LAB EXERCISE-5

COMPANY DATABASE

Consider the schema for Company Database:

EMPLOYEE(SSN, Name, Address, Sex, Salary, SuperSSN, DNo)

DEPARTMENT(DNo, Dname, MgrSSN, MgrStartDate)

DLOCATION(DNo, DLoc)

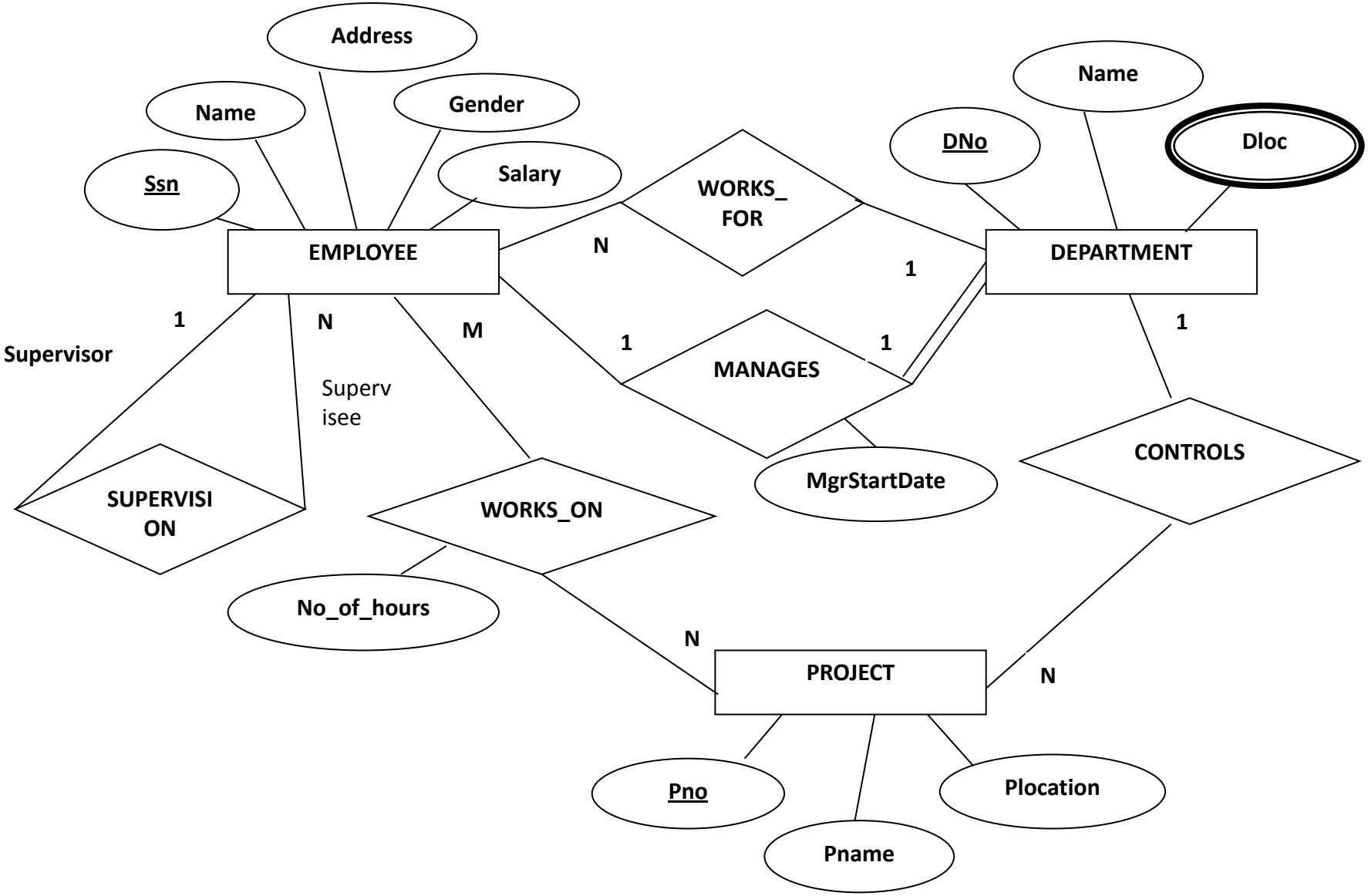
PROJECT(PNo, PName, PLocation, DNo)

