# AMITY UNIVERSITY, PATNA

# AMITY INSTITUTE OF INFORMATION TECHNOLOGY

## Advanced Java Lab File
## LAB - 1

## BCA



Name : Aryan Amar
Program/Semester : BCA – 6 'A'
Enroll. Number : A45304821047
Submitted to : Dr. Naveen Kumar Singh

# CRUD OPERATIONS

**Problem Statement :** Connecting JDBC to Eclipse for CRUD Operations with Menu Options, provided that every operation should be performed in the same class.

**Objective**
The objective of this project is to create a Java program in Eclipse that connects to a database using JDBC (Java Database Connectivity) and allows users to perform CRUD (Create, Read, Update, Delete) operations on an IPL_TEAMS table. The program will present users with a menu containing seven options: insertion, read, update team's name, update team's captain, update team's coach, deletion of a record, and exit. Users will be able to interact with the program through the console to manipulate the team's records stored in the database.

**Detailed Requirements :**

**1. Database Connectivity -** The program must establish a connection to a relational database management system (RDBMS) using JDBC. Database connection parameters such as URL, username, and password should be configurable and provided as constants or properties in the program.

**2. Menu Options -** The program should display a menu with options like Insert, Read, Update team's name, Update team's captain, Update team's coach, Deletion and Exit option. Users will input the corresponding option number to execute the desired operation.

**3. Insert Operation** - Upon selecting the insert option, users will be prompted to input the team's ID, name, city, captain, and coach through the keyboard. The program will then insert this data into the IPL_TEAMS table in the database.

**4. Read Operation** -  Selecting the read option will display all records from the IPL_TEAMS table. The program should fetch and display the team's ID, name, city, captain, and coach for each record in the table.

**5. Update Operation** - The program should provide separate options to update the team's name, captain, and coach. Upon selecting any of these options, users will input the team ID whose details need to be updated, followed by the new value for the corresponding field. The program will then update the specified field for the given team ID in the database.

**6. Delete Operation** - Selecting the delete option will prompt users to input the team ID of the record they want to delete. The program will delete the record with the specified team ID from the IPL_TEAMS table in the database.

**7. Error Handling** - The program should handle exceptions gracefully, displaying meaningful error messages to users in case of connection failures, SQL errors, or invalid input. It should also validate user inputs to prevent SQL injection attacks or database errors due to incorrect data formats.

**8. Exit** - Upon selecting the exit option, the program will terminate gracefully, closing any open resources and releasing database connections.

