

Objective Questions

1. List the different dtypes of columns in table “ball_by_ball” (using information_schema)

- Code:

```
SELECT COLUMN_NAME, DATA_TYPE FROM information_schema.columns
WHERE table_name = 'Ball_by_Ball' AND table_schema = "ipl";
```

- This query fetches the column names and their data types for the table Ball_by_Ball in the IPL database.
- The query uses the information_schema.columns table, which stores metadata about all the columns in the database.
- The WHERE clause ensures that only the columns of the Ball_by_Ball table are returned.
- The table “ball_by_ball” contains the following data types:

	COLUMN_NAME	DATA_TYPE
▶	Ball_Id	int
	Bowler	int
	Innings_No	int
	Match_Id	int
	Non_Striker	int
	Over_Id	Over_Id
	Runs_Scored	int
	Striker	int
	Striker_Batting_Position	int
	Team_Batting	int
	Team_Bowling	int

2. What is the total number of runs scored in 1st season by RCB (bonus: also include the extra runs using the extra runs table)

- Code:

```
WITH extra_run_data AS (
SELECT
Team_Batting AS team_Id,
SUM(e.Extra_Runs) as total_extra
FROM ball_by_ball b
JOIN extra_runs e ON e.Match_Id = b.Match_Id AND e.Innings_No =
b.Innings_No AND e.Over_Id = b.Over_Id AND e.Ball_Id = b.Ball_Id
WHERE Team_Batting = 2
AND b.Match_Id IN(
SELECT distinct Match_Id FROM matches WHERE Season_Id = ( SELECT
MIN(Season_Id) as first_season FROM Matches WHERE Team_1 = 2 OR Team_2
= 2))
),
run_scored_data AS (
SELECT
Team_Batting AS team_Id,
SUM(b.Runs_Scored) AS total_score
FROM ball_by_ball b
JOIN matches m ON m.Match_Id = b.Match_Id
WHERE Team_Batting = 2 AND (Team_1 = 2 OR Team_2 = 2) AND m.Season_Id
= ( SELECT MIN(Season_Id) as first_season FROM Matches WHERE Team_1 = 2
OR Team_2 = 2)
```

```

)
SELECT
(total_score + total_extra) AS total_runs
FROM run_scored_data s
JOIN extra_run_data e ON e.team_Id = s.team_Id;

```

- This query calculates the total number of runs scored by RCB in the first season they played.
- The query uses two subqueries:
 - extra_run_data: It calculates the total extra runs (like wides, no-balls) from the extra_runs table.
 - run_scored_data: It sums up the runs scored by RCB in the first season from the ball_by_ball table.
- The final SELECT sums up both the runs scored and the extra runs to give the total.

Result Grid	
	total_runs
▶	2601

3. How many players were more than the age of 25 during season 2014?

- Code:


```

SELECT COUNT(DISTINCT p.Player_Id) AS Players_Age_Above_25
FROM Player p
JOIN Player_Match pm ON p.Player_Id = pm.Player_Id
JOIN Matches m ON pm.Match_Id = m.Match_Id
JOIN Season s ON m.Season_Id = s.Season_Id
WHERE s.Season_Year = 2014
AND TIMESTAMPDIFF(YEAR, p.DOB, '2014-12-31') > 25;
      
```

Players_Age_Above_25	
	102
▶	

- This query counts the number of distinct players whose age was greater than 25 during the 2014 season.
- The query joins the players table (where player information like birth date is stored) and the matches table (to filter by the 2014 season).
- The age calculation is done by subtracting the year of birth from the current year, and checking if the result is greater than 25.

4. How many matches did RCB win in 2013?

- Code:


```

SELECT COUNT(*) AS Matches_Won
FROM Matches m
JOIN Season s ON m.Season_Id = s.Season_Id
JOIN Team t ON m.Match_Winner = t.Team_Id
WHERE s.Season_Year = 2013
      
```

```

AND t.Team_Name = 'Royal Challengers Bangalore'
AND m.Match_Winner IS NOT NULL;

```

- The query counts the number of matches won by RCB in the 2013 season by filtering matches where RCB was the winner.

A screenshot of a database result grid titled "Result Grid". It contains one row with two columns. The first column is labeled "Matches_Won" and the second column contains the value "9". There are standard grid navigation buttons at the top and bottom of the table.

	Matches_Won
▶	9

5. List the top 10 players according to their strike rate in the last 4 seasons.

- Code:
- ```

WITH Last_4_Seasons AS (
 SELECT Season_Year FROM Season ORDER BY Season_Year DESC LIMIT 4
),
Striker_Rate AS (
 SELECT
 B.Striker,
 ROUND((SUM(B.Runs_Scored) / NULLIF(COUNT(B.Ball_Id), 0)) * 100, 2) AS Strike_Rate
 FROM Ball_by_Ball B
 JOIN Matches M ON B.Match_Id = M.Match_Id
 JOIN Season S ON M.Season_Id = S.Season_Id
 JOIN Last_4_Seasons L4S ON S.Season_Year = L4S.Season_Year
 GROUP BY B.Striker
 HAVING COUNT(B.Ball_Id) > 100
)
SELECT
 RANK() OVER (ORDER BY SR.Strike_Rate DESC) AS Ranking,
 P.Player_Name,
 SR.Strike_Rate
FROM Striker_Rate SR
JOIN Player P ON SR.Striker = P.Player_Id
ORDER BY SR.Strike_Rate DESC
LIMIT 10;

```

- CTE Last\_4\_Seasons: Fetches the last 4 seasons based on the most recent seasons.
- CTE Striker\_Rate:
  - Calculates the strike rate for each striker (batsman) in the last 4 seasons by dividing the total runs scored by the total balls faced, multiplied by 100.
  - Filters to include only players who faced more than 100 balls.
- Main Query:
  - RANK() is used to assign ranks to the players based on their strike rate in descending order.
  - It joins the Player table to get player names and selects the top 10 players with the highest strike rates.

| Result Grid |         |                | Filter Rows: |
|-------------|---------|----------------|--------------|
|             | Ranking | Player_Name    | Strike_Rate  |
| ▶           | 1       | KH Pandya      | 186.61       |
|             | 2       | SN Khan        | 171.84       |
|             | 3       | AB de Villiers | 164.27       |
|             | 4       | AD Russell     | 163.16       |
|             | 5       | CH Morris      | 160.11       |
|             | 6       | GJ Maxwell     | 156.60       |
|             | 7       | A Ashish Reddy | 147.59       |
|             | 8       | KA Pollard     | 142.86       |
|             | 9       | DA Miller      | 141.45       |
|             | 10      | DA Warner      | 140.94       |

## 6. What are the average runs scored by each batsman considering all the seasons?

- Code:

```

SELECT
 p.Player_Name,
 SUM(COALESCE(b.Runs_Scored, 0)) AS Total_Runs,
 COUNT(DISTINCT CONCAT(b.Match_Id, '-', b.Innings_No)) AS
 Innings_Played,
 ROUND(SUM(COALESCE(b.Runs_Scored, 0)) / NULLIF(COUNT(DISTINCT
 CONCAT(b.Match_Id, '-', b.Innings_No)), 0), 2) AS Avg_Runs
FROM Ball_by_Ball b
JOIN Player p ON b.Striker = p.Player_Id
GROUP BY p.Player_Name
ORDER BY Avg_Runs DESC;

```

- The query is designed to calculate the average runs scored by each batsman across all seasons. It first sums up the total runs scored by each player, ensuring that any missing or NULL values in the runs are treated as zeros.
- It then calculates how many innings each player has played by counting the unique combinations of match ID and innings number.
- With these two pieces of information, the query computes the average runs per innings by dividing the total runs by the innings played.
- The final result is a list of players, ranked by their average runs, showing the most consistent batsman at the top.

| Result Grid |                |            |                | Filter Rows: | Export: |
|-------------|----------------|------------|----------------|--------------|---------|
|             | Player_Name    | Total_Runs | Innings_Played | Avg_Runs     |         |
| ▶           | LMP Simmons    | 942        | 22             | 42.82        |         |
|             | V Kohli        | 2472       | 63             | 39.24        |         |
|             | DA Warner      | 2348       | 62             | 37.87        |         |
|             | N Rana         | 104        | 3              | 34.67        |         |
|             | MEK Hussey     | 1019       | 30             | 33.97        |         |
|             | AB de Villiers | 1968       | 58             | 33.93        |         |
|             | AM Rahane      | 1847       | 57             | 32.40        |         |
|             | CH Gayle       | 1634       | 51             | 32.04        |         |
|             | RV Uthappa     | 1852       | 60             | 30.87        |         |
|             | DR Smith       | 1707       | 57             | 29.95        |         |
|             | RG Sharma      | 1899       | 64             | 29.67        |         |
|             | KH Pandya      | 237        | 8              | 29.63        |         |
|             | SE Marsh       | 622        | 21             | 29.62        |         |
|             | MP Stoinis     | 146        | 5              | 29.20        |         |
|             | JP Duminy      | 1015       | 35             | 29.00        |         |
|             | SK Raina       | 1844       | 65             | 28.37        |         |
|             | AJ Finch       | 1181       | 42             | 28.12        |         |
|             | S Dhawan       | 1542       | 55             | 28.04        |         |
|             | Q de Kock      | 726        | 26             | 27.92        |         |
|             | R Dravid       | 471        | 17             | 27.71        |         |
|             | F du Plessis   | 889        | 33             | 26.94        |         |
|             | BB McCullum    | 1201       | 45             | 26.69        |         |
|             | G Gambhir      | 1569       | 59             | 26.59        |         |
|             | HM Amla        | 157        | 6              | 26.17        |         |

| Result Grid |              |            |                | Filter Rows: | Export: |
|-------------|--------------|------------|----------------|--------------|---------|
|             | Player_Name  | Total_Runs | Innings_Played | Avg_Runs     |         |
| ▶           | CJ Anderson  | 379        | 15             | 25.27        |         |
|             | DA Miller    | 1382       | 55             | 25.13        |         |
|             | SPD Smith    | 879        | 35             | 25.11        |         |
|             | MS Dhoni     | 1488       | 60             | 24.80        |         |
|             | KP Pietersen | 367        | 15             | 24.47        |         |
|             | KL Rahul     | 725        | 30             | 24.17        |         |
|             | YK Pathan    | 1273       | 53             | 24.02        |         |
|             | M Vijay      | 1223       | 51             | 23.98        |         |
|             | SR Watson    | 1315       | 55             | 23.91        |         |
|             | KK Nair      | 877        | 37             | 23.70        |         |
|             | SS Iyer      | 469        | 20             | 23.45        |         |
|             | DJ Hussey    | 351        | 15             | 23.40        |         |
|             | WP Saha      | 951        | 41             | 23.20        |         |
|             | S Narwal     | 23         | 1              | 23.00        |         |
|             | AD Russell   | 527        | 23             | 22.91        |         |
|             | M Vohra      | 728        | 32             | 22.75        |         |
|             | AC Gilchrist | 294        | 13             | 22.62        |         |
|             | AT Rayudu    | 1241       | 55             | 22.56        |         |
|             | KA Pollard   | 1320       | 59             | 22.37        |         |
|             | V Sehwag     | 849        | 38             | 22.34        |         |
|             | GJ Maxwell   | 913        | 41             | 22.27        |         |
|             | KD Karthik   | 1311       | 59             | 22.22        |         |
|             | DPMD Jaya... | 331        | 15             | 22.07        |         |
|             | EJG Morgan   | 617        | 28             | 22.04        |         |

| Player_Name   | Total_Runs | Innings_Played | Avg_Runs |
|---------------|------------|----------------|----------|
| SW Billings   | 88         | 4              | 22.00    |
| Yuvraj Singh  | 1098       | 50             | 21.96    |
| BJ Hodge      | 394        | 18             | 21.89    |
| CA Pujara     | 217        | 10             | 21.70    |
| SV Samson     | 1040       | 48             | 21.67    |
| MK Pandey     | 1018       | 47             | 21.66    |
| BJ Rohrer     | 193        | 9              | 21.44    |
| LJ Wright     | 106        | 5              | 21.20    |
| UT Khawaja    | 127        | 6              | 21.17    |
| JH Kallis     | 462        | 22             | 21.00    |
| YV Takawale   | 104        | 5              | 20.80    |
| CA Lynn       | 83         | 4              | 20.75    |
| PA Patel      | 1015       | 49             | 20.71    |
| SS Tiwary     | 388        | 19             | 20.42    |
| SR Tendulkar  | 283        | 14             | 20.21    |
| RR Pant       | 198        | 10             | 19.80    |
| KS Williamson | 155        | 8              | 19.38    |
| LA Pomersb... | 58         | 3              | 19.33    |
| A Mukund      | 19         | 1              | 19.00    |
| MJ Guptill    | 57         | 3              | 19.00    |
| MK Tiwary     | 355        | 19             | 18.68    |
| MC Henriques  | 643        | 35             | 18.37    |
| Azhar Mahm... | 202        | 11             | 18.36    |
| JC Buttler    | 255        | 14             | 18.21    |

7. What are the average wickets taken by each bowler considering all the seasons?

- Code:

```

WITH wickets_count_per_player_per_season AS
(SELECT b.Bowler, m.Season_Id, COUNT(w.Player_Out) AS wickets_taken
FROM ball_by_ball b
JOIN wicket_taken w
ON b.Match_Id = w.Match_Id AND
b.Over_Id = w.Over_Id AND
b.Ball_Id = w.Ball_Id AND
b.Innings_No = w.Innings_No
JOIN Matches m
ON m.Match_Id = w.Match_Id
GROUP BY 1,2
ORDER BY b.Bowler ASC, m.Season_Id ASC),
avg_per_season AS (
SELECT *, AVG(wickets_taken) OVER(PARTITION BY Bowler) AS
avg_wicket_per_bowler
FROM wickets_count_per_player_per_season)

SELECT DISTINCT p.Player_Name, ROUND(a.avg_wicket_per_bowler,2) AS
Avg_wicket
FROM avg_per_season a
JOIN Player p
ON p.Player_Id = a.Bowler
WHERE a.avg_wicket_per_bowler > 0
ORDER BY Avg_wicket DESC;

```

- This query calculates the average wickets taken by each bowler over all seasons. First, it counts the number of wickets taken by each bowler in every match and season.
- The data is grouped by bowler and season, ensuring we know how many wickets each bowler took in each season. Then, using a window function (AVG), it calculates the average number of wickets each bowler took across all seasons they played.

