

Khoi Duong

Prof. VB

CS457

11/11/2022

HW 4B

Exercise 8.15

Show the result of each of the sample queries in Section 8.5 as it would apply to the database state in Figure 5.6.

Query 1:

Retrieve the name and address of all employees who work for the 'Research' department.

```
MariaDB [19610dm]> select Fname, Lname, Address
-> from (EMPLOYEE join DEPARTMENT on Dno = Dnumber)
-> where Dname = 'Research';
```

Fname	Lname	Address
John	Smith	731 Fondren, Houston, TX
Franklin	Wong	638 Voss, Houston, TX
Joyce	English	5631 Rice, Houston, TX
Ramesh	Narayan	975 Fire Oak, Humble, TX

4 rows in set (0.000 sec)

Query 2

For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birth date.

```
MariaDB [19610dm]> select Pnumber, Dnum, Lname, Address, Bdate
-> from PROJECT, DEPARTMENT, EMPLOYEE
-> where Dnum = Dnumber and Mgr_ssn = Ssn and Plocation = 'Stafford';
Empty set (0.000 sec)
```

Query 3

Find the names of employees who work on all the projects controlled by department number 5.

```
MariaDB [19610dm]> select E.Fname, E.Lname
-> from EMPLOYEE as E join WORKS_ON as W on W.Essn = E.Ssn
-> join PROJECT as P on W.Pno = P.Pnumber
-> where P.Dnum = 5;
```

Fname	Lname
John	Smith
Joyce	English
John	Smith
Franklin	Wong
Joyce	English
Franklin	Wong
Ramesh	Narayan
Franklin	Wong

8 rows in set (0.001 sec)

Query 4

Make a list of project numbers for projects that involve an employee whose last name is 'Smith', either as a worker or as a manager of the department that controls the project.

```

MariaDB [19610dm]> (select distinct Pnumber from PROJECT, DEPARTMENT, EMPLOYEE
-> where Dnum = Dnumber and Mgr_ssn = Ssn and Lname = 'Smith')
-> union
-> (select Pnumber from PROJECT, WORKS_ON, EMPLOYEE
-> where Pnumber = Pno and Essn = Ssn and Lname = 'Smith');
+-----+
| Pnumber |
+-----+
|        1 |
|        2 |
+-----+
2 rows in set (0.001 sec)

```

Query 5

List the names of all employees with two or more dependents.

```

MariaDB [19610dm]> select Lname, Fname from EMPLOYEE where (select count(*) from
DEPENDENT where Ssn = Essn) >= 2;
+-----+-----+
| Lname | Fname |
+-----+-----+
| Smith | John  |
| Wong  | Franklin |
+-----+-----+
2 rows in set (0.001 sec)

```

Query 6

Retrieve the names of employees who have no dependents.

```

MariaDB [19610dm]> select Lname, Fname from EMPLOYEE where (select count(*) from
DEPENDENT where Ssn = Essn) = 0;
+-----+-----+
| Lname | Fname |
+-----+-----+
| Smith | Alicia |
| English | Joyce |
| Narayan | Ramesh |
| Zelaya | John |
| Borg | James |
+-----+-----+
5 rows in set (0.001 sec)

```

Query 7

List the names of managers who have at least one dependent.

```
MariaDB [19610dm]> select Fname, Lname from EMPLOYEE
-> where exists (select * from DEPARTMENT where Ssn = Mgr_ssn)
-> and exists (select * from DEPENDENT where Ssn = Essn);
+-----+-----+
| Fname  | Lname |
+-----+-----+
| Franklin | Wong  |
+-----+-----+
1 row in set (0.001 sec)
```

Exercise 8.16

a.

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

```
MariaDB [19610dm]> select e.Fname, e.Minit, e.Lname
-> from EMPLOYEE e, WORKS_ON w, PROJECT p
-> where e.Dno = 5 and e.Ssn = w.Essn and w.Pno = p.Pnumber
-> and p.Pname = 'ProductX' and w.Hours > 10;
+-----+-----+-----+
| Fname | Minit | Lname  |
+-----+-----+-----+
| John  | B     | Smith  |
| Joyce | A     | English|
+-----+-----+-----+
2 rows in set (0.001 sec)
```

b.

