Database Final Exam

Total Marks : 10 x 10 = 100

This is a schema of the Mini Student Library Management System.
Answer question 1 - 5 from this schema.

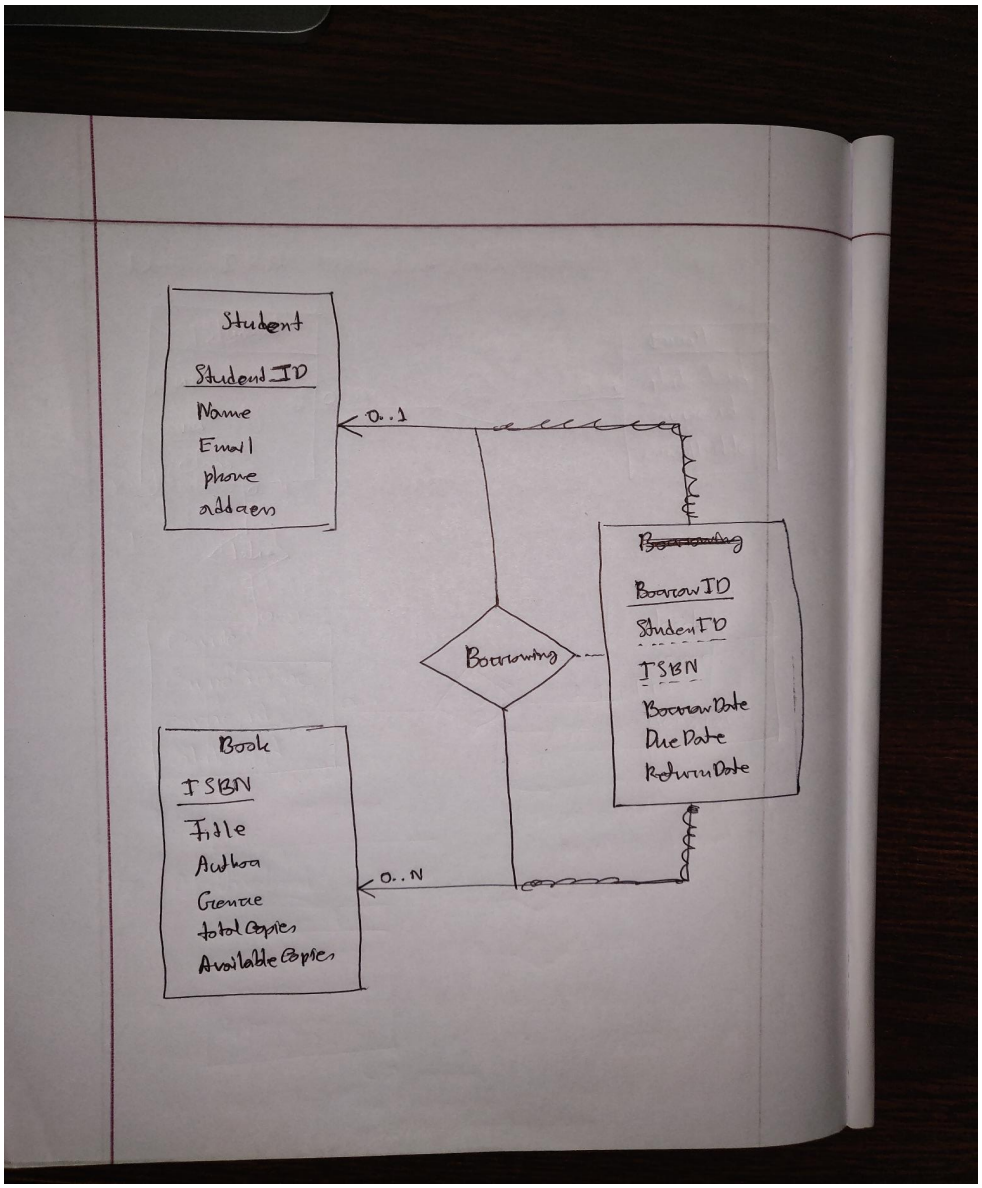
```
CREATE TABLE Student (
    StudentID INT AUTO_INCREMENT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Email VARCHAR(255) NOT NULL,
    Phone VARCHAR(15),
    Address TEXT
);

CREATE TABLE Book (
    ISBN VARCHAR(13) PRIMARY KEY,
    Title VARCHAR(255) NOT NULL,
    Author VARCHAR(255) NOT NULL,
    Genre VARCHAR(50),
    TotalCopies INT NOT NULL,
    AvailableCopies INT NOT NULL
);

CREATE TABLE Borrowing (
    BorrowID INT AUTO_INCREMENT PRIMARY KEY,
    StudentID INT,
    ISBN VARCHAR(13),
    BorrowDate DATE NOT NULL,
```

```
    DueDate DATE NOT NULL,
    ReturnDate DATE,
    FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
    FOREIGN KEY (ISBN) REFERENCES Book(ISBN)
);
```

