IPL STRATEGY FOR PLAYER SELECTION

ANALYSING PLAYER PERFORMANCE FOR AUCTION

AYAN DAS

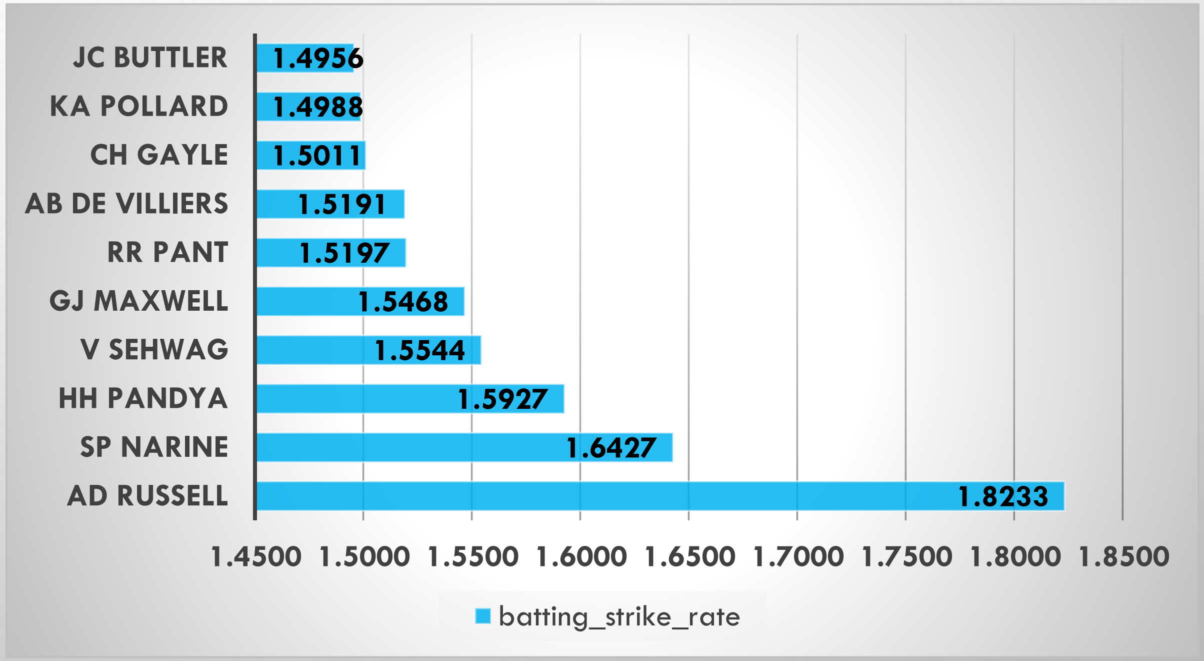
16-06-2024

Identifying Batsmen with High Strike Rates

QUERY:

```
SELECT
    BATSMAN,
    COUNT(BALL) AS BALLS_FACED,
    SUM(BATSMAN_RUNS) AS
TOTAL_RUNS_SCORED,
    (SUM(BATSMAN_RUNS) * 1.0) / COUNT(BALL)
AS BATTING_STRIKE_RATE
FROM DELIVERIES
WHERE EXTRAS_TYPE != 'WIDES'
GROUP BY BATSMAN
HAVING COUNT(BALL) >= 500
ORDER BY BATTING_STRIKE_RATE DESC
LIMIT 10;
```

batsman	balls_faced	total_runs_scored	batting_strike_rate
AD Russell	832	1517	1.8233
SP Narine	543	892	1.6427
HH Pandya	847	1349	1.5927
V Sehwag	1755	2728	1.5544
GJ Maxwell	973	1505	1.5468
RR Pant	1368	2079	1.5197
AB de Villiers	3192	4849	1.5191
CH Gayle	3179	4772	1.5011
KA Pollard	2017	3023	1.4988
JC Buttler	1146	1714	1.4956

