DBMS - Mini Project
Football Player Statistics Management

Submitted By: Name: Aditya Warrier
SRN:PES2UG20CS026
V Semester Section A

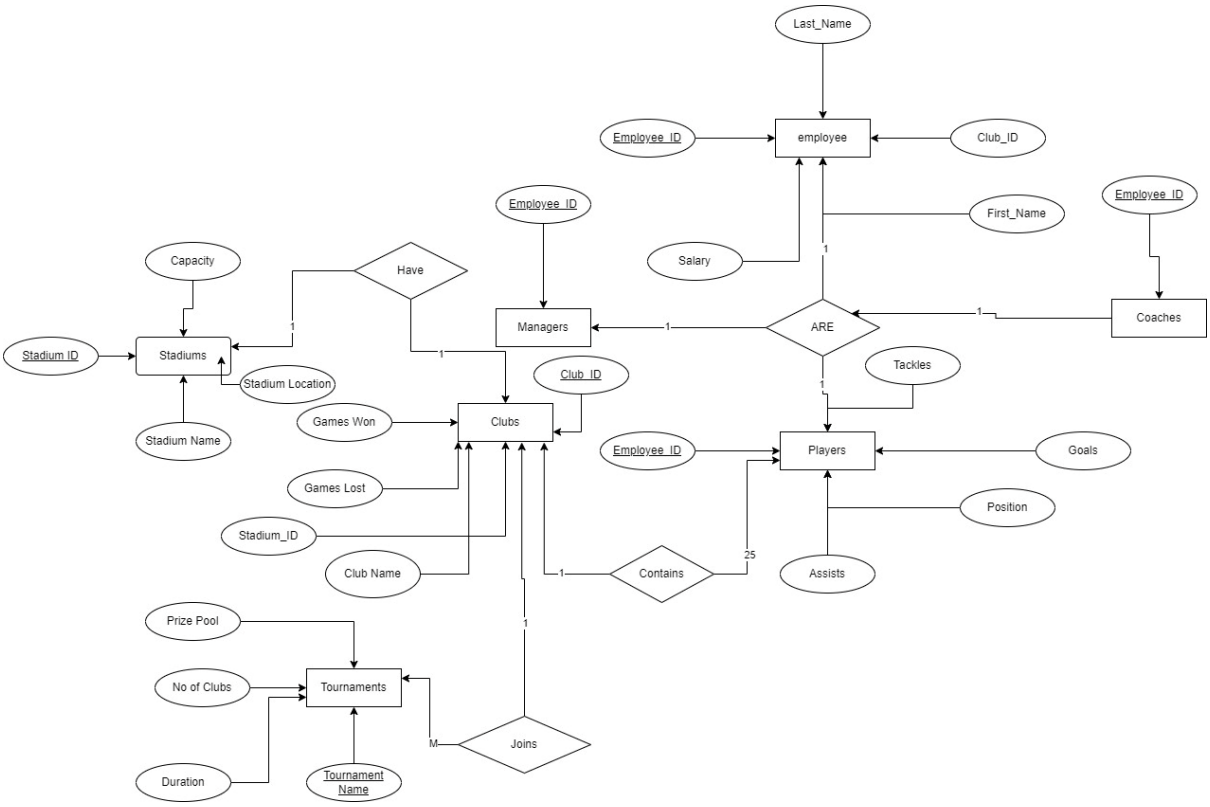
Short Description and Scope of the Project

Easy to access and detailed system to maintain the Statistics of Football Player League for transferring of football players and for checking the Statistics required to decide the winner of trophies. For Insights on which Player has scored the highest goals or has highest amount of assists or to check the players of a certain Club the system is used. A club owner can use this to buy the best players suited to their club.

