LAB ASSIGNMENT REPORT

Name: Prothito Shovon Majumder

Course Code: CSE-2201

Course Name: Database Management Systems – I

Submitted to: Abu Ahmed Ferdaus

Associate Professor

Department of Computer Science and Engineering

University of Dhaka

Description:

The database I have implemented is for storing and processing queries on information of cricket players of national teams. A player plays for a team and a coach coaches a team. Teams participate in matches, which are held in stadiums situated in different cities of different countries. The players have batting and bowling records in the matches they participate in.

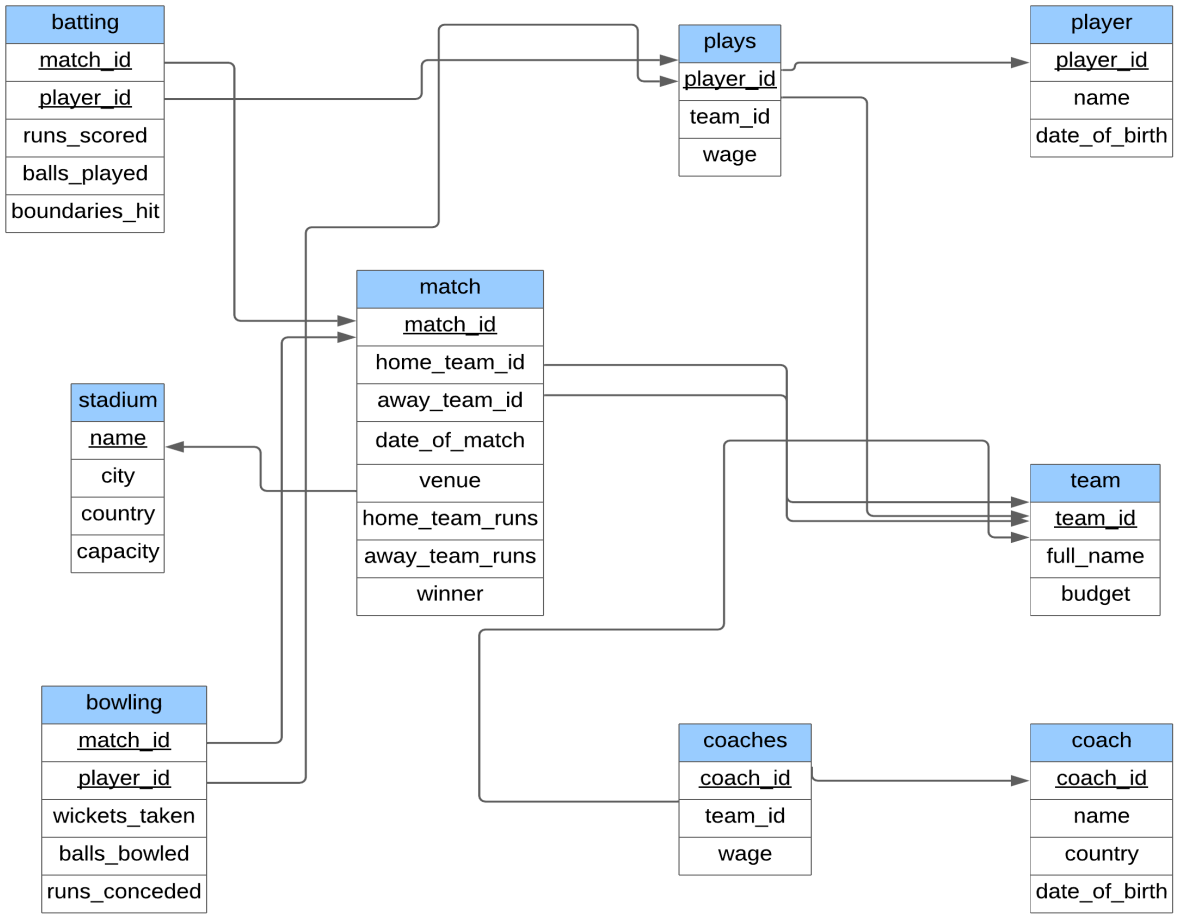
The data contained in the database is that of ICC Champions Trophy 2017 and was collected from espncricinfo and cricbuzz. I have implemented the database in PostgreSQL.

Schemas and Attributes:

The schemas I needed are listed below with respective attributes:

- player(id, name, date_of_birth)
- coach(id, name, country, date_of_birth)
- team(id, full_name, budget)
- plays(player_id, team_id, wage)
- coaches(coach_id, team_id, wage)
- matches(match_id, home_team_id, away_team_id, date_of_match, venue, home_team_runs, away_team_runs, winner)
- stadium(name, city, country, capacity)
- batting(match_id, player_id, runs_scored, balls_played, boundaries_hit)
- bowling(match_id, player_id, wickets_taken, balls_bowled, runs_conceded)

