National University of Computer and Emerging Sciences, Lahore Campus

Course:	**Database Systems**
Program:	**BS(Computer Science)**
Instructor:	**Muhammad Ishaq Raza**
Practice Problem:	**SQL (3) - SOLUTION**

Question 1: Consider the LIBRARY relational database schema given below which is used to keep track of books, borrowers, and book loans.

Book (book_id, title, publisher_name)
Book_Author (book_id, author_name)
Publisher(name, address, phone)
Book_Copies (book_id, branch_id, no_of_copies)
Book_loan (book_id, branch_id, card_no, date_out, due_date)
Library_Branch (branch_id, name, address)
Borrower (Card_no, name, address, phone, age, gender)

Write down SQL statements for the following queries:

i. List complete details of all library branches.
ii. List all the books reserved by the borrowers aged less than 30 years.
iii. Find the library branches which do not have any copy of book whose book_id is EE1234.
iv. Find the library branch which has the highest number of total books than all other branches.
v. Find out the total number of copies of the book titled *Emma* are possessed by each library branch?
vi. For each library branch, retrieve the branch name and the total number of books loaned out from that branch.
vii. Retrieve the names of borrowers who have not borrowed books from library branch that is in Faisalabad.
viii. Retrieve the names of the books that are loaned by borrower with card no 1000 *only.*
ix. How many library branches are there?

Question 2. Write down SQL statements for the following queries.

Consider the following Movie database for all the questions; for simplicity assume that the title of a movie is unique. The length of the movie is its running time in minutes, and net worth of the studio is its monetary value in dollars. The underlined attributes are part of the primary key. Foreign keys are studioName, actorSSN, and movieTitle.

Movie (Title, Year, Length, StudioName, ProductionCost)

StarsIn (ActorSSN, MovieTitle)

Actor (SSN, FirstName, LastName, Age, Gender, Address)

Studio (Name, Address, Networth)

i. List all movies made during time period 1970-1990 with a production cost below $7500000, in ascending order of cost.

ii. List the movies for which no production cost has been specified.

iii. Find the minimum, maximum and average production cost of movies made by Fox Studio?

iv. List the number of movies produced by each studio, consider only those studios that have net worth of at least 10 million US dollars and produced at least 5 movies.

v. Find those actors who starred in at least 3 movies made in 1990-1995 by studios based in California.

vi. List those actor SSN pairs whose last name is the same.

vii. List the name of those actors who worked together in the same movie.

viii. List those actors who have not yet starred in any movie.

ix. Display the name and age of all actors who have appeared in any movie made by Fox Studio but not in the movie made by Universal Studios.

x. Display the name and age of all actresses who have starred in at least 2 movies made by Universal Studios as well as in at least 3 movies made by 21st Century Fox.

xi. List all Studios which have net worth greater than Universal Studios'.

xii. List those movie titles along with the studio name in which Actor Tom Hanks did not star. (Note: Name is divided into first name and last name in Actor table)

xiii. List those actresses who costarred in at least one movie in which actor Harrison Ford has starred.

xiv. List all those studios that have made more movies than Universal Studios or 21st Century Fox during 1990-2000.

xv. Find those actresses who have worked in more movies than all actors.

xvi. Find the actresses who have worked on all the movies produced by 'Disney' studio in year 2000.

Solution:

Question 1:

```
Use Assignment2
go

--Book (book_id, title, publisher_name)
--Book_Author (book_id, author_name)
--Publisher(name, address, phone)
--Book_Copies (book_id, branch_id, no_of_copies)
--Book_loan (book_id, branch_id, card_no, date_out, due_date)
--Library_Branch (branch_id, name, address)
--Borrower (Card_no, name, address, phone, age, gender)

--Book (book_id, title, publisher_name)
Create Table Book
(
  book_id varchar(10) primary key,
  title varchar(50),
  publisher_name varchar(50)

)

--Book_Author (book_id, author_name)
Create Table Book_Author(
 book_id varchar(10) primary key,
 author_name varchar(100)
)

--Publisher(name, address, phone)
Create Table Publisher
(
  name varchar(50) primary key,
  address varchar(100),
  phone nvarchar(15)
)

--Library_Branch (branch_id, name, addres
Create Table Library_Branch
(
   branch_id int primary key,
         name varchar(50),
         address varchar(50)
)

--Book_Copies (book_id, branch_id, no_of_copies)
Create Table Book_Copies
(
  book_id varchar(10) foreign key references Book (book_id),
  branch_id int foreign key references Library_Branch (branch_id),
  no_of_copies int
)

--Book_loan (book_id, branch_id, card_no, date_out, due_date)
Create Table Book_Loan
(
  book_id varchar(10) foreign key references Book (book_id),
  branch_id int foreign key references Library_Branch (branch_id),
  card_no int,
  date_out date,
  due_date date


)


--Borrower (Card_no, name, address, phone, age, gender)
Create Table Borrower
(
  card_no int primary key,
  name varchar(50),
  address varchar(50),
  phone varchar(15),
  age int,
  gender char check (Gender In ('F','M'))

)

--i.	List complete details of all library branches.
Select *
From Library_Branch

--ii.	List all the books reserved by the borrowers aged less than 30 years.
Select *
From Book_Loan Join Borrower On Book_Loan.card_no=Borrower.card_no
where Borrower.age<30


---iii. Find the library branches which do not have any copy of book whose book_id is EE1234.
Select Library_Branch.branch_id
From Library_Branch
where Library_Branch.branch_id Not In  (
                       Select Library_Branch.branch_id
                                                                         From Library_Branch
Join Book_Copies On Library_Branch.branch_id=Book_Copies.branch_id
                                                                         where
Book_Copies.book_id='EE1234'

                                                                       )
```

```
--alternative solution
Select Library_Branch.branch_id
From Library_Branch
Except
Select Library_Branch.branch_id
From Library_Branch Join Book_Copies On Library_Branch.branch_id=Book_Copies.branch_id
where Book_Copies.book_id='EE1234'

--iv. Find the library branch which has the highest number of total books than all other branches.
Select Top 1 Library_Branch.branch_id, count(*) as TotalBooks
From Library_Branch Join Book_Copies On Library_Branch.branch_id=Book_Copies.branch_id
group by Library_Branch.branch_id
order by TotalBooks DESC

--v. Find out the total number of copies of the book titled Emma are possessed by each library branch?
Select Library_Branch.branch_id, Book_Copies.no_of_copies As "Total Books Titled Emma"
From Library_Branch Join Book_Copies On Library_Branch.branch_id= Book_Copies.branch_id Join
Book On Book_Copies.book_id=.book_id
where Book.title='Emma'
group By Library_Branch.branch_id


--vi.    For each library branch, retrieve the branch name and the total number of books loaned out from
that branch.
Select Library_Branch.name, Count(*) "Total Books Loaned"
From Library_Branch Join Book_Loan On Library_Branch.branch_id=Book_Loan.branch_id
group by Library_Branch.name


--vii. Retrieve the names of borrowers who have not borrowed books from library branch that is in
Faisalabad
Select Borrower.name
From Borrower
Except
Select Borrower.name
From Borrower Join Book_Loan On Borrower.card_no=Book_Loan.card_no Join Library_Branch On
Book_Loan.branch_id= Library_Branch.branch_id
where Library_Branch.address Like '%Faisalabad%'

--viii.   Retrieve the names of the books that are loaned by borrower with card no 1000 only.
Select Book.title
From Book Join Book_Loan On Book.book_id=Book_Loan.book_id
where Book_Loan.card_no=1000

--ix.    How many library branches are there?
Select count(*)  "Total Library Branches"
From Library_Branch
```

