

# FANTASY HELPER DATABASE

DBMS GROUP PROJECT #66

## Best11

### TEAM MEMBERS

Ajeet Yadav	2019010
Ananya Kansal	2019458
Anisha Bansal	2019234
Ankit Kumar Kamat	2019020
Rishita Chauhan	2019383

### TEAM CONTRIBUTION

Everybody worked hard for the project and contributed equally. We all would come up on google meet and do our work unanimously.

### INTRODUCTION

[Fantasy cricket](#) is a part of the fantasy sports genre. It is an online game in which a virtual team of real cricket players is created and points are scored depending on how those players perform in real-life matches. To win a tournament, players must work towards attaining the maximum points and the highest rank on the leaderboard. There are various fantasy league platforms on which one can play these fantasy games, for example [Dream11](#), [My11Circle](#), [Gamezy](#), etc. When we toss a coin we don't know the exact outcome of the toss, but in case of these real-life matches, when there are actual players playing, we can predict the outcome to an extent by collecting and

analyzing their performance in every single match. Now as we know there are such a large number of players in a tournament like [IPL\(Indian Premier League\)](#), it would get very difficult to check every players' stats for a single fantasy player, thus we aim to build a statistical database with its variety of attributes that can be used by a fantasy player to select the players for his fantasy team. We will be focusing on IPL only, with 8 different teams, having 11 players each. Our database will contain each player's past 5-10 matches' performance. We will have certain parameters or we can say mathematical equations which we will use to give points or ratings or we can say a "selection preference percentage".

For example- Taking a batsman Virat Kohli from RCB, we will have his runs scored, balls faced, no. of innings, no. of sixes, no. of fours, half centuries and centuries, no. of times out on spin, no. of times out on pace.

These attributes will be used to get derived attributes like strike rate(runs/balls), average(runs/innings), 4s hitting rate(4s/balls), etc., on each of these derived attributes we will give a certain criteria for rating each batsman and comparing them.

Similarly we will have different attributes and criterias for all rounders, wicket keepers and bowlers.

Batting Points	
Run	+1
Boundary Bonus	+1
Six Bonus	+2
Half-century Bonus	+8
Century Bonus	+16
Dismissal for a duck Batsman, Wicket-Keeper & All-Rounder	-2

Also the fantasy platforms play an important role in giving preference to certain categories based on their [pointing systems](#). For eg. a fantasy platform having a higher bonus for 6 runs, will prefer a batsman with higher 6s percentage or strike rate.

**Note:** This is not going to be a fantasy application for playing fantasy cricket, it's a fantasy cricket helper database.

# WEEKLY REPORT

## Week 1-2 : Identifying Stakeholders

### STAKEHOLDERS

Whenever developing a product , the foremost thing that comes to our mind is who will be the one using our product/technology and ultimately get affected by its development. Those people or organisations who are getting affected and have vested interest in the final product are the ones we refer to as “Stakeholders”. While designing any technology, it is desired to make sure that we keep in mind the needs and interests of each stakeholder and try to incorporate their inputs as well, which if not done will lead to waste of investment as stakeholders will be the ones who are ultimately going to use our software/technology. Overall, involving all stakeholders from the beginning is the most impactful step for the success of any project.

Keeping this in mind, stakeholders that we identified in our project are:

1. Administrator
2. Teams
3. Fantasy Player
4. Fantasy Platform
5. Cricket Analysts

---

# ROLES ASSOCIATED WITH EACH STAKEHOLDER

## 1. ADMINISTRATOR:

This person is the one who will be responsible for updating, managing and maintaining the database system. He/She will keep track of the past performances of all the players in the IPL teams and will periodically revise the database accordingly. The administrator has full editing and viewing rights to the database.

Some of the queries relevant to this stakeholder would be:

- Creates the database tables and injects the information available about the performances of the players into the database.
- To update/alter the tables and attributes of the database periodically.
- To make sure that the access and viewing rights of any stakeholder is not violated.
- Enrolling new users and maintaining system security.
- Optimizing and monitoring the performance of the database.

## 2. TEAM:

There will be 8 different IPL teams in our database with each team having 11 players, with 1 or 2 wicket keepers, 3 or 4 batsmen, 3 or 4 bowlers and 3 all rounders. Now each team has viewing access to every other teams' data, but has editing access for some of the attributes only for its own players.

Some of the queries relevant to this stakeholder would be:

- To update/alter the attributes of its own players in the database.
- To maintain the different strengths/weaknesses of its players against other players.
- To maintain the performance of players in homeground or away matches.

### 3. FANTASY PLAYER:

This stakeholder will be the main user of the database. They will access the database for fetching information that would help them choose the best 11 players to play their bets. They will not have editing rights for the application and hence cannot make any changes to the database. They will have only view access.

Queries associated to this stakeholder would be :-

- Choose the best 11 players
- Which player has the maximum strike rate
- Which bowler has taken the maximum wickets
- Which bowler has delivered maximum maiden overs
- Which batsman has hit the maximum number of boundaries and sixes

### 4. FANTASY PLATFORM:

Since our database application is for the people who bet on different Fantasy league platforms in IPL. So this stakeholder will be the instance of those such fantasy league platforms.

Our database application Administrator will query this stakeholder to gather information like no. of users registered on their platform and other such info and then feed the collected data in the database. Each Fantasy Platform would favour one of the players type- like batsmen, bowlers , keeper etc

## 5. CRICKET ANALYSTS:

The database would be helpful for people to analyse the performance of various cricketers and map them to various factors such as the team they are playing against, age of the cricketers, team composition etc. This can help to generate predictive match models based purely on statistical data. IPL teams can look far beyond averages when bidding on players and finding out highly specific skills that players have.

Queries associated:

- Performance of players with age in a given range
- Best performance of a player under which team composition
- Selecting captain of a team- analysing consistency/past performances
- Weak and strong points of a team against other teams
- Putting which players under one team - strengths and weaknesses



## Week 5-6 : Table Schemas

### 1. Fantasy Player

```
{
    User_tag:          VARCHAR(20) , NOT NULL
    Username:          VARCHAR(200) , NOT NULL
    Email:              VARCHAR(200) , NOT NULL
    Phone_Number:      VARCHAR(10)
    platformName       VARCHAR(20)
    Primary Key(User_tag)
}
```

Primary key constraint applicable to every table would ensure that the attribute is unique and Not NULL.

Username and email cannot be NULL whereas phone number and platformName are optional. Phone numbers should contain 10 numeric characters.

### 2. Batsman

```
{
    BatsmanID:         INT , NOT NULL
    Name:              VARCHAR(50) , NOT NULL
    Age:               INT , NOT NULL, CHECK (18<=Age<=60)
    Nationality:       VARCHAR(50), NOT NULL
    Year:              VARCHAR(10) , NOT NULL
    Matches:           INT , NOT NULL, CHECK(Matches>=0)
    Not Outs:          INT , CHECK( Not outs>=0)
    Runs:              INT, CHECK( Not outs>=0)
}
```



```

Highest Score:      INT, CHECK(highScore>=0)
battingAvg:         DECIMAL(10,3), CHECK( battingAvg>=0)
ballsFaced:         INT, CHECK( ballsFaced>=0)
battingSR:          DECIMAL(10,3), CHECK( battingSR>=0)
100s:               INT, CHECK( 100s>=0)
50s:                INT , CHECK( 50s>=0)
4s:                 INT, CHECK( 4s>=0)
6s:                 INT, CHECK( 6s>=0)
Catches:            INT, CHECK( catches>=0)
Team:               VARCHAR(100), not null
Primary Key(BatsmanID)
}

```

For the entities Batsman, Bowler, Wicket Keeper and All Rounder -Name, age, year, matches and Nationality attribute cannot be Null. All the attributes having int/ float value cannot be negative. Team attribute of these players is a foreign key, referencing the relation Team.

