

**Introduction**

The Indian Premier League (IPL) is a professional Twenty20 cricket league played in India. It was founded by the Board of Control for Cricket in India(BCCI) in the year 2008. The two and half months long cricketing event happens every year during the summer. With players coming from all over the world, IPL is the most successful premier franchise cricket in the world right now. IPL also holds the record of most viewed sporting event online with 18.6 million concurrent viewers.

There are eight teams representing eight different regions in India. Each team has a squad of 15-25 players. Out of which 11 players are on field on the match day. To keep it fair, only 4 international players are allowed to play a game from each team. All the teams are either owned by businessmen or movie stars or politicians. With one of the teams owned by richest man in Asia, Mr. Mukesh Ambani of Reliance India Limited. With so much glamour and money in the sports, each team is supported by best in class support staff, analysts and media coverage.

We will be focusing on the organisational aspect of the tournament considering the number of teams playing, stadiums, players, matches, records and finance. For the sake of the project, we are focusing only on the tournament which was held in 2019. Hence, and all our data is derived from the 2019 edition of the IPL. We have created this dataset on our own with constant help for records from the official website of the Indian Premier League ([www.iplt20.com](http://www.iplt20.com)), CricBuzz.com and Cricbuzz.com. We have focused on every possible detail about this organisation in order to bring out the deep analysis about this famous tournament.

**Entities And Attributes:**

We have a total of 8 entities in our dataset each of which are described below.

1. **TEAMS**

****

As mentioned earlier 8 teams compete in this tournament. This entity will capture all the information of the team like the name, location (region they represent), matches\_played,etc.

A short description of each attribute is added below for better understanding of the dataset.

* **Team\_ID**: Each Team has a unique ID.
* **Stadium\_Name**: Each team has its own home stadium ground. **Eg.** Eden Gardens
* **Team\_abbreviation:** Each team has a short form **Eg.** CSK is abbreviation for the team Chennai Super Kings.
* **Team\_Name:** Each team has a team name. **Eg.** Chennai Super Kings.
* **Base\_location:** Location city for each of these teams. **Eg.** Chennai.
* **Team\_Color:** Jersey color.
* **Fans\_Locations\_percentage:** Fan following of each team by their city.
* **Matches\_played:** Number of matches played by each team throughout the tournament.
* **Matches\_Won:** Total number of matches won by each team.
* **Ranking:** Final standings of each team by the end of the season.

1. **PLAYERS**

****

This entity will capture information about each player like his name, team, age,etc.

**Attributes:**

* **Player\_Name:** Player’s full name. **Eg.** Virat Kohli
* **Team\_ID :** Each team’s ID
* **Player\_Matches\_played:** Number of matches played by each player.
* **Age:** Age of players.
* **Player\_Type:** A player can be a batsmen, bowler or an all-rounder and each player has details about the runs they score, wickets taken, their strike rate and their economy.
  + **Batsmen:** This table captures information about the runs and strike rate of a particular batsmen.
  + **Bowler:** This table captures information about the economy and wickets taken by a particular bowler.
  + **All- rounder:** A cricketer who is a batsmen and a bowler is classified as an all rounder. Hence, this table captures information about runs, strike rate, economy and wickets taken by a cricketer.

1. **MATCHES**

****

This entity will capture all the information about the matches like where they were played, by which team, the date, time, winner, etc.

**Attributes:**

* **Match\_ID:** Each match has its unique ID number.
* **Stadium\_Name:** Each match is played in a stadium.
* **Match\_Type:** The match can be league, knockouts or Finals.
* **Teams\_played:** Team names playing against each other for a particular match.
* **Winner:** The winning team for a particular match.
* **Margin:** Won by how many runs or wickets.
* **Match\_Date:** The date on which a particular match was played.
* **Match\_Time:** Afternoon/ Evening match.

1. **STADIUM**

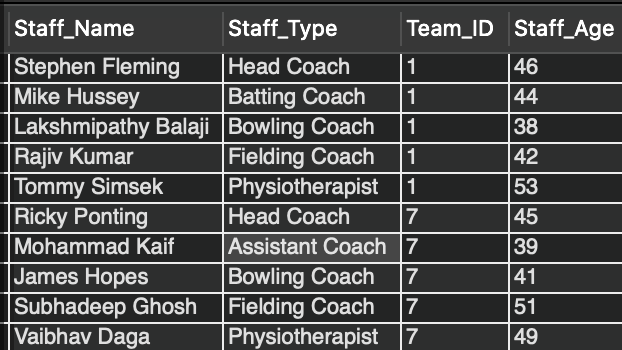
****

This entity will capture details of the stadium where the match was played like name, city,etc.

