1. **Find the player with the most wickets in a given series or tournament and their team information**.

```
SELECT P.FULLNAME, T.TEAMNAME, SUM(MP.WICKETStaken) AS TOTAL_WICKETS
FROM MATCH M
JOIN performance MP ON M.MATCHID = MP.MATCHID
JOIN PLAYER P ON MP.PLAYERID = P.PLAYERID
JOIN TEAM T ON P.TEAMNAME = T.TEAMNAME
GROUP BY P.FULLNAME, T.TEAMNAME
ORDER BY TOTAL_WICKETS DESC
limit 1;
```

Relational Algebra:

$$\tau_{total\_wickets\downarrow}\ \pi_{p.fullname,\ t.teamname,\ SUM(wicketstaken)\rightarrow total\_wickets}\ \gamma_{fullname,\ teamname,\ SUM(wicketstaken)}\ (\rho_{m}\ match \bowtie_{m.matchid = mp.matchid} \rho_{mp}\ performance \bowtie_{mp.playerid = p.playerid} \rho_{p}\ player \bowtie_{p.teamname = t.teamname} \rho_{t}\ team)$$

Output:

	fullname character varying (100)	teamname character varying (100)	total_wickets bigint
1	David Warner	Australia	4

2. **List teams and players who share the same sponsor, showing the sponsor's name and sponsorship amount.**

```
SELECT T.TEAMNAME, P.FULLNAME, S.SPONSORNAME, S.Amount
FROM Team T
JOIN Sponsors SP ON T.TEAMNAME = SP.TeamName
JOIN Sponsorships S ON SP.SponsorID = S.SponsorID
JOIN SponsoredBy SB ON S.SponsorID = SB.SponsorID
JOIN Player P ON SB.PlayerID = P.PlayerID;
```

Relational Algebra:

$$\pi_{t.teamname,\ p.fullname,\ s.sponsorname,\ s.amount}\ (\rho_{t}\ team \bowtie_{t.teamname = sp.teamname} \rho_{sp}\ sponsors \bowtie_{sp.sponsorid = s.sponsorid} \rho_{s}\ sponsorships \bowtie_{s.sponsorid = sb.sponsorid}$$

$\rho_{sb}\, sponsoredby \bowtie_{sb.playerid = p.playerid}$
$\rho_{p}\, player)$

Output:

	teamname character varying (100)	fullname character varying (100)	sponsorname character varying (100)	amount numeric (10,2)
1	India	Babar Azam	Nike	500000.00
2	India	Rohit Sharma	Nike	500000.00
3	India	Virat Kohli	Nike	500000.00
4	Australia	Babar Azam	Nike	500000.00
5	Australia	Rohit Sharma	Nike	500000.00
6	Australia	Virat Kohli	Nike	500000.00
7	England	Ben Stokes	Adidas	400000.00
8	England	Kane Williamson	Adidas	400000.00
9	England	Steve Smith	Adidas	400000.00
10	South Africa	Ben Stokes	Adidas	400000.00
11	South Africa	Kane Williamson	Adidas	400000.00
12	South Africa	Steve Smith	Adidas	400000.00
13	Pakistan	Trent Boult	Coca-Cola	300000.00
14	Pakistan	Mitchell Starc	Coca-Cola	300000.00
15	Pakistan	Jasprit Bumrah	Coca-Cola	300000.00
16	New Zealand	Trent Boult	Coca-Cola	300000.00
17	New Zealand	Mitchell Starc	Coca-Cola	300000.00
18	New Zealand	Jasprit Bumrah	Coca-Cola	300000.00
19	Bangladesh	David Warner	Pepsi	350000.00
20	Bangladesh	Mohammad Nabi	Pepsi	350000.00
21	Bangladesh	Rashid Khan	Pepsi	350000.00
22	Sri Lanka	David Warner	Pepsi	350000.00
23	Sri Lanka	Mohammad Nabi	Pepsi	350000.00
24	Sri Lanka	Rashid Khan	Pepsi	350000.00
25	West Indies	MS Dhoni	Puma	450000.00

3. **Find umpires who have officiated in matches where a specific team (e.g., 'India', 'Australia') played.**

```
SELECT U.FullName AS UmpireName, M.MATCHID, M.DATE, M.Team1Name, M.Team2Name
FROM Umpire U
JOIN Umpires UM ON U.UmpireID = UM.UmpireID
JOIN Match M ON UM.MatchID = M.MatchID
```

```
WHERE M.Team1Name = 'India' OR M.Team2Name = 'India' OR M.Team1Name = 'Australia'
OR M.Team2Name = 'Australia'
ORDER BY M.DATE DESC;
```

Relational Algebra:

$$\tau_{m.date\downarrow}\ \pi_{u.fullname \rightarrow umpirename,\ m.matchid,\ m.date,\ m.team1name,\ m.team2name}\ \sigma_{m.team1name = \text{"India"}\ OR\ m.team2name = \text{"India"}\ OR\ m.team1name = \text{"Australia"}\ OR\ m.team2name = \text{"Australia"}}\ (\rho_{u}\ umpire \bowtie_{u.umpireid = um.umpireid} \rho_{um}\ umpires \bowtie_{um.matchid = m.matchid} \rho_{m}\ match)$$

