**IPL Match Schedule using JavaFX and Mysql**

1. **Main Java**
2. **Data base**

* Db Connection.java

1. **dao**

* Matches.java

1. **Controller**

* Login Controller.java
* MainPanelController.java
* Page01Controller.java
* Page02Controller.java
* Page03Controller.java
* Page04Controller.java
* RegisterController.java

1. **View**

* HomeView.fxml
* LoginView.fxml
* MainPanelView.fxml
* Page01View.fxml
* Page02View.fxml
* Page03View.fxml
* Page04View.fxml
* Page05View.fxml
* RegisterView.fxml

1. **Helper**

* AlertHelper.java

**Application:**

**Main.java**

package application;

import javafx.application.Application;

import javafx.stage.Stage;

import javafx.scene.Scene;

import javafx.scene.layout.BorderPane;

public class Main extends Application {

@Override

public void start(Stage primaryStage) {

try {

BorderPane root = new BorderPane();

Scene scene = new Scene(root,400,400);

scene.getStylesheets().add(getClass().getResource("application.css").toExternalForm());

primaryStage.setScene(scene);

primaryStage.show();

} catch(Exception e) {

e.printStackTrace();

}

}

public static void main(String[] args) {

launch(args);

}

}

**DataBase:**

**DbConnection.java**

**package** database;

**import** java.io.FileInputStream;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.util.Properties;

**import** java.util.logging.Level;

**import** java.util.logging.Logger;

**public** **class** DbConnection {

**private** Connection con;

**private** **static** DbConnection dbc;

**private** DbConnection() {

**try** {

DriverManager.registerDriver(**new** oracle.jdbc.driver.OracleDriver());

FileInputStream fis=**new** FileInputStream("C:\\Users\\siddi\\eclipse-workspace\\IPL\\assets\\database\\connection.prop");

Properties p=**new** Properties ();

p.load(fis);

con=DriverManager.getConnection((String)p.get("url"),(String)p.get("username"),(String)p.get("password"));

} **catch** (Exception ex) {

ex.printStackTrace();

Logger.getLogger(DbConnection.**class**.getName()).log(Level.SEVERE, **null**, ex);

}

}

**public** **static** DbConnection getDatabaseConnection() {

**if** (dbc == **null**) {

dbc = **new** DbConnection();

}

**return** dbc;

}

**public** Connection getConnection() {

**return** con;

}

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

**new** DbConnection();

}

}

**Dao:**

**Matches.java**

**package** dao;

**public** **class** Matches {

**private** **final** **int** id;

**private** **final** String location;

**private** **final** String team1;

**private** **final** String team2;

**private** **final** String match\_time;

**private** **final** String match\_outcome;

**private** **final** String match\_date;

**public** Matches(**int** id, String location, String team1, String team2,String match\_time,String match\_outcome,String match\_date) {

**super**();

**this**.id = id;

**this**.location = location;

**this**.team1 = team1;

**this**.team2 = team2;

**this**.match\_time = match\_time;

**this**.match\_outcome = match\_outcome;

**this**.match\_date = match\_date;

}

**public** **int** getId() {

**return** id;

}

**public** String getLocation() {

**return** location;

}

**public** String getTeam1() {

**return** team1;

}

**public** String getTeam2() {

**return** team2;

}

**public** String getMatch\_time() {

**return** match\_time;

}

**public** String getMatch\_outcome() {

**return** match\_outcome;

}

**public** String getMatch\_date() {

**return** match\_date;

}

**LoginController.java**

**package** controller;

**import** database.DbConnection;

**import** helper.AlertHelper;

**import** java.io.IOException;

**import** java.net.URL;

**import** java.sql.Connection;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.util.ResourceBundle;

**import** javafx.application.Application;

**import** javafx.fxml.FXML;

**import** javafx.fxml.FXMLLoader;

**import** javafx.fxml.Initializable;

**import** javafx.scene.Parent;

**import** javafx.scene.Scene;

**import** javafx.scene.control.Alert;

**import** javafx.scene.control.Button;

**import** javafx.scene.control.TextField;

**import** javafx.scene.image.Image;

**import** javafx.stage.Stage;

**import** javafx.stage.Window;

**import** main.MainPanel;

**public** **class** LoginController **implements** Initializable {

**private** **final** Connection con;

@FXML

**private** TextField username;

@FXML

**private** TextField password;

@FXML

**private** Button loginButton;

Window window;

@Override

**public** **void** initialize(URL url, ResourceBundle rb) {

}

**public** LoginController() {

DbConnection dbc = DbConnection.getDatabaseConnection();

con = dbc.getConnection();

}

@FXML

**private** **void** login() **throws** Exception {

**if** (**this**.isValidated()) {

PreparedStatement ps;

ResultSet rs;

String query = "select \* from Fara\_amreen\_users WHERE username = ? and password = DES\_ENCRYPT(?,'ipl')";

**try** {

ps = con.prepareStatement(query);

ps.setString(1, username.getText());

ps.setString(2, password.getText());

rs = ps.executeQuery();

**if** (rs.next()) {

Stage stage = (Stage) loginButton.getScene().getWindow();

stage.close();

Parent root = FXMLLoader.load(getClass().getResource("/view/MainPanelView.fxml"));

Scene scene = **new** Scene(root);

stage.setScene(scene);

stage.setTitle("Admin Panel");

stage.getIcons().add(**new** Image("/asset/icon.png"));

stage.show();

query = "update Fara\_amreen\_login set role='" + rs.getString(4) + "'";

**try** {

ps = con.prepareStatement(query);

ps.executeUpdate();

} **catch** (SQLException ex) {

//Logger.getLogger(Page01Controller.class.getName()).log(Level.SEVERE, null, ex);

ex.printStackTrace();

}

} **else** {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Invalid username and password.");

username.requestFocus();

}

} **catch** (SQLException ex) {

System.out.println(ex);

}

}

}

**private** **boolean** isValidated() {

window = loginButton.getScene().getWindow();

**if** (username.getText().equals("")) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Username text field cannot be blank.");

username.requestFocus();

} **else** **if** (username.getText().length() < 5 || username.getText().length() > 25) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Username text field cannot be less than 5 and greator than 25 characters.");

username.requestFocus();

} **else** **if** (password.getText().equals("")) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Password text field cannot be blank.");

password.requestFocus();

} **else** **if** (password.getText().length() < 5 || password.getText().length() > 25) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Password text field cannot be less than 5 and greator than 25 characters.");

password.requestFocus();

} **else** {

**return** **true**;

}

**return** **false**;

}

@FXML

**private** **void** showRegisterStage() **throws** IOException {

Stage stage = (Stage) loginButton.getScene().getWindow();

stage.close();

Parent root = FXMLLoader.load(getClass().getResource("/view/RegisterView.fxml"));

Scene scene = **new** Scene(root);

stage.setScene(scene);

stage.setTitle("User Registration");

stage.getIcons().add(**new** Image("/asset/icon.png"));

stage.show();

}

}

**MainPanelController.java**

**package** controller;

**import** database.DbConnection;

**import** java.io.IOException;

**import** java.net.URL;

**import** java.sql.Connection;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.util.Iterator;

**import** java.util.List;

**import** java.util.ResourceBundle;

**import** java.util.logging.Level;

**import** java.util.logging.Logger;

**import** javafx.event.ActionEvent;

**import** javafx.fxml.FXML;

**import** javafx.fxml.FXMLLoader;

**import** javafx.fxml.Initializable;

**import** javafx.scene.Parent;

**import** javafx.scene.Scene;

**import** javafx.scene.chart.AreaChart;

**import** javafx.scene.chart.LineChart;

**import** javafx.scene.control.Button;

**import** javafx.scene.image.Image;

**import** javafx.scene.layout.BorderPane;

**import** javafx.stage.Stage;

**public** **class** MainPanelController **implements** Initializable {

@FXML

**private** BorderPane borderPane;

**private** List<Button> menus;

@FXML

**private** AreaChart<?, ?> chartPurchase;

@FXML

**private** AreaChart<?, ?> chartSale;

@FXML

**private** LineChart<?, ?> chartReceipt;

/\*\*

\* Initializes the controller class.