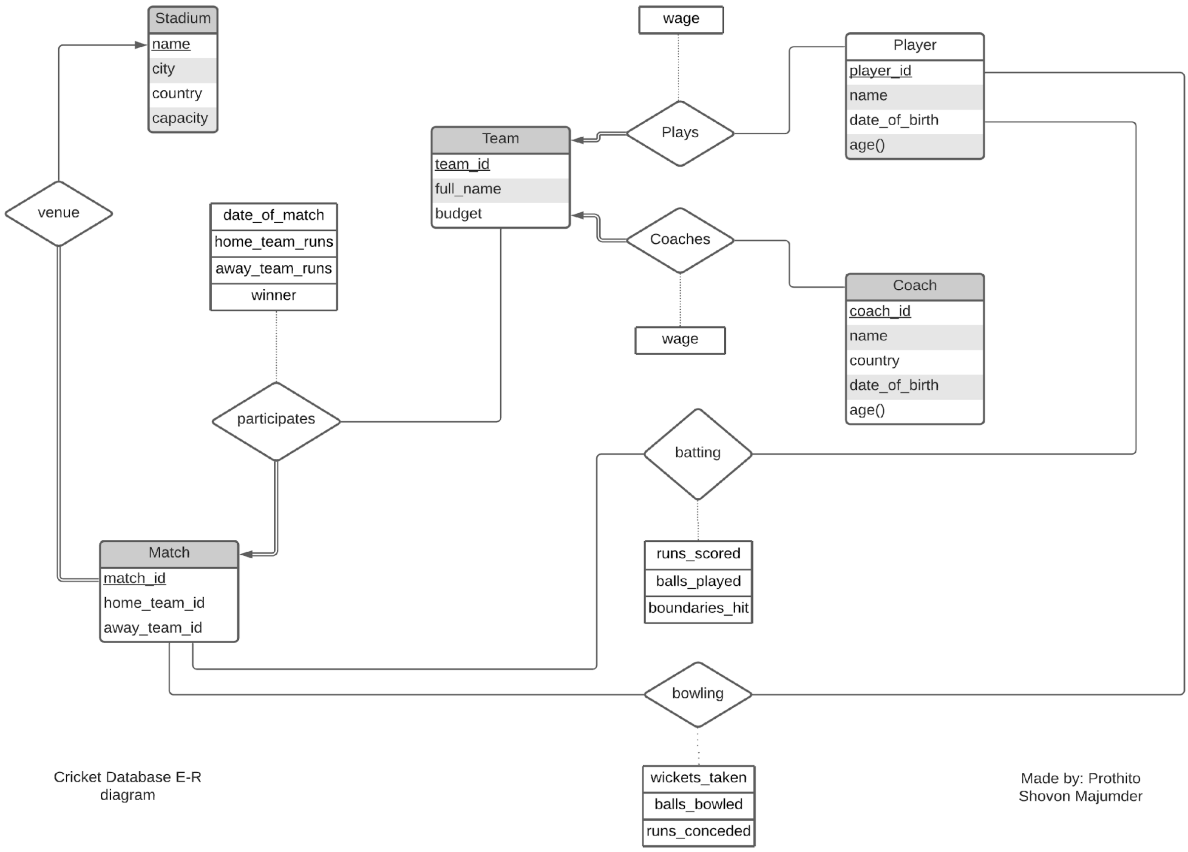
Schema Diagram:



Cricket Database Schema

Made by: Prothito Shovon Majumder

E-R Diagram:



Snapshots of DDL:

```
-- create player table
CREATE TABLE player(
    player_id SERIAL,
    name varchar(50) NOT NULL,
    date_of_birth DATE NOT NULL,
        CONSTRAINT player_pKey PRIMARY KEY(id)
);

-- create coach table
CREATE TABLE coach(
    coach_id SERIAL,
    name varchar(50) NOT NULL,
    country varchar(50) NOT NULL,
    date_of_birth DATE NOT NULL,
        CONSTRAINT coach_pKey PRIMARY KEY(id)
);

-- create stadium table
CREATE TABLE stadium(
    name varchar(50),
    city varchar(50) NOT NULL,
    country varchar(50) NOT NULL,
    capacity INT
        CONSTRAINT stadium_capacity_constraint CHECK (capacity > 0),
        CONSTRAINT stadium_pKey PRIMARY KEY(name)
);

-- create team table
CREATE TABLE team(
    team_id SERIAL,
    full_name varchar(70) NOT NULL UNIQUE,
    budget NUMERIC(15, 2)
        CONSTRAINT team_budget_constraint CHECK (budget >= 0),
        CONSTRAINT team_pKey PRIMARY KEY (id)
);
```

```sql
-- create plays table
CREATE TABLE plays(
    player_id SERIAL NOT NULL,
    team_id SERIAL NOT NULL,
    wage NUMERIC(10, 2)
        CONSTRAINT plays_wage_constraint CHECK (wage >= 0),
        CONSTRAINT plays_pKey PRIMARY KEY(player_id),
        CONSTRAINT plays_playerID_fKey FOREIGN KEY(player_id) REFERENCES player(id)
            ON DELETE CASCADE
            ON UPDATE CASCADE,
        CONSTRAINT plays_teamID_fKey FOREIGN KEY(team_id) REFERENCES team(id)
            ON DELETE SET NULL
            ON UPDATE CASCADE
);

-- create coaches table
CREATE TABLE coaches(
    coach_id SERIAL NOT NULL,
    team_id SERIAL NOT NULL,
    wage NUMERIC(10, 2)
        CONSTRAINT coaches_wage_constraint CHECK (wage >= 0),
        CONSTRAINT coaches_pKey PRIMARY KEY (coach_id),
        CONSTRAINT coaches_coachID_fKey FOREIGN KEY(coach_id) REFERENCES coach(id)
            ON DELETE CASCADE
            ON UPDATE CASCADE,
        CONSTRAINT coaches_teamID_fKey FOREIGN KEY(team_id) REFERENCES team(id)
            ON DELETE SET NULL
            ON UPDATE CASCADE
);
```

```sql
-- create matches table
CREATE TABLE matches(
    match_id SERIAL,
    home_team_id SERIAL NOT NULL,
    away_team_id SERIAL NOT NULL,
    date_of_match DATE,
    venue varchar(50),
    home_team_runs INT
        CONSTRAINT matches_home_team_goal_constraint CHECK (home_team_runs >= 0),
    away_team_runs INT
        CONSTRAINT matches_away_team_goal_constraint CHECK (away_team_runs >= 0),
    winner INT NOT NULL
        CONSTRAINT matches_winner_constraint CHECK ((winner>=0) and (winner<=2)), -- 1 if home team wins, 2 if away team wins, 0 if draw
        CONSTRAINT matches_pKey PRIMARY KEY(match_id),
        CONSTRAINT matches_homeID_fKey FOREIGN KEY(home_team_id) REFERENCES team(id)
            ON DELETE No Action
            ON UPDATE CASCADE,
        CONSTRAINT matches_awayID_fKey FOREIGN KEY(away_team_id) REFERENCES team(id)
            ON DELETE No Action
            ON UPDATE CASCADE,
        CONSTRAINT matches_venue_fKey FOREIGN KEY(venue) REFERENCES stadium(name)
            ON DELETE No Action
            ON UPDATE CASCADE
);
```

```sql
-- create batting table
CREATE TABLE batting(
    match_id SERIAL,
    player_id SERIAL,
    runs_scored INT,
    balls_played INT,
    boundaries_hit INT
        CONSTRAINT batting_runs_scored_constraint check (runs_scored >= 0),
        CONSTRAINT batting_balls_played_constraint check (balls_played >= 0),
    CONSTRAINT batting_pKey PRIMARY KEY (match_id, player_id),
    CONSTRAINT batting_matchID_fKey FOREIGN KEY(match_id) REFERENCES matches(match_id)
        ON DELETE No Action
        ON UPDATE CASCADE,
    CONSTRAINT batting_playerID_fKey FOREIGN KEY(player_id) REFERENCES player(id)
        ON DELETE No Action
        ON UPDATE CASCADE
);
--create bowling table
CREATE TABLE bowling(
    match_id SERIAL,
    player_id SERIAL,
    wickets_taken INT,
    balls_bowled INT,
    runs_conceded INT
        CONSTRAINT bowling_wickets_taken_constraint check (wickets_taken >= 0 and wickets_taken <= balls_bowled),
        CONSTRAINT bowling_balls_bowled_constraint check (balls_bowled >= 0),
        CONSTRAINT bowling_runs_conceded_constraint check (runs_conceded >= 0),
    CONSTRAINT bowling_pKey PRIMARY KEY (match_id, player_id),
    CONSTRAINT bowling_matchID_fKey FOREIGN KEY(match_id) REFERENCES matches(match_id)
        ON DELETE No Action
        ON UPDATE CASCADE,
    CONSTRAINT bowling_playerID_fKey FOREIGN KEY(player_id) REFERENCES player(id)
        ON DELETE No Action
        ON UPDATE CASCADE
);
```

```sql
--insert into stadium
INSERT INTO stadium VALUES
('Kennington Oval', 'London', 'England', 25500);
INSERT INTO stadium VALUES
('Old Trafford', 'Manchester', 'England', 19000);
INSERT INTO stadium VALUES
('Trent Bridge', 'Nottingham', 'England', 17000);
INSERT INTO stadium VALUES
('Edgbaston Cricket Ground', 'Birmingham', 'England', 24803);
INSERT INTO stadium VALUES
('Lord''s', 'London', 'England', 30000);
INSERT INTO stadium VALUES
('Sophia Gardens', 'Cardiff', 'England',15643);
INSERT INTO stadium VALUES
('Rose Bowl', 'Southampton', 'England', 25000);
INSERT INTO stadium VALUES
('WACA Ground', 'Perth', 'Australia', 24500);
INSERT INTO stadium VALUES
('Melbourne Cricket Ground', 'Melbourne', 'Australia', 100024);
INSERT INTO stadium VALUES
('Sydney Cricket Ground', 'Sydney', 'Australia', 48000);
INSERT INTO stadium VALUES
('Adelaide Oval', 'Adelaide', 'Australia', 53583);
INSERT INTO stadium VALUES
('The Gabba', 'Brisbane', 'Australia', 36000);
INSERT INTO stadium VALUES
('Sher-e-Bangla Cricket Stadium','Dhaka','Bangladesh',25416);
INSERT INTO stadium VALUES
('Zohur Ahmed Chowdhury Stadium','Chittagong','Bangladesh',22000);
INSERT INTO stadium VALUES
('Sylhet International Cricket Stadium','Sylhet','Bangladesh',18500);
INSERT INTO stadium VALUES
('Sheikh Abu Naser Stadium','Khulna','Bangladesh',15000);
```

```
INSERT INTO player VALUES
('201','Tamim Iqbal', '03-20-1989');
