```
CREATE TABLE DEPARTMENT (DNO VARCHAR2 (20) PRIMARY KEY, DNAME
VARCHAR2 (20), MGRSTARTDATE DATE);

CREATE TABLE EMPLOYEE (SSN VARCHAR2 (20) PRIMARY KEY, FNAME VARCHAR2
(20), LNAME VARCHAR2 (20), ADDRESS VARCHAR2 (20), SEX CHAR (1), SALARY
INTEGER, SUPERSSN REFERENCES EMPLOYEE (SSN), DNO REFERENCES
DEPARTMENT (DNO));

CREATE TABLE DLOCATION (DLOC VARCHAR2 (20), DNO REFERENCES DEPARTMENT
(DNO), PRIMARY KEY (DNO, DLOC));

CREATE TABLE PROJECT (PNO INTEGER PRIMARY KEY, PNAME VARCHAR2 (20),
PLOCATION VARCHAR2 (20), DNO REFERENCES DEPARTMENT (DNO));

CREATE TABLE WORKS_ON (HOURS NUMBER (2), SSN REFERENCES EMPLOYEE
(SSN), PNO REFERENCES PROJECT(PNO), PRIMARY KEY (SSN, PNO));
```

Insertion of values to tables

```
INSERT INTO EMPLOYEE (SSN, FNAME, LNAME, ADDRESS, SEX, SALARY) VALUES
('SPCSE100','JOHN','SCOTT','BANGALORE','M', 450000);
INSERT INTO DEPARTMENT VALUES (_1','ACCOUNTS','01-JAN-01','RNSACC02');
INSERT INTO DLOCATION VALUES ('BANGALORE', _1');
INSERT INTO PROJECT VALUES (100,'IOT','BANGALORE','5');
```

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');
```

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';
```

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.FNAME, E.LNAME
FROM EMPLOYEE E
WHERE NOT EXISTS((SELECT PNO
FROM PROJECT
WHERE DNO='5')
MINUS (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;
```