

**JAVA AWT BASED-  
HOCKEY CLUB EVENTS ACTIVITY TRACKER  
- SQL CONNECTIVITY USING JDBC**

*A Mini Project Report*

*Submitted in partial fulfilment of the  
Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

**IN**

**INFORMATION TECHNOLOGY**

**By**

**TASNEEM BEGUM (1602-18-737-112)**

**UNDER THE GUIDANCE OF B.LEELAVATHY**



**Department of Information Technology**

**Vasavi College of Engineering (Autonomous)**

**(Affiliated to Osmania University)**

**Ibrahim Bagh, Hyderabad-31**

**2020**

# BONAFIDE CERTIFICATE

This is to Certify that the project report titled **“HOCKEY CLUB EVENTS ACTIVITY TRACKER”** project work of Miss. TASNEEM BEGUM bearing Roll.no:1602-18-737-112 who carried out this project under my supervision in the IV semester for the academic year 2019-2020.

Signature

External examine

Signature

Internal examine

# ABSTRACT

A sports club will organize several events with different player's participation in the events, on different dates and at different locations. They need to maintain record of all this events and different players participated. This database project helps to keep the record of hockey club events and participations of players in those events. A new user can easily track the store details of different events conducted at different locations and players participated it that events by entering his/her details into the database.

## a) Aim and Priority of the project:

To create a **Java GUI based “Hockey Club Event Activity Tracker”** which has the entities like: Admin, Events and Player whose values are taken from the user. These values are to be inserted, updated and deleted in the database using **JDBC connectivity**.

## Introduction:

The purpose of this project is to provide an easy to use application for storing various activity of a club with consistence and later track them with ease. For an admin using this application has to enter his/her admin\_id and directly can insert, update or delete any record without any ambiguity. Later, user can track events effectively as the application works efficiently.

## b) Requirements:

Tables required-5

TABLE NAME	ATTRIBUTES	DOMAIN
Admin	i)u_id ii)fname iii)lname	number(5) varchar2(20) varchar2(20)
Events	i)e_id ii)location iii)e_date	number(5) varchar2(20) DATE
Tracks	i)u_id ii)e_id iii)time	number(5) number (5) varchar2(20)
Player	i)p_id ii)fname iii)lname iv)position v)rank	number (5) varchar2(20) varchar2(20) varchar2(20) number (5)
Participate	i)e_id ii)p_id	number (5) number (5)

## c) Architecture and Technology:

### Software used :

- i. Java Eclipse
- ii. Oracle 11g Database Enterprise Edition Release 11.2.0.1.0-64 bit  
SQL\*Plus Release 9.0.1.3.0.

### Java AWT:

**Java AWT** (Abstract Window Toolkit) is *an API to develop GUI or window-based applications* in java.

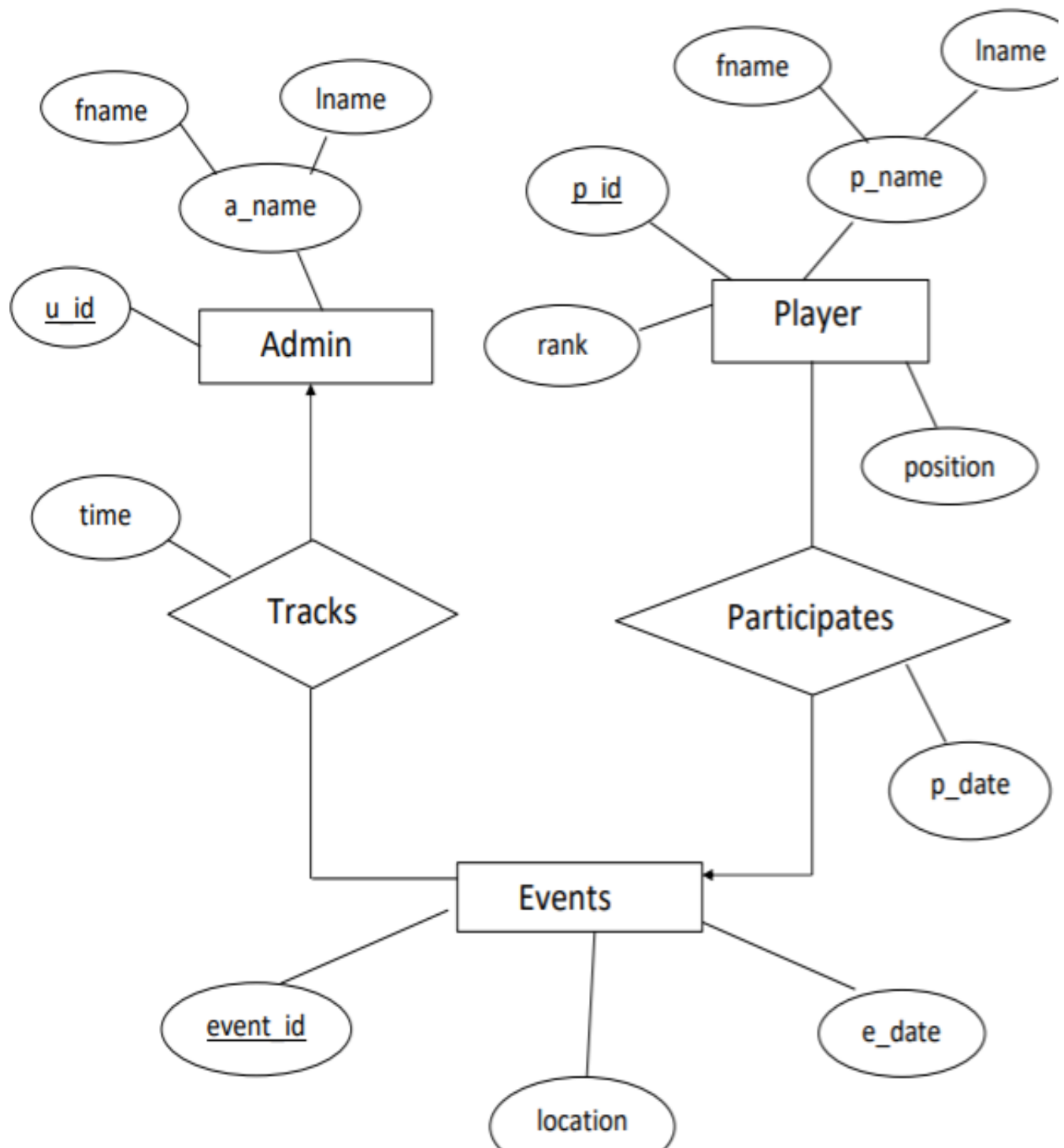
Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system. AWT is heavyweight i.e. its components are using the resources of OS. The java.awt package provides classes such as TextField, Label, TextArea, Button, Choice, List ,etc.