### 3. Bowler

```

{
    BowlerID:         INT, NOT NULL
    Name:             VARCHAR(100), NOT NULL
    Age:              INT, NOT NULL CHECK(Age<=60)
    Nationality:      VARCHAR(50) , NOT NULL
    Year:             VARCHAR(10) , NOT NULL
    Matches:          INT, NOT NULL CHECK(Matches>=0)
    Balls:            INT, CHECK(Balls>=0)
    runsGiven:        INT, CHECK(Runs>=0)
    WKTS:             INT, CHECK(Wickets>=0)
    BBM:              INT, CHECK(BBM>=0)
}

```

```

Average:          DECIMAL(10,3), CHECK(Average>=0)
Economy:          DECIMAL(10,3), CHECK(Economy>=0)
BowlingSR:        DECIMAL(10,3), CHECK(BowlingSR>=0)
4W:               INT, CHECK(4W>=0)
5W:               INT, CHECK(5W>=0)
Team:             VARCHAR(20), NOT NULL
Primary Key(BowlerID)
}

```

#### 4. WicketKeeper

```

{
    keeperID:      INT , NOT NULL
    Name:          VARCHAR(50) , NOT NULL
    Age:           INT  , NOT NULL, CHECK (18<=Age<=60)
    Nationality:   VARCHAR(50),  NOT NULL
    Year:          VARCHAR(10) , NOT NULL
    Matches:       INT , NOT NULL, CHECK(Matches>=0)
    Not Outs:      INT , CHECK( Not outs>=0)
    Runs:          INT, CHECK( Not outs>=0)
    Highest Score: INT, CHECK(highScore>=0)
    Average:       DECIMAL(10,3), CHECK( battingAvg>=0)
    ballsFaced:    INT, CHECK( ballsFaced>=0)
    Strike Rate:   DECIMAL(10,3), CHECK( battingSR>=0)
    100s:          INT, CHECK( 100s>=0)
    50s:           INT , CHECK( 50s>=0)
    4s:            INT, CHECK( 4s>=0)
    6s:            INT, CHECK( 6s>=0)
    Catches:       INT, CHECK( catches>=0)
}

```

```

Team: VARCHAR(100), not null
Stumpings: INT, check(stumpings>=0)
Primary Key(KeeperID)
}

```

## 5. AllRounder

```

{
    allRounderID: INT, NOT NULL
    Name: VARCHAR(50) , NOT NULL
    Age: INT, NOT NULL, CHECK (18<=Age<=60)
    Nationality: VARCHAR(50), NOT NULL
    Year: VARCHAR(10) , NOT NULL
    Matches: INT , NOT NULL, CHECK(Matches>=0)
    Not Outs: INT , CHECK( Not outs>=0)
    Runs: INT, CHECK( Not outs>=0)
    Highest Score: INT, CHECK(highScore>=0)
    battingAvg: DECIMAL(10,3), CHECK( battingAvg>=0)
    ballsFaced: INT, CHECK( ballsFaced>=0)
    battingSR: DECIMAL(10,3), CHECK( battingSR>=0)
    100s: INT, CHECK( 100s>=0)
    50s: INT , CHECK( 50s>=0)
    4s: INT, CHECK( 4s>=0)
    6s: INT, CHECK( 6s>=0)
    Catches: INT, CHECK( catches>=0)
    Balls: INT, CHECK(Balls>=0)
    runsGiven: INT, CHECK(Runs>=0)
    WKTS: INT, CHECK(Wickets>=0)
}

```

```

    BBM:                INT, CHECK(BBM>=0)
    bowlingAvg:         DECIMAL(10,3), CHECK(Average>=0)
    Economy:            DECIMAL(10,3), CHECK(Economy>=0)
    BowlingSR:          DECIMAL(10,3), CHECK(BowlingSR>=0)
    4W:                 INT, CHECK(4W>=0)
    5W:                 INT, CHECK(5W>=0)
    Team:               VARCHAR(20), NOT NULL
    Primary Key(allRounderID)
}

```

## 6. FantasyPlatform

```

{
    PlatformID:          int, NOT NULL
    platformName:        VARCHAR(20), NOT NULL
    No. of users:        INT, CHECK(no. Of users>=0)
    Favour:              VARCHAR(20)
    Primary Key(PlatformID)
}

```

No of users cannot be negative.

## 7. PointSystem

```

{
    FantasyPlatform ID   VARCHAR(20)
    Dismissal for a duck  INT
    Run                  INT
    Half Century Bonus    INT
    Century Bonus         INT
}

```

---

Six Bonus	INT
Four Bonus	INT
Maiden Over	INT
Wicket	INT
4 Wicket haul bonus	INT
5 Wicket haul bonus	INT
SR ( 0.6-0.7 )	INT
SR ( 0.5-0.59 )	INT
SR ( <0.5 )	INT
ER ( <4 )	INT
ER ( >10)	INT
ER ( 4-4.99)	INT
ER ( 6-10)	INT
ER ( 5-6)	INT

}

Point system is a weak entity and is associated with a strong entity which is Fantasy Platform . The primary key of the fantasy platform will act as a foreign key for the point system and will be its primary key too. Some of these attributes having data type int can be null.

## 8. Teams

{

TeamName:	VARCHAR(100), NOT NULL
City:	VARCHAR(200) , NOT NULL
HomeGround:	VARCHAR(200), NOT NULL
Primary Key(TeamName)	

}

For the analysis component we have 4 tables separately for different players and the analysis entity is a weak entity associated with a strong entity which is the player itself. The id of the player in the analysis table (foreign key) will be used as the primary key to uniquely identify the performance analysis of that particular player. The constraints involved will be that all the values of the data type float will lie between 0 and 10 and cannot be null.

## 9. Batsman Analysis

{	
BatsmanID (primary key):	VARCHAR(20)
Total points	DECIMAL(4,2)
High score points	DECIMAL(4,2)
Average points	DECIMAL(4,2)
Strike rate points	DECIMAL(4,2)
100s points	DECIMAL(4,2)
4s rate points	DECIMAL(4,2)
6s rate points	DECIMAL(4,2)
50s points	DECIMAL(4,2)
Catches rate points	DECIMAL(4,2)
Experience	DECIMAL(4,2)
}	

## 10. Bowler Analysis

{	
BowlerID (Primary Key):	VARCHAR(20)
Total points	DECIMAL(4,2)
Wicket rate points	DECIMAL(4,2)
Average points	DECIMAL(4,2)
Strike rate points	DECIMAL(4,2)

---

5 wicket points	DECIMAL(4,2)
4 wicket points	DECIMAL(4,2)
Economy points	DECIMAL(4,2)
Best bowling figure points	DECIMAL(4,2)
Experience	DECIMAL(4,2)

### 11. wicketKeeper Analysis

KeeperID (Primary Key):	VARCHAR(20)
Total points	DECIMAL(4,2)
High score points	DECIMAL(4,2)
Average points	DECIMAL(4,2)
Strike rate points	DECIMAL(4,2)
100s points`	DECIMAL(4,2)
4s rate points	DECIMAL(4,2)
6s rate points	DECIMAL(4,2)
50s points	DECIMAL(4,2)
Catches rate points	DECIMAL(4,2)
Stumping rate points	DECIMAL(4,2)
Experience	DECIMAL(4,2)

### 12. AllRounder Analysis

AllRounderID (Primary Key):	VARCHAR(20)
Total points	DECIMAL(4,2)
Wicket rate points	DECIMAL(4,2)
High score points	DECIMAL(4,2)

