A Database Management Project Report on

INDIAN PREMIER LEAGUE

PERFORMED BY:
Mukul Pathak
Shalin Shanghavi
Disha Shah &
Madhu Kaushik

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), 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

Team_ID	Team_Name	Team_abbreviation	Base_Location	Team_Color	Fan_Locations_percenta...	Matches_played	Matches_Won	Ranking
1	Chennai Super Kings	CSK	Chennai	Yellow	33	17	10	2
2	Mumbai Indians	MI	Mumbai	Blue and Gold	19	16	11	1
3	Sunrisers Hyderabad	SRH	Hyderabad	Black and Orange	10	15	6	4
4	Royal Challengers Bangalore	RCB	Bangalore	Black and Red	14	14	5	8
5	Kings XI Punjab	KXIP	Mohali	Red and Light Gray	4	14	5	6
6	Rajasthan Royals	RR	Jaipur	Dark Blue and Gold	3	14	5	7
7	Delhi Capitals	DC	Delhi	Violet and Red	6	16	10	3
8	Kolkata Knight Riders	KKR	Kolkata	Purple and Gold	11	14	6	5

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.

2. PLAYERS

Player_Name	Team_ID	Player_Matches_played	Age	Player_Type
Mohit Sharma	1	1	31	Bowler
Ruturaj Gaikwad	1	0	23	Batsmen
MS Dhoni	1	15	38	Batsmen
Suresh Raina	1	17	33	Batsmen
Deepak Chahar	1	17	27	Bowler
KM Asif	1	0	26	Bowler
Karn Sharma	1	1	32	Bowler
Dhruv Shorey	1	1	27	Batsmen
Faf du Plessis	1	12	35	Batsmen
M Vijay	1	2	35	Batsmen
Ravindra Jadeja	1	16	31	All-rounder
Sam Billings	1	0	28	Batsmen
Mitchell Santner	1	4	28	All-rounder
David Willey	1	0	30	Bowler
Dwayne Bravo	1	12	36	All-rounder
Shane Watson	1	17	38	All-rounder

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.

3. MATCHES

Match_ID	Match_Type	Teams_played	Winner	Win_Type	Margin	Stadium_Name	Match_Date	Match_Time
1	League	CSK,RCB	CSK	Chase	7	MA Chidambaram	03-23-2019	20:00:00
2	League	KKR,SRH	KKR	Chase	6	Eden Gardens	03-24-2019	16:00:00
3	League	MI,DC	DC	Defended	37	Wankhede	03-24-2019	20:00:00
4	League	RR,KXIP	KXIP	Defended	14	Sawai Mansingh	03-25-2019	20:00:00
5	League	DC,CSK	CSK	Chase	6	Feroz Shah Kotla	03-26-2019	20:00:00
6	League	KKR,KXIP	KKR	Defended	28	Eden Gardens	03-27-2019	20:00:00
7	League	RCB,MI	MI	Defended	6	M Chinaswamy	03-28-2019	20:00:00
8	League	SRH,RR	SRH	Chase	5	R Gandhi International	03-29-2019	20:00:00
9	League	KXIP,MI	KXIP	Chase	8	PCA	03-30-2019	16:00:00

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.

4. STADIUM

Stadium_City	Stadium_Name	Stadium_Matches_Played	Capacity
Bangalore	M Chinnaswamy	7	35,000
Chennai	MA Chidambaram	8	39,000
Mumbai	Wankhede	7	33,000
Hyderabad	R Gandhi International	8	55,000
Delhi	Feroz Shah Kotla	7	41,000
Mohali	PCA	7	26,000
Kolkata	Eden Gardens	7	68,000
Jaipur	Sawai Mansingh	7	25,000
Vishakhapatnam	ACA-VDCA	2	27,500

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.