- Finally, the query joins the results with the Player table to display the bowler's name and their average wickets taken, sorting the players by their average wickets in descending order.

|   | Player_Name       | Avg_wicket |
|---|-------------------|------------|
| ► | DJ Bravo          | 27.00      |
|   | SL Malinga        | 22.00      |
|   | MM Sharma         | 19.50      |
|   | MA Starc          | 19.50      |
|   | YS Chahal         | 19.33      |
|   | B Kumar           | 19.25      |
|   | Mustafizur Rahman | 19.00      |
|   | MJ McClenaghan    | 18.00      |
|   | SP Narine         | 17.00      |
|   | D Wiese           | 17.00      |
|   | Harbhajan Singh   | 16.75      |
|   | CH Morris         | 16.00      |
|   | AR Patel          | 15.33      |
|   | JP Faulkner       | 15.25      |
|   | Sandeep Sharma    | 14.75      |
|   | A Nehra           | 14.25      |
|   | RA Jadeja         | 14.00      |
|   | UT Yadav          | 14.00      |
|   | BB Sran           | 14.00      |
|   | A Mishra          | 13.75      |
|   | MG Johnson        | 13.75      |
|   | SR Watson         | 13.50      |
|   | R Ashwin          | 13.50      |
|   | KK Cooper         | 13.00      |

8. List all the players who have average runs scored greater than the overall average and who have taken wickets greater than the overall average.

- Code:

```

WITH Player_Avg_Runs AS (
 SELECT
 bb.Striker AS Player_Id,
 p.Player_Name,
 COUNT(bb.Ball_Id) AS Balls_Faced,
 SUM(bb.Runs_Scored) AS Total_Runs,
 SUM(bb.Runs_Scored) * 1.0 / NULLIF(COUNT(bb.Ball_Id), 0) AS Avg_Runs
 FROM Ball_by_Ball bb
 JOIN Player p ON bb.Striker = p.Player_Id
 GROUP BY bb.Striker, p.Player_Name
), Overall_Avg AS (
 SELECT SUM(Runs_Scored) * 1.0 / NULLIF(COUNT(Ball_Id), 0) AS
 Overall_Avg_Runs
 FROM Ball_by_Ball
)
SELECT p.Player_Id, p.Player_Name, p.Total_Runs, p.Avg_Runs
FROM Player_Avg_Runs p
JOIN Overall_Avg oa
ON p.Avg_Runs > oa.Overall_Avg_Runs
ORDER BY p.Avg_Runs DESC;

```

```

WITH Player_Wickets AS (
 SELECT
 wt.Player_Out AS Player_Id,
 p.Player_Name,
 COUNT(*) AS Total_Wickets
 FROM Wicket_Taken wt
 JOIN Player p ON wt.Player_Out = p.Player_Id
)

```

```

 GROUP BY wt.Player_Out, p.Player_Name
), Overall_Wickets AS (
 SELECT AVG(Total_Wickets) AS Overall_Avg_Wickets
 FROM (
 SELECT COUNT(*) AS Total_Wickets FROM Wicket_Taken GROUP BY
 Player_Out
) AS Wicket_Data
)
 SELECT pw.Player_Id, pw.Player_Name, pw.Total_Wickets
 FROM Player_Wickets pw
 JOIN Overall_Wickets ow
 ON pw.Total_Wickets > ow.Overall_Avg_Wickets
 ORDER BY pw.Total_Wickets DESC;

```

- This dataset represents the total wickets taken by players over the seasons.
- It helps to identify the key bowlers based on their total wicket count.

The image shows two separate data grids side-by-side. The left grid has columns: Player\_Id, Player\_Name, Total\_Runs, and Avg\_Runs. The right grid has columns: Player\_Id, Player\_Name, and Total\_Wickets.

| Player_Id | Player_Name    | Total_Runs | Avg_Runs |
|-----------|----------------|------------|----------|
| 259       | RV Gomez       | 4          | 4.00000  |
| 246       | AN Ahmed       | 14         | 2.33333  |
| 408       | CR Brathwaite  | 83         | 2.07500  |
| 116       | PJ Sangwan     | 4          | 2.00000  |
| 228       | Bipul Sharma   | 52         | 1.92593  |
| 413       | KH Pandya      | 237        | 1.86614  |
| 66        | B Lee          | 21         | 1.75000  |
| 190       | M Morkel       | 66         | 1.73684  |
| 402       | SN Khan        | 177        | 1.71845  |
| 385       | BCJ Cutting    | 73         | 1.69767  |
| 358       | LJ Wright      | 106        | 1.68254  |
| 381       | Shivam Sharma  | 5          | 1.66667  |
| 120       | MS Gony        | 45         | 1.66667  |
| 110       | AB de Villiers | 1968       | 1.64274  |
| 334       | AD Russell     | 527        | 1.63158  |
| 416       | KJ Abbott      | 13         | 1.62500  |

| Player_Id | Player_Name    | Total_Wickets |
|-----------|----------------|---------------|
| 46        | RV Uthappa     | 58            |
| 40        | G Gambhir      | 57            |
| 147       | DR Smith       | 56            |
| 88        | KD Karthik     | 56            |
| 21        | SK Raina       | 55            |
| 187       | DA Warner      | 52            |
| 57        | RG Sharma      | 52            |
| 8         | V Kohli        | 50            |
| 185       | M Vijay        | 49            |
| 85        | AM Rahane      | 48            |
| 32        | SR Watson      | 48            |
| 208       | AT Rayudu      | 48            |
| 42        | S Dhawan       | 47            |
| 17        | PA Patel       | 46            |
| 27        | Yuvraj Singh   | 45            |
| 110       | AB de Villiers | 45            |
| 2         | BB McCullum    | 43            |
| 162       | CH Gayle       | 43            |
| 351       | SV Samson      | 42            |
| 320       | DA Miller      | 39            |
| 254       | AJ Finch       | 38            |
| 96        | MK Pandey      | 38            |
| 62        | WP Saha        | 38            |
| 305       | GJ Maxwell     | 38            |

9. Create a table rcb\_record table that shows the wins and losses of RCB in an individual venue.

- Code:

```
DROP TABLE IF EXISTS rcb_record_table;
```

```

CREATE TABLE IF NOT EXISTS rcb_record_table AS
WITH rcb_record AS
(SELECT m.Venue_Id, v.Venue_Name,
SUM(CASE WHEN Match_Winner = 2 THEN 1 ELSE 0 END) AS Win_record,
SUM(CASE WHEN Match_Winner != 2 THEN 1 ELSE 0 END) AS Loss_record
FROM matches m
JOIN venue v
ON m.Venue_Id = v.Venue_Id
WHERE (Team_1 = 2 OR Team_2 = 2) AND m.Outcome_type != 2
GROUP BY 1,2)

```

```
SELECT *, Win_record + Loss_record AS Total_Played,
```

```

ROUND((Win_record/(Win_record + Loss_record))*100,2) AS Win_percentage,
ROUND((Loss_record/(Win_record + Loss_record))*100,2) AS Loss_percentage
FROM rcb_record
ORDER BY Venue_Id;

```

### SELECT

```

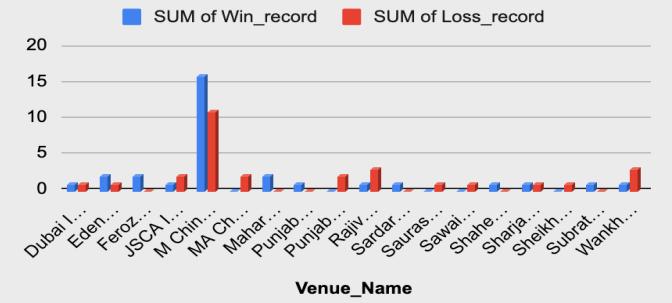
Venue_Name,Win_record,Loss_record,Total_Played,Win_percentage,Loss_per-
centage FROM rcb_record_table;

```

- The query first removes any existing table to avoid duplication and then creates a new table to track RCB's performance at different venues. It calculates the number of matches RCB won and lost at each venue and ensures that ties are excluded.
- Once the total matches are counted, the query computes win percentage and loss percentage for each venue. The final table displays venue name, wins, losses, total matches played, and win/loss percentages, ordered by venue.

| Venue_Name                                    | Win_record | Loss_record | Total_Played | Win_percentage | Loss_percentage |
|-----------------------------------------------|------------|-------------|--------------|----------------|-----------------|
| M Chinnaswamy Stadium                         | 16         | 11          | 27           | 59.26          | 40.74           |
| Punjab Cricket Association Stadium, Mohali    | 0          | 2           | 2            | 0.00           | 100.00          |
| Feroz Shah Kotla                              | 2          | 0           | 2            | 100.00         | 0.00            |
| Wankhede Stadium                              | 1          | 3           | 4            | 25.00          | 75.00           |
| Eden Gardens                                  | 2          | 1           | 3            | 66.67          | 33.33           |
| Sawai Mansingh Stadium                        | 0          | 1           | 1            | 0.00           | 100.00          |
| Rajiv Gandhi International Stadium, Uppal     | 1          | 3           | 4            | 25.00          | 75.00           |
| MA Chidambaram Stadium, Chepauk               | 0          | 2           | 2            | 0.00           | 100.00          |
| Sardar Patel Stadium, Motera                  | 1          | 0           | 1            | 100.00         | 0.00            |
| Subrata Roy Sahara Stadium                    | 1          | 0           | 1            | 100.00         | 0.00            |
| Shahid Veer Narayan Singh Internation...      | 1          | 0           | 1            | 100.00         | 0.00            |
| JSCA International Stadium Complex            | 1          | 2           | 3            | 33.33          | 66.67           |
| Sheikh Zayed Stadium                          | 0          | 1           | 1            | 0.00           | 100.00          |
| Sharjah Cricket Stadium                       | 1          | 1           | 2            | 50.00          | 50.00           |
| Dubai International Cricket Stadium           | 1          | 1           | 2            | 50.00          | 50.00           |
| Maharashtra Cricket Association Stadium       | 2          | 0           | 2            | 100.00         | 0.00            |
| Punjab Cricket Association IS Bindra Stadi... | 1          | 0           | 1            | 100.00         | 0.00            |
| Saurashtra Cricket Association Stadium        | 0          | 1           | 1            | 0.00           | 100.00          |

RCB Performance at Each Venue.



### 10. What is the impact of bowling style on wickets taken?

- Code:

```

WITH no_of_wicket_per_bowler AS (
SELECT bb.bowler, COUNT(w.Player_Out) AS no_of_wickets
FROM wicket_taken w
JOIN ball_by_ball bb
ON w.Match_Id = bb.Match_Id
AND w.Over_Id = bb.Over_Id
AND w.Ball_Id = bb.Ball_Id
AND w.Innings_No = bb.Innings_No
GROUP BY bb.Bowler),
bowler_skill_wicket AS
(SELECT n.bowler, st.Bowling_skill, no_of_wickets
FROM no_of_wicket_per_bowler n
JOIN player p
ON n.bowler = p.Player_Id
JOIN bowling_style st
ON st.Bowling_Id = p.Bowling_Skill
ORDER BY no_of_wickets DESC)
SELECT Bowling_skill AS Bowling_Style, SUM(no_of_wickets) AS
total_wickets_taken
FROM bowler_skill_wicket

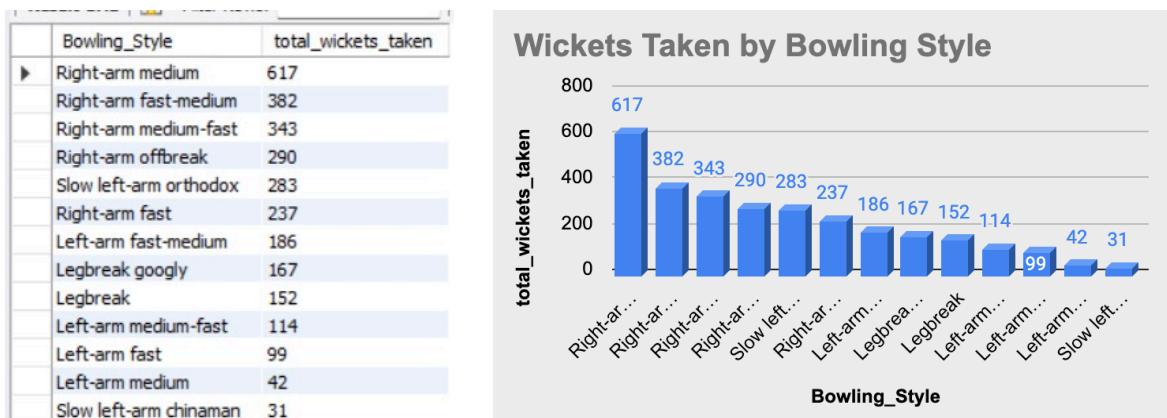
```

```

GROUP BY Bowling_skill
ORDER BY total_wickets_taken DESC;

```

- The query starts by calculating the total number of wickets taken by each bowler from the wicket\_taken and ball\_by\_ball tables. The no\_of\_wicket\_per\_bowler CTE counts how many wickets each bowler took in each match.
- Next, the bowling style is linked to each bowler by joining the player and bowling\_style tables. The bowler\_skill\_wicket CTE brings together the number of wickets taken by each bowler and their bowling style.
- Finally, the query groups the data by bowling style and calculates the total wickets taken for each bowling style, ordering the results by the number of wickets in descending order.