INSERT INTO player VALUES
('202','Soumya Sarkar','02-25-1993');
INSERT INTO player VALUES
('203','Imrul Kayes','02-02-1987');
INSERT INTO player VALUES
('204','Sabbir Rahman','11-22-1991');
INSERT INTO player VALUES
('205','Shakib Al Hasan','03-24-1987');
INSERT INTO player VALUES
('206','Mahmudullah','02-04-1986');
INSERT INTO player VALUES
('207','Mosaddek Hossain','12-10-1995');
INSERT INTO player VALUES
('208','Mehidy Hasan','10-25-1997');
INSERT INTO player VALUES
('209','Mushfiqur Rahim','06-09-1987');
INSERT INTO player VALUES
('210','Mashrafe Mortaza','10-05-1983');
INSERT INTO player VALUES
('211','Mustafizur Rahman','09-06-1995');
INSERT INTO player VALUES
('212','Rubel Hossain','01-01-1990');
INSERT INTO player VALUES
('213','Taskin Ahmed','03-04-1995');
INSERT INTO player VALUES
('214','Shafiul Islam','10-06-1989');
INSERT INTO player VALUES
('215','Sunzamul Islam','01-17-1990');
INSERT INTO player VALUES
('301','Steven Smith','06-02-1989');
INSERT INTO player VALUES
('302','David Warner','10-27-1986');
```

```
--insert into coach
INSERT INTO coach VALUES('21','Trevor Bayliss','Australia','12-21-1962');
INSERT INTO coach VALUES('22','Chandika Hathurusingha','Sri Lanka','09-13-1968');
INSERT INTO coach VALUES('23','Darren Lehmann','Australia','02-05-1970');
INSERT INTO coach VALUES('24', 'Mike Hesson', 'New Zealand', '10-30-1974');
INSERT INTO coach VALUES('25', 'Anil Kumble', 'India', '10-17-1970');
INSERT INTO coach VALUES('26', 'Mickey Arthur', 'South Africa', '05-17-1968');
INSERT INTO coach VALUES('27', 'Russell Domingo', 'South Africa', '08-30-1974');
INSERT INTO coach VALUES('28', 'Graham Ford', 'South Africa', '11-16-1960');
INSERT INTO coach VALUES('29', 'Stuart Law', 'Australia', '10-18-1968');
INSERT INTO coach VALUES('30', 'Lalchand Rajput', 'India', '12-18-1961');
```

```
--insert into team
INSERT INTO team VALUES
('1','England',59000000);
INSERT INTO team VALUES
('2','Bangladesh',51000000);
INSERT INTO team VALUES
('3','Australia',24000000);
INSERT INTO team VALUES
('4','New Zealand',9000000);
INSERT INTO team VALUES
('5','India',295000000);
INSERT INTO team VALUES
('6','Pakistan',55000000);
INSERT INTO team VALUES
('7','South Africa',79000000);
INSERT INTO team VALUES
('8','Sri Lanka',20000000);
INSERT INTO team VALUES
('9','West Indies',15000000);
```

```sql
INSERT INTO plays VALUES('101','1',120000);
INSERT INTO plays VALUES('102','1',75000);
INSERT INTO plays VALUES('103','1',150000);
INSERT INTO plays VALUES('104','1',80000);
INSERT INTO plays VALUES('105','1',100000);
INSERT INTO plays VALUES('106','1',150000);
INSERT INTO plays VALUES('107','1',50000);
INSERT INTO plays VALUES('108','1',75500);
INSERT INTO plays VALUES('109','1',105000);
INSERT INTO plays VALUES('110','1',64500);
INSERT INTO plays VALUES('111','1',130000);
INSERT INTO plays VALUES('112','1',55000);
INSERT INTO plays VALUES('113','1',85000);
INSERT INTO plays VALUES('114','1',97500);
INSERT INTO plays VALUES('115','1',100000);
INSERT INTO plays VALUES('201','2',105000);
INSERT INTO plays VALUES('202','2',80000);
INSERT INTO plays VALUES('203','2',69000);
INSERT INTO plays VALUES('204','2',35000);
INSERT INTO plays VALUES('205','2',140000);
INSERT INTO plays VALUES('206','2',115000);
INSERT INTO plays VALUES('207','2',84500);
INSERT INTO plays VALUES('208','2',98500);
INSERT INTO plays VALUES('209','2',115000);
INSERT INTO plays VALUES('210','2',120000);
INSERT INTO plays VALUES('211','2',92000);
INSERT INTO plays VALUES('212','2',81000);
INSERT INTO plays VALUES('213','2',70000);
INSERT INTO plays VALUES('214','2',52000);
INSERT INTO plays VALUES('215','2',38750);
INSERT INTO plays VALUES('301','3',160000);
```

```sql
INSERT INTO coaches VALUES('21','1',120000);
INSERT INTO coaches VALUES('22','2',100000);
INSERT INTO coaches VALUES('23','3',117500);
INSERT INTO coaches VALUES('24','4',90000);
INSERT INTO coaches VALUES('25','5',150000);
INSERT INTO coaches VALUES('26','6',97500);
INSERT INTO coaches VALUES('27','7',110000);
INSERT INTO coaches VALUES('28','8',84500);
INSERT INTO coaches VALUES('29','9',72000);
INSERT INTO coaches VALUES('30','10',65500);
```

```sql
INSERT INTO matches VALUES
('1','1','2','06-01-2017','Kennington Oval',308,305,1);
INSERT INTO matches VALUES
('2','8','7','06-03-2017','Kennington Oval',203,299,2);
INSERT INTO matches VALUES
('3','5','6','06-04-2017','Edgbaston',319,164,1);
INSERT INTO matches VALUES
('4','1','4','06-06-2017','Sophia Gardens',310,223,1);
INSERT INTO matches VALUES
('5','7','6','06-07-2017','Edgbaston',219,119,2);
INSERT INTO matches VALUES
('6','5','8','06-08-2017','Kennington Oval',321,322,2);
INSERT INTO matches VALUES
('7','4','2','06-09-2017','Kennington Oval',265,268,2);
INSERT INTO matches VALUES
('8','1','3','06-10-2017','Edgbaston',240,277,1);
INSERT INTO matches VALUES
('9','5','7','06-11-2017','Kennington Oval',193,191,1);
INSERT INTO matches VALUES
('10','8','6','06-12-2017','Sophia Gardens',236,237,2);
INSERT INTO matches VALUES
('11','1','6','06-14-2017','Sophia Gardens',211,215,2);
INSERT INTO matches VALUES
('12','2','5','06-15-2017','Edgbaston',264,265,2);
INSERT INTO matches VALUES
('13','6','5','06-17-2017','Kennington Oval',338,158,1);
```

```
INSERT INTO batting VALUES
('1','201',128,142,15);
INSERT INTO batting VALUES
('1','202',28,34,5);
INSERT INTO batting VALUES
('1','203',19,20,3);
INSERT INTO batting VALUES
('1','209',79,72,8);
INSERT INTO batting VALUES
('1','205',10,8,2);
INSERT INTO batting VALUES
('1','204',24,15,3);
INSERT INTO batting VALUES
('1','206',6,6,0);
INSERT INTO batting VALUES
('1','207',2,3,0);
INSERT INTO bowling VALUES
('1','108',0,12,4);
INSERT INTO bowling VALUES
('1','115',0,60,58);
INSERT INTO bowling VALUES
('1','112',1,60,82);
INSERT INTO bowling VALUES
('1','106',0,42,42);
INSERT INTO bowling VALUES
('1','113',4,60,59);
INSERT INTO bowling VALUES
('1','105',0,48,40);
INSERT INTO bowling VALUES
('1','103',0,18,18);
```