WORKS_ON(SSN, PNo, Hours)

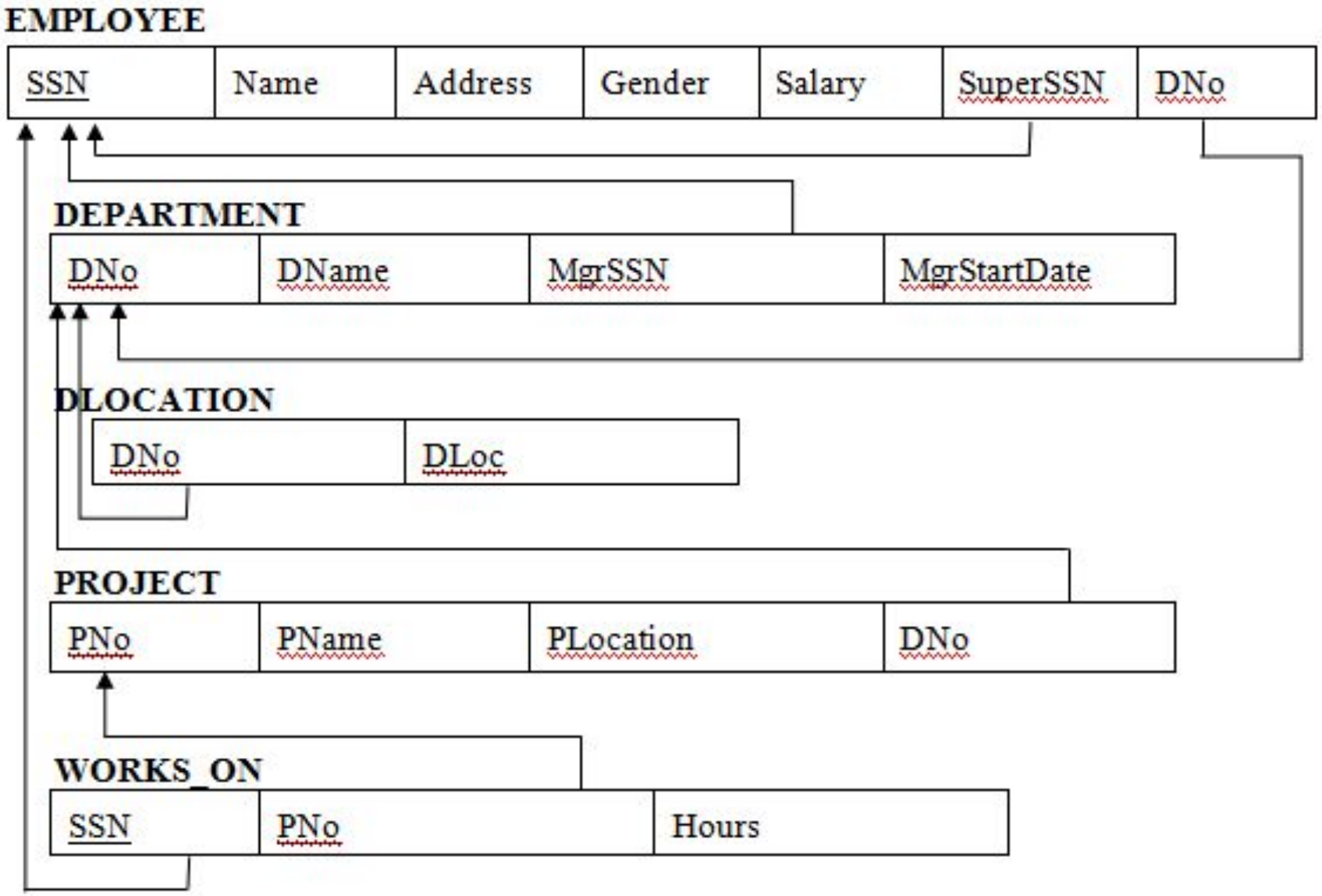
Write SQL queries to

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 (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.**

