IPL players ranking system

**Context**:

Find the ranking of a player in IPL.

**Solution:**

Every team play a match and a player in concern is a part of a Team.

By this every player will have an individual scorecard for each match that he plays.

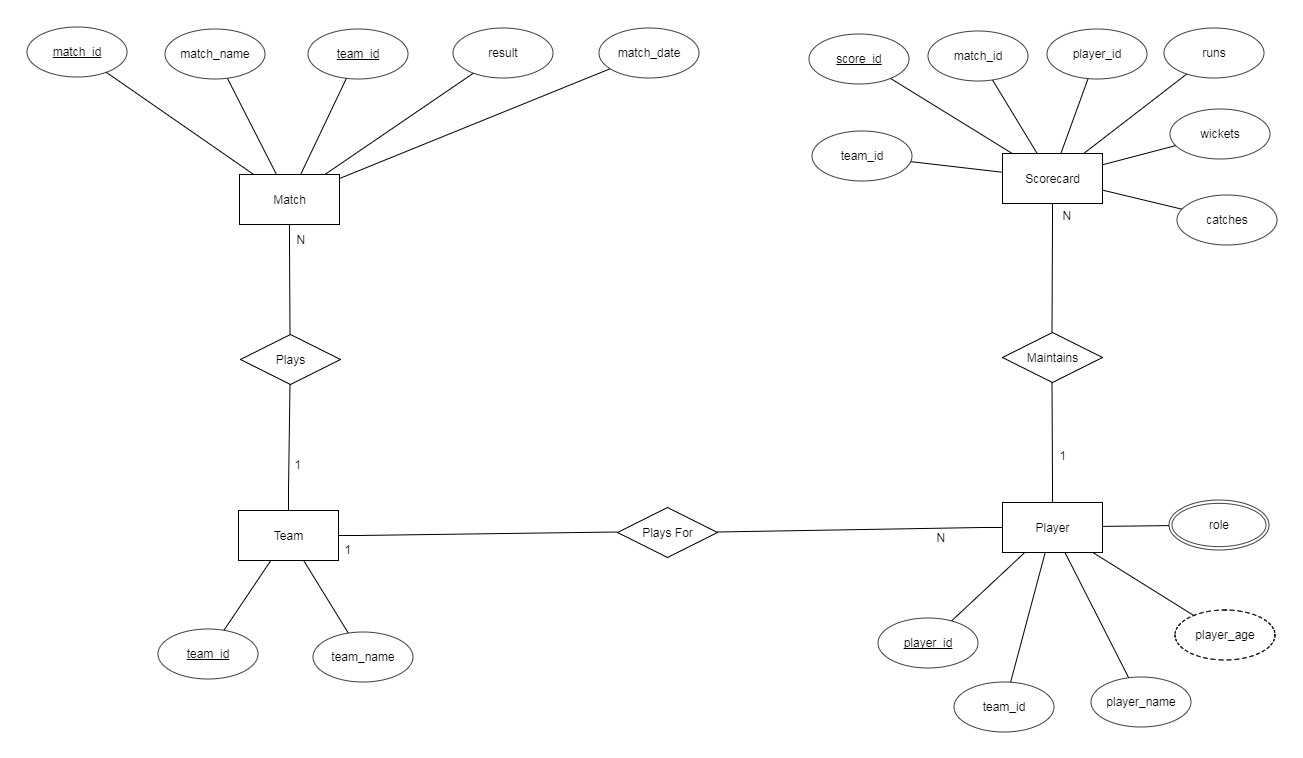
The scorecard depicts the runs, wickets, catches for a given match.

From this I can the get the no.of Matches played and runs or wickets taken by a player.

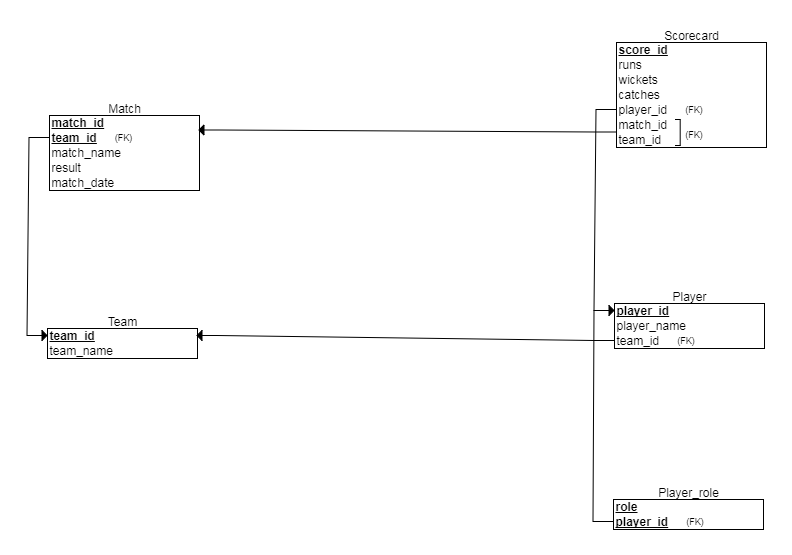
Making it simple, I would just go for the best batting and bowling average and then determine the ranking of a player.

