CHANDIGARH UNIVERSITY
Discover. Learn. Empower.

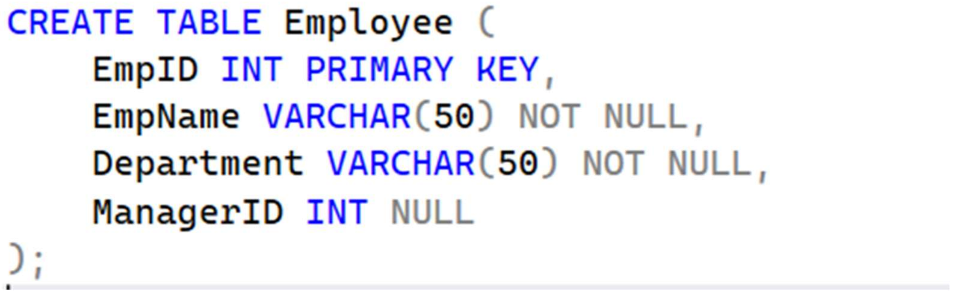
Experiment 1.2

Student Name: Umang Kumar	**UID:** 23BAI70024
Branch: BE-AIT-CSE	**Section/Group:** 23AML_KRG-1(G2)
Semester: 5th	**Date of Performance:** 28 July, 2025
Subject Name: ADBMS	**Subject Code:** 23CSP-333

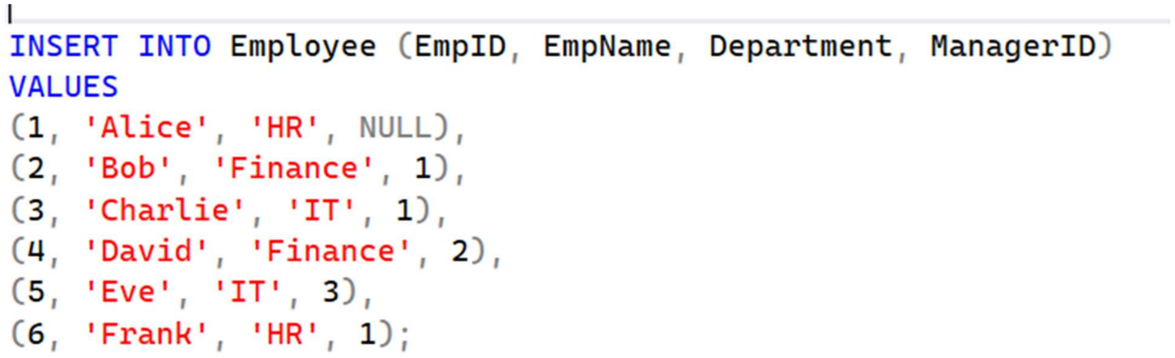
MEDIUM - LEVEL

1. **Problem Title:** Organizational Hierarchy Explorer
2. **Procedure (Step-by-Step):** 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).
   1. Your task is to generate a report that **maps employees to their respective managers**, showing:
   2. The employee's name and department
   3. Their manager's name and department (if applicable)
   4. This will help the HR department visualize the internal reporting hierarchy.

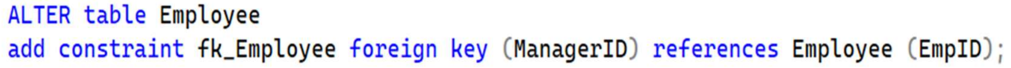
   Sample Output Description: When the join is performed, we get a list where each book title is shown along with its author's name and their country.
3. **SQL Commands:**