**Tools and Technologies used** : Eclipse IDE and Oracle Database 21C Express Edition

**Design**

**Table description -**

```
SQL> desc IPL_TEAMS;
 Name                                     Null?    Type
 ---------------------------------------- -------- ------------------------------
 TEAM_ID                                  NOT NULL NUMBER
 TEAM_NAME                                         VARCHAR2(100)
 CITY                                              VARCHAR2(50)
 CAPTAIN                                           VARCHAR2(50)
 COACH                                             VARCHAR2(50)
```
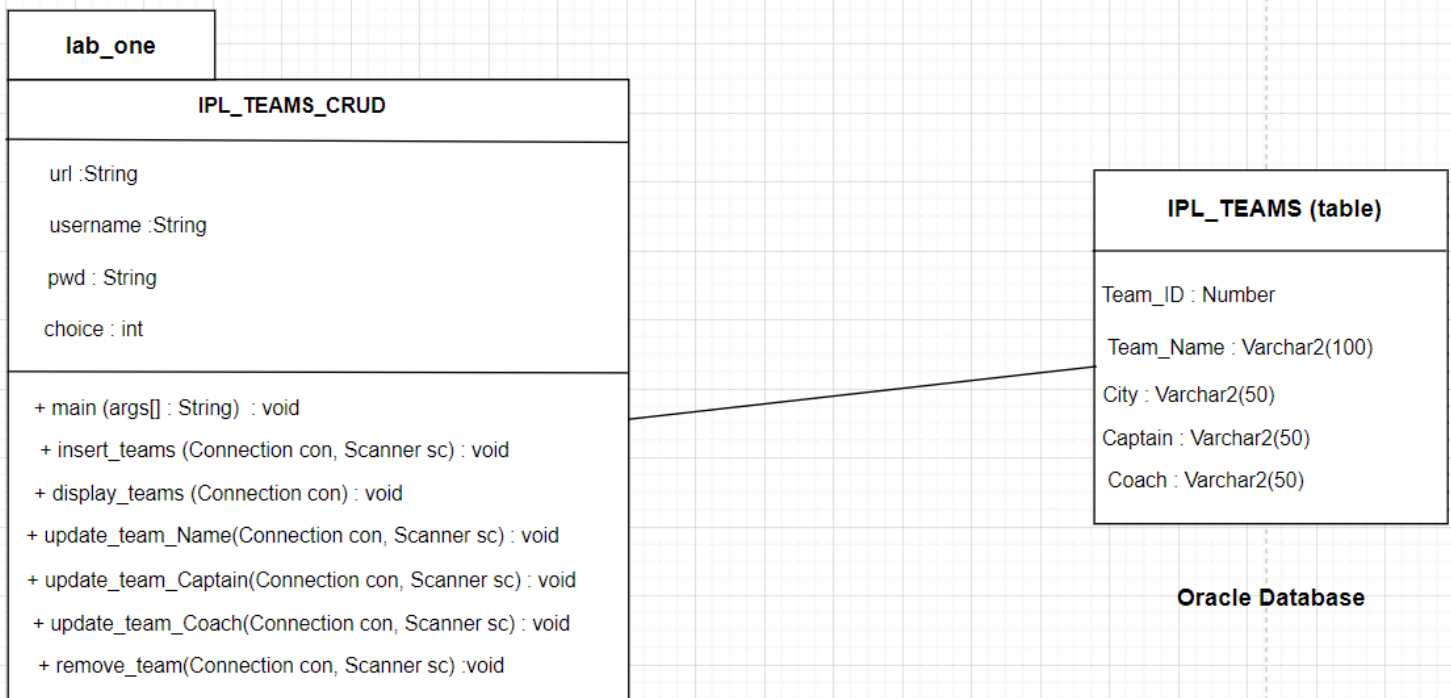
**Folder Structure -**

```
∨ 📁 lab_one
  > ➔ JRE System Library [JavaSE-21]
  ∨ 📁 src
    ∨ ⊞ lab_one
      > 🗎 IPL_TEAMS_CRUD.java
    > 🗎 module-info.java
  ∨ ➔ Referenced Libraries
    > 📦 ojdbc11.jar - C:\app\aryan\product\21c\dbhomeXE\jdbc\lib
```

**Class Diagram** – The class digram for our project will be as follows :

**lab_one**

**IPL_TEAMS_CRUD**

url :String

username :String

pwd : String

choice : int

+ main (args[] : String)  : void

 + insert_teams (Connection con, Scanner sc) : void

 + display_teams (Connection con) : void

+ update_team_Name(Connection con, Scanner sc) : void

+ update_team_Captain(Connection con, Scanner sc) : void

+ update_team_Coach(Connection con, Scanner sc) : void

 + remove_team(Connection con, Scanner sc) :void

**IPL_TEAMS (table)**

Team_ID : Number

 Team_Name : Varchar2(100)

City : Varchar2(50)

Captain : Varchar2(50)

 Coach : Varchar2(50)

**Oracle Database**

# CODE

**IPL_TEAMS_CRUD.java**

```java
package lab_one;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Scanner;

public class IPL_TEAMS_CRUD {

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

        Class.forName("oracle.jdbc.driver.OracleDriver");

        String url = "jdbc:oracle:thin:@localhost:1521:xe";
        String username = "SYSTEM";
        String pwd = "aryan123";

        Connection con = DriverManager.getConnection(url,
username, pwd);
        Scanner sc = new Scanner(System.in);
        IPL_TEAMS_CRUD ipl = new IPL_TEAMS_CRUD();
        while (true) {
            System.out.println("\nChoose the operation you
want to perform :");
```

# CODE

```java
                System.out.println("1. Perform Insert
operation");

                System.out.println("2. Perform Read operation");

                System.out.println("3. Update Team's name");

                System.out.println("4. Update Team's captain");

                System.out.println("5. Update Team's coach");

                System.out.println("6. Perform Delete
operation");

                System.out.println("7. Exit");

                System.out.println("\nPlease enter your choice
:");

                int choice = sc.nextInt();

                switch (choice) {
                case 1:
                    System.out.println("\nPerforming INSERT
operation......\n");

                    ipl.insert_teams(con, sc);

                    break;
                case 2:
                    System.out.println("\nPerforming READ
operation......\n");

                    ipl.display_teams(con);

                    break;
                case 3:
                    System.out.println("\nUpdating Team's
name......\n");

                    ipl.update_team_Name(con, sc);

                    break;
                case 4:
```

```
                    System.out.println("\nUpdating Team's
captain......\n");

                    ipl.update_team_Captain(con, sc);

                    break;

              case 5:

                    System.out.println("\nUpdating Team's
coach......\n");

                    ipl.update_team_Coach(con, sc);

                    break;

              case 6:

                    System.out.println("\nPerforming DELETE
operation......\n");

                    ipl.remove_team(con, sc);

                    break;

              case 7:

                    System.out.println("\nExiting......\n");

                    sc.close();

                    System.exit(0);

                    break;

              default:

                    System.out.println("\nInvalid choice\n");

              }


         }


    public void insert_teams(Connection con, Scanner sc) throws
SQLException
```

