/\***STEP 1:** Create a table (SELF REFERENTIAL), with the following statement \*/

**CREATE TABLE EMPLOYEE(EMPLOYEE\_ID SERIAL CONSTRAINT EMPLOYEE\_PK1 PRIMARY KEY,**

**EMPLOYEE\_FIRST\_NAME CHAR(60) NOT NULL,**

**EMPLOYEE\_MIDDLE\_INITIAL CHAR(1),**

**EMPLOYEE\_LAST\_NAME CHAR(60) NOT NULL,**

**EMPLOYEE\_DESIGNATION CHAR(80),**

**MANAGER\_ID BIGINT, CONSTRAINT EMPLOYEE\_FK FOREIGN KEY (MANAGER\_ID) REFERENCES EMPLOYEE(EMPLOYEE\_ID)**

**);**

/\* **STEP 2:** Populate the table with some sample data \*/

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('SEKHAR',NULL,'MEKALA', 'CEO', NULL);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('DAVID','M','PAUL', 'CTO', 1);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('CRAIG','P','MULLINS', 'CFO', 1);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('NEERAJ','K','SHARMA', 'COO', 1);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('YOGESH',NULL,'CHAWLA', 'SVP, HR', 1);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('NIKITA',NULL,'GANDMAN', 'VP - DBA', 2);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('PRAKASH',NULL,'JHA', 'VP - SYSTEMS', 2);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('JEFF','O','HUNT', 'NETWORK - LEAD', 7);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('ROBERT','L','COOK', 'ORACLE Competency lead', 6);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('JOHN','B','LY', 'DB2 Competency lead', 6);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('SIDDHANTH','S','SIDDHU', 'SQL SERVER Competency lead', 6);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('ANNA','S','JACOB', 'Sr. DBA', 11);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('SANTI','J','LEE', 'Sr. DBA', 9);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('JACOB','L','SCHMIDT', 'VP', 3);**

**INSERT INTO EMPLOYEE**

**(**

**EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_MIDDLE\_INITIAL,**

**EMPLOYEE\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID**

**)**

**VALUES**

**('HANH','L','KIM', 'VP', 3);**

/\* **STEP 3:** The following query will give you all the data currently present in the table\*/

**SELECT \* FROM EMPLOYEE**

/\*The following query will display relationships between all the employees, along with the manager's ID, manager's first and manager's last name and manager's designation \*/

/\*The report starts with CEO, and displays all the employees recursively, following the reporting structure\*/

/\***QUERY 1:**\*/

**WITH RECURSIVE ROOT (EMPLOYEE\_ID,EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME,EMPLOYEE\_DESIGNATION,MANAGER\_ID,**

**LEVEL,MANAGER\_FIRST\_NAME, MANAGER\_LAST\_NAME, MANAGER\_DESIGNATION)**

**AS**

**(**

**SELECT EMPLOYEE\_ID,EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME,EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID, 1 AS LEVEL,**

**EMPLOYEE\_FIRST\_NAME AS MANAGER\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME AS MANAGER\_LAST\_NAME,**

**EMPLOYEE\_DESIGNATION AS MANAGER\_DESIGNATION**

**FROM**

**EMPLOYEE WHERE MANAGER\_ID IS NULL**

**UNION ALL**

**SELECT LEAF.EMPLOYEE\_ID, LEAF.EMPLOYEE\_FIRST\_NAME, LEAF.EMPLOYEE\_LAST\_NAME, LEAF.EMPLOYEE\_DESIGNATION,**

**LEAF.MANAGER\_ID, (ROOT.LEVEL + 1) AS LEVEL, ROOT.EMPLOYEE\_FIRST\_NAME, ROOT.EMPLOYEE\_LAST\_NAME, ROOT.EMPLOYEE\_DESIGNATION MANAGER\_DESIGNATION**

**FROM ROOT, EMPLOYEE LEAF WHERE ROOT.EMPLOYEE\_ID = LEAF.MANAGER\_ID**

**)**

**SELECT EMPLOYEE\_ID,EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME,EMPLOYEE\_DESIGNATION,MANAGER\_ID,**

**CASE WHEN LEVEL= 1 then '---NONE---' ELSE MANAGER\_FIRST\_NAME END AS MANAGER\_FIRST\_NAME,**

**CASE WHEN LEVEL= 1 then '---NONE---' ELSE MANAGER\_LAST\_NAME END AS MANAGER\_LAST\_NAME,**

**CASE WHEN LEVEL= 1 then '---NONE---' ELSE MANAGER\_DESIGNATION END AS MANAGER\_DESIGNATION,**

**LEVEL**

**FROM ROOT**

**ORDER BY LEVEL**

**/\* ADDITIONAL Queries\*/**

/\***QUERY 2:** To get all the managers above a particular employee ID, use this query. The employee ID must be supplied as input\*/

