Net Id: nxp154130 Name: Naveenraj Palanisamy CS 6360.004 – Assignment 2

1. Write following queries in SQL.

a. **For each department whose average employee salary is more than $30,000, retrieve the department name and the number of employees working for that department.**

```
SELECT D.DNAME,COUNT(*) AS NUM_OF_EMPLOYEE FROM EMPLOYEE E,DEPARTMENT D
WHERE E.DNO=D.DNO GROUP BY D.DNAME HAVING AVG(E.SALARY)>30000 ORDER BY
D.DNAME;
```

b. **Same as a, except output the number of male employees instead of the number of employees.**

```
 SELECT A.DEPNAME,B.M_COUNT FROM
 (SELECT D.DNAME AS DEPNAME,AVG(E.SALARY) AS SAL FROM EMPLOYEE E,DEPARTMENT D
WHERE E.DNO=D.DNO GROUP BY D.DNAME HAVING AVG(E.SALARY)>30000) A,
 (SELECT DD.DNAME AS DEPNAME,(SELECT COUNT('X') FROM EMPLOYEE EEE WHERE
EEE.DNO=EE.DNO AND EEE.SEX='M') AS M_COUNT FROM EMPLOYEE EE,DEPARTMENT DD
WHERE EE.DNO=DD.DNO) B
 WHERE B.DEPNAME=A.DEPNAME GROUP BY A.DEPNAME,B.M_COUNT ORDER BY A.DEPNAME;
```

c. **Retrieve the names of all employees who work in the department that has the employee with the highest salary among all employees.**

```
SELECT D.DNAME,E.FNAME,E.LNAME FROM EMPLOYEE E,DEPARTMENT D
WHERE E.DNO=D.DNO AND E.dno = ( SELECT dno FROM EMPLOYEE EE WHERE ee.salary=(select
max(eee.salary) from EMPLOYEE eee) ) ORDER BY D.DNAME;
```

d. **Retrieve the names of employees who make at least $10,000 more than the employee who is paid the least in the company**.

```
SELECT D.DNAME,E.FNAME,E.LNAME FROM EMPLOYEE E,DEPARTMENT D
WHERE E.DNO=D.DNO AND E.SALARY >= ( SELECT MIN(EE.SALARY) FROM EMPLOYEE EE
)+10000 ORDER BY D.DNAME;
```

e. **Retrieve the names of employees who is making least in their departments and have more than one dependent. (solve using correlated nested queries)**

```
SELECT D.DNAME,E.FNAME,E.LNAME FROM EMPLOYEE E,DEPARTMENT D WHERE
E.DNO=D.DNO AND E.SALARY = ( SELECT MIN(EE.SALARY) FROM EMPLOYEE EE WHERE
EE.DNO=E.DNO) AND E.SSN IN ( SELECT DE.ESSN FROM DEPENDENT DE WHERE DE.ESSN=E.SSN
GROUP BY DE.ESSN HAVING COUNT('X') >1 ) ORDER BY D.DNAME;
```

2. Specify following views in SQL. Solve questions using correlated nested queries (except a).

a. **A view that has the department name, manager name and manager salary for every department.**

```
CREATE VIEW DEPARTMENT_MANAGER_SALARY AS (SELECT
D.DNAME,E.FNAME,E.LNAME,E.SALARY FROM EMPLOYEE E,DEPARTMENT D
WHERE E.DNO=D.DNO AND E.SSN =D.MGRSSN);
```

b. **A view that has the department name, its manager's name, number of employees working in that department, and the number of projects controlled by that department (for each department).**

```
CREATE VIEW DEPARTMENT_DETAILS AS (SELECT D.DNAME,E.FNAME,E.LNAME,(SELECT
COUNT('X') FROM EMPLOYEE EE WHERE EE.DNO=E.DNO) COUNT_OF_EMPLOYEES,(SELECT
COUNT('X') FROM PROJECT P WHERE P.DNO=E.DNO) PROJECT_NUMBERS  FROM EMPLOYEE
E,DEPARTMENT D WHERE E.DNO=D.DNO AND E.SSN IN ( SELECT DD.MGRSSN FROM
DEPARTMENT DD WHERE DD.DNO=D.DNO));
```

c. **A view that has the project name, controlling department name, number of employees working on the project, and the total hours per week they work on the project (for each project).**

```
CREATE VIEW PROJECT_DETAILS AS (SELECT P.PNAME,D.DNAME,(SELECT COUNT('X') FROM
WORKS_ON W WHERE W.PNO=P.PNO) NUMBER_OF_EMPLOYEES,(SELECT SUM(HOURS) FROM
WORKS_ON W WHERE W.PNO=P.PNO) WORKING_HOURS  FROM PROJECT P, DEPARTMENT D
WHERE P.DNO=D.DNO);
```

d. **A view that has the project name, controlling department name, number of employees, and total hours worked per week on the project for each project with more than one employee working on it.**