---

100s points	DECIMAL(4,2)
4s rate points	DECIMAL(4,2)
6s rate points	DECIMAL(4,2)
50s points	DECIMAL(4,2)
Average points	DECIMAL(4,2)
Strike rate points	DECIMAL(4,2)
5 wicket points	DECIMAL(4,2)
4 wicket points	DECIMAL(4,2)
Economy points	DECIMAL(4,2)
Best bowling figure points	DECIMAL(4,2)
Catches rate points	DECIMAL(4,2)
Experience	DECIMAL(4,2)

}



## Week 7 : Normalisation, Indexing and Writing Algebraic Queries

All the tables were normalised to 1NF, 2NF , 3NF and BCNF step by step. Following are the normalised tables:

### 1. Fantasy Player

{User\_tag, Username, Email, Phone\_Number, platformName}

### 2. Player

{PlayerID, Name, Age, Nationality}

### 3. Batsman

{ID, PlayerID, Year, Matches, Not Outs, Runs, Highest Score, battingAvg, ballsFaced, battingSR, 100s, 50s, 4s, 6s, Catches, Team}

### 4. Bowler

{ID, PlayerID, Year, Matches, Balls, runsGiven, WKTS, BBM, Average, Economy, BowlingSR, 4W, 5W, Team}

### 5. WicketKeeper

{ID, PlayerID, Year, Matches, Not Outs, Runs, Highest Score, Average, ballsFaced, Strike Rate, 100s, 50s, 4s, 6s, Catches, Team, Stumpings}

## 6. AllRounder

{ID, PlayerID, Year, Matches, Not Outs, Runs, Highest Score, battingAvg, ballsFaced, battingSR, 100s, 50s, 4s, 6s, Catches, Balls, runsGiven, WKTS, BBM, bowlingAvg, Economy, BowlingSR, 4W, 5W, Team}

## 7. FantasyPlatform

{PlatformID, No. of users, Favour}

## 8. PointSystem

{FantasyPlatform ID, Dismissal for a duck, Run, Half Century Bonus, Century Bonus, Six Bonus, Four Bonus, Maiden Over, Wicket, 4 Wicket haul bonus, 5 Wicket haul bonus, SR ( 0.6-0.7 ), SR ( 0.5-0.59 ), SR ( <0.5 ), ER(<4), ER ( >10), ER ( 4-4.99), ER ( 6-10), ER ( 5-6)}

## 9. Teams

{TeamName, City, HomeGround}

## 10. Batsman Analysis

{BatsmanID, Total points, High score points, Average points, Strike rate points, 100s points, 4s rate points, 6s rate points, 50s points, Catches rate points, Experience}

## 11. Bowler Analysis

{BowlerID, Total points, Wicket rate points, Average points, 5 wicket points, 4 wicket points, Economy points, Best bowling figure points, Experience}

## 12. wicketKeeper Analysis

{KeeperID, Total points, High score points, Average points, Strike rate points, 100s points, 4s rate points, 6s rate points, 50s points, Catches rate points, Stumping rate points, Experience}

## 13. AllRounder Analysis

{AllRounderID, Total points, Wicket rate points, High score points, 100s points, 4s rate points, 6s rate points, 50s points, Average points, Strike rate points, 5 wicket points, 4 wicket points, Economy points, Best bowling figure points, Catches rate points, Experience}

**[LINK TO THE UPDATED ER DIAGRAM](#)**

## Algebraic Queries

1. Find the names and details of all the Indian Players who have played for Mumbai Indians in most of their career.

```
SELECT * FROM player WHERE playerID IN ((SELECT playerID FROM batsman WHERE year='Career' AND team="Mumbai Indians") UNION ALL (SELECT playerID FROM bowler WHERE year='Career' AND team="Mumbai Indians") UNION ALL (SELECT playerID FROM allrounder WHERE year='Career' AND team="Mumbai Indians") UNION ALL (SELECT playerID FROM wicketkeeper WHERE year='Career' AND team="Mumbai Indians")) AND nationality="Indian";
```

Output:

	playerID	name	age	nationality
▶	1	Rohit Sharma	33	Indian
	2	Saurabh Tiwary	31	Indian
	3	Suryakumar Yadav	30	Indian
	31	Ishan Kishan	22	Indian
	39	Jasprit Bumrah	27	Indian
	41	Rahul Chahar	21	Indian
	67	Hardik Pandya	27	Indian
	69	Krunal Pandya	29	Indian

2. List down the names of all the Bowlers who played more than 5 matches and have taken more than 10 wickets in the year 2020.

```
SELECT name FROM player WHERE playerID IN ( SELECT playerID FROM bowler
WHERE year=2020 AND matches > 5 and wickets>10);
```

Output:

	name	
▶	Jasprit Bumrah	
	Trent Boult	
	Rahul Chahar	
	Ravi Ashwin	
	Kagiso Rabada	
	Anrich Nortje	
	Yuzvendra Chahal	
	Mohammad Siraj	
	Ravi Bishnoi	
	Pat Cummins	
	Varun Chakravarty	
	Jofra Archer	
	T natrajan	
	Sandeep Sharma	

3. List down the names of all the wicketKeepers who belong to a team whose home ground is Wankhede Stadium, in most of their Career.

```
SELECT name as 'WicketKeepers_HomeGround_Wankhede' FROM player where
playerID IN (SELECT playerID FROM wicketkeeper JOIN teams ON wicketkeeper.team =
teams.teamName WHERE homeGround="Wankhede Stadium" and year="Career");
```

Output:

	WicketKeepers_HomeGround_Wankhede
▶	Quinton De Kock
	Ishan Kishan

4. Find the PlayerID, Name, age of the youngest batsman to score a 100 in his career.

```
SELECT Batsman.playerID, name , age FROM batsman JOIN player ON
batsman.playerID=player.playerID where 100s>=1 AND age = (SELECT MIN(age) FROM
(SELECT age FROM batsman JOIN player ON player.playerID = batsman.playerID
WHERE 100s>=1 GROUP BY batsman.playerID) AS MAXAVG) GROUP BY
batsman.playerID, name, age;
```

Output:

	playerID	name	age
▶	14	Nicholas Pooran	25

---

5. Find the names and teams of all Indian Players who do not have M A Chidambaram as their home ground.

```
SELECT name,teamName FROM teams JOIN batsman ON
teams.teamName=batsman.team JOIN player ON batsman.playerID = player.playerID
WHERE homeGround != "M. A. Chidambaram Stadium" AND nationality="Indian" GROUP
BY name UNION SELECT name,teamName FROM teams JOIN bowler ON
teams.teamName=bowler.team JOIN player ON bowler.playerID = player.playerID
WHERE homeGround != "M. A. Chidambaram Stadium" AND nationality="Indian" GROUP
BY name UNION SELECT name,teamName FROM teams JOIN allrounder ON
teams.teamName=allrounder.team JOIN player ON allrounder.playerID = player.playerID
WHERE homeGround != "M. A. Chidambaram Stadium" AND nationality="Indian" GROUP
BY name UNION SELECT name,teamName FROM teams JOIN wicketkeeper ON
teams.teamName=wicketkeeper.team JOIN player ON wicketkeeper.playerID =
player.playerID WHERE homeGround != "M. A. Chidambaram Stadium" AND
nationality="Indian" GROUP BY name;
```