\*/

@Override

**public** **void** initialize(URL url, ResourceBundle rb) {

}

**private** **final** Connection con;

**public** MainPanelController() {

DbConnection dbc = DbConnection.getDatabaseConnection();

con = dbc.getConnection();

}

**private** **void** changeButtonBackground(ActionEvent e) {

Iterator<Button> iteratorMenus = menus.iterator();

**while** (iteratorMenus.hasNext()) {

Button clickedButton = (Button) e.getSource();

Button OtherButton = iteratorMenus.next();

**if** (clickedButton == OtherButton) {

clickedButton.setStyle("-fx-text-fill:#f0f0f0;-fx-background-color:#2b2a26;");

} **else** {

**if** (OtherButton != **null**) {

OtherButton.setStyle("-fx-text-fill:#f0f0f0;-fx-background-color:#404040;");

}

}

}

}

@FXML

**private** **void** clear() {

borderPane.setCenter(**null**);

}

@FXML

**private** **void** loadFXML(String fileName) {

Parent parent;

**try** {

parent = FXMLLoader.load(getClass().getResource("/view/" + fileName + ".fxml"));

borderPane.setCenter(parent);

} **catch** (IOException ex) {

Logger.getLogger(MainPanelController.**class**.getName()).log(Level.SEVERE, **null**, ex);

}

}

@FXML

**private** **void** close() **throws** IOException {

Stage stage = (Stage) borderPane.getScene().getWindow();

stage.close();

Parent root = FXMLLoader.load(getClass().getResource("/view/LoginView.fxml"));

Scene scene = **new** Scene(root);

stage.setScene(scene);

stage.setTitle("User Login");

stage.getIcons().add(**new** Image("/asset/icon.png"));

stage.show();

}

@FXML

**private** **void** loadPage01View(ActionEvent e) {

loadFXML("Page01View");

changeButtonBackground(e);

}

@FXML

**private** **void** loadPage02View(ActionEvent e) {

PreparedStatement ps;

ResultSet rs;

String team = **null**;

String query = "select role from fara\_amreen\_login where role!='admin'";

**try** {

ps = con.prepareStatement(query);

rs = ps.executeQuery();

**while**(rs.next()){

team = rs.getString(1);

}

}**catch**(SQLException exp){

exp.printStackTrace();

}

**if**(team==**null**){

loadFXML("Page02View");

}**else**{

loadFXML("Page03View");

}

changeButtonBackground(e);

}

@FXML

**private** **void** loadPage03View(ActionEvent e) {

loadFXML("Page03View");

changeButtonBackground(e);

}

@FXML

**private** **void** loadPage04View(ActionEvent e) {

String team = **null**;

**final** String query = "select role from fara\_amreen\_login where role!='admin'";

**try** {

**final** PreparedStatement ps = **this**.con.prepareStatement(query);

**final** ResultSet rs = ps.executeQuery();

**while** (rs.next()) {

team = rs.getString(1);

}

}

**catch** (SQLException exp) {

exp.printStackTrace();

}

**if** (team == **null**) {

**this**.loadFXML("Page04View");

}

**else** {

**this**.loadFXML("Page03View");

}

**this**.changeButtonBackground(e);

}

@FXML

**private** **void** loadPage05View(ActionEvent e) {

loadFXML("Page05View");

changeButtonBackground(e);

}

@FXML

**private** **void** loadPage06View(ActionEvent e) {

loadFXML("Page06View");

changeButtonBackground(e);

}

@FXML

**private** **void** loadPage07View(ActionEvent e) {

loadFXML("Page07View");

changeButtonBackground(e);

}

@FXML

**private** **void** loadPage08View(ActionEvent e) {

loadFXML("Page08View");

changeButtonBackground(e);

}

@FXML

**private** **void** loadPage09View(ActionEvent e) {

loadFXML("Page09View");

changeButtonBackground(e);

}

@FXML

**private** **void** loadPage10View(ActionEvent e) {

loadFXML("Page10View");

changeButtonBackground(e);

}

@FXML

**private** **void** loadHomeView(ActionEvent e) {

loadFXML("HomeView");

changeButtonBackground(e);

}

}

**Page01Controller.java**

**package** controller;

**import** database.DbConnection;

**import** dao.Matches;

**import** java.net.URL;

**import** java.sql.Connection;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.ResourceBundle;

**import** java.util.logging.Level;

**import** java.util.logging.Logger;

**import** javafx.beans.property.SimpleBooleanProperty;

**import** javafx.beans.value.ObservableValue;

**import** javafx.fxml.Initializable;

**import** javafx.collections.FXCollections;

**import** javafx.collections.ObservableList;

**import** javafx.event.ActionEvent;

**import** javafx.event.EventHandler;

**import** javafx.fxml.FXML;

**import** javafx.scene.control.Button;

**import** javafx.scene.control.TableCell;

**import** javafx.scene.control.TableColumn;

**import** javafx.scene.control.TableView;

**import** javafx.scene.control.cell.PropertyValueFactory;

**import** javafx.scene.control.cell.TextFieldTableCell;

**import** javafx.scene.layout.BorderPane;

**import** javafx.util.Callback;

**public** **class** Page01Controller **implements** Initializable {

**private** **final** Connection con;

@FXML **private** BorderPane borderPane;

@FXML **private** TableView<Matches> table;

@FXML **private** TableColumn<Matches, Integer> id;

@FXML **private** TableColumn<Matches, String> location;

@FXML **private** TableColumn<Matches, String> team1;

@FXML **private** TableColumn<Matches, String> team2;

@FXML **private** TableColumn<Matches, String> match\_time;

@FXML **private** TableColumn<Matches, String> match\_outcome;

@FXML **private** TableColumn<Matches, String> match\_date;

**public** Page01Controller() {

DbConnection dbc = DbConnection.getDatabaseConnection();

con = dbc.getConnection();

}

ArrayList<Matches> l1 = **new** ArrayList<>();

**public** ObservableList<Matches> list = list = FXCollections.observableArrayList(

);

@Override

**public** **void** initialize(URL arg0, ResourceBundle arg1) {

id.setCellValueFactory(**new** PropertyValueFactory<Matches, Integer>("id"));

location.setCellValueFactory(**new** PropertyValueFactory<Matches, String>("location"));

team1.setCellValueFactory(**new** PropertyValueFactory<Matches, String>("team1"));

team2.setCellValueFactory(**new** PropertyValueFactory<Matches, String>("team2"));

match\_time.setCellValueFactory(**new** PropertyValueFactory<Matches, String>("match\_time"));

match\_outcome.setCellValueFactory(**new** PropertyValueFactory<Matches, String>("match\_outcome"));

PreparedStatement ps;

ResultSet rs;

String team = **null**;

String query = "select role from Fara\_amreen\_login where role!='admin'";

**try** {

ps = con.prepareStatement(query);

rs = ps.executeQuery();

