

Nathan Gopee

CPS 293

Professor Suchy

2 October 2024

Task 1- Java Practice

1. Get a list of players names and shirt numbers that play on the team with id = 1
import java.sql.*;

```
public class Soccer_dbDatabase {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/soccer_db";
        String user = "root";
        String password = " pass ";

        try (Connection conn = DriverManager.getConnection(url, user,
password);
            Statement stmt = conn.createStatement()) {

            String query = "SELECT firstName, lastName, shirtNumber
FROM PLAYER WHERE teamId = 1;";
            ResultSet rs = stmt.executeQuery(query);

            while (rs.next()) {
                System.out.println(rs.getString("firstName") + " " +
rs.getString("lastName") + " - Shirt Number: " +
rs.getInt("shirtNumber"));
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

2. Get a list of players names and shirt numbers that play on the team named
"Arsenal" and display them in alphabetic order based on players last name

```
String query = "SELECT firstName, lastName, shirtNumber FROM PLAYER " +
"JOIN TEAM ON PLAYER.teamId = TEAM.teamId " +
"WHERE TEAM.name = 'Arsenal' ORDER BY PLAYER.lastName ASC;";
```

3. Get a list of players names and shirt numbers that play on the team named
"Arsenal" and display them in numeric order based on players shirt number

```
String query = "SELECT firstName, lastName, shirtNumber FROM PLAYER " +
    "JOIN TEAM ON PLAYER.teamId = TEAM.teamId " +
    "WHERE TEAM.name = 'Arsenal' ORDER BY PLAYER.shirtNumber ASC;";
```

4. Update a players team to be any team you would like. ex. update player with player id =5 to be on team "Aston Villa"

```
String updateQuery = "UPDATE PLAYER SET teamId = (SELECT teamId FROM
TEAM WHERE name = 'Aston Villa') WHERE playerId = 5;";
stmt.executeUpdate(updateQuery);
```

5. Add a new team to the database. You may research a real team or make up your own. Form statements to add 3 new players to the new team.

```
String insertTeamQuery = "INSERT INTO TEAM (name, stadium, city) VALUES
('Newcastle Jets', 'McDonald Jones Stadium', 'Newcastle');";
stmt.executeUpdate(insertTeamQuery);
```

```
String insertPlayersQuery = "INSERT INTO PLAYER (teamId, firstName,
lastName, dob, startYear, shirtNumber) " +
    "VALUES ((SELECT teamId FROM TEAM WHERE
name = 'Newcastle Jets'), 'Player1_First', 'Player1_Last', '1995-01-
01', 2020, 10), " +
    "((SELECT teamId FROM TEAM WHERE name =
'Newcastle Jets'), 'Player2_First', 'Player2_Last', '1997-05-12', 2019,
9), " +
    "((SELECT teamId FROM TEAM WHERE name =
'Newcastle Jets'), 'Player3_First', 'Player3_Last', '1999-11-20', 2021,
7);";
stmt.executeUpdate(insertPlayersQuery);
```

6. Write a program that asks the user using keyboard or command line arguments to enter the name of a team that they want the players for.

```
import java.sql.*;
import java.util.Scanner;

public class Soccer_dbDatabase {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/soccer_db";
        String user = "root";
        String password = " pass ";

        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the name of the team: ");
        String teamName = scanner.nextLine();
```

```

        try (Connection conn = DriverManager.getConnection(url, user,
password));
            PreparedStatement pstmt = conn.prepareStatement(
                "SELECT firstName, lastName, shirtNumber FROM PLAYER "
+
                "JOIN TEAM ON PLAYER.teamId = TEAM.teamId WHERE
TEAM.name = ?")) {

                pstmt.setString(1, teamName);
                ResultSet rs = pstmt.executeQuery();

                while (rs.next()) {
                    System.out.println(rs.getString("firstName") + " " +
rs.getString("lastName") + " - Shirt Number: " +
rs.getInt("shirtNumber"));
                }
            } catch (SQLException e) {
                e.printStackTrace();
            }
        }
    }
}

```

7. Write a Java class that captures the nature of a player. Write a method to create a list or array of player objects where each object has been populated with data from the database. Write a method to return all the players that have shirt number x, where x is a parameter.

```

import java.sql.*;
import java.util.ArrayList;
import java.util.List;

public class Soccer_dbDatabase {
    public static List<Player> getPlayersFromDatabase() {
        List<Player> players = new ArrayList<>();
        String url = "jdbc:mysql://localhost:3306/soccer_db";
        String user = "root";
        String password = " pass ";

        try (Connection conn = DriverManager.getConnection(url, user,
password));
            Statement stmt = conn.createStatement()) {

                ResultSet rs = stmt.executeQuery("SELECT firstName,
lastName, shirtNumber FROM PLAYER");

                while (rs.next()) {
                    Player player = new Player(rs.getString("firstName"),
rs.getString("lastName"), rs.getInt("shirtNumber"));

```

