The University of Texas at Arlington

HW 6.16 Solutions

Sharma Chakravarthy
UT Arlington
sharma@cse.uta.edu
http://www2.uta.edu/sharma

Homework problems

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

EMP_W_X <-- (s PNAME='ProductX' (PROJECT)) J PNUMBER),(PNO) (WORKS_ON)

EMP_WORK_10 <-- (EMPLOYEE) J (SSN),(ESSN) (s HOURS>10 (EMP_W_X))

RESULT <-- P LNAME,FNAME (s DNO=5 (EMP_WORK_10))

LNAME FNAME

Smith John

English Joyce

Homework problems

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

```
SELECT      LNAME, FNAME
FROM        EMPLOYEE, WORKS_ON, PROJECT
WHERE       DNO=5 AND SSN=ESSN AND PNO=PNUMBER
        AND PNAME='ProductX' AND HOURS>10
```

Homework problems

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

```
SELECT    LNAME, FNAME
FROM      EMPLOYEE
WHERE      DNO=5 AND SSN IN
          ( SELECT ESSN
          FROM WORKS_ON
          WHERE HOURS>10 AND PNO IN
                    ( SELECT PNUMBER
                      FROM PROJECT
                      WHERE PNAME='ProductX' ) )
```

HW Solutions

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

E <-- (EMPLOYEE) J (SSN,FNAME),(ESSN,DEPENDENT_NAME) (DEPENDENT)

R <-- P LNAME,FNAME (E)

- Result (empty):

- LNAME FNAME

HW Solutions

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

```
SELECT      LNAME, FNAME
FROM        EMPLOYEE, DEPENDENT
WHERE       SSN=ESSN AND FNAME=DEPENDENT_NAME
```

☞ Another possible SQL query uses nesting as follows:

```
SELECT      LNAME, FNAME
FROM        EMPLOYEE
WHERE       EXISTS ( SELECT *
                     FROM DEPENDENT
                     WHERE FNAME=DEPENDENT_NAME AND
                     SSN=ESSN )
```

HW Solutions

(c) Find the names of employees that are directly supervised by 'Franklin Wong'.

WONG_SSN <-- P SSN (s FNAME='Franklin' AND LNAME='Wong' (EMPLOYEE))

WONG_EMPS <-- (EMPLOYEE) J (SUPERSSN),(SSN) (WONG_SSN)

RESULT <-- P LNAME,FNAME (WONG_EMPS)

☞

Result:

LNAME FNAME

Smith John

Narayan Ramesh

English Joyce

HW Solutions

(c) Find the names of employees that are directly supervised by 'Franklin Wong'.

```
SELECT      E.LNAME, E.FNAME
FROM        EMPLOYEE E, EMPLOYEE S
WHERE       S.FNAME='Franklin' AND S.LNAME='Wong' AND
            E.SUPERSSN=S.SSN
```

☞ Another possible SQL query uses nesting as follows:

```
SELECT      LNAME, FNAME
FROM        EMPLOYEE
WHERE       SUPERSSN IN
                    ( SELECT SSN
                    FROM EMPLOYEE
                    WHERE FNAME='Franklin' AND LNAME='Wong' )
```

HW Solutions

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

PROJ_HOURS(PNO,TOT_HRS) <-- PNO f SUM HOURS (WORKS_ON)

RESULT <-- P PNAME,TOT_HRS ((PROJ_HOURS) J (PNO),(PNUMBER) (PROJECT))

☞

Result:

PNAME TOT_HRS

ProductX 52.5

ProductY 37.5

ProductZ 50.0

Computerization 55.0

Reorganization 25.0

Newbenefits 55.0

HW Solutions

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

```
SELECT      PNAME, SUM (HOURS)
FROM        PROJECT, WORKS_ON
WHERE       PNUMBER=PNO
GROUP BY    PNAME
```

HW Solutions

(e) Retrieve the names of employees who work on every project.