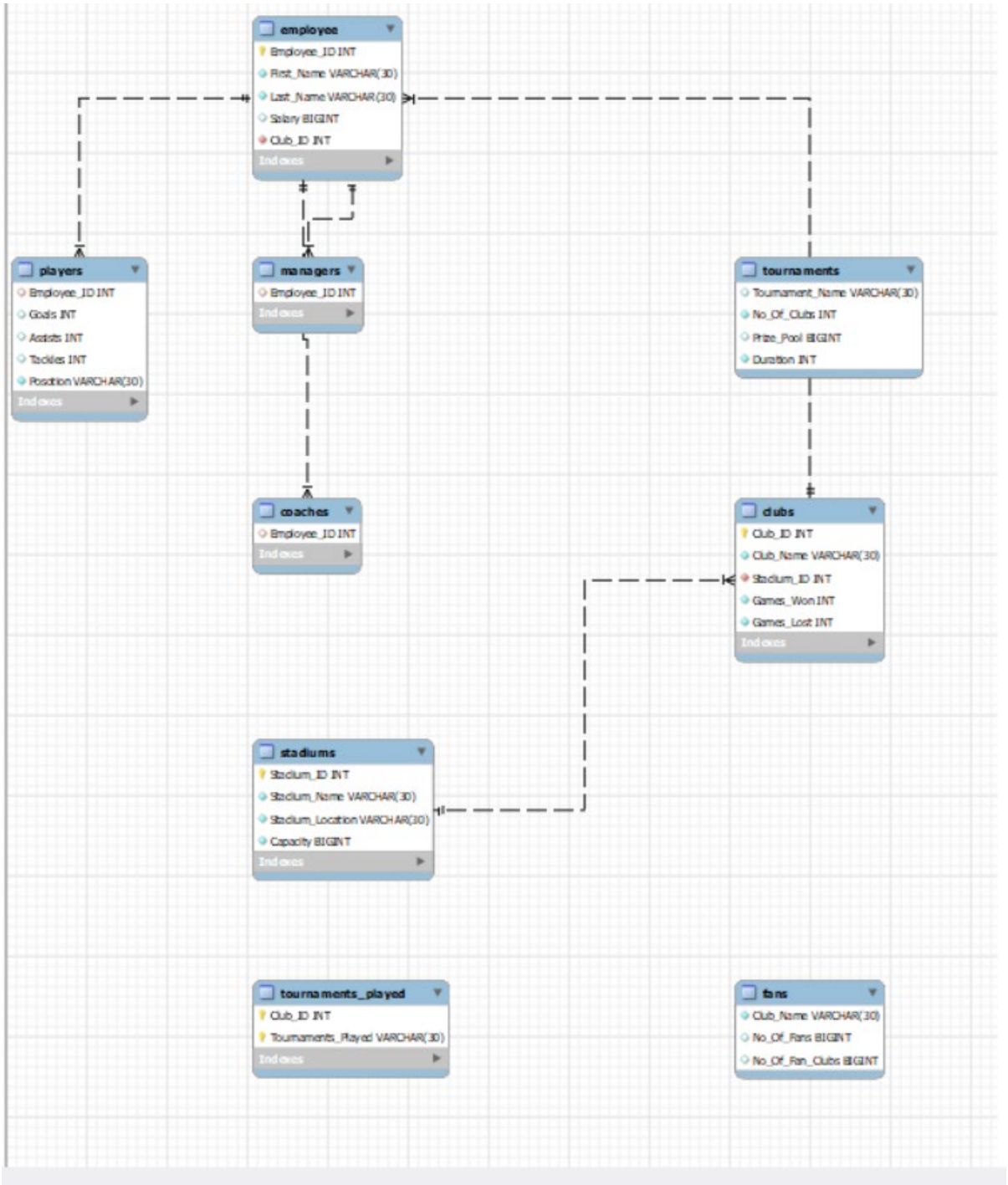
Scope:

- Stores Goals Scored by Players
- Stores Assists by Players
- Stores Clubs of Players
- Stores Managers
- Stores Coaches
- Stores Trophies won
- Insertion, Deletion , Editing , Filtering and Searching of Statistics of Players

ER Diagram


Last_Name
Employee_ID
employee
Club_ID
Employee_ID
First_Name
Employee_ID
Salary
1
Capacity
Have
Managers
1
ARE
1
Coaches
Stadium ID
Stadiums
1
1
Tackles
Stadium Location
Club_ID
1
Stadium Name
Games Won
Clubs
Employee_ID
Players
Goals
Games Lost
Position
Stadium_ID
Club Name
1
Contains
25
Assists
Prize Pool
1
No of Clubs
Tournaments
M
Joins
Duration
Tournament Name