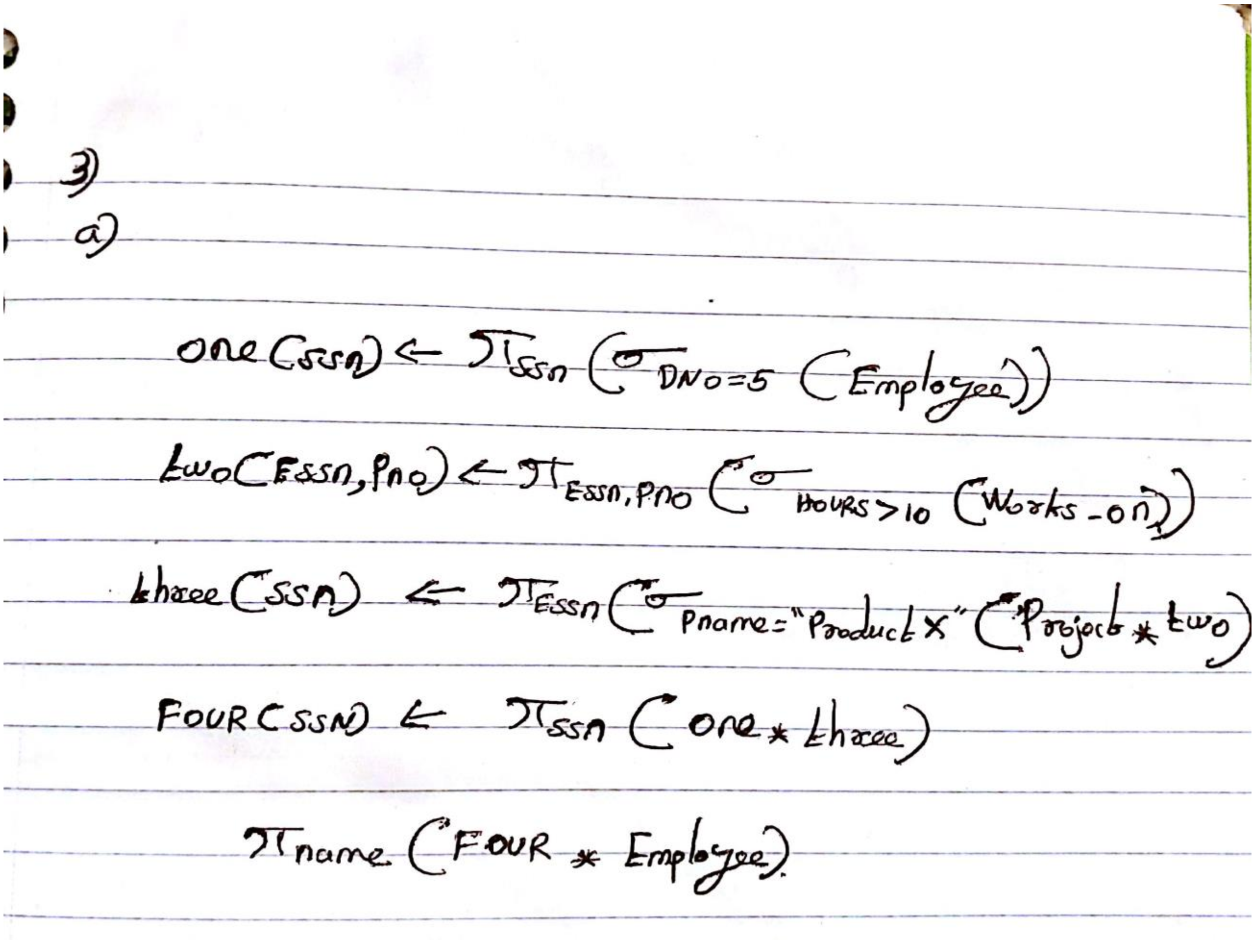
```
CREATE VIEW PROJECT_DETAIL_MORE_THAN_ONE AS (SELECT
PNAME1,DNAME1,EMP_COUNT,TTLHOURS1 FROM (SELECT P.PNAME AS PNAME1,D.DNAME AS
DNAME1,(SELECT COUNT('X') FROM WORKS_ON W WHERE W.PNO=P.PNO) AS
EMP_COUNT,(SELECT SUM(HOURS) FROM WORKS_ON W WHERE W.PNO=P.PNO) AS
TTLHOURS1 FROM PROJECT P, DEPARTMENT D WHERE P.DNO=D.DNO) WHERE
EMP_COUNT>1);
```

