5. Consider the schema for Company Database:
EMPLOYEE (SSN, Name, Address, Sex, Salary, SuperSSN, DNo)
DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate)
**DLOCATION (DNo,DLoc)**
**PROJECT (PNo, PName, PLocation, DNo)**
WORKS_ON (SSN, PNo, Hours)

Write SQL queries to
1. Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or as a manager of the department that controls the project.
2. Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.
3. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department
4 Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator).

5. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000

Entity-Relationship Diagram
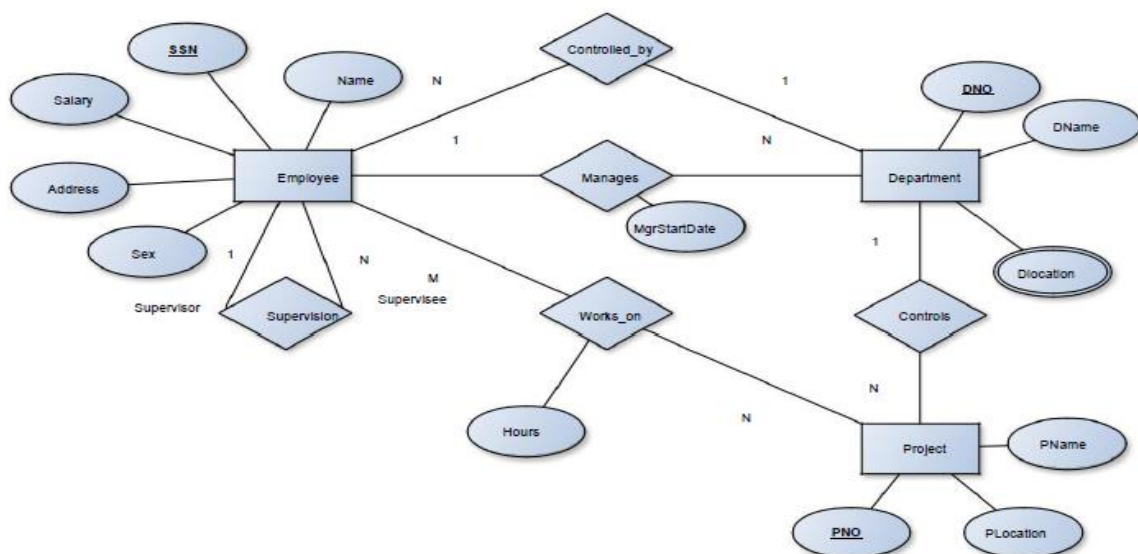
Schema Diagram
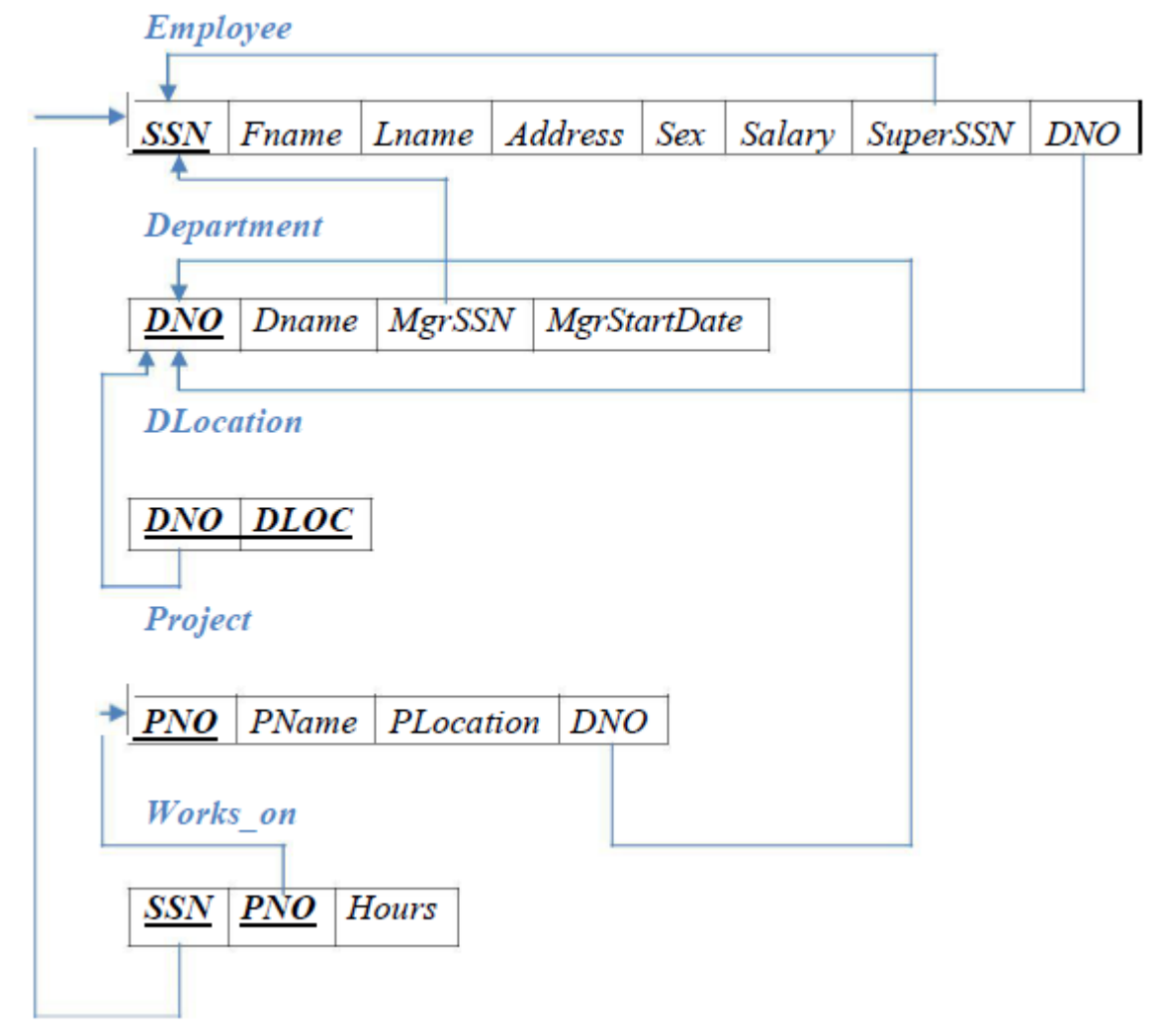
TABLE CREATION

