SQL-COMPREHENSIVE ASSESMENT

Topic : Library Management System

You are going to build a project based on Library Management System. It keeps track of all information about books in the library, their cost, status and total number of books available in the library.

Create a database named library and following TABLES in the database:

1. Branch
2. Employee
3. Books
4. Customer
5. IssueStatus
6. ReturnStatus

Attributes for the tables:

1. Branch

- Branch_no - Set as PRIMARY KEY
- Manager_Id
- Branch_address
- Contact_no

```
CREATE DATABASE Library;
USE Library;

CREATE TABLE Branch(
Branch_no int primary key not null,
Manager_id int,
Branch_Address varchar(100),
Contact_no varchar(15)
);
INSERT INTO Branch VALUES
(1, 101, '123 Main St', '555-1234'),
(2, 102, '456 Oak St', '555-5678');

select * from Branch;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Branch_no	Manager_id	Branch_Address	Contact_no
1	101	123 Main St	555-1234
2	102	456 Oak St	555-5678
NULL	NULL	NULL	NULL

2. Employee

- Emp_Id – Set as PRIMARY KEY
- Emp_name
- Position
- Salary
- Branch_no - Set as FOREIGN KEY and it refer Branch_no in Branch table

```
CREATE TABLE Employee(
Emp_id int primary key not null,
Emp_name varchar(50),
Position varchar(50),
Salary int,
Branch_no int,
foreign key(Branch_no) references Branch(Branch_no)
);

INSERT INTO Employee VALUES
(1, 'Ashwin', 'Manager', 60000, 1),
(2, 'Sara', 'Clerk', 45000, 1),
(3, 'Lilly', 'Manager', 70000, 2),
(4, 'Zackeriya', 'Clerk', 47000, 2);
select * from Employee;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Emp_id	Emp_name	Position	Salary	Branch_no
1	Ashwin	Manager	60000	1
2	Sara	Clerk	45000	1
3	Lilly	Manager	70000	2
4	Zackeriya	Clerk	47000	2
NULL	NULL	NULL	NULL	NULL

3. Books

- ISBN - Set as PRIMARY KEY
- Book_title
- Category
- Rental_Price
- Status [Give yes if book available and no if book not available]
- Author
- Publisher

```
CREATE TABLE Books(
ISBN varchar(20) primary key,
Book_title varchar(100),
Category varchar(50),
Rental_price float,
Status varchar(3) CHECK (Status IN ('yes', 'no')),
Author varchar(50),
Publisher varchar(50)
);

INSERT INTO Books VALUES
('978-3-16-148410-0', 'Book A', 'Fiction', 20.00, 'yes', 'Author A', 'Publisher A'),
('978-1-23-456789-7', 'Book B', 'History', 15.00, 'no', 'Author B', 'Publisher B'),
('978-0-12-345678-9', 'Book C', 'Science', 25.00, 'yes', 'Author C', 'Publisher C');
select * from Books;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

ISBN	Book_title	Category	Rental_price	Status	Author	Publisher
978-0-12-345678-9	Book C	Science	25	yes	Author C	Publisher C
978-1-23-456789-7	Book B	History	15	no	Author B	Publisher B
978-3-16-148410-0	Book A	Fiction	20	yes	Author A	Publisher A
NULL	NULL	NULL	NULL	NULL	NULL	NULL

4. Customer

- Customer_Id - Set as PRIMARY KEY
- Customer_name
- Customer_address
- Reg_date

```
CREATE TABLE Customer(
Customer_id int primary key,
Customer_name varchar (50),
Customer_address varchar(100),
Reg_date datetime
);

INSERT INTO Customer VALUES
(1, 'Alice Green', '789 Pine St', '2021-12-31'),
(2, 'Bob White', '321 Cedar St', '2022-01-15'),
(3, 'Charlie Black', '654 Elm St', '2023-05-20');

select * from Customer;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Customer_id	Customer_name	Customer_address	Reg_date
1	Alice Green	789 Pine St	2021-12-31 00:00:00
2	Bob White	321 Cedar St	2022-01-15 00:00:00
3	Charlie Black	654 Elm St	2023-05-20 00:00:00
NULL	NULL	NULL	NULL

5. IssueStatus

- Issue_Id - Set as PRIMARY KEY
- Issued_cust – Set as FOREIGN KEY and it refer customer_id in CUSTOMER table Issued_book_name
- Issue_date
- Isbn_book – Set as FOREIGN KEY and it should refer isbn in BOOKS table

```
CREATE TABLE IssueStatus(
Issue_id int primary key,
Issued_cust int,
Issue_date date,
ISBN_book varchar(20),
foreign key(Issued_cust) references Customer(Customer_id),
foreign key(ISBN_book) references Books(ISBN)
);

INSERT INTO IssueStatus VALUES
(1, 1, '2023-06-05', '978-3-16-148410-0'),
(2, 3, '2023-06-10', '978-1-23-456789-7');
select * from IssueStatus;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Issue_id	Issued_cust	Issue_date	ISBN_book
1	1	2023-06-05	978-3-16-148410-0
2	3	2023-06-10	978-1-23-456789-7
NULL	NULL	NULL	NULL

6. ReturnStatus

- Return_Id - Set as PRIMARY KEY
- Return_cust
- Return_book_name
- Return_date
- Isbn_book2 - Set as FOREIGN KEY and it should refer isbn in BOOKS table