# CODE

```java
    {
        Statement st = con.createStatement();
        System.out.println("Enter the team's Id: ");
        int id = sc.nextInt();


        System.out.println("Enter the team's name: ");
        String name = sc.next();


        System.out.println("Enter the team's city: ");
        String city = sc.next();


        System.out.println("Enter the team's captain: ");
        String captain = sc.next();


        System.out.println("Enter the team's coach: ");
        String coach = sc.next();


        String query = String.format("insert into IPL_TEAMS
values(%d, '%s', '%s', '%s', '%s')", id, name, city,
                captain, coach);


        int rowsAffected = st.executeUpdate(query);
        System.out.println();
        System.out.println(rowsAffected + " record
inserted!!!");
    }


    public void display_teams(Connection con) throws
SQLException {
```

# CODE

```java
        Statement st = con.createStatement();

        ResultSet rs = st.executeQuery("select * from
IPL_TEAMS");

        while (rs.next()) {

              System.out.println(rs.getInt(1) + "   " +
rs.getString(2) + "   " + rs.getString(3) + "   " +
rs.getString(4)

                         + "   " + rs.getString(5));

        }

    }


    public void update_team_Name(Connection con, Scanner sc)
throws SQLException {

        Statement st = con.createStatement();

        System.out.println("Enter the team's id: ");

        int id = sc.nextInt();

        System.out.println("Enter the team's new name: ");

        String name = sc.next();

        String query = String.format("update IPL_TEAMS set
Team_Name='%s' where Team_ID = %d", name, id);

        int rowsAffected = st.executeUpdate(query);

        System.out.println();

        System.out.println(rowsAffected + " record
updated!!!");

    }


    public void update_team_Captain(Connection con, Scanner sc)
throws SQLException {

        Statement st = con.createStatement();

        System.out.println("Enter the team's id: ");

        int id = sc.nextInt();
```

# CODE

```java
        System.out.println("Enter the team's new captain: ");

        String captain = sc.next();

        String query = String.format("update IPL_TEAMS set
Captain='%s' where Team_ID = %d", captain, id);

        int rowsAffected = st.executeUpdate(query);

        System.out.println();

        System.out.println(rowsAffected + " record
updated!!!");

    }


    public void update_team_Coach(Connection con, Scanner sc)
throws SQLException {

        Statement st = con.createStatement();

        System.out.println("Enter the team's id: ");

        int id = sc.nextInt();

        System.out.println("Enter the team's new coach: ");

        String coach = sc.next();

        String query = String.format("update IPL_TEAMS set
Coach='%s' where Team_ID = %d", coach, id);

        int rowsAffected = st.executeUpdate(query);

        System.out.println();

        System.out.println(rowsAffected + " record
updated!!!");

    }


    public void remove_team(Connection con, Scanner sc) throws
SQLException {

        Statement st = con.createStatement();

        System.out.println("Enter the team's id: ");

        int id = sc.nextInt();
```

# CODE

```
        int rowsAffected = st.executeUpdate("delete from
IPL_TEAMS where Team_ID = " + id);

        System.out.println();

        System.out.println(rowsAffected + " record
deleted!!!");

    }


}
```

# INPUT/OUTPUT

## Insertion of a couple of rows in the table

```
Problems  Servers  Terminal  Data Source Explorer  Properties  Console ×  Progress  Error Log
IPL_TEAMS_CRUD [Java Application] C:\Users\aryan\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64

Choose the operation you want to perform :
1. Perform Insert operation
2. Perform Read operation
3. Update Team's name
4. Update Team's captain
5. Update Team's coach
6. Perform Delete operation
7. Exit

Please enter your choice :
1

Performing INSERT operation......

Enter the team's Id:
121
Enter the team's name:
RCB
Enter the team's city:
Bangalore
Enter the team's captain:
Faf_Du_Plessis
Enter the team's coach:
Sanjay_Bangar

1 record inserted!!!
```

```
Choose the operation you want to perform :
1. Perform Insert operation
2. Perform Read operation
3. Update Team's name
4. Update Team's captain
5. Update Team's coach
6. Perform Delete operation
7. Exit

Please enter your choice :
1

Performing INSERT operation......

Enter the team's Id:
220
Enter the team's name:
MI
Enter the team's city:
Mumbai
Enter the team's captain:
Rohit_Sharma
Enter the team's coach:
Mahela_Jayawardene

1 record inserted!!!
```