PROJ_EMPS(PNO,SSN) <-- P PNO,ESSN (WORKS_ON)

ALL_PROJS(PNO) <-- P PNUMBER (PROJECT)

EMPS_ALL_PROJS <-- PROJ_EMPS -:- ALLPROJS (* DIVISION operation *)

RESULT <-- P LNAME,FNAME (EMPLOYEE * EMP_ALL_PROJS)

☞ Result (empty):

☞ LNAME FNAME

HW Solutions

(e) Retrieve the names of employees who work on every project.

```
SELECT LNAME, FNAME
FROM    EMPLOYEE
WHERE NOT EXISTS
                    ( SELECT        PNUMBER
                    FROM            PROJECT
                    WHERE           NOT EXISTS
                                    ( SELECT *
                                      FROM  WORKS_ON
                                     WHERE PNUMBER=PNO AND
                                           ESSN=SSN ) )
```

HW Solutions

(f) Retrieve the names of employees who do not work on any project

ALL_EMPS <-- P SSN (EMPLOYEE)

WORKING_EMPS(SSN) <-- P ESSN (WORKS_ON)

NON_WORKING_EMPS <-- ALL_EMPS - WORKING_EMPS (* DIFFERENCE *)

RESULT <-- P LNAME,FNAME (EMPLOYEE * NON_WORKING_EMPS)

Result (empty):
LNAME FNAME

HW Solutions

(f) Retrieve the names of employees who do not work on any project

```
SELECT      LNAME, FNAME
FROM        EMPLOYEE
WHERE       NOT EXISTS
            ( SELECT *
              FROM   WORKS_ON
              WHERE ESSN=SSN )
```

HW Solutions

(g) For each department, retrieve the department name, and the average salary of employees working in that department

DEPT_AVG_SALS(DNUMBER,AVG_SAL) <-- DNO f AVG SALARY (EMPLOYEE)

RESULT <-- P DNAME, AVG_SAL (DEPT_AVG_SALS * DEPARTMENT)

Result:

DNUMBER AVG_SAL

Research 33250

Administration 31000

Headquarters 55000

HW Solutions

(g) For each department, retrieve the department name, and the average salary of employees working in that department

```
SELECT      DNAME, AVG (SALARY)
FROM        DEPARTMENT, EMPLOYEE
WHERE       DNUMBER=DNO
GROUP BY    DNAME
```

HW Solutions

(h) Retrieve the average salary of all female employees.

RESULT(AVG_F_SAL) <-- f AVG SALARY (s SEX='F' (EMPLOYEE))

☞

Result:

AVG_F_SAL

31000

HW Solutions

(h) Retrieve the average salary of all female employees.

```
SELECT      AVG (SALARY)
FROM        EMPLOYEE
WHERE       SEX='F'
```

HW Solutions

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

E_P_HOU(SSN) <-- P ESSN (WORKS_ON J(PNO),(PNUMBER) (s PLOCATION='Houston' (PROJECT)))

D_NO_HOU <-- P DNUMBER (DEPARTMENT) - P DNUMBER (s DLOCATION='Houston' (DEPARTMENT))

E_D_NO_HOU <-- P SSN (EMPLOYEE J(PNO),(DNUMBER) (D_NO_HOU))

RESULT_EMPS <-- E_P_HOU - E_D_NO_HOU (* this is set DIFFERENCE *)

RESULT <-- P LNAME,FNAME,ADDRESS (EMPLOYEE * RESULT_EMPS)

☞

☞ Result:

☞ LNAME FNAME ADDRESS

☞ Wallace Jennifer 291 Berry, Bellaire, TX

HW Solutions

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

```
SELECT  LNAME, FNAME, ADDRESS
FROM    EMPLOYEE
WHERE    EXISTS
        ( SELECT *
          FROM WORKS_ON, PROJECT
          WHERE SSN=ESSN AND PNO=PNUMBER AND
                LOCATION='Houston' )
        AND NOT EXISTS
                ( SELECT *
                FROM DEPT_LOCATIONS
                WHERE DNO=DNUMBER AND DLOCATION='Houston' )
```

I would make this clearer by showing the attributes from relations!

HW Solutions

(j) List the last names of department managers who have no dependents

DEPT_MANAGERS(SSN)<-- P MGRSSN (DEPARTMENT)

EMPS_WITH_DEPENDENTS(SSN) <-- P ESSN (DEPENDENT

RESULT_EMPS <-- DEPT_MANAGERS -EMPS_WITH_DEPENDENTS

RESULT <-- P LNAME,FNAME (EMPLOYEE * RESULT_EMPS)

☞

Result:

LNAME FNAME

Borg James

HW Solutions

(j) List the last names of department managers who have no dependents

```
SELECT      LNAME, FNAME
FROM        EMPLOYEE
WHERE        EXISTS
               ( SELECT *
                  FROM        DEPARTMENT
                 WHERE        SSN=MGRSSN )
             AND NOT EXISTS
               ( SELECT *
                  FROM        DEPENDENT
                 WHERE       SSN=ESSN )
```