11. Write the SQL query to provide a status of whether the performance of the team is better than the previous year's performance on the basis of the number of runs scored by the team in the season and the number of wickets taken.

- Code:

```

WITH total_run_match_id AS (
 -- Total runs per innings per match
 SELECT
 Match_Id,
 Innings_No,
 SUM(Runs_Scored) AS total_runs
 FROM Ball_by_Ball
 GROUP BY Match_Id, Innings_No
),
total_runs_per_season AS (
 -- Total runs per season for Team_Id = 2
 SELECT
 m.Season_Id,
 SUM(CASE
 WHEN m.Toss_Decide = 1 AND m.Toss_Winner = 2 AND t.Innings_No = 2 THEN t.total_runs
 WHEN m.Toss_Decide = 2 AND m.Toss_Winner = 2 AND t.Innings_No = 1 THEN t.total_runs
 END) AS total_runs
 FROM Match m
 JOIN Team t ON m.Team_Id = t.Team_Id
 WHERE m.Season_Id = 2
)
SELECT
 s.Season_Id,
 s.Season_Name,
 s.Total_Runs AS Current_Performance,
 COALESCE(trps.Total_Runs, 0) AS Previous_Performance
FROM Season s
LEFT JOIN total_runs_per_season trps ON s.Season_Id = trps.Season_Id

```

```

THEN t.total_runs
 WHEN m.Toss_Decide = 1 AND m.Toss_Winner != 2 AND t.Innings_No =
1 THEN t.total_runs
 WHEN m.Toss_Decide = 2 AND m.Toss_Winner != 2 AND t.Innings_No =
2 THEN t.total_runs
 ELSE 0
END) AS total_runs
FROM total_run_match_id t
JOIN Matches m ON t.Match_Id = m.Match_Id
WHERE m.Team_1 = 2 OR m.Team_2 = 2
GROUP BY m.Season_Id
),

total_wickets_per_match_innings AS (
-- Total wickets per match per innings
SELECT
 w.Match_Id,
 w.Innings_No,
 COUNT(w.Player_Out) AS total_wickets
FROM Wicket_Taken w
JOIN Matches m ON m.Match_Id = w.Match_Id
WHERE m.Team_1 = 2 OR m.Team_2 = 2
GROUP BY w.Match_Id, w.Innings_No
),

total_wickets_per_season AS (
-- Total wickets per season for Team_Id = 2
SELECT
 m.Season_Id,
 SUM(CASE
 WHEN m.Toss_Decide = 1 AND m.Toss_Winner = 2 AND w.Innings_No =
1 THEN w.total_wickets
 WHEN m.Toss_Decide = 2 AND m.Toss_Winner = 2 AND w.Innings_No =
2 THEN w.total_wickets
 WHEN m.Toss_Decide = 1 AND m.Toss_Winner != 2 AND w.Innings_No =
2 THEN w.total_wickets
 WHEN m.Toss_Decide = 2 AND m.Toss_Winner != 2 AND w.Innings_No =
1 THEN w.total_wickets
 ELSE 0
 END) AS total_wickets
FROM total_wickets_per_match_innings w
JOIN Matches m ON m.Match_Id = w.Match_Id
GROUP BY m.Season_Id
)

SELECT
 s.Season_Id,
 COALESCE(r.total_runs, 0) AS total_runs,
 COALESCE(w.total_wickets, 0) AS total_wickets
FROM Season s
LEFT JOIN total_runs_per_season r ON s.Season_Id = r.Season_Id

```

```

LEFT JOIN total_wickets_per_season w ON s.Season_Id = w.Season_Id
ORDER BY s.Season_Id;

```

- The query starts by calculating the total number of wickets taken by each bowler for each match. First gathered the number of wickets taken per bowler for each match by combining data from the wicket\_taken and ball\_by\_ball tables.
- Next, the query associates each bowler with their bowling style by joining the player and bowling\_style tables. Then arrange the data by the total number of wickets taken by each bowler, making sure that to get a clear ranking of bowlers.
- Finally, the query groups the data by bowling style and sums up the total wickets taken for each bowling style. The result is a list showing which bowling styles have contributed the most to taking wickets, with styles that yield more wickets ranked higher.

|   | Season_Id | total_runs | total_wickets |
|---|-----------|------------|---------------|
| ▶ | 1         | 0          | 0             |
|   | 2         | 0          | 0             |
|   | 3         | 0          | 0             |
|   | 4         | 0          | 0             |
|   | 5         | 0          | 0             |
|   | 6         | 2430       | 103           |
|   | 7         | 1992       | 73            |
|   | 8         | 2190       | 103           |
|   | 9         | 2859       | 99            |

12. Can you derive more KPIs for the team strategy?

- Code:
 

```
-- KPI #1 Boundary %
SELECT pm.Player_Id, p.Player_Name,
 ROUND((SUM(CASE WHEN b.Runs_Scored = 4 THEN 1 ELSE 0 END) /
COUNT(*)) * 100, 2) AS Four_Percentage,
 ROUND((SUM(CASE WHEN b.Runs_Scored = 6 THEN 1 ELSE 0 END) /
COUNT(*)) * 100, 2) AS Six_Percentage
FROM Ball_by_Ball b
JOIN Matches m
ON m.Match_Id = b.Match_Id
JOIN Player_Match pm
ON m.Match_Id = pm.Match_Id AND pm.Player_Id = b.Striker
JOIN Player p
ON pm.Player_Id = p.Player_Id
WHERE m.Season_Id IN (SELECT DISTINCT Season_Id FROM Matches WHERE
Team_1 = 2 OR Team_2 = 2)
GROUP BY pm.Player_Id, p.Player_Name
ORDER BY Six_Percentage DESC, Four_Percentage DESC
LIMIT 15;

-- KPI #2 Bowling strike rate (Lower is better)
SELECT bb.Bowler, p.Player_Name,
```

```

 ROUND(COUNT(bb.Ball_Id) / COUNT(w.Player_Out),2) AS Strike_Rate
FROM ball_by_ball bb
LEFT JOIN wicket_taken w
 ON bb.Match_Id = w.Match_Id
 AND bb.Over_Id = w.Over_Id
 AND bb.Ball_Id = w.Ball_Id
JOIN player p
ON p.Player_Id = bb.Bowler
WHERE bb.Team_Bowling = 2
GROUP BY bb.Bowler
HAVING Strike_Rate IS NOT NULL
ORDER BY Strike_Rate ASC
LIMIT 15;

```

- In this query, we focus on deriving two key performance indicators (KPIs) that can provide valuable insights into RCB's (Team 2) batting and bowling performance to guide their strategy.
  - KPI #1: Boundary Percentage:**  
The first part of the query calculates how often RCB's players hit boundaries (4s and 6s) during a match. This is done by checking each ball faced by the players and counting the number of times a 4 or 6 was scored. The results are then presented as percentages of total balls faced to determine the boundary percentage. The query ranks the players based on the percentage of 6s hit, followed by 4s, helping us understand which players are aggressive boundary hitters.
  - KPI #2: Bowling Strike Rate:**  
The second KPI focuses on the strike rate of bowlers, which is an important measure of how often a bowler takes a wicket. The query calculates the strike rate for each bowler, defined as the number of balls bowled per wicket taken. A lower strike rate is preferable, as it means a bowler takes wickets more frequently. The results show the top bowlers based on the best (lowest) strike rate, helping the team assess which bowlers are most effective at taking wickets.

|   | Bowler    | Player_Name     | Strike_Rate     |                | Result Grid | Filter Rows:   | Export: | W/    |
|---|-----------|-----------------|-----------------|----------------|-------------|----------------|---------|-------|
|   | Player_Id | Player_Name     | Four_Percentage | Six_Percentage |             |                |         |       |
| ▶ | 369       | Sachin Baby     | 3.33            |                | 408         | CR Brathwaite  | 15.00   | 20.00 |
|   | 363       | KW Richardson   | 5.80            |                | 228         | Bipul Sharma   | 3.70    | 18.52 |
|   | 15        | Z Khan          | 6.33            |                | 403         | MJ McClenaghan | 6.90    | 17.24 |
|   | 461       | AF Milne        | 8.33            |                | 66          | B Lee          | 8.33    | 16.67 |
|   | 244       | A Mithun        | 8.33            |                | 385         | BCJ Cutting    | 9.30    | 16.28 |
|   | 81        | R Vinay Kumar   | 8.64            |                | 311         | HV Patel       | 4.76    | 14.29 |
|   | 283       | J Syed Mohammad | 8.86            |                | 433         | ER Dwivedi     | 13.33   | 13.33 |
|   | 109       | JA Morkel       | 9.18            |                | 416         | KJ Abbott      | 0.00    | 12.50 |
|   | 349       | R Rampaul       | 9.30            |                | 120         | MS Gony        | 14.81   | 11.11 |
|   | 378       | MA Starc        | 9.30            |                | 334         | AD Russell     | 13.93   | 10.53 |
|   | 311       | HV Patel        | 9.48            |                | 162         | CH Gayle       | 11.02   | 10.34 |
|   | 434       | CJ Jordan       | 9.53            |                | 431         | N Rana         | 8.97    | 10.26 |
|   | 270       | S Aravind       | 9.77            |                | 413         | KH Pandya      | 17.32   | 10.24 |
|   | 382       | YS Chahal       | 9.95            |                | 309         | KK Cooper      | 12.00   | 10.00 |
|   | 163       | MC Henriques    | 10.23           |                | 305         | GJ Maxwell     | 13.04   | 9.61  |

13. Using SQL, write a query to find out the average wickets taken by each bowler in each venue. Also, rank the gender according to the average value.

- Code:

```

WITH player_wickets AS (
 SELECT v.Venue_Id, v.Venue_Name,
 p.Player_Name,
 COUNT(w.Player_Out) AS total_wickets,
 COUNT(DISTINCT m.Match_Id) AS matches_played -- Distinct matches
 where the player bowled
 FROM Wicket_Taken w
 JOIN Ball_by_Ball b
 ON w.Match_Id = b.Match_Id
 AND w.Over_Id = b.Over_Id
 AND w.Ball_Id = b.Ball_Id
 AND w.Innings_No = b.Innings_No -- Ensuring correct innings mapping
 JOIN Matches m
 ON b.Match_Id = m.Match_Id
 JOIN Player_Match pm
 ON pm.Match_Id = m.Match_Id
 AND pm.Player_Id = b.Bowler -- Ensuring only actual bowlers are counted
 JOIN Player p
 ON p.Player_Id = pm.Player_Id
 JOIN Venue v
 ON v.Venue_Id = m.Venue_Id
 GROUP BY v.Venue_Id, v.Venue_Name, p.Player_Name
),
unranked_table AS (
 SELECT Venue_Id, Venue_Name, Player_Name,
 total_wickets,
 matches_played,
 ROUND(total_wickets / matches_played, 2) AS avg_wickets
 FROM player_wickets
)
SELECT *, DENSE_RANK() OVER(ORDER BY avg_wickets DESC) AS Ranking
FROM unranked_table
WHERE matches_played > 10;

```

- This query aims to determine which bowlers perform best at different venues based on their average wickets per match.

- Counting Wickets at Each Venue:

- The query starts by counting the total wickets taken by each bowler at every venue.
- It ensures that only actual bowlers who played in the match are included.
- The number of distinct matches where the player bowled is also counted to calculate averages later.