Output:

	umpirename character varying (100)	matchid integer	date date	team1name character varying (100)	team2name character varying (100)
1	Aleem Dar	2	2024-01-05	Australia	England
2	Simon Taufel	2	2024-01-05	Australia	England
3	Marais Erasmus	2	2024-01-05	Australia	England
4	Richard Kettleborough	2	2024-01-05	Australia	England
5	Simon Taufel	1	2024-01-01	India	Pakistan
6	Marais Erasmus	1	2024-01-01	India	Pakistan
7	Aleem Dar	1	2024-01-01	India	Pakistan
8	Richard Kettleborough	1	2024-01-01	India	Pakistan

4. **List venues where a specific team (e.g., 'India') has played atleast 1 match.**

```
SELECT V.StadiumName, V.City, COUNT(M.MatchID) AS MatchesPlayed
FROM Venue V
JOIN Match M ON V.StadiumName = M.StadiumName
WHERE M.Team1Name = 'India' OR M.Team2Name = 'India'
GROUP BY V.StadiumName, V.City
HAVING COUNT(M.MatchID) > 0;
```

Relational Algebra:

$$\pi_{v.stadiumname,\ v.city,\ COUNT\ (matchid) \rightarrow matchesplayed}\ \sigma_{COUNT\ (matchid) > 0}\ \gamma_{stadiumname,\ city,\ COUNT\ (matchid)}\ \sigma_{m.team1name = \text{"India"}\ OR\ m.team2name = \text{"India"}}\ (\rho_{v}\ venue \bowtie_{v.stadiumname = m.stadiumname} \rho_{m}\ match)$$

Output:

	stadiumname [PK] character varying (100)	city character varying (100)	matchesplayed bigint
1	Eden Gardens	Kolkata	1
2	M. Chinnaswamy Stadium	Bangalore	1

5. **Find players who have taken more than 2 wickets and have an average economy rate below 6.0 runs per over in the matches they played.**

```
SELECT P.FULLNAME, P.TEAMNAME, SUM(Perf.WicketsTaken) AS TotalWickets,
AVG(Perf.Economy) AS AvgEconomyRate
FROM Player P
JOIN Performance Perf ON P.PlayerID = Perf.PlayerID
GROUP BY P.FULLNAME, P.TEAMNAME
HAVING SUM(Perf.WicketsTaken) > 2 AND AVG(Perf.Economy) < 6.0;
```

Relational Algebra:

$$\pi_{p.fullname,\ p.teamname,\ SUM(wicketstaken) \rightarrow totalwickets,\ AVG(economy) \rightarrow avgeconomyrate}$$
$$\sigma_{SUM(wicketstaken) > 2\ AND\ AVG(economy) < 6.0}$$
$$\gamma_{fullname,\ teamname,\ SUM(wicketstaken),\ AVG(economy)}$$
$$(\rho_{p}\ player \bowtie_{p.playerid = perf.playerid} \rho_{perf}\ performance)$$

Output:

	fullname character varying (100)	teamname character varying (100)	totalwickets bigint	avgeconomyrate numeric
1	Jacques Kallis	South Africa	3	5.8000000000000000
2	Jasprit Bumrah	India	3	5.7500000000000000

6. **Find teams that have an average number of matches won per year (since foundation) greater than 0. (Since we have inserted less number of rows in Match Table, we have to keep matches won per year > 0, but with enough number of rows, it could be kept a desired value)**

```
SELECT T.TEAMNAME, COUNT(M.MATCHID) AS TotalWins,
    (COUNT(M.MATCHID) / (EXTRACT(YEAR FROM CURRENT_DATE) - T.YearOfFoundation))
AS WinsPerYear
FROM Team T
JOIN Match M ON T.TEAMNAME = M.WinnerTeamName
GROUP BY T.TEAMNAME, T.YearOfFoundation
```

HAVING (COUNT(M.MATCHID) / (EXTRACT(YEAR FROM CURRENT_DATE) - T.YearOfFoundation)) > 0;

Relational Algebra:

$\pi_{t.teamname, COUNT(matchid) \rightarrow totalwins, COUNT(matchid) / (current\_year - t.yearoffoundation) \rightarrow winsperyear}$

$\sigma_{COUNT(matchid) / (current\_year - t.yearoffoundation) > 0}$

$\gamma_{teamname, yearoffoundation, COUNT(matchid)}$

$(\rho_{t}\, team \bowtie_{t.teamname = m.winnerteamname}$

$\rho_{m}\, match)$

Output:

	teamname [PK] character varying (100)	totalwins bigint	winsperyear numeric
1	Bangladesh	1	0.04166666666666666667
2	UAE	1	0.05555555555555555556
3	Sri Lanka	1	0.02380952380952380952
4	Namibia	1	0.02941176470588235294
5	England	1	0.00680272108843537415
6	Afghanistan	2	0.08695652173913043478
7	India	2	0.02083333333333333333
8	South Africa	2	0.01481481481481481481
9	Zimbabwe	2	0.04651162790697674419
10	New Zealand	1	0.01063829787234042553
11	Scotland	1	0.03333333333333333333
12	Pakistan	2	0.02777777777777777778
13	Australia	2	0.01680672268907563025
14	West Indies	1	0.01041666666666666667

7. **List the players who have won more than 1 "Player of the Match" awards and have an average of at least 1 wickets per match.**

SELECT P.FULLNAME, COUNT(M.POTMPlayerID) AS PlayerOfTheMatchAwards,
AVG(Perf.WicketsTaken) AS AvgWicketsPerMatch
FROM Player P
JOIN Match M ON P.PlayerID = M.POTMPlayerID

```
JOIN Performance Perf ON P.PlayerID = Perf.PlayerID
GROUP BY P.FULLNAME
HAVING COUNT(M.POTMPlayerID) > 1 AND AVG(Perf.WicketsTaken) >= 1;
```