Gradually the other rules can be considered such as most catches taken, most run outs, etc. but these might go as part of the business logic.

**ER-diagram**



**Relational Schema**



ENTITIES

1. **PLAYER**

Attributes -

Simple single-valued attributes - player\_id, player\_name, team\_id

Simple multi-valued attributes – role -> player can be a batsman (ba) , bowler (bo) , wicketkeeper (wi) – (ba && bo or ba && wi).

Derived attibutes - player\_age – derived from date of birth

Primary Key – player\_id

Foreign Key – team\_id

Entity Constraints - player\_id

Domain Constraints - player\_id(int), player\_name(varchar 20),team\_id(int),role (char 2), player\_age(int)

### Referential integrity Constraints - team\_id

1. **SCORECARD**

Attributes -

Simple single-valued attributes - score\_id, match\_id, team\_id, player\_id, runs, wickets, catches

Primary Key – score\_id

Foreign Key – match\_id , team\_id, player\_id

Entity Constraints - score\_id

Domain Constraints - score\_id(int), match\_id(int), player\_id(int), runs(int), wickets(int), catches(int)

Referential integrity Constraints **-** match\_id , player\_id

1. **TEAM**

Attributes -

Simple single-valued attributes - team\_id, team\_name

Primary Key – team\_id

Entity Constraints - team\_id

Domain Constraints - team\_id(int), team\_name(varchar 50)

1. **MATCH**

Attributes -

Simple single-valued attributes - match\_id, match\_name ,team\_id ,result , date

Primary Key – match\_id, team\_id

Foreign Key – team\_id

Entity Constraints - match\_id, team\_id (Together they will decide the unique tuple)

Domain Constraints - match\_id(int), match\_name(int) ,team\_id(int) ,result(char 1) , date

Referential integrity Constraints **-** team\_id

**Data Definition Language (DDL)**

CREATE TABLE Team

(

team\_id INT NOT NULL,

team\_name VARCHAR(50) NOT NULL,

PRIMARY KEY (team\_id)

);

CREATE TABLE Match

(

match\_id INT NOT NULL,

match\_name VARCHAR(50) NOT NULL,

result CHAR(1) NOT NULL,

match\_date DATE NOT NULL,

team\_id INT NOT NULL,

PRIMARY KEY (match\_id),

FOREIGN KEY (team\_id) REFERENCES Team(team\_id)

);

CREATE TABLE Player

(

player\_id INT NOT NULL,

player\_name VARCHAR(20) NOT NULL,

team\_id INT NOT NULL,

PRIMARY KEY (player\_id),

FOREIGN KEY (team\_id) REFERENCES Team(team\_id)

);

CREATE TABLE Scorecard

(

score\_id INT NOT NULL,

runs INT NOT NULL DEFAULT 0,

wickets INT NOT NULL DEFAULT 0,

catches INT NOT NULL DEFAULT 0,

match\_id INT NOT NULL,

player\_id INT NOT NULL,

PRIMARY KEY (score\_id),

FOREIGN KEY (match\_id) REFERENCES Match(match\_id),

FOREIGN KEY (player\_id) REFERENCES Player(player\_id)

);

CREATE TABLE Player\_role

(

role CHAR(2) NOT NULL,

player\_id INT NOT NULL,

PRIMARY KEY (role, player\_id),

FOREIGN KEY (player\_id) REFERENCES Player(player\_id)

);

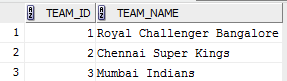
**DATA MANIPULATION LANGUAGE**

1. **TEAM**

Insert into TEAM (TEAM\_ID,TEAM\_NAME) values (1,'Royal Challenger Bangalore');

Insert into TEAM (TEAM\_ID,TEAM\_NAME) values (2,'Chennai Super Kings');