- Calculating Average Wickets per Match:

- It then calculates the average wickets per match for each bowler at a specific venue by dividing total wickets by matches played.
- This helps identify which bowlers are the most effective wicket-takers at different venues

3. Ranking Bowlers by Average Wickets:

- The query assigns a rank to each bowler based on their average wickets per match, using DENSE\_RANK() so that bowlers with the same average wickets share the same rank.
- Only bowlers who have bowled in more than 10 matches are considered to ensure a fair ranking

|   | Venue_Id | Venue_Name                                | Player_Name     | total_wickets | matches_played | avg_wickets | Ranking |
|---|----------|-------------------------------------------|-----------------|---------------|----------------|-------------|---------|
| ▶ | 5        | Eden Gardens                              | SP Narine       | 26            | 12             | 2.17        | 1       |
|   | 8        | MA Chidambaram Stadium, Chepauk           | DJ Bravo        | 29            | 14             | 2.07        | 2       |
|   | 1        | M Chinnaswamy Stadium                     | YS Chahal       | 31            | 17             | 1.82        | 3       |
|   | 4        | Wankhede Stadium                          | SL Malinga      | 32            | 18             | 1.78        | 4       |
|   | 4        | Wankhede Stadium                          | Harbhajan Singh | 30            | 18             | 1.67        | 5       |
|   | 8        | MA Chidambaram Stadium, Chepauk           | R Ashwin        | 16            | 11             | 1.45        | 6       |
|   | 7        | Rajiv Gandhi International Stadium, Uppal | B Kumar         | 17            | 12             | 1.42        | 7       |

14. Which of the given players have consistently performed well in past seasons? (will you use any visualization to solve the problem)

- Code:

```
#Bowling performance
```

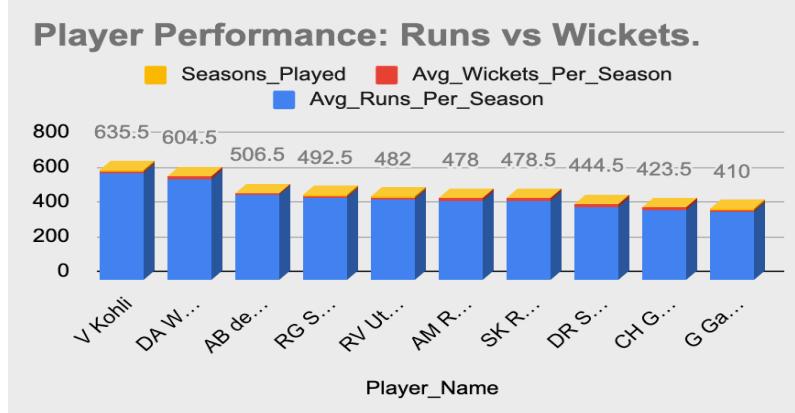
```
WITH Player_Season_Performance AS (
 SELECT
 p.Player_Name,
 s.Season_Year,
 SUM(bbb.Runs_Scored) AS Total_Runs,
 COUNT(wt.Player_Out) AS Total_Wickets,
 COUNT(DISTINCT m.Match_Id) AS Matches_Played
 FROM Player p
 INNER JOIN Ball_by_Ball bbb ON p.Player_Id = bbb.Striker
 LEFT JOIN Wicket_Taken wt ON bbb.Match_Id = wt.Match_Id
 AND bbb.Over_Id = wt.Over_Id
 AND bbb.Ball_Id = wt.Ball_Id
 AND bbb.Innings_No = wt.Innings_No
 INNER JOIN Matches m ON bbb.Match_Id = m.Match_Id
 INNER JOIN Season s ON m.Season_Id = s.Season_Id
 WHERE p.Player_Id = bbb.Bowler OR p.Player_Id = bbb.Striker
 GROUP BY p.Player_Name, s.Season_Year
)
SELECT
 Player_Name,
 AVG(Total_Runs) AS Avg_Runs_Per_Season,
 AVG(Total_Wickets) AS Avg_Wickets_Per_Season,
 COUNT(Season_Year) AS Seasons Played
FROM Player_Season_Performance
GROUP BY Player_Name
HAVING Seasons Played > 3
ORDER BY Avg_Runs_Per_Season DESC, Avg_Wickets_Per_Season DESC
LIMIT 10;
```

- The query aims to identify players who have shown consistent performance over multiple seasons, both in terms of batting and bowling. It first calculates each player's

total runs and total wickets for each season, while also counting how many matches they played. After calculating these statistics for each season, it finds the average runs and average wickets across the seasons.

- The focus is on players who have played more than 3 seasons, ensuring the consistency of their performance. Finally, the query sorts these players based on their average performance and returns the top 10 players who have consistently excelled in both runs and wickets.
- These players stand out as key performers who have been reliable over time, helping the team in both batting and bowling.

|   | Player_Name    | Avg_Runs_Per_Season | Avg_Wickets_Per_Season | Seasons Played |
|---|----------------|---------------------|------------------------|----------------|
| ▶ | V Kohli        | 618.0000            | 13.5000                | 4              |
|   | DA Warner      | 587.0000            | 13.5000                | 4              |
|   | AB de Villiers | 492.0000            | 10.5000                | 4              |
|   | RG Sharma      | 474.7500            | 13.7500                | 4              |
|   | RV Uthappa     | 463.0000            | 15.0000                | 4              |
|   | AM Rahane      | 461.7500            | 12.2500                | 4              |
|   | SK Raina       | 461.0000            | 13.5000                | 4              |
|   | DR Smith       | 426.7500            | 13.7500                | 4              |
|   | CH Gayle       | 408.5000            | 11.0000                | 4              |
|   | G Gambhir      | 392.2500            | 13.7500                | 4              |



15. Are there players whose performance is more suited to specific venues or conditions? (how would you present this using charts?)

- Code:  
**#Batting performance**  

```
SELECT p.Player_Name, v.Venue_Name,
 SUM(b.Runs_Scored) AS Total_Runs,
 COUNT(b.Ball_Id) AS Balls_Faced,
 ROUND(SUM(b.Runs_Scored) / COUNT(b.Ball_Id), 2) * 100 AS Strike_Rate
FROM Ball_by_Ball b
JOIN Matches m ON m.Match_Id = b.Match_Id
JOIN Player p ON p.Player_Id = b.Striker
JOIN Venue v ON m.Venue_Id = v.Venue_Id
GROUP BY p.Player_Name, v.Venue_Name
HAVING Total_Runs > 0 AND Balls_Faced > 100
ORDER BY Total_Runs DESC, p.Player_Name;
```

### #Bowling performance

```

SELECT p.Player_Name, v.Venue_Name,
 COUNT(w.Player_Out) AS Wickets_Taken,
 COUNT(b.Ball_Id) AS Balls_Bowled
 FROM ball_by_ball b
 JOIN wicket_taken w ON b.Match_Id = w.Match_Id
 AND b.Over_Id = w.Over_Id AND b.Ball_Id = w.Ball_Id AND b.Innings_No =
 w.Innings_No
 JOIN matches m ON m.Match_Id = w.Match_Id
 JOIN player p ON p.Player_Id = b.Bowler
 JOIN venue v ON m.Venue_Id = v.Venue_Id
 GROUP BY p.Player_Name, v.Venue_Name
 HAVING Balls_Bowled > 5
 ORDER BY Wickets_Taken DESC, p.Player_Name;

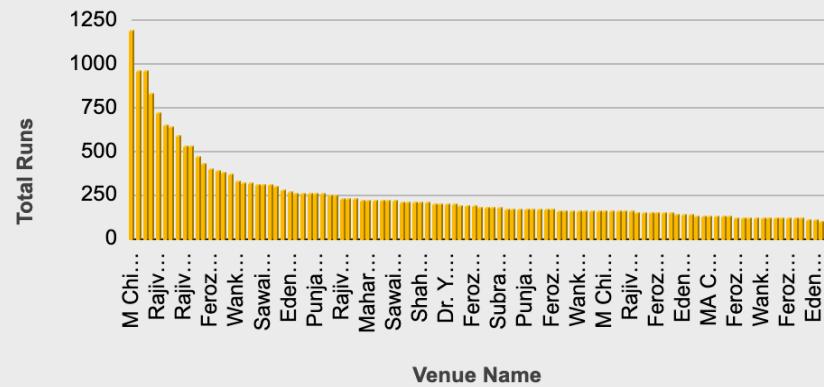
```

| Player_Name   | Venue_Name                                 | Total_Runs | Balls_Faced | Strike_Rate |
|---------------|--------------------------------------------|------------|-------------|-------------|
| V Kohli       | M Chinnaswamy Stadium                      | 1200       | 824         | 146.00      |
| AB de Villers | M Chinnaswamy Stadium                      | 967        | 569         | 170.00      |
| CH Gayle      | M Chinnaswamy Stadium                      | 963        | 660         | 146.00      |
| RG Sharma     | Wankhede Stadium                           | 832        | 585         | 142.00      |
| DA Warner     | Rajiv Gandhi International Stadium, Uppal  | 724        | 467         | 155.00      |
| G Gambhir     | Eden Gardens                               | 648        | 548         | 118.00      |
| KA Pollard    | Wankhede Stadium                           | 646        | 417         | 155.00      |
| RV Uthappa    | Eden Gardens                               | 594        | 451         | 132.00      |
| S Dhawan      | Rajiv Gandhi International Stadium, Uppal  | 537        | 482         | 111.00      |
| YK Pathan     | Eden Gardens                               | 532        | 350         | 152.00      |
| MS Dhoni      | MA Chidambaram Stadium, Chepauk            | 468        | 321         | 146.00      |
| AT Rayudu     | Wankhede Stadium                           | 433        | 324         | 134.00      |
| JP Duminy     | Feroz Shah Kotla                           | 399        | 326         | 122.00      |
| SK Raina      | MA Chidambaram Stadium, Chepauk            | 395        | 312         | 127.00      |
| MK Pandey     | Eden Gardens                               | 377        | 307         | 123.00      |
| DA Miller     | Punjab Cricket Association Stadium, Mohali | 374        | 245         | 153.00      |

| Player_Name     | Venue_Name                                | Wickets_Taken | Balls_Bowled |
|-----------------|-------------------------------------------|---------------|--------------|
| SL Malinga      | Wankhede Stadium                          | 32            | 32           |
| YS Chahal       | M Chinnaswamy Stadium                     | 31            | 31           |
| Harbhajan Singh | Wankhede Stadium                          | 30            | 30           |
| DJ Bravo        | MA Chidambaram Stadium, Chepauk           | 29            | 29           |
| SP Narine       | Eden Gardens                              | 26            | 26           |
| A Mishra        | Rajiv Gandhi International Stadium, Uppal | 19            | 19           |
| MC Henriques    | Rajiv Gandhi International Stadium, Uppal | 18            | 18           |
| MG Johnson      | Wankhede Stadium                          | 18            | 18           |
| PP Chawla       | Eden Gardens                              | 18            | 18           |
| SR Watson       | M Chinnaswamy Stadium                     | 18            | 18           |
| A Mishra        | Feroz Shah Kotla                          | 17            | 17           |
| AD Russell      | Eden Gardens                              | 17            | 17           |
| B Kumar         | Rajiv Gandhi International Stadium, Uppal | 17            | 17           |
| MM Sharma       | MA Chidambaram Stadium, Chepauk           | 17            | 17           |
| UT Yadav        | Feroz Shah Kotla                          | 17            | 17           |
| Imran Tahir     | Feroz Shah Kotla                          | 16            | 16           |
| R Ashwin        | MA Chidambaram Stadium, Chepauk           | 16            | 16           |
| JP Faulkner     | Sawai Mansingh Stadium                    | 14            | 14           |
| MJ McClenaghan  | Wankhede Stadium                          | 14            | 14           |
| NLTC Perera     | Rajiv Gandhi International Stadium, Uppal | 14            | 14           |
| DS Kulkarni     | Wankhede Stadium                          | 13            | 13           |
| DW Steyn        | Rajiv Gandhi International Stadium, Uppal | 13            | 13           |
| MA Starc        | M Chinnaswamy Stadium                     | 13            | 13           |
| R Vinay Kumar   | M Chinnaswamy Stadium                     | 13            | 13           |

### Venue Name By Total Runs

■ Total\_Runs



## Subjective Questions

1. How does the toss decision affect the result of the match? (which visualizations could be used to present your answer better) And is the impact limited to only specific venues?

- Code:

```

SELECT v.Venue_Id, v.Venue_Name,
 CASE WHEN m.Toss_Decide = 1 THEN 'Field' ELSE 'Bat' END AS
Toss_Decide,
 COUNT(*) AS Total_Matches,
 SUM(CASE WHEN m.Toss_Winner = m.Match_Winner THEN 1 ELSE 0 END)
AS Toss_Winner_Wins,
 SUM(CASE WHEN m.Toss_Winner != m.Match_Winner THEN 1 ELSE 0
END) AS Toss_Winner_Losses,
 ROUND((SUM(CASE WHEN m.Toss_Winner = m.Match_Winner THEN 1
ELSE 0 END) * 100.0) / COUNT(*), 2) AS Win_Percentage
FROM Matches m
JOIN Venue v ON m.Venue_Id = v.Venue_Id
GROUP BY v.Venue_Id, v.Venue_Name, m.Toss_Decide
ORDER BY v.Venue_Name, Toss_Decide;

```

- This query examines how the toss decision (whether a team chooses to bat or field) impacts the outcome of the match. It looks at each venue and how the toss influences the match results. The query breaks down the data by venue and toss decision (whether the team decided to bat or field). It calculates the total number of matches, toss winner wins, and toss winner losses. The win percentage of the team that won the toss is also calculated, giving us an insight into how often the team winning the toss ends up winning the match.
- By analyzing this for different venues, the toss decision has a stronger impact at certain venues. This could indicate that certain conditions (like pitch conditions, weather, etc.) at particular venues make the toss decision more critical to the outcome.