**while**(rs.next()){

team = rs.getString(1);

}

}**catch**(Exception e){

e.printStackTrace();

}

**if**(team == **null**)

match\_outcome.setCellFactory(TextFieldTableCell.forTableColumn());

match\_outcome.setOnEditCommit(event -> {

Matches row = event.getRowValue();

String mo = event.getNewValue();

PreparedStatement ps1;

String query1 = "update Fara\_amreen\_matches set match\_outcome='" + mo + "' where match\_id=" + row.getId();

**try** {

ps1 = con.prepareStatement(query1);

ps1.executeUpdate();

} **catch** (SQLException ex) {

//Logger.getLogger(Page01Controller.class.getName()).log(Level.SEVERE, null, ex);

ex.printStackTrace();

}

});

match\_date.setCellValueFactory(**new** PropertyValueFactory<Matches, String>("match\_date"));

table.setEditable(**true**);

**if**(team==**null**){

//Insert Button

TableColumn col\_action = **new** TableColumn<>("Action");

table.getColumns().add(col\_action);

col\_action.setCellValueFactory(

**new** Callback<TableColumn.CellDataFeatures<Matches, Boolean>,

ObservableValue<Boolean>>() {

@Override

**public** ObservableValue<Boolean> call(TableColumn.CellDataFeatures<Matches, Boolean> p) {

**return** **new** SimpleBooleanProperty(p.getValue() != **null**);

}

});

//Adding the Button to the cell

col\_action.setCellFactory(

**new** Callback<TableColumn<Matches, Boolean>, TableCell<Matches, Boolean>>() {

@Override

**public** TableCell<Matches, Boolean> call(TableColumn<Matches, Boolean> p) {

**return** **new** ButtonCell();

}

});

}

table.setItems(list);

Matches s1 = **null**;

/\*Matches s2 = new Matches(2, "Nava2", "nava", 2);

Matches s3 = new Matches(3, "Nava3", "nava", 2);\*/

query = "select \* from Fara\_amreen\_matches";

**if**(team!=**null**){

query = query + " where team1='" + team + "' or team2='" + team + "'";

}

**try** {

ps = con.prepareStatement(query);

rs = ps.executeQuery();

**while**(rs.next()){

s1 = **new** Matches(rs.getInt(1), rs.getString(2), rs.getString(3), rs.getString(4), rs.getString(5), rs.getString(6), rs.getString(7));

list.add(s1);

}

}**catch**(Exception e){

//e.printStackTrace();

}

}

//Define the button cell

**private** **class** ButtonCell **extends** TableCell<Matches, Boolean> {

**final** Button cellButton = **new** Button("Delete");

ButtonCell(){

//Action when the button is pressed

cellButton.setOnAction(**new** EventHandler<ActionEvent>(){

@Override

**public** **void** handle(ActionEvent t) {

// get Selected Item

Matches currentMatch = (Matches) ButtonCell.**this**.getTableView().getItems().get(ButtonCell.**this**.getIndex());

//remove selected item from the table list

list.remove(currentMatch);

String sql = "Delete from Fara\_amreen\_matches where match\_id =" + currentMatch.getId();

PreparedStatement ps;

**try** {

ps = con.prepareStatement(sql);

ps.executeUpdate();

} **catch** (SQLException ex) {

//Logger.getLogger(Page01Controller.class.getName()).log(Level.SEVERE, null, ex);

}

}

});

}

//Display button if the row is not empty

@Override

**protected** **void** updateItem(Boolean t, **boolean** empty) {

**super**.updateItem(t, empty);

**if**(!empty){

setGraphic(cellButton);

}

}

}

}

Page02Controller.java

**package** controller;

**import** java.sql.ResultSet;

**import** java.util.Collection;

**import** javafx.collections.FXCollections;

**import** java.sql.PreparedStatement;

**import** java.sql.Statement;

**import** java.sql.SQLException;

**import** helper.AlertHelper;

**import** javafx.scene.control.Alert;

**import** database.DbConnection;

**import** java.util.ArrayList;

**import** java.util.ResourceBundle;

**import** java.net.URL;

**import** javafx.stage.Window;

**import** javafx.scene.control.Button;

**import** javafx.collections.ObservableList;

**import** java.util.List;

**import** javafx.scene.control.ComboBox;

**import** javafx.fxml.FXML;

**import** javafx.scene.control.TextField;

**import** java.sql.Connection;

**import** javafx.fxml.Initializable;

**public** **class** Page02Controller **implements** Initializable {

**private** **final** Connection con;

@FXML

**private** TextField location;

@FXML

**private** TextField team1;

@FXML

**public** ComboBox<String> cbnTeam1Button;

@FXML

**public** List<String> ComboBoxTeam1;

**public** ObservableList<String> obsComboBoxTeam1;

@FXML

**private** TextField team2;

@FXML

**public** ComboBox<String> cbnTeam2Button;

@FXML

**public** List<String> ComboBoxTeam2;

**public** ObservableList<String> obsComboBoxTeam2;

@FXML

**private** TextField match\_time;

@FXML

**private** TextField match\_outcome;

@FXML

**private** TextField match\_date;

@FXML

**private** Button insertmatchButton;

Window window;

**public** **void** initialize(**final** URL url, **final** ResourceBundle rb) {

**this**.loadComboBox();

}

**public** Page02Controller() {

**this**.ComboBoxTeam1 = **new** ArrayList<String>();

**this**.ComboBoxTeam2 = **new** ArrayList<String>();

DbConnection dbc = DbConnection.getDatabaseConnection();

con = dbc.getConnection();

}

@FXML

**private** **void** insertmatch() {

window = insertmatchButton.getScene().getWindow();

Statement stmt;

**try** {

PreparedStatement ps;

stmt = con.createStatement();

String query = "insert into Fara\_amreen\_matches (location,team1,team2,match\_time,match\_outcome,match\_date)values (?,?,?,?,?,?)";

ps = con.prepareStatement(query);

ps.setString(1, location.getText());

ps.setString(2, **this**.cbnTeam1Button.getValue());

ps.setString(3, **this**.cbnTeam2Button.getValue());

ps.setString(4, match\_time.getText());

ps.setString(5, match\_outcome.getText());

ps.setString(6, match\_date.getText());

**if** (ps.executeUpdate() > 0) {

**this**.clearForm();

//AlertHelper.showAlert(Alert.AlertType.INFORMATION, window, "Information",

// "Inserted Successfully.");

} **else** {

// AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

//"Something went wrong.");

}

} **catch** (SQLException ex) {

//AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

//"Something went wrong.");

}

}

**public** **void** loadComboBox() {

**final** String sql = "select team\_code from Fara\_amreen\_team";

**try** {

**final** PreparedStatement ps = **this**.con.prepareStatement(sql);

**final** ResultSet rs = ps.executeQuery();

**while** (rs.next()) {

**this**.ComboBoxTeam1.add(rs.getString(1));

**this**.ComboBoxTeam2.add(rs.getString(1));

}

}

**catch** (Exception ex) {}

**this**.obsComboBoxTeam1 = (ObservableList<String>)FXCollections.observableArrayList((Collection)**this**.ComboBoxTeam1);

**this**.obsComboBoxTeam2 = (ObservableList<String>)FXCollections.observableArrayList((Collection)**this**.ComboBoxTeam2);

**this**.cbnTeam1Button.setItems((ObservableList)**this**.obsComboBoxTeam1);

**this**.cbnTeam2Button.setItems((ObservableList)**this**.obsComboBoxTeam2);

}

**private** **boolean** clearForm() {

location.clear();

match\_time.clear();

match\_outcome.clear();

match\_date.clear();