### SQL:

Structure Query Language(SQL) is a database query language used for storing and managing data in Relational DBMS. SQL was the first commercial language introduced for E.F Codd's Relational model of database. Today almost all RDBMS (MySQL, Oracle, Infomix, Sybase, MS Access) use SQL as the standard database query language. SQL is used to perform all types of data operations in RDBMS.

## d) Design:

### i) Entity Relationship Model:



## ii) Database Design:

Creation of tables using DDL commands as follows:

### 1. Admin Table:

```
SQL> create table admin(
  2  u_id number(5) primary key,
  3  fname varchar2(20),
  4  lname varchar2(20));

Table created.

SQL> desc admin;
Name                               Null?    Type
-----
U_ID                                NOT NULL NUMBER(5)
FNAME                              VARCHAR2(20)
LNAME                              VARCHAR2(20)

SQL>
```

### 2.Event Table:

```
SQL> create table events(
  2  e_id number(5) primary key,
  3  location varchar2(20),
  4  e_date DATE);

Table created.

SQL> desc events;
Name                               Null?    Type
-----
E_ID                                NOT NULL NUMBER(5)
LOCATION                              VARCHAR2(20)
E_DATE                              DATE

SQL>
```

### 3.Tracks Relationship Table:

```
SQL> create table tracks(
  2  u_id number(5),
  3  e_id number(5),
  4  time varchar2(20));

Table created.

SQL> alter table tracks add foreign key(u_id) references admin;

Table altered.

SQL> alter table tracks add foreign key(e_id) references events;

Table altered.

SQL> alter table tracks add primary key(u_id,e_id);

Table altered.

SQL> desc tracks;
Name                               Null?    Type
-----
U_ID                                NOT NULL NUMBER(5)
E_ID                                NOT NULL NUMBER(5)
TIME                              VARCHAR2(20)
```

### 4. Player Table:

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

```
SQL> create table player(
2  p_id number(5) primary key,
3  pfname varchar2(20),
4  plname varchar2(20),
5  position varchar2(20),
6  rank number(5));
```

Table created.

```
SQL> desc player;
```

Name	Null?	Type
P_ID	NOT NULL	NUMBER(5)
PFNAME		VARCHAR2(20)
PLNAME		VARCHAR2(20)
POSITION		VARCHAR2(20)
RANK		NUMBER(5)

```
SQL>
```

## 5. Participation Relationship Table:

```
SQL> create table participate(
2  e_id number(5),
3  p_id number(5),
4  date DATE);
```

```
date DATE)
```

```
SQL> alter table participate add foreign key(e_id) references events;
```

Table altered.

```
SQL> alter table participate add foreign key(p_id) references player;
```

Table altered.

```
SQL> alter table participate add primary key(e_id,p_id);
```

Table altered.

```
SQL> desc participate;
```

Name	Null?	Type
E_ID	NOT NULL	NUMBER(5)
P_ID	NOT NULL	NUMBER(5)
P_DATE		DATE

```
SQL>
```

## Inserting records using DML operation :

### 1.Admin Table:

```
SQL> insert into admin values(101,'Bihar','11-FEB-2020');
```

1 row created.



```
SQL> select * from admin;
```

U_ID	FNAME	LNAME
1	Varsha	Kumari
2	sri	raj
3	sana	fatima
4	abdul	wahed
5	abdul	majed

```
SQL>
```

## 2.Events Table:

```
SQL> select * from events;
```

E_ID	LOCATION	E_DATE
101	Bihar	11-FEB-20
102	lucknow	20-FEB-20
103	Delhi	24-MAR-20
104	Hyderabad	19-AUG-20
105	Mumbai	13-MAY-20

```
SQL>
```