Create Table DEPARTMENT with PRIMARY KEY as DNO

CREATE TABLE DEPARTMENT

(DNO VARCHAR(20) PRIMARY KEY,

DNAME VARCHAR(20),

MGR_SSN VARCHAR(20),

MGR_START_DATE DATE);

DESC DEPARTMENT;

Create Table EMPLOYEE with PRIMARY KEY as SSN

CREATE TABLE EMPLOYEE

(SSN VARCHAR(20) PRIMARY KEY,

NAME VARCHAR(20),

ADDRESS VARCHAR(20),

SEX CHAR(1),

SALARY INTEGER,

SUPERSSN VARCHAR(20),

DNO VARCHAR(20),

FOREIGN KEY (SUPERSSN) REFERENCES EMPLOYEE (SSN),

FOREIGN KEY (DNO) REFERENCES DEPARTMENT (DNO));

DESC EMPLOYEE;

ADD FOREIGN KEY Constraint to DEPARTMENT table

ALTER TABLE DEPARTMENT

ADD FOREIGN KEY (MGR_SSN) REFERENCES EMPLOYEE(SSN);

Create Table DLOCATION with PRIMARY KEY as DNO and DLOC and FOREIGN KEY DNO referring DEPARTMENT table

CREATE TABLE DLOCATION

(DLOC VARCHAR(20),

DNO VARCHAR(20),

FOREIGN KEY (DNO) REFERENCES DEPARTMENT(DNO),

PRIMARY KEY (DNO, DLOC));

DESC DLOCATION;

Create Table PROJECT with PRIMARY KEY as PNO and FOREIGN KEY DNO referring DEPARTMENT table