e. **A view that has the employee name, employee salary, department that the employee works in, department manager name, manager salary, and average salary for the department.**

```
CREATE VIEW EMPLOYEE_DETAILS AS (SELECT E.FNAME,E.LNAME,E.SALARY,D.DNAME,(SELECT
EE.FNAME FROM EMPLOYEE EE WHERE EE.SSN=E.SUPERSSN) AS MANAGER_NAME,(SELECT
EE.SALARY FROM EMPLOYEE EE WHERE EE.SSN=E.SUPERSSN) AS MANAGER_SALARY,(SELECT
AVG(EE.SALARY) FROM EMPLOYEE EE WHERE EE.DNO=E.DNO) AS DEPT_AVG_SALARY FROM
EMPLOYEE E,DEPARTMENT D WHERE E.DNO=D.DNO);
```

3. Specify following queries in relational algebra.

a. **Retrieve the names of all employees in department 5 who work more than 10 hours per week on the ProductX project.**



3)

a)

$one(ssn) \leftarrow \pi_{ssn}(\sigma_{DNo=5}(Employee))$

$two(Essn, Pno) \leftarrow \pi_{Essn, Pno}(\sigma_{HOURS>10}(Works\text{-}on))$

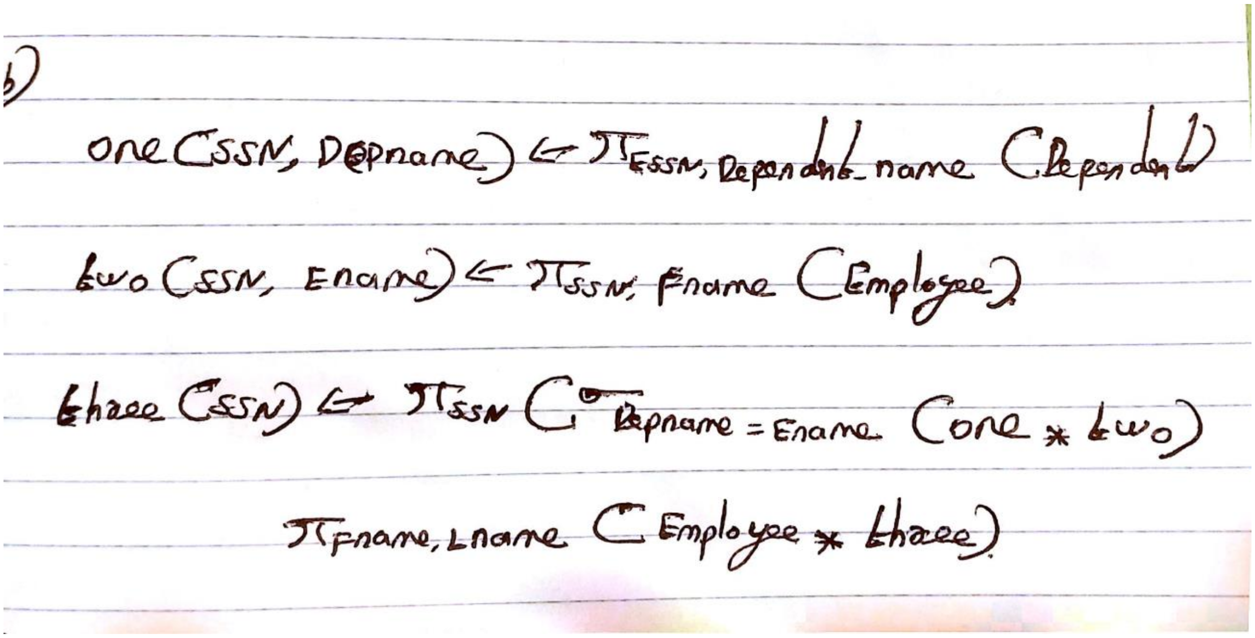
$three(SSN) \leftarrow \pi_{Essn}(\sigma_{Pname=\text{"ProductX"}}(Project * two)$

$FOUR(SSN) \leftarrow \pi_{ssn}(one * three)$

$\pi_{name}(FOUR * Employee)$

b. **List the names of all employees who have a dependent with the same first name as themselves.**

s



b)