Relational Schema



DDL statements - Building the database

```
CREATE DATABASE IF NOT EXISTS Football_League;
use Football_League;
drop table if exists Players;
drop table if exists Managers;
drop table if exists Coaches;
drop table if exists Tournaments_Played;
drop table if exists Fans;
drop table if exists Tournaments;
drop table if exists Employee;
drop table if exists Clubs;
drop table if exists Stadiums;
create table Stadiums(
        Stadium_ID INT NOT NULL,
        Stadium_Name VARCHAR(30) NOT NULL,
   Stadium_Location VARCHAR(30) NOT NULL,
        Capacity BIGINT NOT NULL,
        PRIMARY KEY (Stadium_ID)
);
create table Clubs(
        Club_ID INT NOT NULL,
        Club_Name VARCHAR(30) NOT NULL,
        Stadium_ID INT NOT NULL,
        Games_Won INT NOT NULL,
        Games_Lost INT NOT NULL,
        PRIMARY KEY (Club_ID),
   FOREIGN KEY (Stadium_ID) REFERENCES Stadiums(Stadium_ID)
);
create table Employee(
        Employee_ID INT NOT NULL,
        First_Name VARCHAR(30) NOT NULL,
        Last_Name VARCHAR(30) NOT NULL,
        Salary BIGINT,
        Club_ID INT NOT NULL,
        PRIMARY KEY (Employee_ID),
   FOREIGN KEY (Club_ID) REFERENCES Clubs(Club_ID)
);
create table Tournaments_Played(
        Club_ID INT NOT NULL,
        Tournaments_Played VARCHAR(30) NOT NULL,
        PRIMARY KEY(Club_ID, Tournaments_Played)
);
create table Fans(
        Club_Name VARCHAR(30) NOT NULL,
        No_Of_Fans BIGINT,
        No_Of_Fan_Clubs BIGINT
);
create table Tournaments(
        Tournament_Name VARCHAR(30),
        No_Of_Clubs INT NOT NULL,
        Prize_Pool BIGINT,
        Duration INT NOT NULL
);
create table Players(
```

```
        Employee_ID INT,
        Goals INT,
    Assists INT,
    Tackles INT,
    Posotion VARCHAR(30) NOT NULL,
        FOREIGN KEY (Employee_ID) REFERENCES Employee(Employee_ID)
);
create table Managers(
        Employee_ID INT,
        FOREIGN KEY (Employee_ID) REFERENCES Employee(Employee_ID)
);
create table Coaches(
        Employee_ID INT,
        FOREIGN KEY (Employee_ID) REFERENCES Employee(Employee_ID)
);
```

Populating the Database

```
INSERT INTO Stadiums VALUES (1,'Trafford','Manchester',19000);
INSERT INTO Stadiums VALUES (2,'Emirates','Arsenal',20000);
INSERT INTO Stadiums VALUES (3,'CampNou','Barcelona',21000);


INSERT INTO Clubs VALUES (1,'ManU',1,2,3);
INSERT INTO Clubs VALUES (2,'Arsenal',2,0,0);
INSERT INTO Clubs VALUES (3,'Barca',3,1,2);

INSERT INTO Employee VALUES (1,'Romelu','Lukaku',102383,1);
INSERT INTO Employee VALUES (2,'Jorjo','Mason',102343,2);
INSERT INTO Employee VALUES (3,'Lionel','Messi',1023383,3);
INSERT INTO Employee VALUES (4,'Kiya','Titan',102213,2);
INSERT INTO Employee VALUES (5,'Rand','Kreed',10383,3);

INSERT INTO Tournaments VALUES ('WorldCup',2,1023833,10);
INSERT INTO Tournaments VALUES ('League',2,2021833,15);

INSERT INTO Tournaments_Played VALUES (1,'League');
INSERT INTO Tournaments_Played VALUES (3,'League');
INSERT INTO Tournaments_Played VALUES (2,'WorldCup');
INSERT INTO Tournaments_Played VALUES (3,'WorldCup');

INSERT INTO Players VALUES (1,5,2,0,'CF');
INSERT INTO Players VALUES (3,10,9,4,'LW');
INSERT INTO Managers VALUES (2);
INSERT INTO Managers VALUES (5);
INSERT INTO Coaches VALUES (4);

INSERT INTO Fans VALUES ('ManU', 123, 12);
INSERT INTO Fans VALUES ('Arsenal', 78, 7);
INSERT INTO Fans VALUES ('Barca', 431, 123);
```