Output:

name	teamName
Rohit Sharma	Mumbai Indians
Saurabh Tiwary	Mumbai Indians
Saurabh Tiwary	Rising Pune Supergiant
Suryakumar Yadav	Mumbai Indians
Suresh Raina	Gujarat Lions
Shreyas Iyer	Delhi Capitals
Ajinkya Rahane	Delhi Capitals
Ajinkya Rahane	Rajasthan Royals
Prithvi Shaw	Delhi Capitals
Shikhar Dhawan	Delhi Capitals
Shikhar Dhawan	Sunrisers Hyderabad
Virat Kohli	Royal Challengers Bangalore
Gurkeerat Singh	Royal Challengers Bangalore
Devdutt Padikkal	Royal Challengers Bangalore
Shubhman Gill	Kolkata Knight Riders
Rahul Tripathi	Kolkata Knight Riders
Yashasvi Jaiswal	Rajasthan Royals
Sanju Samson	Rajasthan Royals
Manish Pandey	Sunrisers Hyderabad
Jasprit Bumrah	Mumbai Indians
Rahul Chahar	Mumbai Indians
Ravi Ashwin	Delhi Capitals
Ishant Sharma	Delhi Capitals
Yuzvendra Chahal	Royal Challengers Bangalore
Navdeep Saini	Royal Challengers Bangalore
Mohammad Siraj	Royal Challengers Bangalore
Mohammad Shami	Delhi Capitals
Murugan Ashwin	Royal Challengers Bangalore
Shubham Mavi	Sunrisers Hyderabad



6. Name the bowler who has taken the highest wickets and output his points for wickets taken in Dream 11 fantasy platform for his entire career.

```
SELECT (SELECT name from player where playerID IN (SELECT playerID from bowler
where wickets = (SELECT MAX(wickets) FROM bowler WHERE year="Career")))) AS
"BowlerWithMaximumWicketsTaken",
(SELECT (SELECT wickets FROM pointsystem where platformID in (SELECT platformID
from fantasyPlatform where platformName='Dream11')) *
(SELECT wickets FROM bowler where playerID = (SELECT playerID from bowler where
wickets = (SELECT MAX(wickets) FROM bowler WHERE year="Career")) AND
year="Career")) AS 'POINTS';
```

Output:

	BowlerWithMaximumWicketsTaken	POINTS	
▶	Ravi Ashwin	3450	

7. List down the details of the wicket keepers who belong to either mumbai indians or Kolkata knight riders and who have stumpings greater than 10.

```
SELECT * FROM player where playerID IN (SELECT playerID FROM wicketkeeper
WHERE team = "Mumbai Indians" OR team="Kolkata Knight Riders" AND stumpings>10);
```

Output:

	playerID	name	age	nationality
▶	30	Quinton De Kock	28	South African
	31	Ishan Kishan	22	Indian
	35	Dinesh Kartik	36	Indian

8. List the best 3 bowlers, 4 batsman, 3 all rounders and 1 wicket keeper each which a fantasy user can take in his team.

```

SELECT name AS "Top_Players","batsman" AS 'CATEGORY' FROM (SELECT 'batsman' ,
name, batsman.playerID FROM batsmananalysis JOIN batsman ON
batsmananalysis.ID=batsman.ID JOIN
player ON batsman.playerID=player.playerID GROUP BY batsman.playerID ,
batsmananalysis.totalPoints order by batsmananalysis.totalPoints DESC LIMIT 4) AS X
UNION ALL
SELECT name AS "Top_Players","bowler" AS 'CATEGORY' FROM (SELECT 'bowler',name,
bowler.playerID,bowleranalysis.totalPoints FROM bowleranalysis JOIN bowler ON
bowleranalysis.ID=bowler.ID JOIN
player ON bowler.playerID=player.playerID GROUP BY bowler.playerID,
bowleranalysis.totalPoints order by bowleranalysis.totalPoints DESC LIMIT 3) AS Y
UNION ALL
SELECT name AS "Top_Players", "allrounder" AS 'CATEGORY' FROM (SELECT
'allrounder',name,allrounder.playerID FROM allrounderanalysis JOIN allrounder ON
allrounderanalysis.ID=allrounder.ID JOIN
player ON allrounder.playerID=player.playerID GROUP BY allrounder.playerID,
AllRounderAnalysis.totalPoints order by allrounderanalysis.totalPoints DESC LIMIT 3)
AS Z UNION ALL
SELECT name AS "Top_Players","wicketkeeper" AS 'CATEGORY' FROM (SELECT
'wicketkeeper' ,name, wicketkeeper.playerID FROM wicketkeeperanalysis JOIN
wicketkeeper ON wicketkeeperanalysis.ID=wicketkeeper.ID JOIN

```

```
player ON wicketkeeper.playerID=player.playerID GROUP BY wicketkeeper.playerID,
wicketkeeperanalysis.totalPoints order by wicketkeeperanalysis.totalPoints DESC
LIMIT 1) AS W;
```

Output:

Top_Players	CATEGORY
Chris Gayle	batsman
David Warner	batsman
Virat Kohli	batsman
Suresh Raina	batsman
Jasprit Bumrah	bowler
Kagiso Rabada	bowler
Jasprit Bumrah	bowler
Ravindra Jadeja	allrounder
Sunil Naraine	allrounder
Kieron Pollard	allrounder
AB de Villiers	wicketkeeper

9. Give the points for runs made by the batsman Virat Kohli for his entire career in fantasy platform which is most popular and favours a batsman.

```
SELECT (SELECT run FROM pointsystem where platformID IN (SELECT platformID
FROM fantasyplatform where totalUsers IN (SELECT MAX(totalUsers) FROM
fantasyplatform GROUP BY favour) AND favour="batsman"))*
```

```
(SELECT runs FROM batsman WHERE playerID = (SELECT playerID FROM player
WHERE name="Virat Kohli") AND year="Career") AS "RunPoints_For_ViratKohli";
```

Output:

RunPoints_For_ViratKohli
▶ 23512

## 10. Name the best Australian batsman on the basis of his overall performance (total points).

```
SELECT name AS 'BEST_AUSTRALIAN_BATSMAN' FROM batsmananalysis JOIN
batsman ON batsmananalysis.ID=batsman.ID JOIN player ON batsman.playerID =
player.playerID WHERE nationality="Australian" AND
batsmananalysis.totalPoints=(SELECT MAX(totalPoints) FROM (SELECT totalPoints
FROM batsmananalysis JOIN batsman ON batsmananalysis.ID=batsman.ID JOIN player
ON batsman.playerID = player.playerID WHERE nationality="Australian") AS M) GROUP
BY player.playerID;
```

Output:

BEST_AUSTRALIAN_BATSMAN
▶ David Warner

# Indexing

Attributes used as Index in the corresponding tables:

1. BatsmanAnalysis - Total Points
2. BowlerAnalysis - Total Points
3. WicketKeeperAnalysis - TotalPoints
4. AllRounderAnalysis - TotalPoints
5. Player - Nationality
6. Batsman - PlayerID, (Year, Team)
7. Bowler - PlayerID, (Year, Team)
8. WicketKeeper - PlayerID, (Year, Team)
9. AllRounder - PlayerID, (Year, Team)
10. Team - HomeGround
11. FantasyPlatform - (favour, totalUsers)

```
CREATE INDEX year_team_wicketKeeper ON wicketKeeper(year,team);
CREATE INDEX year_team_batsman ON batsman(year,team);
CREATE INDEX year_team_bowler ON bowler(year,team);
CREATE INDEX year_team_AllRounder ON AllRounder(year,team);
CREATE INDEX num_user_favor ON FantasyPlatform(favour,totalUsers);
CREATE INDEX team_homeground ON teams(homeground);
CREATE INDEX player_nationality ON Player(nationality);
CREATE INDEX analysis_totalpoints_batsman ON batsmanAnalysis(totalpoints);
CREATE INDEX analysis_totalpoints_bowler ON bowlerAnalysis(totalpoints);
CREATE INDEX analysis_totalpoints_wicketKeeper ON wicketKeeperAnalysis(totalpoints);
CREATE INDEX analysis_totalpoints_AllRounder ON AllRounderAnalysis(totalpoints);
```

## Week 8: Embedded SQL Queries

1. Name the Fantasy Platform which is the most popular among users overall and w.r.t it favouring the bowler, batsman , wicket keeper.

