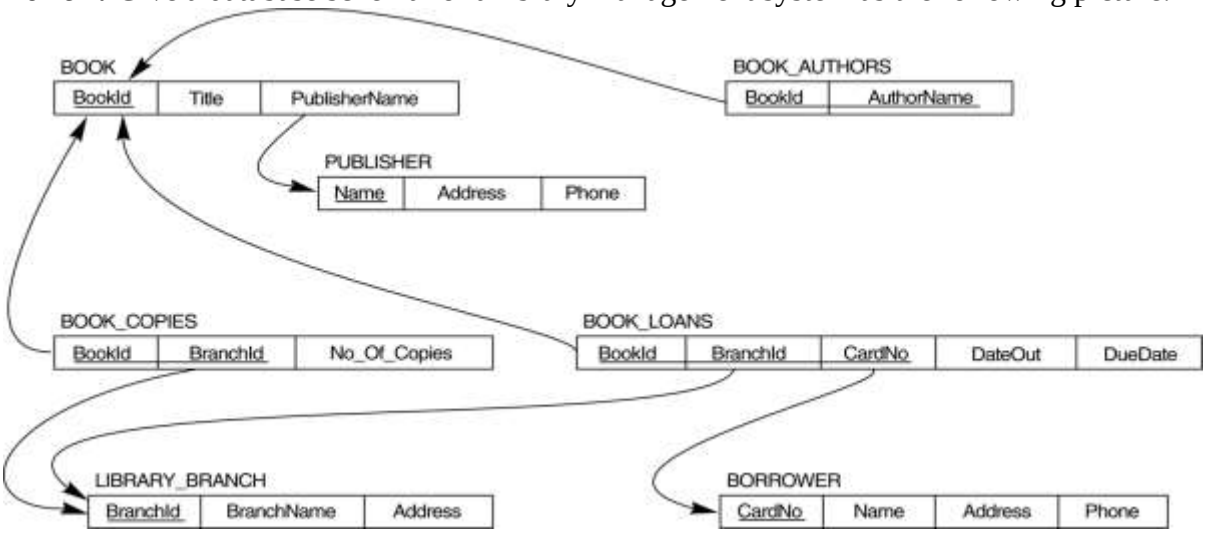
SQL EXERCISE

Retrieve the required information using SQL language.

Part I. Give a database schema for a library management system as the following picture.



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

```
SELECT    No_Of_Copies
FROM      BOOK as B, BOOK_COPIES as BC, LIBRARY_BRANCH as LB
WHERE     B.BookId = BC.BookId and BC. BranchId = LB.BranchId and
          Title=”The Lost Tribe” and BranchName=”Sharpstown”
```

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

```
SELECT    BranchName, No_Of_Copies
FROM      BOOK, BOOK_COPIES, LIBRARY_BRANCH
WHERE     Title=”The Lost Tribe”
```

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

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

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

```
SELECT    B.Title, R.Name, R.Address
FROM      BOOKas B, LIBRARY_BRANCH as LB, BOOK_LOANS as BL, BORROWER as R
WHERE     LB.BranchId=BL.BranchId and BL.CardNo=R.CardNo  and BL.BookId=B.BookId  and
          LB.BranchName=”Sharpstown” and BL.DueDate=date()
```

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

```
SELECT    LB.BranchName, COUNT(BL.BookId, CardNo )
FROM      BOOK_LOANS as BL, LIBRARY_BRANCH as LB
WHERE     LB.BranchId=BL.BranchId
GROUP BY  LB.BranchName
```