1. Make an ER Diagram of this Schema
   Ans :



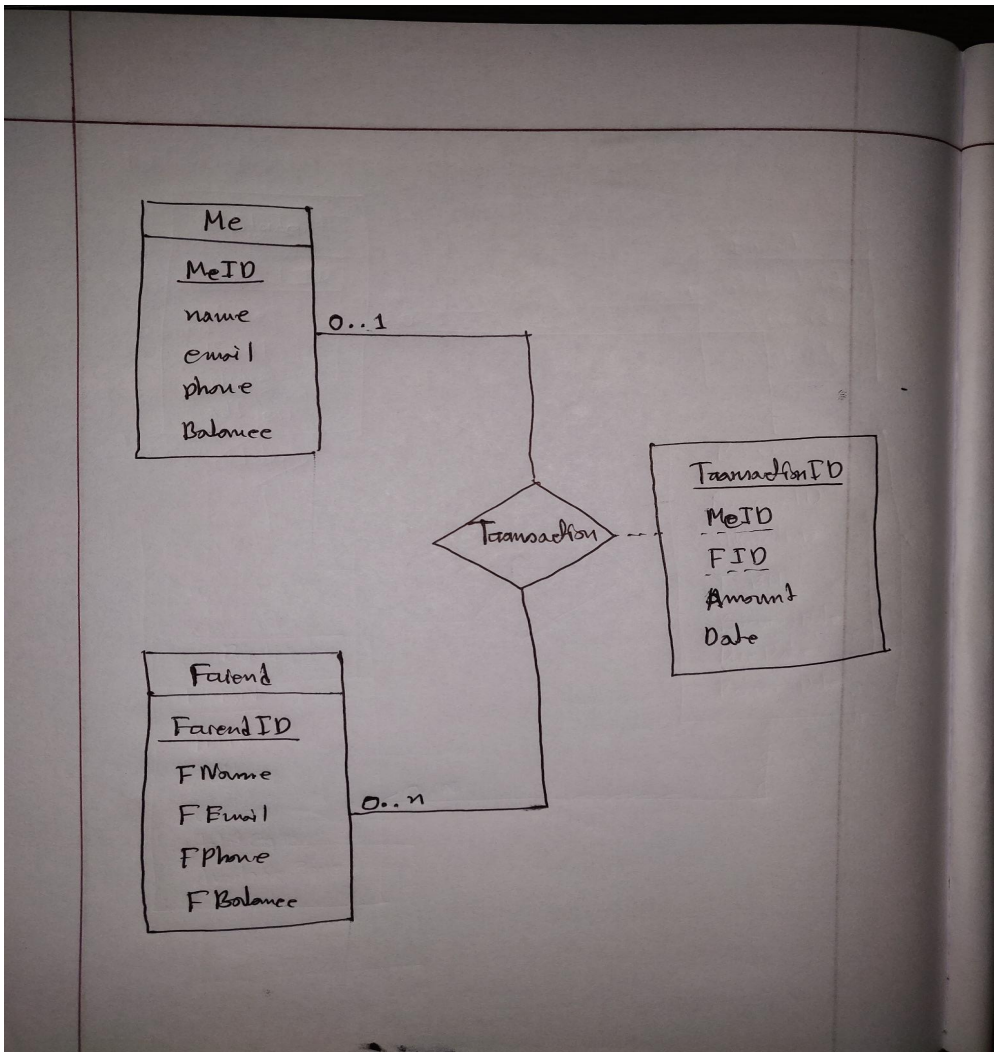
2. Insert a new borrowing record for a student (e.g., StudentID 3) for a book with the most available copies.
   Ans :
   ```
   select ISBN
   from Book
   order by AvailableCopies desc;
   ```

