Experiment - 2

Student Name: Shivansh Bhatt	**UID:** 23BCS10552
Branch: BE-CSE	**Section/Group:** KRG-1 (B)
Semester: 5th	**Date:** 29-07-25
Subject Name: ADBMS	**Subject Code:** 23CSP-333

Aim:

Q1. Organizational Hierarchy Explorer

You are a Database Engineer at Talent Tree 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.

Q2. NPV Lookup Using Left Join

Implement the following tables:

- Year: contains actual NPV (Net Present Value) for different years and IDs.
- Queries: contains (ID, YEAR) pairs for which you want to find the NPV.
- Goal is to return the queried (ID, YEAR) pairs along with the corresponding NPV value.
- If NPV doesn't exist for that (ID, YEAR), show 0.

Objective:

Q1: Organizational Hierarchy Explorer

The objective of this task is to explore and represent the internal reporting structure of an organization using self-referencing relationships within a single employee table. A centralized Employee table will be designed to store each employee's ID, name, department, and their manager's ID. Since managers are also employees, the table will reference itself through the manager ID. The goal is to write a query that joins the table to itself in order to map every employee to their respective manager. The final output should include each employee's name and department, along with their manager's name and department, where applicable. This information will enable the HR department to better understand and visualize the organizational hierarchy.

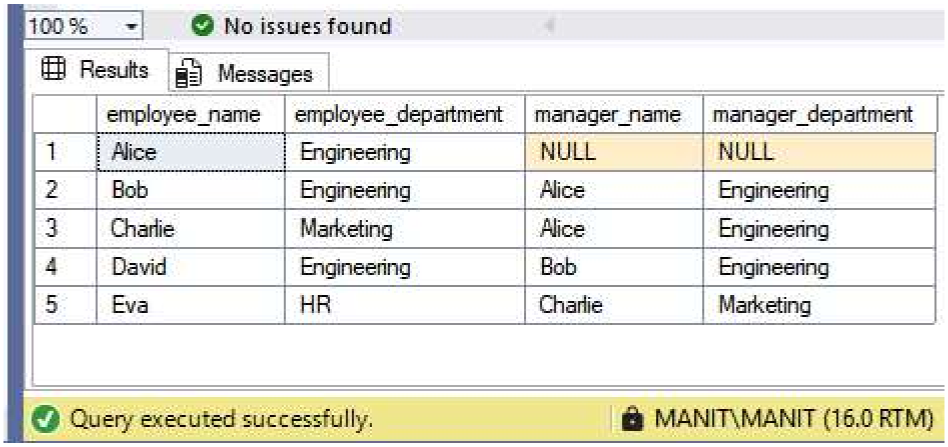
Q2: NPV Lookup Using Left Join

The objective here is to perform a lookup operation to retrieve financial data using a left join strategy. Two tables are involved: the first (Year) stores actual NPV (Net Present Value) values for various ID and year combinations; the second (Queries) contains a list of (ID, YEAR) pairs for which the NPV values are needed. The goal is to return each pair from the Queries table along with its corresponding NPV value, if available. If there is no matching NPV in the first table, the result should show 0 for that entry. This ensures completeness of the query results and helps provide default values in the absence of recorded data.

DBMS Code & Output:

Q1: Organizational Hierarchy Explorer

```
use KRG_1B;

-- Employees table
create table Employees (
    emp_id int primary key,
    name varchar(100),
    department varchar(100),
    manager_id int
);

-- Sample data
insert into Employees
values
    (1, 'Alice', 'Engineering', NULL),
    (2, 'Bob', 'Engineering', 1),
    (3, 'Charlie', 'Marketing', 1),
    (4, 'David', 'Engineering', 2),
    (5, 'Eva', 'HR', 3);

-- Output query through left join
select
    e.name as 'employee_name',
    e.department as 'employee_department',
    m.name as 'manager_name',
    m.department as 'manager_department'
from
    Employees e
LEFT JOIN
    Employees m
on
    e.manager_id = m.emp_id;
```



	employee_name	employee_department	manager_name	manager_department
1	Alice	Engineering	NULL	NULL
2	Bob	Engineering	Alice	Engineering
3	Charlie	Marketing	Alice	Engineering
4	David	Engineering	Bob	Engineering
5	Eva	HR	Charlie	Marketing

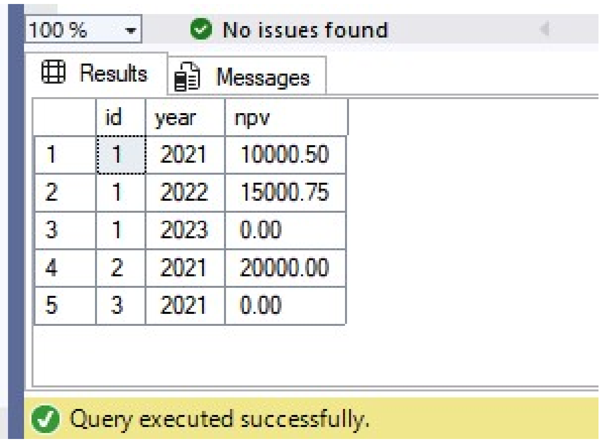
Query executed successfully.

MANIT\MANIT (16.0 RTM)

Q2: NPV Lookup Using Left Join

```
use KRG_1B;

-- Year_tbl to store actual NPV data
create table Year_tbl (
    id int,
    year int,
    npv decimal(10, 2)
);

-- Queries table containing lookup pairs
create table Queries (
    id int,
    year int
);

-- Sample data for Year_tbl
insert into Year_tbl
values
    (1, 2021, 10000.50),
    (2, 2021, 20000.00),
    (1, 2022, 15000.75);

-- Sample data for Queries table
insert into Queries
values
    (1, 2021),
    (1, 2022),
    (1, 2023),
    (2, 2021),
    (3, 2021);

-- Output query through left join
select
    q.id,
    q.year,
    isnull(y.npv, 0) as 'npv'
from
    Queries q
```

```
left join
    Year_tbl y
on
    q.id = y.id
    and
    q.year = y.year;
```



No issues found

	id	year	npv
1	1	2021	10000.50
2	1	2022	15000.75
3	1	2023	0.00
4	2	2021	20000.00
5	3	2021	0.00

Query executed successfully.