Overall :- `SELECT platformName AS "MOST_POPULAR_PLATFORM" FROM fantasyplatform where totalUsers IN (SELECT MAX(totalUsers) FROM fantasyplatform);`

output:

MOST_POPULAR_PLATFORM	
Real11	

For each category :- `SELECT platformName AS "MOST_POPULAR_PLATFORM",favour FROM fantasyplatform where totalUsers IN (SELECT MAX(totalUsers) FROM fantasyplatform GROUP BY favour);`

output:

MOST_POPULAR_PLATFORM	favour	
BalleBaazi	Bowler	
FanFight	wicketKeeper	
Real11	Batsman	

2. Name the categories which are best suited for selecting the Captain & V. Captain of the fantasy team. (Captain gives 2x points, V. Captain gives 1.5x points)

```

SELECT (SELECT TOP _ CATEGORIES FROM (SELECT TOP _ CATEGORIES FROM (SELECT
"batsman" AS 'TOP _ CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
'batsman',totalPoints FROM batsmananalysis JOIN batsman ON batsmananalysis.ID=batsman.ID
JOIN player ON batsman.playerID=player.playerID GROUP BY batsman.playerID,
BatsmanAnalysis.totalPoints order by batsmananalysis.totalPoints DESC LIMIT 10) AS X UNION
ALL
SELECT "bowler" AS 'TOP _ CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
'bowler',totalPoints FROM bowleranalysis JOIN bowler ON bowleranalysis.ID=bowler.ID JOIN
player ON bowler.playerID=player.playerID GROUP BY bowler.playerID
,BowlerAnalysis.totalPoints order by bowleranalysis.totalPoints DESC LIMIT 10) AS Y UNION
ALL
SELECT 'allrounder' AS 'TOP _ CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
'allrounder',totalPoints FROM allrounderanalysis JOIN allrounder ON
allrounderanalysis.ID=allrounder.ID JOIN player ON allrounder.playerID=player.playerID
GROUP BY allrounder.playerID, AllRounderAnalysis.totalPoints order by
allrounderanalysis.totalPoints DESC LIMIT 5) AS Z UNION ALL
SELECT 'wicketkeeper' AS 'TOP _ CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
'wicketkeeper',totalPoints FROM wicketkeeperanalysis JOIN wicketkeeper ON
wicketkeeperanalysis.ID=wicketkeeper.ID JOIN player ON
wicketkeeper.playerID=player.playerID GROUP BY wicketkeeper.playerID,
WicketKeeperAnalysis.totalPoints order by wicketkeeperanalysis.totalPoints DESC LIMIT 5) AS
W ORDER BY AvgTotalPts DESC limit 2) AS W) AS Y LIMIT 1) AS "CAPTAIN",
(SELECT TOP _ CATEGORIES FROM (SELECT TOP _ CATEGORIES FROM (SELECT "batsman" AS
'TOP _ CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT 'batsman',totalPoints
FROM batsmananalysis JOIN batsman ON batsmananalysis.ID=batsman.ID JOIN player ON
batsman.playerID=player.playerID GROUP BY batsman.playerID, BatsmanAnalysis.totalPoints
order by batsmananalysis.totalPoints DESC LIMIT 10) AS X UNION ALL
SELECT "bowler" AS 'TOP _ CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
'bowler',totalPoints FROM bowleranalysis JOIN bowler ON bowleranalysis.ID=bowler.ID JOIN
player ON bowler.playerID=player.playerID GROUP BY bowler.playerID,
BowlerAnalysis.totalPoints order by bowleranalysis.totalPoints DESC LIMIT 10) AS Y UNION
ALL
SELECT 'allrounder' AS 'TOP _ CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
'allrounder',totalPoints FROM allrounderanalysis JOIN allrounder ON
allrounderanalysis.ID=allrounder.ID JOIN player ON allrounder.playerID=player.playerID
GROUP BY allrounder.playerID,AllRounderAnalysis.totalPoints order by
allrounderanalysis.totalPoints DESC LIMIT 5) AS Z UNION ALL

```

```
SELECT 'wicketkeeper' AS 'TOP_CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
'wicketkeeper',totalPoints FROM wicketkeeperanalysis JOIN wicketkeeper ON
wicketkeeperanalysis.ID=wicketkeeper.ID JOIN player ON
wicketkeeper.playerID=player.playerID GROUP BY wicketkeeper.playerID ,
WicketKeeperAnalysis.totalPoints order by wicketkeeperanalysis.totalPoints DESC LIMIT 5) AS
W ORDER BY AvgTotalPts DESC limit 2) AS W) AS Y LIMIT 1,1) AS "VICE_CAPTAIN";
```

Output:

	CAPTAIN	VICE_CAPTAIN	
▶	allrounder	wicketkeeper	

### 3. To which nationality do the maximum number of players belong? Give the no of players belonging to each nationality.

For no. of players belonging to each nationality :- SELECT nationality , COUNT(\*) AS  
'NO\_OF\_PLAYERS' FROM player group by nationality ;

Output:



	nationality	NO_OF_PLAYERS	
▶	Afghan	1	
	Australian	7	
	English	6	
	Engllish	3	
	Indian	51	
	Jamaican	2	
	New Zealander	3	
	South African	6	
	Trinidadian	3	
	West Indian	4	

For nationality having maximum players :- `SELECT nationality AS nationality_having_max_players FROM (SELECT nationality , COUNT(*) AS C FROM player group by nationality ORDER BY C DESC LIMIT 1) AS MAX_PLAYERS ;`

Output:

	nationality_having_max_players	
▶	Indian	

- List the details of the top 40 players based on their performance and then display the top teams (rank them) based on the number of best players that they have.

For best 40 players :-

```
SELECT * FROM player where playerID IN (SELECT playerID FROM (SELECT
playerID,totalPoints FROM batsman JOIN batsmananalysis ON batsman.ID=batsmananalysis.ID
GROUP BY playerID UNION ALL
SELECT playerID,totalPoints FROM bowler JOIN bowleranalysis ON
bowler.ID=bowleranalysis.ID GROUP BY playerID UNION ALL
SELECT playerID,totalPoints FROM allrounder JOIN allrounderanalysis ON
allrounder.ID=allrounderanalysis.ID GROUP BY playerID UNION ALL
SELECT playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID ORDER BY totalPoints DESC
LIMIT 40) AS X);
```