```

        players.add(player);
    }
} catch (SQLException e) {
    e.printStackTrace();
}
return players;
}

public static List<Player> getPlayersByShirtNumber(int shirtNumber)
{
    List<Player> filteredPlayers = new ArrayList<>();
    for (Player player : getPlayersFromDatabase()) {
        if (player.getShirtNumber() == shirtNumber) {
            filteredPlayers.add(player);
        }
    }
    return filteredPlayers;
}

public static void main(String[] args) {
    List<Player> players = getPlayersByShirtNumber(10);
    for (Player player : players) {
        System.out.println(player);
    }
}

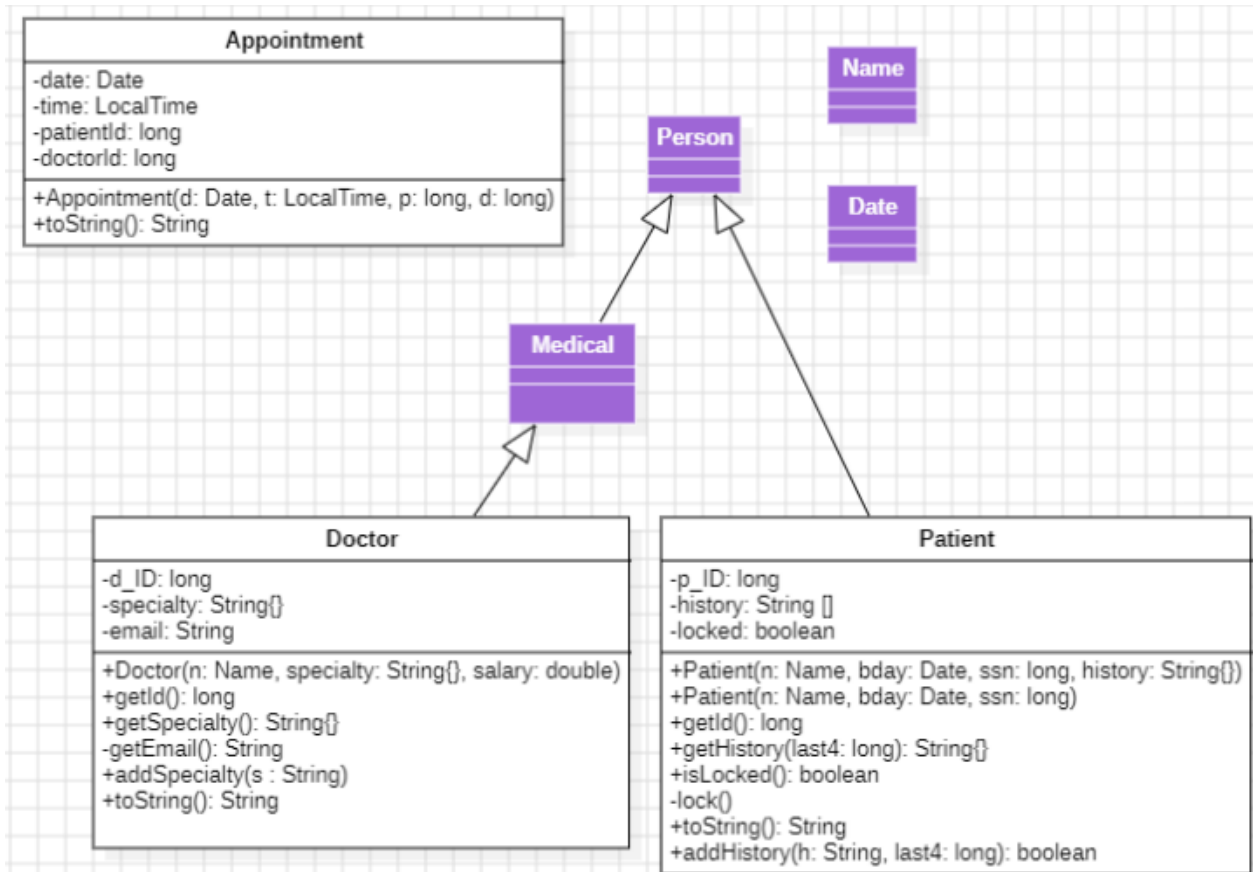
public class Player {
    private String firstName;
    private String lastName;
    private int shirtNumber;

    public Player(String firstName, String lastName, int shirtNumber) {
        this.firstName = firstName;
        this.lastName = lastName;
        this.shirtNumber = shirtNumber;
    }

    public String getFirstName() { return firstName; }
    public String getLastName() { return lastName; }
    public int getShirtNumber() { return shirtNumber; }

    @Override
    public String toString() {
        return firstName + " " + lastName + " - Shirt Number: " +
shirtNumber;
    }
}
}

```



```

mysql> USE medical_db;
Database changed
mysql> SHOW TABLES;
+-----+
| Tables_in_medical_db |
+-----+
| appointment          |
| doctor               |
| patient              |
+-----+
3 rows in set (0.00 sec)
  
```