Relational Algebra:

$$\pi_{p.fullname,\ COUNT(potmplayerid) \rightarrow playerofthematchawards,\ AVG(wicketstaken) \rightarrow avgwicketspermatch}$$
$$\sigma_{COUNT(potmplayerid) > 1\ AND\ AVG(wicketstaken) >= 1}$$
$$\gamma_{fullname,\ COUNT(potmplayerid),\ AVG(wicketstaken)}$$
$$(\rho_{p}\ player \bowtie_{p.playerid = m.potmplayerid}$$
$$\rho_{m}\ match \bowtie_{p.playerid = perf.playerid}$$
$$\rho_{perf}\ performance)$$

Output:

	fullname character varying (100)	playerofthematchawards bigint	avgwicketspermatch numeric
1	Ben Stokes	2	1.5000000000000000
2	Jasprit Bumrah	2	1.5000000000000000
3	Shikhar Dhawan	2	1.0000000000000000000
4	David Warner	2	2.0000000000000000
5	Trent Boult	2	2.0000000000000000
6	Babar Azam	2	1.0000000000000000000

8. **List teams with an average player strike rate greater than 80, along with their average strike rate and number of players.**

```
SELECT T.TEAMNAME, AVG(P.StrikeRate) AS AvgStrikeRate, COUNT(P.PlayerID) AS
NumberOfPlayers
FROM Team T
JOIN Player P ON T.TEAMNAME = P.TEAMNAME
GROUP BY T.TEAMNAME
HAVING AVG(P.StrikeRate) > 80;
```

Relational Algebra:

$$\pi_{t.teamname,\ AVG(strikerate) \rightarrow avgstrikerate,\ COUNT(playerid) \rightarrow numberofplayers}$$
$$\sigma_{AVG(strikerate) > 80}$$
$$\gamma_{teamname,\ AVG(strikerate),\ COUNT(playerid)}$$
$$(\rho_{t}\ team \bowtie_{t.teamname = p.teamname}$$
$$\rho_{p}\ player)$$

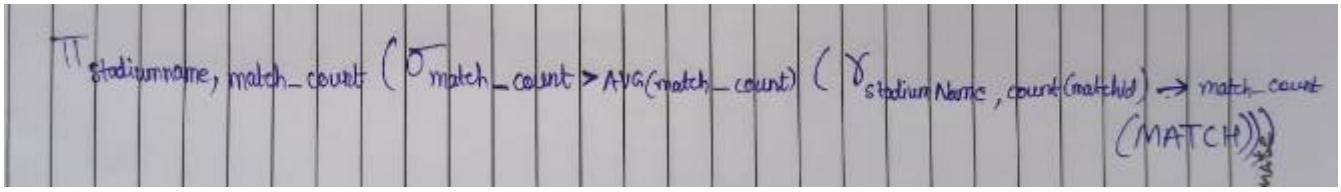
Output:

	teamname [PK] character varying (100)	avgstrikerate numeric	numberofplayers bigint
1	Sri Lanka	130.0000000000000000	1
2	England	112.5750000000000000	2
3	India	98.4571428571428571	7
4	South Africa	122.5000000000000000	2
5	Pakistan	111.1750000000000000	2
6	Australia	97.0500000000000000	5
7	West Indies	140.0000000000000000	1

9. **Find the names and the number of matches of the stadiums that have hosted more number of matches than the average matches played at each stadium.**

```
SELECT STADIUMNAME, COUNT(matchid)
FROM MATCH
GROUP BY STADIUMNAME
HAVING COUNT(matchid) > (SELECT AVG(match_count)
                         FROM (SELECT STADIUMNAME, COUNT(matchid) AS match_count
                         FROM MATCH
                         GROUP BY STADIUMNAME) AS stadium_matches);
```

Relational Algebra:



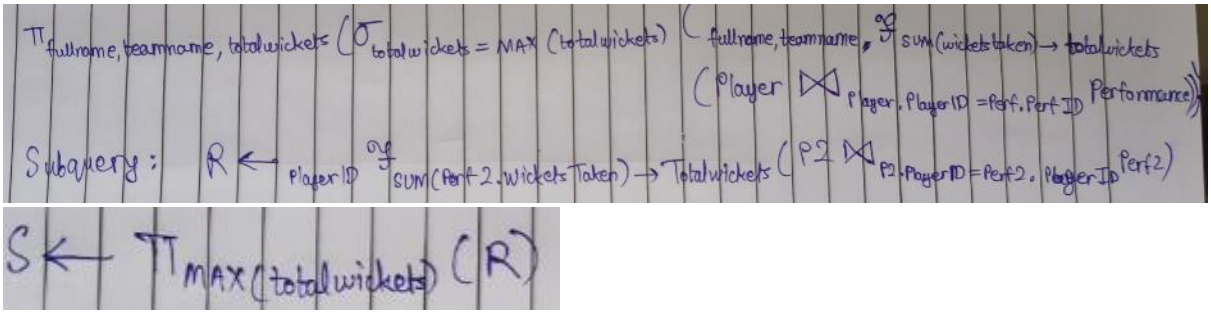
Output:

	stadiumname character varying (100)	count bigint
1	Wanderers Stadium	2
2	R Premadasa Stadium	2