Snapshot of instances:

Only portions of data from each table are shown.

Stadium table:

	name [PK] character varying (50)	city character varying (50)	country character varying (50)	capacity integer
1	Kennington Oval	London	England	25500
2	Old Trafford	Manchester	England	19000
3	Trent Bridge	Nottingham	England	17000
4	Lord's	London	England	30000
5	Sophia Gardens	Cardiff	England	15643
6	Rose Bowl	Southampton	England	25000
7	WACA Ground	Perth	Australia	24500
8	Melbourne Cricket Ground	Melbourne	Australia	100024
9	Sydney Cricket Ground	Sydney	Australia	48000
10	Adelaide Oval	Adelaide	Australia	53583
11	The Gabba	Brisbane	Australia	36000

Player table:

id [PK] integer	name character varying (50)	date_of_birth date
105	Moeen Ali	1987-06-18
106	Ben Stokes	1991-06-04
107	David Willey	1990-02-28
108	Chris Woakes	1989-03-02
109	Jonny Bairstow	1989-09-26
110	Sam Billings	1991-06-15
111	Jos Buttler	1990-09-08
112	Jake Ball	1991-03-14
113	Liam Plunkett	1985-04-06
114	Adil Rashid	1988-02-17
115	Mark Wood	1990-01-11
201	Tamim Iqbal	1989-03-20
202	Soumya Sarkar	1993-02-25
203	Imrul Kayes	1987-02-02
204	Sabbir Rahman	1991-11-22
205	Shakib Al Hasan	1987-03-24
206	Mahmudullah	1986-02-04
207	Mosaddek Hossain	1995-12-10

Coach table:

id [PK] integer	name character varying (50)	country character varying (50)	date_of_birth date
21	Trevor Bayliss	Australia	1962-12-21
22	Chandika Hathurusingha	Sri Lanka	1968-09-13
23	Darren Lehmann	Australia	1970-02-05
24	Mike Hesson	New Zealand	1974-10-30
25	Anil Kumble	India	1970-10-17
26	Mickey Arthur	South Africa	1968-05-17
27	Russell Domingo	South Africa	1974-08-30
28	Graham Ford	South Africa	1960-11-16
29	Stuart Law	Australia	1968-10-18
30	Lalchand Rajput	India	1961-12-18

Plays table:

player_id [PK] integer	team_id integer	wage numeric (10,2)
101	1	120000.00
102	1	75000.00
103	1	150000.00
104	1	80000.00
105	1	100000.00
106	1	150000.00
107	1	50000.00
108	1	75500.00
109	1	105000.00
110	1	64500.00

Coaches table:

coach_id [PK] integer	team_id integer	wage numeric (10,2)
21	1	120000.00
22	2	100000.00
23	3	117500.00
24	4	90000.00
25	5	150000.00
26	6	97500.00
27	7	110000.00
28	8	84500.00
29	9	72000.00
30	10	65500.00

Team table:

id [PK] integer	full_name character varying (70)	budget numeric (15,2)
1	England	59000000.00
2	Bangladesh	51000000.00
3	Australia	24000000.00
4	New Zealand	9000000.00
5	India	295000000.00
6	Pakistan	55000000.00
7	South Africa	79000000.00
8	Sri Lanka	20000000.00
9	West Indies	15000000.00
10	Afghanistan	4220000.00

Matches table:

match_id [PK] integer	home_team_id integer	away_team_id integer	date_of_match date	venue character varying (50)	home_team_runs integer	away_team_runs integer	winner integer
1	1	2	2017-06-01	Kennington Oval	308	305	1
2	8	7	2017-06-03	Kennington Oval	203	299	2
3	5	6	2017-06-04	Edgbaston	319	164	1
4	1	4	2017-06-06	Sophia Gardens	310	223	1
5	7	6	2017-06-07	Edgbaston	219	119	2
6	5	8	2017-06-08	Kennington Oval	321	322	2
7	4	2	2017-06-09	Kennington Oval	265	268	2
8	1	3	2017-06-10	Edgbaston	240	277	1
9	5	7	2017-06-11	Kennington Oval	193	191	1

Batting table:

match_id [PK] integer	player_id [PK] integer	runs_scored integer	balls_played integer	boundaries_hit integer
1	104	1	8	0
1	102	95	86	13
1	103	133	129	12
1	101	75	61	10
2	701	103	115	7
2	710	23	42	2
2	702	75	70	6
2	711	4	4	0
2	703	18	22	2
2	704	38	20	6

Bowling table:

match_id [PK] integer	player_id [PK] integer	wickets_taken integer	balls_bowled integer	runs_conceded integer
2	814	2	60	54
2	805	0	60	64
2	815	1	60	72
2	712	1	48	46
2	707	0	60	54
2	715	1	36	31
2	706	2	42	32
2	713	4	51	27
2	704	0	12	7
3	613	0	49	32

Queries:

Query #1:

Query Statement:

Show name, city, country and capacity of stadiums situated in Australia with capacity more than 10000.

Relational Algebra Expression:

$$\Pi_{name,\ city,\ country,\ capacity}(\sigma_{country = \text{“Australia”} \wedge capacity > 10{,}000}(\text{stadium}))$$

SQL Statement:

```
select * from stadium
where country = 'Australia' and capacity > 10000;
```

Snapshot of the output:

	name [PK] character varying (50)	city character varying (50)	country character varying (50)	capacity integer
1	WACA Ground	Perth	Australia	24500
2	Melbourne Cricket Ground	Melbourne	Australia	100024
3	Sydney Cricket Ground	Sydney	Australia	48000
4	Adelaide Oval	Adelaide	Australia	53583
5	The Gabba	Brisbane	Australia	36000

Query #2:

Query Statement:

Show name, team ID, date of birth and wages of all players.