- Recommendations:**

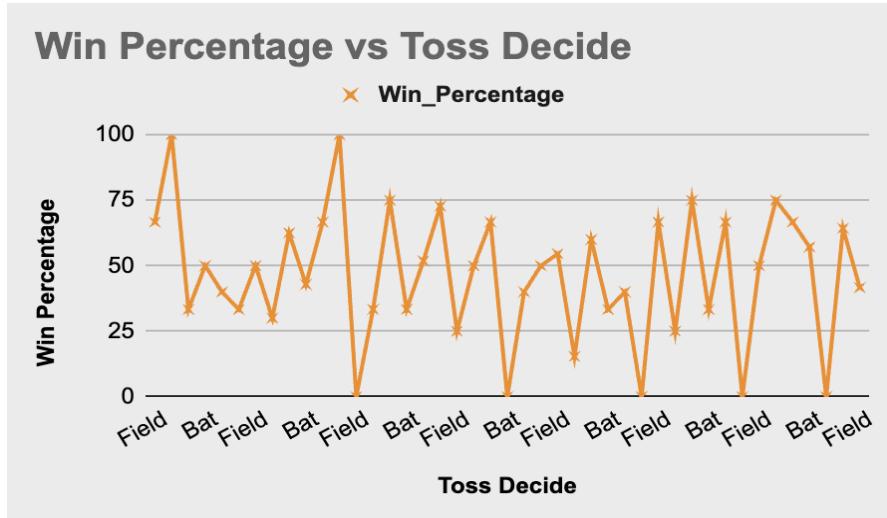
- Strategic Focus on Toss Decision:**

- Teams should focus on making better decisions after winning the toss, especially at venues where the decision to bat or field has a high impact on the result (e.g., Brabourne Stadium).

- Monitor Toss Influence:**

- Analyze weather conditions or pitch characteristics at different venues to refine the impact of toss decisions.

| Venue_Id | Venue_Name                                         | Toss_Decide | Total_Matches | Toss_Winner_Wins | Toss_Winner_Losses | Win_Percentage |
|----------|----------------------------------------------------|-------------|---------------|------------------|--------------------|----------------|
| 20       | Barabati Stadium                                   | Field       | 3             | 2                | 1                  | 66.67          |
| 18       | Brabourne Stadium                                  | Bat         | 1             | 1                | 0                  | 100.00         |
| 18       | Brabourne Stadium                                  | Field       | 3             | 1                | 2                  | 33.33          |
| 25       | Dr. Y.S. Rajasekhar Reddy ACA-VDCA Cricket ...     | Bat         | 4             | 2                | 2                  | 50.00          |
| 25       | Dr. Y.S. Rajasekhar Reddy ACA-VDCA Cricket ...     | Field       | 5             | 2                | 3                  | 40.00          |
| 31       | Dubai International Cricket Stadium                | Bat         | 3             | 1                | 2                  | 33.33          |
| 31       | Dubai International Cricket Stadium                | Field       | 4             | 2                | 2                  | 50.00          |
| 5        | Eden Gardens                                       | Bat         | 10            | 3                | 7                  | 30.00          |
| 5        | Eden Gardens                                       | Field       | 16            | 10               | 6                  | 62.50          |
| 3        | Feroz Shah Kotla                                   | Bat         | 7             | 3                | 4                  | 42.86          |
| 3        | Feroz Shah Kotla                                   | Field       | 18            | 12               | 6                  | 66.67          |
| 35       | Green Park                                         | Field       | 2             | 2                | 0                  | 100.00         |
| 22       | Himachal Pradesh Cricket Association Stadium       | Field       | 2             | 0                | 2                  | 0.00           |
| 28       | JSCA International Stadium Complex                 | Bat         | 3             | 1                | 2                  | 33.33          |
| 28       | JSCA International Stadium Complex                 | Field       | 4             | 3                | 1                  | 75.00          |
| 1        | M Chinnaswamy Stadium                              | Bat         | 3             | 1                | 2                  | 33.33          |
| 1        | M Chinnaswamy Stadium                              | Field       | 27            | 14               | 11                 | 51.85          |
| 8        | MA Chidambaram Stadium, Chepauk                    | Bat         | 11            | 8                | 3                  | 72.73          |
| 8        | MA Chidambaram Stadium, Chepauk                    | Field       | 4             | 1                | 3                  | 25.00          |
| 32       | Maharashtra Cricket Association Stadium            | Bat         | 2             | 1                | 1                  | 50.00          |
| 32       | Maharashtra Cricket Association Stadium            | Field       | 6             | 4                | 2                  | 66.67          |
| 33       | Punjab Cricket Association IS Bindra Stadium, M... | Bat         | 2             | 0                | 2                  | 0.00           |
| 33       | Punjab Cricket Association IS Bindra Stadium, M... | Field       | 5             | 2                | 3                  | 40.00          |
| 2        | Punjab Cricket Association Stadium, Mohali         | Bat         | 2             | 1                | 1                  | 50.00          |



2. Suggest some of the players who would be best fit for the team.

- Code:

```
#List of consistently performing batsmen
SELECT p.Player_Name,
 SUM(b.Runs_Scored) AS Total_Runs,
 COUNT(b.Ball_Id) AS Balls_Faced,
 ROUND((SUM(b.Runs_Scored) / COUNT(b.Ball_Id)) * 100, 2) AS Strike_Rate,
 ROUND(SUM(b.Runs_Scored) / COUNT(DISTINCT m.Match_Id), 2) AS
Average_Runs
FROM Player p
JOIN Ball_by_Ball b ON p.Player_Id = b.Striker
JOIN Matches m ON b.Match_Id = m.Match_Id
WHERE m.Season_Id >= 4
GROUP BY p.Player_Name

ORDER BY Total_Runs DESC, Strike_Rate DESC
LIMIT 10;
```

#List of consistent bowlers

```
SELECT p.Player_Name,
 COUNT(w.Player_Out) AS Wickets_Taken,
 ROUND(SUM(bb.Ball_Id) / COUNT(w.Player_Out),2) AS Strike_Rate,
 ROUND(SUM(bb.Runs_Scored) / (SUM(bb.Ball_Id)/6),2) AS Economy_Rate
FROM Player p
JOIN Ball_by_Ball bb ON p.Player_Id = bb.Bowler
JOIN Matches m ON bb.Match_Id = m.Match_Id
JOIN Wicket_Taken w
ON bb.Match_Id = w.Match_Id AND bb.Over_Id = w.Over_Id AND bb.Innings_No
= w.Innings_No AND bb.Ball_Id = w.Ball_Id
WHERE m.Season_Id >= 4
GROUP BY p.Player_Id, p.Player_Name
ORDER BY Wickets_Taken DESC, Economy_Rate ASC, Strike_Rate ASC
LIMIT 10;
```

- This query identifies the best-performing batsmen and bowlers based on their consistent performances over multiple seasons.

### 1. Identifying the Best Batsmen:

- The first query analyzes players who have scored the most runs while maintaining a high strike rate and strong batting average.
- It filters for matches from Season 4 onwards, ensuring that only recent and consistent performers are considered.
- The results list the top 10 batsmen who have scored the most runs, with priority given to those who have a better strike rate

### 2. Identifying the Best Bowlers:

- The second query focuses on bowlers who have taken the most wickets while maintaining a good economy rate and low strike rate.
- A low strike rate means the bowler takes wickets frequently, and a low economy rate means they concede fewer runs.
- The top 10 bowlers are selected based on their wickets taken, economy rate, and strike rate to find those who can consistently deliver match-winning performances.

### ● Recommendations:

#### 1. Target Efficient Bowlers:

- Players like DJ Bravo and MM Sharma should be prioritized for recruitment, as they offer a good balance of wickets taken, strike rate, and economy rate.
- Bowlers who excel in both wickets taken and economy rate (such as B Kumar and DJ Bravo) should be valued for their consistency.

#### 2. Bowling Strategy:

- For better performance, the team should consider having bowlers with both low economy rates and quick wicket-taking abilities, ensuring they can contain the opposition while also picking up wickets.

| Result Grid                       |                |            |             |             |              |
|-----------------------------------|----------------|------------|-------------|-------------|--------------|
| Filter Rows: <input type="text"/> |                |            |             |             |              |
|                                   | Player_Name    | Total_Runs | Balls_Faced | Strike_Rate | Average_Runs |
| ▶                                 | V Kohli        | 2472       | 1822        | 135.68      | 39.87        |
|                                   | DA Warner      | 2348       | 1666        | 140.94      | 38.49        |
|                                   | AB de Villiers | 1968       | 1198        | 164.27      | 34.53        |
|                                   | RG Sharma      | 1899       | 1433        | 132.52      | 29.67        |
|                                   | RV Uthappa     | 1852       | 1899        | 127.11      | 30.87        |
|                                   | AM Rahane      | 1847       | 1580        | 116.90      | 32.40        |
|                                   | SK Raina       | 1844       | 1380        | 133.62      | 28.37        |
|                                   | DR Smith       | 1707       | 1343        | 127.10      | 29.95        |
|                                   | CH Gayle       | 1634       | 1189        | 137.43      | 33.35        |
|                                   | G Gambhir      | 1569       | 1367        | 114.78      | 26.59        |

|   | Player_Name     | Wickets_Taken | Strike_Rate | Economy_Rate |
|---|-----------------|---------------|-------------|--------------|
| ▶ | DJ Bravo        | 81            | 3.77        | 0.02         |
|   | MM Sharma       | 78            | 3.90        | 0.06         |
|   | B Kumar         | 77            | 3.60        | 0.00         |
|   | SP Narine       | 68            | 3.81        | 0.07         |
|   | Harbhajan Singh | 67            | 3.73        | 0.00         |
|   | SL Malinga      | 66            | 3.80        | 0.05         |
|   | JP Faulkner     | 61            | 3.97        | 0.12         |
|   | Sandeep Sharma  | 59            | 4.07        | 0.05         |
|   | YS Chahal       | 58            | 3.69        | 0.00         |
|   | A Nehra         | 57            | 3.89        | 0.08         |

### 3. What are some of the parameters that should be focused on while selecting the players?

- Code:

#Key parameters for selecting players

# A. Death over bowling performance

SELECT p.Player\_Name,

    SUM(CASE WHEN bb.Over\_Id >= 16 AND bb.Over\_Id <= 20 AND p.Player\_Id

    IN (SELECT Bowler FROM ball\_by\_ball) THEN bb.Runs\_Scored ELSE 0 END)

    AS Death\_Over\_Runs\_Conceded

    FROM Player p

    JOIN ball\_by\_ball bb ON p.Player\_Id = bb.Striker OR p.Player\_Id = bb.Bowler

```

JOIN Matches m ON bb.Match_Id = m.Match_Id
WHERE m.Season_Id >= 4
GROUP BY p.Player_Name
HAVING COUNT(bb.Ball_Id) > 100 AND Death_Over_Runs_Conceded != 0
ORDER BY Death_Over_Runs_Conceded ASC
LIMIT 10;

```

#### # B. Batting performance across different venues

```

SELECT p.Player_Name,
v.Venue_Id, v.Venue_Name,
SUM(bb.Runs_Scored) AS Total_Runs,
COUNT(bb.Ball_Id) AS Balls_Faced,
ROUND(SUM(bb.Runs_Scored) / COUNT(bb.Ball_Id), 2) * 100 AS
Strike_Rate
FROM Player p
JOIN Ball_by_Ball bb ON p.Player_Id = bb.Striker
JOIN Matches m ON bb.Match_Id = m.Match_Id
JOIN Venue v ON m.Venue_Id = v.Venue_Id
JOIN Ball_by_Ball bb2
ON bb.Match_Id = bb2.Match_Id
AND bb.Over_Id = bb2.Over_Id
AND bb.Ball_Id = bb2.Ball_Id
AND bb.Innings_No = bb2.Innings_No
GROUP BY p.Player_Name, v.Venue_Id, v.Venue_Name
ORDER BY Total_Runs DESC, Strike_Rate DESC
LIMIT 10;

```

- When selecting players for the team, there are key parameters to focus on that reflect both batting and bowling performances under different circumstances:

##### 1. Death Over Bowling Performance:

- The first parameter focuses on death-over bowling, which is critical in the final overs of a match. This performance is important because a bowler's ability to restrict runs during the final overs (overs 16 to 20) can make or break a game.
- The query calculates the runs conceded by bowlers during death overs. The best bowlers are those who have a low number of runs conceded in these crucial overs, signaling their ability to restrict scoring in high-pressure situations.
- It filters out players who have bowled less than 100 balls and only includes those who have conceded runs in death overs, providing a focus on specialist death-over bowlers.