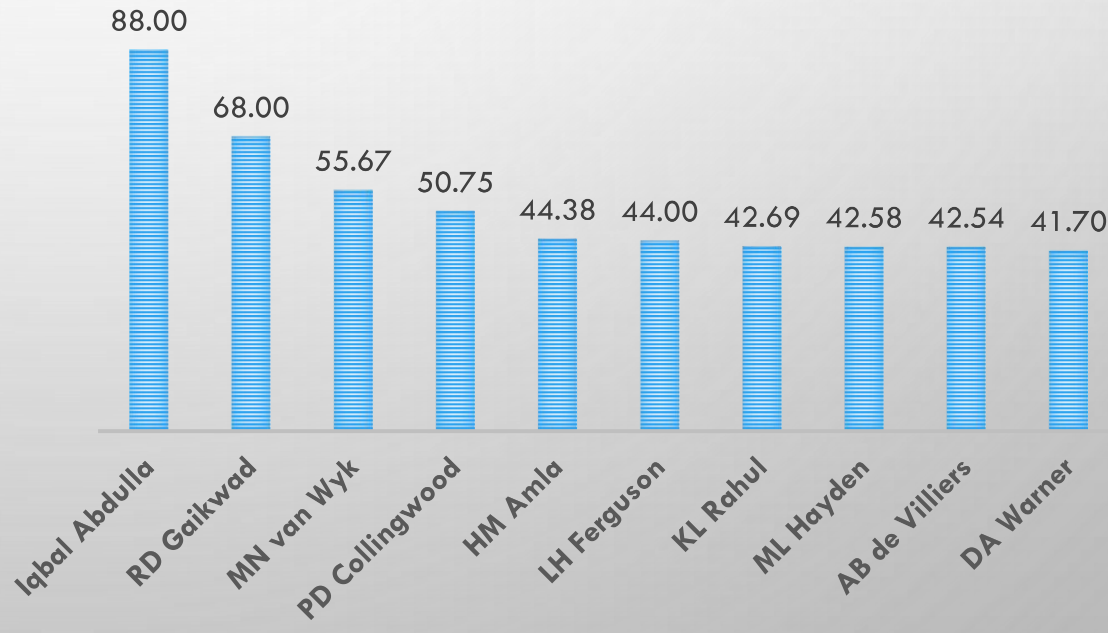


Identifying Batsmen with High Averages

QUERY:

```
SELECT
    BATSMAN,
    (SUM(BATSMAN_RUNS) * 1.0) /
SUM(IS_WICKET) AS AVERAGE_RUN
FROM DELIVERIES
WHERE EXTRAS_TYPE != 'WIDES'
GROUP BY BATSMAN
HAVING COUNT(ID) > 28 AND SUM(IS_WICKET) !=
0
ORDER BY AVERAGE_RUN DESC
LIMIT 10;
```

batsman	average_run
Iqbal Abdulla	88.00
RD Gaikwad	68.00
MN van Wyk	55.67
PD Collingwood	50.75
HM Amla	44.38
LH Ferguson	44.00
KL Rahul	42.69
ML Hayden	42.58
AB de Villiers	42.54
DA Warner	41.70

