**Lab Exercise 15– Recursive CTE in dbt for Structured Table**

**Objective:**

Learn how to use a recursive CTE in dbt to process hierarchical data in a structured table. The exercise will involve creating a hierarchical "Employee Manager" dataset and building a dbt model to generate a hierarchy report.

**Exercise Steps**

**Step 1: Create the employees Table in Snowflake**

Run the following SQL in Snowflake to create a structured table with a hierarchy:

CREATE OR REPLACE TABLE raw\_db.raw\_data.employees (

employee\_id INT,

employee\_name STRING,

manager\_id INT

);

INSERT INTO raw\_db.raw\_data.employees (employee\_id, employee\_name, manager\_id)

VALUES

(1, 'Alice', NULL), -- CEO

(2, 'Bob', 1), -- Reports to Alice

(3, 'Charlie', 1), -- Reports to Alice

(4, 'David', 2), -- Reports to Bob

(5, 'Eve', 2), -- Reports to Bob

(6, 'Frank', 3); -- Reports to Charlie

**Step 2: Create a Recursive CTE in dbt**

1. **File Structure**:
   * Place the following SQL in models/recursive\_hierarchy.sql.
2. **Recursive CTE in the dbt Model**:

WITH RECURSIVE employee\_hierarchy AS (

    -- Anchor member: Start with top-level employees (no manager)

    SELECT

        employee\_id,

        employee\_name,

        manager\_id,

        CAST(employee\_name AS STRING) AS hierarchy\_path,

        1 AS hierarchy\_level

    FROM raw\_db.raw\_data.employees

    WHERE manager\_id IS NULL

    UNION ALL

    -- Recursive member: Add employees who report to the current level

    SELECT

        e.employee\_id,

        e.employee\_name,

        e.manager\_id,

        CONCAT(h.hierarchy\_path, ' > ', e.employee\_name) AS hierarchy\_path,

        h.hierarchy\_level + 1 AS hierarchy\_level

    FROM raw\_db.raw\_data.employees e

    JOIN employee\_hierarchy h

    ON e.manager\_id = h.employee\_id

)

SELECT

    employee\_id,

    employee\_name,

    manager\_id,

    hierarchy\_path,

    hierarchy\_level

FROM employee\_hierarchy

ORDER BY hierarchy\_level, hierarchy\_path

**Step 4: Run the Model**

Execute the dbt commands:

dbt run --select recursive\_hierarchy

**Step 5: Verify the Output**

Run the following query in Snowflake to verify the hierarchy:

SELECT \* FROM raw\_db.raw\_data.recursive\_hierarchy;

**Expected Output**

| **employee\_id** | **employee\_name** | **manager\_id** | **hierarchy\_path** | **hierarchy\_level** |
| --- | --- | --- | --- | --- |
| 1 | Alice | NULL | Alice | 1 |
| 2 | Bob | 1 | Alice > Bob | 2 |
| 3 | Charlie | 1 | Alice > Charlie | 2 |
| 4 | David | 2 | Alice > Bob > David | 3 |
| 5 | Eve | 2 | Alice > Bob > Eve | 3 |
| 6 | Frank | 3 | Alice > Charlie > Frank | 3 |

**Key Concepts Covered**

1. **Recursive CTE**:
   * Process hierarchical data.
2. **Hierarchy Levels**:
   * Assign levels based on reporting structure.
3. **Hierarchy Path**:
   * Build a readable path for reporting relationships.