##### 2. Batting Performance Across Different Venues:

- The second key parameter focuses on a player's batting consistency across different venues. A player's strike rate and total runs at different venues offer insights into how adaptable they are in varying conditions (e.g., different pitch types and weather conditions).
- The query calculates total runs and strike rate for each player at each venue and ranks players who have performed the best. A high strike rate coupled with a high total runs shows a player's ability to dominate in different conditions, making them versatile and valuable

in various match scenarios.

- **Recommendations:**

1. **Prioritize High Strike Rate Players:**

- Select players with high strike rates (like AB de Villiers and DA Warner), as they provide the ability to score quickly and change the course of the game.

2. **Consistency in Runs:**

- Focus on players who consistently score high total runs, like V Kohli, as their consistent performance over multiple seasons adds stability to the batting lineup.

| Result Grid   |    | Player_Name    | Venue_Id | Venue_Name                                | Total_Runs | Balls_Faced | Strike_Rate |
|---------------|----|----------------|----------|-------------------------------------------|------------|-------------|-------------|
| KC Cariappa   | 6  | V Kohli        | 1        | M Chinnaswamy Stadium                     | 1200       | 824         | 146.00      |
| LR Shukla     | 7  | AB de Villiers | 1        | M Chinnaswamy Stadium                     | 967        | 569         | 170.00      |
| A Chandila    | 10 | CH Gayle       | 1        | M Chinnaswamy Stadium                     | 963        | 660         | 146.00      |
| JDS Neesham   | 13 | RG Sharma      | 4        | Wankhede Stadium                          | 832        | 585         | 142.00      |
| J Yadav       | 17 | DA Warner      | 7        | Rajiv Gandhi International Stadium, Uppal | 724        | 467         | 155.00      |
| Parvez Rasool | 17 | G Gambhir      | 5        | Eden Gardens                              | 648        | 548         | 118.00      |
| AC Gilchrist  | 20 | KA Pollard     | 4        | Wankhede Stadium                          | 646        | 417         | 155.00      |
| S Gopal       | 20 | RV Uthappa     | 5        | Eden Gardens                              | 594        | 451         | 132.00      |
| AS Rajpoot    | 22 | S Dhawan       | 7        | Rajiv Gandhi International Stadium, Uppal | 537        | 482         | 111.00      |
| AM Nayar      | 28 | YK Pathan      | 5        | Eden Gardens                              | 532        | 350         | 152.00      |

4. Which players offer versatility in their skills and can contribute effectively with both bat and ball? (can you visualize the data for the same)

- Code:

#We can find all-rounder performance for all players

```

WITH batting_performance AS (
 SELECT p.Player_Id, p.Player_Name,
 SUM(b.Runs_Scored) AS Total_Runs,
 COUNT(bb.Ball_Id) AS Balls_Faced,
 ROUND((SUM(b.Runs_Scored) / COUNT(bb.Ball_Id)) * 100, 2) AS
Batting_Strike_Rate
 FROM Player p
 JOIN Ball_by_Ball bb ON p.Player_Id = bb.Striker
 JOIN Ball_by_Ball b
 ON bb.Match_Id = b.Match_Id
 AND bb.Over_Id = b.Over_Id
 AND bb.Ball_Id = b.Ball_Id
 AND bb.Innings_No = b.Innings_No
 WHERE b.Runs_Scored IS NOT NULL -- Ensuring only valid scoring
deliveries are considered
 GROUP BY p.Player_Id, p.Player_Name
),
bowling_performance AS (
 SELECT p.Player_Id, p.Player_Name,

```

```

 COUNT(w.Player_Out) AS Total_Wickets,
 ROUND(SUM(bb.Team_Batting) / COUNT(bb.Ball_Id),2) AS
Economy_Rate
FROM player p
JOIN ball_by_ball bb ON p.Player_Id = bb.Bowler
JOIN wicket_taken w ON bb.Match_Id = w.Match_Id
 AND bb.Over_Id = w.Over_Id
 AND bb.Ball_Id = w.Ball_Id
 AND bb.Innings_No = w.Innings_No
GROUP BY p.Player_Id, p.Player_Name
)
SELECT bp.Player_Id, bp.Player_Name,
bp.Total_Runs, bp.Batting_Strike_Rate, bp.Balls_Faced,
bw.Total_Wickets, bw.Economy_Rate
FROM batting_performance bp
JOIN bowling_performance bw ON bp.Player_Id = bw.Player_Id
ORDER BY bp.Batting_Strike_Rate DESC, bw.Economy_Rate ASC
LIMIT 10;

```

- This query aims to identify the all-rounders who perform well both with the bat and the ball, contributing to the team in multiple ways. The focus is on players who can dominate with the bat while also restricting runs and taking wickets with the ball.

#### 1. Batting Performance:

- The first part of the query calculates batting performance by summarizing each player's total runs and strike rate (how quickly they score runs). It also includes the number of balls faced to give context to the strike rate.
- Players with high strike rates and total runs are generally more aggressive batsmen, helping the team build or accelerate the innings.

#### 2. Bowling Performance:

- The second part calculates bowling performance by counting total wickets taken by the player and calculating their economy rate (how many runs they concede per over). A low economy rate is desirable, as it shows the bowler is effective in restricting runs while taking wickets.
- Bowlers who consistently take wickets and maintain a low economy rate are vital for restricting the opposition's scoring.

#### 3. Combining Both Performances:

- The final query joins the batting performance and bowling performance to identify all-rounders. The players are ranked first by batting strike rate (to highlight aggressive batsmen) and then by economy rate (to prioritize efficient bowlers).

#### • Recommendations:

##### 1. Prioritize All-Rounders:

- KH Pandya, CR Brathwaite, and AN Ahmed should be prioritized as they provide strong batting and bowling contributions, making them versatile assets for the team.

##### 2. Balance Batting and Bowling:

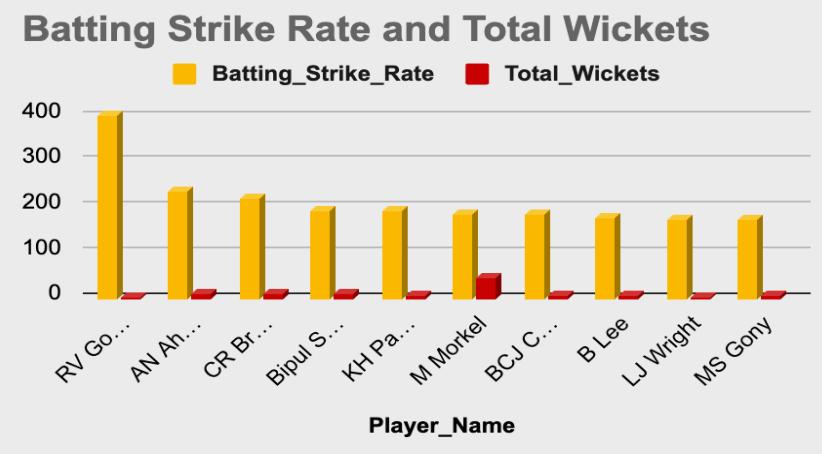
- Players like Bipul Sharma and BCJ Cutting offer balance, contributing significantly in both batting and bowling departments. These players

should be considered for a well-rounded team.

### 3. Consider Bowler Efficiency:

- Ensure that players with good economy rates (like CR Brathwaite and KH Pandya) are selected, as they restrict runs while taking wickets.

|   | Player_Id | Player_Name   | Total_Runs | Batting_Strike_Rate | Balls_Faced | Total_Wickets | Economy_Rate |
|---|-----------|---------------|------------|---------------------|-------------|---------------|--------------|
| ▶ | 259       | RV Gomez      | 4          | 400.00              | 1           | 1             | 1.00         |
|   | 246       | AN Ahmed      | 14         | 233.33              | 6           | 8             | 4.88         |
|   | 408       | CR Brathwaite | 83         | 218.42              | 38          | 9             | 4.89         |
|   | 228       | Bipul Sharma  | 52         | 192.59              | 27          | 10            | 6.30         |
|   | 413       | KH Pandya     | 237        | 191.13              | 124         | 7             | 5.29         |
|   | 190       | M Morkel      | 66         | 183.33              | 36          | 43            | 6.00         |
|   | 385       | BCJ Cutting   | 73         | 182.50              | 40          | 7             | 6.14         |
|   | 66        | B Lee         | 21         | 175.00              | 12          | 4             | 5.75         |
|   | 358       | LJ Wright     | 106        | 173.77              | 61          | 2             | 2.50         |
|   | 120       | MS Gony       | 45         | 173.08              | 26          | 7             | 6.29         |



### 5. Are there players whose presence positively influences the morale and performance of the team? (justify your answer using visualization)

- Code:

```
WITH cte AS (
```

```
-- Extract relevant match details for the 2015 and 2016 seasons
```

```
SELECT bbb.Striker, m.Season_Id, s.Season_Year,
```

```
bbb.Match_Id, bbb.Over_Id, bbb.Ball_Id,
```

```
bbb.Innings_No, bbb.Runs_Scored
```

```
FROM ball_by_ball bbb
```

```
JOIN matches m
```

```
ON bbb.Match_Id = m.Match_Id
```

```
JOIN season s
```

```
ON m.Season_Id = s.Season_Id
```

```
WHERE s.Season_Year IN (2015, 2016)
```

```
),
```

```
cte2 AS (
```

```
-- Calculate total runs per player
```

```
SELECT Striker, SUM(Runs_Scored) AS Total_Runs
```

```
FROM cte
```

```
GROUP BY Striker
```

```
),
```

```

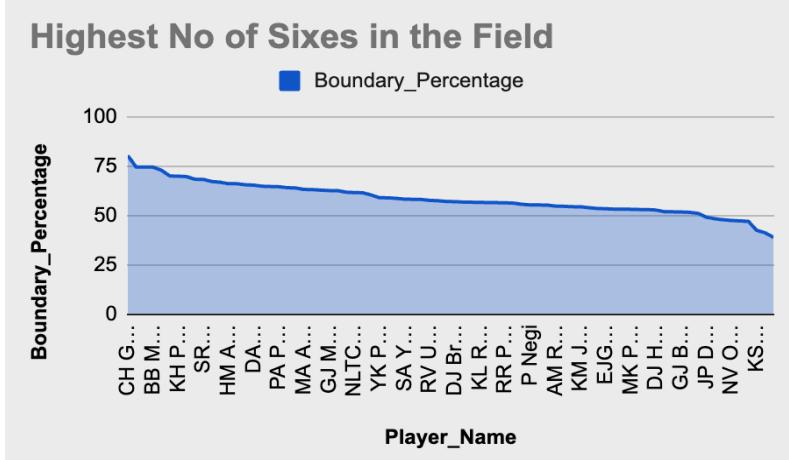
cte3 AS (
 -- Calculate runs from boundaries (4s and 6s) per player
 SELECT Striker, SUM(Runs_Scored) AS Runs_In_Boundaries
 FROM cte
 WHERE Runs_Scored IN (4, 6)
 GROUP BY Striker
)

-- Final output with boundary percentage calculation
SELECT c2.Striker AS Player_Id, p.Player_Name,
 c2.Total_Runs, c3.Runs_In_Boundaries,
 ROUND((c3.Runs_In_Boundaries * 100.0 / c2.Total_Runs), 2) AS
Boundary_Percentage
FROM cte2 c2
JOIN cte3 c3 ON c2.Striker = c3.Striker
JOIN player p ON c2.Striker = p.Player_Id
WHERE c2.Total_Runs >= 100
ORDER BY Boundary_Percentage DESC;

```

- This code calculates the percentage of runs scored from boundaries (fours and sixes) for players in the 2015 and 2016 IPL seasons, and filters players with a total run count of 100 or more.
- It analyzes the performance of batsmen in the 2015 and 2016 IPL seasons, focusing on the percentage of their runs scored from boundaries (fours and sixes).
- Then it calculates the total runs and boundary runs for each player, filters for those who have scored 100 or more total runs, and orders the results by the percentage of runs from boundaries, in descending order.
- The output includes the player ID, player name, total runs, runs from boundaries, and the boundary percentage

| Player_Id | Player_Name     | Total_Runs | Runs_In_Boundaries | Boundary_Percentage |
|-----------|-----------------|------------|--------------------|---------------------|
| 162       | CH Gayle        | 718        | 578                | 80.50               |
| 334       | AD Russell      | 514        | 384                | 74.71               |
| 147       | DR Smith        | 723        | 540                | 74.69               |
| 2         | BB McCullum     | 790        | 590                | 74.68               |
| 431       | N Rana          | 104        | 76                 | 73.08               |
| 372       | CJ Anderson     | 114        | 80                 | 70.18               |
| 413       | KH Pandya       | 237        | 166                | 70.04               |
| 50        | Harbhajan Singh | 209        | 146                | 69.86               |
| 110       | AB de Villiers  | 1200       | 822                | 68.50               |
| 32        | SR Watson       | 526        | 360                | 68.44               |
| 355       | Q de Kock       | 567        | 382                | 67.37               |
| 221       | KA Pollard      | 627        | 420                | 66.99               |
| 427       | HM Amla         | 157        | 104                | 66.24               |
| 236       | Mandeep Singh   | 157        | 104                | 66.24               |
| 27        | Yuvraj Singh    | 484        | 318                | 65.70               |
| 187       | DA Warner       | 1410       | 924                | 65.53               |
| 383       | LMP Simmons     | 548        | 356                | 64.96               |
| 393       | SS Iyer         | 469        | 304                | 64.62               |
| 17        | PA Patel        | 516        | 334                | 64.73               |
| 350       | CH Morris       | 271        | 174                | 64.21               |
| 400       | HH Pandya       | 156        | 100                | 64.10               |
| 260       | MA Agarwal      | 240        | 152                | 63.33               |
| 402       | SN Khan         | 177        | 112                | 63.28               |
| 254       | AJ Finch        | 416        | 262                | 62.98               |