For team ranking :-

```
SELECT team AS 'TopTeams', COUNT(*) as 'No_of_TopPlayers' FROM (SELECT team,playerID
from batsman WHERE playerID IN (SELECT playerID from player where name in (SELECT name
as 'best_40_players' FROM player where playerID IN (SELECT playerID FROM (SELECT
playerID,totalPoints FROM batsman JOIN batsmananalysis ON batsman.ID=batsmananalysis.ID
GROUP BY playerID, totalPoints UNION ALL SELECT playerID,totalPoints FROM bowler JOIN
bowleranalysis ON bowler.ID=bowleranalysis.ID GROUP BY playerID, totalPoints UNION ALL
SELECT playerID,totalPoints FROM allrounder JOIN allrounderanalysis ON
allrounder.ID=allrounderanalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT
playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID, totalPoints ORDER BY
totalPoints DESC LIMIT 40) AS X))) GROUP BY team, playerID UNION ALL SELECT
team,playerID from bowler WHERE playerID IN (SELECT playerID from player where name in
(SELECT name as 'best_40_players' FROM player where playerID IN (SELECT playerID FROM
(SELECT playerID,totalPoints FROM batsman JOIN batsmananalysis ON
batsman.ID=batsmananalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT
playerID,totalPoints FROM bowler JOIN bowleranalysis ON bowler.ID=bowleranalysis.ID GROUP
BY playerID, totalPoints UNION ALL SELECT playerID,totalPoints FROM allrounder JOIN
allrounderanalysis ON allrounder.ID=allrounderanalysis.ID GROUP BY playerID, totalPoints
UNION ALL SELECT playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID, totalPoints ORDER BY
totalPoints DESC LIMIT 40) AS X))) GROUP BY team, playerID UNION ALL SELECT
team,playerID from allrounder WHERE playerID IN (SELECT playerID from player where name in
(SELECT name as 'best_40_players' FROM player where playerID IN (SELECT playerID FROM
(SELECT playerID,totalPoints FROM batsman JOIN batsmananalysis ON
batsman.ID=batsmananalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT
```

```
playerID,totalPoints FROM bowler JOIN bowleranalysis ON bowler.ID=bowleranalysis.ID GROUP
BY playerID, totalPoints UNION ALL SELECT playerID,totalPoints FROM allrounder JOIN
allrounderanalysis ON allrounder.ID=allrounderanalysis.ID GROUP BY playerID, totalPoints
UNION ALL SELECT playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID, totalPoints ORDER BY
totalPoints DESC LIMIT 40) AS X))) GROUP BY team, playerID UNION ALL SELECT
team,playerID from wicketkeeper WHERE playerID IN (SELECT playerID from player where name
in (SELECT name as 'best_40_players' FROM player where playerID IN (SELECT playerID FROM
(SELECT playerID,totalPoints FROM batsman JOIN batsmananalysis ON
batsman.ID=batsmananalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT
playerID,totalPoints FROM bowler JOIN bowleranalysis ON bowler.ID=bowleranalysis.ID GROUP
BY playerID, totalPoints UNION ALL SELECT playerID,totalPoints FROM allrounder JOIN
allrounderanalysis ON allrounder.ID=allrounderanalysis.ID GROUP BY playerID, totalPoints
UNION ALL SELECT playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID, totalPoints ORDER BY
totalPoints DESC LIMIT 40) AS X) )) GROUP BY team, playerID ) as y GROUP BY team ORDER
BY No_of_TopPlayers DESC;
```

Output:

	TopTeams	No_of_TopPlayers
▶	Mumbai Indians	5
	Chennai Super Kings	5
	Delhi Capitals	5
	Kings XI Punjab	5
	Royal Challengers Bangalore	5
	Sunrisers Hyderabad	3
	Kolkata Knight Riders	3
	Rajasthan Royals	2
	Gujarat Lions	1
	Sunriser Hyderabad	1

## Python code to run embedded queries

```
from re import M
import mysql.connector

myDb = mysql.connector.connect(host="localhost", user="root", passwd="Mysql@9695766",
database="Best")
print("DB CONNECTED!!")

myCursor = myDb.cursor()

print()
print()
print()
print()

#myCursor.execute("SELECT name AS 'HighestAvg_IndianWicketKeeper_ForYear2020' FROM (SELECT
name,player.playerID,year,battingAvg,nationality FROM wicketkeeper JOIN player ON player.playerID =
wicketkeeper.playerID WHERE nationality='Indian' AND year='2020') AS NAME WHERE battingAvg=(SELECT
MAX(battingAvg) FROM (SELECT year,battingAvg,nationality FROM wicketkeeper JOIN player ON
player.playerID = wicketkeeper.playerID WHERE nationality='Indian' AND year=2020) AS MAXAVG);")
#1 a
#myCursor.execute("SELECT platformName AS 'MOST_POPULAR_PLATFORM' FROM fantasyplatform where
totalUsers IN (SELECT MAX(totalUsers) FROM fantasyplatform);")
#1 b

#myCursor.execute("SELECT platformName AS 'MOST_POPULAR_PLATFORM',favour FROM fantasyplatform where
totalUsers IN (SELECT MAX(totalUsers) FROM fantasyplatform GROUP BY favour);")

#2
```

```
# myCursor.execute("SELECT (SELECT TOP_CATEGORIES FROM (SELECT TOP_CATEGORIES FROM (SELECT
batsman.playerID AS `TOP_CATEGORIES` ,avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
batsman.playerID,totalPoints FROM batsmananalysis JOIN batsman ON batsmananalysis.ID=batsman.ID
JOIN player ON batsman.playerID=player.playerID GROUP BY batsman.playerID,
BatsmanAnalysis.totalPoints order by batsmananalysis.totalPoints DESC LIMIT 10) AS X UNION ALL
SELECT 'bowler' AS 'TOP_CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
'bowler',totalPoints FROM bowleranalysis JOIN bowler ON bowleranalysis.ID=bowler.ID JOIN player ON
bowler.playerID=player.playerID GROUP BY bowler.playerID ,BowlerAnalysis.totalPoints order by
bowleranalysis.totalPoints DESC LIMIT 10) AS Y UNION ALL SELECT 'allrounder' AS
'TOP_CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT 'allrounder',totalPoints FROM
allrounderanalysis JOIN allrounder ON allrounderanalysis.ID=allrounder.ID JOIN player ON
allrounder.playerID=player.playerID GROUP BY allrounder.playerID, AllRounderAnalysis.totalPoints
order by allrounderanalysis.totalPoints DESC LIMIT 5) AS Z UNION ALL SELECT 'wicketkeeper' AS
'TOP_CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT 'wicketkeeper',totalPoints FROM
wicketkeeperanalysis JOIN wicketkeeper ON wicketkeeperanalysis.ID=wicketkeeper.ID JOIN player ON
wicketkeeper.playerID=player.playerID GROUP BY wicketkeeper.playerID,
WicketKeeperAnalysis.totalPoints order by wicketkeeperanalysis.totalPoints DESC LIMIT 5) AS W ORDER
BY AvgTotalPts DESC limit 2) AS W) AS Y LIMIT 1) AS 'CAPTAIN',(SELECT TOP_CATEGORIES FROM (SELECT
TOP_CATEGORIES FROM (SELECT 'batsman' AS 'TOP_CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM
(SELECT 'batsman',totalPoints FROM batsmananalysis JOIN batsman ON batsmananalysis.ID=batsman.ID
JOIN player ON batsman.playerID=player.playerID GROUP BY batsman.playerID,
BatsmanAnalysis.totalPoints order by batsmananalysis.totalPoints DESC LIMIT 10) AS X UNION ALL
SELECT 'bowler' AS 'TOP_CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT
'bowler',totalPoints FROM bowleranalysis JOIN bowler ON bowleranalysis.ID=bowler.ID JOIN player ON
bowler.playerID=player.playerID GROUP BY bowler.playerID, BowlerAnalysis.totalPoints order by
bowleranalysis.totalPoints DESC LIMIT 10) AS Y UNION ALL SELECT 'allrounder' AS
'TOP_CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT 'allrounder',totalPoints FROM
allrounderanalysis JOIN allrounder ON allrounderanalysis.ID=allrounder.ID JOIN player ON
allrounder.playerID=player.playerID GROUP BY allrounder.playerID,AllRounderAnalysis.totalPoints
order by allrounderanalysis.totalPoints DESC LIMIT 5) AS Z UNION ALL SELECT 'wicketkeeper' AS
'TOP_CATEGORIES',avg(totalPoints) AS 'AvgTotalPts' FROM (SELECT 'wicketkeeper',totalPoints FROM
wicketkeeperanalysis JOIN wicketkeeper ON wicketkeeperanalysis.ID=wicketkeeper.ID JOIN player ON
wicketkeeper.playerID=player.playerID GROUP BY wicketkeeper.playerID ,
WicketKeeperAnalysis.totalPoints order by wicketkeeperanalysis.totalPoints DESC LIMIT 5) AS W ORDER
BY AvgTotalPts DESC limit 2) AS W) AS Y LIMIT 1,1) AS 'VICE_CAPTAIN')

#3a
#myCursor.execute("SELECT nationality , COUNT(*) AS 'NO_OF_PLAYERS' FROM player group by
nationality")
```