ER diagram


Address
Name
Gender
Ssn
Salary
EMPLOYEE
WORKS_
FOR
N
1
DEPARTMENT
Name
DNo
Dloc
1
N
M
Supervisor
Superv
isee
1
1
MANAGES
1
SUPERVISI
ON
WORKS_ON
MgrStartDate
CONTROLS
No_of_hours
N
PROJECT
N
Pno
Plocation
Pname

Schema diagram



```
CREATE TABLE EMPLOYEE(
   SSN INT PRIMARY KEY,
   NAME VARCHAR2(20),
   ADDRESS VARCHAR2(20),
   GENDER CHAR(1) CHECK(GENDER='M' OR
   GENDER='F'),
   SALARY NUMBER(6),
   SUPERSSN REFERENCES EMPLOYEE(SSN),
   DNO NUMBER);
```

- INSERT INTO EMPLOYEE

  VALUES(1,'Scott','Mangaluru','F',35000,1,NULL);
- INSERT INTO EMPLOYEE

  VALUES(2,'Sahana','Mangaluru','F',35000,1,NULL);
- INSERT INTO EMPLOYEE

  VALUES(3,'Sagar','Bengaluru','M',35000,1,NULL);
- INSERT INTO EMPLOYEE

  VALUES(4,'Sagarik','Mangaluru','M',35000,1,NULL);
- INSERT INTO EMPLOYEE

  VALUES(5,'Sajaan','Mysore','M',600000,1,NULL);

```
SELECT *FROM EMPLOYEE_PGM;
```

SSN	NAME	ADDRESS	G	SALARY	SUPERSSN	DNO
1	Scott	Mangaluru	F	35000	1	4
2	Sahana	Mangaluru	F	35000	1	1
3	Sagar	Bengaluru	M	35000	1	3
4	Sagarik	Mangaluru	M	35000	1	3
5	Sajaan	Mysore	M	600000	1	3

CREATE TABLE DEPARTMENT_PGM
(DNO NUMBER(5) PRIMARY KEY,
DNAME VARCHAR2(10) ,
MGRSSN REFERENCES EMPLOYEE_PGM,
MGRSTARTDATE DATE);

- INSERT INTO DEPARTMENT_PGM VALUES(1,'CSE',1,'2-Nov-2007');
- INSERT INTO DEPARTMENT_PGM VALUES(2,'IOT',2,'2-Nov-2007');
- INSERT INTO DEPARTMENT_PGM VALUES(3,'Account',2,'2-Nov-2017');
- INSERT INTO DEPARTMENT_PGM VALUES(4,'ISE',1,'2-Nov-2000');
- INSERT INTO DEPARTMENT_PGM VALUES(5,'Finance',1,'3-Nov-2001');

```
SQL> SELECT *FROM DEPARTMENT_PGM;

       DNO DNAME           MGRSSN MGRSTARTD
---------- ---------- ---------- ----------
         1 CSE                 1 02-NOV-07
         2 IOT                 2 02-NOV-07
         3 Account             2 02-NOV-17
         4 ISE                 1 02-NOV-00
         5 Finance             1 03-NOV-01
```

Relating employee and department

- **ALTER TABLE EMPLOYEE_PGM** ADD CONSTRAINT FK FOREIGN KEY(DNO) REFERENCES DEPARTMENT_PGM;

- **Now update employee to set dno:**
- UPDATE EMPLOYEE_PGM SET DNO=&DNO where SSN=&SSN;

- CREATE TABLE **DLOCATION**
  (DNO REFERENCES DEPARTMENT_PGM,
  LOCATION VARCHAR2(10),
  PRIMARY KEY(DNO,LOCATION));

