DATABASE MANAGEMENT SYSTEM

Cricket score card

MAHESH KM – PES1UG21CS315

LALATAAKSH S – PES1UG21CS299

SEC: E

INTRODUCTION

Cricket score card – A database system to keep track of cricket tournament statistics

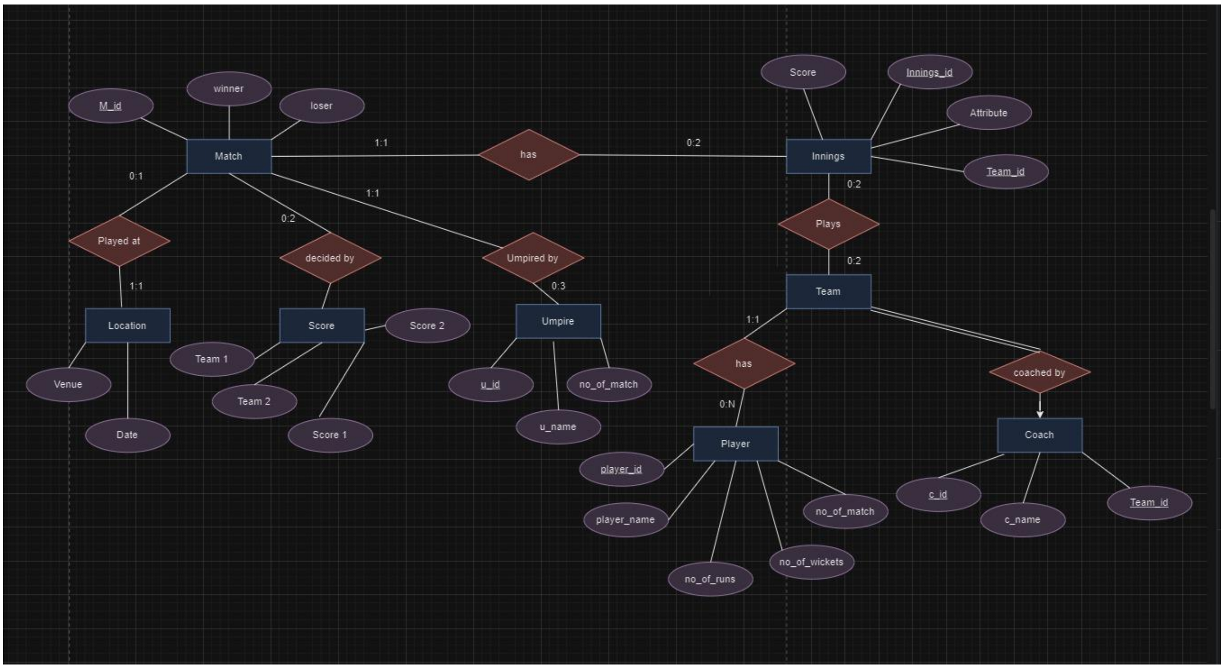
OBJECTIVES

- Maintain record of team score
- Maintain record of individual player performance
- Maintain record of matches in tournaments(win/loss)
- Maintain ranking of teams in tournament
- Creation, Maintenance, modifying of details about the teams and players

