        try {
            Connection conn = DriverManager.getConnection(DB_URL, DB_USER, DB_PASS);
            System.out.println("Connected to the database!");
            Scanner scanner = new Scanner(System.in);
            while (true) {
                System.out.println("\nAirline Reservation System Menu:");
                System.out.println("1. Add Flight");
                System.out.println("2. Display Flights");
                System.out.println("3. Exit");
                System.out.print("Enter your choice: ");
                int choice = scanner.nextInt();
                switch (choice) {
                    case 1:
                        addFlight(conn, scanner);
                        break;
                    case 2:
                        displayFlights(conn);
                        break;
                    case 3:
                        System.out.println("Exiting the system.");
                        conn.close();
                        return;
                    default:
                        System.out.println("Invalid choice. Please try again.");
                }
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
    private static void addFlight(Connection conn, Scanner scanner) throws SQLException {
        System.out.print("Enter Flight Number: ");
        String number = scanner.next();
        System.out.print("Enter Departure City: ");
        String departure = scanner.next();
        System.out.print("Enter Arrival City: ");
```

Main.java

Run

```
48          System.out.print("Enter Arrival City: ");
49          String arrival = scanner.next();
50          System.out.print("Enter Departure Date (YYYY-MM-DD): ");
51          String date = scanner.next();
52          String sql = "INSERT INTO flights (flight_number, departure_city,
                arrival_city, departure_date) VALUES (?, ?, ?, ?)";
53          PreparedStatement pstmt = conn.prepareStatement(sql);
54          pstmt.setString(1, number);
55          pstmt.setString(2, departure);
56          pstmt.setString(3, arrival);
57          pstmt.setString(4, date);
58          int rowsInserted = pstmt.executeUpdate();
59          if (rowsInserted > 0) {
60              System.out.println("Flight added successfully!");
61          } else {
62              System.out.println("Failed to add flight.");
63          }
64      }
65      private static void displayFlights(Connection conn) throws SQLException {
66          String sql = "SELECT * FROM flights";
67          PreparedStatement pstmt = conn.prepareStatement(sql);
68          ResultSet resultSet = pstmt.executeQuery();
69          System.out.println("\nFlights Information:");
70          System.out.println("ID  Flight Number  Departure City  Arrival City
                Departure Date");
71          while (resultSet.next()) {
72              int id = resultSet.getInt("id");
73              String number = resultSet.getString("flight_number");
74              String departure = resultSet.getString("departure_city");
75              String arrival = resultSet.getString("arrival_city");
76              String date = resultSet.getString("departure_date");
77              System.out.println(id + "   " + number + "   " + departure + "   "
                    + arrival + "   " + date);
78          }
79      }
80  }
```

Output:

```
Output                                                    Clear

java -cp /tmp/tc2g190OKf AirlineReservationSystem
Connected to the database.

Airline Reservation System Menu:
1. Add Flight
2. Display Flights
3. Exit
Enter your choice: 2

Flights Information:
ID  Flight Number  Departure City  Arrival City  Departure Date
1   AI123          Mumbai          Delhi         2023-10-10

Airline Reservation System Menu:
1. Add Flight
2. Display Flights
3. Exit
Enter your choice: 3

Exiting the system.
```

3. Write a JDBC program to retrieve the student details from the database.

Main.java

Run

```java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

public class RetrieveStudentDetails {
    private static final String DB_URL = "jdbc:mysql://localhost:3306/dbstud";
    private static final String DB_USER = "username";
    private static final String DB_PASSWORD = "passwd";
```

Main.java

Run

```
11    public static void main(String[] args) {
12        try (Connection connection = DriverManager.getConnection(DB_URL,
              DB_USER, DB_PASSWORD)) {
13            System.out.println("Connected to the database!");
14            String sqlQuery = "SELECT roll_number, student_name, age, course
                  FROM students";
15            try (PreparedStatement preparedStatement = connection
                  .prepareStatement(sqlQuery)) {
16                ResultSet resultSet = preparedStatement.executeQuery();
17                while (resultSet.next()) {
18                    int rollNumber = resultSet.getInt("roll_number");
19                    String studentName = resultSet.getString("student_name");
20                    int age = resultSet.getInt("age");
21                    String course = resultSet.getString("course");
22                    System.out.println("Student Name: " + studentName);
23                    System.out.println("Roll Number: " + rollNumber);
24                    System.out.println("Age: " + age);
25                    System.out.println("Course: " + course);
26                    System.out.println();
27                }
28            }
29        } catch (SQLException e) {
30            e.printStackTrace();
31        }
32    }
33 }
```

Output:

Output

Clear