Questions 2:

```
Create Database Assignment2
go
--use Assignment2


--Movie (Title, Year, Length, StudioName, ProductionCost)
--StarsIn (ActorSSN, MovieTitle)
--Actor (SSN, FirstName, LastName, Age, Gender, Address)
--Studio (Name, Address, Networth)

Create Table Movie
(  title varchar(100) primary key,
   year int,

   length int,
   productionCost float
)



Create Table Actor(
 SSN int primary key,
 firstName varchar(40),
 lastName varchar(40),
 age int,
 gender char check (gender in ('Male', 'Female')),
 address varchar(100)
)

Create Table StarsIn
(
   actorSSN int foreign key references Actor (SSN),
   movieTitle varchar (100) foreign key references Movie (title)


)

Create Table Studio
(
   name varchar(100)  primary key,
   address varchar(100),
   networth int

)

alter table Movie Add studioName varchar(100) foreign key references Studio (name)

--i.        List all movies made during time period 1970-1990 with a production cost below $7500000, in
ascending order of cost.
```

```
Select *
From Movie
where( Movie.year Between 1970 AND 1990 ) AND productionCost<75000
ORDER By Movie.productionCost ASC

--ii.     List the movies for which no production cost has been specified.
Select *
From Movie
where movie.productionCost is NULL

--iii. Find the minimum, maximum and average production cost of movies made by Fox Studio?
Select min(Movie.productionCost), max(Movie.productionCost), Avg(Movie.productionCost)
From Movie
where Movie.studioName='Fox Studio'


--iv.List the number of movies produced by each studio, consider only those studios that have networth at
least 10 million US dollars and produced at least 5 movies.
Select  Movie.studioName, count(*) as "Total Movies Made"
From Movie Join Studio On Movie.studioName=Studio.name
where Studio.networth>1000000
group By Movie.studioName
having count(*)>=5




--v. Find those actors who starred in at least 3 movies made in 1990-1995 by studios based in California.
Select Actor.firstName
From Actor Join StarsIn ON Actor.SSN=StarsIn.actorSSN JOIN Movie On
StarsIn.movieTitle=Movie.title JOIN Studio ON Movie.studioName=Studio.name
where (Movie.year Between 1990 AND 1995) AND studio.address LIKE '%California%'
group by  Actor.firstName
having count(*)>2


--vi. List those actor SSN pairs whose last name is the same
Select A1.SSN as Actor_1, A2.SSN as Actor_2
From Actor A1 Join Actor A2 On A1.lastName= A2.lastName
where A1.SSN!= A2.SSN




--vii. List the name of those actors who worked together in the same movie.
Select A1.firstName, A2.firstName
From Actor A1  JOIN StarsIn S1 ON A1.SSN=S1.actorSSN, Actor A2 JOIN StarsIn S2 ON
A2.SSN=S2.actorSSN
where  A1.SSN!= A2.SSN  AND S1.movieTitle=S2.movieTitle
```

```
--viii.    List those actors who have not yet starred in any movie.
Select Actor.firstName
From Actor left Join StarsIn On Actor.SSN=StarsIn.actorSSN
where StarsIn.actorSSN IS NULL


--SET Operations

--ix. Display the name and age of all actors who have worked with Fox Studio but not with Universal
Studio.
Select Actor.firstName, Actor.age
From Actor Join StarsIn ON Actor.SSN=StarsIn.actorSSN Join Movie ON
Movie.title=StarsIn.movieTitle
where studioName='Fox Studio'
Except
Select Actor.firstName, Actor.age
From Actor Join StarsIn ON Actor.SSN=StarsIn.actorSSN Join Movie ON
Movie.title=StarsIn.movieTitle
where studioName='Universal Studio'

--x.  Display the name and age of all actresses who have starred in at  least 2 movies made
--    by Universal Studio as well as in at least 3 movies made by 21st Century Fox.  (Hint: Use set, join
and group by)
 Select Actor.firstName+' '+Actor.lastName as "Actor Name", Actor.age
 From Actor Join StarsIn On Actor.SSN=StarsIn.actorSSN Join Movie On StarsIn.movieTitle=
Movie.title
 where Actor.gender='Female' AND Movie.studioName='21st Century Fox'
 group by Movie.studioName, Actor.firstName, Actor.lastName, Actor.age
 having count(*)>=2
 Intersect
  Select Actor.firstName+' '+Actor.lastName as "Actor Name", Actor.age
 From Actor Join StarsIn On Actor.SSN=StarsIn.actorSSN Join Movie On StarsIn.movieTitle=
Movie.title
 where Actor.gender='Female' AND Movie.studioName='Universal Studio'
 group by Movie.studioName, Actor.firstName, Actor.lastName, Actor.age
 having count(*)>=3

--NESTED QUERIES

--xi.    List all Studios which have net worth greater than Universal Studio’s
Select Studio.name
From Studio
where Studio.networth> ( Select Studio.networth
               From Studio
                                              where Studio.name='Universal Studio'
                                        )
```

```
--xii. List those movie titles along with the movie studio name in which Actor Tom Hanks did not star.
Select Movie.name
From Movie
where Not Exists ( Select Actor.firstName
                From Actor Join StarsIn On Actor.SSN= StarsIn.actorSSN
                                              where Actor.firstName+' '+Actor.lastName='Tom Hanks' AND
StarsIn.movieTitle=Movie.title

                                             )
--xiii. List those actresses who costarred in at least one movie in which actor Harrison Ford has starred.
Select Actor.SSN, Actor.firstName
From Actor Join StarsIn S On Actor.SSN= S.actorSSN
Where Actor.gender='Female' AND Exists (
                                      Select *
                                                                                                                     From Actor
JOIN StarsIn On Actor.SSN= StarsIn.actorSSN
                                                                                                                     where
(actor.firstName+' '+actor.lastName)='Harrison Ford' AND StarsIn.movieTitle=S.movieTitle
                                 )

--xiv. List all those studios that have made more movies than Universal Studio or 21st Century Fox from
1990-2000.
Select Studio.name
From Studio JOIN Movie On Studio.name=Movie.studioName
where Studio.name!='Universal Studio' AND Studio.name!='21st Century Fox' AND Movie.year>=1990
AND Movie.year<=2000
group by Studio.name
having count(*) > ANY (
                  Select  count(*)
                                                        From Studio Join Movie On Studio.name=Movie.studioName
                                                        where (Studio.name='Universal Studio' OR Studio.name='21st
Century Fox') AND Movie.year>=1990 AND Movie.year<=2000
                                                        group by studio.name
             )


--xv.     Find those actresses who have worked in more movies than all actors.
Select Actor.SSN, Actor.firstName
From Actor JOIN StarsIn ON Actor.SSN=StarsIn.actorSSN
where Actor.gender='Female'
group by Actor.SSN, Actor.firstName
having count(*)>  (
                        Select top 1 count(*)
                                                                             From Actor Join StarsIn ON Actor.SSN=
StarsIn.actorSSN
                                                                             where Actor.gender='Male'
                                                                             group by Actor.SSN
                                                                             Order by count(*) DESC

                   )
```

```
 --xvi.Find the actresses who have worked on all the movies produced by ‘Disney’ studio in year 2000.
Select Actor.firstName
From Actor Join StarsIn On Actor.SSN=StarsIn.actorSSN Join Movie ON Movie.title=StarsIn.movieTitle
where Movie.studioName='Disney' AND Movie.year=2000 AND gender='Female'
group By Actor.firstName
having count(*)= ( Select count (*)
           From Movie
                                 where Movie.studioName='Disney' AND year=2000)
```