Assignment 5

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

DEPT_LOCATIONS

Dnumber	Dlocation

PROJECT

Pname	Pnumber	Plocation	Dnum

WORKS_ON

Essn	Pno	Hours

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship

Figure 5.5
Schema diagram for the COMPANY relational database schema.

7.7. In SQL, specify the following queries on the database in Figure 5.5 using the concept of nested queries and other concepts described in this chapter.

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

b. Retrieve the names of all employees whose supervisor's supervisor has '888665555' for Ssn.

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

```
a. SELECT   Fname, Minit, Lname
   FROM     EMPLOYEE
   WHERE    Dno = (SELECT   Dno
                   FROM     EMPLOYEE
                   WHERE    Salary = (SELECT_MAX(Salary)
                                      FROM     EMPLOYEE))

b. SELECT   Fname, Minit, Lname
   FROM     EMPLOYEE
   WHERE    Super_ssn = (SELECT   Ssn
                         FROM     EMPLOYEE
                         WHERE    Super_ssn = '888665555')
c. SELECT   Fname, Minit, Lname
   FROM     EMPLOYEE
   WHERE    Salary - 10000 >= (SELECT_MIN(Salary)
                               FROM     EMPLOYEE)
```

7.8. Specify the following views in SQL on the COMPANY database schema shown in Figure 5.5.

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

b. A view that has the employee name, supervisor name, and employee salary for each employee who works in the 'Research' department

c. 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

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*

```
a.  CREATE VIEW   DEPT_MGR(Dept_name, Mgr_fname, Mgr_minit, Mgr_lname
                                                    Mgr_sal)
    AS SELECT     Dname, Fname, Minit, Lname, Salary
    FROM          DEPARTMENT, EMPLOYEE
    WHERE         Mgr_ssn = Ssn;

b.  CREATE VIEW   RESEARCH_DEPT(E.Fname, E.Minit, E.Lname S.Fname,
                                                    S.Minit, S.Lname, E.salary)
    AS SELECT     E.Fname, E.Minit, Lname, S.Fname, S.Minit, S.Lname, E.Salary
    FROM          EMPLOYEE E, EMPLOYEE S, Department D
    WHERE         E.Super_ssn = S.ssn AND E.Dno = D.Dnumber AND
                  D.Dname = 'Research';
c.  CREATE VIEW   PROJECT_INFO(Project_name, Dept_name,  Num_employees,
                                                    Total_hrs)
    AS SELECT     Pname, Dname, COUNT(*), SUM(Hours)
    FROM          PROJECT, DEPARTMENT, WORKS_ON
    WHERE         Pnumber = Pno AND Dnumber = Dnum
    GROUP BY      Pnumber, Pname;
d.  CREATE VIEW   PROJECT_INFO2(Project_name, Dept_name,  Num_employees,
                                                    Total_hrs)
    AS SELECT     Pname, Dname, COUNT(*), SUM(Hours)
    FROM          PROJECT, DEPARTMENT, WORKS_ON
    WHERE         Pnumber = Pno AND Dnumber = Dnum
    GROUP BY      Pnumber, Pname
    HAVING        COUNT(*) > 1;
```

AIRPORT

Airport_code	Name	City	State

FLIGHT

Flight_number	Airline	Weekdays

FLIGHT_LEG

Flight_number	Leg_number	Departure_airport_code	Scheduled_departure_time
		Arrival_airport_code	Scheduled_arrival_time

LEG_INSTANCE

Flight_number	Leg_number	Date	Number_of_available_seats	Airplane_id
		Departure_airport_code	Departure_time	Arrival_airport_code	Arrival_time

FARE

Flight_number	Fare_code	Amount	Restrictions

AIRPLANE_TYPE

Airplane_type_name	Max_seats	Company

CAN_LAND

Airplane_type_name	Airport_code

AIRPLANE

Airplane_id	Total_number_of_seats	Airplane_type

SEAT_RESERVATION

Flight_number	Leg_number	Date	Seat_number	Customer_name	Customer_phone

Figure 5.8
The AIRLINE relational database schema.

8.17. Consider the AIRLINE relational database schema shown in Figure 5.8, which was described in Exercise 5.12. Specify the following queries in relational algebra:

a. For each flight, list the flight number, the departure airport for the first leg of the flight, and the arrival airport for the last leg of the flight.

b. List the flight numbers and weekdays of all flights or flight legs that depart from Houston Intercontinental Airport (airport code 'iah') and arrive in Los Angeles International Airport (airport code 'lax').

c. List the flight number, departure airport code, scheduled departure time, arrival airport code, scheduled arrival time, and weekdays of all flights or flight legs that depart from some airport in the city of Houston and arrive at some airport in the city of Los Angeles.

d. List all fare information for flight number 'co197'.

e. Retrieve the number of available seats for flight number 'co197' on '2009-10-09'.