- INSERT INTO DLOCATION VALUES(1,'Mangaluru');
- INSERT INTO DLOCATION VALUES(1,'Mysore');
- INSERT INTO DLOCATION VALUES(2,'Mangaluru');
- INSERT INTO DLOCATION VALUES(3,'Bengaluru');
- INSERT INTO DLOCATION VALUES(4,'Mangaluru');
- INSERT INTO DLOCATION VALUES(5,'Mangaluru');

```
SQL> SELECT *FROM DLOCATION;

       DNO LOCATION
---------- ----------
         1 Mangaluru
         1 Mysore
         2 Mangaluru
         3 Bengaluru
         4 Mangaluru
         5 Mangaluru
```

```
CREATE TABLE PROJECT_PGM
  (PNO NUMBER(2) PRIMARY KEY,
  PNAME VARCHAR2(20),
  PLOCATION VARCHAR2(20),
  DNO NUMBER  REFERENCES
  DEPARTMENT_PGM);
```

- INSERT INTO PROJECT_PGM VALUES(1,'IOT','Managluru',1);
- INSERT INTO PROJECT_PGM VALUES(2,'Data Mining','Managluru',1);
- INSERT INTO PROJECT_PGM VALUES(3,'CC','Hubli',3);
- INSERT INTO PROJECT_PGM VALUES(4,'Image processing','Managluru',4);
- INSERT INTO PROJECT_PGM VALUES(5,'Research','Managluru',5);

```
SQL> SELECT *FROM PROJECT_PGM;
```

PNO	PNAME	PLOCATION	DNO
1	IOT	Managluru	1
2	Data Mining	Managluru	1
3	CC	Hubli	3
4	Image processing	Managluru	4
5	Research	Managluru	5

```
CREATE TABLE  WORKSON
   (SSN  NUMBER(5) REFERENCES
   EMPLOYEE_PGM,
   PNO   NUMBER(2) REFERENCES
   PROJECT_PGM,
   HOURS NUMBER(5,2),
   PRIMARY KEY(SSN, PNO));
```

- INSERT INTO WORKSON VALUES(1,1,4);
- INSERT INTO WORKSON VALUES(2,1,5);
- INSERT INTO WORKSON VALUES(3,2,4);
- INSERT INTO WORKSON VALUES(4,3,4);
- INSERT INTO WORKSON VALUES(5,5,4);

```
SQL> SELECT *FROM WORKSON;
```

SSN	PNO	HOURS
1	1	4
2	1	5
3	2	4
4	3	4
5	5	4

```
SELECT *FROM EMPLOYEE_PGM;
```

SSN	NAME	ADDRESS	G	SALARY	SUPERSSN	DNO
1	Scott	Mangaluru	F	35000	1	4
2	Sahana	Mangaluru	F	35000	1	1
3	Sagar	Bengaluru	M	35000	1	3
4	Sagarik	Mangaluru	M	35000	1	3
5	Sajaan	Mysore	M	600000	1	3

```
SQL> SELECT *FROM DEPARTMENT_PGM;
```

DNO	DNAME	MGRSSN	MGRSTARTD
1	CSE	1	02-NOV-07
2	IOT	2	02-NOV-07
3	Account	2	02-NOV-17
4	ISE	1	02-NOV-00
5	Finance	1	03-NOV-01

```
SQL> SELECT *FROM DLOCATION;
```

DNO	LOCATION
1	Mangaluru
1	Mysore
2	Mangaluru
3	Bengaluru
4	Mangaluru
5	Mangaluru

```
SQL> SELECT *FROM PROJECT_PGM;
```

PNO	PNAME	PLOCATION	DNO
1	IOT	Managluru	1
2	Data Mining	Managluru	1
3	CC	Hubli	3
4	Image processing	Managluru	4
5	Research	Managluru	5

```
SQL> SELECT *FROM WORKSON;
```