Insert into TEAM (TEAM\_ID,TEAM\_NAME) values (3,'Mumbai Indians');



1. **MATCH**

Insert into MATCH (MATCH\_ID,MATCH\_NAME,RESULT,MATCH\_DATE,TEAM\_ID) values (1,'IPL-RCB-CSK','W',to\_date('26-OCT-17 14:00:00','DD-MON-RR HH24:MI:SS'),2);

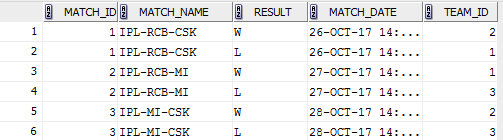
Insert into MATCH (MATCH\_ID,MATCH\_NAME,RESULT,MATCH\_DATE,TEAM\_ID) values (1,'IPL-RCB-CSK','L',to\_date('26-OCT-17 14:00:00','DD-MON-RR HH24:MI:SS'),1);

Insert into MATCH (MATCH\_ID,MATCH\_NAME,RESULT,MATCH\_DATE,TEAM\_ID) values (2,'IPL-RCB-MI','W',to\_date('27-OCT-17 14:00:00','DD-MON-RR HH24:MI:SS'),1);

Insert into MATCH (MATCH\_ID,MATCH\_NAME,RESULT,MATCH\_DATE,TEAM\_ID) values (2,'IPL-RCB-MI','L',to\_date('27-OCT-17 14:00:00','DD-MON-RR HH24:MI:SS'),3);

Insert into MATCH (MATCH\_ID,MATCH\_NAME,RESULT,MATCH\_DATE,TEAM\_ID) values (3,'IPL-MI-CSK','W',to\_date('28-OCT-17 14:00:00','DD-MON-RR HH24:MI:SS'),2);

Insert into MATCH (MATCH\_ID,MATCH\_NAME,RESULT,MATCH\_DATE,TEAM\_ID) values (3,'IPL-MI-CSK','L',to\_date('28-OCT-17 14:00:00','DD-MON-RR HH24:MI:SS'),3);

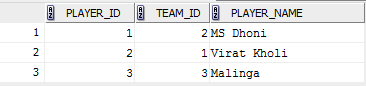


1. **PLAYER**

Insert into LS2USER.PLAYER (PLAYER\_ID,PLAYER\_NAME,TEAM\_ID) values (1,'MS Dhoni',2);

Insert into LS2USER.PLAYER (PLAYER\_ID,PLAYER\_NAME,TEAM\_ID) values (2,'Virat Kholi',1);

Insert into LS2USER.PLAYER (PLAYER\_ID,PLAYER\_NAME,TEAM\_ID) values (3,'Malinga',3);



1. **SCORECARD**

Insert into LS2USER.SCORECARD (SCORE\_ID,RUNS,WICKETS,CATCHES,PLAYER\_ID,MATCH\_ID,TEAM\_ID) values (1,110,0,2,1,1,2);

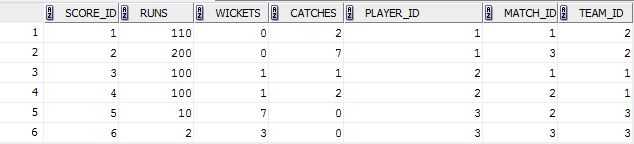
Insert into LS2USER.SCORECARD (SCORE\_ID,RUNS,WICKETS,CATCHES,PLAYER\_ID,MATCH\_ID,TEAM\_ID) values (2,200,0,7,1,3,2);

Insert into LS2USER.SCORECARD (SCORE\_ID,RUNS,WICKETS,CATCHES,PLAYER\_ID,MATCH\_ID,TEAM\_ID) values (3,100,1,1,2,1,1);

Insert into LS2USER.SCORECARD (SCORE\_ID,RUNS,WICKETS,CATCHES,PLAYER\_ID,MATCH\_ID,TEAM\_ID) values (4,100,1,2,2,2,1);

Insert into LS2USER.SCORECARD (SCORE\_ID,RUNS,WICKETS,CATCHES,PLAYER\_ID,MATCH\_ID,TEAM\_ID) values (5,10,7,0,3,2,3);

Insert into LS2USER.SCORECARD (SCORE\_ID,RUNS,WICKETS,CATCHES,PLAYER\_ID,MATCH\_ID,TEAM\_ID) values (6,2,3,0,3,3,3);



1. **PLAYER\_ROLE**

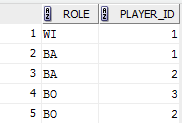
Insert into LS2USER.PLAYER\_ROLE (ROLE,PLAYER\_ID) values ('BA',1);