**Attributes:**

* **Stadium\_Name:** Name of the stadium.
* **Stadium\_City:** In which city is that particular stadium located.
* **Stadium\_Matches\_Played:** Number of matches played in a particular stadium.
* **Capacity:** The number of people a particular stadium can hold.

1. **SUPPORT\_STAFF**

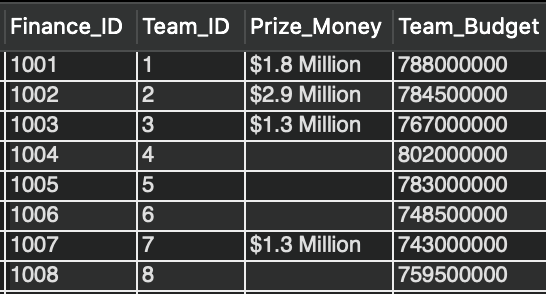
****

This entity will capture the information about the details of staff members.

**Attributes:**

* **Staff\_Name:** Name of different coaches.
* **Team\_ID:** ID of the team.
* **Staff\_Type:** Batting coach, bowling coach, fielding coach, etc.
* **Staff\_Age:** Age of these coaches.

1. **FINANCE**

****

This entity will capture information related to financial records as in the team budget, the price money.

**Attributes:**

* **Finance\_ID –** Unique Id to store the team revenues.
* **Team\_ID –** Team Id numbers
* **Prize\_Money –** Prize money of each team by the end of the season.
* **Team\_Budget –** Total spending on every player for this season for each team

1. **AUCTION**

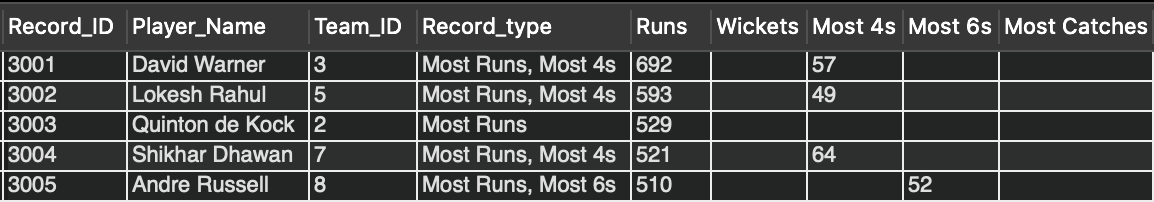
****

This entity will capture the details of the auctions held like base price, final price, players, teams, etc.

**Attributes:**

* **Auction\_ID –** Unique Id for a particular player’s auction.
* **Player\_Name –** Name of the player participating in the auction.
* **Team\_ID –** Team Id
* **Base\_Price –** Starting price of a particular player.
* **Final\_Price –** Price at which a player was sold to a team.
* **Player\_category –** Capped, Uncapped or International.
* **Auction\_Status –** Sold, Retained, Traded or Transferred

1. **RECORDS**

****

This entity will capture the details of records made by players like most number of runs, wickets, fours, sixes etc.

**Attributes**

* **Record\_ID** – Unique Id to store every player’s record(s) throughout the tournament.
* **Player\_Name** – Every player’s name.
* **Team\_ID** – Team Id
* **Record\_Type** – Each player can have more than one record.
* **Most\_Runs** – Most runs by a player
* **Most\_Wickets** – Most wickets by a player
* **Most\_Fours** – Most fours by a player
* **Most\_Sixes** – Most sixes by a player
* **Most\_Catches** – Most catches by a player

**BUSINESS RULES**

**The business rules for our organisation are mentioned below:**

* Each team must have no more than one stadium.
* Each stadium must have no more than one home team.
* Each team must have at least three staff for support.
* Each staff member must be assigned to one team.
* Each team must have no more than one finance record.
* Each finance record must be assigned to one team.
* Each team must have many players to play in a match.
* Each player may be in a team to play the tournament.
* Each stadium hosts at least one match in the tournament.
* Each match is played at least once in every stadium.
* Each match is played by many players from a team.
* Each player from a team plays at least one match in the whole tournament.
* Each player is auctioned not more than once.
* Each auction must host no more than one player at a time.
* Each player can have no more than one record during a season.
* Each record can be maintained for a particular player during a season.

