Experiment 2

Student Name: Mandeep Kaur   **UID:** 23BCS10854
Branch: BE-CSE   **Section/Group:** KRG_2B
Semester: 5th   **Date of Performance:** 28/7/25
Subject Name: Advanced Database and Management System
Subject Code: 23CSP-333

1. Aim:

Medium-Problem Title: Organizational Hierarchy Explorer

You are a Database Engineer at TalentTree Inc., an enterprise HR analytics platform that stores employee data, including their reporting relationships. The company maintains a centralized Employee relation that holds:

Each employee's ID, name, department, and manager ID (who is also an employee in the same table).

Your task is to generate a report that maps employees to their respective managers, showing:

1. The employee's name and department.
2. Their manager's name and department (if applicable).

This will help the HR department visualize the internal reporting hierarchy.

Hard-Problem Title: Financial Forecast Matching with Fallback Strategy

You are a Data Engineer at FinSight Corp, a company that models Net Present Value (NPV) projections for investment decisions. Your system maintains two key datasets:

Year_tbl: Actual recorded NPV's of various financial instruments over different years:

   ID: Unique Financial instrument identifier.

   YEAR: Year of record

   NPV: Net Present Value in that year

Queries_tbl: A list of instrument-year pairs for which stakeholders are requesting NPV values:

   ID: Financial instrument identifier

YEAR: Year of interest.

Find the NPV of each query from the Queries table. Return the output order by ID and Year in the sorted form.

However, not all ID-YEAR combinations in the Queries table are present in the Year_tbl. If an NPV is missing for a requested combination, assume it to be 0 to maintain a consistent financial report.

2. **Objective:** To design efficient SQL queries that map employees to their managers for clear reporting hierarchy visualization and to retrieve NPV values for requested instrument-year pairs, ensuring missing values default to 0 for accurate and complete financial forecasting reports.

3. **Expected Results-**

Medium Level Problem-

Input Table: Employee

EmpID	Ename	Department	ManagerID
1	Alice	HR	NULL
2	Bob	Finance	1
3	Charlie	IT	1
4	David	Finance	2
5	Eve	IT	3
6	Frank	HR	1

OUTPUT

EmployeeName	EmployeeDept	ManagerName	ManagerDept
Alice	HR	NULL	NULL
Bob	Finance	Alice	HR
Charlie	IT	Alice	HR
David	Finance	Bob	Finance
Eve	IT	Charlie	IT
Frank	HR	Alice	HR

Hard Level Problem -

Year_tbl

ID	YEAR	NPV
1	2018	100
7	2020	30
13	2019	40
1	2019	113
2	2008	121
3	2009	12
11	2020	99
7	2019	0

ID	YEAR
1	2019
2	2008
3	2009
7	2018
7	2019
7	2020
13	2019

Queries_tbl

OUTPUT:

ID	YEAR	NPV
1	2019	113
2	2008	121
3	2009	12
7	2018	0
7	2019	0
7	2020	30
13	2019	40

4. SQL QUERY -

```
--EXPERIMENT 2--
--MEDIUM PROBLEM--


CREATE TABLE EMPLOYEE(

EMPID INT PRIMARY KEY IDENTITY(1,1),ENAME VARCHAR(MAX),DEPARTMENT VARCHAR(MAX),MANAGERID INT,

FOREIGN KEY (MANAGERID) REFERENCES EMPLOYEE(EMPID));


INSERT INTO EMPLOYEE(ENAME,DEPARTMENT,MANAGERID) VALUES ('ALICE','HR',NULL)

INSERT INTO EMPLOYEE(ENAME,DEPARTMENT,MANAGERID) VALUES ('BOB','FINANCE',1)

INSERT INTO EMPLOYEE(ENAME,DEPARTMENT,MANAGERID) VALUES ('CHARLIE','IT',1)

INSERT INTO EMPLOYEE(ENAME,DEPARTMENT,MANAGERID) VALUES ('DAVID','FINANCE',2)

INSERT INTO EMPLOYEE(ENAME,DEPARTMENT,MANAGERID) VALUES ('EVE','IT',3)
```

```sql
INSERT INTO EMPLOYEE(ENAME,DEPARTMENT,MANAGERID) VALUES ('FRANK','HR',1)

SELECT * FROM EMPLOYEE;


SELECT M.ENAME AS [EMPLOYEE NAME],M.DEPARTMENT AS [EMPLOYEE DEPARTMENT],E.ENAME AS
[MANAGER NAME],E.DEPARTMENT AS [MANAGERDEPT]

FROM EMPLOYEE AS E

RIGHT OUTER JOIN

EMPLOYEE AS M

ON E.EMPID=M.MANAGERID




--EXPERIMENT 2--

--HARD PROBLEM--


-- Create Year_tbl (holds actual NPV values)

CREATE TABLE YEAR_TBL (

    ID INT,

    YEAR INT,

    NPV INT

);


-- Create Queries table (requested values)

CREATE TABLE QUERIES_TBL (

    ID INT,

    YEAR INT

);


-- Insert data into Year_tbl

INSERT INTO YEAR_TBL (ID, YEAR, NPV)

VALUES
```

```sql
(1, 2018, 100),
(7, 2020, 30),
(13, 2019, 40),
(1, 2019, 113),
(2, 2008, 121),
(3, 2009, 12),
(11, 2020, 99),
(7, 2019, 0);


-- Insert data into Queries
INSERT INTO QUERIES_TBL (ID, YEAR)
VALUES
(1, 2019),
(2, 2008),
(3, 2009),
(7, 2018),
(7, 2019),
(7, 2020),
(13, 2019);


SELECT * FROM YEAR_TBL;
SELECT * FROM QUERIES_TBL;


SELECT Q.ID AS [ID],Q.YEAR AS [YEAR],ISNULL(NPV,0) AS [NPV]
FROM QUERIES_TBL AS Q
LEFT OUTER JOIN
YEAR_TBL AS Y
ON Q.ID=Y.ID AND Q.YEAR=Y.YEAR
```

OUTPUTS OBTAINED -

Medium Problem-

```
13    INSERT INTO EMPLOYEE(ENAME,DEPARTME
14    SELECT * FROM EMPLOYEE;
15
16    SELECT M.ENAME AS [EMPLOYEE NAME]
```

100 %   ✓ No issues found

Results   Messages

	EMPID	ENAME	DEPARTMENT	MANAGERID
1	1	ALICE	HR	NULL
2	2	BOB	FINANCE	1
3	3	CHARLIE	IT	1
4	4	DAVID	FINANCE	2
5	5	EVE	IT	3
6	6	FRANK	HR	1

```
16    SELECT M.ENAME AS [EMPLOYEE NAME],M.DEPARTMENT AS [EMPLOYE
17    FROM EMPLOYEE AS E
18    RIGHT OUTER JOIN
19    EMPLOYEE AS M
20    ON E.EMPID=M.MANAGERID
21
22
```

100 %   ✓ No issues found   Ln: 16   Ch: 1   SPC   CRLF

Results   Messages

	EMPLOYEE NAME	EMPLOYEE DEPARTMENT	MANAGER NAME	MANAGERDEPT
1	ALICE	HR	NULL	NULL
2	BOB	FINANCE	ALICE	HR
3	CHARLIE	IT	ALICE	HR
4	DAVID	FINANCE	BOB	FINANCE
5	EVE	IT	CHARLIE	IT
6	FRANK	HR	ALICE	HR

Hard Problem -

```
64
65    SELECT * FROM YEAR_TBL;
66    SELECT * FROM QUERIES_TBL;
67
```
100 %   ✓ No issues found

Results | Messages

	ID	YEAR	NPV
1	1	2018	100
2	7	2020	30
3	13	2019	40
4	1	2019	113
5	2	2008	121
6	3	2009	12
7	11	2020	99
8	7	2019	0

```
66    SELECT * FROM QUERIES_TBL;
```
Programmability

100 %   ✓ No issues found

Results | Messages

	ID	YEAR
1	1	2019
2	2	2008
3	3	2009
4	7	2018
5	7	2019
6	7	2020
7	13	2019

```
68    SELECT Q.ID AS [ID],Q.YEAR AS [YEAR],ISNULL(NPV,0) AS [NPV
69    FROM QUERIES_TBL AS Q
70    LEFT OUTER JOIN
71    YEAR_TBL AS Y
72    ON Q.ID=Y.ID AND Q.YEAR=Y.YEAR
```
100 %   ✓ No issues found   ►   Ln: 68   Ch: 1   SPC   CRL

Results | Messages

	ID	YEAR	NPV
1	1	2019	113
2	2	2008	121
3	3	2009	12
4	7	2018	0
5	7	2019	0
6	7	2020	30
7	13	2019	40