10. **Find the player with the most wickets in each team.**

```
SELECT P.FULLNAME, P.TEAMNAME, SUM(Perf.WicketsTaken) AS TotalWickets
FROM Player P
JOIN Performance Perf ON P.PlayerID = Perf.PlayerID
GROUP BY P.FULLNAME, P.TEAMNAME
HAVING SUM(Perf.WicketsTaken) = (SELECT MAX(TotalWickets)
                                 FROM (SELECT SUM(Perf2.WicketsTaken) AS TotalWickets
                                       FROM Player P2
                                       JOIN Performance Perf2 ON P2.PlayerID = Perf2.PlayerID
                                       WHERE P2.TEAMNAME = P.TEAMNAME
                                       GROUP BY P2.PlayerID) AS TeamWicketCount);
```

Relational Algebra:



Output:

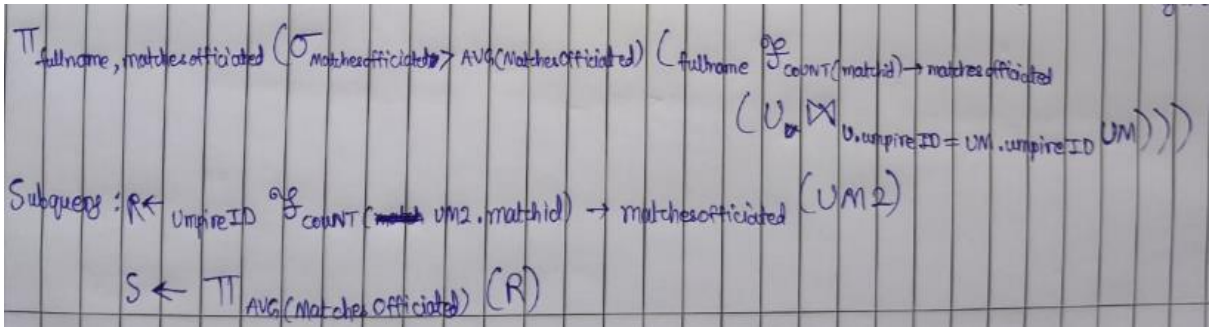
	fullname character varying (100)	teamname character varying (100)	totalwickets bigint
1	Trent Boult	New Zealand	4
2	Jacques Kallis	South Africa	3
3	Babar Azam	Pakistan	2
4	David Warner	Australia	4
5	Kumar Sangakkara	Sri Lanka	2
6	Mohammad Nabi	Afghanistan	1
7	Jasprit Bumrah	India	3
8	Ben Stokes	England	3
9	Chris Gayle	West Indies	0

11. Find umpires who have officiated more matches than the average number of matches officiated by all umpires.

```
SELECT U.FullName, COUNT(UM.MatchID) AS MatchesOfficiated
FROM Umpire U
```

```
    JOIN Umpires UM ON U.UmpireID = UM.UmpireID
    GROUP BY U.FullName
    HAVING COUNT(UM.MatchID) > (SELECT AVG(MatchesOfficiated)
                                FROM (SELECT COUNT(UM2.MatchID) AS MatchesOfficiated
                                      FROM Umpires UM2
                                      GROUP BY UM2.UmpireID) AS UmpireMatches);
```

Relational Algebra:



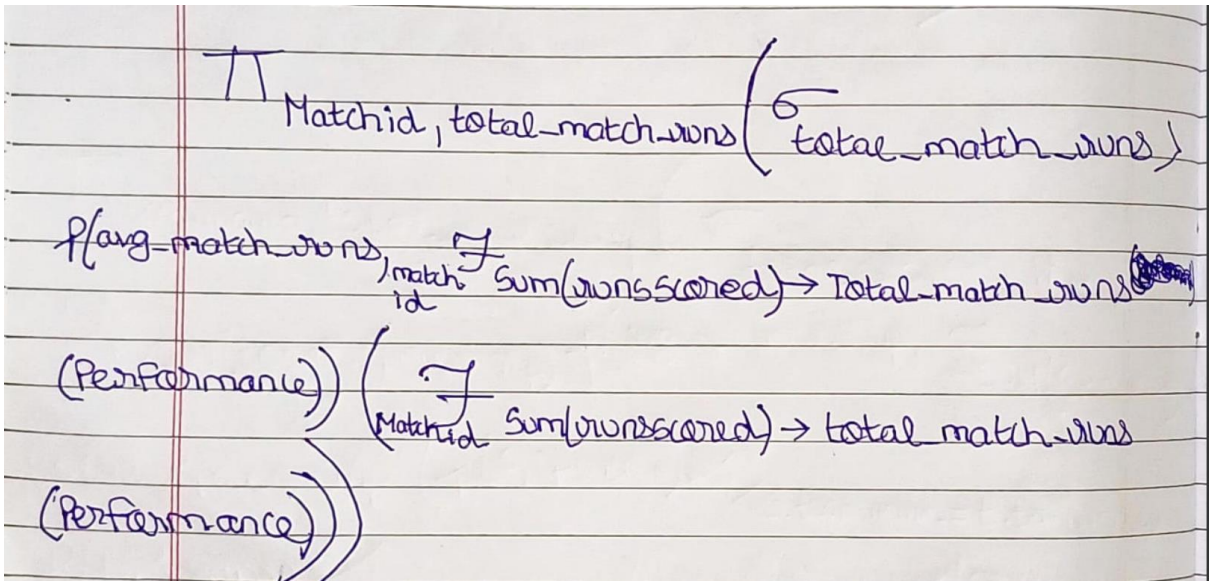
Output:

	fullname character varying (100)	matchesofficiated bigint
1	Aleem Dar	6

12. Find the matches where the total runs are greater than the average total runs in all matches.

```
SELECT MATCHID, TOTAL_MATCH_RUNS
FROM (
  SELECT MATCHID, SUM(RUNSSCORED) AS TOTAL_MATCH_RUNS
  FROM PERFORMANCE
  GROUP BY MATCHID
) AS MATCH_RUNS
WHERE TOTAL_MATCH_RUNS > (
  SELECT AVG(TOTAL_MATCH_RUNS)
  FROM (
    SELECT SUM(RUNSSCORED) AS TOTAL_MATCH_RUNS
    FROM PERFORMANCE
    GROUP BY MATCHID
  ) AS AVG_MATCH_RUNS
);
```