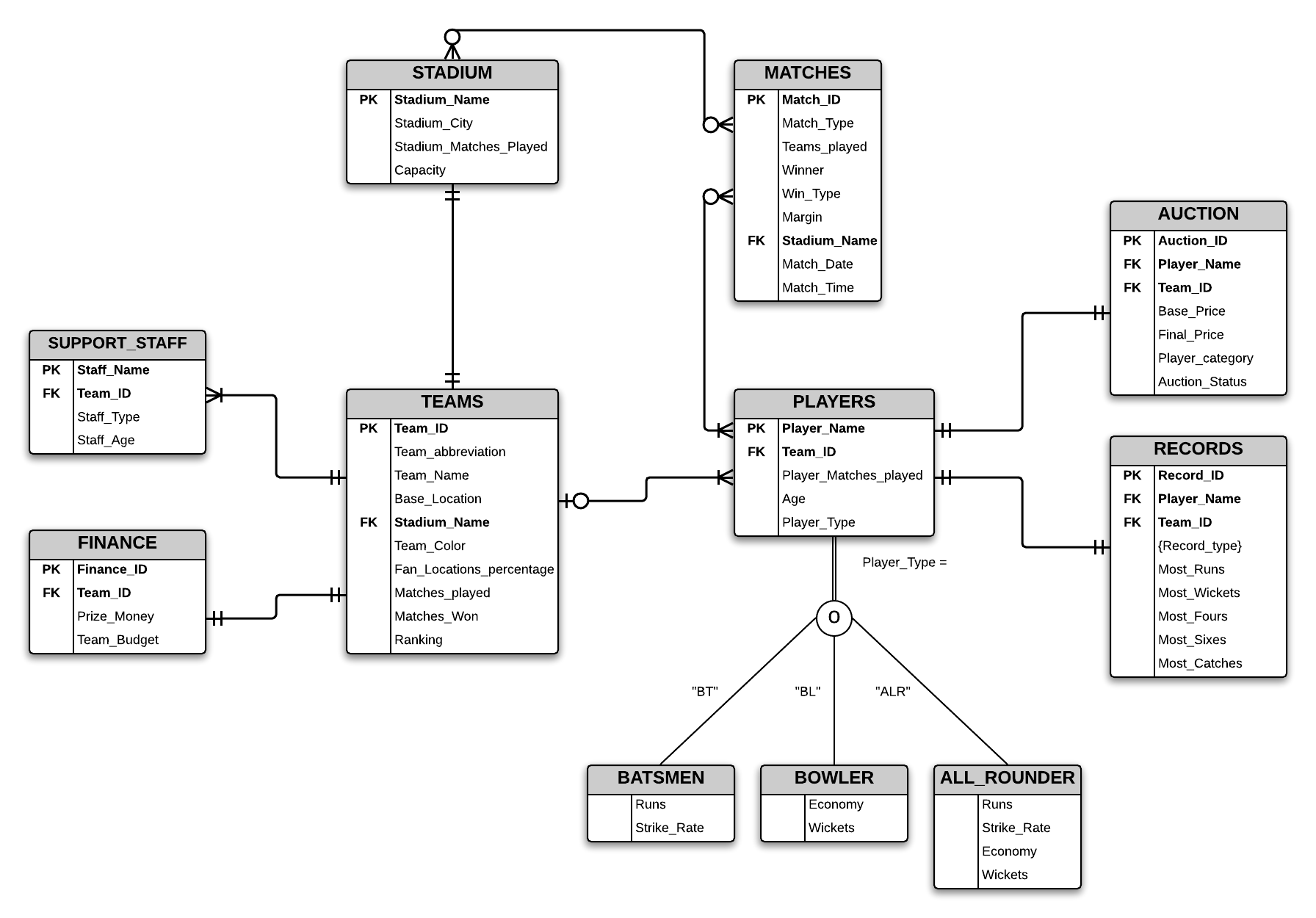
**EER DIAGRAM**

The EER Diagram represents relationships among all the tables in our database.

The diagram provides a visual starting point for database design that will also help in determining the information system requirements throughout an organisation.