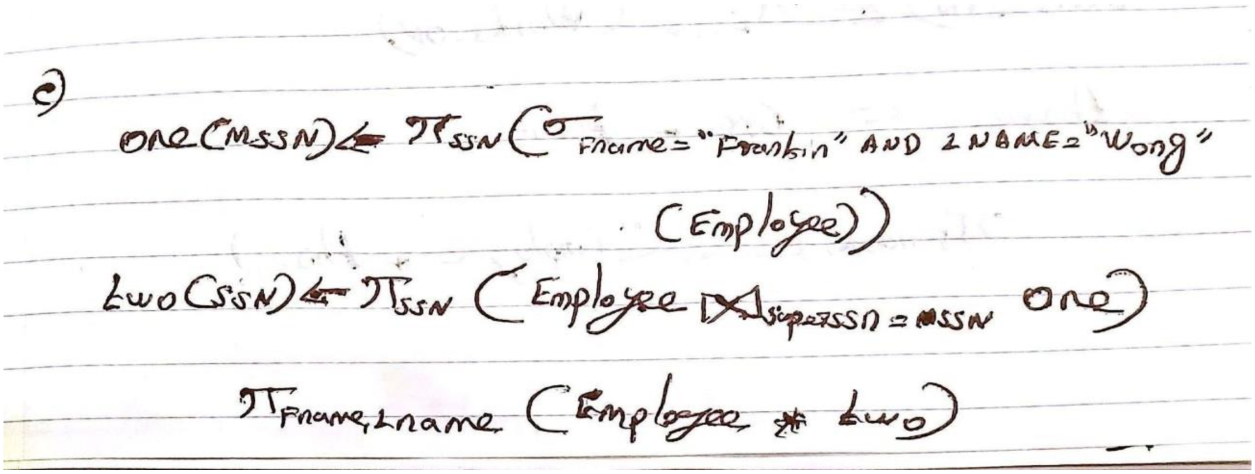
$one(SSN, Depname) \leftarrow \pi_{ESSN, Dependent\_name}(Dependent)$

$two(SSN, Ename) \leftarrow \pi_{SSN, Fname}(Employee)$

$three(SSN) \leftarrow \pi_{SSN}(\sigma_{Depname = Ename}(one * two)$

$\pi_{Fname, Lname}(Employee * three)$

c. **Find the names of all employees who are directly supervised by ‘Franklin Wong’.**



c)

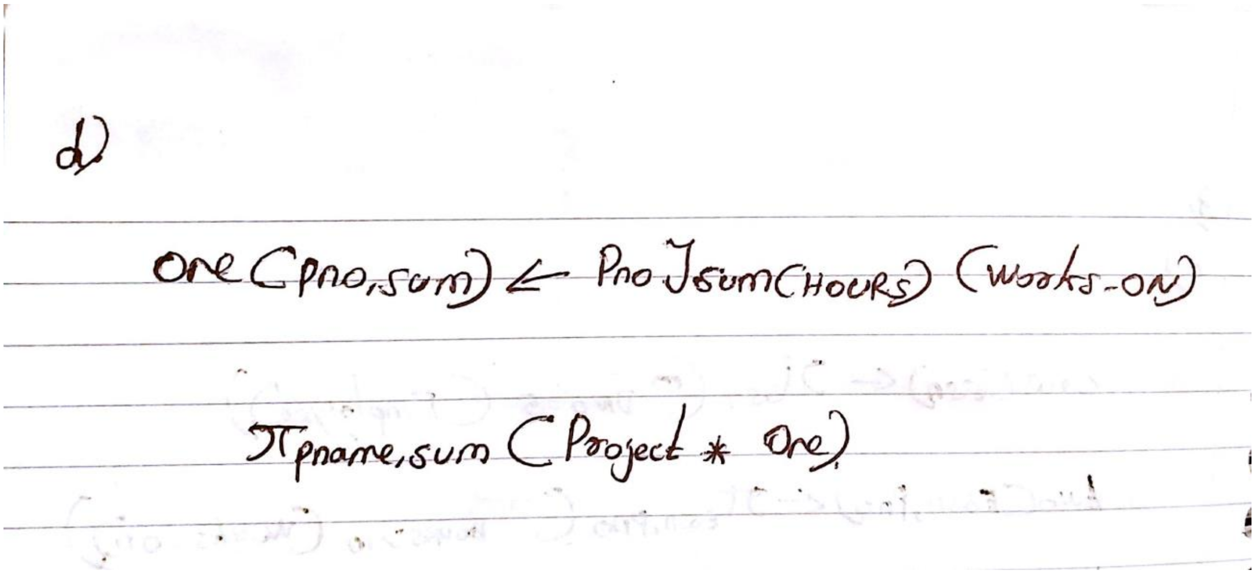
$one(MSSN) \leftarrow \pi_{SSN}(\sigma_{Fname = \text{"Franklin"} \text{ AND } LNAME = \text{"Wong"}}$