## 3. Tracks Relationship Table:

```
SQL> select * from tracks;
```

U_ID	E_ID	TIME
1	101	2:00pm
2	102	10:00am
3	103	8:30pm
4	104	5:34pm
5	105	9:09am

#### 4. Player Table:

```
SQL> select * from player;
```

P_ID	PFNAME	PLNAME	POSITION
17	Dhyan	Chand	forwards
28	Dhanraj	Pillay	midfields
36	Udhan	Singh	goalie

P_ID	PFNAME	PLNAME	POSITION
50	Luslie	Claudis	fullbacks
73	Mohammed	Shahid	fullbacks

```
SQL>
```

#### 5. Participation Relationship Table:

```
SQL> select * from participate;
```

E_ID	P_ID	P_DATE
101	17	09-FEB-20
102	28	16-FEB-20
103	36	21-MAR-20
104	50	18-AUG-20
105	73	10-MAY-20

### e) Implementation

i) Front end programs and connectivity

//CONNECTIVITY PROGRAM:

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
import java.sql.*;

public class OracleConnection
{
    public static void main(String args[])
    {
        try
        {
            //step1 load the driver class
            Class.forName("oracle.jdbc.driver.OracleDriver"); // or Class.forName("oracle.jdbc.OracleDriver");

            //step2 create the connection object
            Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:tasneem","tasneem","tasneem");

            //step3 create the statement object
            Statement stmt=con.createStatement();

            //step4 close the connection object
            con.close();

        }catch(Exception e){ System.out.println(e);}
    }
}
```

Thus, the connection from Java to Oracle database is performed and therefore, can be used for inserting, updating and deleting tables in the database directly.

Programs:

### 1.Inserting in Admin Table:

```
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
import javax.swing.JOptionPane;
public class InsertAdmin extends Frame
{
    Button insertAdminButton;
    TextField u_idText, fnameText, lnameText;
    TextArea errorText;
    Connection connection;
    Statement statement;
    public InsertAdmin()
    {
        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
    }
}
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
        catch (Exception e)
        {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        connectToDB();
    }

    public void connectToDB()
    {
        try
        {
            connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:tasneem","tasneem","tasneem")
;
            statement = connection.createStatement();

        }
        catch (SQLException connectException)
        {
            System.out.println(connectException.getMessage());
            System.out.println(connectException.getSQLState());
            System.out.println(connectException.getErrorCode());
            System.exit(1);
        }
    }

    public void buildGUI()
    {
        insertAdminButton = new Button("Insert Admin");
        insertAdminButton.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                try
                {
                    String query= "INSERT INTO admin VALUES(" +
u_idText.getText() + ", " + " " + "'" + fnameText.getText() + "'" + ", "+"'" + lnameText.getText()
+"'" + " )";

                    int i = statement.executeUpdate(query);
                    errorText.append("\nInserted " + i + " rows successfully");
                }
                catch (SQLException insertException)
                {
                    displaySQLErrors(insertException);
                    JOptionPane.showMessageDialog(null,"Input is in wrong
format.", "Error Message",JOptionPane.ERROR_MESSAGE);
                }
            }
        });

        u_idText = new TextField(15);
        fnameText = new TextField(15);
        lnameText = new TextField(15);

        errorText = new TextArea(10, 40);
        errorText.setEditable(false);

        Panel first = new Panel();
        first.setLayout(new GridLayout(4, 2));
        first.add(new Label("Admin/User ID:"));
        first.add(u_idText);
        first.add(new Label("First Name:"));
        first.add(fnameText);
    }
}
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
        first.add(new Label("Last Name:"));
        first.add(lnameText);
        first.setBounds(125,90,200,100);

        Panel second = new Panel(new GridLayout(4, 1));
        second.add(insertAdminButton);
        second.setBounds(125,220,150,100);

        Panel third = new Panel();
        third.add(errorText);
        third.setBounds(125,320,300,200);

        setLayout(null);

        add(first);
        add(second);
        add(third);

        setTitle("New Admin Creation");
        setSize(500, 600);
        setVisible(true);
    }

    private void displaySQLExceptions(SQLException e)
    {
        errorText.append("\nSQLException: " + e.getMessage() + "\n");
        errorText.append("SQLState: " + e.getSQLState() + "\n");
        errorText.append("VendorError: " + e.getErrorCode() + "\n");
    }

    public static void main(String[] args)
    {
        InsertAdmin sail = new InsertAdmin();

        sail.addWindowListener(new WindowAdapter(){
            public void windowClosing(WindowEvent e)
            {
                System.exit(0);
            }
        });

        sail.buildGUI();
    }
}
```

## 2. Updating Admin Table:

```
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

import javax.swing.JOptionPane;
public class UpdateAdmin extends Frame
{
    Button updateAdminButton;
    List adminIDList;
    TextField u_idText, fnameText, lnameText;
    TextArea errorText;
    Connection connection;
    Statement statement;
    ResultSet rs;
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
public UpdateAdmin()
{
    try
    {
        Class.forName("oracle.jdbc.driver.OracleDriver");
    }
    catch (Exception e)
    {
        System.err.println("Unable to find and load driver");
        System.exit(1);
    }
    connectToDB();
}

public void connectToDB()
{
    try
    {
        connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:tasneem", "tasneem", "tasneem")
;
        statement = connection.createStatement();
    }
    catch (SQLException connectException)
    {
        System.out.println(connectException.getMessage());
        System.out.println(connectException.getSQLState());
        System.out.println(connectException.getErrorCode());
        System.exit(1);
    }
}

private void loadAdmin()
{
    try
    {
        rs = statement.executeQuery("SELECT u_id FROM admin");
        while (rs.next())
        {
            adminIDList.add(rs.getString("u_id"));
        }
    }
    catch (SQLException e)
    {
        displaySQLErrors(e);
    }
}

public void buildGUI()
{
    adminIDList = new List(10);
    loadAdmin();
    add(adminIDList);

    adminIDList.addItemListener(new ItemListener()
    {
        public void itemStateChanged(ItemEvent e)
        {
            try
            {
                rs = statement.executeQuery("SELECT * FROM admin where
u_id =" + adminIDList.getSelectedItem());
                rs.next();
            }
            catch (SQLException e)
            {
                displaySQLErrors(e);
            }
        }
    });
}
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
        u_idText.setText(rs.getString("u_id"));
        fnameText.setText(rs.getString("fname"));
        lnameText.setText(rs.getString("lname"));
    }
    catch (SQLException selectException)
    {
        displaySQLErrors(selectException);
    }
}

});

updateAdminButton = new Button("Update admin");
updateAdminButton.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        try
        {
            Statement statement = connection.createStatement();
            int i = statement.executeUpdate("UPDATE admin "
            + "SET fname='" + fnameText.getText() + "', "
            + "lname='" + lnameText.getText() + "'"
            + " WHERE u_id = "
            + adminIDList.getSelectedIndex());
            errorText.append("\nUpdated " + i + " rows
successfully");

            adminIDList.removeAll();
            loadAdmin();
        }
        catch (SQLException insertException)
        {
            displaySQLErrors(insertException);
            JOptionPane.showMessageDialog(null,"Input is in
wrong format.", "Error Message",JOptionPane.ERROR_MESSAGE);
        }
    }
});

u_idText = new TextField(15);
fnameText = new TextField(15);
lnameText = new TextField(15);

errorText = new TextArea(10, 40);
errorText.setEditable(false);

Panel first = new Panel();
first.setLayout(new GridLayout(4, 2));
first.add(new Label("Admin/User ID:"));
first.add(u_idText);
first.add(new Label("First Name:"));
first.add(fnameText);
first.add(new Label("Last Name:"));
first.add(lnameText);

Panel second = new Panel(new GridLayout(4, 1));
second.add(updateAdminButton);

Panel third = new Panel();
third.add(errorText);

add(first);
add(second);
add(third);

setTitle("Update Admin");
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
        setSize(500, 600);
        setLayout(new FlowLayout());
        setVisible(true);
    }

    private void displaySQLExceptions(SQLException e)
    {
        errorText.append("\nSQLException: " + e.getMessage() + "\n");
        errorText.append("SQLState:      " + e.getSQLState() + "\n");
        errorText.append("VendorError:  " + e.getErrorCode() + "\n");
    }

    public static void main(String[] args)
    {
        UpdateAdmin ups = new UpdateAdmin();

        ups.addWindowListener(new WindowAdapter(){
            public void windowClosing(WindowEvent e)
            {
                System.exit(0);
            }
        });

        ups.buildGUI();
    }
}
```

### 3. Deleting Admin Table:

```
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
public class DeleteAdmin extends Frame
{
    Button deleteAdminButton;
    List adminIDList;
    TextField u_idText, fnameText, lnameText;
    TextArea errorText;
    Connection connection;
    Statement statement;
    ResultSet rs;

    public DeleteAdmin()
    {
        try
        {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
        catch (Exception e)
        {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        connectToDB();
    }