Also, once the relational database is rolled out, an ERD will still serve as a reference point, should any debugging or business process re-engineering be needed later on.

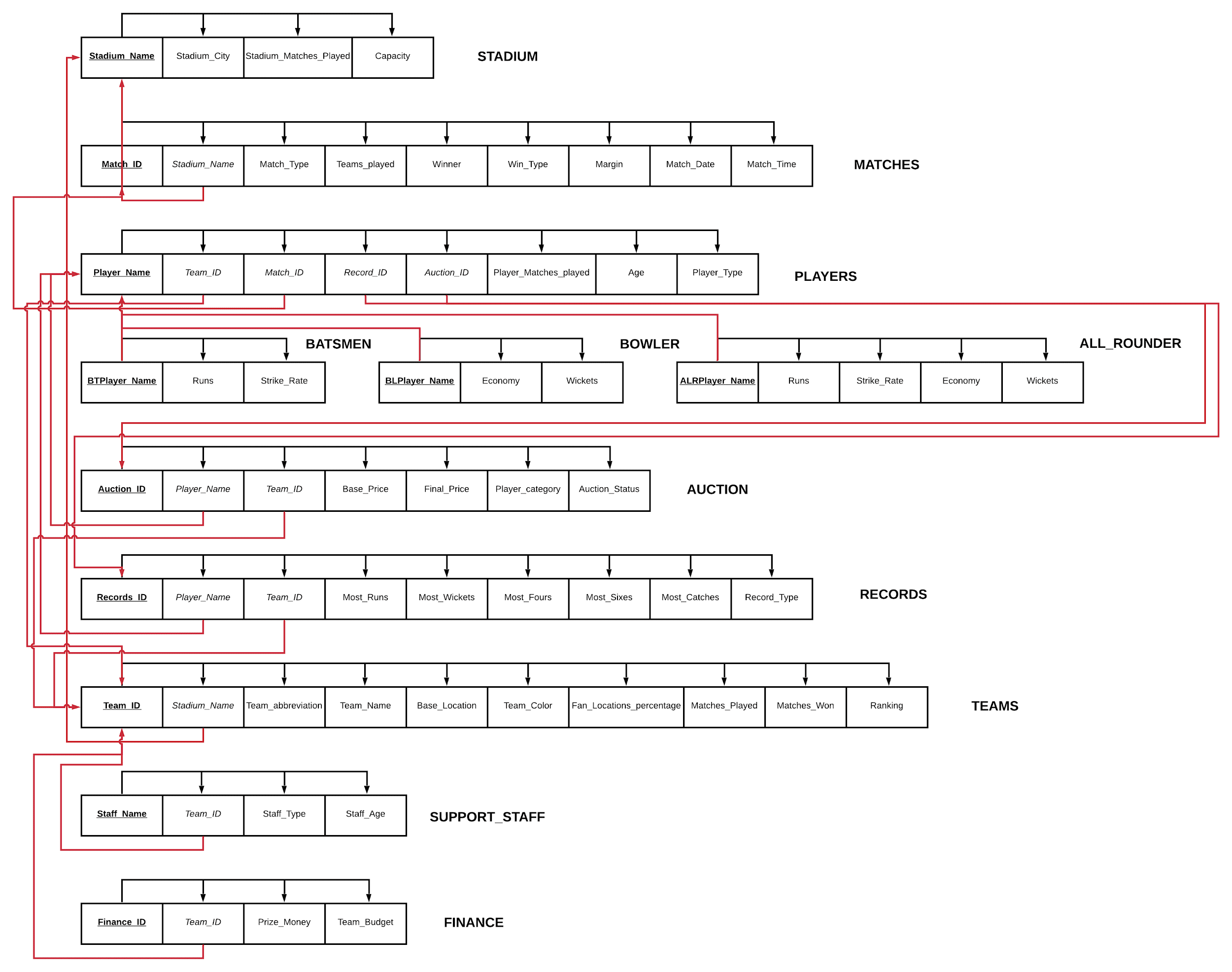
The EER Diagram according to business rules is as below:

****

One important thing to observe is that there is no specific business process here. There is no flow for this organisation. And this project is about finding details of specific data that we want to generate for this tournament. As seen in the above diagram we have 8 entities and one-to-one, many-to-one, many-to-many relationships. The entity **players** we have applied total specialisation rules as a player has to be either a batsmen, a bowler or an all rounder. Also, there is the presence of overlap along with total specialisation rule which means that A player can be a batsman ,a bowler or an all-rounder but he must be one or the other due to the presence of total specialization rule.