**return** **true**;

}

}

Page03Controller.java

**package** controller;

**import** java.net.URL;

**import** java.util.ResourceBundle;

**import** javafx.fxml.Initializable;

**public** **class** Page03Controller **implements** Initializable {

@Override

**public** **void** initialize(URL url, ResourceBundle rb) {

}

}

Page04Controller.java

**package** controller;

**import** java.sql.PreparedStatement;

**import** java.sql.Statement;

**import** java.sql.SQLException;

**import** helper.AlertHelper;

**import** javafx.scene.control.Alert;

**import** database.DbConnection;

**import** java.util.ResourceBundle;

**import** java.net.URL;

**import** javafx.stage.Window;

**import** javafx.scene.control.Button;

**import** javafx.fxml.FXML;

**import** javafx.scene.control.TextField;

**import** java.sql.Connection;

**import** javafx.fxml.Initializable;

**public** **class** Page04Controller **implements** Initializable

{

**private** **final** Connection con;

@FXML

**private** TextField team\_name;

@FXML

**private** TextField team\_code;

@FXML

**private** TextField team\_captain;

@FXML

**private** Button insertteamButton;

Window window;

**public** **void** initialize(**final** URL url, **final** ResourceBundle rb) {

}

**public** Page04Controller() {

**final** DbConnection dbc = DbConnection.getDatabaseConnection();

**this**.con = dbc.getConnection();

}

@FXML

**private** **void** insertteam() {

**this**.window = **this**.insertteamButton.getScene().getWindow();

**try** {

**final** Statement stmt = **this**.con.createStatement();

**final** String query = "insert into fara\_amreen\_team (team\_name,team\_code,team\_captain)values (?,?,?)";

**final** PreparedStatement ps = **this**.con.prepareStatement(query);

ps.setString(1, **this**.team\_name.getText());

ps.setString(2, **this**.team\_code.getText());

ps.setString(3, **this**.team\_captain.getText());

**if** (ps.executeUpdate() > 0) {

**this**.clearForm();

AlertHelper.showAlert(Alert.AlertType.INFORMATION, **this**.window, "Information", "Inserted Successfully.");

}

**else** {

AlertHelper.showAlert(Alert.AlertType.ERROR, **this**.window, "Error", "Something went wrong.");

}

}

**catch** (SQLException ex) {

ex.printStackTrace();

AlertHelper.showAlert(Alert.AlertType.ERROR, **this**.window, "Error", "Something went wrong.");

}

}

**private** **boolean** clearForm() {

**this**.team\_name.clear();

**this**.team\_code.clear();

**this**.team\_captain.clear();

**return** **true**;

}

}

Page05Controller.java

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

**package** controller;

**import** java.net.URL;

**import** java.util.ResourceBundle;

**import** javafx.fxml.Initializable;

/\*\*

\*

\* **@author** Ramesh Godara

\*/

**public** **class** Page05Controller **implements** Initializable {

@Override

**public** **void** initialize(URL url, ResourceBundle rb) {

}

}

RegisterController.java

**package** controller;

**import** java.io.IOException;

**import** javafx.scene.image.Image;

**import** javafx.scene.Scene;

**import** javafx.fxml.FXMLLoader;

**import** javafx.scene.Parent;

**import** javafx.stage.Stage;

**import** java.sql.ResultSet;

**import** java.util.Collection;

**import** javafx.collections.FXCollections;

**import** java.sql.PreparedStatement;

**import** java.sql.Statement;

**import** java.sql.SQLException;

**import** helper.AlertHelper;

**import** javafx.scene.control.Alert;

**import** database.DbConnection;

**import** java.util.ArrayList;

**import** java.util.ResourceBundle;

**import** java.net.URL;

**import** javafx.stage.Window;

**import** javafx.scene.control.Button;

**import** javafx.collections.ObservableList;

**import** java.util.List;

**import** javafx.scene.control.ComboBox;

**import** javafx.fxml.FXML;

**import** javafx.scene.control.TextField;

**import** java.sql.Connection;

**import** javafx.fxml.Initializable;

**public** **class** RegisterController **implements** Initializable {

**private** **final** Connection con;

@FXML

**private** TextField firstName;

@FXML

**private** TextField lastName;

@FXML

**private** TextField email;

@FXML

**private** TextField username;

@FXML

**private** TextField password;

@FXML

**private** TextField confirmPassword;

@FXML

**private** TextField team;

@FXML

**public** ComboBox<String> cbnTeamButton;

@FXML

**public** List<String> ComboBoxTeam;

**public** ObservableList<String> obsComboBoxTeam;

@FXML

**private** Button registerButton;

Window window;

@Override

**public** **void** initialize(URL url, ResourceBundle rb) {

**this**.loadComboBox();

}

**public** RegisterController() {

**this**.ComboBoxTeam = **new** ArrayList<String>();

DbConnection dbc = DbConnection.getDatabaseConnection();

con = dbc.getConnection();

}

@FXML

**private** **void** register() {

window = registerButton.getScene().getWindow();

**if** (**this**.isValidated()) {

Statement stmt;

**try** {

PreparedStatement ps;

stmt = con.createStatement();

String query = "insert into Fara\_amreen\_users (first\_name,last\_name,email,username,password,role)values (?,?,?,?,DES\_ENCRYPT(?,'ipl'),?)";

ps = con.prepareStatement(query);

ps.setString(1, firstName.getText());

ps.setString(2, lastName.getText());

ps.setString(3, email.getText());

ps.setString(4, username.getText());

ps.setString(5, password.getText());

ps.setString(6, **this**.cbnTeamButton.getValue());

**if** (ps.executeUpdate() > 0) {

**this**.clearForm();

AlertHelper.showAlert(Alert.AlertType.INFORMATION, window, "Information",

"You have registered successfully.");

} **else** {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Something went wrong.");

}

} **catch** (SQLException ex) {

//AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

//"Something went wrong.");

}

}

}

**public** **void** loadComboBox() {

**final** String sql = "select team\_code from Fara\_amreen\_team";

**try** {

**final** PreparedStatement ps = **this**.con.prepareStatement(sql);

**final** ResultSet rs = ps.executeQuery();

**while** (rs.next()) {

**this**.ComboBoxTeam.add(rs.getString(1));

}

}

**catch** (Exception ex) {

ex.printStackTrace();

}

**this**.obsComboBoxTeam = (ObservableList<String>)FXCollections.observableArrayList((Collection)**this**.ComboBoxTeam);

**this**.cbnTeamButton.setItems((ObservableList)**this**.obsComboBoxTeam);

}

**private** **boolean** isAlreadyRegistered() {

PreparedStatement ps;

ResultSet rs;

**boolean** usernameExist = **false**;

String query = "select \* from Fara\_amreen\_users WHERE username = ?";

**try** {

ps = con.prepareStatement(query);

ps.setString(1, username.getText());

rs = ps.executeQuery();

**if** (rs.next()) {

usernameExist = **true**;

}

} **catch** (SQLException ex) {

System.out.println(ex);

}

**return** usernameExist;

}

**private** **boolean** isValidated() {

window = registerButton.getScene().getWindow();

**if** (firstName.getText().equals("")) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"First name text field cannot be blank.");

firstName.requestFocus();

} **else** **if** (firstName.getText().length() < 2 || firstName.getText().length() > 25) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"First name text field cannot be less than 2 and greator than 25 characters.");

firstName.requestFocus();