5. SUPPORT_STAFF

Staff_Name	Staff_Type	Team_ID	Staff_Age
Stephen Fleming	Head Coach	1	46
Mike Hussey	Batting Coach	1	44
Lakshmipathy Balaji	Bowling Coach	1	38
Rajiv Kumar	Fielding Coach	1	42
Tommy Simsek	Physiotherapist	1	53
Ricky Ponting	Head Coach	7	45
Mohammad Kaif	Assistant Coach	7	39
James Hopes	Bowling Coach	7	41
Subhadeep Ghosh	Fielding Coach	7	51
Vaibhav Daga	Physiotherapist	7	49

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.

6. FINANCE

Finance_ID	Team_ID	Prize_Money	Team_Budget
1001	1	$1.8 Million	788000000
1002	2	$2.9 Million	784500000
1003	3	$1.3 Million	767000000
1004	4		802000000
1005	5		783000000
1006	6		748500000
1007	7	$1.3 Million	743000000
1008	8		759500000

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

7. AUCTION

Auction_ID	Player_Name	Base_Price	Final_price	Team_ID	Player_category	Auction_Status
2135	Jaydev Unadkat	15,000,000	84000000.00	6	Capped	Sold
2062	Varun Chakaravarthy	20,00,000	84000000.00	5	Uncapped	Sold
2063	Sam Cu6an	2,000,000	72000000.00	5	International	Sold
2045	Colin Ingram	2,000,000	64000000.00	7	International	Sold
2069	Prabhsimran Singh	20,00,000	48000000.00	5	Uncapped	Sold
2061	Nicholas Pooran	75,00,000	42000000.00	5	International	Sold
2064	Mohammad Shami	1,000,000	42000000.00	5	Capped	Sold
2158	Shimron Hetmyer	50,00,000	42000000.00	4	International	Sold
2185	Akshdeep Nath	20,00,000	36000000.00	4	Uncapped	Sold
2114	Barinder Sran	50,00,000	34000000.00	2	Capped	Sold

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

8. RECORDS

Record_ID	Player_Name	Team_ID	Record_type	Runs	Wickets	Most 4s	Most 6s	Most Catches
3001	David Warner	3	Most Runs, Most 4s	692		57		
3002	Lokesh Rahul	5	Most Runs, Most 4s	593		49		
3003	Quinton de Kock	2	Most Runs	529				
3004	Shikhar Dhawan	7	Most Runs, Most 4s	521		64		
3005	Andre Russell	8	Most Runs, Most 6s	510			52	

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.