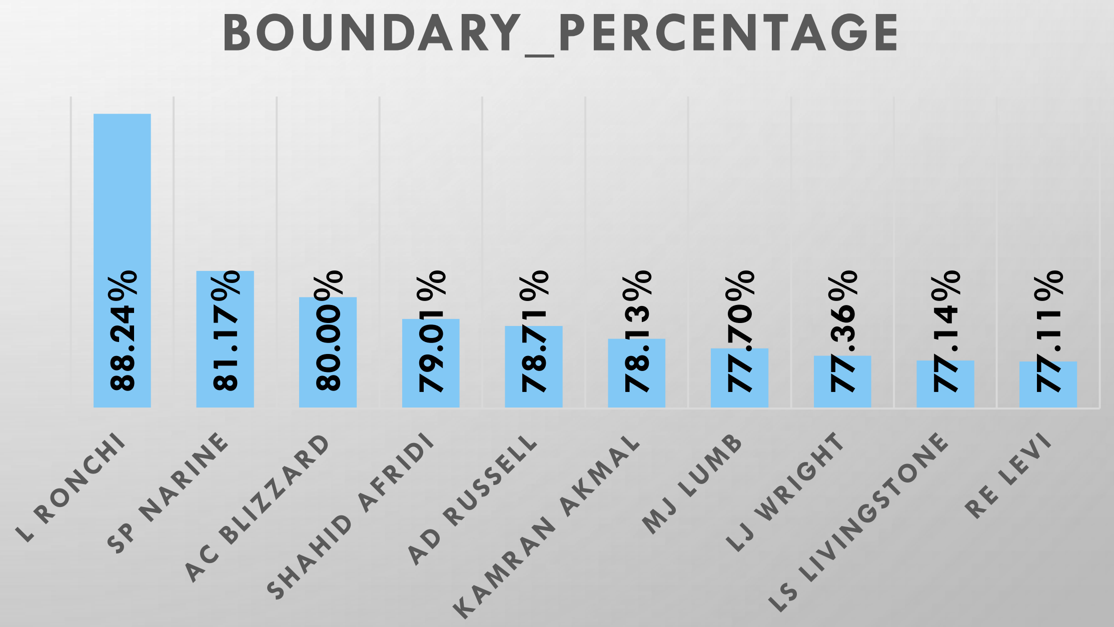


Hard-Hitting Batsmen

QUERY:

```
SELECT
   BATSMAN,
   COUNT(CASE WHEN BATSMAN_RUNS IN (4, 6) THEN 1
END) AS BOUNDARY_COUNT,
   SUM(CASE WHEN BATSMAN_RUNS IN (4, 6) THEN
BATSMAN_RUNS ELSE 0 END) AS BOUNDARY_RUNS,
   SUM(CASE WHEN BATSMAN_RUNS IN (4, 6) THEN
BATSMAN_RUNS ELSE 0 END) * 1.0 /
SUM(BATSMAN_RUNS) AS BOUNDARY_PERCENTAGE
FROM DELIVERIES
WHERE EXTRAS_TYPE != 'WIDES'
GROUP BY BATSMAN
HAVING COUNT(ID) > 28
ORDER BY BOUNDARY_PERCENTAGE DESC
LIMIT 10;
```

batsman	boundary_count	boundary_runs	boundary_percentage
L Ronchi	7	30	88.24%
SP Narine	155	724	81.17%
AC Blizzard	23	96	80.00%
Shahid Afridi	13	64	79.01%
AD Russell	234	1194	78.71%
Kamran Akmal	21	100	78.13%
MJ Lumb	51	216	77.70%
LJ Wright	19	82	77.36%
LS Livingstone	11	54	77.14%
RE Levi	14	64	77.11%

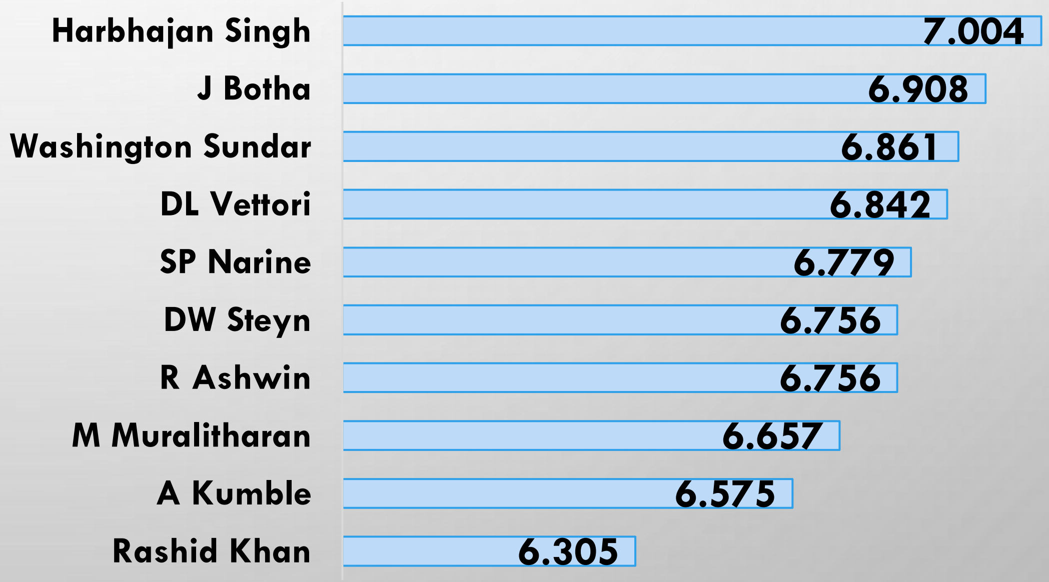


Bowlers with Good Economy Rates

QUERY:

```
SELECT
    BOWLER,
    SUM(TOTAL_RUNS) / (COUNT(*) * 1.0 / 6) AS ECONOMY
FROM DELIVERIES
WHERE EXTRAS_TYPE != 'WIDES'
GROUP BY BOWLER
HAVING COUNT(*) > 500
ORDER BY ECONOMY
LIMIT 10;
```

bowler	economy
Rashid Khan	6.305
A Kumble	6.575
M Muralitharan	6.657
R Ashwin	6.756
DW Steyn	6.756
SP Narine	6.779
DL Vettori	6.842
Washington Sundar	6.861
J Botha	6.908
Harbhajan Singh	7.004