**A RELATIONAL MODEL**

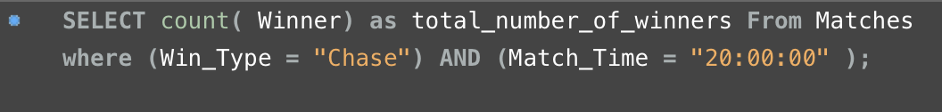
We did not have any partial or transitive dependencies hence our diagram was already in 3NF. The **third normal form (3NF)** is a database schema which uses normalizing principles to reduce the duplication of data, avoid data anomalies and simplify data management. A database relation is said to be in 3NF if all the attributes are functionally dependent on solely the primary key which can be seen in the diagram below.



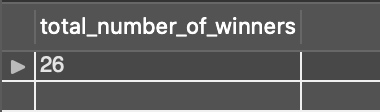
CREATING & LOADING TABLES:

**QUERIES & APPROACH:**

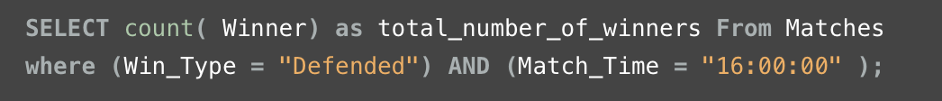
1. We want to know how many teams have won a cricket game while chasing an evening game. For this we are operating on the “Matches” table. To find the result, we will first use COUNT function on Winner attribute and then apply WHERE on Win\_Type and Match\_Time to filter out teams which won by chasing and matches which started at 20:00(evening matches).



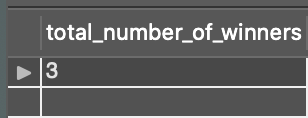
This gave us an output result of 26. Which means 26 teams won while chasing in an evening match.



2. Similar to the previous problem, for the second query, we want to find out how many teams won an afternoon game while defending. For this we will again operate on the Matches table. Use the WHERE filter on Win\_Type and Match\_Time to filter out Defended and 16:00 time respectively. With a count on Winner then will give us the number of winners in this scenario.



This gave us an output of 3. Which means only 3 teams won in this scenario.



3. For the third query, we aimed at finding the age of the youngest support staff of a team. So we operated on the Support\_staff table. SELECT the attributes for desired results and then applied ORDER BY on staff\_Age attribute in ascending order and filter out only the top entry.

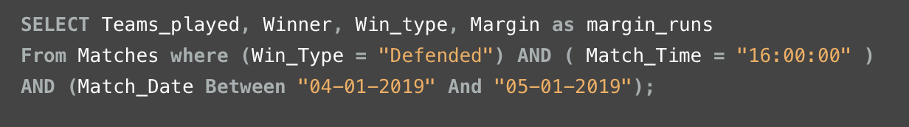


For this, the output coming out it DIshant Yagnik, who is aged 32.

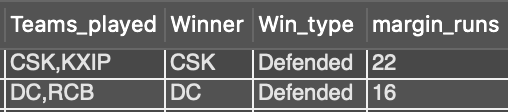


4. We all know that it is hard to play on an open field in a summer afternoon, that too in India. So for this problem we found out how many teams won an afternoon game batting first in the month of April. Which is a sunny month in India.

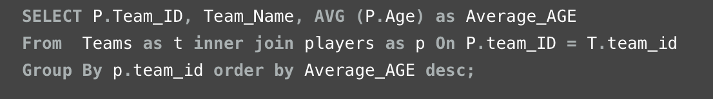
In order to find the result, we select all the required attributes on the Matches table, then we use WHERE to filter out Defended as win type and 16:00 as match time. At last we applied the time selection filter BETWEEN on the match date to select only April month.



We can see that only two teams won a match defending in an April afternoon. These were Chennai Super Kings and Delhi Capitals. Their winning margins were 22 & 16 respectively.



5. For the fifth query, we wanted to know the average age of the players for each team. This will let us know about what aged players are the teams targeting. For this, we SELECT required attributes. Then apply inner join on tables Teams and Players. Group by team ID and at last apply ORDER BY on Average\_Age which we found out by applying AVG command on Age attribute while writing down the SELECT part.