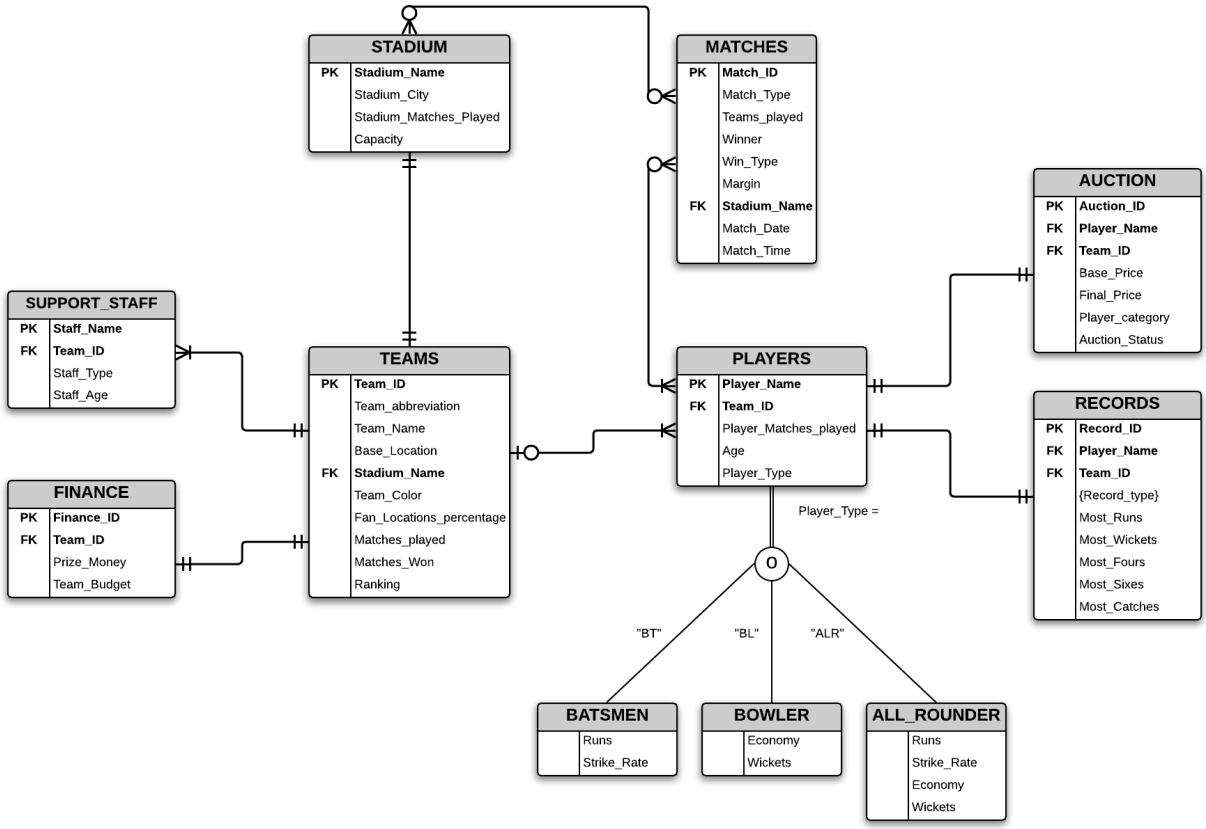
EER DIAGRAM

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.

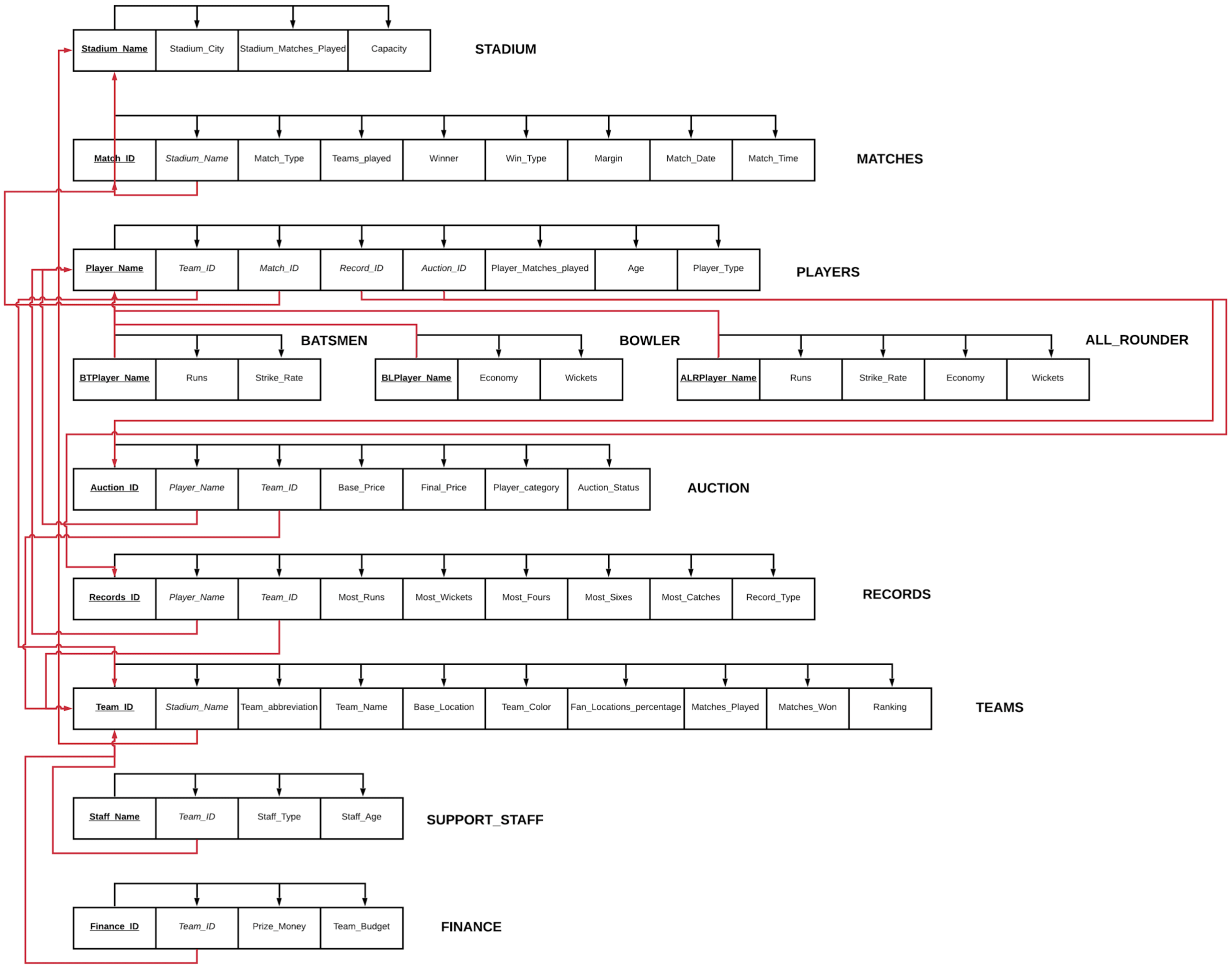
The EER Diagram according to business rules is as below:



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.



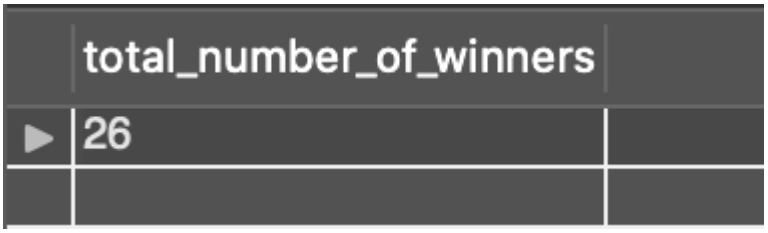
CREATING & LOADING TABLES:

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).

```
SELECT count( Winner) as total_number_of_winners From Matches
where (Win_Type = "Chase") AND (Match_Time = "20:00:00" );
```

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



total_number_of_winners
26

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.

```
SELECT count( Winner) as total_number_of_winners From Matches
where (Win_Type = "Defended") AND (Match_Time = "16:00:00" );
```

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

total_number_of_winners
3

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.

```
SELECT Staff_Name, Team_ID, staff_age From Support_staff Order by Staff_Age ASC limit 1;
```

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

	Staff_Name	Team_ID	Staff_age
▶	Dishant Yagnik	6	36

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.

```
SELECT Teams_played, Winner, Win_type, Margin as margin_runs
From Matches where (Win_Type = "Defended") AND ( Match_Time = "16:00:00" )
AND (Match_Date Between "04-01-2019" And "05-01-2019");
```

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.

Teams_played	Winner	Win_type	margin_runs
CSK,KXIP	CSK	Defended	22
DC,RCB	DC	Defended	16

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.

```
SELECT P.Team_ID, Team_Name, AVG (P.Age) as Average_AGE
From  Teams as t inner join players as p On P.team_ID = T.team_id
Group By p.team_id order by Average_AGE desc;
```

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).

Team_ID	Team_Name	Average_AGE
1	Chennai Super Kings	30.9200
3	Sunrisers Hyderabad	29.0000
4	Royal Challengers Bangalore	28.6000
2	Mumbai Indians	27.5833
8	Kolkata Knight Riders	27.5714
6	Rajasthan Royals	26.8800
5	Kings XI Punjab	26.8636
7	Delhi Capitals	26.7917

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.

```
SELECT Stadium_Name, Capacity, Stadium_City From Stadium order by Capacity Desc;
```

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.