$(Employee))$

$two(SSN) \leftarrow \pi_{SSN}(Employee \bowtie_{superSSN = MSSN} one)$

$\pi_{Fname, Lname}(Employee * two)$

d. **For each project, list the project name and the total hours per week (by all employees) spent on that project.**

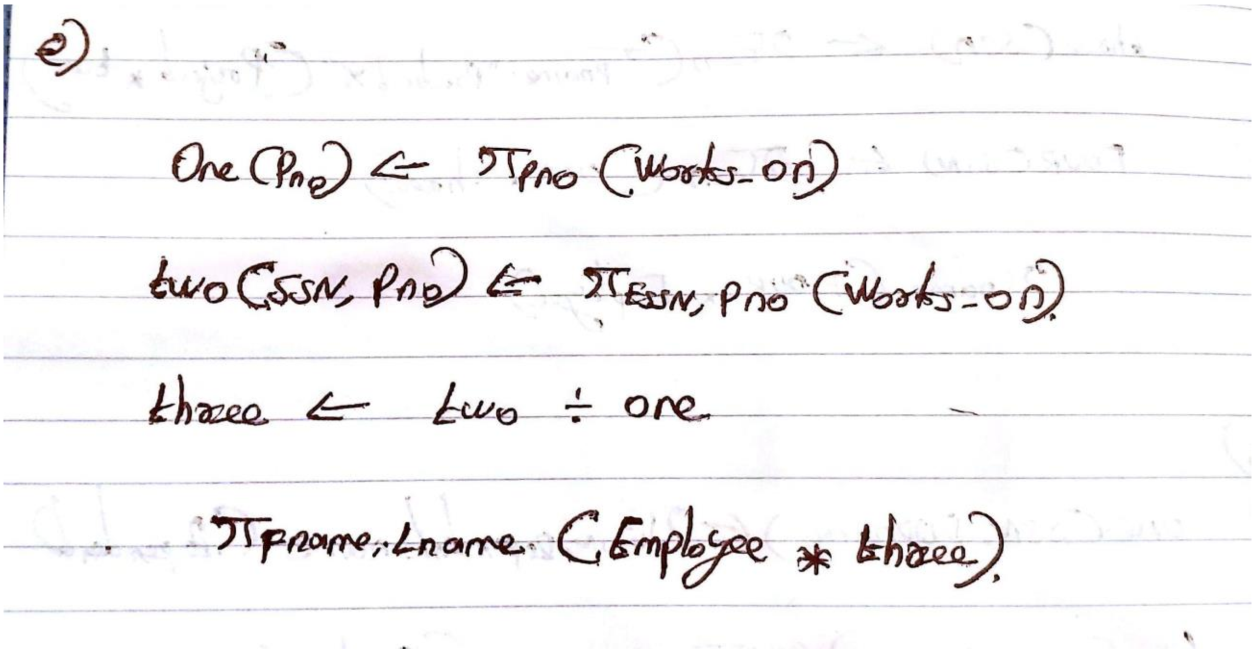


d)

$One(Pno, sum) \leftarrow {}_{Pno}\Im_{sum(HOURS)}(Works\text{-}ON)$

$\pi_{pname, sum}(Project * One)$

e. **Retrieve the names of all employees who work on every project.**



e)