} **else** **if** (lastName.getText().equals("")) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Last name text field cannot be blank.");

lastName.requestFocus();

} **else** **if** (lastName.getText().length() < 2 || lastName.getText().length() > 25) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Last name text field cannot be less than 2 and greator than 25 characters.");

lastName.requestFocus();

} **else** **if** (email.getText().equals("")) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Email text field cannot be blank.");

email.requestFocus();

} **else** **if** (email.getText().length() < 5 || email.getText().length() > 45) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Email text field cannot be less than 5 and greator than 45 characters.");

email.requestFocus();

} **else** **if** (username.getText().equals("")) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Username text field cannot be blank.");

username.requestFocus();

} **else** **if** (username.getText().length() < 5 || username.getText().length() > 25) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Username text field cannot be less than 5 and greator than 25 characters.");

username.requestFocus();

} **else** **if** (password.getText().equals("")) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Password text field cannot be blank.");

password.requestFocus();

} **else** **if** (password.getText().length() < 5 || password.getText().length() > 25) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Password text field cannot be less than 5 and greator than 25 characters.");

password.requestFocus();

} **else** **if** (confirmPassword.getText().equals("")) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Confirm password text field cannot be blank.");

confirmPassword.requestFocus();

} **else** **if** (confirmPassword.getText().length() < 5 || password.getText().length() > 25) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Confirm password text field cannot be less than 5 and greator than 25 characters.");

confirmPassword.requestFocus();

} **else** **if** (!password.getText().equals(confirmPassword.getText())) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"Password and confirm password text fields does not match.");

password.requestFocus();

} **else** **if** (isAlreadyRegistered()) {

AlertHelper.showAlert(Alert.AlertType.ERROR, window, "Error",

"The username is already taken by someone else.");

username.requestFocus();

} **else** {

**if** (!((String)**this**.cbnTeamButton.getValue()).equals("")) {

**return** **true**;

}

AlertHelper.showAlert(Alert.AlertType.ERROR, **this**.window, "Error", "Team text field cannot be blank.");

}

**return** **false**;

}

**private** **boolean** clearForm() {

firstName.clear();

lastName.clear();

email.clear();

username.clear();

password.clear();

confirmPassword.clear();

**return** **true**;

}

@FXML

**private** **void** showLoginStage() **throws** IOException {

Stage stage = (Stage) registerButton.getScene().getWindow();

stage.close();

Parent root = FXMLLoader.load(getClass().getResource("/view/LoginView.fxml"));

Scene scene = **new** Scene(root);

stage.setScene(scene);

stage.setTitle("User Login");

stage.getIcons().add(**new** Image("/asset/icon.png"));

stage.show();

}

}

HomeView.fxml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<?import javafx.scene.control.TableColumn?>

<?import javafx.scene.control.TableView?>

<?import javafx.scene.layout.AnchorPane?>

<AnchorPane maxHeight=*"-Infinity"* maxWidth=*"-Infinity"* minHeight=*"-Infinity"* minWidth=*"-Infinity"* prefHeight=*"600.0"* prefWidth=*"1200.0"* style=*"-fx-background-color: grey;"* xmlns=*"http://javafx.com/javafx/11.0.1"* xmlns:fx=*"http://javafx.com/fxml/1"* fx:controller=*"controller.Page01Controller"*>

<children>

<TableView fx:id=*"table"* layoutX=*"14.0"* layoutY=*"14.0"* prefHeight=*"300.0"* prefWidth=*"500.0"* AnchorPane.bottomAnchor=*"0.0"* AnchorPane.leftAnchor=*"0.0"* AnchorPane.rightAnchor=*"0.0"* AnchorPane.topAnchor=*"0.0"*>

<columns>

<TableColumn fx:id=*"id"* prefWidth=*"75.0"* text=*"ID"* />

<TableColumn fx:id=*"location"* prefWidth=*"75.0"* text=*"Location"* />

<TableColumn fx:id=*"team1"* prefWidth=*"75.0"* text=*"Team 1"* />

<TableColumn fx:id=*"team2"* prefWidth=*"75.0"* text=*"Team 2"* />

<TableColumn fx:id=*"match\_date"* prefWidth=*"75.0"* text=*"Match Date"* />

<TableColumn fx:id=*"match\_time"* prefWidth=*"75.0"* text=*"Match Time"* />

<TableColumn fx:id=*"match\_outcome"* prefWidth=*"75.0"* text=*"Match Outcome"* />

</columns>

</TableView>

</children>

</AnchorPane>

**LoginView.fxml**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<?import javafx.scene.Cursor?>

<?import javafx.scene.control.Button?>

<?import javafx.scene.control.PasswordField?>

<?import javafx.scene.control.TextField?>

<?import javafx.scene.layout.AnchorPane?>

<?import javafx.scene.layout.BorderPane?>

<?import javafx.scene.text.Font?>

<?import javafx.scene.text.Text?>

<BorderPane maxHeight=*"-Infinity"* maxWidth=*"-Infinity"* minHeight=*"-Infinity"* minWidth=*"-Infinity"* prefHeight=*"400.0"* prefWidth=*"600.0"* xmlns=*"http://javafx.com/javafx/11.0.1"* xmlns:fx=*"http://javafx.com/fxml/1"* fx:controller=*"controller.LoginController"*>

<left>

<AnchorPane prefHeight=*"400.0"* prefWidth=*"300.0"* style=*"-fx-background-color: #404040;"* BorderPane.alignment=*"CENTER"*>

<children>

<Text fill=*"WHITE"* layoutX=*"24.0"* layoutY=*"184.0"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"IPL"* wrappingWidth=*"263.51171875"*>

<font>

<Font name=*"System Bold"* size=*"54.0"* />

</font>

</Text>

<Text fill=*"WHITE"* layoutX=*"82.0"* layoutY=*"225.0"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"Indian Premier League"*>

<font>

<Font size=*"18.0"* />

</font>

</Text>

</children>

</AnchorPane>

</left>

<right>

<AnchorPane prefHeight=*"400.0"* prefWidth=*"300.0"* BorderPane.alignment=*"CENTER"*>

<children>

<TextField fx:id=*"username"* layoutX=*"49.0"* layoutY=*"163.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Username"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<PasswordField fx:id=*"password"* layoutX=*"49.0"* layoutY=*"202.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Password"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<Button fx:id=*"loginButton"* layoutX=*"112.0"* layoutY=*"255.0"* mnemonicParsing=*"false"* onAction=*"#login"* prefHeight=*"32.0"* prefWidth=*"76.0"* style=*"-fx-background-color: #404040; -fx-background-radius: 0px;"* text=*"Login"* textFill=*"WHITE"*>

<cursor>

<Cursor fx:constant=*"OPEN\_HAND"* />

</cursor></Button>

<Text fill=*"#404040"* layoutX=*"81.0"* layoutY=*"143.0"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"User Login"* wrappingWidth=*"137.24869537353516"*>

<font>

<Font name=*"Wingdings 3"* size=*"27.0"* />

</font>

</Text>

<Text fill=*"#404040"* layoutX=*"65.0"* layoutY=*"313.0"* onMouseClicked=*"#showRegisterStage"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"Don't have an account? Register"*>