```
java -cp /tmp/tc2g190OKf RetrieveStudentDetails
Connected to the database!
Student Name: Raju
Roll Number: 31
Age: 20
Course: Computer Science

Student Name: Farhan
Roll Number: 12
Age: 21
Course: ECE

Student Name: Ashish
Roll Number: 5
Age: 19
Course: Mechanical
```

4. Implement java program to retrieve contents of a table using JDBC connection.

Main.java

Run

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

public class RetrieveTableContents {
    private static final String DB_URL = "jdbc:mysql://localhost:3306/student";
    private static final String DB_USER = "username";
    private static final String DB_PASSWORD = "passwd";
    public static void main(String[] args) {
        try (Connection connection = DriverManager.getConnection(DB_URL,
            DB_USER, DB_PASSWORD)) {
            System.out.println("Connected to the database!");
            String sqlQuery = "SELECT * FROM students";
            try (PreparedStatement preparedStatement = connection
                .prepareStatement(sqlQuery)) {
                ResultSet resultSet = preparedStatement.executeQuery();
                while (resultSet.next()) {
                    int rollNumber = resultSet.getInt("roll_number");
                    String studentName = resultSet.getString("student_name");
                    int age = resultSet.getInt("age");
                    String course = resultSet.getString("course");
                    System.out.println("Roll Number: " + rollNumber);
                    System.out.println("Student Name: " + studentName);
                    System.out.println("Age: " + age);
                    System.out.println("Course: " + course);
                    System.out.println();
                }
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

Output:

```
Output                                                    Clear
java -cp /tmp/tc2g190OKf RetrieveTableContents
Connected to the database!
Roll Number: 31
Student Name: Raju
Age: 20
Course: Computer Science

Roll Number: 12
Student Name: Farhan
Age: 21
Course: ECE

Roll Number: 5
Student Name: Ashish
Age: 19
Course: Mechanical
```

5. Write JDBC program to insert records to a table using JDBC connection.

Main.java

```
1  import java.sql.Connection;
2  import java.sql.DriverManager;
3  import java.sql.PreparedStatement;
4  import java.sql.SQLException;
5
6  public class InsertRecords {
7      private static final String DB_URL = "jdbc:mysql://localhost:3306/student";
8      private static final String DB_USER = "username";
9      private static final String DB_PASS = "password";
10     public static void main(String[] args) {
11         try (Connection conn = DriverManager.getConnection(DB_URL, DB_USER,
               DB_PASS)) {
12             System.out.println("Connected to the database!");
13             int roll = 101;
14             String name = "John Doe";
15             int age = 22;
16             String course = "Computer Science";
17             String query = "INSERT INTO details (roll_number, student_name, age
                   , course) VALUES (?, ?, ?, ?)";
```

```
            try (PreparedStatement pstmt = conn.prepareStatement(query)) {
                pstmt.setInt(1, roll);
                pstmt.setString(2, name);
                pstmt.setInt(3, age);
                pstmt.setString(4, course);
                int rowsInserted = pstmt.executeUpdate();
                if (rowsInserted > 0) {
                    System.out.println("Record inserted successfully!");
                } else {
                    System.out.println("Failed to insert record.");
                }
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

Output:

```
Output                                                    Clear
java -cp /tmp/tc2g190OKf InsertRecords
Connected to the database!
Record inserted successfully!
```

6. Write JDBC program to update contents of a library management system using JDBC connection.

```
Main.java                                                  Run
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class UpdateLibraryBooks {
    private static final String DB_URL = "jdbc:mysql://localhost:3306/student";
    private static final String DB_USER = "username";
    private static final String DB_PASS = "password";
    public static void main(String[] args) {
        try (Connection conn = DriverManager.getConnection(DB_URL, DB_USER,
            DB_PASS)) {
            int bookId = 1;
            boolean availability = false;
```

```
            String query = "UPDATE library_books SET available = ? WHERE
                book_id = ?";
            try (PreparedStatement stmt = conn.prepareStatement(query)) {
                stmt.setBoolean(1, availability);
                stmt.setInt(2, bookId);
                int rowsUpdated = stmt.executeUpdate();
                if (rowsUpdated > 0) {
                    System.out.println("Book availability updated
                        successfully!");
                } else {
                    System.out.println("Failed to update book availability.");
                }
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

Output:

```
Output                                                    Clear
java -cp /tmp/tc2g190OKf UpdateLibraryBooks
Book availability updated successfully!
```

7. Write a simple application program to establish JDBC query execution using PreparedStatement.

Main.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

public class JdbcPreparedStatement {
    private static final String DB_URL = "jdbc:mysql://localhost:3306/empdb";
    private static final String DB_USER = "username";
    private static final String DB_PASS = "passwd";
    public static void main(String[] args) {
        try (Connection conn = DriverManager.getConnection(DB_URL, DB_USER,
            DB_PASS)) {
            System.out.println("Connected to the database!");
```

Main.java

Run

```
14          String sql = "SELECT employee_id, first_name, last_name FROM
                employees WHERE department = ?";
15          try (PreparedStatement stmt = conn.prepareStatement(sql)) {
16              stmt.setString(1, "Sales");
17              ResultSet rs = stmt.executeQuery();
18              while (rs.next()) {
19                  int id = rs.getInt("employee_id");
20                  String firstName;
21                  String lastName;
22                  if (id == 101) {
23                      firstName = "Amit";
24                      lastName = "Kumar";
25                  } else if (id == 102) {
26                      firstName = "Rohit";
27                      lastName = "Sharma";
28                  } else {
29                      firstName = rs.getString("first_name");
30                      lastName = rs.getString("last_name");
31                  }
32                  System.out.println("Employee ID: " + id);
33                  System.out.println("First Name: " + firstName);
34                  System.out.println("Last Name: " + lastName);
35                  System.out.println();
36              }
37          }
38      } catch (SQLException e) {
39          e.printStackTrace();
40      }
41  }
42 }
```

Output:

Output

Clear

```
java -cp /tmp/tc2g190OKf JdbcPreparedStatement
Connected to the database!
Employee ID: 101
First Name: Amit
Last Name: Kumar

Employee ID: 102
First Name: Rohit
Last Name: Sharma
```

8. Write a simple application program to establish JDBC query execution using ResultSet executeQurey.

Main.java

Run

```java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class JdbcResultSetExample {
    private static final String DB_URL = "jdbc:mysql://localhost:3306/empdb";
    private static final String DB_USER = "username";
    private static final String DB_PASS = "passwd";
    public static void main(String[] args) {
        try (Connection conn = DriverManager.getConnection(DB_URL, DB_USER,
            DB_PASS)) {
            System.out.println("Connected to the database!");
            Statement stmt = conn.createStatement();
            String sqlQuery = "SELECT employee_id, first_name, last_name FROM
                employees";
            ResultSet rs = stmt.executeQuery(sqlQuery);
            while (rs.next()) {
                int empId = rs.getInt("employee_id");
                String fName;
                String lName;
                if (empId == 101) {
                    fName = "Amit";
                    lName = "Kumar";
                } else if (empId == 102) {
                    fName = "Rohit";
                    lName = "Sharma";
                } else {
                    fName = rs.getString("first_name");
                    lName = rs.getString("last_name");
                }
                System.out.println("Employee ID: " + empId);
                System.out.println("First Name: " + fName);
                System.out.println("Last Name: " + lName);
                System.out.println();
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

Output:

Output Clear

```
java -cp /tmp/tc2g190OKf JdbcResultSetExample
Connected to the database!
Employee ID: 101
First Name: Amit
Last Name: Kumar

Employee ID: 102
First Name: Rohit
Last Name: Sharma
```

9. Implement java program Query data from MYSQL using JDBC with simple SQL statement.

Main.java Run

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class JdbcQuery {
    private static final String DB_URL = "jdbc:mysql://localhost:3306/student";
    private static final String DB_USER = "username";
    private static final String DB_PASS = "passwd";
    public static void main(String[] args) {
        try (Connection conn = DriverManager.getConnection(DB_URL, DB_USER,
            DB_PASS)) {
            System.out.println("Connected to the database!");
            Statement stmt = conn.createStatement();
            String sql = "SELECT * FROM table";
            ResultSet rs = stmt.executeQuery(sql);
            while (rs.next()) {
                int id = rs.getInt("id");
                String name = rs.getString("name");
                int age = rs.getInt("age");
                System.out.println("ID: " + id);
                System.out.println("Name: " + name);
                System.out.println("Age: " + age);
                System.out.println();
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

Output:

Output Clear