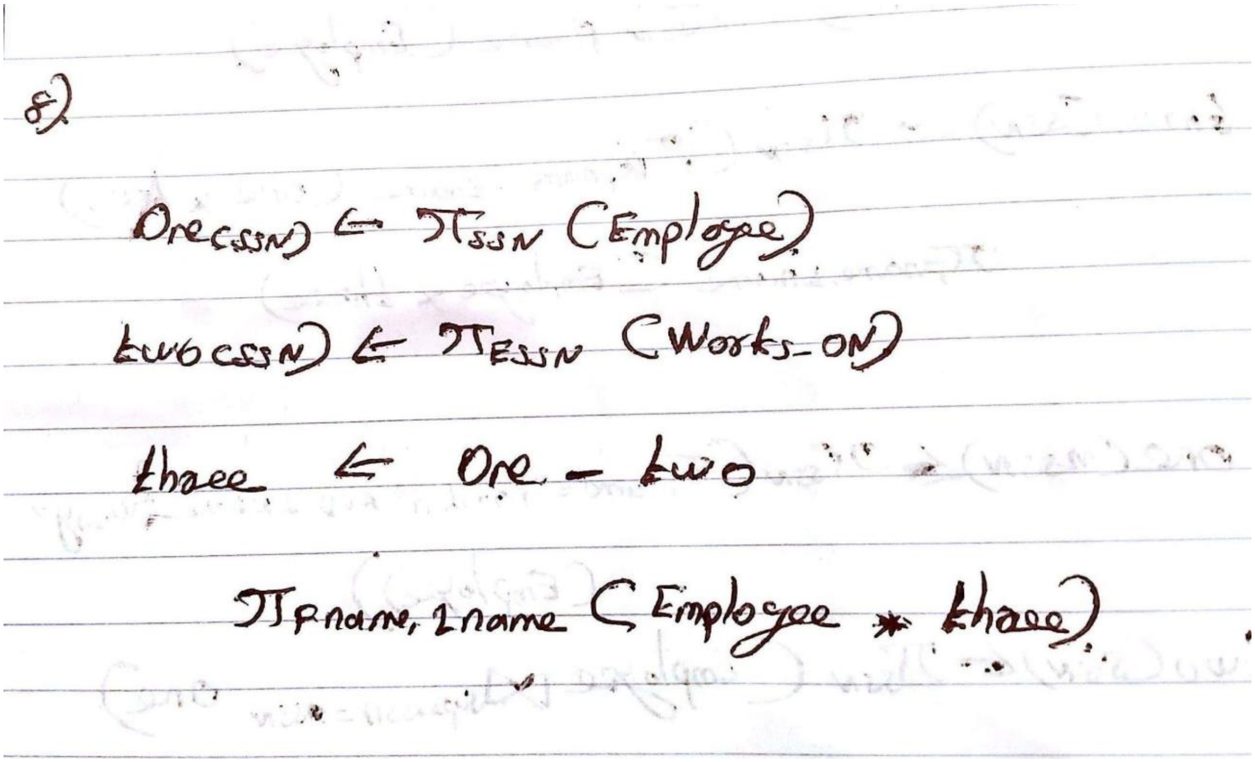
$One(Pno) \leftarrow \pi_{pno}(Works\text{-}on)$

$two(SSN, Pno) \leftarrow \pi_{ESSN, Pno}(Works\text{-}on)$

$three \leftarrow two \div one$

$\pi_{fname, Lname}(Employee * three)$

f. **Retrieve the names of all employees who do not work on any project.**



8)

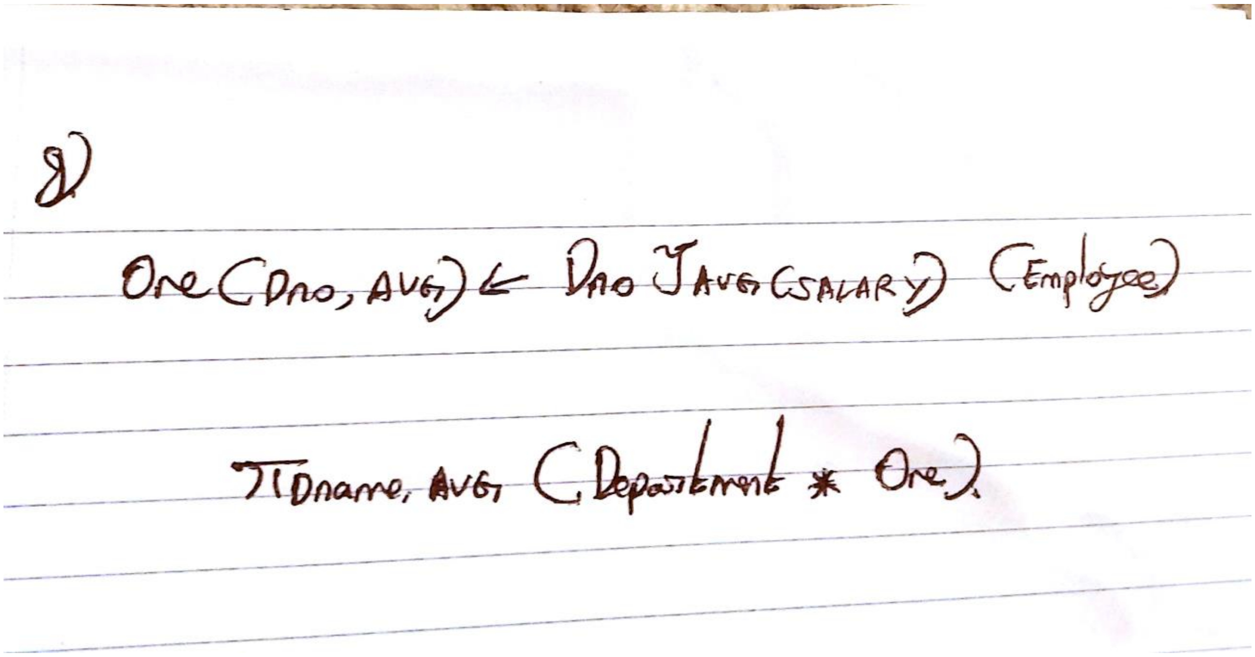
One(SSN) ← π_{SSN} (Employee)