Relational Algebra Expression:

$$\Pi_{name,\ team\_id,\ date\_of\_birth,\ wage}\ (\text{player} \bowtie \text{plays})$$

SQL Statement:

```
select name, team_id, date_of_birth, wage
from player natural join plays;
```

Snapshot of the output:

name character varying (50)	team_id integer	date_of_birth date	wage numeric (10,2)
Eoin Morgan	1	1986-09-10	120000.00
Alex Hales	1	1989-01-03	75000.00
Joe Root	1	1990-12-30	150000.00
Jason Roy	1	1990-07-21	80000.00
Moeen Ali	1	1987-06-18	100000.00
Ben Stokes	1	1991-06-04	150000.00
David Willey	1	1990-02-28	50000.00
Chris Woakes	1	1989-03-02	75500.00
Jonny Bairstow	1	1989-09-26	105000.00
Sam Billings	1	1991-06-15	64500.00
Jos Buttler	1	1990-09-08	130000.00
Jake Ball	1	1991-03-14	55000.00
Liam Plunkett	1	1985-04-06	85000.00
Adil Rashid	1	1988-02-17	97500.00
Mark Wood	1	1990-01-11	100000.00
Tamim Iqbal	2	1989-03-20	105000.00

Since the number of tuples in the query output is too large, only a few are shown.

Query #3:

Query Statement:

Show cross product of player and team tables.

Relational Algebra Expression:

$$\Pi_{player\_id\ ,\ player.name,\ date\_of\_birth,\ team\_id,\ team.full\_name,\ team.budget}\ (\text{player} \times \text{team})$$

SQL Statement:

```
select *
from player,team;
```

Snapshot of the output:

player_id integer	name character varying (50)	date_of_birth date	team_id integer	full_name character varying (70)	budget numeric (15,2)
101	Eoin Morgan	1986-09-10	1	England	59000000.00
102	Alex Hales	1989-01-03	1	England	59000000.00
103	Joe Root	1990-12-30	1	England	59000000.00
104	Jason Roy	1990-07-21	1	England	59000000.00
105	Moeen Ali	1987-06-18	1	England	59000000.00
106	Ben Stokes	1991-06-04	1	England	59000000.00
107	David Willey	1990-02-28	1	England	59000000.00
108	Chris Woakes	1989-03-02	1	England	59000000.00
109	Jonny Bairstow	1989-09-26	1	England	59000000.00
110	Sam Billings	1991-06-15	1	England	59000000.00
111	Jos Buttler	1990-09-08	1	England	59000000.00
112	Jake Ball	1991-03-14	1	England	59000000.00
113	Liam Plunkett	1985-04-06	1	England	59000000.00
114	Adil Rashid	1988-02-17	1	England	59000000.00
115	Mark Wood	1990-01-11	1	England	59000000.00
201	Tamim Iqbal	1989-03-20	1	England	59000000.00

Since the number of tuples in the query output is too large, only a few are shown.

Query #4:

Query Statement:

Show names of teams which have not played any matches.

Relational Algebra Expression:

$\prod_{full\_name} (\prod_{team\_id}(\text{team}) - (\prod_{team\_id}(\prod_{home\_team\_id}(\text{matches}) \cup \prod_{away\_team\_id}(\text{matches}))))$

SQL Statement:

```
select full_name
from team
where team_id not in
((select home_team_id
  from matches)
 union
 (select away_team_id
  from matches));
```

Snapshot of the output:

full_name character varying (70)
West Indies
Afghanistan

Query #5:

Query Statement:

Show names of players whose name contains ‘Khan’ as a substring.

Relational Algebra Expression:

N/A

SQL Statement:

```
select name
from player
where name like '%Khan%';
```

Snapshot of the output:

name character varying (50)
Shadab Khan
Junaid Khan
Rashid Khan

Query #6:

Query Statement:

Find the cricketers whose wage is higher than that of all cricketers from Bangladesh.

Relational Algebra Expression:

$$\Pi_{name}(\sigma_{wage >(\Pi_{wage}(plays) - \Pi_{plays.wage}(\sigma_{plays.x<d.x}(A \times \rho_{d}(\sigma_{plays.team\_id= (\Pi_{team\_id}(\sigma_{full\_name = 'Bangladesh'}(team)))))))}(player \bowtie plays))$$

SQL Statement:

```
select name
from player natural join plays
where wage > all (select wage
                  from plays
                  where team_id = (select team_id
                                   from team
                                   where full_name = 'Bangladesh'));
```

Snapshot of the output:

name character varying (50)
Joe Root
Ben Stokes
Steven Smith
David Warner
Kane Williamson
Virat Kohli

Query #7:

Query Statement:

Show the names of the stadiums in which a century has been scored.

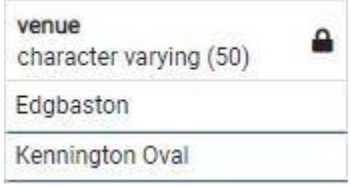
Relational Algebra Expression:

$$\Pi_{venue}(\sigma_{(\sigma_{S.match\_id = T.match\_id \wedge runs\_scored >= 100}(\Pi_{s}(\text{batting})))}(\Pi_{T}(\text{matches}))$$

SQL Statement:

```
select distinct(venue)
from matches as T
where exists (select *
              from batting as S
              where S.match_id = T.match_id
              and runs_scored >=100);
```

Snapshot of the output:



venue character varying (50)
Edgbaston
Kennington Oval

Query #8:

Query Statement:

Show the list of coaches currently in duty sorted according to their wage in descending order.

