Indian Premier League (IPL)
Database Management System

GROUP : G3_3

TOP 3 QUERIES :

1. POINT TABLE FOR IPL2024.

->

SQL STATEMENT :

```
SELECT
   s2.tid,
          14 AS Matches,
   s1.points AS win,
          (14 - s1.points-
                   CASE
          WHEN s1.no_result_matches IS NULL THEN 0
          ELSE s1.no_result_matches
      END ) AS LOSS,
   (s1.points * 2 +
      CASE
         WHEN s1.no_result_matches IS NULL THEN 0
         ELSE s1.no_result_matches
      END
   ) AS POINTS,

          s2.net_runrate/10 AS NRR
FROM (
   SELECT p1.tid, p1.points, p2.no_result_matches
   FROM (
      SELECT winner AS tid, COUNT(date) AS points
      FROM (
         SELECT * FROM result WHERE EXTRACT(YEAR FROM date) = 2024 LIMIT 70
      ) AS j1
      WHERE winner IS NOT NULL
      GROUP BY winner
   ) AS p1
```

```
    LEFT JOIN (
        SELECT tid, SUM(cnt) AS no_result_matches
        FROM (
            SELECT t1 AS tid, COUNT(*) AS cnt
            FROM result
            WHERE winner IS NULL
            GROUP BY t1
            UNION ALL
            SELECT t2 AS tid, COUNT(*) AS cnt
            FROM result
            WHERE winner IS NULL
            GROUP BY t2
        ) AS combined
        GROUP BY tid
    ) AS p2 ON p1.tid = p2.tid

    UNION

    SELECT p2.tid, p1.points, p2.no_result_matches
    FROM (
        SELECT winner AS tid, COUNT(date) AS points
        FROM (
            SELECT * FROM result WHERE EXTRACT(YEAR FROM date) = 2024 LIMIT 70
        ) AS j1
        WHERE winner IS NOT NULL
        GROUP BY winner
    ) AS p1
    RIGHT JOIN (
        SELECT tid, SUM(cnt) AS no_result_matches
        FROM (
            SELECT t1 AS tid, COUNT(*) AS cnt
            FROM result
            WHERE winner IS NULL
            GROUP BY t1
            UNION ALL
            SELECT t2 AS tid, COUNT(*) AS cnt
            FROM result
            WHERE winner IS NULL
            GROUP BY t2
        ) AS combined
        GROUP BY tid
    ) AS p2 ON p1.tid = p2.tid
) AS s1
JOIN (
    SELECT r1.t1 AS tid, r1.n1 + r2.n2 AS net_runrate
    FROM (
        SELECT t1, SUM(t1_net_runrate) AS n1
        FROM (
```

```
        SELECT * FROM result WHERE EXTRACT(YEAR FROM date) = 2024 LIMIT 70
      ) AS j2
      GROUP BY t1
   ) AS r1
   JOIN (
      SELECT t2, SUM(t2_net_runrate) AS n2
      FROM (
        SELECT * FROM result WHERE EXTRACT(YEAR FROM date) = 2024 LIMIT 70
      ) AS j3
      GROUP BY t2
   ) AS r2 ON r1.t1 = r2.t2
) AS s2
ON s1.tid = s2.tid
ORDER BY POINTS DESC, net_runrate DESC;
```

OUTPUT :

	tid character varying (5)	matches integer	win bigint	loss numeric	points numeric	nrr double precision
1	KKR	14	10	2	22	1.73412951114815
2	RR	14	8	5	17	0.380527892732677
3	SRH	14	7	6	15	0.410561315403295
4	RCB	14	7	7	14	0.560904452824141
5	CSK	14	7	7	14	0.454694531786026
6	DC	14	7	7	14	-0.284546293594744
7	LSG	14	7	7	14	-1.30936585912591
8	GT	14	5	7	12	-1.3886896976801
9	PBKS	14	5	9	10	-0.212160378910799
10	MI	14	4	10	8	-0.346055474582732

REASON FOR CHOOSING:

->The point table is a central metric used to determine:

- Team rankings throughout the tournament
- Playoff qualifications
- Eliminations and tie-breakers (via Net Run Rate)
- Strategic decisions for teams (e.g., do-or-die matches)

-> By the SQL viewpoint here we can apply maximum operation from sql to solve this query

2. Top 5 matches (Successful run chase)

->

SQL STATEMENT :

```
SELECT *
FROM (
  SELECT
    date,
    CASE
      WHEN tossresult = true THEN
        CASE WHEN t1 = toss THEN t2 ELSE t1 END
      ELSE
        CASE WHEN t1 = toss THEN t1 ELSE t2 END
    END AS cteam,

    CASE
      WHEN tossresult = true THEN
        CASE WHEN t1 = toss THEN t2_run ELSE t1_run END
      ELSE
        CASE WHEN t1 = toss THEN t2_run ELSE t2_run END
    END AS chasing_score,

    CASE
      WHEN
        (CASE
          WHEN tossresult = true THEN
            CASE WHEN t1 = toss THEN t2 ELSE t1 END
          ELSE
            CASE WHEN t1 = toss THEN t1 ELSE t2 END
        END) = t1 THEN t2
      ELSE t1
    END AS dteam,

    CASE
      WHEN
        (CASE
          WHEN tossresult = true THEN
            CASE WHEN t1 = toss THEN t2_run ELSE t1_run END
          ELSE
```

```
                CASE WHEN t1 = toss THEN t2_run ELSE t2_run END
            END) = t1_run THEN t2_run
        ELSE t1_run
    END AS defending_score
  FROM result
  WHERE toss IS NOT NULL
) AS sub
WHERE chasing_score > defending_score
ORDER BY defending_score DESC
LIMIT 5;
```

OUTPUT :

	date date	cteam character varying (5)	chasing_score integer	dteam character varying (5)	defending_score integer
1	2024-04-15	RCB	263	SRH	262
2	2024-04-26	PBKS	262	KKR	261
3	2024-04-16	RR	224	KKR	223
4	2024-05-19	SRH	215	PBKS	214
5	2024-04-23	LSG	213	CSK	210

REASON FOR CHOOSING:

->Highlights Exceptional Match Performances & Fan-Favorite Moments

- The most thrilling matches in T20 cricket
- A strong indicator of team composure under pressure
- Frequently remembered and celebrated by fans and media

BY this query solution we can learn about importance of conditional statements .

3. Orange cap holder race for perticular year(most run scorer)

->

SQL STATEMENT :

```
SELECT * FROM
(
SELECT uid, pname , matches,runs , runs/matches AS avg_score
FROM
   (
   SELECT uid , COUNT(date) AS Matches ,SUM(run) AS RUNS
   FROM
   (SELECT * FROM batting_performence WHERE run IS NOT NULL and EXTRACT(YEAR FROM
   date) = 2024) as r GROUP BY uid
   ) as a
 NATURAL JOIN players  WHERE matches < 18
 ) as player
 NATURAL JOIN
 (SELECT UID , TID FROM sold_players where year = 2024 ) AS team  ORDER BY RUNS DESC ,
   LIMIT 5;
```

OUTPUT:

	uid character (7)	pname character varying (30)	matches bigint	runs bigint	avg_score bigint	tid character varying (5)
1	IPL0401	Virat Kohli	15	741	49	RCB
2	IPL0201	Ruturaj Gaikwad	16	584	36	CSK
3	IPL0604	Riyan Parag	14	573	40	RR
4	IPL0413	Travis Head	15	567	37	SRH
5	IPL0104	Sai Sudharsan	12	560	46	GT

REASON FOR CHOOSING:

The Orange Cap is awarded to the top run-scorer of the season, making this query one of the most prestigious and fan-followed metrics in the IPL. It's not just about numbers—it highlights consistency, dominance, and match-winning ability with the bat.

- This query is key because it tracks the Orange Cap contenders

(Note We can also add for purple cap holder)