

SQL PROBLEM CHALLENGE

DAY 12

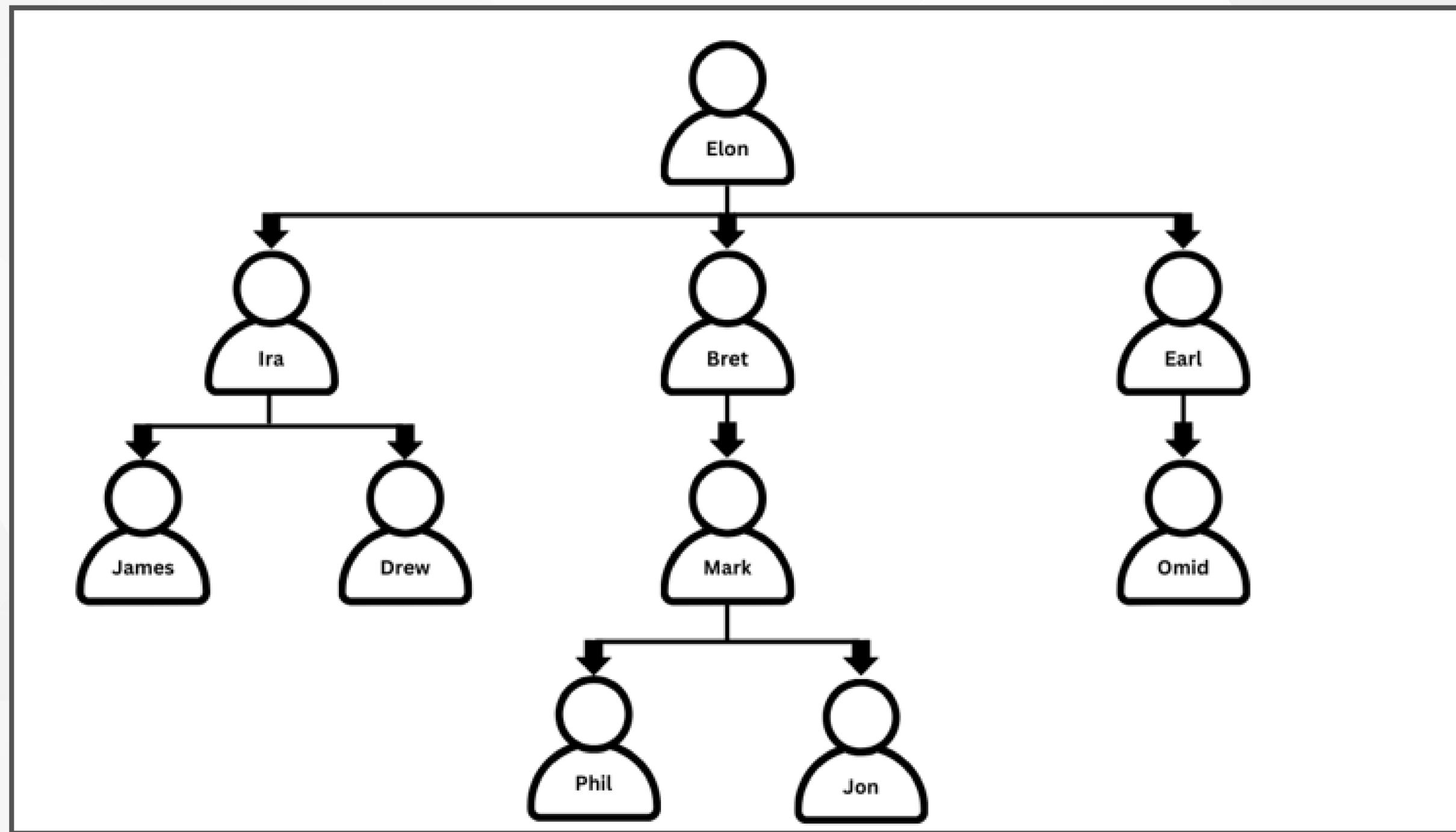
- Problem Statement
- Expected Output
- Solution
- Solution Breakdown



PROBLEM STATEMENT

Below graph shows the hierarchy of employees in a company.

ToDo: split the hierarchy and show the employees corresponding to their team.



EXPECTED OUTPUT

TEAMS	MEMBERS
Team 1	Elon, Bret, Mark, Phil, Jon
Team 2	Elon, Earl, Omid
Team 3	Elon, Ira, James, Drew