Join Queries

Showcase at least 4 join queries
Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

```
mysql> SELECT employee.salary,employee.First_Name,clubs.Club_Name
    -> FROM employee
    -> INNER JOIN clubs
    -> ON employee.Club_ID = clubs.Club_ID;
+---------+------------+-----------+
| salary  | First_Name | Club_Name |
+---------+------------+-----------+
|  102383 | Romelu     | ManU      |
|  102343 | Jorjo      | Arsenal   |
|  102213 | Kiya       | Arsenal   |
| 1023383 | Lionel     | Barca     |
|   10383 | Rand       | Barca     |
|    1000 | Rajesh     | Barca     |
+---------+------------+-----------+
6 rows in set (0.00 sec)
```

```
mysql> SELECT employee.Employee_ID,employee.First_Name,clubs.Club_Name
    -> FROM employee
    -> INNER JOIN clubs
    -> ON employee.Club_ID = clubs.Club_ID;
+-------------+------------+-----------+
| Employee_ID | First_Name | Club_Name |
+-------------+------------+-----------+
|           1 | Romelu     | ManU      |
|           2 | Jorjo      | Arsenal   |
|           3 | Lionel     | Barca     |
|           4 | Kiya       | Arsenal   |
|           5 | Rand       | Barca     |
+-------------+------------+-----------+
5 rows in set (0.00 sec)
```

```
mysql> SELECT employee.First_Name,employee.Last_Name,clubs.Club_Name
    -> FROM employee
    -> INNER JOIN clubs
    -> ON employee.Club_ID = clubs.Club_ID;
+------------+-----------+-----------+
| First_Name | Last_Name | Club_Name |
+------------+-----------+-----------+
| Romelu     | Lukaku    | ManU      |
| Jorjo      | Mason     | Arsenal   |
| Kiya       | Titan     | Arsenal   |
| Lionel     | Messi     | Barca     |
| Rand       | Kreed     | Barca     |
| Rajesh     | Kumar     | Barca     |
+------------+-----------+-----------+
```

```
mysql> SELECT employee.First_Name,employee.Last_Name,players.Goals
    -> FROM employee
    -> INNER JOIN players
    -> ON employee.Employee_ID = players.Employee_ID;
+------------+-----------+-------+
| First_Name | Last_Name | Goals |
+------------+-----------+-------+
| Romelu     | Lukaku    |     0 |
| Lionel     | Messi     |    10 |
+------------+-----------+-------+
2 rows in set (0.00 sec)
```

Aggregate Functions

Showcase at least 4 Aggregate function queries
Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

```
mysql> SELECT SUM(Goals) as Totalgoals
    -> FROM players;
+------------+
| Totalgoals |
+------------+
|         15 |
+------------+
```

```
mysql> SELECT min(Goals) as Mingoals
    -> FROM players;
+----------+
| Mingoals |
+----------+
|        5 |
+----------+
```

```
mysql> select max(Goals) as maxgoals from players
+----------+
| maxgoals |
+----------+
|       10 |
+----------+
1 row in set (0.00 sec)
```

```
mysql> select avg(Salary) from employee
+-------------+
| avg(Salary) |
+-------------+
| 223617.5000 |
+-------------+
```

Set Operations

Showcase at least 4 Set Operations queries

Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

```
mysql> select First_Name from employee
    -> UNION
    -> select Assists from players;
+-------------+
| First_Name  |
+-------------+
| Romelu      |
| Jorjo       |
| Lionel      |
| Kiya        |
| Rand        |
| Rajesh      |
| 2           |
| 9           |
+-------------+
8 rows in set (0.00 sec)
```

```
mysql> SELECT Stadium_Name FROM stadiums
    -> UNION
    -> SELECT Club_Name FROM clubs;
+---------------+
| Stadium_Name  |
+---------------+
| Trafford      |
| Emirates      |
| CampNou       |
| ManU          |
| Arsenal       |
| Barca         |
+---------------+
```

```
mysql> select Employee_ID from coaches
    -> UNION
    -> select First_Name from employee
    -> ;
+-------------+
| Employee_ID |
+-------------+
| 4           |
| Romelu      |
| Jorjo       |
| Lionel      |
| Kiya        |
| Rand        |
| Rajesh      |
+-------------+
```

\

```
mysql> select Employee_ID from coaches
    -> UNION ALL
    -> select Employee_ID from employee;
+-------------+
| Employee_ID |
+-------------+
|           4 |
|           1 |
|           2 |
|           4 |
|           3 |
|           5 |
|          10 |
+-------------+
```