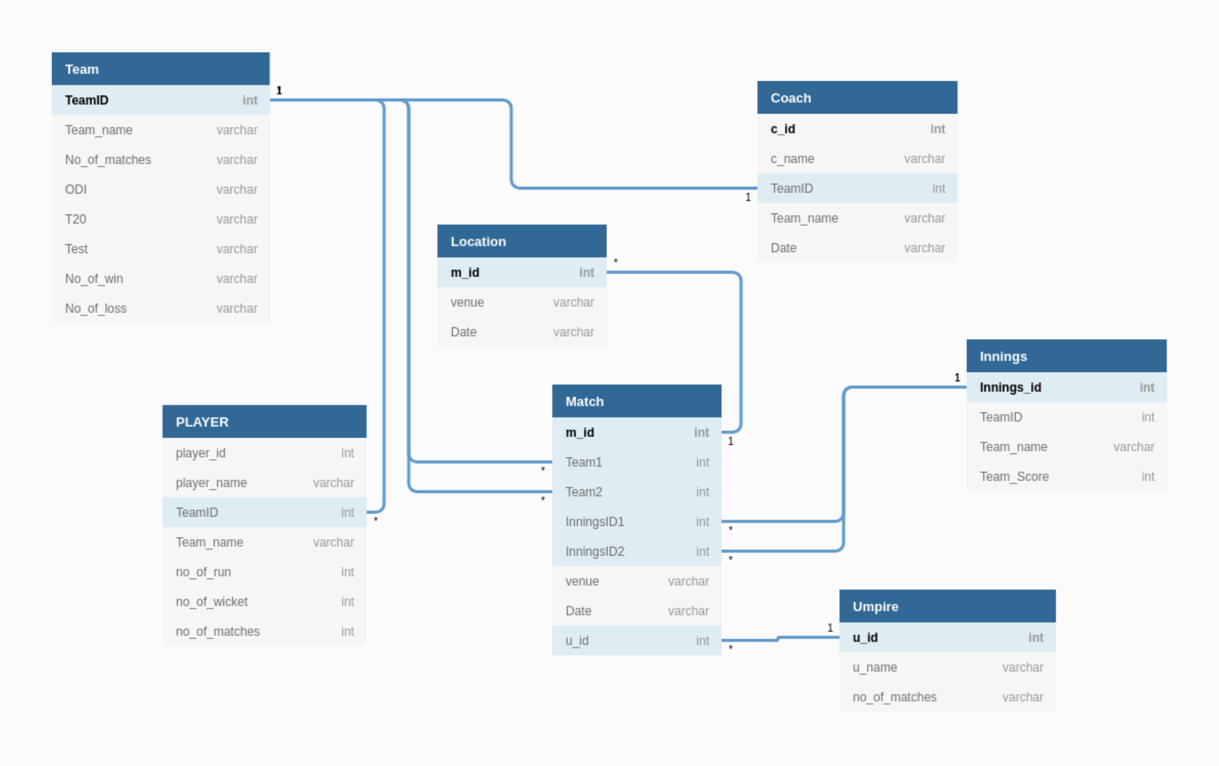
FUNCTIONAL REQUIREMENTS

- The system is able to keep track of all matches records.
- The performances of players in the team are updated in accordance to matches.
- The system is able to find out winner team and loser team and also records individual performance of players.

ER DIAGRAM:



RELATIONAL SCHEMA



Team	
TeamID	int
Team_name	varchar
No_of_matches	varchar
ODI	varchar
T20	varchar
Test	varchar
No_of_win	varchar
No_of_loss	varchar

Coach	
c_id	int
c_name	varchar
TeamID	int
Team_name	varchar
Date	varchar

Location	
m_id	int
venue	varchar
Date	varchar

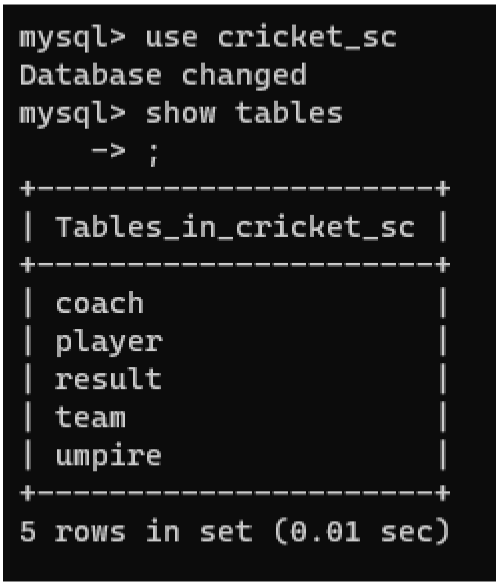
Innings	
Innings_id	int
TeamID	int
Team_name	varchar
Team_Score	int