SOLUTION

```
WITH recursivecte
AS (
    SELECT employee, manager, CONCAT('Team ', ROW_NUMBER() OVER (ORDER BY employee)) AS Teams
    FROM company
    WHERE manager = (SELECT employee FROM company WHERE manager IS NULL)

    UNION ALL

    SELECT d.employee, d.manager, Teams
    FROM recursivecte c JOIN company d ON c.employee = d.manager
),
distinctMembersCte |
AS (
    SELECT employee, Teams
    FROM recursivecte

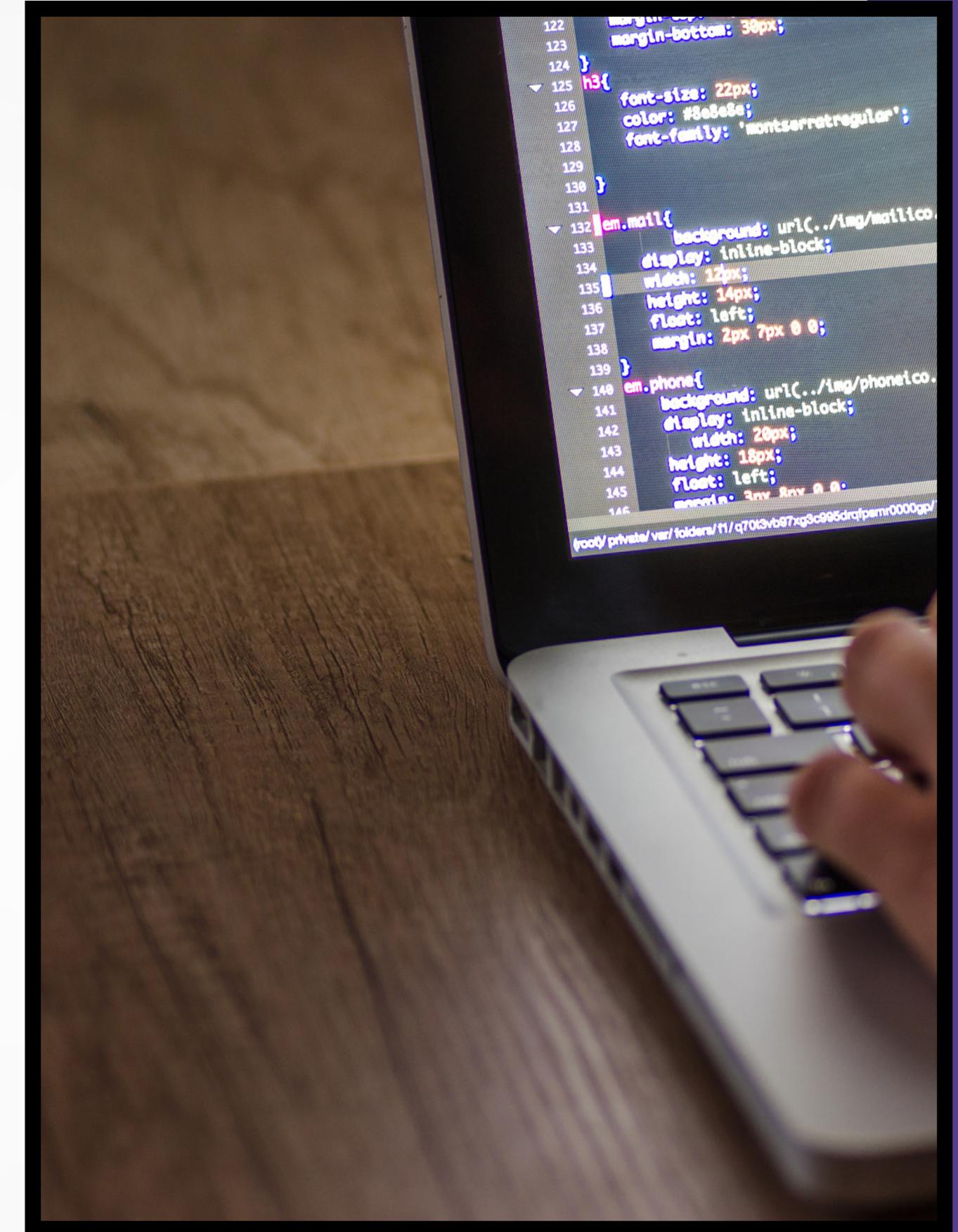
    UNION

    SELECT manager, Teams
    FROM recursivecte
)
SELECT Teams, STRING_AGG(employee, ',') AS Members
FROM distinctMembersCte
GROUP BY teams
ORDER BY teams;
```

OUTPUT

	Teams	Members
1	Team 1	Bret,Elon,Jon,Mark,Phil
2	Team 2	Earl,Elon,Omid
3	Team 3	Drew,Elon,Ira,James

LET'S CHECK OUT THE BREAKDOWN OF THIS SOLUTION



```

WITH recursivecte
AS (
    SELECT employee, manager, CONCAT('Team ', ROW_NUMBER() OVER (ORDER BY employee)) AS Teams
    FROM company
    WHERE manager = (SELECT employee FROM company WHERE manager IS NULL) 1
    UNION ALL 2

    SELECT d.employee, d.manager, Teams
    FROM recursivecte c JOIN company d ON c.employee = d.manager
),

```

1

	employee	manager	Teams
1	Bret	Elon	Team 1
2	Earl	Elon	Team 2
3	Ira	Elon	Team 3

2

	employee	manager	Teams
1	Bret	Elon	Team 1
2	Earl	Elon	Team 2
3	Ira	Elon	Team 3
4	Drew	Ira	Team 3
5	James	Ira	Team 3
6	Omid	Earl	Team 2
7	Mark	Bret	Team 1
8	Jon	Mark	Team 1
9	Phil	Mark	Team 1

```
,  
distinctMembersCte  
AS (  
    SELECT employee,Teams  
    FROM recursivecte  
  
    UNION  
  
    SELECT manager,Teams  
    FROM recursivecte  
)
```

	employee	Teams
1	Bret	Team 1
2	Drew	Team 3
3	Earl	Team 2
4	Elon	Team 1
5	Elon	Team 2
6	Elon	Team 3
7	Ira	Team 3
8	James	Team 3
9	Jon	Team 1
10	Mark	Team 1
11	Omid	Team 2
12	Phil	Team 1

We are also including the root employee (i.e. Elon) here along with the Teams, each employee belongs to.
Union - Helps us remove duplicates that we received in previous output

```
SELECT Teams, STRING_AGG(employee,',') AS Members  
FROM distinctMembersCte  
GROUP BY teams  
ORDER BY teams;
```

Finally, concatenating the employees using **STRING_AGG** function. We get the final result NOW!

	Teams	Members
1	Team 1	Bret,Elon,Jon,Mark,Phil
2	Team 2	Earl,Elon,Omid
3	Team 3	Drew,Elon,Ira,James



THANK YOU
