



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## Experiment 2

**Student Name:** Tanisha Kumari

**UID:** 23BCS12542

**Branch:** BE-CSE

**Section/Group:** KRG-2B

**Semester:** 5<sup>th</sup>

**Date of Performance:** 28-07-2025

**Subject Name:** ADBMS

**Subject Code:** 23CSP-333

### 1. Aim:

- (A) To create a table Employee\_tbl to capture employee details along with their reporting manager.
- (B) To create two tables Year\_tbl and Queries for tracking financial Net Present Value (NPV) by year and ID.

### 2. Objective:

- To understand and apply self-joins in SQL to relate rows within the same table, such as employee-manager relationships.
- To design and populate relational tables for temporal financial data, enabling time-series queries.
- To retrieve data using outer joins and display comprehensive information, including unmatched rows.
- To utilize SQL functions like ISNULL() for handling NULL values in results.
- To enhance skills in data modeling, joining strategies, and conditional data retrieval in SQL Server.

### 3. SQL SCRIPT:

#### --Program 1

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

```
ALTER TABLE Employee_data  
ADD CONSTRAINT FK_Manager_data FOREIGN KEY (ManagerID) REFERENCES  
    Employee_data(EmpID);
```

```
INSERT INTO Employee_data (EmpID, EmpName, Department, ManagerID)  
VALUES
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
(1, 'Alice', 'HR', NULL),  
(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_data E  
LEFT JOIN  
    Employee_data M  
ON  
    E.ManagerID = M.EmpID;
```

## --Program 2

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

```
INSERT INTO Year_info (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_info (  
    id INT,  
    year INT  
);
```

```
INSERT INTO Queries_info (id, year)  
VALUES
```

(1, 2019),  
(2, 2008),  
(3, 2009),  
(7, 2018),  
(7, 2019),  
(7, 2020),  
(13, 2019);

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

## 4. OUTPUT:

Results		Messages		
	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

Results		Messages	
	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



## 5. Learning Outcomes:

- Understand how to create and manage relational tables using primary and foreign keys.
- Learn to maintain referential integrity between related datasets.
- Gain practical experience in retrieving data from multiple tables using INNER JOIN.
- Learn to write subqueries for filtering and extracting specific information.