SSN	PNO	HOURS
1	1	4
2	1	5
3	2	4
4	3	4
5	5	4

- **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 PNO
FROM PROJECT_PGM
WHERE DNO IN
(SELECT DNO
FROM DEPARTMENT_PGM
WHERE MGRSSN IN
(SELECT SSN
FROM EMPLOYEE_PGM
WHERE NAME ='Scott'))
UNION
(SELECT PNO
FROM WORKSON
WHERE SSN IN
(SELECT SSN
FROM EMPLOYEE_PGM
WHERE NAME='Scott'));
```

PNO
1
2
4
5

- (SELECT PNAME, P.PNO
  FROM PROJECT_PGM P, DEPARTMENT_PGM D, EMPLOYEE_PGM E
  WHERE NAME = 'Scott' AND MGRSSN = SSN AND P.DNO = D.DNO)
  UNION
  (SELECT PNAME, P.PNO
  FROM PROJECT_PGM P, WORKSON W, EMPLOYEE_PGM E
  WHERE E.SSN= W.SSN AND W.PNO = P.PNO AND NAME = 'Scott');

PNAME	PNO
Data Mining	2
IOT	1
Image processing	4
Research	5

- **Show the resulting salaries if every employee working on the ‘IoT’ project is given a 10 percent raise.**

```
SELECT SSN, NAME, SALARY,
SALARY+0.1*SALARY as INC_SAL
FROM EMPLOYEE_PGM
WHERE SSN IN
(SELECT SSN
FROM WORKSON
WHERE PNO IN
(SELECT PNO
FROM PROJECT_PGM
WHERE PNAME ='IOT'));
```

SSN	NAME	SALARY	INC_SAL
1	Scott	35000	38500
2	Sahana	35000	38500

```
SELECT E.SSN, NAME, PNAME,
SALARY+0.1*SALARY AS INC_SAL
FROM EMPLOYEE_PGM E, WORKSON W,
PROJECT_PGM P
WHERE PNAME = 'IOT' AND E.SSN = W.SSN AND
W.PNO = P.PNO;
```

OR

```
SELECT E.SSN, NAME, PNAME, 1.1*SALARY AS
INC_SAL
FROM EMPLOYEE_PGM E, WORKSON W,
PROJECT_PGM P
WHERE PNAME = 'IOT' AND E.SSN = W.SSN AND
W.PNO = P.PNO;
```

SSN	NAME	PNAME	INC_SAL
1	Scott	IOT	38500
2	Sahana	IOT	38500

- **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(SALARY),MAX(SALARY),
MIN(SALARY), AVG(SALARY)
FROM EMPLOYEE_PGM
WHERE DNO IN
(
SELECT DNO
FROM DEPARTMENT_PGM
WHERE DNAME = 'Account');
```

```
SELECT SUM(SALARY),MAX(SALARY),
MIN(SALARY), AVG(SALARY)
FROM EMPLOYEE_PGM E,
DEPARTMENT_PGM D
WHERE E.DNO = D.DNO AND DNAME =
'Account';
```

SUM(SALARY)	MAX(SALARY)	MIN(SALARY)	AVG(SALARY)
670000	600000	35000	223333.333

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

Exists operator

The SQL EXISTS Operator

The EXISTS operator is used to test for the existence of any record in a subquery.

The EXISTS operator returns true if the subquery returns one or more records.

```
SELECT column_name(s)
FROM table_name
WHERE EXISTS
(SELECT column_name FROM table_name WHERE condition);
```

```
SELECT SSN,NAME
FROM EMPLOYEE_PGM E
WHERE NOT EXISTS
(
(SELECT PNO
FROM PROJECT_PGM
WHERE DNO = 5)
MINUS
(SELECT PNO
FROM WORKSON W
WHERE W.SSN = E.SSN));
```

SSN	NAME
5	Sajaan

- **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 COUNT(SSN) AS NO_OF_EMPS
  FROM EMPLOYEE_PGM
  WHERE SALARY >= 600000)
  UNION
  SELECT DNO
  FROM EMPLOYEE_PGM
  GROUP BY DNO
  HAVING COUNT(SSN)>= 5;

NO_OF_EMPS
1