A. 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 employeeworking on the 'IoT' project is given a 10 percent raise.
3. Find the sum Of the salaries of all the 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.

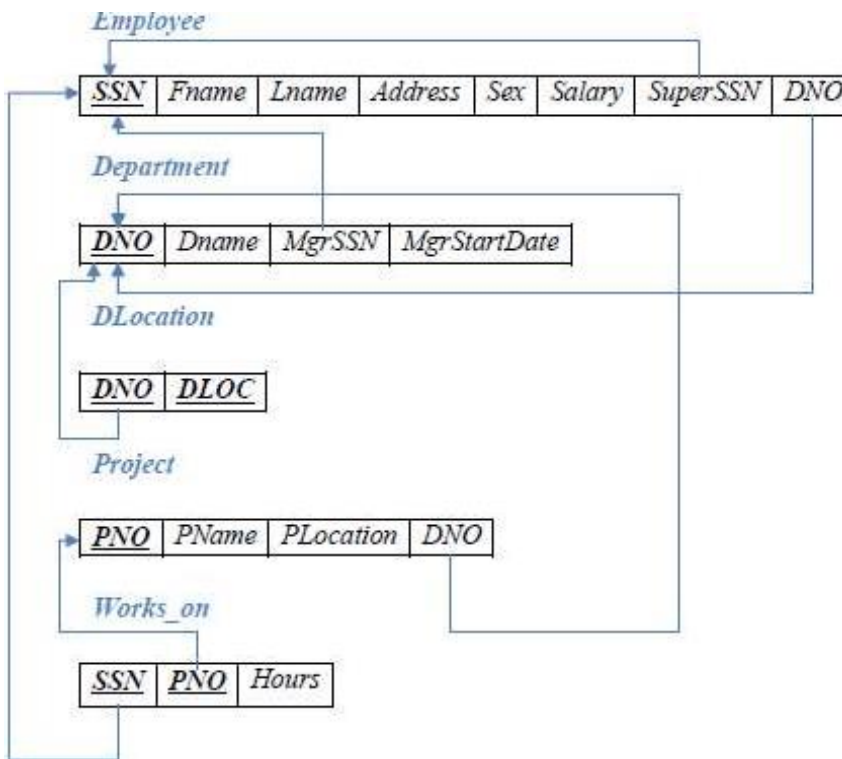Solution:
Entity-Relationship Diagram
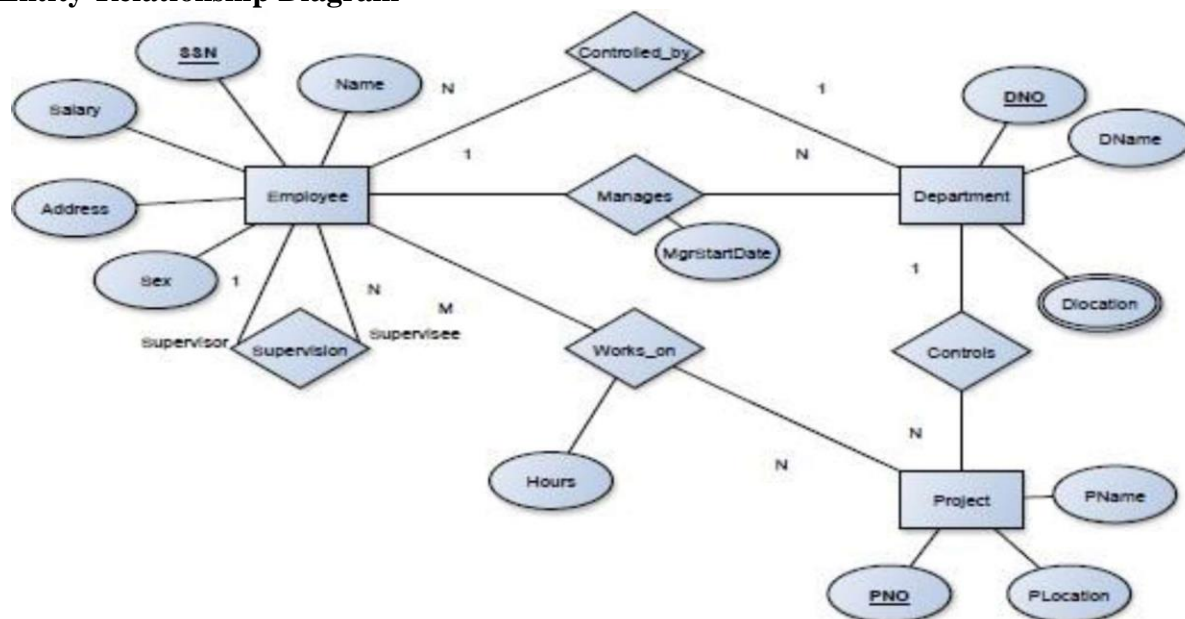
Schema Diagram