Relational Algebra:



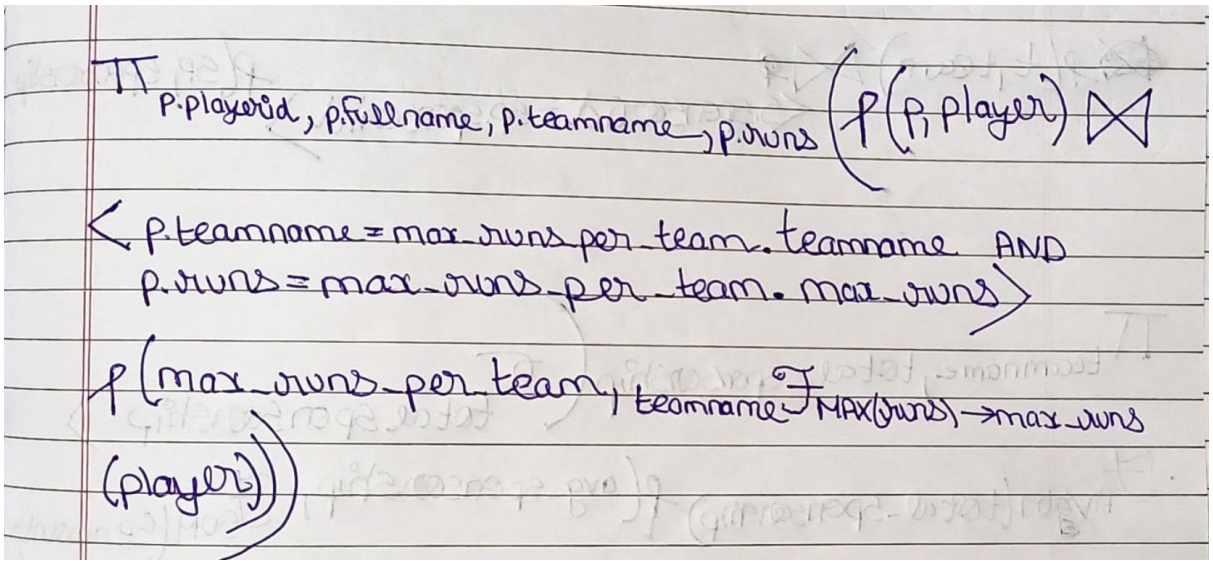
Output:

	matchid integer	total_match_runs bigint
1	3	140
2	5	140
3	4	150
4	10	90
5	6	105
6	2	160
7	16	110
8	1	150

13. **Find the player with most runs in each team.**

```
SELECT p.playerid,p.fullname,p.teamname, p.runs
FROM player as p
JOIN (
   SELECT teamname, MAX(runs) AS max_runs
   FROM player
   GROUP BY teamname
) AS max_runs_per_team
ON p.teamname = max_runs_per_team.teamname
AND p.runs = max_runs_per_team.max_runs;
```

Relational Algebra:



Output:

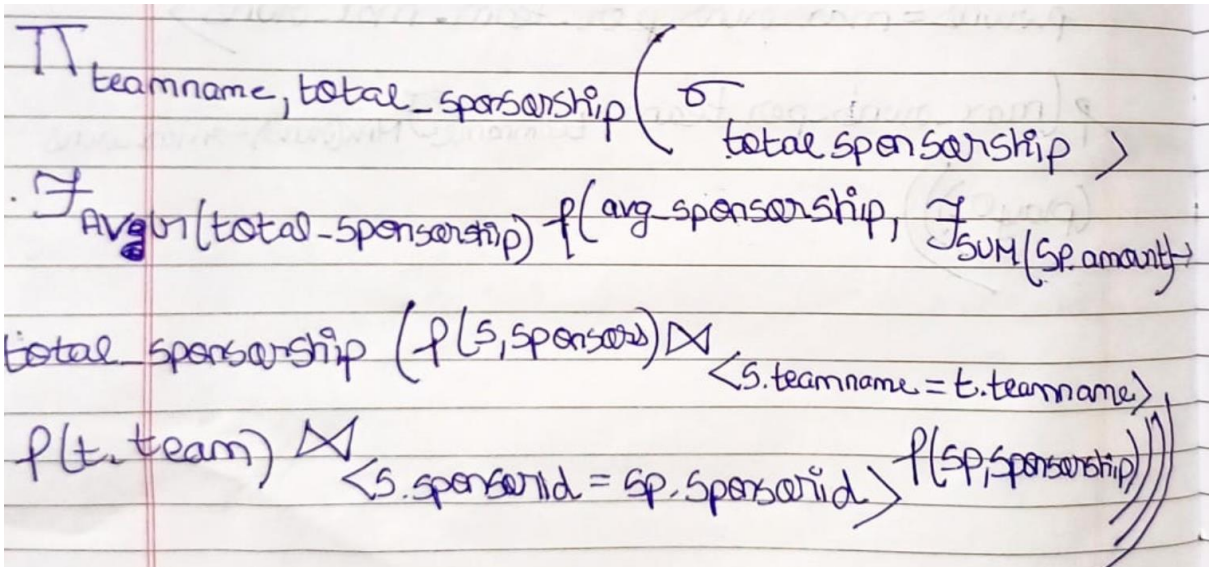
	playerid [PK] integer	fullname character varying (100)	teamname character varying (100)	runs integer
1	4	Steve Smith	Australia	8000
2	11	Mohammad Nabi	Afghanistan	2000
3	12	David Warner	Australia	8000
4	17	Chris Gayle	West Indies	11000
5	20	Jacques Kallis	South Africa	13000
6	1	Virat Kohli	India	12000
7	5	Kane Williamson	New Zealand	6000
8	18	Kumar Sangakkara	Sri Lanka	14000
9	22	James Smith	England	4100
10	25	Shoaib Malik	Pakistan	5200