   E.g- ISBN = 12343234
   After we get the isbn number we will use this to insert a new record.

   ```
   Insert into
   Borrowing(BorrowId,StudentId,ISBN,BorrowDate,DueDate,ReturnDat
   e)
   values(1,3,12343234,06-11-2023,10-11-2023,11-11-2023);
   ```

3. Using Update Query, decrease the available copies of a book (e.g., ISBN '9781234567890') by 1 when a student borrows it.
   Ans :
   ```
   update Book
   set AvailableCopies = AvailableCopies-1
   where ISBN = (select ISBN
                 from Borrowing
                 where ISBN = '9781234567890'
   );
   ```
4. Retrieve the names of students who have borrowed the most books.
   Ans:
   ```
   select s.Name
   from Student as s
   join Borrowing as b
   on s.StudentId = b.StudentId
   group by s.StudentId, s.Name
   order by count(b.StudentId) desc
   limit 1;
   ```

5. Retrieve the books that are overdue (i.e., the return date is before the current date).
   Ans :
   ```
   select Title
   from Book
   where ISBN in (select ISBN
                  from Borrowing
                  where ReturnDate < curdate()
   );
   ```
6. You want to make a mobile banking platform for sending and receiving money from your friends. Make an ERD of this system. (Keep it simple)
   Ans :



7. Explain UNION and UNION ALL set operations in MySQL
   Ans : UNION stores data in sorted order and duplication is not allowed.
   a = {1,2,3,4,5}
   b = {2,3,4}
   a union b ={1,2,3,4,5} union {2,3,4} ----> {1,2,3,4,5}

   In contrast UNION ALL union stores duplicate elements also.
   a = {1,2,3,4,5}
   b = {2,3,4}
   a union all b ={1,2,3,4,5} union {2,3,4} ----> {1,2,3,4,5,2,3,4}

8. There is a table named Employee. In that table there is a field named Salary. Determine the second lowest salary.
   Ans :
   ```
   select max(salary)
   from employees
   where salary<(select max(salary)
                                 from employees
   );
   ```

9. There are tables named Employee, Job History, Department.
   a. Use ON DELETE CASCADE on Job History for deleting Employee
      Ans:
      ```
      create table JobHistory(
      jobId varchar(5) primary key,
      designation varchar(20) not null,
      location varchar(50) not null,
      empId varchar(5),
      foreign key(empId) references Employee(empId) on delete cascade
      );
      ```

b. Use ON DELETE SET NULL on Employee for deleting Department

Ans :

```
create table Department(
departmentId varchar(5) primary key,
departmentName varchar(20) not null,
);

create table Employee(
empId varchar(5) primary key,
empName varchar(20) not null,
empDepartment varchar(20)not null,
empSalary float check(empSalary>0),
departmentId varchar(5),
foreign  key (departmentId) references
Department(departmentId) on delete set null
);
```

10. In this course, which topic you found most interesting. Explain the topic in short and why you found it most interesting?

Ans : I found join operations are quite useful and interesting in this database management course.

There are 4 types of join operations we can perform: inner,left,right,cross join.

*inner join or join :

it joins the common elements from both table

* left outer join or left join:

If there are 2 tables and the left join will show all elements of table which is on the left and will show null at the right table's location where there are no common elements between these two tables.

* right join :

If there are 2 tables and the right join will show all elements of table which is on the right and will show null at the left table's location where there are no common elements between these two tables.

* cross join:
it will show both table elements if any of the elements don't match then on their respective part it will show null.

I find these concepts most interesting because I can perform different queries between different tables.