	Stadium_Name	Capacity	Stadium_City
▶	Eden Gardens	68,000	Kolkata
	R Gandhi International	55,000	Hyderabad
	Feroz Shah Kotla	41,000	Delhi
	MA Chidambaram	39,000	Chennai
	M Chinnaswamy	35,000	Bangalore
	Wankhede	33,000	Mumbai
	ACA-VDCA	27,500	Vishakhapatnam
	PCA	26,000	Mohali
	Sawai Mansingh	25,000	Jaipur

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.

```
Select B.Player_Name,player_type, strike_rate
From Auction, Players , Batsman as B
where (final_price = (select max(final_price) from Auction)) and auction_status = "Sold"
and player_type = "Batsmen" and (strike_rate = ( select max( Strike_rate) from Batsman)) limit 1;
```

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.

	Player_Name	player_type	strike_rate
▶	Rishabh Pant	Batsmen	163

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.

```
select t.team_Name, t.Ranking, f.team_ID , Prize_money
from Finance as F , teams as t
where Fan_locations_percentage = ( SELECT  Max(Fan_locations_percentage )  from Teams)limit 1;
```

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.

team_Name	Ranking	team_ID	Prize_money
Chennai Super Kings	2	1	$1.8 Million

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.

```
Select t.Team_Name, Team_Budget,Ranking
From Finance as F inner join Teams as t on t.team_id=f.team_id
WHERE TEAM_BUDGET = (SELECT  Max(Team_Budget) from Finance);
```

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.