Table Creation

```
CREATE TABLE DEPARTMENT (
DNO VARCHAR (20) PRIMARY KEY,
DNAME VARCHAR (20),
MGRSTARTDATE DATE,
MGRSSN VARCHAR (20));

CREATE TABLE EMPLOYEE (
SSN VARCHAR (20) PRIMARY KEY,
FNAME VARCHAR (20),
LNAME VARCHAR (20),
ADDRESS VARCHAR (100),
SEX CHAR (1),
SALARY INT (10),
SUPERSSN VARCHAR (20),
DNO VARCHAR (20),
FOREIGN KEY (SUPERSSN) REFERENCES EMPLOYEE (SSN),
FOREIGN KEY (DNO) REFERENCES DEPARTMENT (DNO));
```

NOTE: Once DEPARTMENT and EMPLOYEE tables are created we must alter department table to add foreign constraint MGRSSN using sql command
ALTER TABLE DEPARTMENT ADD FOREIGN KEY(MGRSSN) REFERENCES EMPLOYEE(SSN);

CREATE TABLE DLOCATION (
DLOC VARCHAR (20),
DNO VARCHAR (20),
PRIMARY KEY (DNO, DLOC),
FOREIGN KEY (DNO) REFERENCES DEPARTMENT (DNO));

CREATE TABLE PROJECT (
PNO INT (10) PRIMARY KEY,
PNAME VARCHAR (20),
PLOCATION VARCHAR (20),
DNO VARCHAR (20),
FOREIGN KEY (DNO) REFERENCES DEPARTMENT (DNO));

CREATE TABLE WORKS_ON (
HOURS INT (4),
SSN VARCHAR (20),
PNO INT (10),
PRIMARY KEY (SSN, PNO),
FOREIGN KEY (SSN) REFERENCES EMPLOYEE (SSN),
FOREIGN KEY (PNO) REFERENCES PROJECT (PNO));

Table Descriptions
DESC EMPLOYEE;

```
mysql> DESC EMPLOYEE;
+----------+--------------+------+-----+---------+-------+
| Field    | Type         | Null | Key | Default | Extra |
+----------+--------------+------+-----+---------+-------+
| SSN      | varchar(20)  | NO   | PRI | NULL    |       |
| FNAME    | varchar(20)  | YES  |     | NULL    |       |
| LNAME    | varchar(20)  | YES  |     | NULL    |       |
| ADDRESS  | varchar(100) | YES  |     | NULL    |       |
| SEX      | char(1)      | YES  |     | NULL    |       |
| SALARY   | int(10)      | YES  |     | NULL    |       |
| SUPERSSN | varchar(20)  | YES  | MUL | NULL    |       |
| DNO      | varchar(20)  | YES  | MUL | NULL    |       |
+----------+--------------+------+-----+---------+-------+
8 rows in set (0.00 sec)
```

DESC DEPARTMENT;

```
mysql> DESC DEPARTMENT;
+--------------+-------------+------+-----+---------+-------+
| Field        | Type        | Null | Key | Default | Extra |
+--------------+-------------+------+-----+---------+-------+
| DNO          | varchar(20) | NO   | PRI | NULL    |       |
| DNAME        | varchar(20) | YES  |     | NULL    |       |
| MGRSTARTDATE | date        | YES  |     | NULL    |       |
| MGRSSN       | varchar(20) | YES  | MUL | NULL    |       |
+--------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

DESC DLOCATION;

```
mysql> DESC DLOCATION;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| DLOC  | varchar(20) | NO   | PRI |         |       |
| DNO   | varchar(20) | NO   | PRI |         |       |
+-------+-------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