<cursor>

<Cursor fx:constant=*"HAND"* />

</cursor>

</Text>

</children>

</AnchorPane>

</right>

</BorderPane>

**MainPanelView.fxml**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<?import javafx.geometry.Insets?>

<?import javafx.scene.Cursor?>

<?import javafx.scene.control.Button?>

<?import javafx.scene.layout.BorderPane?>

<?import javafx.scene.layout.HBox?>

<?import javafx.scene.layout.VBox?>

<?import javafx.scene.text.Font?>

<BorderPane fx:id=*"borderPane"* maxHeight=*"-Infinity"* maxWidth=*"-Infinity"* minHeight=*"-Infinity"* minWidth=*"-Infinity"* prefHeight=*"640.0"* prefWidth=*"1350.0"* style=*"-fx-background-color: grey;"* xmlns=*"http://javafx.com/javafx/11.0.1"* xmlns:fx=*"http://javafx.com/fxml/1"* fx:controller=*"controller.MainPanelController"*>

<left>

<VBox prefHeight=*"601.0"* prefWidth=*"160.0"* style=*"-fx-background-color: #404040;"* BorderPane.alignment=*"CENTER"*>

<children>

<Button fx:id=*"addPurchase"* mnemonicParsing=*"false"* onAction=*"#loadHomeView"* prefHeight=*"30.0"* prefWidth=*"189.0"* style=*"-fx-text-fill: #f0f0f0;"* styleClass=*"bt"* stylesheets=*"@mainpanel.css"* textAlignment=*"JUSTIFY"* visible=*"false"*>

<VBox.margin>

<Insets top=*"10.0"* />

</VBox.margin>

<font>

<Font name=*"System Bold"* size=*"12.0"* />

</font>

</Button>

<Button fx:id=*"purchaseDetail"* mnemonicParsing=*"false"* onAction=*"#loadHomeView"* prefHeight=*"30.0"* prefWidth=*"185.0"* style=*"-fx-text-fill: #f0f0f0;"* styleClass=*"bt"* stylesheets=*"@mainpanel.css"* textAlignment=*"JUSTIFY"* visible=*"false"*>

<font>

<Font name=*"System Bold"* size=*"12.0"* />

</font>

</Button>

<Button fx:id=*"home"* mnemonicParsing=*"false"* onAction=*"#loadHomeView"* prefHeight=*"30.0"* prefWidth=*"173.0"* style=*"-fx-text-fill: #f0f0f0;"* styleClass=*"bt"* stylesheets=*"@mainpanel.css"* text=*"Home"* textAlignment=*"JUSTIFY"*>

<font>

<Font name=*"System Bold"* size=*"17.0"* />

</font>

<cursor>

<Cursor fx:constant=*"HAND"* />

</cursor>

</Button>

<!--Button fx:id="page01" mnemonicParsing="false" onAction="#loadPage01View" prefHeight="30.0" prefWidth="172.0" style="-fx-text-fill: #f0f0f0;" styleClass="bt" stylesheets="@mainpanel.css" text="Page 01" textAlignment="JUSTIFY">

<font>

<Font name="System Bold" size="17.0" />

</font>

<cursor>

<Cursor fx:constant="HAND" />

</cursor>

</Button-->

<Button fx:id=*"page02"* mnemonicParsing=*"false"* onAction=*"#loadPage02View"* prefHeight=*"30.0"* prefWidth=*"168.0"* style=*"-fx-text-fill: #f0f0f0;"* styleClass=*"bt"* stylesheets=*"@mainpanel.css"* text=*"Match Schedule"* textAlignment=*"JUSTIFY"*>

<font>

<Font name=*"System Bold"* size=*"17.0"* />

</font>

<cursor>

<Cursor fx:constant=*"HAND"* />

</cursor>

</Button>

<!--Button fx:id="page03" mnemonicParsing="false" onAction="#loadPage03View" prefHeight="30.0" prefWidth="168.0" style="-fx-text-fill: #f0f0f0;" styleClass="bt" stylesheets="@mainpanel.css" text="Page 03" textAlignment="JUSTIFY">

<font>

<Font name="System Bold" size="17.0" />

</font>

<cursor>

<Cursor fx:constant="HAND" />

</cursor>

</Button-->

<Button fx:id=*"page04"* mnemonicParsing=*"false"* onAction=*"#loadPage04View"* prefHeight=*"30.0"* prefWidth=*"168.0"* style=*"-fx-text-fill: #f0f0f0;"* styleClass=*"bt"* stylesheets=*"@mainpanel.css"* text=*"Add Team"* textAlignment=*"JUSTIFY"*>

<font>

<Font name=*"System Bold"* size=*"17.0"* />

</font>

<cursor>

<Cursor fx:constant=*"HAND"* />

</cursor>

</Button>

<!--Button fx:id="page05" mnemonicParsing="false" onAction="#loadPage05View" prefHeight="30.0" prefWidth="168.0" style="-fx-text-fill: #f0f0f0;" styleClass="bt" stylesheets="@mainpanel.css" text="Page 05" textAlignment="JUSTIFY">

<font>

<Font name="System Bold" size="17.0" />

</font>

<cursor>

<Cursor fx:constant="HAND" />

</cursor>

</Button>

<Button fx:id="page06" mnemonicParsing="false" onAction="#loadPage06View" prefHeight="30.0" prefWidth="168.0" style="-fx-text-fill: #f0f0f0;" styleClass="bt" stylesheets="@mainpanel.css" text="Page 06" textAlignment="JUSTIFY">

<font>

<Font name="System Bold" size="17.0" />

</font>

<cursor>

<Cursor fx:constant="HAND" />

</cursor>

</Button>

<Button fx:id="page07" mnemonicParsing="false" onAction="#loadPage07View" prefHeight="30.0" prefWidth="168.0" style="-fx-text-fill: #f0f0f0;" styleClass="bt" stylesheets="@mainpanel.css" text="Page 07" textAlignment="JUSTIFY">

<font>

<Font name="System Bold" size="17.0" />

</font>

<cursor>

<Cursor fx:constant="HAND" />

</cursor>

</Button>

<Button fx:id="page08" mnemonicParsing="false" onAction="#loadPage08View" prefHeight="30.0" prefWidth="168.0" style="-fx-text-fill: #f0f0f0;" styleClass="bt" stylesheets="@mainpanel.css" text="Page 08" textAlignment="JUSTIFY">

<font>

<Font name="System Bold" size="17.0" />

</font>

<cursor>

<Cursor fx:constant="HAND" />

</cursor>

</Button>

<Button fx:id="page09" mnemonicParsing="false" onAction="#loadPage09View" prefHeight="30.0" prefWidth="168.0" style="-fx-text-fill: #f0f0f0;" styleClass="bt" stylesheets="@mainpanel.css" text="Page 09" textAlignment="JUSTIFY">

<font>

<Font name="System Bold" size="17.0" />

</font>

<cursor>

<Cursor fx:constant="HAND" />

</cursor>

</Button>

<Button fx:id="page10" mnemonicParsing="false" onAction="#loadPage10View" prefHeight="30.0" prefWidth="168.0" style="-fx-text-fill: #f0f0f0;" styleClass="bt" stylesheets="@mainpanel.css" text="Page 10" textAlignment="JUSTIFY">

<font>

<Font name="System Bold" size="17.0" />

</font>

<cursor>

<Cursor fx:constant="HAND" />

</cursor>

</Button-->

</children>

</VBox>

</left>

<top>

<HBox alignment=*"CENTER\_RIGHT"* prefHeight=*"39.0"* prefWidth=*"1360.0"* style=*"-fx-background-color: #2b2a26;"* BorderPane.alignment=*"CENTER"*>