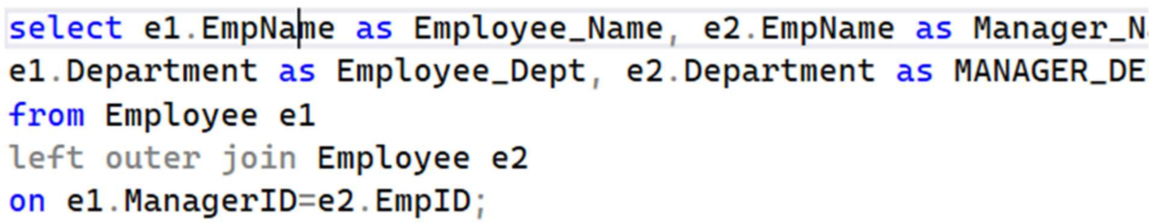
```
CREATE TABLE Employee (
    EmpID INT PRIMARY KEY,
    EmpName VARCHAR(50) NOT NULL,
    Department VARCHAR(50) NOT NULL,
    ManagerID INT NULL
);
```



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

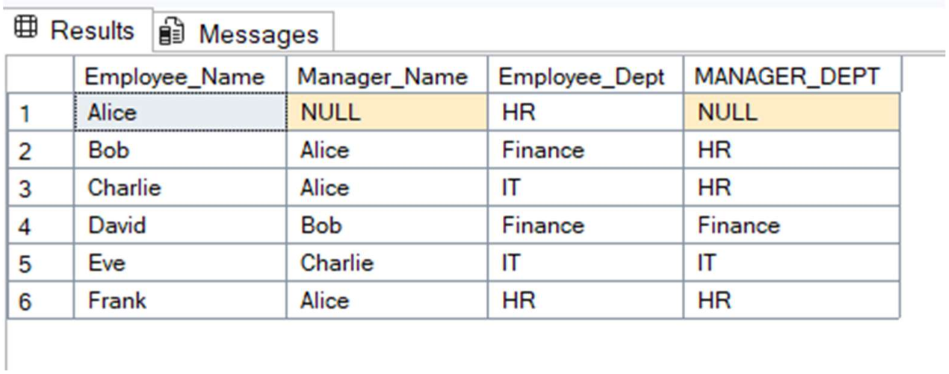


```
ALTER table Employee
add constraint fk_Employee foreign key (ManagerID) references Employee (EmpID);
```



```
select e1.EmpName as Employee_Name, e2.EmpName as Manager_N
e1.Department as Employee_Dept, e2.Department as MANAGER_DE
from Employee e1
left outer join Employee e2
on e1.ManagerID=e2.EmpID;
```

4. **Output:**



	Employee_Name	Manager_Name	Employee_Dept	MANAGER_DEPT
1	Alice	NULL	HR	NULL
2	Bob	Alice	Finance	HR
3	Charlie	Alice	IT	HR
4	David	Bob	Finance	Finance
5	Eve	Charlie	IT	IT
6	Frank	Alice	HR	HR

HARD - LEVEL

1. **Problem Title:** Financial Forecast Matching with Fallback Strategy
2. **Procedure (Step-by-Step):**

   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.

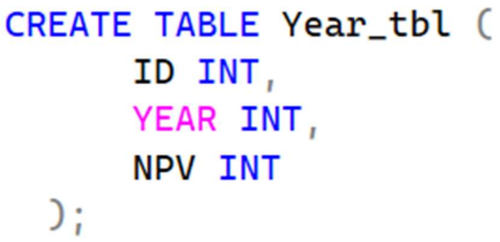
   YEAR: Year of record

   NPV: Net Present Value in that year

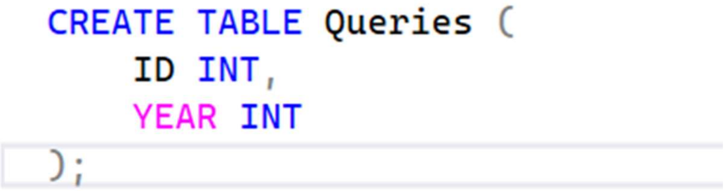
   Queries_tbl: A list of instrument-year pairs for which stakeholders are requesting NPV values:

   ID: Financial instrument identifier

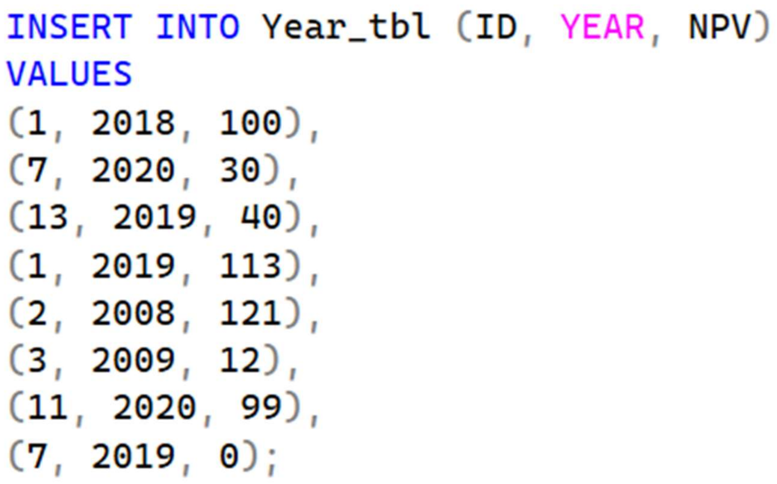
   YEAR: Year of interest.
3. **Sample Output Description:** The result shows the names of departments which are associated with more than two courses in the system.
4. **SQL Commands:**



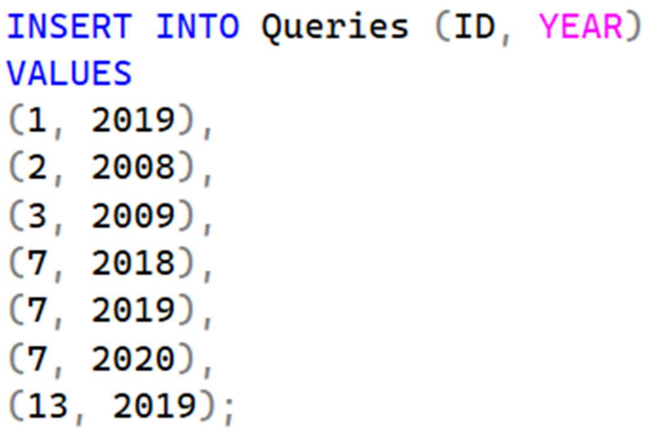
```
CREATE TABLE Year_tbl (
      ID INT,
      YEAR INT,
      NPV INT
  );
```



```
CREATE TABLE Queries (
    ID INT,
    YEAR INT
);
```

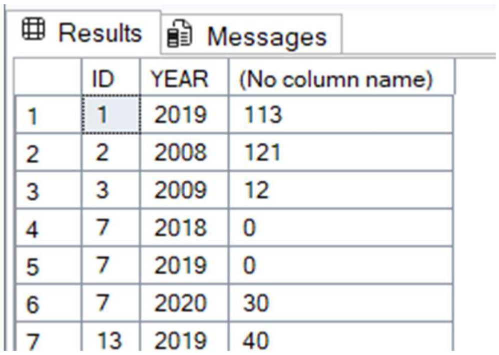


```
INSERT INTO Year_tbl (ID, YEAR, NPV)
VALUES
(1, 2018, 100),
(7, 2020, 30),
(13, 2019, 40),
(1, 2019, 113),
(2, 2008, 121),
(3, 2009, 12),
(11, 2020, 99),
(7, 2019, 0);
```



```
INSERT INTO Queries (ID, YEAR)
VALUES
(1, 2019),
(2, 2008),
(3, 2009),
(7, 2018),
(7, 2019),
(7, 2020),
(13, 2019);
```

5. **Output:**



	ID	YEAR	(No column name)
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

6. **Learning Outcomes:**

   a. Learnt about Left Joins.
   b. Handling missing data with COALESCE.
   c. Creating and populating tables.
   d. Data integration from multiple tables.