DESC PROJECT;

```
mysql> DESC PROJECT;
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| PNO       | int(10)     | NO   | PRI | NULL    |       |
| PNAME     | varchar(20) | YES  |     | NULL    |       |
| PLOCATION | varchar(20) | YES  |     | NULL    |       |
| DNO       | varchar(20) | YES  | MUL | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

DESC WORKS_ON;

```
mysql> DESC WORKS_ON;
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| HOURS | int(4)      | YES  |     | NULL    |       |
| SSN   | varchar(20) | NO   | PRI |         |       |
| PNO   | int(10)     | NO   | PRI | 0       |       |
+-------+-------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

Insertion of values to tables
INSERT INTO EMPLOYEE VALUES ('ATMEECE01','JOHN','SCOTT','BANGALORE','M',
450000,NULL,NULL);
INSERT INTO EMPLOYEE VALUES ('ATMECSE01','JAMES','SMITH','BANGALORE','M',
500000,NULL,NULL);
INSERT INTO EMPLOYEE VALUES ('ATMECSE02','HEARN','BAKER','BANGALORE','M',
700000,NULL,NULL);
INSERT INTO EMPLOYEE VALUES ('ATMECSE03','EDWARD','SCOTT','MYSORE','M',
500000,NULL,NULL);
INSERT INTO EMPLOYEE VALUES ('ATMECSE04','PAVAN','HEGDE','MANGALORE','M',
650000,NULL,NULL);
INSERT INTO EMPLOYEE VALUES ('ATMECSE05','GIRISH','MALYA','MYSORE','M',
450000,NULL,NULL);
INSERT INTO EMPLOYEE VALUES ('ATMECSE06','NEHA','SN','BANGALORE','F',
800000,NULL,NULL);
INSERT INTO EMPLOYEE VALUES ('ATMEACC01','AHANA','K','MANGALORE','F',
350000,NULL,NULL);
INSERT INTO EMPLOYEE VALUES
('ATMEACC02','SANTHOSH','KUMAR','MANGALORE','M', 300000,NULL,NULL);
INSERT INTO EMPLOYEE VALUES ('ATMEISE01','VEENA','M','MYSORE','F',
600000,NULL,NULL);

```sql
INSERT INTO EMPLOYEE  VALUES ('ATMEIT01','NAGESH','HR','BANGALORE','M',
500000,NULL,NULL);

INSERT INTO DEPARTMENT VALUES ('1','ACCOUNTS','2001-01-01','ATMEACC02');
INSERT INTO DEPARTMENT VALUES ('2','IT','2016-08-01','ATMEIT01');
INSERT INTO DEPARTMENT VALUES ('3','ECE','2008-6-01','ATMEECE01');
INSERT INTO DEPARTMENT VALUES ('4','ISE','2015-06-01','ATMEISE01');
INSERT INTO DEPARTMENT VALUES ('5','CSE','2002-06-01','ATMECSE05');
```

Note: update entries of employee table to fill missing fields SUPERSSN and DNO

```sql
UPDATE EMPLOYEE SET SUPERSSN='ATMECSE02', DNO='5' WHERE
SSN='ATMECSE01';

UPDATE EMPLOYEE SET SUPERSSN='ATMECSE03', DNO='5' WHERE SSN='ATMECSE02';

UPDATE EMPLOYEE SET SUPERSSN='ATMECSE04', DNO='5' WHERE SSN='ATMECSE03';

UPDATE EMPLOYEE SET DNO='5', SUPERSSN='ATMECSE05' WHERE SSN='ATMECSE04';

UPDATE EMPLOYEE SET DNO='5', SUPERSSN='ATMECSE06' WHERE SSN='ATMECSE05';

UPDATE EMPLOYEE SET DNO='5', SUPERSSN=NULL WHERE SSN='ATMECSE06';

UPDATE EMPLOYEE SET DNO='1', SUPERSSN='ATMEACC02' WHERE SSN='ATMEACC01';

UPDATE EMPLOYEE SET DNO='1', SUPERSSN=NULL WHERE
SSN='ATMEACC02';

UPDATE EMPLOYEE SET DNO='4', SUPERSSN=NULL WHERE
SSN='ATMEISE01';

UPDATE EMPLOYEE SET DNO='2', SUPERSSN=NULL WHERE
SSN='ATMEIT01';

INSERT INTO  DLOCATION VALUES ('BANGALORE', '1');
INSERT INTO  DLOCATION VALUES ('BANGALORE', '2');
INSERT INTO  DLOCATION VALUES ('BANGALORE', '3');
INSERT INTO  DLOCATION VALUES ('MANGALORE', '4');
INSERT INTO  DLOCATION VALUES ('MANGALORE', '5');

INSERT INTO PROJECT VALUES (100,'IOT','BANGALORE','5');
INSERT INTO PROJECT VALUES (101,'CLOUD','BANGALORE','5');
INSERT INTO PROJECT VALUES (102,'BIGDATA','BANGALORE','5');
INSERT INTO PROJECT VALUES (103,'SENSORS','BANGALORE','3');
INSERT INTO PROJECT VALUES (104,'BANK MANAGEMENT','BANGALORE','1');
INSERT INTO PROJECT VALUES (105,'SALARY MANAGEMENT','BANGALORE','1');
```

