CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

Experiment 1.2

Student Name: Gagnesh Kakkar

UID: 23BCS11196

Branch: B.E-C.S.E

Section/Group: 23BCS_KRG-2B

Semester: 5th

Date of Performance: 28 July, 2025

Subject Name: ADBMS

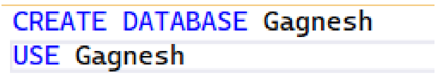
Subject Code: 23CSP-333

MEDIUM - LEVEL

1. **Problem Title:** Organizational Hierarchy Explorer

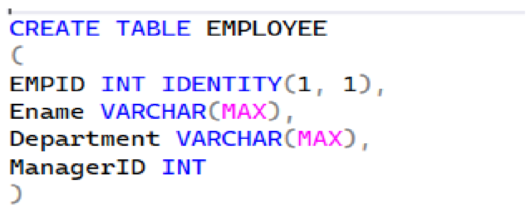
2. **Procedure (Step-by-Step):** You are a Database Engineer at TalentTree Inc., an enterprise HR analytics platform that stores employee data, including their reporting relationships. The company maintains a centralized Employee relation that holds:
   Each employee's ID, name, department, and manager ID (who is also an employee in the same table).
   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.

3. **SQL Commands:**
   a. Create the database and use it:



```sql
CREATE DATABASE Gagnesh
USE Gagnesh
```

   b. Create tables EMPLOYEE:



```sql
CREATE TABLE EMPLOYEE
(
EMPID INT IDENTITY(1, 1),
Ename VARCHAR(MAX),
Department VARCHAR(MAX),
ManagerID INT
)
```

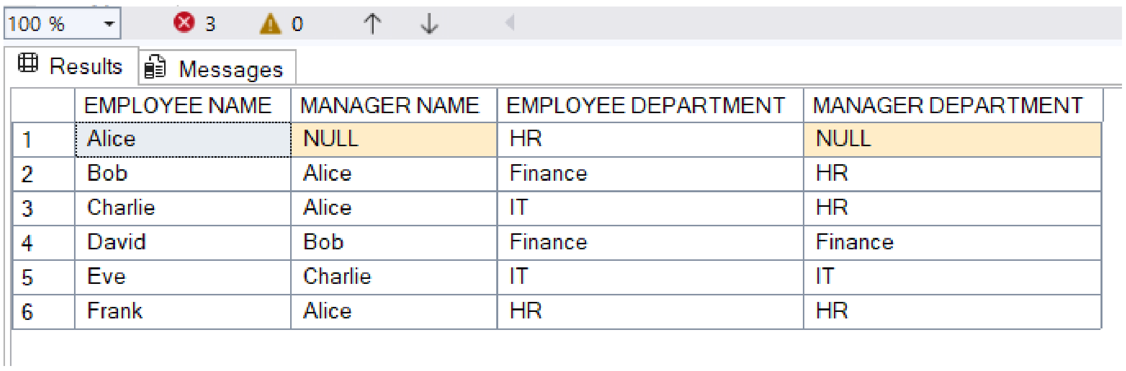
c. Insert the values in the tables:

```
INSERT INTO EMPLOYEE (Ename, Department, ManagerID)
VALUES ('Alice', 'HR', NULL),
('Bob', 'Finance', 1),
('Charlie', 'IT', 1),
('David', 'Finance', 2),
('Eve', 'IT', 3),
('Frank', 'HR', 1)
```

d. Selecting the Employee Name, Manager name, and Employee and Manager Department:

```
SELECT E1.Ename AS [EMPLOYEE NAME], E2.Ename AS [MANAGER NAME],
       E1.Department AS [EMPLOYEE DEPARTMENT], E2.Department AS [MANAGER DEPARTMENT]
FROM EMPLOYEE AS E1
LEFT OUTER JOIN
EMPLOYEE AS E2
ON
E1.ManagerID = E2.EMPID
```

5. **Output:**



	EMPLOYEE NAME	MANAGER NAME	EMPLOYEE DEPARTMENT	MANAGER DEPARTMENT
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

6. **Learning Outcome:**

a. I learnt how to create and manage relational databases using SQL.
b. I learnt how to define primary and foreign key constraints to link tables.
c. I learnt how to insert multiple records into SQL tables efficiently.
d. I learnt how to use LEFT OUTER JOIN to retrieve combined data from related tables.