#3b

```
#myCursor.execute("SELECT nationality AS nationality_having_max_players FROM (SELECT nationality ,
COUNT(*) AS C FROM player group by nationality ORDER BY C DESC LIMIT 1 ) AS MAX_PLAYERS;")
```

#4a

```
#myCursor.execute("SELECT * FROM player where playerID IN (SELECT playerID FROM (SELECT
playerID,totalPoints FROM batsman JOIN batsmananalysis ON batsman.ID=batsmananalysis.ID GROUP BY
playerID, totalPoints UNION ALL SELECT playerID,totalPoints FROM bowler JOIN bowleranalysis ON
bowler.ID=bowleranalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT playerID,totalPoints
FROM allrounder JOIN allrounderanalysis ON allrounder.ID=allrounderanalysis.ID GROUP BY playerID,
totalPoints UNION ALL SELECT playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID, totalPoints ORDER BY totalPoints DESC
LIMIT 40) AS X);")
```

#4b

```
#myCursor.execute("SELECT team AS 'TopTeams', COUNT(*) as 'No_of_TopPlayers' FROM (SELECT
team,playerID from batsman WHERE playerID IN (SELECT playerID from player where name in (SELECT
name as 'best_40_players' FROM player where playerID IN (SELECT playerID FROM (SELECT
playerID,totalPoints FROM batsman JOIN batsmananalysis ON batsman.ID=batsmananalysis.ID GROUP BY
playerID, totalPoints UNION ALL SELECT playerID,totalPoints FROM bowler JOIN bowleranalysis ON
bowler.ID=bowleranalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT playerID,totalPoints
FROM allrounder JOIN allrounderanalysis ON allrounder.ID=allrounderanalysis.ID GROUP BY playerID,
totalPoints UNION ALL SELECT playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID, totalPoints ORDER BY totalPoints DESC
LIMIT 40) AS X))) GROUP BY team, playerID UNION ALL SELECT team,playerID from bowler WHERE playerID
IN (SELECT playerID from player where name in (SELECT name as 'best_40_players' FROM player where
playerID IN (SELECT playerID FROM (SELECT playerID,totalPoints FROM batsman JOIN batsmananalysis ON
batsman.ID=batsmananalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT playerID,totalPoints
FROM bowler JOIN bowleranalysis ON bowler.ID=bowleranalysis.ID GROUP BY playerID, totalPoints UNION
ALL SELECT playerID,totalPoints FROM allrounder JOIN allrounderanalysis ON
allrounder.ID=allrounderanalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT
playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID, totalPoints ORDER BY totalPoints DESC
LIMIT 40) AS X))) GROUP BY team, playerID UNION ALL SELECT team,playerID from allrounder WHERE
playerID IN (SELECT playerID from player where name in (SELECT name as 'best_40_players' FROM
player where playerID IN (SELECT playerID FROM (SELECT playerID,totalPoints FROM batsman JOIN
```

```

batsmananalysis ON batsman.ID=batsmananalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT
playerID,totalPoints FROM bowler JOIN bowleranalysis ON bowler.ID=bowleranalysis.ID GROUP BY
playerID, totalPoints UNION ALL SELECT playerID,totalPoints FROM allrounder JOIN allrounderanalysis
ON allrounder.ID=allrounderanalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT
playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID, totalPoints ORDER BY totalPoints DESC
LIMIT 40) AS X))) GROUP BY team, playerID UNION ALL SELECT team,playerID from wicketkeeper WHERE
playerID IN (SELECT playerID from player where name in (SELECT name as 'best_40_players' FROM
player where playerID IN (SELECT playerID FROM (SELECT playerID,totalPoints FROM batsman JOIN
batsmananalysis ON batsman.ID=batsmananalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT
playerID,totalPoints FROM bowler JOIN bowleranalysis ON bowler.ID=bowleranalysis.ID GROUP BY
playerID, totalPoints UNION ALL SELECT playerID,totalPoints FROM allrounder JOIN allrounderanalysis
ON allrounder.ID=allrounderanalysis.ID GROUP BY playerID, totalPoints UNION ALL SELECT
playerID,totalPoints FROM wicketkeeper JOIN wicketkeeperanalysis ON
wicketkeeper.ID=wicketkeeperanalysis.ID GROUP BY playerID, totalPoints ORDER BY totalPoints DESC
LIMIT 40) AS X) )) GROUP BY team, playerID ) as y GROUP BY team ORDER BY No_of_TopPlayers DESC;")

#to display output
for i in myCursor:
    print(i)

print()
print()
print()

myDb.close()

```

## Necessary queries related to different stakeholders according to their Rights and Grants :

### Fantasy Platform

#### 1. Register new platform

```
CREATE USER '11Wickets' IDENTIFIED BY '11Wickets';
CREATE OR REPLACE SQL SECURITY DEFINER VIEW `11Wickets` AS
  SELECT * FROM `fantasyPlatform`
  WHERE
    (`fantasyPlatform`.`platformID` = 9);
CREATE OR REPLACE SQL SECURITY DEFINER VIEW `11Wickets_Points` AS
  SELECT * FROM `pointSystem`
  WHERE
    (`pointSystem`.`platformID` = 9);
CREATE OR REPLACE SQL SECURITY DEFINER VIEW `11Wickets_users` AS
  SELECT * FROM `fanatasyPlayer`
  WHERE
    (`fanatasyPlayer`.`platformName` in (select platformName from fantasyPlatform where
platformID = 9));
grant select,update,delete on `Best`.`11Wickets` to '11Wickets';
grant select,update,delete on `Best`.`11Wickets_Points` to '11Wickets';
grant select on `Best`.`11Wickets_users` to '11Wickets';
grant update (`platformName`) on `Best`.`11Wickets_users` to '11Wickets';
grant select on best.allrounder to 11Wickets;
grant select on best.allrounderanalysis to 11Wickets;
grant select on best.batsman to 11Wickets;
grant select on best.batsmananalysis to 11Wickets;
grant select on best.bowler to 11Wickets;
grant select on best.bowleranalysis to 11Wickets;
grant select on best.fantasyplatform to 11Wickets;
grant select on best.player to 11Wickets;
grant select on best.pointsystem to 11Wickets;
grant select on best.teams to 11Wickets;
```



```
grant select on best.wicketkeeper to 11Wickets;
grant select on best.wicketkeeperanalysis to 11Wickets;
show grants for 11Wickets;
```

## 2. Update existing details

```
update 11Wickets set totalUsers=20000;
```

## 3. Update the Point System Values

```
update 11Wickets_Points set run= 2;
```

## 4. Delete registered platform

```
delete from 11Wickets;
delete from 11Wickets_Points;
update 11Wickets_users set platformName = null;
```