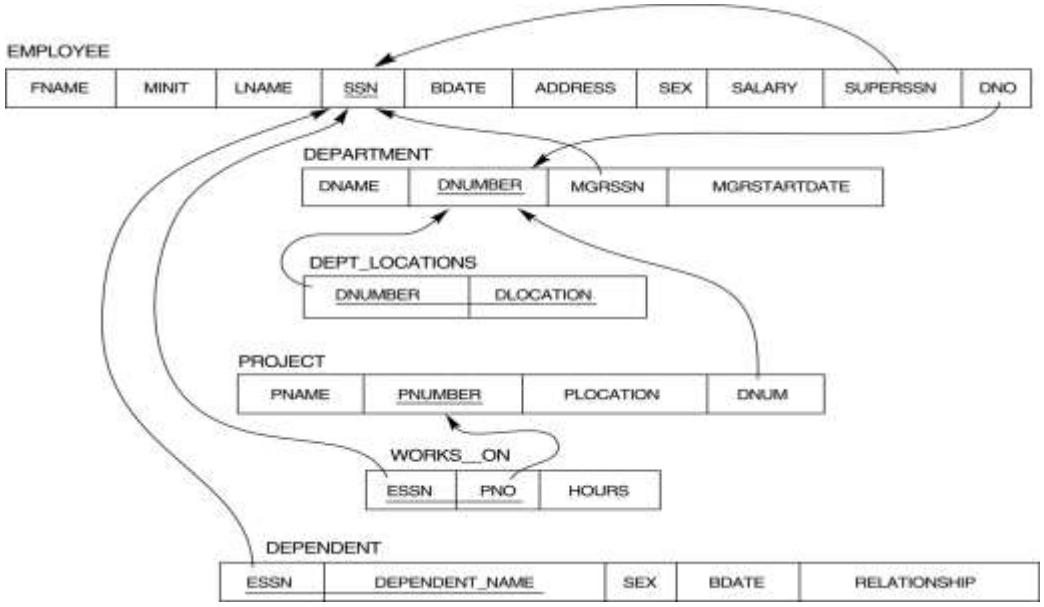
6. Retrieve the names, addresses, and number of books checked out for all borrowers who have more than five books checked out.

```
SELECT      Name, Address, COUNT(BookId, BranchId)
FROM        BORROWER as B, BOOK_LOANS as BL
WHERE       (BL.CardNo=B.CardNo)
GROUP BY    B.CardNo, Name, Address
HAVING      COUNT(BL.BookId, BranchId)>5
```

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

```
SELECT      Title, No_Of_Copies
FROM        BOOK_AUTHORS as BA, BOOK as B, BOOK_COPIES as BC,
            LIBRARY_BRANCH as LB
WHERE       B. BookId = BA.BookId and B.BookId = BC.BookId and
            BC.BookId = LB.BookId and
            (AuthorName=”Stephen King”) and(BranchName=”Central”)
```

Part II Give a database schema of a company as the following picture.



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

```
SELECT      FNAME, LNAME
FROM        EMPLOYEE, WORK_ON, PROJECT
WHERE       SSN = ESSN and PNO = PNUMBER and
            DNO = 5 and PNAME=”ProductX” and HOURS>10
```

2. For each project, list the project name and the total hours per week (by all employees) spent on that project if the total hours is greater than 10 hours/week

```
SELECT      PNAME, SUM(works_on.hours) AS "total_hours"
FROM        PROJECT as P, WORK_ON as W
WHERE       PNUMBER = PNO
GROUP BY    PNO, PNAME
HAVING      SUM(HOUR) >10
```

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

```
Select      FNAME, LNAME
From        EMPLOYEE, WORK_ON
Where       SSN = ESSN
Group by    SSN, FNAME, LNAME
HAVING      count(PNO) = (select count (PNUMBER) from PROJECT)
```

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

```
SELECT    LNAME, FNAME
FROM      EMPLOYEE as E
WHERE     NOT EXISTS ( SELECT    *
                       FROM      WORK_ON
                       WHERE     ESSN = E.SSN)
```

5. 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    E.fname, E.lname, E.address
from      employee as E, works_on as W, project as P
where     ssn = essn and pno = pnumber and Pno = PNumber and
          PLocation = "Houston" and not exist (select    *
                                               From      Dept_Location
                                               Where     DNumber = E.DNo and
                                                         DLocation = "Houston")
```

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

```
select    LNAME
from      EMPLOYEE
where     SSN in (select MGRSSN from DEPARTMENT) and
          SSN not in (select ESSN from DEPENDENT)
```

7. Find details of those employees whose salary is greater than the average salary for all employees. Output salary in descending order.

```
select    *
from      EMPLOYEE
where     SALARY >  (select  avg(SALARY)
                     from    EMPLOYEE)
order by SALARY desc;
```

8. Find details of those employees whose salary is greater than the average salary for all employees in his/her department. Output salary in ascending order.

```
select    *
from      EMPLOYEE as E
where     salary >        (select  avg(salary)
                           from    Employee
                           where DNO = E.DNO)
order by salary
```