WORKSHEET 1

Student Name: Supriya Dutta	**UID: 23BCS13291**
Branch: CSE(3rd Year)	**Section/Group: Krg-1-A**
Semester: 5th	**Date of Performance: 24/07/25**
Subject Name: ADBMS	**Subject Code: 23CSP-333**

1. **AIM:**

   [MEDIUM]

   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:

   . The employee's name and department
   . Their manager's name and department (if applicable)
   . This will help the HR department visualize the internal reporting hierarchy.

   [HARD]

   To write SQL queries that retrieve requested Net Present Value (NPV) data by performing conditional joins on multiple columns (ID and YEAR) across two related tables, and to handle missing data using appropriate SQL functions.

2. **Tools Used :** SQL Server Management Studio

DBMS SCRIPT:

--Q1: Medium Level

```
CREATE TABLE Employee (
    EmpID INT PRIMARY KEY,
    EmpName VARCHAR(50) NOT NULL,
    Department VARCHAR(50) NOT NULL,
    ManagerID INT NULL  -- Self-reference to EmpID
);


ALTER TABLE Employee
ADD CONSTRAINT FK_Manager FOREIGN KEY (ManagerID) REFERENCES
Employee(EmpID);



INSERT INTO Employee (EmpID, EmpName, Department, ManagerID)
VALUES
(1, 'Alice', 'HR', NULL),        -- Top-level manager
(2, 'Bob', 'Finance', 1),
(3, 'Charlie', 'IT', 1),
(4, 'David', 'Finance', 2),
(5, 'Eve', 'IT', 3),
(6, 'Frank', 'HR', 1);


SELECT
    E.EmpName AS EmployeeName,
    E.Department AS EmployeeDept,
    M.EmpName AS ManagerName,
    M.Department AS ManagerDept
FROM
    Employee E
LEFT JOIN
    Employee M
ON
E.ManagerID = M.EmpID;
```

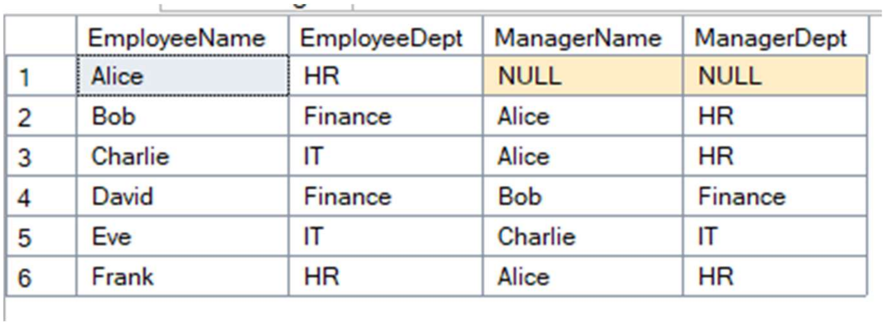
--Q2: Hard Level

```
CREATE TABLE Year_tbl (
    id INT,
    year INT,
    NPV INT
);

-- Sample data for Year_tbl
INSERT INTO Year_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);

CREATE TABLE Queries_tbl (
    id INT,
    year INT
);

-- Sample data for Queries_tbl
INSERT INTO Queries_tbl (id, year)
VALUES
(1, 2019),
(2, 2008),
(3, 2009),
(7, 2018),
(7, 2019),
(7, 2020),
(13, 2019);

--query:

SELECT
    Y.id AS ID,
    Y.year AS Year,
    ISNULL(Q.NPV, 0) AS NPV
FROM
    Queries_tbl AS Y
LEFT OUTER JOIN
    Year_tbl AS Q
ON
    Y.id = Q.id AND Y.year = Q.year;
```

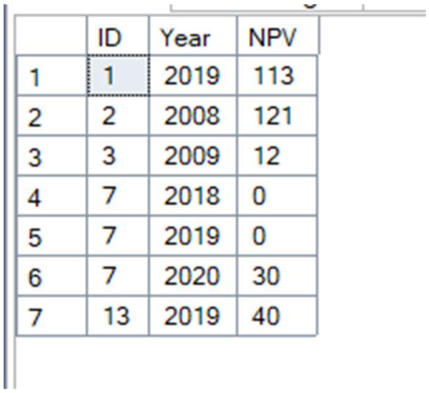
3. OUTPUT:

--Medium Level:



	EmployeeName	EmployeeDept	ManagerName	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

--Medium Level:



	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

4. Learning Outcomes:

- Learn to implement **self-joins** to represent hierarchical relationships within a single table (e.g., employees reporting to managers).

- Construct relational queries to retrieve information such as employee–manager relationships, handling missing data with LEFT JOIN.

- Design and populate tables using CREATE TABLE and INSERT INTO for real-world hierarchical or time-series data scenarios.

- Perform **multi-table joins** to compare and match data across datasets, such as actual vs. requested values (e.g., NPV for specific years).

- Use functions like ISNULL() to replace missing values with defaults during join operations.

- Apply conditional joins involving multiple keys (e.g., matching by both ID and YEAR) to ensure precise data mapping.

- Develop SQL-based problem-solving skills to extract insights from HR and financial datasets in enterprise systems.