```
82
83 • CREATE TABLE ReturnStatus(
84   Return_id int primary key,
85   Return_cust int,
86   Return_book_name varchar(100),
87   Return_date date,
88   ISBN_book2 varchar(20),
89   foreign key(ISBN_book2) references Books(ISBN)
90   );
91 • INSERT INTO ReturnStatus VALUES (1, 1, 'Book A', '2023-06-15', '978-3-16-148410-0');
92 • select * from ReturnStatus;
93
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Return_id	Return_cust	Return_book_name	Return_date	ISBN_book2
1	1	Book A	2023-06-15	978-3-16-148410-0
NULL	NULL	NULL	NULL	NULL

1. Retrieve the book title, category, and rental price of all available books.

```
95   -- 1. Retrieve the book title, category, and rental price of all available books.
96
97 • select book_title,Category,rental_price from Books where status='yes';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

book_title	Category	rental_price
Book C	Science	25
Book A	Fiction	20

2. List the employee names and their respective salaries in descending order of salary.

```
99   -- 2. List the employee names and their respective salaries in descending order of salary.
100
101 • select Emp_name,Salary from employee order by Salary desc;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Emp_name	Salary
Lilly	70000
Ashwin	60000
Zackeriya	47000
Sara	45000

3. Retrieve the book titles and the corresponding customers who have issued those books.

```
103  -- 3. Retrieve the book titles and the corresponding customers who have issued those books.
104
105 • select Books.Book_title as Books , Customer.Customer_name as Issued_Customer from Books inner join Customer,IssueStatus
106  where IssueStatus.ISBN_book = Books.ISBN and IssueStatus.Issued_cust=customer.Customer_id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Books	Issued_Customer
Book A	Alice Green
Book B	Charlie Black

4. Display the total count of books in each category.

```
108     -- 4. Display the total count of books in each category.
109
110 •   select Category,count(*) as No_of_Books from Books group by category order by category desc;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Category	No_of_Books
Science	1
History	1
Fiction	1

5. Retrieve the employee names and their positions for the employees whose salaries are above Rs.50,000.

```
111
112     -- 5. Retrieve the employee names and their positions for the employees whose salaries are above Rs.50,000.
113
114 •   select Emp_name as Employee_Name , Position from Employee where Salary>50000;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Employee_Name	Position
Ashwin	Manager
Lilly	Manager

6. List the customer names who registered before 2022-01-01 and have not issued any books yet.

```
118 •   select customer_name from Customer left join IssueStatus on IssueStatus.Issued_cust = Customer.Customer_id
119     where Customer.reg_date < '2022-01-01' and IssueStatus.issued_cust is Null;
120
121
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_name

7. Display the branch numbers and the total count of employees in each branch.

```
121     -- 7. Display the branch numbers and the total count of employees in each branch.
122
123 •   select Branch_no ,count(Branch_no) as Total_Number_of_Employee from Employee group by Branch_no ;
124
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Branch_no	Total_Number_of_Employee
1	2
2	2

8. Display the names of customers who have issued books in the month of June 2023.

```
125
126     -- 8. Display the names of customers who have issued books in the month of June 2023.
127
128 •   select customer_name from customer left join IssueStatus on IssueStatus.Issued_cust = Customer.Customer_id
129     where IssueStatus.Issue_date between '2023-06-01' and '2023-06-30';
130
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_name
Alice Green
Charlie Black

9. Retrieve book_title from book table containing history.

```
131     -- 9. Retrieve book_title from book table containing history.
132
133 •   select book_title from Books where Category='History';
134
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

book_title
Book B

10.Retrieve the branch numbers along with the count of employees for branches having more than 5 employees

```
135    -- 10.Retrieve the branch numbers along with the count of employees for branches having more than 5 employees
136
137 •  select Branch_no ,count(Branch_no) as Total_Number_of_Employee from Employee group by Branch_no having count(Branch_no)>5;
138
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Branch_no	Total_Number_of_Employee

11. Retrieve the names of employees who manage branches and their respective branch addresses.

```
139    -- 11. Retrieve the names of employees who manage branches and their respective branch addresses.
140
141 •  select Emp_name , Branch.Branch_address from Employee right join Branch on
142     Employee.Branch_no=Branch.Branch_no where Employee.position='Manager';
143
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Emp_name	Branch_address
Ashwin	123 Main St
Lilly	456 Oak St

12. Display the names of customers who have issued books with a rental price higher than Rs. 25.

```
144    -- 12.  Display the names of customers who have issued books with a rental price higher than Rs. 25.
145
146 •  select customer.customer_name from customer join Issuestatus on IssueStatus.Issued_cust=customer.Customer_id
147    join Books on  Books.ISBN = IssueStatus.ISBN_book where Books.Rental_price > 25;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_name