Team_Name	Team_Budget	Ranking
Royal Challengers Bangalore	802000000	8

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.

```
CREATE TABLE `all rounder` (
  `Player_Name` text,
  `Strike_Rate` double DEFAULT NULL,
  `Runs` int DEFAULT NULL,
  `Economy` text,
  `Wickets` text
)
```

```
INSERT INTO `all rounder`
VALUES ('Ravindra Jadeja',120.45,106,'6.35','15'),
('Mitchell Santner',139.13,32,'6.71','4'),
('Dwayne Bravo',121.21,80,'8.01','11'),
('Shane Watson',127.56,398,'',''),
('Kedar Jadhav',95.85,162,'',''),
('Colin Munro',120,84,'',''),
('Chris Morris',86.48,32,'9.27','13'),
('Axar Patel',125,110,'7.13','10'),
('Sherfane Rutherford',135.18,73,'8.63','1'),
('Keemo Paul',75,18,'8.72','9'),
('Ravichandran Ashwin',150,42,'7.27','15'),
('Sam Curran',172.72,95,'9.78','10'),
('Harpreet Brar',166.66,20,'9.6','0'),
('Andre Russell',204.81,510,'9.51','11'),
('Piyush Chawla',113.51,42,'8.96','10'),
('Carlos Brathwaite',110,11,'9.66','0'),
('Hardik Pandya',191.42,402,'9.17','14'),
('Krunal Pandya',122,183,'7.28','12'),
('Kieron Pollard',156.74,279,'','');
```

```sql
CREATE TABLE `auction` (
  `Auction_ID` int DEFAULT NULL,
  `Player_Name` text,
  `Base_Price` text,
  `Final_price` text,
  `Team_ID` int DEFAULT NULL,
  `Player_category(capped/uncapped)` text,
  `Auction_Status` text
)
```

```sql
INSERT INTO `auction`
VALUES (2001,'Mohit Sharma ','50,00,000','5,000,000',1,'Capped','Sold'),
(2002,'Ruturaj Gaikwad','20,00,000','20,00,000',1,'Uncapped','Sold'),
(2003,'MS Dhoni','','',1,'Capped','Retained'),
(2004,'Suresh Raina','','',1,'Capped','Retained'),
(2005,'Deepak Chahar','','',1,'Capped','Retained'),
(2006,'KM Asif','','',1,'Uncapped','Retained'),
(2007,'Karn Sharma','','',1,'Capped','Retained'),
(2008,'Dhruv Shorey','','',1,'Uncapped','Retained');
```

```
INSERT INTO `batsman`
VALUES ('Ruturaj Gaikwad','',''),
('Suresh Raina','121.97','383'),
('Dhruv Shorey','62.5','5'),
('Faf du Plessis','123.36','396'),
('M Vijay','104.91','64'),
('Ambati Rayudu','93.06','282'),
('Shreyas Iyer','119.94','463'),
('Prithvi Shaw','133.71','353'),
('Manjot Kalra','',''),
('Colin Ingram','119.48','184'),
('Shikhar Dhawan','135.67','521');
```

```
CREATE TABLE `batsman` (
  `Player_Name` text,
  `Strike_Rate` text,
  `Runs` text
)
```

```
INSERT INTO `bowler`
VALUES('Mohit Sharma ','9','1'),
('Deepak Chahar','7.47','22'),
('KM Asif','',''),
('Karn Sharma','11.64','1'),
('David Willey','',''),
('Lungi Ngidi','',''),
('Imran Tahir','6.69','26'),
('Harbhajan Singh','7.09','16'),
('Shardul Thakur','9.36','8'),
('Amit Mishra','6.75','11'),
('Avesh Khan','10','0');
```

```
CREATE TABLE `bowler` (
  `Player_Name` text,
  `Economy` text,
  `Wickets` text
)
```

```sql
INSERT INTO `Finance`
VALUES (1001,'1','$1.8 Million','788000000'),
(1002,'2','$2.9 Million','784500000'),
(1003,'3','$1.3 Million','767000000'),
(1004,'4','','802000000'),
(1005,'5','','783000000'),
(1006,'6','','748500000'),
(1007,'7','$1.3 Million','743000000'),
(1008,'8','','759500000');
```

```sql
CREATE TABLE `Finance` (
  `Finance_ID` int NOT NULL,
  `Team_ID` varchar(45) DEFAULT NULL,
  `Prize_Money` varchar(45) DEFAULT NULL,
  `Team_Budget` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`Finance_ID`)
)
```

```sql
INSERT INTO `matches`
VALUES (1,'League','CSK,RCB','CSK','Chase',7,'MA Chidambaram','03-23-2019','20:00:00'),
(2,'League','KKR,SRH','KKR','Chase',6,'Eden Gardens','03-24-2019','16:00:00'),
(3,'League','MI,DC','DC','Defended',37,'Wankhede','03-24-2019','20:00:00'),
(4,'League','RR,KXIP','KXIP','Defended',14,'Sawai Mansingh','03-25-2019','20:00:00'),
(5,'League','DC,CSK','CSK','Chase',6,'Feroz Shah Kotla','03-26-2019','20:00:00'),
(6,'League','KKR,KXIP','KKR','Defended',28,'Eden Gardens','03-27-2019','20:00:00'),