PLAYER	
player_id	int
player_name	varchar
TeamID	int
Team_name	varchar
no_of_run	int
no_of_wicket	int
no_of_matches	int

Match	
m_id	int
Team1	int
Team2	int
InningsID1	int
InningsID2	int
venue	varchar
Date	varchar
u_id	int

Umpire	
u_id	int
u_name	varchar
no_of_matches	varchar

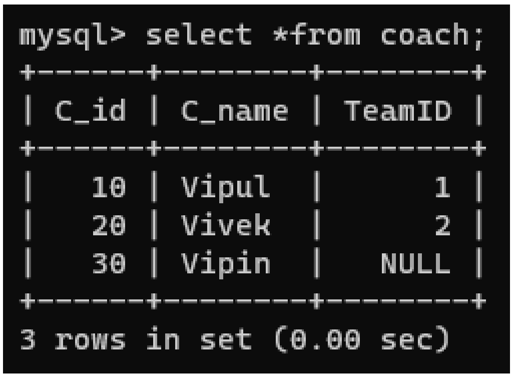
Tables created:



```
mysql> use cricket_sc
Database changed
mysql> show tables
    -> ;
+----------------------+
| Tables_in_cricket_sc |
+----------------------+
| coach                |
| player               |
| result               |
| team                 |
| umpire               |
+----------------------+
5 rows in set (0.01 sec)
```

5 tables are created

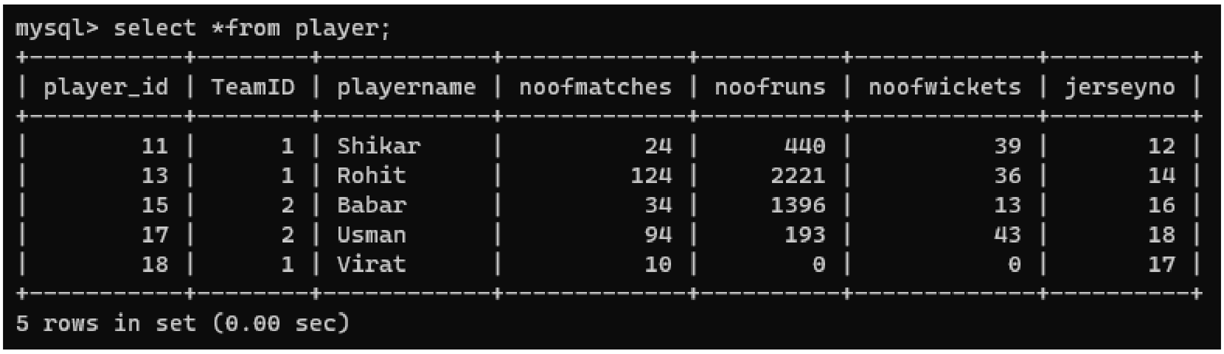
Coach:



```
mysql> select *from coach;
+------+--------+--------+
| C_id | C_name | TeamID |
+------+--------+--------+
|   10 | Vipul  |      1 |
|   20 | Vivek  |      2 |
|   30 | Vipin  |   NULL |
+------+--------+--------+
3 rows in set (0.00 sec)
```

Each team will have a different coach.