CREATE TABLE PROJECT

(PNO INTEGER PRIMARY KEY,

PNAME VARCHAR(20),

PLOCATION VARCHAR(20),

DNO VARCHAR(20),

FOREIGN KEY (DNO) REFERENCES DEPARTMENT(DNO));

DESC PROJECT;

Create Table WORKS_ON with PRIMARY KEY as PNO and SSN and FOREIGN KEY SSN and PNO referring EMPLOYEE and PROJECT table

CREATE TABLE WORKS_ON

(HOURS INTEGER,

SSN VARCHAR(20),

PNO INTEGER,

FOREIGN KEY (SSN) REFERENCES EMPLOYEE(SSN),

FOREIGN KEY (PNO) REFERENCES PROJECT(PNO),

PRIMARY KEY (SSN, PNO));

DESC WORKS_ON;

<u>INSERTION VALUES</u>

Inserting records into EMPLOYEE table

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES
 ('ABC01','BEN SCOTT','BANGALORE','M', 450000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES
 ('ABC02','HARRY SMITH','BANGALORE','M', 500000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES
 ('ABC03','LEAN BAKER','BANGALORE','M', 700000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES
 ('ABC04','MARTIN SCOTT','MYSORE','M', 500000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES
 ('ABC05','RAVAN HEGDE','MANGALORE','M', 650000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES
 ('ABC06','GIRISH HOSUR','MYSORE','M', 450000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES
 ('ABC07','NEELA SHARMA','BANGALORE','F', 800000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES
 ('ABC08','ADYA KOLAR','MANGALORE','F', 350000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES
 ('ABC09','PRASANNA KUMAR','MANGALORE','M', 300000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES

('ABC10','VEENA KUMARI','MYSORE','M', 600000);

INSERT INTO EMPLOYEE (SSN, NAME, ADDRESS, SEX, SALARY) VALUES

('ABC11','DEEPAK RAJ','BANGALORE','M', 500000);

SELECT * FROM EMPLOYEE;

Inserting records into DEPARTMENT table

INSERT INTO DEPARTMENT VALUES ('1','ACCOUNTS','ABC09', '2016-01-03');

INSERT INTO DEPARTMENT VALUES ('2','IT','ABC11', '2017-02-04');

INSERT INTO DEPARTMENT VALUES ('3','HR','ABC01', '2016-04-05');

INSERT INTO DEPARTMENT VALUES ('4','HELPDESK', 'ABC10', '2017-06-03');

INSERT INTO DEPARTMENT VALUES ('5','SALES','ABC06', '2017-01-08');

SELECT * FROM DEPARTMENT;

Updating EMPLOYEE records

UPDATE EMPLOYEE SET
SUPERSSN=NULL, DNO='3'
WHERE SSN='ABC01';

UPDATE EMPLOYEE SET
SUPERSSN='ABC03', DNO='5'
WHERE SSN='ABC02';

```sql
UPDATE EMPLOYEE SET
SUPERSSN='ABC04', DNO='5'
WHERE SSN='ABC03';


UPDATE EMPLOYEE SET
SUPERSSN='ABC06', DNO='5'
WHERE SSN='ABC04';


UPDATE EMPLOYEE SET
DNO='5', SUPERSSN='ABC06'
WHERE SSN='ABC05';


UPDATE EMPLOYEE SET
DNO='5', SUPERSSN='ABC07'
WHERE SSN='ABC06';


UPDATE EMPLOYEE SET
DNO='5', SUPERSSN=NULL
WHERE SSN='ABC07';


UPDATE EMPLOYEE SET
DNO='1', SUPERSSN='ABC09'
WHERE SSN='ABC08';


UPDATE EMPLOYEE SET
DNO='1', SUPERSSN=NULL
WHERE SSN='ABC09';


UPDATE EMPLOYEE SET
DNO='4', SUPERSSN=NULL
```

```
WHERE SSN='ABC10';


UPDATE EMPLOYEE SET
DNO='2', SUPERSSN=NULL
WHERE SSN='ABC11';


SELECT * FROM EMPLOYEE;
```