INSERT INTO PROJECT VALUES (106,'OPENSTACK','BANGALORE','4');
INSERT INTO PROJECT VALUES (107,'SMART CITY','BANGALORE','2');

INSERT INTO WORKS_ON VALUES (4, 'ATMECSE01', 100);
INSERT INTO WORKS_ON VALUES (6, 'ATMECSE01', 101);
INSERT INTO WORKS_ON VALUES (8, 'ATMECSE01', 102);
INSERT INTO WORKS_ON VALUES (10, 'ATMECSE02', 100);
INSERT INTO WORKS_ON VALUES (3, 'ATMECSE04', 100);
INSERT INTO WORKS_ON VALUES (4, 'ATMECSE05', 101);
INSERT INTO WORKS_ON VALUES (5, 'ATMECSE06', 102);
INSERT INTO WORKS_ON VALUES (6, 'ATMECSE03', 102);
INSERT INTO WORKS_ON VALUES (7, 'ATMEECE01', 103);
INSERT INTO WORKS_ON VALUES (5, 'ATMEACC01', 104);
INSERT INTO WORKS_ON VALUES (6, 'ATMEACC02', 105);
INSERT INTO WORKS_ON VALUES (4, 'ATMEISE01', 106);
INSERT INTO WORKS_ON VALUES (10, 'ATMEIT01', 107);

SELECT * FROM EMPLOYEE;

```
mysql> SELECT * FROM EMPLOYEE;
+-----------+----------+--------+-----------+-----+--------+-----------+-----+
| SSN       | FNAME    | LNAME  | ADDRESS   | SEX | SALARY | SUPERSSN  | DNO |
+-----------+----------+--------+-----------+-----+--------+-----------+-----+
| ATMEACC01 | AHANA    | K      | MANGALORE | F   | 350000 | ATMEACC02 | 1   |
| ATMEACC02 | SANTHOSH | KUMAR  | MANGALORE | M   | 300000 | NULL      | 1   |
| ATMECSE01 | JAMES    | SMITH  | BANGALORE | M   | 500000 | ATMECSE02 | 5   |
| ATMECSE02 | HEARN    | BAKER  | BANGALORE | M   | 700000 | ATMECSE03 | 5   |
| ATMECSE03 | EDWARD   | SCOTT  | MYSORE    | M   | 500000 | ATMECSE04 | 5   |
| ATMECSE04 | PAVAN    | HEGDE  | MANGALORE | M   | 650000 | ATMECSE05 | 5   |
| ATMECSE05 | GIRISH   | MALYA  | MYSORE    | M   | 450000 | ATMECSE06 | 5   |
| ATMECSE06 | NEHA     | SN     | BANGALORE | F   | 800000 | NULL      | 5   |
| ATMEECE01 | JOHN     | SCOTT  | BANGALORE | M   | 450000 | NULL      | 3   |
| ATMEISE01 | VEENA    | M      | MYSORE    | F   | 600000 | NULL      | 4   |
| ATMEIT01  | NAGESH   | HR     | BANGALORE | M   | 500000 | NULL      | 2   |
+-----------+----------+--------+-----------+-----+--------+-----------+-----+
11 rows in set (0.00 sec)

mysql>
```