    public void connectToDB()
    {
        try
        {
            connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:tasneem", "tasneem", "tasneem")
;
TASNEEM BEGUM
ROLL NO: 1602-18-737-112
```



## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
        statement = connection.createStatement();
    }
    catch (SQLException connectException)
    {
        System.out.println(connectException.getMessage());
        System.out.println(connectException.getSQLState());
        System.out.println(connectException.getErrorCode());
        System.exit(1);
    }
}

private void loadAdmin()
{
    try
    {
        rs = statement.executeQuery("SELECT * FROM admin");
        while (rs.next())
        {
            adminIDList.add(rs.getString("u_id"));
        }
    }
    catch (SQLException e)
    {
        displaySQLErrors(e);
    }
}

public void buildGUI()
{
    adminIDList = new List(10);
    loadAdmin();
    add(adminIDList);

    adminIDList.addItemListener(new ItemListener()
    {
        public void itemStateChanged(ItemEvent e)
        {
            try
            {
                rs = statement.executeQuery("SELECT * FROM admin");
                while (rs.next())
                {
                    if
(rs.getString("u_id").equals(adminIDList.getSelectedItem()))
                        break;
                }
                if (!rs.isAfterLast())
                {
                    u_idText.setText(rs.getString("u_id"));
                    fnameText.setText(rs.getString("fname"));
                    lnameText.setText(rs.getString("lname"));
                }
            }
            catch (SQLException selectException)
            {
                displaySQLErrors(selectException);
            }
        }
    });

    deleteAdminButton = new Button("Delete Admin");
    deleteAdminButton.addActionListener(new ActionListener()
    {
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
        public void actionPerformed(ActionEvent e)
        {
            try
            {
                Statement statement = connection.createStatement();
                int i = statement.executeUpdate("DELETE FROM admin
WHERE u_id = "
                + adminIDList.getSelectedItem());
                errorText.append("\nDeleted " + i + " rows
successfully");

                u_idText.setText(null);
                fnameText.setText(null);
                lnameText.setText(null);
                adminIDList.removeAll();
                loadAdmin();
            }
            catch (SQLException insertException)
            {
                displaySQLErrors(insertException);
            }
        }
    });

    u_idText = new TextField(15);
    fnameText = new TextField(15);
    lnameText = new TextField(15);

    errorText = new TextArea(10, 40);
    errorText.setEditable(false);

    Panel first = new Panel();
    first.setLayout(new GridLayout(4, 2));
    first.add(new Label("Admin/User ID:"));
    first.add(u_idText);
    first.add(new Label("First Name:"));
    first.add(fnameText);
    first.add(new Label("Last Name:"));
    first.add(lnameText);

    Panel second = new Panel(new GridLayout(4, 1));
    second.add(deleteAdminButton);

    Panel third = new Panel();
    third.add(errorText);

    add(first);
    add(second);
    add(third);

    setTitle("Remove Admin");
    setSize(450, 600);
    setLayout(new FlowLayout());
    setVisible(true);
}

private void displaySQLErrors(SQLException e)
{
    errorText.append("\nSQLException: " + e.getMessage() + "\n");
    errorText.append("SQLState: " + e.getSQLState() + "\n");
    errorText.append("VendorError: " + e.getErrorCode() + "\n");
}
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

```
public static void main(String[] args)
{
    DeleteAdmin dels = new DeleteAdmin();

    dels.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e)
        {
            System.exit(0);
        }
    });

    dels.buildGUI();
}
}
```

## Main Program:

```
import java.awt.*;
import java.awt.event.*;

class HockeyClubEventTracker extends Frame implements ActionListener
{
    String msg = "";
    Label l1,l2;
    Frame f;
    InsertAdmin iadmin;
    UpdateAdmin uadmin;
    DeleteAdmin dadmin;
    InsertEvents ievents;
    DeleteEvents devents;
    UpdateEvents uevents;
    InsertPlayer iplayer;
    DeletePlayer dplayer;
    UpdatePlayer uplayer;
    InsertTracks itracks;
    DeleteTracks dtracks;
    UpdateTracks utracks;
    InsertParticipate iparticipate;
    DeleteParticipate dparticipate;
    UpdateParticipate uparticipate;

    HockeyClubEventTracker()
    {
        l1 = new Label("Welcome to");
        l2=new Label("Hockey Club Event Activity Tracker");
        l1.setAlignment(Label.CENTER);
        l2.setAlignment(Label.CENTER);
        l1.setFont(new Font("SansSerif", Font.BOLD, 40));
        l2.setFont(new Font("SansSerif", Font.BOLD, 50));
        setLayout(new BorderLayout());
        add(l1,BorderLayout.NORTH);
        add(l2,BorderLayout.CENTER);

        MenuBar mbar = new MenuBar();
        setMenuBar(mbar);

        Menu admin = new Menu("Admin");
        MenuItem item1, item2, item3;
        admin.add(item1 = new MenuItem("Insert Admin"));
        admin.add(item2 = new MenuItem("Update Admin"));
        admin.add(item3 = new MenuItem("Delete Admin"));
        mbar.add(admin);
    }
}
```

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
Menu events = new Menu("Events");
MenuItem item4, item5, item6;
events.add(item4 = new MenuItem("Insert Events"));
events.add(item5 = new MenuItem("Update Events"));
events.add(item6 = new MenuItem("Delete Events"));
mbar.add(events);

Menu player = new Menu("Player");
MenuItem item7, item8, item9;
player.add(item7 = new MenuItem("Insert Player"));
player.add(item8 = new MenuItem("Update Player"));
player.add(item9 = new MenuItem("Delete Player"));
mbar.add(player);

Menu tracks= new Menu("Tracks");
MenuItem item10,item11,item12;
tracks.add(item10=new MenuItem("Insert Tracks"));
tracks.add(item11=new MenuItem("Update Tracks"));
tracks.add(item12=new MenuItem("Delete Tracks"));
mbar.add(tracks);

Menu participate=new Menu("Participate");
MenuItem item13,item14,item15;
participate.add(item13=new MenuItem("Insert Participate"));
participate.add(item14=new MenuItem("Update Participate"));
participate.add(item15=new MenuItem("Delete Participate"));
mbar.add(participate);

item1.addActionListener(this);
item2.addActionListener(this);
item3.addActionListener(this);
item4.addActionListener(this);
item5.addActionListener(this);
item6.addActionListener(this);
item7.addActionListener(this);
item8.addActionListener(this);
item9.addActionListener(this);
item10.addActionListener(this);
item11.addActionListener(this);
item12.addActionListener(this);
item13.addActionListener(this);
item14.addActionListener(this);
item15.addActionListener(this);

addWindowListener(new WindowAdapter(){
    public void windowClosing(WindowEvent we)
    {
        System.exit(0);
    }
});

setTitle("Hockey Club Event Activity Tracker ");
Color clr = new Color(150, 150, 150);
setBackground(clr);
setFont(new Font("SansSerif", Font.BOLD, 14));
setSize(600, 600);
setVisible(true);

}

public void actionPerformed(ActionEvent ae)
{
    String arg = ae.getActionCommand();
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
        if(arg.equals("Insert Admin"))
        {
            iadmin = new InsertAdmin();

            iadmin.addWindowListener(new WindowAdapter(){
                public void windowClosing(WindowEvent e)
                {
                    iadmin.dispose();
                }
            });
            iadmin.buildGUI();
        }

        else if(arg.equals("Update Admin"))
        {
            uadmin = new UpdateAdmin();
            uadmin.addWindowListener(new WindowAdapter(){
                public void windowClosing(WindowEvent e)
                {
                    uadmin.dispose();
                }
            });
            uadmin.buildGUI();
        }

        else if(arg.equals("Delete Admin"))
        {
            dadmin = new DeleteAdmin();

            dadmin.addWindowListener(new WindowAdapter(){
                public void windowClosing(WindowEvent e)
                {
                    dadmin.dispose();
                }
            });
            dadmin.buildGUI();
        }

        else if(arg.equals("Insert Events"))
        {
            ievents = new InsertEvents();
            ievents.addWindowListener(new WindowAdapter(){
                public void windowClosing(WindowEvent e)
                {
                    ievents.dispose();
                }
            });
            ievents.buildGUI();
        }
        else if(arg.equals("Update Events"))
        {
            uevents = new UpdateEvents();
            uevents.addWindowListener(new WindowAdapter(){
                public void windowClosing(WindowEvent e)
                {
                    uevents.dispose();
                }
            });
            uevents.buildGUI();
        }
        else if(arg.equals("Delete Events"))
        {
            devents = new DeleteEvents();
            devents.addWindowListener(new WindowAdapter(){
                public void windowClosing(WindowEvent e)
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER

```
        {
            devents.dispose();
        }
    });
    devents.buildGUI();
}
else if(arg.equals("Insert Player"))
{
    iplayer = new InsertPlayer();
    iplayer.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e)
        {
            iplayer.dispose();
        }
    });
    iplayer.buildGUI();
}
else if(arg.equals("Update Player"))
{
    uplayer = new UpdatePlayer();
    uplayer.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e)
        {
            uplayer.dispose();
        }
    });
    uplayer.buildGUI();
}
else if(arg.equals("Delete Player"))
{
    dplayer = new DeletePlayer();
    dplayer.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e)
        {
            dplayer.dispose();
        }
    });
    dplayer.buildGUI();
}
else if(arg.equals("Insert Tracks"))
{
    itracks = new InsertTracks();
    itracks.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e)
        {
            itracks.dispose();
        }
    });
    itracks.buildGUI();
}
else if(arg.equals("Update Tracks"))
{
    utracks = new UpdateTracks();
    utracks.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e)
        {
            utracks.dispose();
        }
    });
    utracks.buildGUI();
}
else if(arg.equals("Delete Tracks"))
{
    dtracks = new DeleteTracks();
    dtracks.addWindowListener(new WindowAdapter(){
```

TASNEEM BEGUM

ROLL NO: 1602-18-737-112

```
        public void windowClosing(WindowEvent e)
        {
            dtracks.dispose();
        }
    });
    dtracks.buildGUI();
}
else if(arg.equals("Insert Participate"))
{
    iparticipate = new InsertParticipate();
    iparticipate.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e)
        {
            iparticipate.dispose();
        }
    });
    iparticipate.buildGUI();
}
else if(arg.equals("Update Participate"))
{
    uparticipate = new UpdateParticipate();
    uparticipate.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e)
        {
            uparticipate.dispose();
        }
    });
    uparticipate.buildGUI();
}
else if(arg.equals("Delete Participate"))
{
    dparticipate = new DeleteParticipate();
    dparticipate.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e)
        {
            dparticipate.dispose();
        }
    });
    dparticipate.buildGUI();
}