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

```
MariaDB [19610dm]> select e.Fname, e.Minit, e.Lname from EMPLOYEE e, DEPENDENT d
where e.Ssn = d.Essn and e.Fname = d.Dependent_name;
Empty set (0.000 sec)
```

c.

Find the names of all employees who are directly supervised by 'Franklin Wong'.

```
MariaDB [19610dm]> select e.Fname, e.Minit, e.Lname from EMPLOYEE e
-> where Super_ssn in (select Ssn from EMPLOYEE where
-> Fname = 'Franklin' and Lname = 'Wong');
```

```
+-----+-----+-----+
| Fname  | Minit | Lname  |
+-----+-----+-----+
| John   | B     | Smith  |
| Joyce  | A     | English|
| Ramesh | K     | Narayan|
+-----+-----+-----+
3 rows in set (0.000 sec)
```

d.

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

```
MariaDB [19610dm]> select Pname, sum(Hours) as Total_hours from PROJECT P, WORKS
_ON W where P.Pnumber = W.Pno group by W.Pno;
```

```
+-----+-----+
| Pname          | Total_hours |
+-----+-----+
| ProductX       | 52.5        |
| ProductY       | 37.5        |
| ProductZ       | 50.0        |
| Computerization| 55.0        |
| Reorganization | 35.0        |
| Newbenefits    | 55.0        |
+-----+-----+
6 rows in set (0.000 sec)
```

e.

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

```
MariaDB [19610dm]> select e.Fname, e.Minit, e.Lname  
(  
  -> from EMPLOYEE e where not exists(  
  -> select p.Pnumber from PROJECT p where not exists(  
  -> select w.Essn from WORKS_ON w  
  -> where w.Pno = p.Pnumber and w.Essn = e.Ssn));  
Empty set (0.001 sec)
```

f.

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

```
MariaDB [19610dm]> select Fname, Minit, Lname from EMPLOYEE  
  -> where not exists (select Essn from WORKS_ON where Ssn = Essn);  
+-----+-----+-----+  
| Fname  | Minit | Lname  |  
+-----+-----+-----+  
| Alicia | T     | Smith  |  
| John   | G     | Zelaya |  
+-----+-----+-----+  
2 rows in set (0.001 sec)
```

g.

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

```
MariaDB [19610dm]> select Dname, avg(Salary) as Average_salary
-> from DEPARTMENT, EMPLOYEE
-> where Dno = Dnumber group by Dname;
```

Dname	Average_salary
Administration	44000.000000
Headquarters	64000.000000
Research	42575.000000

```
3 rows in set (0.001 sec)
```

h.

Retrieve the average salary of all female employees.

```
MariaDB [19610dm]> select avg(Salary) as Average_salary
-> from EMPLOYEE where Sex = 'F';
```

Average_salary
38500.000000

```
1 row in set (0.000 sec)
```

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.

```
MariaDB [19610dm]> select e.Fname, e.Minit, e.Lname, e.Address
-> from EMPLOYEE e, PROJECT p, WORKS_ON w
-> where e.Ssn = w.Essn and p.Pnumber = w.Pno and p.Plocation = 'Houston'
-> and not exists(
-> select d.Dnumber from DEPT_LOCATIONS d where e.Dno = d.Dnumber and d.Dlocation = 'Houston');
```

```
Empty set (0.001 sec)
```

j.

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

```
MariaDB [19610dm]> select e.Lname from EMPLOYEE e, DEPARTMENT d
-> where e.Ssn = d.Mgr_ssn and not exists(
-> select Essn from DEPENDENT where d.Mgr_ssn = Essn);
+-----+
| Lname |
+-----+
| Borg  |
+-----+
1 row in set (0.001 sec)
```

Exercise 8.18

a.

How many copies of the book titled The Lost Tribe are owned by the library branch whose name is 'Sharpstown'?