two(SSN) ← π_{ESSN} (Works_ON)

three ← One − two

$\pi_{Fname, Lname}$ (Employee * three)

g. **For each department, retrieve the department name and the average salary of all employees working in that department.**

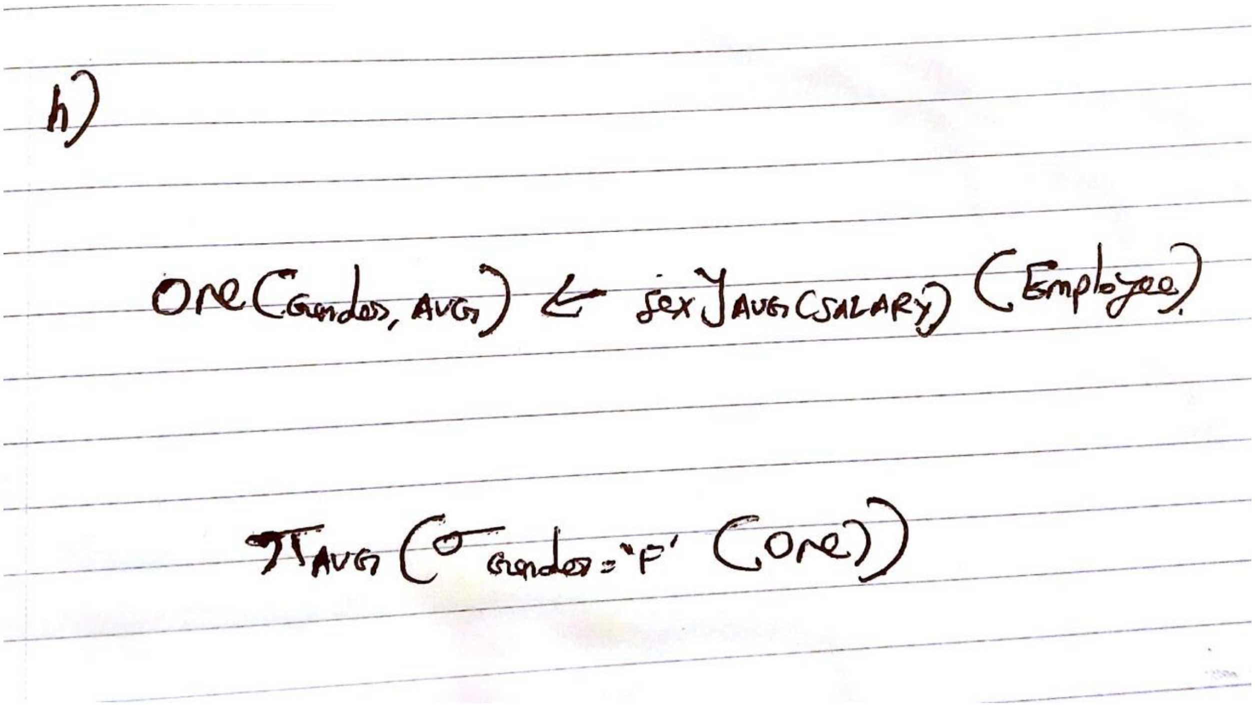


8)

One(Dno, AVG) ← $_{Dno}\mathfrak{I}_{AVG(SALARY)}$ (Employee)

$\pi_{Dname, AVG}$ (Department * One).

h. Retrieve the average salary of all female employees.



h)

$One(Gender, AVG) \leftarrow {}_{sex}\mathfrak{J}_{AVG(SALARY)}(Employee)$

$\pi_{AVG}(\sigma_{Gender='F'}(One))$

i. Find the names and addresses of all employees who work on at least one project located in Houston but whose department has no location in Houston.

i).

$One(pno) \leftarrow \pi_{pnumber} (\sigma_{Plocation = 'Houston'} (Project))$.