Inserting records into DLOCATION table

```
INSERT INTO DLOCATION VALUES ('BENGALURU', '1');
INSERT INTO DLOCATION VALUES ('BENGALURU', '2');
INSERT INTO DLOCATION VALUES ('BENGALURU', '3');
INSERT INTO DLOCATION VALUES ('MYSORE', '4');
INSERT INTO DLOCATION VALUES ('MYSORE', '5');


SELECT * FROM DLOCATION;
```

Inserting records into PROJECT table

```
INSERT INTO PROJECT VALUES (1000,'IOT','BENGALURU','5');
INSERT INTO PROJECT VALUES (1001,'CLOUD','BENGALURU','5');
INSERT INTO PROJECT VALUES (1002,'BIGDATA','BENGALURU','5');
INSERT INTO PROJECT VALUES (1003,'SENSORS','BENGALURU','3');
INSERT INTO PROJECT VALUES (1004,'BANK MANAGEMENT','BENGALURU','1');
INSERT INTO PROJECT VALUES (1005,'SALARY
MANAGEMENT','BANGALORE','1');
```

INSERT INTO PROJECT VALUES (1006,'OPENSTACK','BENGALURU','4');

INSERT INTO PROJECT VALUES (1007,'SMART CITY','BENGALURU','2');

SELECT * FROM PROJECT;

Inserting records into WORKS_ON table

INSERT INTO WORKS_ON VALUES (4, 'ABC02', 1000);

INSERT INTO WORKS_ON VALUES (6, 'ABC02', 1001);

INSERT INTO WORKS_ON VALUES (8, 'ABC02', 1002);

INSERT INTO WORKS_ON VALUES (10,'ABC03', 1000);

INSERT INTO WORKS_ON VALUES (3, 'ABC05', 1000);

INSERT INTO WORKS_ON VALUES (4, 'ABC06', 1001);

INSERT INTO WORKS_ON VALUES (5, 'ABC07', 1002);

INSERT INTO WORKS_ON VALUES (6, 'ABC04', 1002);

INSERT INTO WORKS_ON VALUES (7, 'ABC01', 1003);

INSERT INTO WORKS_ON VALUES (5, 'ABC08', 1004);

INSERT INTO WORKS_ON VALUES (6, 'ABC09', 1005);

INSERT INTO WORKS_ON VALUES (4, 'ABC10', 1006);

INSERT INTO WORKS_ON VALUES (10,'ABC11', 1007);

SELECT * FROM WORKS_ON;

QUERIES

1. Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or as a manager of the department that controls the project.

SELECT DISTINCT P.PNO

FROM PROJECT P, DEPARTMENT D, EMPLOYEE E

WHERE E.DNO=D.DNO

AND D.MGR_SSN=E.SSN

AND E.NAME LIKE '%SCOTT'

UNION

SELECT DISTINCT P1.PNO

FROM PROJECT P1, WORKS_ON W, EMPLOYEE E1

WHERE P1.PNO=W.PNO

AND E1.SSN=W.SSN

AND E1.NAME LIKE '%SCOTT';

2. Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.

SELECT E.NAME, 1.1*E.SALARY AS INCR_SAL

FROM EMPLOYEE E, WORKS_ON W, PROJECT P

WHERE E.SSN=W.SSN

AND W.PNO=P.PNO

AND P.PNAME='IOT';

3. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department

SELECT SUM(E.SALARY), MAX(E.SALARY), MIN(E.SALARY), AVG(E.SALARY)

FROM EMPLOYEE E, DEPARTMENT D

WHERE E.DNO=D.DNO

AND D.DNAME='ACCOUNTS';

4. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator).

SELECT E.NAME

FROM EMPLOYEE E

WHERE NOT EXISTS(SELECT PNO FROM PROJECT WHERE DNO='5' AND PNO NOT IN (SELECT

PNO FROM WORKS_ON

WHERE E.SSN=SSN));

5. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.

SELECT D.DNO, COUNT(*)

FROM DEPARTMENT D, EMPLOYEE E

WHERE D.DNO=E.DNO

AND E.SALARY > 600000

AND D.DNO IN (SELECT E1.DNO

FROM EMPLOYEE E1

GROUP BY E1.DNO

HAVING COUNT(*)>5)

GROUP BY D.DNO;