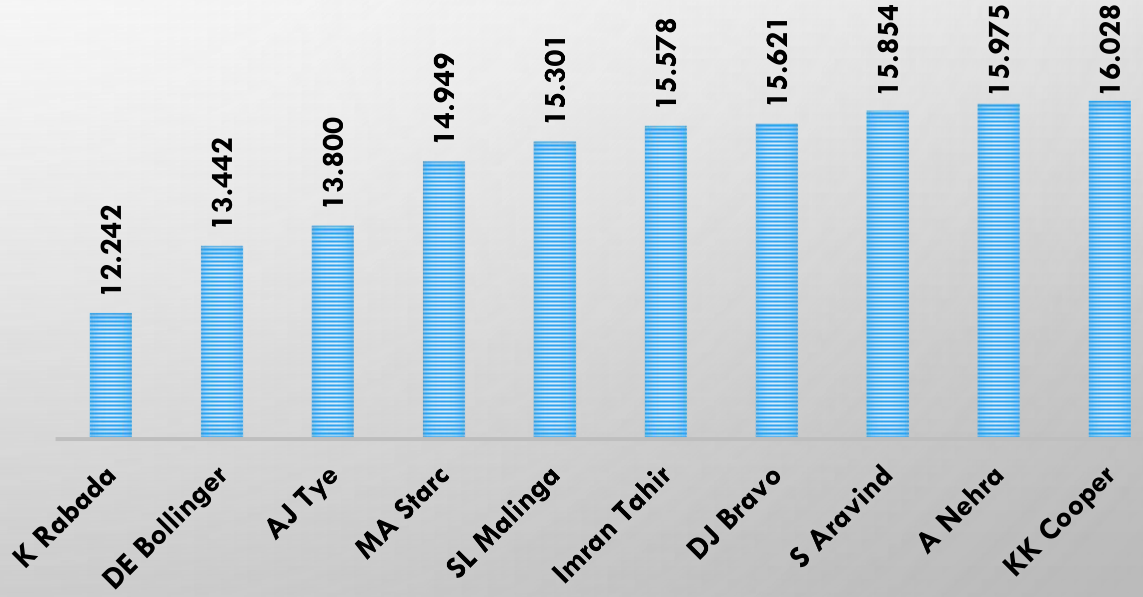


Identifying Bowlers with Best Strike Rates

QUERY:

```
SELECT
   BOWLER,
   COUNT(*) * 1.0 / SUM(IS_WICKET) AS
BOWLING_STRIKE_RATE
FROM DELIVERIES
WHERE EXTRAS_TYPE != 'WIDES'
GROUP BY BOWLER
HAVING COUNT(*) > 500 AND SUM(IS_WICKET) > 0
ORDER BY BOWLING_STRIKE_RATE
LIMIT 10;
```

bowler	bowling_strike_rate
K Rabada	12.242
DE Bollinger	13.442
AJ Tye	13.800
MA Starc	14.949
SL Malinga	15.301
Imran Tahir	15.578
DJ Bravo	15.621
S Aravind	15.854
A Nehra	15.975
KK Cooper	16.028



Identifying Top All-Rounders

QUERY:

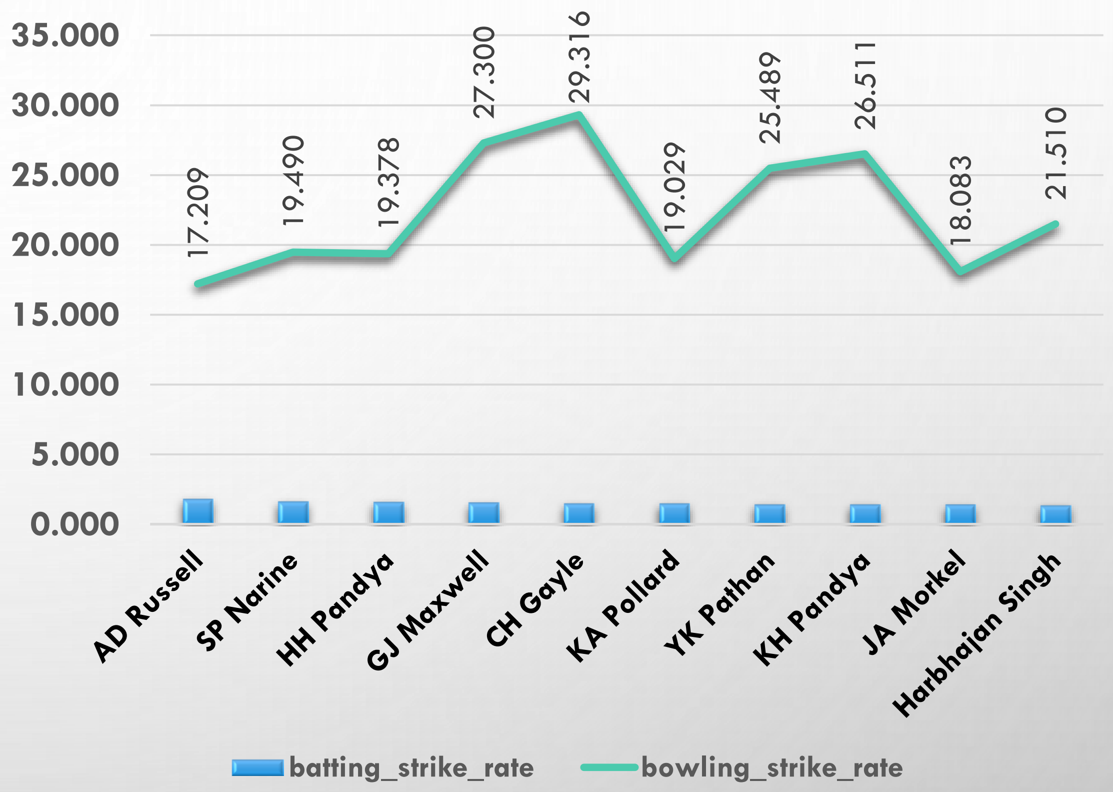
```
WITH BATTING_STATS AS (
    SELECT
        BATSMAN AS PLAYER,
        SUM(BATSMAN_RUNS) AS TOTAL_RUNS,
        COUNT(*) AS BALLS_FACED,
        (SUM(BATSMAN_RUNS) * 1.0 / COUNT(*)) AS
BATTING_STRIKE_RATE
    FROM DELIVERIES
    WHERE EXTRAS_TYPE != 'WIDES'
    GROUP BY BATSMAN
    HAVING COUNT(*) >= 500),
BOWLING_STATS AS (
    SELECT
        BOWLER AS PLAYER,
        COUNT(*) AS BALLS_BOWLED,
        SUM(IS_WICKET) AS WICKETS,
        (COUNT(*) * 1.0 / SUM(IS_WICKET)) AS
BOWLING_STRIKE_RATE
```

Query:

```
 FROM Deliveries
    WHERE extras_type != 'wides'
    GROUP BY bowler
    HAVING COUNT(*) >= 300 AND SUM(is_wicket) > 0)
SELECT
    b.player,
    b.batting_strike_rate,
    bw.bowling_strike_rate
FROM batting_stats b
JOIN bowling_stats bw ON b.player = bw.player
ORDER BY
    b.batting_strike_rate DESC,
    bw.bowling_strike_rate ASC
LIMIT 10;
```

Identifying Top All-Rounders

player	batting_strike_rate	bowling_strike_rate
AD Russell	1.823	17.209
SP Narine	1.643	19.490
HH Pandya	1.593	19.378
GJ Maxwell	1.547	27.300
CH Gayle	1.501	29.316
KA Pollard	1.499	19.029
YK Pathan	1.430	25.489
KH Pandya	1.425	26.511
JA Morkel	1.420	18.083
Harbhajan Singh	1.382	21.510



Criteria for Selecting Wicketkeepers

- Batting Strike Rate: High strike rate for aggressive batting.
- Total Runs: Significant run contributions in T20 matches.
- Dismissals: High number of catches and stumpings.
- Versatility: Ability to bowl a few overs if needed.
- Consistency: Steady performance across multiple matches/seasons.
- Experience: Participation in pressure situations and crucial matches.