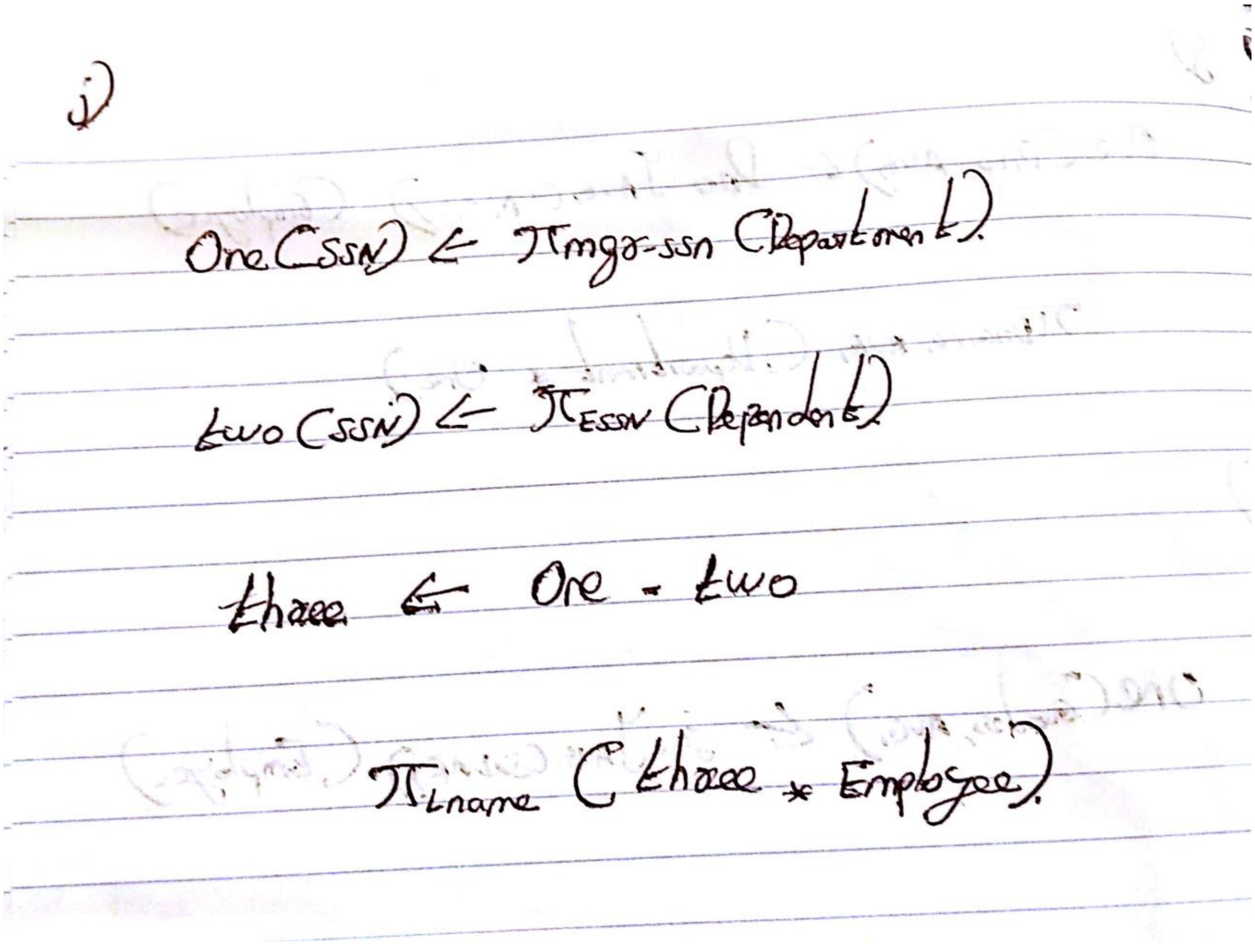
$two(SSN) \leftarrow \pi_{ESSN} (Works\_ON * One)$.

$three(SSN, Dnumber) \leftarrow \pi_{SSN, DNO} (Employee)$

$Four(SSN) \leftarrow \pi_{SSN} (\sigma_{Dlocation <> 'Houston'} (Dept\text{-}Location * three))$

$\pi_{Fname, Lname, Address} (two * Four * Employee)$.

j. List the last names of all department managers who have no dependents.



j)

$One(SSN) \leftarrow \pi_{mgr\text{-}ssn}(Department)$

$two(SSN) \leftarrow \pi_{ESSN}(Dependent)$

$three \leftarrow One - two$

$\pi_{Lname}(three * Employee)$