6. What would you suggest to RCB before going to the mega auction?

- Code:

```
Identify good all-rounders for better team combinations.
WITH batting_performance AS (
 SELECT p.Player_Id, p.Player_Name,
 SUM(bb.Runs_Scored) AS Total_Runs,
 COUNT(bb.Ball_Id) AS Balls_Faced,
 ROUND((SUM(bb.Runs_Scored) / COUNT(bb.Ball_Id)) * 100, 2) AS
 Batting_Strike_Rate
 FROM player p
 JOIN ball_by_ball bb
 ON p.Player_Id = bb.Striker
 JOIN matches m
 ON bb.Match_Id = m.Match_Id
 JOIN ball_by_ball b
 ON bb.Match_Id = b.Match_Id
 AND bb.Over_Id = b.Over_Id
 AND bb.Ball_Id = b.Ball_Id
 AND bb.Innings_No = b.Innings_No
 WHERE bb.Runs_Scored IS NOT NULL
 GROUP BY p.Player_Id, p.Player_Name
,

bowling_performance AS (
 SELECT p.Player_Id, p.Player_Name,
 COUNT(w.Player_Out) AS Total_Wickets,
 ROUND(SUM(bb.Runs_Scored) / (COUNT(bb.Ball_Id) / 6.0), 2) AS
 Economy_Rate
 FROM player p
 JOIN ball_by_ball bb ON p.Player_Id = bb.Bowler
 LEFT JOIN wicket_taken w
 ON bb.Match_Id = w.Match_Id
 AND bb.Over_Id = w.Over_Id
 AND bb.Ball_Id = w.Ball_Id
 AND bb.Innings_No = w.Innings_No
 JOIN ball_by_ball bs
 ON bs.Match_Id = bb.Match_Id
 AND bs.Over_Id = bb.Over_Id
 AND bs.Ball_Id = bb.Ball_Id
 AND bs.Innings_No = bb.Innings_No
 GROUP BY p.Player_Id, p.Player_Name
 HAVING COUNT(bb.Ball_Id) > 100
)

SELECT DISTINCT bp.Player_Id, bp.Player_Name,
 bp.Total_Runs, bp.Batting_Strike_Rate, bp.Balls_Faced,
 bw.Total_Wickets, bw.Economy_Rate
FROM batting_performance bp
JOIN bowling_performance bw ON bp.Player_Id = bw.Player_Id
JOIN player_match pm ON bp.Player_Id = pm.Player_Id
WHERE pm.Role_Id NOT IN (SELECT Role_Id FROM rolee WHERE Role_Desc
```

```

IN ("Keeper", "CaptainKeeper")
AND bp.Balls_Faced > 100
ORDER BY bp.Batting_Strike_Rate DESC, bw.Economy_Rate ASC
LIMIT 10;

```

- Before heading into the mega auction, RCB needs to focus on building a well-balanced squad that covers both batting and bowling strengths. This query helps in identifying top-performing all-rounders who can contribute in both departments.
- First, the query examines batting performance, filtering out players who have faced at least 100 balls and calculating their total runs and batting strike rate. This ensures only consistent batsmen are considered.
- Then, it analyzes bowling performance, calculating the total wickets taken and the economy rate of each bowler. Only bowlers who have bowled more than 100 balls are included to avoid small sample sizes.
- After identifying strong batters and bowlers, the query finds players who excel in both aspects, ensuring RCB targets genuine all-rounders in the auction. It also excludes wicket keepers and captains, as the focus is on players who contribute as both batsmen and bowlers.
- The final list prioritizes players with a high batting strike rate and a low economy rate, ensuring they can score quickly while also being effective with the ball.

- **Recommendation:**

- Invest in all-rounders who consistently perform in both batting and bowling.
- Prioritize power-hitters with a high strike rate and bowlers who keep the economy rate low.
- Avoid relying too much on specialist batsmen or bowlers; instead, target multi-dimensional players who provide flexibility in team selection.

|   | Player_Id | Player_Name    | Total_Runs | Batting_Strike_Rate | Balls_Faced | Total_Wickets | Economy_Rate |
|---|-----------|----------------|------------|---------------------|-------------|---------------|--------------|
| ▶ | 413       | KH Pandya      | 237        | 191.13              | 124         | 7             | 6.96         |
|   | 334       | AD Russell     | 527        | 172.22              | 306         | 35            | 7.71         |
|   | 350       | CH Morris      | 285        | 166.67              | 171         | 48            | 6.98         |
|   | 305       | GJ Maxwell     | 913        | 163.62              | 558         | 3             | 9.63         |
|   | 325       | A Ashish Reddy | 245        | 149.39              | 164         | 8             | 9.33         |
|   | 221       | KA Pollard     | 1320       | 148.65              | 888         | 19            | 9.20         |
|   | 280       | NLTC Perera    | 397        | 144.36              | 275         | 32            | 8.32         |
|   | 31        | YK Pathan      | 1273       | 143.52              | 887         | 4             | 7.92         |
|   | 310       | JP Faulkner    | 459        | 143.44              | 320         | 61            | 7.89         |
|   | 162       | CH Gayle       | 1634       | 143.21              | 1141        | 5             | 6.41         |

7. What do you think could be the factors contributing to the high-scoring matches and the impact on viewership and team strategies.

- Code:

```

/* Powerplay and Death Over Utilization: In high-scoring matches, teams aim to
maximize the power play (overs 1-6) and death overs (Overs 16-20) by scoring
aggressively. */

```

```

SELECT t.Team_Name,

```

```

SUM(CASE WHEN bb.Over_Id BETWEEN 1 AND 6 THEN bb.Runs_Scored
ELSE 0 END) AS Powerplay_Runs,
```

```

 SUM(CASE WHEN bb.Over_Id BETWEEN 16 AND 20 THEN bb.Runs_Scored
ELSE 0 END) AS Death_Over_Runs
FROM team t
JOIN matches m ON t.Team_Id = m.Team_1 OR t.Team_Id = m.Team_2
JOIN ball_by_ball bb ON m.Match_Id = bb.Match_Id
GROUP BY t.Team_Name
ORDER BY Powerplay_Runs DESC, Death_Over_Runs DESC;

```

```

/* High Scoring Venues: Some venues favour the batsmen more than others,
venues play a significant role in a high-scoring match */
SELECT v.Venue_Name,
 AVG(match_runs.Total_Runs) AS Avg_Runs_Per_Match,
 COUNT(m.Match_Id) AS Total_Matches
FROM venue v
JOIN matches m ON v.Venue_Id = m.Venue_Id
JOIN (
 SELECT bb.Match_Id, SUM(bb.Runs_Scored) AS Total_Runs
 FROM ball_by_ball bb
 GROUP BY bb.Match_Id
) AS match_runs ON m.Match_Id = match_runs.Match_Id
GROUP BY v.Venue_Name
ORDER BY Total_Matches DESC, Avg_Runs_Per_Match DESC
LIMIT 10;

```

- In high-scoring matches, there are several critical factors that contribute to a higher run rate. These factors impact not only the match outcome but also the viewership and team strategies.

#### 1. Powerplay and Death Over Utilization:

- The first factor contributing to high-scoring matches is how well teams utilize the power play (overs 1-6) and death overs (overs 16-20). Teams that maximize scoring in these overs by taking risks and being aggressive can post higher totals.
- The query calculates how many runs each team scores during the powerplay and death overs, helping us understand the aggressive approach teams adopt in these periods. Teams that score heavily in powerplay and death overs often push the total to higher levels.

#### 2. High-Scoring Venues:

- Some venues tend to favour batsmen more than others due to factors like pitch conditions, weather, and altitude. These venues naturally lead to high-scoring matches.
- The second query identifies high-scoring venues by calculating the average runs per match at each venue. This allows us to see which venues tend to produce higher scores, which could impact team strategy (e.g., choosing to bowl first or setting aggressive targets).
- Understanding these venues helps teams tailor their strategies to the conditions and even influences viewership, as fans love watching high-scoring, exciting matches.

- Impact on Viewership and Team Strategies:

#### 1. Viewership:

- High-scoring matches are more entertaining and exciting to watch, which attracts more fans. These matches, especially in the powerplay and death overs, are full of big shots and intense moments, which is exactly what viewers enjoy. This contributes to higher ratings and fan engagement.

## 2. Team Strategies:

- In high-scoring matches, teams may focus on aggressive batting strategies, maximizing runs in the powerplay and death overs.
- Teams may opt for fast bowlers who can bowl at the death, or spin bowlers who can restrict runs in the middle overs.
- Venue-specific strategies come into play, where teams choose to bowl or bat first depending on whether the venue is historically batsman-friendly or bowler-friendly.

**Result Grid**

| Team_Name                   | Powerplay_Runs | Death_Over_Runs |
|-----------------------------|----------------|-----------------|
| Kings XI Punjab             | 5429           | 4930            |
| Royal Challengers Bangalore | 5275           | 5115            |
| Mumbai Indians              | 5258           | 5503            |
| Sunrisers Hyderabad         | 5134           | 4856            |
| Kolkata Knight Riders       | 5000           | 4624            |
| Delhi Daredevils            | 4650           | 4574            |
| Chennai Super Kings         | 4421           | 4253            |
| Rajasthan Royals            | 3677           | 3849            |
| Gujarat Lions               | 1431           | 1290            |
| Pune Warriors               | 1263           | 1346            |
| Rising Pune Supergiants     | 1230           | 1086            |

**Venue Statistics**

| Venue_Name                                                         | Avg_Runs_Per_Match | Total_Matches |
|--------------------------------------------------------------------|--------------------|---------------|
| M Chinnaswamy Stadium                                              | 310.7667           | 30            |
| Wankhede Stadium                                                   | 318.0000           | 26            |
| Eden Gardens                                                       | 293.8462           | 26            |
| Feroz Shah Kotla                                                   | 284.0000           | 25            |
| Rajiv Gandhi International Stadium, Uppal                          | 273.0435           | 23            |
| MA Chidambaram Stadium, Chepauk                                    | 302.1333           | 15            |
| Punjab Cricket Association Stadium, Mohali                         | 280.6923           | 13            |
| Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Stadium, Visakhapatnam | 269.6667           | 9             |
| Maharashtra Cricket Association Stadium                            | 310.3750           | 8             |
| Sardar Patel Stadium, Motera                                       | 303.2500           | 8             |

## 8. Analyze the impact of home-ground advantage on team performance and identify strategies to maximize this advantage for RCB.

- Code:

```
Home vs Away Win/Loss record
WITH win_loss_record AS (
 SELECT m.Match_Id, v.Venue_Name,
 CASE WHEN m.Match_Winner = 2 THEN 'Win' ELSE 'Loss'
 END AS Result,
 CASE WHEN v.Venue_Id = 1 THEN 'Home' ELSE 'Away'
 END AS Venue_Type
 FROM matches m
 JOIN venue v ON m.Venue_Id = v.Venue_Id
 WHERE (m.Team_1 = 2 OR m.Team_2 = 2) AND Outcome_type != 2
)
SELECT
Venue_Type,
COUNT(CASE WHEN Result = 'Win' THEN 1 END) AS Wins,
COUNT(CASE WHEN Result = 'Loss' THEN 1 END) AS Losses,
COUNT(*) AS Total_Matches,
ROUND(COUNT(CASE WHEN Result = 'Win' THEN 1 END) /
COUNT(*)) * 100, 2) AS Win_Percentage
FROM win_loss_record
```

```

 GROUP BY Venue_Type;

#Home away batting performance

WITH rcb_run_stats AS (
 SELECT m.Match_Id, v.Venue_Name,
 CASE WHEN v.Venue_Id = 1 THEN 'Home' ELSE 'Away' END AS
Venue_Type,
 SUM(CASE WHEN bb.Team_Batting = 2 THEN bb.Runs_Scored ELSE 0
END) AS Runs_Scored,
 SUM(CASE WHEN bb.Team_Bowling = 2 THEN bb.Runs_Scored ELSE 0
END) AS Runs_Conceded
 FROM matches m
 JOIN venue v ON m.Venue_Id = v.Venue_Id
 JOIN ball_by_ball bb ON m.Match_Id = bb.Match_Id
 WHERE (m.Team_1 = 2 OR m.Team_2 = 2) -- 2 is the team ID for RCB
 GROUP BY m.Match_Id, v.Venue_Name
)
SELECT Venue_Type,
 ROUND(AVG(Runs_Scored), 2) AS Avg_Runs_Scored,
 ROUND(SUM(Runs_Scored), 2) AS Total_Runs_Scored
FROM rcb_run_stats
GROUP BY Venue_Type;

WITH bowling_performance AS (
 SELECT v.Venue_Name,
 CASE WHEN v.Venue_Id = 1 THEN 'Home' ELSE 'Away' END AS
Venue_Type,
 SUM(CASE WHEN bb.Team_Bowling = 2 THEN bb.Runs_Scored ELSE 0
END) AS Runs_Conceded,
 COUNT(CASE WHEN bb.Team_Bowling = 2 AND w.Player_Out IS NOT
NULL THEN 1 ELSE NULL END) AS Wickets_Taken,
 COUNT(CASE WHEN bb.Team_Bowling = 2 THEN bb.Ball_Id ELSE NULL
END) AS Balls_Bowled
 FROM matches m
 JOIN venue v ON m.Venue_Id = v.Venue_Id
 JOIN ball_by_ball bb ON m.Match_Id = bb.Match_Id
 LEFT JOIN wicket_taken w ON bb.Match_Id = w.Match_Id
 AND bb.Over_Id = w.Over_Id
 AND bb.Ball_Id = w.Ball_Id
 AND bb.Innings_No = w.Innings_No -- Ensuring correct innings mapping
 WHERE (m.Team_1 = 2 OR m.Team_2 = 2) -- 2 is the team ID for RCB
 GROUP BY v.Venue_Name, Venue_Type
)
SELECT Venue_Type,
 ROUND(SUM(Wickets_taken), 2) AS Total_Wickets_taken,
 ROUND(SUM(Runs_Conceded) / SUM(Balls_Bowled), 2) AS Economy_Rate
FROM bowling_performance
GROUP BY Venue_Type;