# INPUT/OUTPUT

## Performing read/retrieve operation on the table

```
Problems  Servers  Terminal  Data Source Explorer  Properties  Console ×  Progress  Error Log
IPL_TEAMS_CRUD [Java Application] C:\Users\aryan\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64

Choose the operation you want to perform :
1. Perform Insert operation
2. Perform Read operation
3. Update Team's name
4. Update Team's captain
5. Update Team's coach
6. Perform Delete operation
7. Exit

Please enter your choice :
2

Performing READ operation......

121  RCB  Bangalore  Faf_Du_Plessis  Sanjay_Bangar
220  MI  Mumbai  Rohit_Sharma  Mahela_Jayawardene
```

## Updating the name of the team

```
Problems  Servers  Terminal  Data Source Explorer  Properties  Console ×  Progress  Error Log
IPL_TEAMS_CRUD [Java Application] C:\Users\aryan\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64
Please enter your choice :
3

Updating Team's name......

Enter the team's id:
121
Enter the team's new name:
Royal_Challengers_Bangalore

1 record updated!!!

Choose the operation you want to perform :
1. Perform Insert operation
2. Perform Read operation
3. Update Team's name
4. Update Team's captain
5. Update Team's coach
6. Perform Delete operation
7. Exit

Please enter your choice :
2

Performing READ operation......

121  Royal_Challengers_Bangalore  Bangalore  Faf_Du_Plessis  Sanjay_Bangar
220  MI  Mumbai  Rohit_Sharma  Mahela_Jayawardene
```

# INPUT/OUTPUT

**<u>Updating the captain of the team</u>**

```
Problems    Servers    Terminal    Data Source Explorer    Properties    Console ×    Progress    Error Log
IPL_TEAMS_CRUD [Java Application] C:\Users\aryan\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.
Please enter your choice :
4

Updating Team's captain......

Enter the team's id:
121
Enter the team's new captain:
Virat_Kohli

1 record updated!!!

Choose the operation you want to perform :
1. Perform Insert operation
2. Perform Read operation
3. Update Team's name
4. Update Team's captain
5. Update Team's coach
6. Perform Delete operation
7. Exit

Please enter your choice :
2

Performing READ operation......

121   Royal_Challengers_Bangalore   Bangalore   Virat_Kohli   Sanjay_Bangar
220   MI   Mumbai   Rohit_Sharma   Mahela_Jayawardene
```

# INPUT/OUTPUT

**Updating the coach of the team**

```
Problems  Servers  Terminal  Data Source Explorer  Properties  Console ×  Progress  Error Log
IPL_TEAMS_CRUD [Java Application] C:\Users\aryan\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.
Please enter your choice :
5

Updating Team's coach......

Enter the team's id:
121
Enter the team's new coach:
Andy_Flower

1 record updated!!!

Choose the operation you want to perform :
1. Perform Insert operation
2. Perform Read operation
3. Update Team's name
4. Update Team's captain
5. Update Team's coach
6. Perform Delete operation
7. Exit

Please enter your choice :
2

Performing READ operation......

121  Royal_Challengers_Bangalore  Bangalore  Virat_Kohli  Andy_Flower
220  MI  Mumbai  Rohit_Sharma  Mahela_Jayawardene
```

**Updating the coach of the team**

# INPUT/OUTPUT

**Removing a record from the table**

```
Problems   Servers   Terminal   Data Source Explorer   Properties   Console ×   Progress   Error Log
IPL_TEAMS_CRUD [Java Application] C:\Users\aryan\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_2
Please enter your choice :
6

Performing DELETE operation......

Enter the team's id:
220

1 record deleted!!!

Choose the operation you want to perform :
1. Perform Insert operation
2. Perform Read operation
3. Update Team's name
4. Update Team's captain
5. Update Team's coach
6. Perform Delete operation
7. Exit

Please enter your choice :
2

Performing READ operation......

121  Royal_Challengers_Bangalore  Bangalore  Virat_Kohli  Andy_Flower
```

**Removing a record from the table**