<children>

<Button fx:id=*"logout"* mnemonicParsing=*"false"* onAction=*"#close"* prefHeight=*"18.0"* prefWidth=*"274.0"* style=*"-fx-text-fill: #f0f0f0;"* styleClass=*"bt"* stylesheets=*"@mainpanel.css"* text=*"Log Out"* textAlignment=*"JUSTIFY"*>

<font>

<Font name=*"System Bold"* size=*"12.0"* />

</font>

<opaqueInsets>

<Insets />

</opaqueInsets>

<HBox.margin>

<Insets right=*"4.0"* />

</HBox.margin>

<cursor>

<Cursor fx:constant=*"HAND"* />

</cursor>

</Button>

</children>

<BorderPane.margin>

<Insets right=*"5.0"* />

</BorderPane.margin></HBox>

</top>

<center>

<VBox prefHeight=*"200.0"* prefWidth=*"100.0"* BorderPane.alignment=*"CENTER"*>

<children>

<HBox prefHeight=*"108.0"* prefWidth=*"1190.0"* />

<HBox prefHeight=*"244.0"* prefWidth=*"1190.0"* />

<HBox prefHeight=*"254.0"* prefWidth=*"1190.0"* />

</children>

</VBox>

</center>

</BorderPane>

Page01View.fxml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<?import javafx.scene.control.Label?>

<?import javafx.scene.layout.AnchorPane?>

<?import javafx.scene.text.Font?>

<AnchorPane maxHeight=*"-Infinity"* maxWidth=*"-Infinity"* minHeight=*"-Infinity"* minWidth=*"-Infinity"* prefHeight=*"600.0"* prefWidth=*"1200.0"* style=*"-fx-background-color: grey;"* xmlns=*"http://javafx.com/javafx/11.0.1"* xmlns:fx=*"http://javafx.com/fxml/1"* fx:controller=*"controller.Page01Controller"*>

<children>

<Label alignment=*"CENTER"* contentDisplay=*"CENTER"* layoutX=*"440.0"* layoutY=*"240.0"* text=*"Page 01"* textFill=*"WHITE"*>

<font>

<Font size=*"96.0"* />

</font>

</Label>

</children>

</AnchorPane>

Page02View.fxml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<?import javafx.scene.control.Label?>

<?import javafx.scene.layout.AnchorPane?>

<?import javafx.scene.text.Font?>

<?import javafx.scene.control.ComboBox?>

<?import javafx.scene.Cursor?>

<?import javafx.scene.control.Button?>

<?import javafx.scene.control.PasswordField?>

<?import javafx.scene.control.TextField?>

<?import javafx.scene.layout.BorderPane?>

<?import javafx.scene.text.Text?>

<BorderPane maxHeight=*"-Infinity"* maxWidth=*"-Infinity"* minHeight=*"-Infinity"* minWidth=*"-Infinity"* prefHeight=*"400.0"* prefWidth=*"600.0"* xmlns=*"http://javafx.com/javafx/11.0.1"* xmlns:fx=*"http://javafx.com/fxml/1"* fx:controller=*"controller.Page02Controller"*>

<!--left>

<AnchorPane prefHeight="400.0" prefWidth="300.0" style="-fx-background-color: #404040;" BorderPane.alignment="CENTER">

<children>

<Text fill="WHITE" layoutX="24.0" layoutY="184.0" strokeType="OUTSIDE" strokeWidth="0.0" text="IPL" wrappingWidth="263.51171875">

<font>

<Font name="System Bold" size="54.0" />

</font>

</Text>

<Text fill="WHITE" layoutX="82.0" layoutY="220.0" strokeType="OUTSIDE" strokeWidth="0.0" text="Indian Premier League">

<font>

<Font size="18.0" />

</font>

</Text>

</children>

</AnchorPane>

</left-->

<center>

<AnchorPane prefHeight=*"400.0"* prefWidth=*"300.0"* BorderPane.alignment=*"CENTER"*>

<children>

<Text fill=*"#FFFFFF"* layoutX=*"47.0"* layoutY=*"53.0"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"Match Schedule"* wrappingWidth=*"206.24869537353516"*>

<font>

<Font name=*"Wingdings 3"* size=*"27.0"* />

</font>

</Text>

<TextField fx:id=*"location"* layoutX=*"49.0"* layoutY=*"76.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Location"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<ComboBox id=*""* fx:id=*"cbnTeam1Button"* layoutX=*"49.0"* layoutY=*"113.0"* minHeight=*"22.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Team"* styleClass=*"ComboBox"*>

</ComboBox>

<ComboBox id=*""* fx:id=*"cbnTeam2Button"* layoutX=*"49.0"* layoutY=*"147.0"* minHeight=*"22.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Team"* styleClass=*"ComboBox"*>

</ComboBox>

<TextField fx:id=*"match\_date"* layoutX=*"49.0"* layoutY=*"183.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Match Date"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<TextField fx:id=*"match\_time"* layoutX=*"49.0"* layoutY=*"219.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Match Time"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<TextField fx:id=*"match\_outcome"* layoutX=*"49.0"* layoutY=*"254.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Match Outcome"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<Button fx:id=*"insertmatchButton"* layoutX=*"112.0"* layoutY=*"324.0"* mnemonicParsing=*"false"* onAction=*"#insertmatch"* prefHeight=*"32.0"* prefWidth=*"76.0"* style=*"-fx-background-color: #404040; -fx-background-radius: 0px;"* text=*"Submit"* textFill=*"WHITE"*>

<cursor>

<Cursor fx:constant=*"HAND"* />

</cursor></Button>

</children>

</AnchorPane>

</center>

</BorderPane>

Page03View.fxml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<?import javafx.scene.control.Label?>

<?import javafx.scene.layout.AnchorPane?>

<?import javafx.scene.text.Font?>

<AnchorPane maxHeight=*"-Infinity"* maxWidth=*"-Infinity"* minHeight=*"-Infinity"* minWidth=*"-Infinity"* prefHeight=*"600.0"* prefWidth=*"1200.0"* style=*"-fx-background-color: grey;"* xmlns=*"http://javafx.com/javafx/11.0.1"* xmlns:fx=*"http://javafx.com/fxml/1"* fx:controller=*"controller.Page03Controller"*>

<children>

<Label alignment=*"CENTER"* contentDisplay=*"CENTER"* layoutX=*"340.0"* layoutY=*"240.0"* text=*"You do not have access"* textFill=*"WHITE"*>

<font>

<Font size=*"60.0"* />

</font>

</Label>

</children>

</AnchorPane>

Page04View.fxml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<?import javafx.scene.control.Label?>

<?import javafx.scene.layout.AnchorPane?>

<?import javafx.scene.text.Font?>

<?import javafx.scene.Cursor?>

<?import javafx.scene.control.Button?>

<?import javafx.scene.control.PasswordField?>

<?import javafx.scene.control.TextField?>

<?import javafx.scene.layout.BorderPane?>

<?import javafx.scene.text.Text?>

<BorderPane maxHeight=*"-Infinity"* maxWidth=*"-Infinity"* minHeight=*"-Infinity"* minWidth=*"-Infinity"* prefHeight=*"400.0"* prefWidth=*"600.0"* style=*"-fx-background-color: grey;"* xmlns=*"http://javafx.com/javafx/11.0.1"* xmlns:fx=*"http://javafx.com/fxml/1"* fx:controller=*"controller.Page04Controller"*>

<!--children>

<Label alignment="CENTER" contentDisplay="CENTER" layoutX="440.0" layoutY="240.0" text="Page 04" textFill="WHITE">