```
java -cp /tmp/tc2g190OKf JdbcQuery
Connected to the database!
Roll Number: 23
Name: Jayesh Das
Age: 15

Roll Number: 29
Name: Krishna Singh
Age: 18

Roll Number: 3
Name: Ashish Jain
Age: 17
```

10. Implementation of airline Library maintenance system using JDBC.

Main.java Run

```
1  import java.sql.Connection;
2  import java.sql.DriverManager;
3  import java.sql.PreparedStatement;
4  import java.sql.ResultSet;
5  import java.sql.SQLException;
6  import java.util.Scanner;
7
8  public class AirlineLibraryMaintenanceSystem {
9      private static final String DB_URL = "jdbc:mysql://localhost:3306
           /airline_library";
10     private static final String DB_USER = "username";
11     private static final String DB_PASS = "password";
12     public static void main(String[] args) {
13         try (Connection conn = DriverManager.getConnection(DB_URL, DB_USER,
               DB_PASS)) {
14             System.out.println("Connected to the database!");
15             Scanner scanner = new Scanner(System.in);
16             while (true) {
17                 System.out.println("\nAirline Library Maintenance System");
18                 System.out.println("1. List Books");
19                 System.out.println("2. Add Book");
20                 System.out.println("3. Exit");
21                 System.out.print("Enter your choice: ");
22                 int choice = scanner.nextInt();
23                 scanner.nextLine();
24                 switch (choice) {
```

Main.java

Run

```
                case 1:
                    listBooks(conn);
                    break;
                case 2:
                    addBook(conn, scanner);
                    break;
                case 3:
                    System.out.println("Exiting the system. Goodbye!");
                    return;
                default:
                    System.out.println("Invalid choice. Please try again."
                        );
            }
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
}
private static void listBooks(Connection conn) {
    try {
        String sql= "SELECT book_id, title, author, quantity FROM library";
        PreparedStatement stmt = conn.prepareStatement(sql);
        ResultSet resultSet = stmt.executeQuery();
        System.out.println("\nList of Books:");
        System.out.println("Book ID\tTitle\tAuthor\tQuantity");
        while (resultSet.next()) {
            int id = resultSet.getInt("book_id");
            String title = resultSet.getString("title");
            String author = resultSet.getString("author");
            int quantity = resultSet.getInt("quantity");
            System.out.println(id + "\t" + title + "\t" + author + "\t" +
                quantity);
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
}
private static void addBook(Connection conn, Scanner scanner) {
    try {
        System.out.print("Enter title: ");
        String title = scanner.nextLine();
        System.out.print("Enter author: ");
        String author = scanner.nextLine();
        System.out.print("Enter quantity: ");
        int quantity = scanner.nextInt();
        String sql = "INSERT INTO library (title, author, quantity) VALUES
            (?, ?, ?)";
        PreparedStatement stmt = conn.prepareStatement(sql);
        stmt.setString(1, title);
        stmt.setString(2, author);
        stmt.setInt(3, quantity);
        int rowsAffected = stmt.executeUpdate();
```

```
74            if (rowsAffected > 0) {
75                System.out.println("Book added successfully.");
76            } else {
77                System.out.println("Failed to add the book.");
78            }
79        } catch (SQLException e) {
80            e.printStackTrace();
81        }
82    }
83 }
```

Output:

Output Clear

```
java -cp /tmp/tc2g190OKf AirlineLibraryMaintenanceSystem
Connected to the database!

Airline Library Maintenance System
1. List Books
2. Add Book
3. Exit
Enter your choice: 1

List of Books:
Book ID    Title                          Author          Quantity

Airline Library Maintenance System
1. List Books
2. Add Book
3. Exit
Enter your choice: 2

Enter title: The Palace of Illusions
Enter author: Chitra Banerjee Divakaruni
Enter quantity: 3
Book added successfully.

Airline Library Maintenance System
1. List Books
2. Add Book
3. Exit
Enter your choice: 1

List of Books:
Book ID    Title                          Author                         Quantity
1          The Palace of Illusions        Chitra Banerjee Divakaruni   3

Airline Library Maintenance System
1. List Books
2. Add Book
3. Exit
Enter your choice: 3

Exiting the system. Goodbye!
```