



DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

EXPERIMENT - 2

Name: Neha Sharma	UID: 23BCS10766
Branch: CSE	Section: KRG1-B
Semester: 5	Date of Performance: 29/05/2025
Subject: ADBMS	Subject Code: 23CSP-333

Question 1 : Medium-Level Problem

Problem Title: Organizational Hierarchy Explorer

Scenario:

You are a Database Engineer at TalentTree Inc., an enterprise HR analytics platform. The company maintains a centralized Employee table that stores:

- emp_id: Unique employee ID
- name: Employee's name
- department: Department name
- manager_id: ID of their reporting manager (also an employee in the same table)

Task:

1. Create a single table named Employee that includes columns for emp_id, name, department, and manager_id.
2. Insert at least six records, ensuring:
 - Some employees have managers (i.e., manager_id not null)
 - Some are top-level managers (i.e., manager_id is null)
3. Write a self-join query to generate a report that includes:
 - The employee's name and department
 - Their manager's name and department (if applicable)
4. The result should help HR visualize the internal reporting hierarchy.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Expected Output:

A table showing each employee's name & department alongside their manager's name & department.

For top-level managers, the manager columns should show NULL.

Solution :

Program Code:

```
CREATE TABLE E (  
    EmpID INT,  
    Ename VARCHAR(20),  
    Department VARCHAR(20),  
    ManagerID INT  
);  
  
INSERT INTO E (EmpID, Ename, Department, ManagerID) VALUES  
(1, 'Alice', 'HR', NULL),  
(2, 'Bob', 'Finance', 1),  
(3, 'Charlie', 'IT', 2),  
(4, 'David', 'Finance', 5),  
(5, 'Eve', 'HR', 3),  
(6, 'Frank', 'IT', NULL);  
  
SELECT E1.Ename AS EmployeeName,  
E1.Department AS EmployeeDept,  
E2.Ename AS ManagerName,  
E2.Department AS ManagerDepartment  
FROM E E1  
JOIN E E2  
ON  
E2.EmpID = E1.ManagerID
```

Output:

Results		Messages		
	EmployeeName	EmployeeDept	ManagerName	ManagerDepartment
1	Bob	Finance	Alice	HR
2	Charlie	IT	Bob	Finance
3	David	Finance	Eve	HR
4	Eve	HR	Charlie	IT

Question 2 : Hard - Level Problem

Problem Title: NPV Lookup with Missing Data Handling (Medium)

Scenario:

You are a Data Engineer at FinSight Corp, where Net Present Value (NPV) data is stored and queried regularly.

You maintain two tables:

1. Year_tbl — Actual recorded NPV values:

- ID: Unique Financial instrument identifier
- YEAR: Year of record
- NPV: Net Present Value in that year

2. Queries_tbl — Stakeholder NPV queries:

- ID: Financial instrument identifier
- YEAR: Year of interest

Task:

1. Create the two tables described above: Year_tbl and Queries_tbl.



2. Insert at least 5–6 rows of data into each, ensuring:
3. Write an SQL query to:
 - Return each ID, YEAR, and the corresponding NPV (if it exists)
 - Replace missing NPV values with 0
 - Order the output by ID and YEAR in ascending order

Solution :

Program Code:

```
CREATE TABLE Y_TBL (  
    ID INT,  
    YEAR INT,  
    NPV INT  
);  
  
CREATE TABLE Q_TBL (  
    ID INT,  
    YEAR INT  
);  
  
INSERT INTO Y_TBL (ID, YEAR, NPV) VALUES  
(1, 2018, 100),  
(7, 2020, 30),  
(13, 2019, 40),  
(1, 2019, 113),  
(2, 2008, 121),  
(3, 2009, 12),  
(11, 2020, 99),  
(7, 2019, 0);
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
INSERT INTO Q_TBL (ID, YEAR) VALUES
```

```
(1, 2019),
```

```
(2, 2008),
```

```
(3, 2009),
```

```
(7, 2018),
```

```
(7, 2019),
```

```
(7, 2020),
```

```
(13, 2019);
```

```
SELECT E2.ID, E2.YEAR, E1.NPV
```

```
FROM
```

```
Y_TBL AS E1
```

```
INNER JOIN
```

```
Q_TBL AS E2
```

```
ON
```

```
E1.ID = E2.ID AND E1.YEAR = E2.YEAR;
```

Output:

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