$\sigma_{<\text{selection condition}>}(R)$

$\pi_{<\text{attribute list}>}(R)$

$\rho_{S(B1, B2, \ldots, Bn)}(R)$ or $\rho_{S}(R)$ or $\rho_{(B1, B2, \ldots, Bn)}(R)$

$R_1 \times R_2$

$R_1 \bowtie_{<\text{join condition}>} R_2$

$R_1 *_{<\text{join condition}>} R_2$

a. **INT1←$\sigma_{<Leg\_number = 1>}$(FLIGHT_LEG)**
INT2←$\pi_{<Flight\_number, Departure\_airport\_code>}$(INT1)
INT3←$\sigma_{<Leg\_number = MAX(Leg\_number)>}$(FLIGHT_LEG)
INT4←$\pi_{<Flight\_number, Arrival\_airport\_code>}$(INT3)
RESULT←(INT2 * INT4)

b. **INT1←$\sigma_{<Departure\_airport\_code = 'iah' AND Arrival\_airport\_code = 'lax'>}$(FLIGHT_LEG)**
INT2←$\pi_{<Flight\_number>}$(INT1)
INT3←(INT2 * FLIGHT)
RESULT←$\pi_{<Flight\_number,Weekdays>}$(INT3)

c. **INT1←$\sigma_{<Departure\_airport\_code = 'iah' AND Arrival\_airport\_code = 'lax'>}$(FLIGHT_LEG)**
INT2←$\pi_{<Flight\_number, Departure\_airport\_code, Departure\_time, Arrival\_airport\_code, Arrival\_time>}$(INT1)
INT3←(INT2 * FLIGHT)
RESULT←$\pi_{<Flight\_number, Departure\_airport\_code, Departure\_time, Arrival\_airport\_code, Arrival\_time, Weekdays>}$(INT3)

d. **RESULT←$\sigma_{<Flight\_number = 'co197'>}$(FARE)**

e. **INT1←$\sigma_{<Flight\_number = 'co197' AND Date = '2009-10-09'>}$(LEG_INSTANCE)**
RESULT←$\pi_{<Number\_of\_available\_seats>}$(INT1)

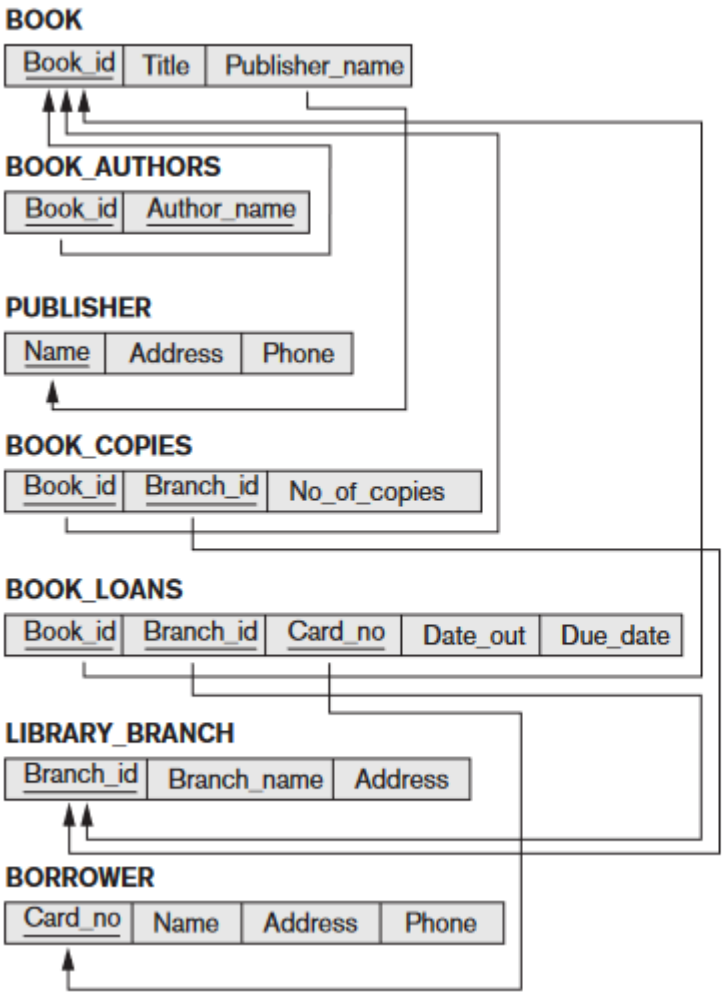

Figure 8.14
A relational database schema for a LIBRARY database.