}

public static void main(String ... args)
{
    new HockeyClubEventTracker();
}
}
```

## ii) GitHub Link and folder structure:

Link-<https://github.com/tasneembegum/Tasneem-Begum>

## Folder Structure:

- 1.HockeyClubEventActivityTracker.pdf
- 2.MiniProjectReport.pdf
- 3.1602-18-737-112.zip

1602-18-737-112

HockeyClubEventTracke

→ .settings

→ bin

→ src

→ .classpath

→ .project

dml operations and  
connectivity programs for 5 tables

## f) Testing:

New Admin Creation

Admin/User ID: a

First Name: Rahul

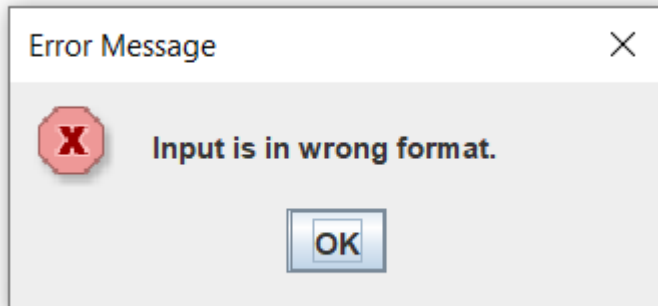
Last Name: Kumar

Insert Admin

LException: ORA-00984: column not allowed here  
LState: 42000  
IdorError: 984

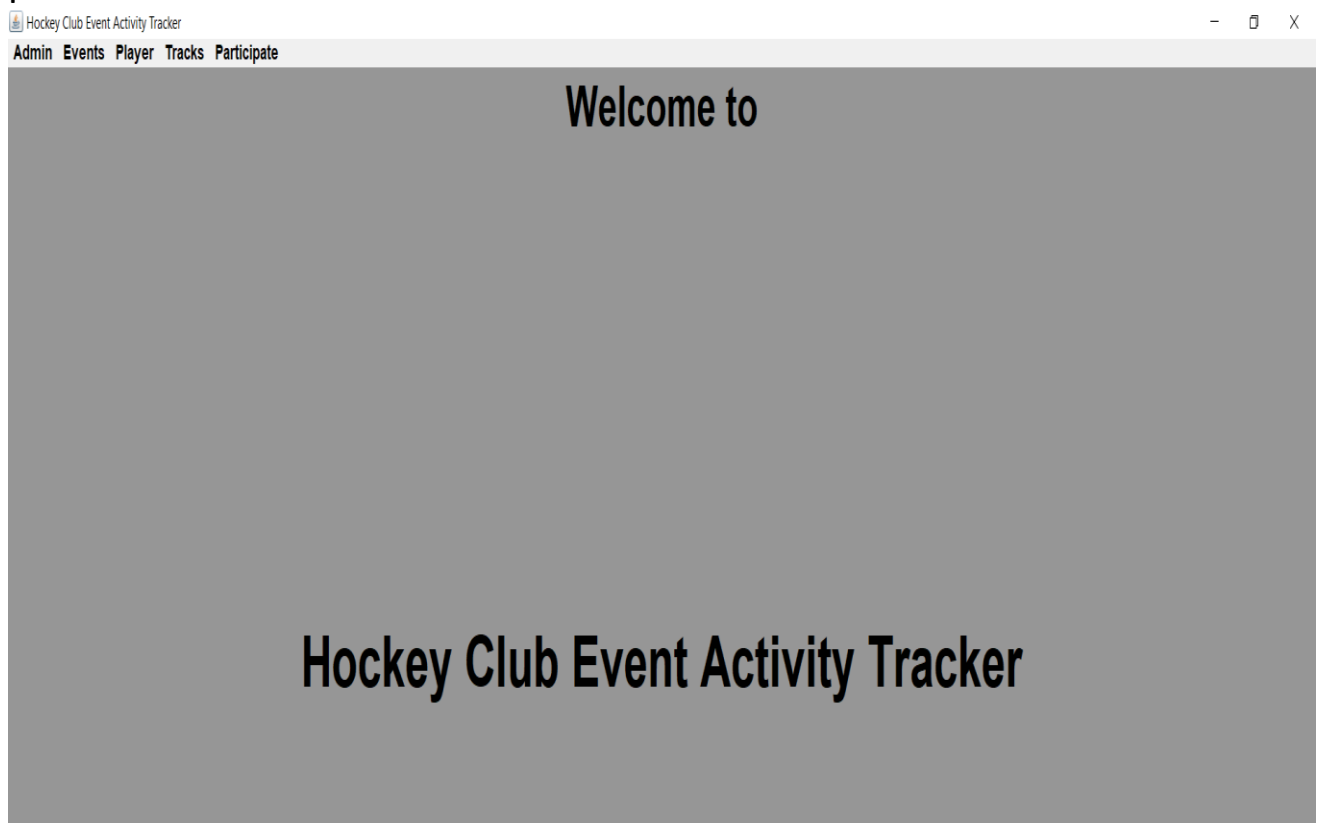
If incorrect values are entered which mismatch datatypes, the following error pop up box is shown:



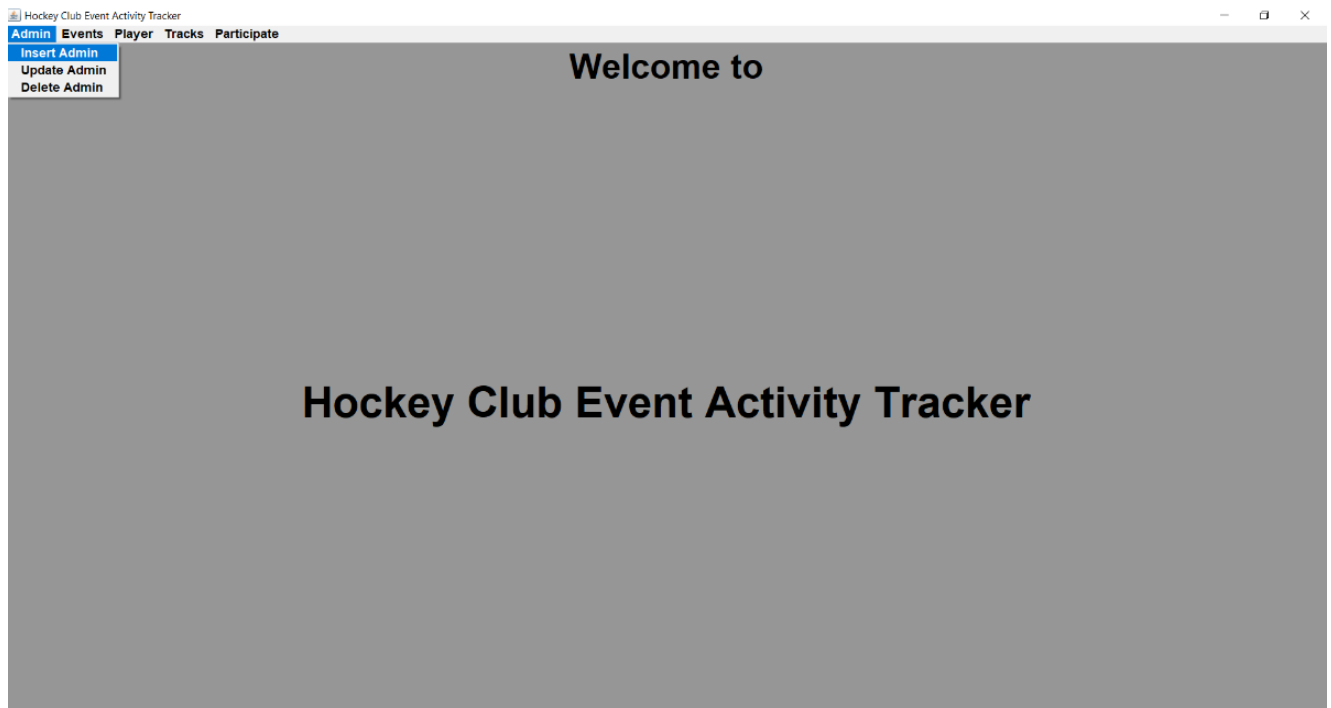


## **RESULTS**

Main Program when run displays the following window frame from which all tables can be accessed and different DML operations can be performed:



## TITLE: HOCKEY CLUB EVENTS ACTIVITY TRACKER



I. Here are DML operations for Admin Table:

1. Inserting into Admin Table:

The screenshot shows a web form titled "New Admin Creation". The form has three input fields for "Admin/User ID:", "First Name:", and "Last Name:". Below the fields is a button labeled "Insert Admin".

2. Updating Admin Table:

1  
2  
3  
4  
5  
6  
34

Admin/User ID:  
First Name:  
Last Name:

34  
Rahul  
Deshmukh

Update admin

Updated 1 rows successfully

3. Delete Admin:

34  
1  
2  
3  
4  
5  
6

Admin/User ID:  
First Name:  
Last Name:

34  
Rahul  
Deshmukh

Delete Admin

1  
2  
3  
4  
5  
6

Admin/User ID:  
First Name:  
Last Name:

Delete Admin

Deleted 1 rows successfully

## II. Event Table: Insert, Update and Delete GUI Pages

New Event Creation

Event ID: 106

Location: hyd

Event Date: 28-JUN-2020

Insert Event

erted 1 rows successfully

Update Events

101  
102  
103  
104  
105  
106

Event ID: 106

Location: HYD

Event Date: 28-JUN-2020

Update events

SQLException: ORA-01861: literal does not match format stri

SQLState: 22008

