Ahsanullah University of Science and Technology (AUST)
Department of Computer Science and Engineering

Offline 1

Course No.: CSE4126

Course Title: Distributed Database Systems Lab

Date of Submission-
23.05.23

Submitted To-
Zarin Tasnim Shejuti

&

Sanzana Karim Lora

Submitted By-
Name- Kazi Atiqur Rahman
ID- 190204086
Section- B1
Year- 4th
Semester- 1st
Department- CSE

Question: Consider the following database schema that is maintained at a university library:

Student(studentID: integer, name: string, phone: string, age: integer)

Borrows(studentID: integer, bookID: integer, dateBorrowed: date)

Book(bookID: integer, authorID: integer, title: string, genre: string)

Author(authorID: integer, name: string, age: integer)

Tasks:

1. Create the four tables and insert necessary data in the tables using SQL commands

(necessary data can be within 5 to 10 rows).

2. Write the following queries in SQL –

a. Show the name of the students who borrowed the book titled “Gulliver’s Travels”.

b. Show the age of the oldest author from among those who have published books belonging to the genre “Non-Fiction”.

c. Show the phone of the student who borrowed the book titled “Gitanjali” more than twice.

Solution:

```
clear screen;

drop table borrows;

drop table book;

drop table author;

drop table student;

create table student(studentID number primary key, name varchar2(30), phone varchar2(20), age number);

create table author (authorID number primary key, name varchar2(30), age number);

create table book (bookID number primary key, authorID number, title varchar2(60), genre varchar2(20), foreign key (authorID) references author(authorID));

create table borrows(studentID number, bookID number, dateBorrowed date, foreign key (studentID) references student(studentID), foreign key (bookID) references book(bookID));

--insert values into student table

insert into student values (1, 'Atiq', '0195123641', 23);

insert into student values (2, 'Anik', '0180452210', 22);

insert into student values (3, 'Bristy', '0196856937', 23);
```

```
insert into student values (4, 'Faizah', '0151235345', 21);
insert into student values (5, 'Soniya', '0194956721', 21);
insert into student values (6, 'Raihan', '0163173862', 22);
--insert values into author table
insert into author values (109, 'Humayun Ahmed', 65);
insert into author values (110, 'Jhumpa Lahiri', 54);
insert into author values (111, 'Khaled Hosseini', 56);
insert into author values (112, 'Rohinton Mistry', 67);
insert into author values (113, 'Arundhati Roy', 59);
insert into author values (114, 'Sukumar Ray', 50);
insert into author values (115, 'Sharat Chandra Chattopadhyay', 84);
insert into author values (116, 'Rabindranath Tagore', 70);
insert into author values (117, 'Satyajit Ray', 71);
insert into author values (118, 'Jonathan Swift', 50);
--insert values into book table
insert into book values (216, 109, 'Misir Ali series', 'Mystery');
insert into book values (217, 110, 'The Namesake', 'Fiction');
insert into book values (218, 111, 'The Kite Runner', 'Non-Fiction');
insert into book values (219, 112, 'A Fine Balance', 'Non-Fiction');
insert into book values (220, 113, 'The God of Small Things', 'Fiction');
insert into book values (221, 114, 'Abol Tabol', 'Poetry');
insert into book values (222, 115, 'Devdas', 'Romance');
insert into book values (223, 116, 'Gitanjali', 'Poetry');
insert into book values (224, 117, 'Feluda series', 'Non-Fiction');
insert into book values (225, 118, 'Gulliver''s Travels', 'Satire');
--insert values into borrows table
insert into borrows values (6, 225, '21-JAN-2023');
insert into borrows values (1, 225, '26-JAN-2023');
insert into borrows values (2, 223, '24-JAN-2023');
```

```
insert into borrows values (3, 224, '26-JAN-2023');

insert into borrows values (1, 216, '21-JAN-2023');

insert into borrows values (2, 223, '29-JAN-2023');

insert into borrows values (3, 217, '28-JAN-2023');

insert into borrows values (3, 225, '29-JAN-2023');

insert into borrows values (5, 221, '12-JAN-2023');

insert into borrows values (2, 223, '12-MAY-2023');

commit;

SELECT STUDENT.NAME FROM STUDENT JOIN BORROWS ON STUDENT.STUDENTID =
BORROWS.STUDENTID JOIN BOOK ON BORROWS.BOOKID = BOOK.BOOKID WHERE BOOK.TITLE =
'Gulliver''s Travels';

SELECT MAX(AUTHOR.AGE) FROM AUTHOR JOIN BOOK ON AUTHOR.AUTHORID = BOOK.AUTHORID
WHERE BOOK.GENRE = 'Non-Fiction';

SELECT STUDENT.PHONE FROM STUDENT JOIN BORROWS ON STUDENT.STUDENTID =
BORROWS.STUDENTID JOIN BOOK ON BORROWS.BOOKID = BOOK.BOOKID WHERE BOOK.TITLE =
'Gitanjali' GROUP BY PHONE HAVING COUNT(STUDENT.STUDENTID) > 2;
```

Output:

```
NAME

------------------------------

Raihan

Atiq

Bristy

MAX(AUTHOR.AGE)

---------------

          71

PHONE

--------------------

0180452210
```