As a conclusion, we can see that Chennai Super Kings had a team of senior players with an average age of 30.920 and the Delhi Capitals had the youngest team with an average age of 26.79. It is also amusing to know that these are the only two teams which won an afternoon game by defending in the month of April (as seen on the previous query).



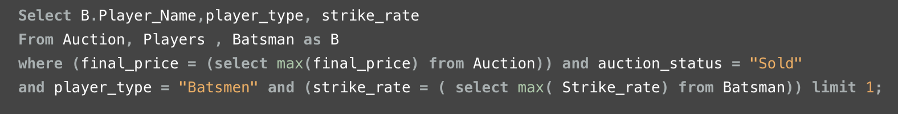
6. As a part of our sixth query, we wanted to find out which team has the maximum number of seating arrangements i.e. the capacity of the stadium. For this, we have used SELECT to get important attributes and then applied ORDER BY on Capacity in descending order.



This gave us list of all the stadiums in a descending order by it’s capacity of all the stadiums which were used in Indian Premier League 2019.



7. For this query, we wanted to know the name of the batsmen who was sold for the highest value in the auction and had the highest strike rate. We SELECT the required attributes from multiple entities. We put a condition by using WHERE to draw out that player name who’s final price was the maximum in the Batsmen category and also had the maximum strike rate.

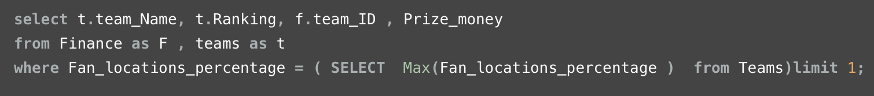


We found out that Rishabh Pant was the batsmen who was sold at the highest value in the auction who had the highest strike rate of 163 in the whole tournament.

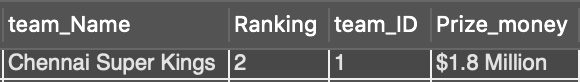


8. A one of the complicated query, we here found out a team which has most fans across India with their season rank and prize money which they won in the year 2019.

Here, we applied a joint on Finance and Teams tables and then selected the team which has maximum fans using WHERE command. Then as a part of subquery, we selected to display it’s season ranking and prize money which they won from the Finance table.

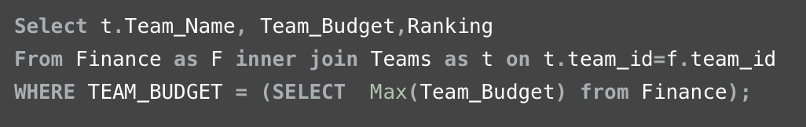


As a result, Chennai Super Kings comes out to be the team which has 33% of fans, had a season rank of 2 and won $1.8 Million as prize money.



9. For our last query, we wanted to find out the team which spent the most amount of money on their team and what their season ranking.

Here, we have used inner join on Finance and Teams tables. Used WHERE to find the team with maximum team budget and then printed the rank of the team.



As a result we can see that Royal Challengers Bangalore has spent the maximum money for team building yet their season ranking is 8th. Which means they ended up on the bottom of the table.



**CONCLUSION**

We had an amazing experience both in terms of learning and structuring the insights of this popular tournament. It is a sport that always keeps us on our feet to enjoy and it is an entertainment sport that is upcoming. We had a very challenging experience building the dataset from the scratch and gathering every possible detail about this tournament. Gathering minor details from every possible source available was a big task we chose to take but at the same time it was exciting since it is a sport we all love and that interest got us to focus on taking this project. Creating the database was another challenge since this was something we built from the start and getting the database right was important. Changing the structure of the database multiple times to make the relationships feel right was something that took us a lot of time but our interest in this organisation got us to figure out every single complexity. We had to create relational models to understand the details about the database model we created and that got us our solution to the problems. Overall this experience enabled us to get an in depth knowledge of this organisation and a much more insight of how it works every year with these complexities over them. Along with this we definitely got a better understanding of how database models enable people to get a better conceptual knowledge of a particular organisation. After creating models and relationships we feel how easy it is for people to understand the insights of any organisation. We also realised during this project how important it is to structure the model because even a slight arrangement might hamper an individual’s understanding of the model.