8.18. Consider the LIBRARY relational database schema shown in Figure 8.14, which is used to keep track of books, borrowers, and book loans. Referential integrity constraints are shown as directed arcs in Figure 8.14, as in the notation of Figure 5.7. Write down relational expressions for the following queries:

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

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

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

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.

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

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

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.

a. $INT1 \leftarrow \sigma_{<Branch\_name = 'Sharpstown'>}(LIBRARY\_BRANCH)$
$INT2 \leftarrow (INT1 * BOOK\_COPIES)$
$INT3 \leftarrow \sigma_{<Title = 'The\ Lost\ Tribe'>}(BOOK)$
$INT4 \leftarrow (INT3 * INT2)$
$RESULT \leftarrow \pi_{<No\_of\_copies>}(INT4)$

b. $INT1 \leftarrow \sigma_{<Title = 'The\ Lost\ Tribe'>}(BOOK)$
$INT2 \leftarrow \pi_{<Book\_id>}(INT1)$
$INT3 \leftarrow (INT2 * BOOK\_COPIES)$
$INT4 \leftarrow \pi_{<Branch\_id,\ No\_of\_copies>}(INT3)$
$INT5 \leftarrow (INT4 * LIBRARY\_BRANCH)$
$RESULT \leftarrow \pi_{<Branch\_name,\ No\_of\_copies>}(INT5)$

c. $INT1 \leftarrow (BORROWER * BOOK\_LOANS)$
$INT2 \leftarrow \pi_{<Card\_no,\ Name,\ Address,\ Phone>}(INT1)$
$INT3 \leftarrow (BORROWER - INT2)$
$RESULT \leftarrow \pi_{<Name>}(INT3)$

d. $INT1 \leftarrow \sigma_{<Branch\_name = 'Sharpstown'>}(LIBRARY\_BRANCH)$
$INT2 \leftarrow \pi_{<Branch\_id>}(INT1)$
$INT3 \leftarrow (INT2 * BOOK\_LOANS)$
$INT4 \leftarrow \sigma_{<Due\_date = Current\ Date>}(INT3)$
$INT5 \leftarrow (INT4 * BOOK)$
$INT6 \leftarrow (INT5 * BORROWER)$
$RESULT \leftarrow \pi_{<Title,\ Name,\ Address>}(INT6)$

e. **INT1←$\pi_{<Branch\_id,\ Branch\_name>}$(LIBRARY_BRANCH)**
INT2←$_{<Branch\_id>}\Im_{<COUNT\ Book\_id>}$(BOOK_LOANS)
INT3←(INT1 * INT2)
RESULT←$\pi_{<Branch\_name,\ COUNT\ Book\_id>}$(INT3)

f. **INT1←$_{<Card\_no>}\Im_{<COUNT\ Book\_id>}$(BOOK_LOANS)**
INT2←(INT1 *$_{COUNT\ Book\_id\ >\ 5}$ BORROWER)
RESULT←$\pi_{<Name,\ Address,\ COUNT\ Book\_id>}$(INT2)

g. **INT1←$\sigma_{<Author\_name\ =\ 'Steven\ King'>}$(BOOK_AUTHORS)**
INT2←(INT1 * BOOK)
INT3←(INT2 * BOOK_COPIES)
INT4←(INT3 * LIBRARY_BRANCH)
INT5←$\sigma_{<Branch\_name\ =\ 'Central'>}$(INT4)
RESULT←$\pi_{<Title,\ No\_of\_copies>}$(INT5)

8.32. A nested query is a query within a query. More specifically, a nested query is a parenthesized query whose result can be used as a value in a number of places, such as instead of a relation. Specify the following queries on the database specified in Figure 5.5 using the concept of nested queries and the relational operators discussed in this chapter. Also show the result of each query as it would apply to the database state in Figure 5.6.

a. List the names of all employees who work in the department that has the employee with the highest salary among all employees.

b. List the names of all employees whose supervisor's supervisor has '888665555' for Ssn.

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

a. RESULT←$\pi_{Fname,\ Lname}$ (σ_{Dno} = π_{Dno}(σ_{Salary} = MAX(Salary)(EMPLOYEE)) (EMPLOYEE))

b. RESULT←$\pi_{Fname,\ Lname}$ (σ_{Super_ssn} = π_{Ssn}(σ_{Super_ssn} = '888665555'(EMPLOYEE)) (EMPLOYEE))

c. RESULT←$\pi_{Fname,\ Lname}$ ($\sigma_{Salary \geq 10000 + (}$$\pi_{Salary}$($\sigma_{MIN(Salary)}$ (EMPLOYEE)) (EMPLOYEE))

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

DEPT_LOCATIONS

Dnumber	Dlocation

PROJECT

Pname	Pnumber	Plocation	Dnum

WORKS_ON

Essn	Pno	Hours

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship

Figure 5.5
Schema diagram for the COMPANY relational database schema.