CELEBAL TECHNOLOGIES

Final Project Report

1. Problem Statement

The task is to generate an employee hierarchy based on the data stored in the Employee_Master table. Each employee has an associated manager (ReportingTo field), and their name needs to be derived from the EmailID column. The goal is to create a stored procedure that builds a hierarchy of employees, calculates the level of each employee in the hierarchy, and extracts the first and last names from the email addresses.

2. Objective

To write a SQL stored procedure that builds an employee hierarchy table with the following fields:

- EMPLOYEEID
- REPORTINGTO
- EMAILID
- LEVEL (depth in hierarchy)
- FIRSTNAME (from EmailID)
- LASTNAME (from EmailID)

3. SQL Code

```
CREATE TABLE Employee_Master (
    EmployeeID VARCHAR(20),
    ReportingTo NVARCHAR(MAX),
    EmailID NVARCHAR(MAX)
);

INSERT INTO Employee_Master (EmployeeID, ReportingTo, EmailID) VALUES
('H1', NULL, 'john.doe@example.com'),
('H2', NULL, 'jane.smith@example.com'),
('H3', 'John Smith H1', 'alice.jones@example.com'),
('H4', 'Jane Doe H1', 'bob.white@example.com'),
('H5', 'John Smith H3', 'charlie.brown@example.com'),
('H6', 'Jane Doe H3', 'david.green@example.com'),
('H7', 'John Smith H4', 'emily.gray@example.com'),
('H8', 'Jane Doe H4', 'frank.wilson@example.com'),
('H9', 'John Smith H5', 'george.harris@example.com'),
('H10', 'Jane Doe H5', 'hannah.taylor@example.com'),
('H11', 'John Smith H6', 'irene.martin@example.com'),
('H12', 'Jane Doe H6', 'jack.roberts@example.com'),
```

```
('H13', 'John Smith H7', 'kate.evans@example.com'),
('H14', 'Jane Doe H7', 'laura.hall@example.com'),
('H15', 'John Smith H8', 'mike.anderson@example.com'),
('H16', 'Jane Doe H8', 'natalie.clark@example.com'),
('H17', 'John Smith H9', 'oliver.davis@example.com'),
('H18', 'Jane Doe H9', 'peter.edwards@example.com'),
('H19', 'John Smith H10', 'quinn.fisher@example.com'),
('H20', 'Jane Doe H10', 'rachel.garcia@example.com'),
('H21', 'John Smith H11', 'sarah.hernandez@example.com'),
('H22', 'Jane Doe H11', 'thomas.lee@example.com'),
('H23', 'John Smith H12', 'ursula.lopez@example.com'),
('H24', 'Jane Doe H12', 'victor.martinez@example.com'),
('H25', 'John Smith H13', 'william.nguyen@example.com'),
('H26', 'Jane Doe H13', 'xavier.ortiz@example.com'),
('H27', 'John Smith H14', 'yvonne.perez@example.com'),
('H28', 'Jane Doe H14', 'zoe.quinn@example.com'),
('H29', 'John Smith H15', 'adam.robinson@example.com'),
('H30', 'Jane Doe H15', 'barbara.smith@example.com');

CREATE FUNCTION dbo.FIRST_NAME (@Email NVARCHAR(MAX))
RETURNS NVARCHAR(MAX)
AS
BEGIN
    DECLARE @FirstName NVARCHAR(MAX)
    SET @FirstName = SUBSTRING(@Email, 1, CHARINDEX('.', @Email) - 1)
    RETURN @FirstName
END;

CREATE FUNCTION dbo.LAST_NAME (@Email NVARCHAR(MAX))
RETURNS NVARCHAR(MAX)
AS
BEGIN
    DECLARE @LastName NVARCHAR(MAX)
    SET @LastName = SUBSTRING(@Email,CHARINDEX('.', @Email) + 1,
     CHARINDEX('@', @Email) - CHARINDEX('.', @Email) - 1)
    RETURN @LastName
END;

CREATE  PROCEDURE SP_hierarchy
AS
BEGIN
    TRUNCATE TABLE Employee_Hierarchy;
    WITH EmployeeHierarchyCTE AS (
        SELECT
            e.EmployeeID,
            e.ReportingTo,
            e.EmailID,
            1 AS LEVEL,
            dbo.FIRST_NAME(e.EmailID) AS FIRSTNAME,
            dbo.LAST_NAME(e.EmailID) AS LASTNAME
        FROM
            Employee_Master e
        WHERE
            e.ReportingTo IS NULL
```

```
        UNION ALL
        SELECT
            e.EmployeeID,
            e.ReportingTo,
            e.EmailID,
            eh.LEVEL + 1 AS LEVEL,
            dbo.FIRST_NAME(e.EmailID) AS FIRSTNAME,
            dbo.LAST_NAME(e.EmailID) AS LASTNAME
        FROM
            Employee_Master e
        INNER JOIN
            EmployeeHierarchyCTE eh ON
            (e.ReportingTo LIKE '% ' + eh.EmployeeID OR
             e.ReportingTo LIKE '% ' + eh.EmployeeID + ' %' OR
             e.ReportingTo = eh.EmployeeID)
    )
    INSERT INTO Employee_Hierarchy (EMPLOYEEID, REPORTINGTO, EMAILID, LEVEL,
FIRSTNAME, LASTNAME)
    SELECT DISTINCT
        EmployeeID,
        ReportingTo,
        EmailID,
        LEVEL,
        FIRSTNAME,
        LASTNAME
    FROM
        EmployeeHierarchyCTE
    ORDER BY
        LEVEL, EmployeeID;
END;

EXEC SP_hierarchy;
SELECT * FROM Employee_Hierarchy ORDER BY LEVEL, EMPLOYEEID;
```