ADDITIONAL QUESTIONS FOR FINAL ASSESSMENT

1. **Count of Cities Hosting IPL Matches**

Query:

```
SELECT COUNT(DISTINCT city) AS
NUMBER_OF_CITIES_HOSTED_IPL
FROM matches;
```

Answer:
33

2. **Creating Deliveries_v02 Table**

Query:

```
CREATE TABLE deliveries_v02 AS
SELECT  *, CASE
        WHEN total_runs >= 4 THEN 'boundary'
        WHEN total_runs = 0 THEN 'dot'
              ELSE 'other'
        END AS ball_result
FROM deliveries;
```

Answer:
Query run successfully

3. **Total Boundaries and Dot Balls in Deliveries_v02**

Query:

```
SELECT
    SUM(CASE WHEN ball_result = 'boundary'
THEN 1 ELSE 0 END) AS total_boundaries,
    SUM(CASE WHEN ball_result = 'dot' THEN 1
ELSE 0 END) AS total_dots
FROM deliveries_v02;
```

Answer:

```
Total_boundaries        total_dots
31468                   67841
```

ADDITIONAL QUESTIONS FOR FINAL ASSESSMENT

4. **Total Boundaries Scored by Each Team**

Query:

```
SELECT
    batting_team,
    SUM(CASE WHEN ball_result = 'boundary'
THEN 1 ELSE 0 END) AS total_boundaries
FROM deliveries_v02
GROUP BY batting_team
ORDER BY total_boundaries DESC;
```

Answer:

"batting_team"	"total_boundaries"
"Mumbai Indians"	4118
"Royal Challengers Bangalore"	3800
"Kings XI Punjab"	3780
"Kolkata Knight Riders“	3739
"Chennai Super Kings"	3496
"Rajasthan Royals"	3041
"Delhi Daredevils“	3022
"Sunrisers Hyderabad“	2306
"Deccan Chargers"	1387
"Pune Warriors"	733
"Delhi Capitals"	659
"Gujarat Lions"	624
"Rising Pune Supergiant"	290
"Rising Pune Supergiants"	242
"Kochi Tuskers Kerala"	231

ADDITIONAL QUESTIONS FOR FINAL ASSESSMENT

5. **Total Dot Balls Bowled by Each Team**

Query:

```
SELECT
    bowling_team,
    SUM(CASE WHEN ball_result = 'dot' THEN 1
ELSE 0 END) AS total_dots
FROM deliveries_v02
GROUP BY bowling_team
ORDER BY total_dots DESC;
```

Answer:

```
"bowling_team"                    "total_dots"
"Mumbai Indians"                     8714
"Royal Challengers Bangalore"        7955
"Kolkata Knight Riders"              7894
"Kings XI Punjab"                    7679
"Chennai Super Kings“                7593
"Rajasthan Royals"                   6665
"Delhi Daredevils"                   6520
"Sunrisers Hyderabad"                5248
"Deccan Chargers"                    3306
"Pune Warriors"                      1900
"Delhi Capitals"                     1338
"Gujarat Lions"                      1095
"Rising Pune Supergiant"             698
"Kochi Tuskers Kerala"               626
"Rising Pune Supergiants"            539
"NA"                                 71
```

ADDITIONAL QUESTIONS FOR FINAL ASSESSMENT

6. Total Dismissals by Dismissal Kind

Query:

```
SELECT
    dismissal_kind,
    COUNT(dismissal_kind) AS dismissal_count
FROM deliveries_v02
WHERE dismissal_kind != 'NA'
GROUP BY dismissal_kind;
```

Answer:

```
"dismissal_kind“            "dismissal_count"
"bowled"                          1700
"caught"                          5743
"caught and bowled“               269
"hit wicket"                      12
"lbw"                             571
"obstructing the field“           2
"retired hurt"                    11
"run out"                         893
"stumped"                         294
```

ADDITIONAL QUESTIONS FOR FINAL ASSESSMENT

7. **Top 5 Bowlers Conceding Maximum Extra Runs**

Query:

```
SELECT
    bowler,
    SUM(extra_runs) AS total_extra_runs
FROM deliveries
GROUP BY bowler
ORDER BY total_extra_runs DESC
LIMIT 5;
```

Answer:

```
"bowler"        "total_extra_runs"
"SL Malinga"          293
"P Kumar“             236
"UT Yadav“            226
"DJ Bravo“            210
"B Kumar"             201
```