Relational expression:

$$\pi_{NoOfCopies}(\sigma_{Title='The\ Lost\ Tribe'\ AND\ BranchName='Sharpstown'}((BOOK \bowtie BOOK_COPIES) \bowtie LIBRARY_BRANCH))$$

SQL query:

```
SELECT NoOfCopies
FROM ( (BOOK JOIN BOOK_COPIES ) JOIN LIBRARY_BRANCH )
WHERE Title='The Lost Tribe' AND BranchName='Sharpstown';
```

b.

How many copies of the book titled The Lost Tribe are owned by each library branch?

Relational expression:

$$\pi_{BranchName, NoOfCopies}(\sigma_{Title='The Lost Tribe'}((BOOK \bowtie BOOK_COPIES) \bowtie LIBRARY_BRANCH))$$

SQL query:

```
SELECT BranchName, NoOfCopies
FROM ( (BOOK JOIN BOOK_COPIES ) JOIN LIBRARY_BRANCH )
WHERE Title='The Lost Tribe';
```

c.

Retrieve the names of all borrowers who do not have any books checked out.

Relational expression:

$$\pi_{Name}(BORROWER - \sigma_{BORROWER.CardNo = BOOKLOAN.CardNo}(BOOKLOAN))$$

SQL Query:

```
SELECT Name
FROM BORROWER B
WHERE NOT EXIST ( SELECT * FROM BOOK_LOANS L
WHERE B.CardNo = L.CardNo );
```

d.

For each book that is loaned out from the Sharpstown branch and whose Due_date is today, retrieve the book title, the borrower's name, and the borrower's address.

Relational expression:

$$DATA \rightarrow (((BOOK_LOANS \bowtie_{Book_id} BOOK) \bowtie_{CardNo} BORROWER) \bowtie_{Branch_id} LIBRARY_BRANCH)$$

$$\pi_{Title, Name, Address}(\sigma_{Branch_name = 'Sharpstown' AND Due_date = 'today'}(DATA))$$

SQL Query:

```
SELECT B.Title, R.Name, R.Address  
  
FROM BOOK B, BORROWER R, BOOK_LOANS BL, LIBRARY_BRANCH LB  
  
WHERE LB.BranchName='Sharpstown' AND LB.BranchId=BL.BranchId AND  
BL.DueDate='today' AND BL.CardNo=R.CardNo AND BL.BookId=B.BookId;
```

e.

For each library branch, retrieve the branch name and the total number of books loaned out from that branch.

Relational expression:

$DATA \rightarrow (BOOK_LOANS \bowtie_{Branch_id} LIBRARY_BRANCH)$

$\pi_{Branch_name, Count(*)} ((DATA) \text{ group by } Branch_name)$

SQL Query:

```
SELECT L.BranchName, COUNT(*)  
  
FROM BOOK_LOANS B, LIBRARY_BRANCH L  
  
WHERE B.BranchId = L.BranchId  
  
GROUP BY L.BranchName;
```

f.

Retrieve the names, addresses, and the number of books checked out for all borrowers who have more than five books checked out.

Relational expression:

$DATA \rightarrow (BOOK_LOANS \bowtie_{Card_no} BORROWER)$

$\pi_{Name, Address, Count(*)}(\sigma_{Count(*) > 5}(DATA \text{ group by Card_no}))$

SQL Query:

```
SELECT B.Name, B.Address, COUNT(*)
FROM BORROWER B, BOOK_LOANS L
WHERE B.CardNo = L.CardNo
GROUP BY B.CardNo
HAVING COUNT(*) > 5;
```

g.

For each book authored (or coauthored) by Stephen King, retrieve the title and the number of copies owned by the library branch whose name is Central.

Relational expression:

$DATA \rightarrow (((BOOK_AUTHORS \bowtie BOOK) \bowtie BOOK_COPIES) \bowtie LIBRARY_BRANCH)$

$\pi_{Title, No_of_copies}(\sigma_{Author_name = 'Stephen King' \text{ AND } Branch_name = 'Central'}(DATA))$

SQL query:

```
SELECT Title, NoOfCopies
FROM ( ( (BOOK_AUTHORS JOIN BOOK)
JOIN BOOK_COPIES)
JOIN LIBRARY_BRANCH)
WHERE Author_name = 'Stephen King' and Branch_name = 'Central';
```