Functions and Procedures

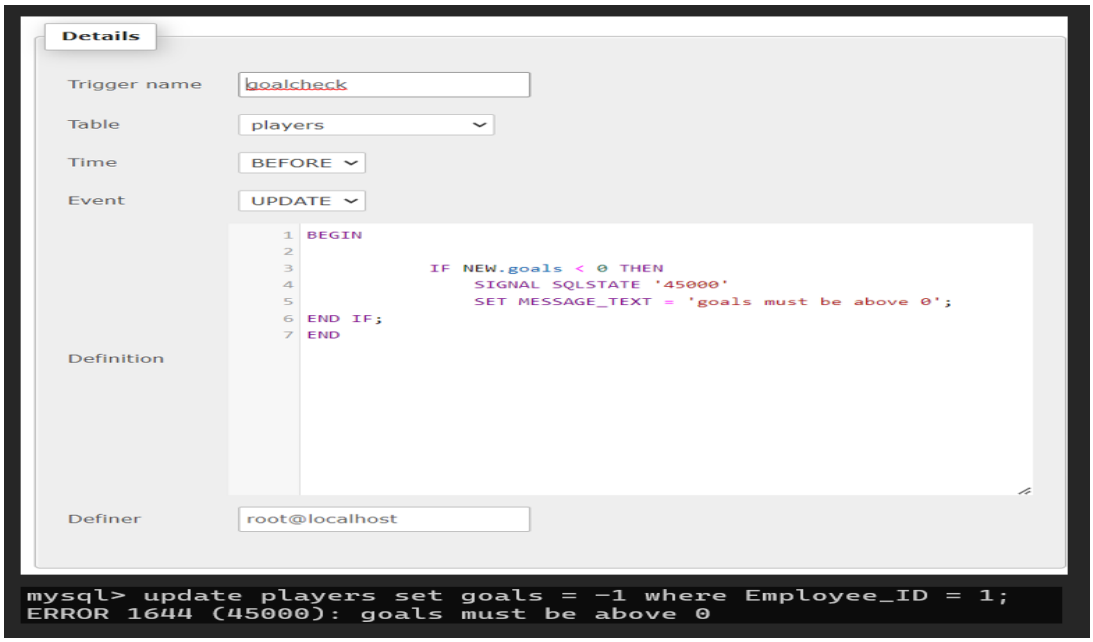
Create a Function and Procedure. State the objective of the function / Procedure. Run and display the results.

```
mysql> select totalassists();
+-----------------+
| totalassists() |
+-----------------+
|              11 |
+-----------------+
1 row in set (0.00 sec)
```

```
mysql> CALL GetAllClubs();
+---------+-----------+------------+-----------+------------+
| Club_ID | Club_Name | Stadium_ID | Games_Won | Games_Lost |
+---------+-----------+------------+-----------+------------+
|       1 | ManU      |          1 |         2 |          3 |
|       2 | Arsenal   |          2 |         0 |          0 |
|       3 | Barca     |          3 |         1 |          2 |
+---------+-----------+------------+-----------+------------+
3 rows in set (0.00 sec)
```

Triggers and Cursors

Create a Trigger and a Cursor. State the objective. Run and display the results.



CURSOR

```
mysql> CREATE FUNCTION goalplayer ( eid INT )
    -> RETURNS INT
    -> DETERMINISTIC
    ->
    -> BEGIN
    ->
    ->    DECLARE done INT DEFAULT FALSE;
    ->    DECLARE goal INT DEFAULT 0;
    ->
    ->    DECLARE c1 CURSOR FOR
    ->      SELECT goals
    ->      FROM players
    ->    WHERE Employee_ID = eid;
    ->
    ->    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
    ->
    ->    OPEN c1;
    ->    FETCH c1 INTO goal;
    ->
    ->    CLOSE c1;
    ->
    ->    RETURN goal;
    ->
    -> END; //
```

```
mysql> select goalplayer(3);
    -> //
+---------------+
| goalplayer(3) |
+---------------+
|            10 |
+---------------+
```

Developing a Frontend

The frontend should support

1. Addition, Modification and Deletion of records from any chosen table
2. There should be an window to accept and run any SQL statement and display the result