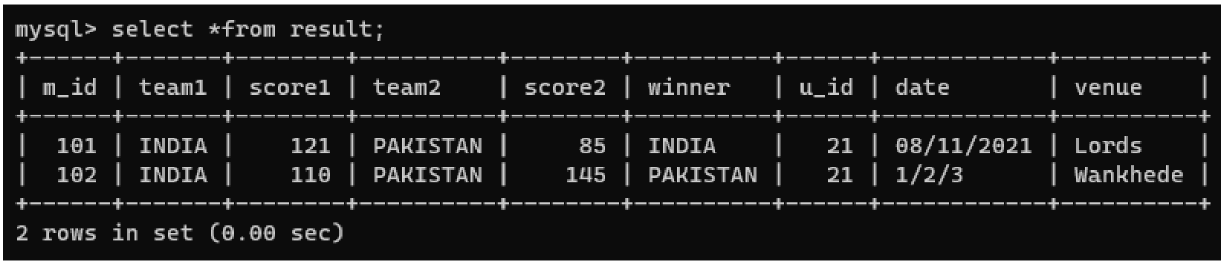
Player:



```
mysql> select *from player;
+-----------+--------+------------+-------------+----------+-------------+----------+
| player_id | TeamID | playername | noofmatches | noofruns | noofwickets | jerseyno |
+-----------+--------+------------+-------------+----------+-------------+----------+
|        11 |      1 | Shikar     |          24 |      440 |          39 |       12 |
|        13 |      1 | Rohit      |         124 |     2221 |          36 |       14 |
|        15 |      2 | Babar      |          34 |     1396 |          13 |       16 |
|        17 |      2 | Usman      |          94 |      193 |          43 |       18 |
|        18 |      1 | Virat      |          10 |        0 |           0 |       17 |
+-----------+--------+------------+-------------+----------+-------------+----------+
5 rows in set (0.00 sec)
```

No.of matches, no.of runs, no.of wickets of a player increases everytime a new match is played.

Result:



```
mysql> select *from result;
+------+-------+--------+----------+--------+----------+------+------------+----------+
| m_id | team1 | score1 | team2    | score2 | winner   | u_id | date       | venue    |
+------+-------+--------+----------+--------+----------+------+------------+----------+
|  101 | INDIA |    121 | PAKISTAN |     85 | INDIA    |   21 | 08/11/2021 | Lords    |
|  102 | INDIA |    110 | PAKISTAN |    145 | PAKISTAN |   21 | 1/2/3      | Wankhede |
+------+-------+--------+----------+--------+----------+------+------------+----------+
2 rows in set (0.00 sec)
```

Result of a match is calculated by:

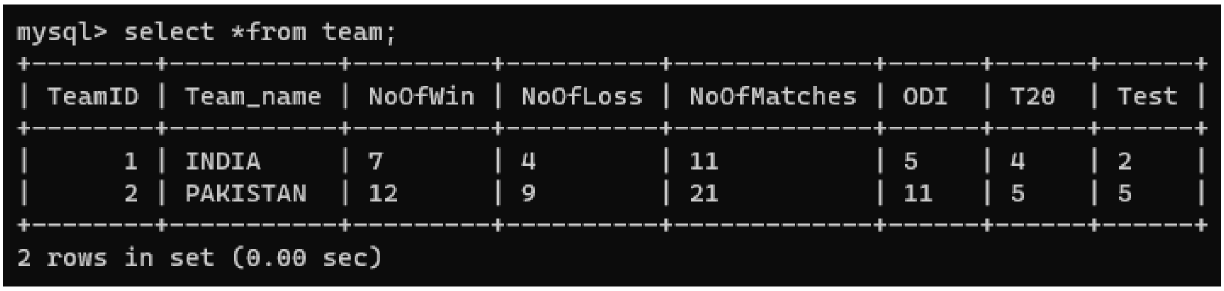
Team1 score = player1 score + player2 score

Team2 score = player3 score + player4 score

If team1 score > team2 score

Team1 is winner.