**WITH RECURSIVE ROOT (EMPLOYEE\_ID,EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME,EMPLOYEE\_DESIGNATION,MANAGER\_ID, LEVEL) AS**

**(**

**SELECT EMPLOYEE\_ID,EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME,EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID, 1 AS LEVEL FROM**

**--Supply desired employee ID, for whom you would like to get all the managers above him. Here I used 13 as the employee ID.**

**EMPLOYEE WHERE EMPLOYEE\_ID = 13**

**UNION ALL**

**SELECT LEAF.EMPLOYEE\_ID AS MANAGER\_ID, LEAF.EMPLOYEE\_FIRST\_NAME AS MANAGER\_FIRST\_NAME, LEAF.EMPLOYEE\_LAST\_NAME MANAGER\_LAST\_NAME, LEAF.EMPLOYEE\_DESIGNATION AS MANAGER\_TITLE,**

**LEAF.MANAGER\_ID, (ROOT.LEVEL + 1) AS LEVEL**

**FROM ROOT, EMPLOYEE LEAF WHERE ROOT.MANAGER\_ID = LEAF.EMPLOYEE\_ID**

**)**

**SELECT \* FROM ROOT ORDER BY LEVEL DESC**

/\* **QUERY 3:** To get all the employees below a particular employee ID, use this query. The employee ID must be supplied as input to the query\*/

**WITH RECURSIVE ROOT (EMPLOYEE\_ID,EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME,EMPLOYEE\_DESIGNATION,MANAGER\_ID, LEVEL) AS**

**(**

**SELECT EMPLOYEE\_ID,EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME,EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID, 1 AS LEVEL FROM**

**--Supply desired employee ID, for whom you would like to get all the employees below him. Here I used 2 as the EMPLOYEE ID.**

**EMPLOYEE WHERE EMPLOYEE\_ID = 2**

**UNION ALL**

**SELECT LEAF.EMPLOYEE\_ID AS MANAGER\_ID, LEAF.EMPLOYEE\_FIRST\_NAME AS MANAGER\_FIRST\_NAME, LEAF.EMPLOYEE\_LAST\_NAME MANAGER\_LAST\_NAME, LEAF.EMPLOYEE\_DESIGNATION AS MANAGER\_TITLE,**

**LEAF.MANAGER\_ID, (ROOT.LEVEL + 1) AS LEVEL**

**FROM ROOT, EMPLOYEE LEAF WHERE ROOT.EMPLOYEE\_ID = LEAF.MANAGER\_ID**

**)**

**SELECT \* FROM ROOT ORDER BY LEVEL**

/\***QUERY 4:** To get all the reporting relationships between all employees, use this query. This query is similar to QUERY 1,

but it does NOT show the manager's name, and manager's designation details, unlike in query 1 above\*/

**WITH RECURSIVE ROOT (EMPLOYEE\_ID,EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME,EMPLOYEE\_DESIGNATION,MANAGER\_ID, LEVEL) AS**

**(**

**SELECT EMPLOYEE\_ID,EMPLOYEE\_FIRST\_NAME,**

**EMPLOYEE\_LAST\_NAME,EMPLOYEE\_DESIGNATION,**

**MANAGER\_ID, 1 AS LEVEL FROM**

**EMPLOYEE WHERE MANAGER\_ID IS NULL**

**UNION ALL**

**SELECT LEAF.EMPLOYEE\_ID, LEAF.EMPLOYEE\_FIRST\_NAME, LEAF.EMPLOYEE\_LAST\_NAME, LEAF.EMPLOYEE\_DESIGNATION,**

**LEAF.MANAGER\_ID, (ROOT.LEVEL + 1) AS LEVEL**

**FROM ROOT, EMPLOYEE LEAF WHERE ROOT.EMPLOYEE\_ID = LEAF.MANAGER\_ID**

**)**

**SELECT \* FROM ROOT ORDER BY LEVEL**