Insert into LS2USER.PLAYER\_ROLE (ROLE,PLAYER\_ID) values ('BA',2);

Insert into LS2USER.PLAYER\_ROLE (ROLE,PLAYER\_ID) values ('BO',2);

Insert into LS2USER.PLAYER\_ROLE (ROLE,PLAYER\_ID) values ('BO',3);

Insert into LS2USER.PLAYER\_ROLE (ROLE,PLAYER\_ID) values ('WI',1);

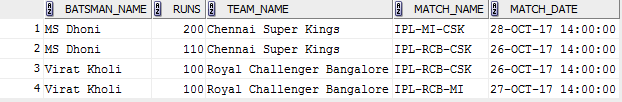


**SQL QUERIES**

**BATSMAN:**

**FETCH THE RECORD FOR ALL THE BATSMAN GROUP BY NAME AND DESCENDING ORDER OF THE RUN SCORED IN EACH MATCH.**

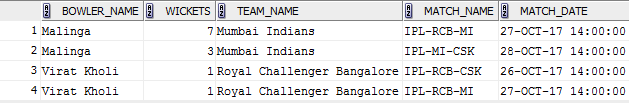
SELECT P.player\_name BATSMAN\_NAME,   
       S.runs,   
       T.team\_name,   
       M.match\_name,   
       M.match\_date   
FROM   scorecard S,   
       player P,   
       team T,   
       match M,   
       player\_role R   
WHERE  S.player\_id = P.player\_id   
       AND S.team\_id = T.team\_id   
       AND S.match\_id = M.match\_id   
       AND P.player\_id = R.player\_id   
       AND R.ROLE IN 'BA'   
GROUP  BY P.player\_name,   
          S.runs,   
          T.team\_name,   
          M.match\_name,   
          M.match\_date   
ORDER  BY S.runs DESC;



**BOWLER:**

**FETCH THE RECORD FOR ALL THE BOWLER GROUP BY NAME AND DESCENDING ORDER OF THE WICKETS TAKEN IN EACH MATCH.**

SELECT P.player\_name BOWLER\_NAME,   
       S.wickets,   
       T.team\_name,   
       M.match\_name,   
       M.match\_date   
FROM   scorecard S,   
       player P,   
       team T,   
       match M,   
       player\_role R   
WHERE  S.player\_id = P.player\_id   
       AND S.team\_id = T.team\_id   
       AND S.match\_id = M.match\_id   
       AND P.player\_id = R.player\_id   
       AND R.ROLE IN 'BO'   
GROUP  BY P.player\_name,   
          S.wickets,   
          T.team\_name,   
          M.match\_name,   
          M.match\_date   
ORDER  BY S.wickets DESC;



**AGGREGATE QUERIES:**

**BATSMAN RANKING**

**Aggregate Query to get the ranking of Batsman from the average run scored by them.**

SELECT P.player\_name,   
       **Avg**(S.runs) AVG\_RUNS,   
       **SUM**(runs)   TOTAL\_RUNS   
FROM   scorecard S,   
       player P,   
       player\_role R   
WHERE  S.player\_id = P.player\_id   
       AND P.player\_id = R.player\_id   
       AND R.ROLE IN 'BA'   
GROUP  BY P.player\_name   
ORDER  BY **Avg**(S.runs) DESC



**BOWLER RANKING**

**Aggregate Query to get the ranking of Bowler from the average wicket taken by them.**

SELECT P.player\_name,   
       **Avg**(S.wickets) AVG\_WICKETS,   
       **SUM**(S.wickets) TOTAL\_WICKETS   
FROM   scorecard S,   
       player P,   
       player\_role R   
WHERE  S.player\_id = P.player\_id   
       AND P.player\_id = R.player\_id   
       AND R.ROLE IN 'BO'   
GROUP  BY P.player\_name   
ORDER  BY **Avg**(S.wickets) DESC



**VARIATIONS:**

1. **Finding the overall best player in the IPL**
2. **Checking the number of catches taken by players , runouts, stumping ,etc**
3. **Bating rate and Bowler’s economy rate can be taken into consideration.**

**SIDE EFFECTS:**

**The current MATCH table repeats the entry for each player for columns**

**match\_id and match\_name.**

**May be it can be normalized to two tables – MATCH and MATCH\_RESULT**

**MATCH – match\_id,match\_name and match\_date**

**MATCH\_RESULT – match\_result,team\_id**