## 5. Can view the details of its users

```
select * from 11Wickets_users;
```

## Fantasy Player

### 1. Enter Sign Up Details

```
CREATE USER 'aaleavy6r'@'localhost' IDENTIFIED BY 'aaleavy6r';
CREATE OR REPLACE SQL SECURITY DEFINER VIEW `aaleavy6r` AS
SELECT
    `fanatasyPlayer`.`userTag` AS `userTag`,
    `fanatasyPlayer`.`userName` AS `userName`,
    `fanatasyPlayer`.`email` AS `email`,
    `fanatasyPlayer`.`phoneNo` AS `phoneNo`,
```

```

`fanatasyPlayer`.`platformName` AS `platformName`
FROM
`fanatasyPlayer`
WHERE
(`fanatasyPlayer`.`userTag` = 'aaleavy6r');
grant select,update,delete on Best.aaleavy6r to 'aaleavy6r'@'localhost';
grant select on best.allrounder to 'aaleavy6r'@'localhost';
grant select on best.allrounderanalysis to 'aaleavy6r'@'localhost';
grant select on best.batsman to 'aaleavy6r'@'localhost';
grant select on best.batsmananalysis to 'aaleavy6r'@'localhost';
grant select on best.bowler to 'aaleavy6r'@'localhost';
grant select on best.bowleranalysis to 'aaleavy6r'@'localhost';
grant select on best.fantasyplatform to 'aaleavy6r'@'localhost';
grant select on best.player to 'aaleavy6r'@'localhost';
grant select on best.pointsystem to 'aaleavy6r'@'localhost';
grant select on best.teams to 'aaleavy6r'@'localhost';
grant select on best.wicketkeeper to 'aaleavy6r'@'localhost';
grant select on best.wicketkeeperanalysis to 'aaleavy6r'@'localhost';
show grants for 'aaleavy6r'@'localhost';

```

## 2. Update the existing information

```
update Best.aaleavy6r set email='aaleavy6r_newmail@gmail.com';
```

## 3. Delete existing Account

```
delete from Best.aaleavy6r;
```

## 4. Log In into the Account and view details

```
select * from Best.aaleavy6r;
```

Team

---

grant select on best.allrounder to CSKcoach;  
 grant select on best.allrounderanalysis to CSKcoach;  
 grant select on best.batsman to CSKcoach;  
 grant select on best.batsmananalysis to CSKcoach;  
 grant select on best.bowler to CSKcoach;  
 grant select on best.bowleranalysis to CSKcoach;  
 grant select on best.fantasyplatform to CSKcoach;  
 grant select on best.player to CSKcoach;  
 grant select on best.pointsystem to CSKcoach;  
 grant select on best.teams to CSKcoach;  
 grant select on best.wicketkeeper to CSKcoach;  
 grant select on best.wicketkeeperanalysis to CSKcoach;  
 grant select on best.allrounder to DCcoach;  
 grant select on best.allrounderanalysis to DCcoach;  
 grant select on best.batsman to DCcoach;  
 grant select on best.batsmananalysis to DCcoach;  
 grant select on best.bowler to DCcoach;  
 grant select on best.bowleranalysis to DCcoach;  
 grant select on best.fantasyplatform to DCcoach;  
 grant select on best.player to DCcoach;  
 grant select on best.pointsystem to DCcoach;  
 grant select on best.teams to DCcoach;  
 grant select on best.wicketkeeper to DCcoach;  
 grant select on best.wicketkeeperanalysis to DCcoach;  
 grant select on best.allrounder to KKRcoach;  
 grant select on best.allrounderanalysis to KKRcoach;  
 grant select on best.batsman to KKRcoach;  
 grant select on best.batsmananalysis to KKRcoach;  
 grant select on best.bowler to KKRcoach;  
 grant select on best.bowleranalysis to KKRcoach;  
 grant select on best.fantasyplatform to KKRcoach;  
 grant select on best.player to KKRcoach;  
 grant select on best.pointsystem to KKRcoach;  
 grant select on best.teams to KKRcoach;  
 grant select on best.wicketkeeper to KKRcoach;  
 grant select on best.wicketkeeperanalysis to KKRcoach;  
 grant select on best.allrounder to KXIPcoach;

---

grant select on best.allrounderanalysis to KXIPcoach;  
grant select on best.batsman to KXIPcoach;  
grant select on best.batsmananalysis to KXIPcoach;  
grant select on best.bowler to KXIPcoach;  
grant select on best.bowleranalysis to KXIPcoach;  
grant select on best.fantasyplatform to KXIPcoach;  
grant select on best.player to KXIPcoach;  
grant select on best.pointsystem to KXIPcoach;  
grant select on best.teams to KXIPcoach;  
grant select on best.wicketkeeper to KXIPcoach;  
grant select on best.wicketkeeperanalysis to KXIPcoach;  
grant select on best.allrounder to MIcoach;  
grant select on best.allrounderanalysis to MIcoach;  
grant select on best.batsman to MIcoach;  
grant select on best.batsmananalysis to MIcoach;  
grant select on best.bowler to MIcoach;  
grant select on best.bowleranalysis to MIcoach;  
grant select on best.fantasyplatform to MIcoach;  
grant select on best.player to MIcoach;  
grant select on best.pointsystem to MIcoach;  
grant select on best.teams to MIcoach;  
grant select on best.wicketkeeper to MIcoach;  
grant select on best.wicketkeeperanalysis to MIcoach;  
grant select on best.allrounder to RCBcoach;  
grant select on best.allrounderanalysis to RCBcoach;  
grant select on best.batsman to RCBcoach;  
grant select on best.batsmananalysis to RCBcoach;  
grant select on best.bowler to RCBcoach;  
grant select on best.bowleranalysis to RCBcoach;  
grant select on best.fantasyplatform to RCBcoach;  
grant select on best.player to RCBcoach;  
grant select on best.pointsystem to RCBcoach;  
grant select on best.teams to RCBcoach;  
grant select on best.wicketkeeper to RCBcoach;  
grant select on best.wicketkeeperanalysis to RCBcoach;  
grant select on best.allrounder to SRHcoach;  
grant select on best.allrounderanalysis to SRHcoach;

---

```
grant select on best.batsman to SRHcoach;
grant select on best.batsmananalysis to SRHcoach;
grant select on best.bowler to SRHcoach;
grant select on best.bowleranalysis to SRHcoach;
grant select on best.fantasyplatform to SRHcoach;
grant select on best.player to SRHcoach;
grant select on best.pointsystem to SRHcoach;
grant select on best.teams to SRHcoach;
grant select on best.wicketkeeper to SRHcoach;
grant select on best.wicketkeeperanalysis to SRHcoach;
grant select on best.allrounder to RRcoach;
grant select on best.allrounderanalysis to RRcoach;
grant select on best.batsman to RRcoach;
grant select on best.batsmananalysis to RRcoach;
grant select on best.bowler to RRcoach;
grant select on best.bowleranalysis to RRcoach;
grant select on best.fantasyplatform to RRcoach;
grant select on best.player to RRcoach;
```

## Cricket Analyst

Cricket Analyst can analyse the performance of all the cricketers and hence the team they belong to.

```
grant select on best.allrounder to CricketAnalyst;
grant select on best.allrounderanalysis to CricketAnalyst;
grant select on best.batsman to CricketAnalyst;
grant select on best.batsmananalysis to CricketAnalyst;
grant select on best.bowler to CricketAnalyst;
grant select on best.bowleranalysis to CricketAnalyst;
grant select on best.fantasyplatform to CricketAnalyst;
grant select on best.player to CricketAnalyst;
grant select on best.pointsystem to CricketAnalyst;
grant select on best.teams to CricketAnalyst;
grant select on best.wicketkeeper to CricketAnalyst;
```

---

```
grant select on best.wicketkeeperanalysis to CricketAnalyst;  
grant update on best.allrounderanalysis to CricketAnalyst;  
grant update on best.wicketkeeperanalysis to CricketAnalyst;  
grant update on best.bowleranalysis to CricketAnalyst;  
grant update on best.batsmananalysis to CricketAnalyst;
```

### Administrator

Administrator have the right to update and modify the entire database. He has the overall control on the database.