Relational Algebra Expression:

$$\Pi_{name,\ wage}(\tau_{wage\ DESC}(\text{coach} \bowtie \text{coaches}))$$

SQL Statement:

```
select name, wage
from coach natural join coaches
order by wage desc;
```

Snapshot of the output:

name character varying (50)	wage numeric (10,2)
Anil Kumble	150000.00
Trevor Bayliss	120000.00
Darren Lehmann	117500.00
Russell Domingo	110000.00
Chandika Hathurusingha	100000.00
Mickey Arthur	97500.00
Mike Hesson	90000.00
Graham Ford	84500.00
Stuart Law	72000.00
Lalchand Rajput	65500.00

Query #9:

Query Statement:

Find the total runs scored by Virat Kohli.

Relational Algebra Expression:

$\Pi_{sum(runs\_scored)}(\sigma_{player\_id = (\Pi player\_id\ (\sigma_{name = 'ViratKohli'}(player))}(\gamma_{sum(runs\_scored)}(batting)))$

SQL Statement:

```
select sum(runs_scored)
from batting
where player_id = (select player_id
                   from player
                   where name = 'Virat Kohli');
```

Snapshot of the output:

sum bigint
258

Query #10:

Query Statement:

Find names of players whose salary is more than any coach.

Relational Algebra Expression:

highly_paid_coach$\leftarrow \Pi_{wage(coaches)}$ - $\Pi_{coaches.wage}(\sigma_{coaches.x<d.x}(A \times \boldsymbol{\rho_d}$ **(coaches))**)

$(\sigma_{plays.wage>highly\_paid\_coach.salary}\ ((player \bowtie plays) \times highly\_paid\_coach))$

SQL Statement:

```
with highly_paid_coach(salary) as
(select max(wage) from coaches)
select name
from player natural join plays,highly_paid_coach
where plays.wage > highly_paid_coach.salary;
```

Snapshot of the output:

name character varying (50)
Steven Smith
Virat Kohli

Query #11:

Query Statement:

Find the ID and average wage of the teams whose average wage is more than 80000.

Relational Algebra Expression:

$$\Pi_{team\_id,\ avg\_wage}(\sigma_{avg\_wage > 80000}\ (_{team\_id}\gamma_{avg(wage)\ as\ avg\_wage}(plays)))$$

SQL Statement:

```
select team_id, avg(wage) as avg_wage
from plays
group by team_id
having avg(wage)>80000;
```

Snapshot of the output:

team_id integer	avg_wage numeric
7	81066.666666666667
1	95833.333333333333
5	92866.666666666667
2	86383.333333333333
6	82666.666666666667
3	102933.333333333333

Query #12:

Query Statement:

Find the names of players from South Africa who have higher wages than that of any player from Australia.

Relational Algebra Expression:

$\Pi_{player\_name,\ wage}\ (\sigma_{team\_id\ =\ (\Pi team\_id\ (\sigma_{full\_ame\ =\ 'South\ Africa'}\ (team))) \wedge\ (\sigma_{P.wage>Q.wage \wedge Q.full\_name\ =\ 'Australia'}\ (\rho_{P}\ (plays) \times \rho_{Q}(plays))))}\ ((player \bowtie plays))$

SQL Statement:

```
select player.name, wage
from player natural join plays
where team_id = (select team_id
                 from team
                 where full_name = 'South Africa')
and wage > any(select wage
               from plays
               where team_id = (select team_id
                                from team
                                where full_name = 'Australia'));
```

Snapshot of the output:

name character varying (50)	wage numeric (10,2)
Hashim Amla	115000.00
Faf du Plessis	100000.00
David Miller	76000.00
Jean-Paul Duminy	68000.00
Chris Morris	90000.00
Wayne Parnell	61000.00
Quinton de Kock	105000.00
AB de Villiers	135000.00
Kagiso Rabada	100000.00
Imran Tahir	88000.00
Keshav Maharaj	60000.00

Query #13:

Query Statement:

Show all information about all the players in the database. Use null values if any player does not play for any team.

Relational Algebra Expression:

$\Pi_{name,\ country,\ team\_id,\ wage}$ (player ⟕ plays)

SQL Statement:

```
select *
from player natural left outer join plays;
```

Snapshot of the output:

player_id integer	name character varying (50)	date_of_birth date	team_id integer	wage numeric (10,2)
1012	Gulbadin Naib	1991-03-16	10	70500.00
1013	Dawlat Zadran	1988-03-19	10	45500.00
1014	Mujeeb Ur Rahman	2001-03-28	10	60000.00
1015	Shapoor Zadran	1987-07-08	10	40500.00
1101	Ricky Ponting	1972-08-14	[null]	[null]
1102	Glenn McGrath	1975-12-29	[null]	[null]
1103	Shane Warne	1976-07-15	[null]	[null]
1201	Rahul Dravid	1973-11-01	[null]	[null]
1202	Sachin Tendulkar	1975-08-30	[null]	[null]
1203	Sourav Ganguly	1974-05-13	[null]	[null]
1301	Brian Lara	1972-04-02	[null]	[null]
1401	Inzamam Ul Haq	1974-01-17	[null]	[null]
1402	Shoaib Akhtar	1977-06-20	[null]	[null]
1501	Habibul Bashar	1976-11-19	[null]	[null]
1502	Khaled Masud	1981-03-22	[null]	[null]

Since the number of tuples in the query output is too large, only a few are shown.

Query #14:

Query Statement:

Increase the wage of the players from Sri Lanka by 20%.