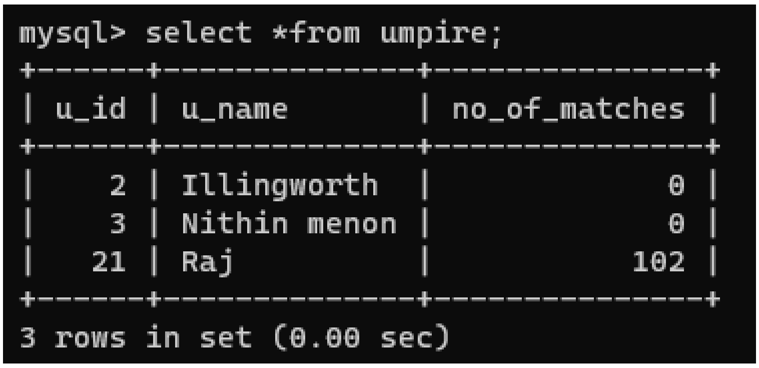
Team:



```
mysql> select *from team;
+--------+-----------+---------+----------+-------------+------+------+------+
| TeamID | Team_name | NoOfWin | NoOfLoss | NoOfMatches | ODI  | T20  | Test |
+--------+-----------+---------+----------+-------------+------+------+------+
|      1 | INDIA     | 7       | 4        | 11          | 5    | 4    | 2    |
|      2 | PAKISTAN  | 12      | 9        | 21          | 11   | 5    | 5    |
+--------+-----------+---------+----------+-------------+------+------+------+
2 rows in set (0.00 sec)
```

Two teams play a match

Umpire:



```
mysql> select *from umpire;
+------+--------------+---------------+
| u_id | u_name       | no_of_matches |
+------+--------------+---------------+
|    2 | Illingworth  |             0 |
|    3 | Nithin menon |             0 |
|   21 | Raj          |           102 |
+------+--------------+---------------+
3 rows in set (0.00 sec)
```

Single umpire is selected for every match.

Set of Functional dependencies that must hold on each table:

- **Team(Team_name) → Team(TeamID, NoOfWin, NoOfLoss, ODI, T20, Test)**

- **PLAYER(TeamID, jersey_no) → player(Player_name, Batting, Bowling, No_of_matches)**

- **RESULT(team1, team2, date, venue) → RESULT(m_id, score1, score2, winner, u_id)**

- **Team(TeamID) → Team(Team_name, NoOfWin, NoOfLoss, ODI, T20, Test)**

- **coach(c_id) → coach(c_id, c_name, TeamID)**

- **player(PlayerID, TeamID) → player(Player_name, Batting, Bowling, No_of_matches, jersey_no)**

- **RESULT(m_id) → RESULT(team1, score1, team2, score2, winner, u_id)**

- **UMPIRE(u_id) → UMPIRE(u_name, no_of_matches)**

Create triggers for each table:

1) Trigger for Player Insert:

- This trigger updates the ‘NoOfPlayers’ column in the ‘Team’ table whenever a new player is inserted.

Code:

```
self.playerCursor.execute('''
    CREATE TRIGGER IF NOT EXISTS update_team_players
    AFTER INSERT ON PLAYER
    BEGIN
        UPDATE Team
        SET NoOfPlayers = NoOfPlayers + 1
        WHERE TeamID = NEW.TeamID;
    END;
''')
```

2) Trigger for Team Delete:

- This trigger ensures that when a team is deleted, all related players are also deleted.

Code:

```
self.deleteTeamCursor.execute('''
    CREATE TRIGGER IF NOT EXISTS delete_related_players
    BEFORE DELETE ON Team
    BEGIN
        DELETE FROM PLAYER WHERE TeamID = OLD.TeamID;
    END;
''')
```

3) Trigger for Coach Update:

- This trigger ensures that when a coach is updated with a new team, the old team's coach count is reduced.

Code:

```
self.coachCursor.execute('''
  CREATE TRIGGER IF NOT EXISTS update_old_team_coach_count
  AFTER UPDATE OF TeamID ON Coach
  BEGIN
    UPDATE Team
    SET NoOfCoaches = NoOfCoaches - 1
    WHERE TeamID = OLD.TeamID;
  END;
''')
```

4) Trigger for Partnership Insert:

- This trigger updates the TotalPartnerships column in the Team table whenever a new partnership is inserted.

Code:

```
self.foreign.execute('''
  CREATE TRIGGER IF NOT EXISTS update_team_partnerships
  AFTER INSERT ON partnership
  BEGIN
    UPDATE Team
    SET TotalPartnerships = TotalPartnerships + 1
    WHERE TeamID = NEW.TeamID;
  END;
''')
```