(7,'League','RCB,MI','MI','Defended',6,'M Chinaswamy','03-28-2019','20:00:00'),
(8,'League','SRH,RR','SRH','Chase',5,'R Gandhi International','03-29-2019','20:00:00');
```

```sql
CREATE TABLE `matches` (
  `Match_ID` int DEFAULT NULL,
  `Match_Type` text,
  `Teams_played` text,
  `Winner` text,
  `Win_Type` text,
  `Margin` int DEFAULT NULL,
  `Stadium_Name` text,
  `Match_Date` text,
  `Match_Time` text
);
```

```sql
CREATE TABLE `players` (
  `Player_Name` text,
  `Team_Name` int DEFAULT NULL,
  `Player_Matches_played` int DEFAULT NULL,
  `Age` int DEFAULT NULL,
  `Player_Type` text
);
```

```sql
INSERT INTO `players`
VALUES ('Mohit Sharma ',1,1,31,'Bowler'),
('Ruturaj Gaikwad',1,0,23,'Batsmen'),
('MS Dhoni',1,15,38,'Batsmen'),
('Suresh Raina',1,17,33,'Batsmen'),
('Deepak Chahar',1,17,27,'Bowler'),
('KM Asif',1,0,26,'Bowler'),
('Karn Sharma',1,1,32,'Bowler');
```

```sql
CREATE TABLE `stadium` (
  `Stadium_City` text,
  `Stadium_Name` text,
  `Stadium_Matches_Played` int DEFAULT NULL,
  `Capacity` int
);
```

```sql
INSERT INTO `stadium`
VALUES('Bangalore','M Chinnaswamy ',7,'35,000'),
('Chennai','MA Chidambaram',8,'39,000'),
('Mumbai','Wankhede',7,'33,000),
('Hyderabad','R Gandhi International',8,'55,000'),
('Delhi','Feroz Shah Kotla',7,'41,000'),
('Mohali','PCA',7,'26,000'),
('Kolkata','Eden Gardens',7,'68,000'),
('Jaipur','Sawai Mansingh',7,'25,000'),
('Vishakhapatnam','ACA-VDCA',2,'27,500');
```

```sql
CREATE TABLE `support staff` (
  `Staff_ID` int DEFAULT NULL,
  `Staff_Name` text,
  `Staff_Type` text,
  `Team_Name` text,
  `Staff_Age` int DEFAULT NULL
);
```

```sql
INSERT INTO `support staff`
VALUES (1,'Stephen Fleming','Head Coach','CSK',46),
(2,'Mike Hussey','Batting Coach','CSK',44),
(3,'Lakshmipathy Balaji','Bowling Coach','CSK',38),
(4,'Rajiv Kumar','Fielding Coach','CSK',42),
(5,'Tommy Simsek','Physiotherapist','CSK',53),
(6,'Ricky Ponting','Head Coach','DC',45),
(7,'Mohammad Kaif','Assistant Coach','DC',39),
(8,'James Hopes','Bowling Coach','DC',41);
```

```sql
CREATE TABLE `Teams` (
  `Teams_ID` int NOT NULL,
  `Team_Name` varchar(45) NOT NULL,
  `Team_abbreviation` varchar(45) NOT NULL,
  `Base_Location` varchar(45) NOT NULL,
  `Team_Color` varchar(45) NOT NULL,
  `Fan_Locations_percentage` varchar(45) NOT NULL,
  `Matches_played` varchar(45) NOT NULL,
  `Matches_Won` varchar(45) NOT NULL,
  `Ranking` varchar(45) NOT NULL,
  PRIMARY KEY (`Teams_ID`),
  UNIQUE KEY `Teams_ID_UNIQUE` (`Teams_ID`)
);
```

```sql
INSERT INTO `Teams`
VALUES (1,'Chennai Super Kings','CSK','Chennai','Yellow','33','17','10','2'),
(2,'Mumbai Indians','MI','Mumbai','Blue and Gold','19','16','11','1'),
(3,'Sunrisers Hyderabad','SRH','Hyderabad','Black and Orange','10','15','6','4'),
(4,'Royal Challengers Bangalore','RCB','Bangalore','Black and Red','14','14','5','8'),
(5,'Kings XI Punjab','KXIP','Mohali','Red and Light Gray','4','14','5','6'),
(6,'Rajasthan Royals','RR','Jaipur','Dark Blue and Gold','3','14','5','7'),
(7,'Delhi Capitals','DC','Delhi','Violet and Red','6','16','10','3'),
(8,'Kolkata Knight Riders','KKR','Kolkata','Purple and Gold','11','14','6','5');
```

```sql
CREATE TABLE `records` (
  `Record_ID` int DEFAULT NULL,
  `Player_Name` text,
  `Team_ID` int DEFAULT NULL,
  `Record_type` text,
  `Runs` int DEFAULT NULL,
  `Wickets` text,
  `Most 4s` text,
  `Most 6s` text,
  `Most Catches` text
);
```

```sql
INSERT INTO `records`
VALUES (3001,'David Warner',3,'Most Runs, Most 4s',692,'','57','','',''),
(3002,'Lokesh Rahul',5,'Most Runs, Most 4s',593,'','49','','',''),
(3003,'Quinton de Kock',2,'Most Runs',529,'','','','',''),
(3004,'Shikhar Dhawan',7,'Most Runs, Most 4s',521,'','64','','',''),
(3005,'Andre Russell',8,'Most Runs, Most 6s',510,'','','52','','');
```