VendorError: 1861

Updated 1 rows successfully

Remove Event

101  
102  
103  
104  
105

Event ID:

Location:

Event Date:

Delete Event

Deleted 1 rows successfully

### III. Player Table: Insert, Update and Delete GUI Pages

Insert New Player

Player ID:

First Name:

Last Name:

Position:

Rank:

Insert Player

Update Player

17  
28  
36  
50  
73  
89

Player ID: 89

First Name: Puneet

Last Name: Agarwal

Position: Center

Rank: 25

Update Player

Updated 1 rows successfully

Remove Player

17  
28  
36  
50  
73

Player ID:

First Name:

Last Name:

Position:

Rank:

Delete Player

Deleted 1 rows successfully

IV. Tracks Table: Insert, Update and Delete GUI Pages

New Event Track Creation

User/Admin ID:  
Event ID:  
Track Time:

1  
2  
3  
4  
5

User/Admin ID:  
Event ID:  
Track Time:

Insert Tracks

Update events

Remove Event Tracked

1  
2  
3  
4  
5

User/Admin ID:  
Event ID:  
Track Time:

Delete track event

## V. Participate Table: Insert, Update and Delete GUI Pages

Insert info of Player Participated in Event

Event ID:

Player ID:

Participate Date:

Insert Player Participate

Update info of Player Participate

101  
102  
103  
104  
105

Event ID:

Player ID:

Participate Date:

Update Participate info

Updated 1 rows successfully

Remove Participated player info

103  
101  
102  
104  
105

Event ID:

Player ID:

Participate Date:

Delete Participated player

## **DISCUSSION & FUTURE WORK :**

The application done till now is basically to store and track the details of few events of Hockey Club like different events taken place and players participated in that event. There are various other activities of club that can be stored and tracked, such as- details of company sponsored, different equipment's required, tickets details etc. And hence, for this number of entities and relationship between them will come into picture, which can be converted into first tables using SQL commands and then into GUI program using java code.

## **REFERENCES**

<https://www.oracle.com/in/database/technologies/112010-win64soft.html>

<https://www.youtube.com/watch?v=fMp63HsIRbc&t=107s>