SELECT * FROM DEPARTMENT ;

```
mysql> SELECT * FROM  DEPARTMENT;
+-----+----------+-------------+-----------+
| DNO | DNAME    | MGRSTARTDATE | MGRSSN   |
+-----+----------+-------------+-----------+
| 1   | ACCOUNTS | 2001-01-01  | ATMEACC02 |
| 2   | IT       | 2016-08-01  | ATMEIT01  |
| 3   | ECE      | 2008-06-01  | ATMEECE01 |
| 4   | ISE      | 2015-06-01  | ATMEISE01 |
| 5   | CSE      | 2002-06-01  | ATMECSE05 |
+-----+----------+-------------+-----------+
5 rows in set (0.02 sec)
```

SELECT * FROM   DLOCATION ;

```
mysql> SELECT * FROM  DLOCATION;
+-----------+-----+
| DLOC      | DNO |
+-----------+-----+
| BANGALORE | 1   |
| BANGALORE | 2   |
| BANGALORE | 3   |
| MANGALORE | 4   |
| MANGALORE | 5   |
+-----------+-----+
5 rows in set (0.03 sec)
```

SELECT * FROM PROJECT ;

```
mysql> SELECT * FROM  PROJECT;
+-----+------------------+-----------+-----+
| PNO | PNAME            | PLOCATION | DNO |
+-----+------------------+-----------+-----+
| 100 | IOT              | BANGALORE | 5   |
| 101 | CLOUD            | BANGALORE | 5   |
| 102 | BIGDATA          | BANGALORE | 5   |
| 103 | SENSORS          | BANGALORE | 3   |
| 104 | BANK MANAGEMENT  | BANGALORE | 1   |
| 105 | SALARY MANAGEMENT| BANGALORE | 1   |
| 106 | OPENSTACK        | BANGALORE | 4   |
| 107 | SMART CITY       | BANGALORE | 2   |
+-----+------------------+-----------+-----+
8 rows in set (0.03 sec)
```

SELECT * FROM WORKS_ON

```
mysql> SELECT * FROM  WORKS_ON;
+-------+-----------+-----+
| HOURS | SSN       | PNO |
+-------+-----------+-----+
|     5 | ATMEACC01 | 104 |
|     6 | ATMEACC02 | 105 |
|     4 | ATMECSE01 | 100 |
|     6 | ATMECSE01 | 101 |
|     8 | ATMECSE01 | 102 |
|    10 | ATMECSE02 | 100 |
|     6 | ATMECSE03 | 102 |
|     3 | ATMECSE04 | 100 |
|     4 | ATMECSE05 | 101 |
|     5 | ATMECSE06 | 102 |
|     7 | ATMEECE01 | 103 |
|     4 | ATMEISE01 | 106 |
|    10 | ATMEIT01  | 107 |
+-------+-----------+-----+
13 rows in set (0.00 sec)
```

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.MGRSSN=E.SSN AND E.LNAME='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.LNAME='SCOTT')

```
+------+
| PNO |
+------+
| 100 |
| 101 |
| 102 |
| 103 |
| 104 |
| 105 |
| 106 |
| 107 |
+------+
8 rows in set (0.00 sec)
```

2. **Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.**
   SELECT E.FNAME, E.LNAME, 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';

```
+--------+--------+-----------+
| FNAME | LNAME | INCR_SAL |
+--------+--------+-----------+
| JAMES | SMITH | 550000.0 |
| HEARN | BAKER | 770000.0 |
| PAVAN | HEGDE | 715000.0 |
+--------+--------+-----------+
3 rows in set (0.01 sec)
```

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';

```
+----------------+----------------+----------------+----------------+
| SUM(E.SALARY) | MAX(E.SALARY) | MIN(E.SALARY) | AVG(E.SALARY) |
+----------------+----------------+----------------+----------------+
|        650000 |        350000 |        300000 | 325000.0000 |
+----------------+----------------+----------------+----------------+
1 row in set (0.00 sec)
```

4. **Retrieve the name of each employee who works on all the projects Controlled by department number 5 (use NOT EXISTS operator).**
SELECT E.FNAME,E.LNAME FROM EMPLOYEE E WHERE NOT EXISTS (SELECT PNO FROM PROJECT P WHERE DNO=5 AND PNO NOT IN (SELECT PNO FROM WORKS_ON W WHERE E.SSN=SSN));

```
+-------+-------+
| FNAME | LNAME |
+-------+-------+
| JAMES | SMITH |
+-------+-------+
1 row in set (0.00 sec)
```

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;

```
+-----+----------+
| DNO | COUNT(*) |
+-----+----------+
| 5   |        3 |
+-----+----------+
1 row in set (0.00 sec)
```