Relational Algebra Expression:

$$plays \leftarrow (plays - \sigma_{team\_id = \prod_{team\_id}(\sigma_{full\_name = 'Srilanka'}(team))}(plays)) \cup (\prod_{player\_id,\ team\_id,\ wage*1.2}(\sigma_{full\_name = 'Srilanka'}(team))(plays))$$

SQL Statement:

```
update plays
set wage = wage * 1.2
where team_id = (select team_id
                 from team
                 where full_name = 'Sri Lanka');
```

Snapshot of the output:

Before update:

team_id integer	player_id integer	wage numeric (10,2)	full_name character varying (70)
8	801	70500.00	Sri Lanka
8	802	78500.00	Sri Lanka
8	803	59000.00	Sri Lanka
8	804	91000.00	Sri Lanka
8	805	42500.00	Sri Lanka
8	806	53000.00	Sri Lanka
8	807	55000.00	Sri Lanka
8	808	85000.00	Sri Lanka
8	809	80500.00	Sri Lanka
8	810	83000.00	Sri Lanka
8	811	93500.00	Sri Lanka
8	812	99000.00	Sri Lanka
8	813	70000.00	Sri Lanka
8	814	50500.00	Sri Lanka
8	815	44000.00	Sri Lanka

After update:

team_id integer	player_id integer	wage numeric (10,2)	full_name character varying (70)
8	801	84600.00	Sri Lanka
8	802	94200.00	Sri Lanka
8	803	70800.00	Sri Lanka
8	804	109200.00	Sri Lanka
8	805	51000.00	Sri Lanka
8	806	63600.00	Sri Lanka
8	807	66000.00	Sri Lanka
8	808	102000.00	Sri Lanka
8	809	96600.00	Sri Lanka
8	810	99600.00	Sri Lanka
8	811	112200.00	Sri Lanka
8	812	118800.00	Sri Lanka
8	813	84000.00	Sri Lanka
8	814	60600.00	Sri Lanka
8	815	52800.00	Sri Lanka

Query #15:

Delete the stadium ‘Harare Sports Club’.

Relational Algebra Expression:

$stadium \leftarrow (stadium - \sigma_{name = \text{'Harare Sports Club'}}(stadium))$

SQL Statement:

```
delete from stadium
where name = 'Harare Sports Club';
```

Snapshot of the output:

Before deletion:

name [PK] character varying (50)	city character varying (50)	country character varying (50)	capacity integer
National Stadium	Karachi	Pakistan	34228
Wanderers Park	Johannesburg	South Africa	28000
Kingsmead Cricket Ground	Durban	South Africa	25000
Newlands Cricket Ground	Cape Town	South Africa	25000
R. Premadasa Stadium	Colombo	Sri Lanka	35002
Harare Sports Club	Harare	Zimbabwe	10000
Kensington Oval	Bridgetown	West Indies	28000
Sabina Park	Kingston	West Indies	20000
Queen's Park Oval	Port Of Spain	West Indies	25000
Brian Lara Stadium	Gasparillo	West Indies	15000
Sheikh Zayed Cricket Stadium	Abu Dhabi	UAE	20000
Sharjah Cricket Association Stadi...	Sharjah	UAE	15000
Dubai International Cricket Stadium	Dubai	UAE	25000
Edgbaston	Birmingham	England	24803

After deletion:

name [PK] character varying (50)	city character varying (50)	country character varying (50)	capacity integer
Multan Cricket Stadium	Multan	Pakistan	30000
National Stadium	Karachi	Pakistan	34228
Wanderers Park	Johannesburg	South Africa	28000
Kingsmead Cricket Ground	Durban	South Africa	25000
Newlands Cricket Ground	Cape Town	South Africa	25000
R. Premadasa Stadium	Colombo	Sri Lanka	35002
Kensington Oval	Bridgetown	West Indies	28000
Sabina Park	Kingston	West Indies	20000
Queen's Park Oval	Port Of Spain	West Indies	25000
Brian Lara Stadium	Gasparillo	West Indies	15000
Sheikh Zayed Cricket Stadium	Abu Dhabi	UAE	20000
Sharjah Cricket Association Stadi...	Sharjah	UAE	15000
Dubai International Cricket Stadium	Dubai	UAE	25000
Edgbaston	Birmingham	England	24803

Query #16:

Query Statement:

Show the runs scored and balls played by the batsman whose player ID is 802 in the match with match ID 6.

Relational Algebra Expression:

$$\Pi_{runs\_scored,\ balls\_played}(\sigma_{match\_id = \text{“6”}} \wedge_{player\_id = \text{“802”}} (\text{batting}))$$

SQL Statement:

```
create view batting_stats_in_match_6 as
select *
from batting
where match_id = '6';

select runs_scored, balls_played
from batting_stats_in_match_6
where player_id = '802';
```

Snapshot of the output:

runs_scored integer	balls_played integer
89	93

Functional dependencies:

- player schema
  player_id -> name, date_of_birth
- coach schema
  coach_id -> name, country, date_of_birth
- team schema
  team_id -> full_name, budget
- plays schema
  player_id -> team_id, wage
- coaches schema
  coach_id -> team_id, wage
- matches schema
  match_id -> home_team_id, away_team_id, date_of_match, venue, home_team_runs, away_team_runs, winner
- batting schema
  match_id, player_id -> runs_scored, balls_played, boundaries_hit
- bowling schema
  match_id, player_id -> wickets_taken, balls_bowled, runs_conceded

Proof that the schemas are in desired normal form:

For a schema to be in Boyce-Codd's Normal Form, for all functional dependencies of the form a -> b must satisfy at least one of the following conditions:

1. The functional dependency is trivial.
2. a is a superkey for the schema.

For all the aforementioned schemas, the left hand sides of all the functional dependencies are superkeys for respective schemas. Therefore, the schemas are in Boyce-Codd's Normal Form. And since all schemas in BCNF must be in 3NF, the schemas are in the third normal form as well.

Canonical cover:

Since all the functional dependencies over all schemas are primary key constraints, there are no redundant dependencies or redundant parts of dependencies. Therefore, the functional dependencies are already in the canonical cover for all schemas.

Conclusion:

The database contains individual details as well as collective information of players, coaches, and teams. It can reflect the overall performance of a player or a team in a match or over several matches. Implementing the database has been an educative experience since it has helped me to get familiarized with the intricate details of SQL along with the big picture of planning a complete database.