8. **Creating Deliveries_v03 Table**

Query:

```
CREATE TABLE deliveries_v03 AS
SELECT
    d.*,
    m.date,
    m.venue
FROM
    deliveries_v02 AS d
LEFT JOIN
    matches AS m
ON
    d.id = m.id;
```

Answer:

Query run successfully

ADDITIONAL QUESTIONS FOR FINAL ASSESSMENT

9. **Total Runs Scored by Venue**

Query:

```
SELECT
    venue,
    SUM(total_runs) AS total_runs_scored
FROM deliveries_v03
GROUP BY venue
ORDER BY total_runs_scored DESC;
```

Answer:

```
"venue"                                          "total_runs_scored"
"Eden Gardens"                                         23658
"Wankhede Stadium"                                     23390
"Feroz Shah Kotla“                                     22947
"M Chinnaswamy Stadium"                                20237
"Rajiv Gandhi International Stadium, Uppal“            19484
"MA Chidambaram Stadium, Chepauk"                      17821
"Sawai Mansingh Stadium"                               14264
"Punjab Cricket Association Stadium, Mohali"           10987
"Dubai International Cricket Stadium"                  10402
"Sheikh Zayed Stadium"                                 8830
"Punjab Cricket Association IS Bindra Stadium, Mohali"
                                                       7021
"Maharashtra Cricket Association Stadium"              6780
"Sharjah Cricket Stadium“                              5924
"M.Chinnaswamy Stadium"                                5127
"Dr DY Patil Sports Academy"                           4810
"Subrata Roy Sahara Stadium"                           4755
```

ADDITIONAL QUESTIONS FOR FINAL ASSESSMENT

Answer:

"Kingsmead"	4353
"Brabourne Stadium"	3842
"Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Stadium"	3746
"Sardar Patel Stadium, Motera“	3746
"SuperSport Park"	3653
"Saurashtra Cricket Association Stadium“	3316
"Himachal Pradesh Cricket Association Stadium"	2897
"Holkar Cricket Stadium"	2872
"New Wanderers Stadium“	2292
"Barabati Stadium"	2278
"JSCA International Stadium Complex"	2056
"St George's Park"	2033
"Newlands"	1764
"Shaheed Veer Narayan Singh International Stadium"	1741
"Nehru Stadium"	1363
"Green Park"	1298
"De Beers Diamond Oval"	897
"Vidarbha Cricket Association Stadium, Jamtha"	882
"Buffalo Park"	799
"OUTsurance Oval"	529

ADDITIONAL QUESTIONS FOR FINAL ASSESSMENT

10. **Year-wise Total Runs at Eden Gardens**

Query:

```
SELECT
    EXTRACT(YEAR FROM date) AS year,
    SUM(total_runs) AS total_runs_scored
FROM deliveries_v03
WHERE venue = 'Eden Gardens'
GROUP BY year
ORDER BY total_runs_scored DESC;
```

Answer:

```
"year"     "total_runs_scored"
2018           2885
2019           2651
2015           2386
2013           2304
2017           2194
2010           2167
2016           2073
2012           2012
2011           1854
2008           1843
2014           1289
```

Additional Queries for Data Preparation

```
-- Matches table creation

CREATE TABLE Matches (id int,city varchar,date varchar,player_of_match varchar,venue varchar,
        neutral_venue int,team1 varchar,team2 varchar,toss_winner varchar,toss_decision varchar,
        winner varchar,result varchar,result_margin int,eliminator varchar,method varchar,umpire1
varchar,umpire2 varchar
);
COPY Matches FROM 'C:\Program Files\PostgreSQL\16\data\Data Copy\IPL_matches.csv'
DELIMITER ',' CSV HEADER;
SELECT * FROM Matches;
ALTER TABLE Matches
ALTER COLUMN date TYPE DATE USING TO_DATE(date, 'DD-MM-YYYY');
```

Additional Queries for Data Preparation

```
-- Deliveries table creation

CREATE TABLE Deliveries (id int,inning int,over int,ball int,batsman varchar,non_striker varchar,
        bowler varchar,batsman_runs int,extra_runs int,total_runs int,is_wicket int,dismissal_kind varchar,
        player_dismissed varchar,fielder varchar,extras_type varchar,batting_team
varchar,bowling_team varchar
);
COPY Deliveries FROM 'C:\Program Files\PostgreSQL\16\data\Data Copy\IPL_Ball.csv' DELIMITER ','
CSV HEADER;
```