4.OUTPUT

Results Messages

	EMPLOYEEID	REPORTINGTO	EMAILID	LEVEL	FIRSTNAME	LASTNAME
1	H1	NULL	john.doe@example.com	1	john	doe
2	H2	NULL	jane.smith@example.com	1	jane	smith
3	H3	John Smith H1	alice.jones@example.com	2	alice	jones
4	H4	Jane Doe H1	bob.white@example.com	2	bob	white
5	H5	John Smith H3	charlie.brown@example.com	3	charlie	brown
6	H6	Jane Doe H3	david.green@example.com	3	david	green
7	H7	John Smith H4	emily.gray@example.com	3	emily	gray
8	H8	Jane Doe H4	frank.wilson@example.com	3	frank	wilson
9	H10	Jane Doe H5	hannah.taylor@example.com	4	hannah	taylor
10	H11	John Smith H6	irene.martin@example.com	4	irene	martin
11	H12	Jane Doe H6	jack.roberts@example.com	4	jack	roberts
12	H13	John Smith H7	kate.evans@example.com	4	kate	evans
13	H14	Jane Doe H7	laura.hall@example.com	4	laura	hall
14	H15	John Smith H8	mike.anderson@example.com	4	mike	anderson
15	H16	Jane Doe H8	natalie.clark@example.com	4	natalie	clark
16	H9	John Smith H5	george.harris@example.com	4	george	harris
17	H17	John Smith H9	oliver.davis@example.com	5	oliver	davis
18	H18	Jane Doe H9	peter.edwards@example.com	5	peter	edwards
19	H19	John Smith H10	quinn.fisher@example.com	5	quinn	fisher
20	H20	Jane Doe H10	rachel.garcia@example.com	5	rachel	garcia
21	H21	John Smith H11	sarah.hernandez@example....	5	sarah	hernandez
22	H22	Jane Doe H11	thomas.lee@example.com	5	thomas	lee
23	H23	John Smith H12	ursula.lopez@example.com	5	ursula	lopez
24	H24	Jane Doe H12	victor.martinez@example.com	5	victor	martinez
25	H25	John Smith H13	william.nguyen@example.com	5	william	nguyen
26	H26	Jane Doe H13	xavier.ortiz@example.com	5	xavier	ortiz
27	H27	John Smith H14	yvonne.perez@example.com	5	yvonne	perez
28	H28	Jane Doe H14	zoe.quinn@example.com	5	zoe	quinn
29	H29	John Smith H15	adam.robinson@example.com	5	adam	robinson