```

The image displays three separate Result Grid tables, each showing data for Home and Away matches. The first table shows Wins, Losses, Total Matches, and Win Percentage. The second table shows Average Runs Scored and Total Runs Scored. The third table shows Total Wickets Taken and Economy Rate.

| Venue_Type | Wins | Losses | Total_Matches | Win_Percentage |
|------------|------|--------|---------------|----------------|
| Home       | 16   | 11     | 27            | 59.26          |
| Away       | 15   | 18     | 33            | 45.45          |

| Venue_Type | Avg_Runs_Scored | Total_Runs_Scored |
|------------|-----------------|-------------------|
| Home       | 163.24          | 4734              |
| Away       | 144.45          | 4767              |

| Venue_Type | Total_Wickets_taken | Economy_Rate |
|------------|---------------------|--------------|
| Home       | 178                 | 1.30         |
| Away       | 202                 | 1.24         |

9. Come up with a visual and analytical analysis of the RCB's past season's performance and potential reasons for them not winning a trophy.

- Code:

#### # A. Win-Loss Performance Over Seasons

```
WITH win_loss_record AS (
 SELECT m.Season_Id,CASE WHEN m.Match_Winner = 2 THEN 'Win' ELSE
 'Loss' END AS Result
 FROM matches m
 WHERE (m.Team_1 = 2 OR m.Team_2 = 2) AND Outcome_type != 2
)
SELECT Season_Id,COUNT(CASE WHEN Result = 'Win' THEN 1 END) AS Wins,
 COUNT(CASE WHEN Result = 'Loss' THEN 1 END) AS Losses,
 COUNT(*) AS Total_Matches,
 ROUND(COUNT(CASE WHEN Result = 'Win' THEN 1 END) / COUNT(*) * 100, 2)
AS Win_Percentage
FROM win_loss_record
GROUP BY Season_Id
ORDER BY Season_Id;
```

#### # B. Batting performance each season

```
WITH rcb_batting_in_powerplay AS (
 SELECT bb.Match_Id, bb.Innings_No, bb.Striker AS Batsman_Id,
 p.Player_Name,
 SUM(bb.Runs_Scored) AS total_runs_in_power_play,
 COUNT(bb.Ball_Id) AS balls_faced_in_power_play
 FROM Ball_by_Ball bb
 JOIN Matches m ON bb.Match_Id = m.Match_Id
 JOIN Player p ON bb.Striker = p.Player_Id
 WHERE (m.Team_1 = 2 OR m.Team_2 = 2)
```

```

 AND bb.Over_Id BETWEEN 1 AND 6
 GROUP BY bb.Match_Id, bb.Innings_No, bb.Striker, p.Player_Name
)
 SELECT rcb.Player_Name,
 SUM(rcb.total_runs_in_power_play) AS total_runs_in_power_play,
 SUM(rcb.balls_faced_in_power_play) AS total_balls_faced_in_powerplay,
 ROUND((SUM(rcb.total_runs_in_power_play) /
 NULLIF(SUM(rcb.balls_faced_in_power_play), 0)) * 100, 2) AS
 strike_rate_in_power_play
 FROM rcb_batting_in_powerplay rcb
 GROUP BY rcb.Player_Name
 HAVING total_balls_faced_in_powerplay > 100
 ORDER BY strike_rate_in_power_play DESC;

C. Bowling performance each season
WITH death_overs_bowling AS (
 SELECT bb.Match_Id, bb.Innings_No, bb.Bowler AS Bowler_Id,
 p.Player_Name,
 SUM(bb.Runs_Scored) AS runs_conceded,
 COUNT(bb.Ball_Id) AS balls_bowled,
 COUNT(w.Player_Out) AS wickets_taken
 FROM Ball_by_Ball bb
 LEFT JOIN Wicket_Taken w ON bb.Match_Id = w.Match_Id
 AND bb.Over_Id = w.Over_Id
 AND bb.Ball_Id = w.Ball_Id
 AND bb.Innings_No = w.Innings_No
 JOIN Player p ON bb.Bowler = p.Player_Id
 JOIN Matches m ON bb.Match_Id = m.Match_Id
 WHERE (m.Team_1 = 2 OR m.Team_2 = 2)
 AND bb.Over_Id BETWEEN 16 AND 20
 GROUP BY bb.Match_Id, bb.Innings_No, bb.Bowler, p.Player_Name
)
SELECT d.Player_Name,
 SUM(d.runs_conceded) AS runs_conceded_in_death,
 SUM(d.balls_bowled) AS total_balls_bowled_in_death,
 SUM(d.wickets_taken) AS total_wickets_in_death,
 ROUND((SUM(d.runs_conceded) / NULLIF((SUM(d.balls_bowled) / 6), 0)), 2)
 AS economy_rate_in_death
FROM death_overs_bowling d
GROUP BY d.Player_Name
HAVING total_balls_bowled_in_death > 100
ORDER BY economy_rate_in_death ASC;

```

- The query focuses on analyzing RCB's performance over the past seasons, breaking down the performance in key areas: Win-Loss record, batting performance in powerplay, and bowling performance in death overs. Here's a breakdown:

### 1. Win-Loss Performance Over Seasons

- This query calculates the win percentage of RCB in each season. It looks at the number of wins and losses for RCB, specifically analyzing seasons where RCB participated (Team 1 or Team 2).
- By calculating the win percentage, we can understand how often RCB won in each season. A low win percentage may indicate

inconsistencies or poor match strategies, contributing to the team's failure to win a trophy.

## 2. Batting Performance in Powerplay

- This query focuses on RCB's batting performance in the powerplay (overs 1-6). The strike rate during powerplay is calculated for each batsman who faced over 100 balls during the season.
- A high strike rate in powerplay is essential to get the team off to a strong start. If RCB's batsmen have low strike rates, it might suggest that the team struggled to build momentum in the early overs, reducing the chances of a high total.

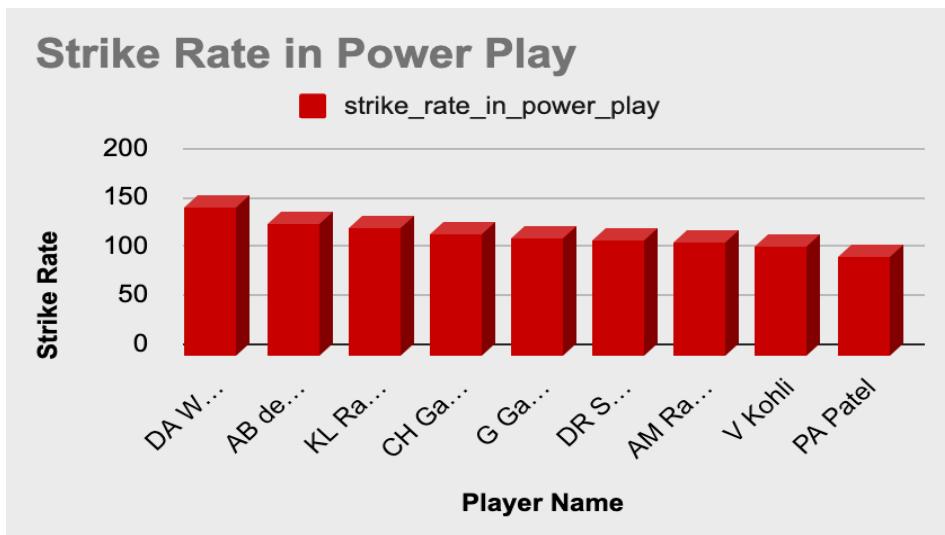
## 3. Bowling Performance in Death Overs:

- This query analyzes RCB's bowling performance in death overs (overs 16-20). It calculates the economy rate and wickets taken by each bowler during the death overs.
- A high economy rate (too many runs conceded) or low wickets taken during death overs could suggest that RCB's bowlers struggled to contain the opposition or take crucial wickets in the final overs, contributing to poor match outcomes.

| Result Grid |           | Filter Rows: |        |               | Export:        |
|-------------|-----------|--------------|--------|---------------|----------------|
|             | Season_Id | Wins         | Losses | Total_Matches | Win_Percentage |
| ▶           | 6         | 9            | 7      | 16            | 56.25          |
|             | 7         | 5            | 9      | 14            | 35.71          |
|             | 8         | 8            | 6      | 14            | 57.14          |
|             | 9         | 9            | 7      | 16            | 56.25          |

| Result Grid |                | Filter Rows:             |                                |                           | Export:            |
|-------------|----------------|--------------------------|--------------------------------|---------------------------|--------------------|
|             | Player_Name    | total_runs_in_power_play | total_balls_faced_in_powerplay | strike_rate_in_power_play | Wrap Cell Content: |
| ▶           | DA Warner      | 220                      | 144                            | 152.78                    |                    |
|             | AB de Villiers | 272                      | 202                            | 134.65                    |                    |
|             | KL Rahul       | 181                      | 137                            | 132.12                    |                    |
|             | CH Gayle       | 886                      | 712                            | 124.44                    |                    |
|             | G Gambhir      | 132                      | 109                            | 121.10                    |                    |
|             | DR Smith       | 147                      | 123                            | 119.51                    |                    |
|             | AM Rahane      | 141                      | 122                            | 115.57                    |                    |
|             | V Kohli        | 681                      | 607                            | 112.19                    |                    |
|             | PA Patel       | 184                      | 181                            | 101.66                    |                    |

|   | Player_Name   | runs_conceded_in_death | total_balls_bowled_in_death | total_wickets_in_death | economy_rate_in_death |
|---|---------------|------------------------|-----------------------------|------------------------|-----------------------|
| ▶ | MA Starc      | 247                    | 210                         | 22                     | 7.06                  |
|   | RP Singh      | 152                    | 113                         | 13                     | 8.07                  |
|   | R Vinay Kumar | 233                    | 163                         | 17                     | 8.58                  |
|   | YS Chahal     | 192                    | 128                         | 11                     | 9.00                  |
|   | SR Watson     | 279                    | 184                         | 20                     | 9.10                  |
|   | HV Patel      | 179                    | 105                         | 9                      | 10.23                 |



**10. How would you approach this problem, if the objective and subjective questions weren't given?**

- If the objective and subjective questions weren't provided, I would approach the problem by first focusing on RCB's performance data and then analyzing key factors that could help improve their chances of winning in the future. Here's a step-by-step approach::

**1. Data Collection and Preparation:**

- Understand the available data by exploring the IPL match data for RCB, including details about team performance, individual player statistics, and match outcomes. This would involve reviewing tables like:
  - Matches (Match outcomes, toss decisions, team performance).
  - Player stats (Batting, bowling, strike rates, runs, wickets).
  - Venue-specific data (to understand performance at different locations).
  - Innings performance (Powerplay and death overs).

**2. Data Cleaning and Structuring:**

- Clean and structure the data to ensure that only relevant information is retained. For example:
  - Remove invalid or incomplete records.
  - Filter out inconsistent data for players who didn't play enough matches.
  - Ensure correct mapping between match IDs, player IDs, and performance statistics.

**3. Comparative Analysis:**

- Comparing RCB's performance with other teams: By comparing RCB's batting and bowling performance to other top-performing teams to identify gaps or areas for improvement.
- Venue Impact: Analyze whether RCB performs better or worse at specific venues. For example, do they have a better win percentage in high-scoring venues or are they struggling in bowler-friendly venues?

11. In the "Match" table, some entries in the "Opponent\_Team" column are incorrectly spelled as "Delhi\_Capitals" instead of "Delhi\_Daredevils". Write an SQL query to replace all occurrences of "Delhi\_Capitals" with "Delhi\_Daredevils".

- Code:

```
Select * from Team;
select Team_Id, replace(Team_name,"Delhi Daredevils","Delhi Capitals") as
Team_name from team;
```