14. Find teams with total sponsorship amount greater than the average sponsorship amount.

```
SELECT TEAMNAME, TOTAL_SPONSORSHIP
FROM (
   SELECT T.TEAMNAME, SUM(SP.AMOUNT) AS TOTAL_SPONSORSHIP
   FROM SPONSORS S
   JOIN SPONSORSHIPS SP ON S.SPONSORID = SP.SPONSORID
```

```
   JOIN TEAM T ON S.TEAMNAME = T.TEAMNAME
   GROUP BY T.TEAMNAME
) AS TEAM_SPONSORSHIPS
WHERE TOTAL_SPONSORSHIP > (
   SELECT AVG(TOTAL_SPONSORSHIP)
   FROM (
      SELECT SUM(SP.AMOUNT) AS TOTAL_SPONSORSHIP
      FROM SPONSORS S
      JOIN SPONSORSHIPS SP ON S.SPONSORID = SP.SPONSORID
      JOIN TEAM T ON S.TEAMNAME = T.TEAMNAME
      GROUP BY T.TEAMNAME
   ) AS AVG_SPONSORSHIP
);
```

Relational Algebra:



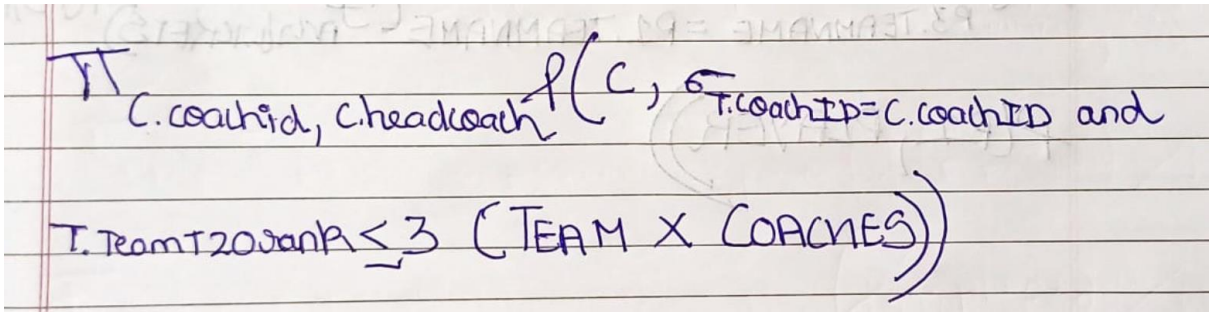
Output:

	teamname [PK] character varying (100)	total_sponsorship numeric
1	UAE	750000.00
2	Ireland	600000.00
3	Kenya	700000.00
4	Zimbabwe	600000.00
5	Scotland	550000.00
6	USA	750000.00
7	Canada	800000.00
8	Namibia	700000.00
9	Nepal	550000.00

15. **Find the coaches of the teams in the Top 3 for T20 Rankings.**

```
SELECT C.COACHID, C.HEADCOACH
FROM COACHES C
WHERE EXISTS (
   SELECT 1
   FROM TEAM T
   WHERE T.COACHID = C.COACHID
   AND T.TEAMT20RANK <= 3
);
```

Relational Algebra:



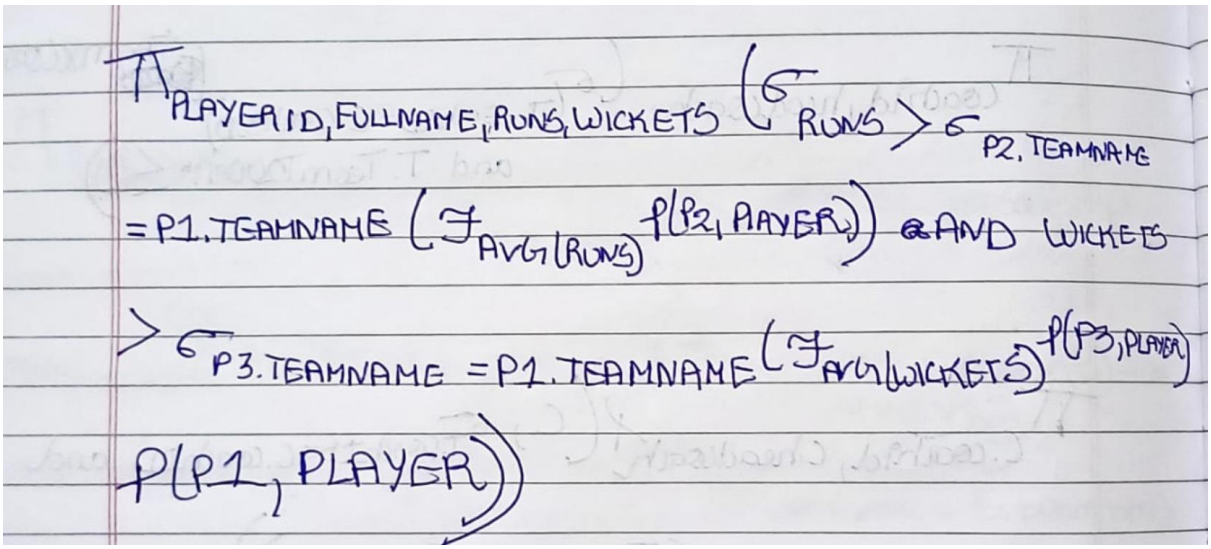
Output:

	coachid [PK] integer	headcoach character varying (100)
1	1	John Smith
2	2	Chris Evans
3	4	Steve Rogers

16. **Find the players with more career Wickets and Runs than their team's average performance metrics.**

```
SELECT PLAYERID, FULLNAME, RUNS, WICKETS
FROM PLAYER P1
WHERE RUNS > (
   SELECT AVG(RUNS)
   FROM PLAYER P2
   WHERE P2.TEAMNAME = P1.TEAMNAME
) AND WICKETS > (
   SELECT AVG(WICKETS)
   FROM PLAYER P3
   WHERE P3.TEAMNAME = P1.TEAMNAME
);
```

Relational Algebra:



Output:

	playerid [PK] integer	fullname character varying (100)	runs integer	wickets integer
1	20	Jacques Kallis	13000	300
2	22	James Smith	4100	120
3	25	Shoaib Malik	5200	105

17. **Find the number of distinct venues where players have played.**

```
SELECT P.FULLNAME, P.TEAMNAME, COUNT(DISTINCT M.StadiumName) AS VenuesPlayed
FROM Player P
JOIN Performance Perf ON P.PlayerID = Perf.PlayerID
JOIN Match M ON Perf.MatchID = M.MatchID
GROUP BY P.FULLNAME, P.TEAMNAME
ORDER BY VenuesPlayed DESC;
```

Relational Algebra:

$\tau_{venuesplayed\downarrow}$
$\pi_{p\,.\,fullname,\ p\,.\,teamname,\ COUNT\ (\backslash delta\ stadiumname)\ \rightarrow\ venuesplayed}$
$\gamma_{fullname,\ teamname,\ COUNT\ (\backslash delta\ stadiumname)}$
$(\rho_{p}\ player \bowtie_{p\,.\,playerid\ =\ perf\,.\,playerid}$
$\rho_{perf}\ performance \bowtie_{perf\,.\,matchid\ =\ m\,.\,matchid}$
$\rho_{m}\ match)$

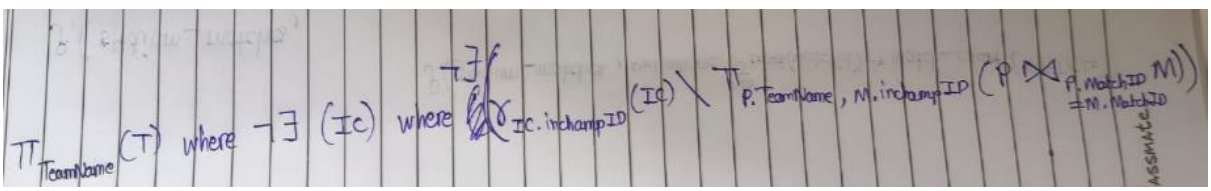
Output:

	fullname character varying (100)	teamname character varying (100)	venuesplayed bigint
1	Rashid Khan	Afghanistan	2
2	Rohit Sharma	India	2
3	Shane Watson	Australia	2
4	Shikhar Dhawan	India	2
5	Steve Smith	Australia	2
6	Trent Boult	New Zealand	2
7	Virat Kohli	India	2
8	Kane Williamson	New Zealand	2
9	David Warner	Australia	2
10	Mitchell Starc	Australia	2
11	Ben Stokes	England	2
12	Babar Azam	Pakistan	2
13	Virender Sehwag	India	1
14	Chris Gayle	West Indies	1
15	Jacques Kallis	South Africa	1
16	Jasprit Bumrah	India	1
17	Kumar Sangakkara	Sri Lanka	1
18	Mohammad Nabi	Afghanistan	1
19	MS Dhoni	India	1
20	AB de Villiers	South Africa	1

18. Find teams that have participated in every international championship.

```
SELECT T.TeamName
FROM Team T
WHERE NOT EXISTS (
   SELECT IC.InChampID
   FROM Match IC
   WHERE NOT EXISTS (
      SELECT 1
      FROM Plays P
      JOIN Match M ON P.MatchID = M.MatchID
      WHERE T.TeamName = P.TeamName
      AND M.InChampID = IC.InChampID
   )
);
```

Relational Algebra:



Output:

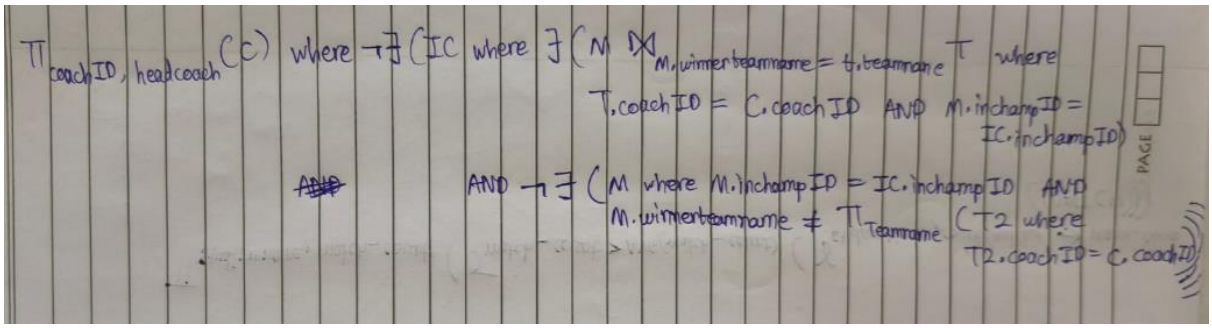
	teamname [PK] character varying (100)
1	India
2	Australia
3	Pakistan
4	New Zealand
5	South Africa
6	England
7	West Indies
8	Sri Lanka
9	Bangladesh
10	Afghanistan
11	Ireland
12	Zimbabwe
13	Scotland
14	UAE
15	Nepal

19. Find coaches of the teams that have played in all international championships.

```
SELECT C.COACHID, C.HEADCOACH
FROM COACHES C
WHERE NOT EXISTS (
  SELECT IC.INCHAMPID
  FROM INTERNATIONALCHAMPIONSHIP IC
  WHERE EXISTS (
    SELECT 1
    FROM MATCH M
    JOIN TEAM T ON M.WINNERTEAMNAME = T.TEAMNAME
    WHERE T.COACHID = C.COACHID
    AND M.INCHAMPID = IC.INCHAMPID
  )
  AND NOT EXISTS (
    SELECT 1
    FROM MATCH M
    WHERE M.INCHAMPID = IC.INCHAMPID
```

```
        AND M.WINNERTEAMNAME <> (
            SELECT T2.TEAMNAME
            FROM TEAM T2
            WHERE T2.COACHID = C.COACHID
        )
    )
);
```

Relational Algebra:



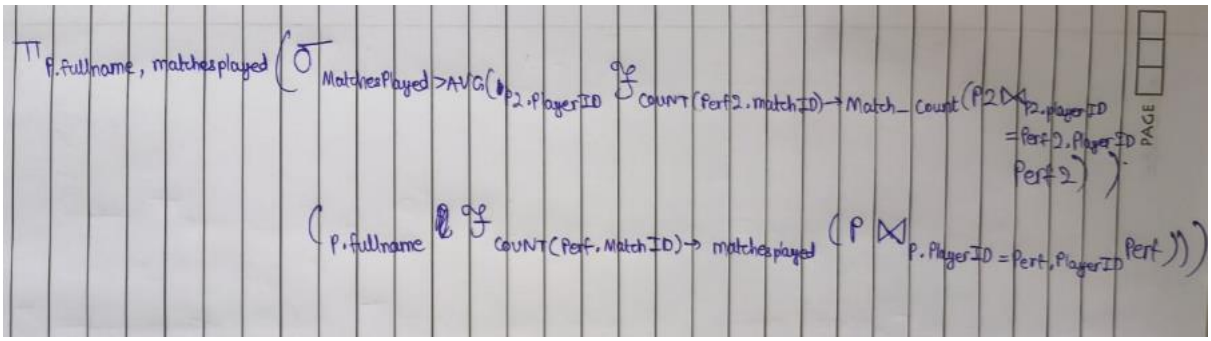
Output:

	coachid [PK] integer	headcoach character varying (100)
1	1	John Smith
2	2	Chris Evans
3	3	Emma Watson
4	4	Steve Rogers
5	5	Bruce Wayne
6	6	Tony Stark
7	7	Clark Kent
8	8	Peter Parker
9	9	Thor Odinson
10	10	Wade Wilson
11	11	Harley Quinn
12	12	James Bond
13	13	Arthur Curry
14	14	Dinah Lance
15	15	Barbara Gordon
16	16	Catherine de Braganza
17	17	Matilda
18	18	Alice
19	19	Frodo Baggins
20	20	Katniss Everdeen

20. Find players who have played more matches than the average number of matches played by all players.

```
SELECT P.FULLNAME, COUNT(Perf.MatchID) AS MatchesPlayed
FROM Player P
JOIN Performance Perf ON P.PlayerID = Perf.PlayerID
GROUP BY P.FULLNAME
HAVING COUNT(Perf.MatchID) >
    (SELECT AVG(MatchCount)
     FROM (SELECT COUNT(Perf2.MatchID) AS MatchCount
         FROM Player P2
         JOIN Performance Perf2 ON P2.PlayerID = Perf2.PlayerID
         GROUP BY P2.PlayerID) AS PlayerMatches);
```

Relational Algebra:



Output:

	fullname character varying (100)	matchesplayed bigint
1	Virat Kohli	2
2	Kane Williamson	2
3	Ben Stokes	2
4	Rohit Sharma	2
5	Mitchell Starc	2
6	Jasprit Bumrah	2
7	Mohammad Nabi	2
8	Shikhar Dhawan	2
9	Rashid Khan	2
10	Shane Watson	2
11	David Warner	2
12	Trent Boult	2
13	Babar Azam	2
14	Steve Smith	2