<font>

<Font size="96.0" />

</font>

</Label>

</children>

</AnchorPane-->

<center>

<AnchorPane prefHeight=*"400.0"* prefWidth=*"300.0"* BorderPane.alignment=*"CENTER"*>

<children>

<Text fill=*"#FFFFFF"* layoutX=*"47.0"* layoutY=*"53.0"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"Add Team"* wrappingWidth=*"206.24869537353516"*>

<font>

<Font name=*"Wingdings 3"* size=*"27.0"* />

</font>

</Text>

<TextField fx:id=*"team\_name"* layoutX=*"49.0"* layoutY=*"76.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Team Name"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<TextField fx:id=*"team\_code"* layoutX=*"49.0"* layoutY=*"113.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Team Code"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<TextField fx:id=*"team\_captain"* layoutX=*"49.0"* layoutY=*"147.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Team Captain"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<Button fx:id=*"insertteamButton"* layoutX=*"112.0"* layoutY=*"219.0"* mnemonicParsing=*"false"* onAction=*"#insertteam"* prefHeight=*"32.0"* prefWidth=*"76.0"* style=*"-fx-background-color: #404040; -fx-background-radius: 0px;"* text=*"Submit"* textFill=*"WHITE"*>

<cursor>

<Cursor fx:constant=*"HAND"* />

</cursor></Button>

</children>

</AnchorPane>

</center>

</BorderPane>

Page05View.fxml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<?import javafx.scene.control.Label?>

<?import javafx.scene.layout.AnchorPane?>

<?import javafx.scene.text.Font?>

<AnchorPane maxHeight=*"-Infinity"* maxWidth=*"-Infinity"* minHeight=*"-Infinity"* minWidth=*"-Infinity"* prefHeight=*"600.0"* prefWidth=*"1200.0"* style=*"-fx-background-color: grey;"* xmlns=*"http://javafx.com/javafx/11.0.1"* xmlns:fx=*"http://javafx.com/fxml/1"* fx:controller=*"controller.Page05Controller"*>

<children>

<Label alignment=*"CENTER"* contentDisplay=*"CENTER"* layoutX=*"440.0"* layoutY=*"240.0"* text=*"Page 05"* textFill=*"WHITE"*>

<font>

<Font size=*"96.0"* />

</font>

</Label>

</children>

</AnchorPane>

RegisterView.fxml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<?import javafx.scene.Cursor?>

<?import javafx.scene.control.Button?>

<?import javafx.scene.control.PasswordField?>

<?import javafx.scene.control.TextField?>

<?import javafx.scene.control.ComboBox?>

<?import javafx.scene.layout.AnchorPane?>

<?import javafx.scene.layout.BorderPane?>

<?import javafx.scene.text.Font?>

<?import javafx.scene.text.Text?>

<BorderPane maxHeight=*"-Infinity"* maxWidth=*"-Infinity"* minHeight=*"-Infinity"* minWidth=*"-Infinity"* prefHeight=*"400.0"* prefWidth=*"600.0"* xmlns=*"http://javafx.com/javafx/11.0.1"* xmlns:fx=*"http://javafx.com/fxml/1"* fx:controller=*"controller.RegisterController"*>

<left>

<AnchorPane prefHeight=*"400.0"* prefWidth=*"300.0"* style=*"-fx-background-color: #404040;"* BorderPane.alignment=*"CENTER"*>

<children>

<Text fill=*"WHITE"* layoutX=*"24.0"* layoutY=*"184.0"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"IPL"* wrappingWidth=*"263.51171875"*>

<font>

<Font name=*"System Bold"* size=*"54.0"* />

</font>

</Text>

<Text fill=*"WHITE"* layoutX=*"82.0"* layoutY=*"220.0"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"Indian Premier League"*>

<font>

<Font size=*"18.0"* />

</font>

</Text>

</children>

</AnchorPane>

</left>

<right>

<AnchorPane prefHeight=*"400.0"* prefWidth=*"300.0"* BorderPane.alignment=*"CENTER"*>

<children>

<Text fill=*"#404040"* layoutX=*"47.0"* layoutY=*"53.0"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"User Registration"* wrappingWidth=*"206.24869537353516"*>

<font>

<Font name=*"Wingdings 3"* size=*"27.0"* />

</font>

</Text>

<TextField fx:id=*"firstName"* layoutX=*"49.0"* layoutY=*"76.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"First Name"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<TextField fx:id=*"lastName"* layoutX=*"49.0"* layoutY=*"113.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Last Name"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<TextField fx:id=*"email"* layoutX=*"49.0"* layoutY=*"147.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Email"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<TextField fx:id=*"username"* layoutX=*"49.0"* layoutY=*"183.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Username"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<PasswordField fx:id=*"password"* layoutX=*"49.0"* layoutY=*"219.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Password"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<PasswordField fx:id=*"confirmPassword"* layoutX=*"49.0"* layoutY=*"254.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Confirm Password"* style=*"-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;"* />

<!--TextField fx:id="team" layoutX="49.0" layoutY="289.0" prefHeight="25.0" prefWidth="202.0" promptText="team" style="-fx-background-color: transparent; -fx-border-color: #404040; -fx-border-width: 0px 0px 2px 0px;" /-->

<ComboBox id=*""* fx:id=*"cbnTeamButton"* layoutX=*"49.0"* layoutY=*"289.0"* minHeight=*"22.0"* prefHeight=*"25.0"* prefWidth=*"202.0"* promptText=*"Team"* styleClass=*"ComboBox"*>

</ComboBox>

<Button fx:id=*"registerButton"* layoutX=*"112.0"* layoutY=*"324.0"* mnemonicParsing=*"false"* onAction=*"#register"* prefHeight=*"32.0"* prefWidth=*"76.0"* style=*"-fx-background-color: #404040; -fx-background-radius: 0px;"* text=*"Submit"* textFill=*"WHITE"*>

<cursor>

<Cursor fx:constant=*"HAND"* />

</cursor></Button>

<Text fill=*"#404040"* layoutX=*"66.0"* layoutY=*"368.0"* onMouseClicked=*"#showLoginStage"* strokeType=*"OUTSIDE"* strokeWidth=*"0.0"* text=*"Already have an account? Login"*>

<cursor>

<Cursor fx:constant=*"HAND"* />

</cursor>

</Text>

</children>

</AnchorPane>

</right>

</BorderPane>

AlertHelper.java:

package helper;

import java.util.Optional;

import javafx.event.ActionEvent;

import javafx.event.EventHandler;

import javafx.geometry.Pos;

import javafx.scene.control.Alert;

import javafx.scene.control.ButtonBar;

import javafx.scene.control.ButtonType;

import javafx.scene.control.DialogPane;

import javafx.scene.paint.Color;

import javafx.scene.shape.Rectangle;

import javafx.stage.Window;

import javafx.util.Duration;

import org.controlsfx.control.Notifications;

public class AlertHelper {

public static boolean result = false;

public static void showAlert(Alert.AlertType alertType, Window owner, String title, String message) {

if (alertType.name() == null ? alertType.INFORMATION.name() == null : alertType.name().equals(alertType.INFORMATION.name())) {

Notifications.create()

.darkStyle()

.title(title)

.text(message).hideAfter(Duration.seconds(10))

.showInformation();

} else if (alertType.name() == null ? Alert.AlertType.ERROR.name() == null : alertType.name().equals(Alert.AlertType.ERROR.name())) {

Notifications.create()

.darkStyle()

.title(title)

.text(message).hideAfter(Duration.seconds(10))

.showError();

}

}

}