HARD - LEVEL

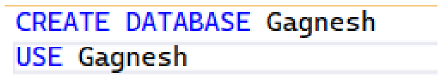
1. **Problem Title:** Financial Forecast Matching with Fallback Strategy

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.

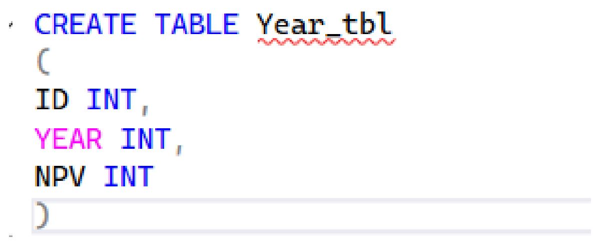
Find the NPV of each query from the Queries table. Return the output order by ID and Year in the sorted form.
However, not all **ID-YEAR combinations** in the Queries table are present in the Year_tbl. If an NPV is missing for a requested combination, assume it to be 0 to maintain a consistent financial report.

3. **SQL Commands:**
   a. Create the database and use it:

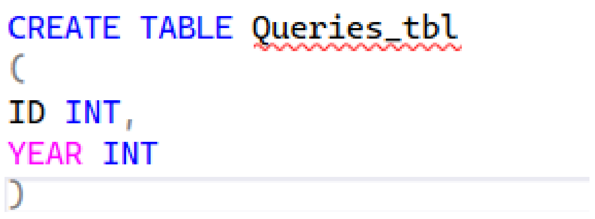


```
CREATE DATABASE Gagnesh
USE Gagnesh
```

   b. Create tables Year_tbl and Queries_tbl:



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



```
CREATE TABLE Queries_tbl
(
ID INT,
YEAR INT
)
```

c. Insert the values in the tables:

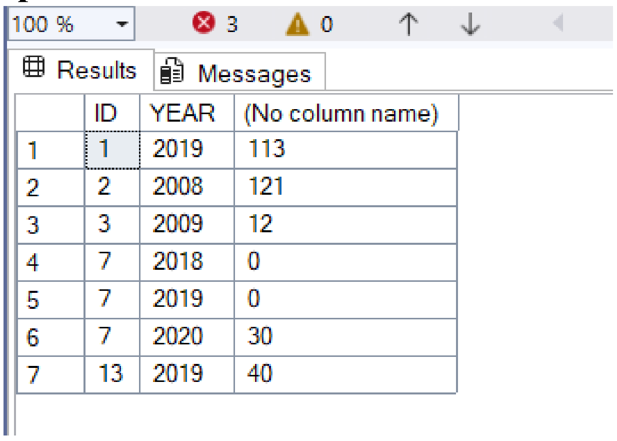
```
INSERT INTO Year_tbl
VALUES (1, 2018, 100),
    (7, 2020, 30),
    (13, 2019, 40),
    (1, 2019, 113),
    (2, 2008, 121),
    (3, 2009, 12),
    (11, 2020, 9),
    (7, 2019, 0)
```

```
INSERT INTO Queries_tbl VALUES (1, 2019),
(2, 2008),
(3, 2009),
(7, 2018),
(7, 2019),
(7, 2020),
(13, 2019)
```

d. Selecting the ID, YEAR and NPV:

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

4. 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

5. Learning Outcome:

a. I learnt how to create and manage relational databases using SQL.
b. I learnt how to define primary and foreign key constraints to link tables.
c. I learnt how to insert multiple records into SQL tables efficiently.
d. I learnt how to use LEFT OUTER JOIN to retrieve combined data from related tables.
e. I learnt how to use ISNULL keyword in SQL queries on tables.