

Mysql Comprehensive Assessment

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.

Screenshots of actual SQL queries implemented on workbench

1. Creation of database and tables

```
9830 • CREATE DATABASE library;
9831
9832 • USE library;
9833
9834
9835 • CREATE TABLE Branch (
9836     Branch_no INT PRIMARY KEY,
9837     Manager_Id INT,
9838     Branch_address VARCHAR(100),
9839     Contact_no VARCHAR(15)
9840 );
9841
9842 • CREATE TABLE Employee (
9843     Emp_Id INT PRIMARY KEY,
9844     Emp_name VARCHAR(50),
9845     Position VARCHAR(50),
9846     Salary DECIMAL(10, 2),
9847     Branch_no INT,
9848     FOREIGN KEY (Branch_no) REFERENCES Branch(Branch_no)
9849 );
9850
9851 • CREATE TABLE Books (
9852     ISBN VARCHAR(20) PRIMARY KEY,
9853     Book_title VARCHAR(100),
9854     Category VARCHAR(50),
9855     Rental_Price DECIMAL(10, 2),
9856     Status ENUM('yes', 'no'),
9857     Author VARCHAR(50),
9858     Publisher VARCHAR(50)
9859 );
-----
```

```

9861 • CREATE TABLE Customer (
9862     Customer_Id INT PRIMARY KEY,
9863     Customer_name VARCHAR(50),
9864     Customer_address VARCHAR(100),
9865     Reg_date DATE
9866 );
9867
9868 • CREATE TABLE IssueStatus (
9869     Issue_Id INT PRIMARY KEY,
9870     Issued_cust INT,
9871     Issued_book_name VARCHAR(100),
9872     Issue_date DATE,
9873     Isbn_book VARCHAR(20),
9874     FOREIGN KEY (Issued_cust) REFERENCES Customer(Customer_Id),
9875     FOREIGN KEY (Isbn_book) REFERENCES Books(ISBN)
9876 );
9877
9878 • CREATE TABLE ReturnStatus (
9879     Return_Id INT PRIMARY KEY,
9880     Return_cust INT,
9881     Return_book_name VARCHAR(100),
9882     Return_date DATE,
9883     Isbn_book2 VARCHAR(20),
9884     FOREIGN KEY (Isbn_book2) REFERENCES Books(ISBN)
9885 );
-----

```

2.Inserting values into the table

```

9887 • INSERT INTO Branch (Branch_no, Manager_Id, Branch_address, Contact_no) VALUES
9888 (1, 101, '123 Main St, City A', '555-0101'),
9889 (2, 102, '456 Oak Rd, City B', '555-0102'),
9890 (3, 103, '789 Pine Ave, City C', '555-0103');
9891
9892 • INSERT INTO Employee (Emp_Id, Emp_name, Position, Salary, Branch_no) VALUES
9893 (101, 'John Doe', 'Manager', 60000, 1),
9894 (102, 'Jane Smith', 'Manager', 62000, 2),
9895 (103, 'Bob Johnson', 'Manager', 61000, 3),
9896 (104, 'Alice Brown', 'Librarian', 45000, 1),
9897 (105, 'Charlie Davis', 'Librarian', 44000, 2),
9898 (106, 'Eve Wilson', 'Clerk', 35000, 3);
9899
9900 • INSERT INTO Books (ISBN, Book_title, Category, Rental_Price, Status, Author, Publisher) VALUES
9901 ('ISBN001', 'The Great Novel', 'Fiction', 20, 'yes', 'Author A', 'Publisher X'),
9902 ('ISBN002', 'History of Europe', 'History', 25, 'yes', 'Author B', 'Publisher Y'),
9903 ('ISBN003', 'Science Explained', 'Science', 30, 'no', 'Author C', 'Publisher Z'),
9904 ('ISBN004', 'The Art Book', 'Art', 35, 'yes', 'Author D', 'Publisher X'),
9905 ('ISBN005', 'World History', 'History', 28, 'yes', 'Author E', 'Publisher Y');
9906
9907 • INSERT INTO Customer (Customer_Id, Customer_name, Customer_address, Reg_date) VALUES
9908 (1, 'Mike Johnson', '321 Elm St, City A', '2021-05-15'),
9909 (2, 'Sarah Lee', '654 Maple Ave, City B', '2022-03-20'),
9910 (3, 'Tom Brown', '987 Oak Rd, City C', '2023-01-10'),
9911 (4, 'Emily Davis', '246 Pine St, City A', '2021-11-30'),
9912 (5, 'David Wilson', '135 Cedar Ln, City B', '2022-09-05');

9914 • INSERT INTO IssueStatus (Issue_Id, Issued_cust, Issued_book_name, Issue_date, Isbn_book) VALUES
9915 (1, 1, 'The Great Novel', '2023-06-01', 'ISBN001'),
9916 (2, 2, 'History of Europe', '2023-06-15', 'ISBN002'),
9917 (3, 3, 'The Art Book', '2023-07-01', 'ISBN004'),
9918 (4, 4, 'World History', '2023-06-20', 'ISBN005');
9919
9920 • INSERT INTO ReturnStatus (Return_Id, Return_cust, Return_book_name, Return_date, Isbn_book2) VALUES
9921 (1, 1, 'The Great Novel', '2023-06-15', 'ISBN001'),
9922 (2, 2, 'History of Europe', '2023-06-30', 'ISBN002');

```

3. Queries

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

```

9926 • SELECT Book_title, Category, Rental_Price
9927 FROM Books
9928 WHERE Status = 'yes';
9929

```

Book_title	Category	Rental_Price
The Great Novel	Fiction	20.00
History of Europe	History	25.00
The Art Book	Art	35.00
World History	History	28.00

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

```

9930 • SELECT Emp_name, Salary
9931 FROM Employee
9932 ORDER BY Salary DESC;
9933

```

Emp_name	Salary
Jane Smith	62000.00
Bob Johnson	61000.00
John Doe	60000.00
Alice Brown	45000.00
Charlie Davis	44000.00
Eve Wilson	35000.00

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

```

9934 • SELECT b.Book_title, c.Customer_name
9935 FROM Books b
9936 JOIN IssueStatus i ON b.ISBN = i.Isbn_book
9937 JOIN Customer c ON i.Issued_cust = c.Customer_Id;
9938
9939
9940

```

Book_title	Customer_name
The Great Novel	Mike Johnson
History of Europe	Sarah Lee
The Art Book	Tom Brown
World History	Emily Davis

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

```

9939 • SELECT Category, COUNT(*) as TotalBooks
9940 FROM Books
9941 GROUP BY Category;
9942

```

Category	TotalBooks
Fiction	1
History	2
Science	1
Art	1

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

```

9940 • SELECT Emp_name, Position
9941 FROM Employee
9942 WHERE Salary > 50000;
9943

```

Emp_name	Position
John Doe	Manager
Jane Smith	Manager
Bob Johnson	Manager

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

```
9944 • SELECT Customer_name
9945 FROM Customer
9946 WHERE Reg_date < '2022-01-01'
9947 AND Customer_Id NOT IN (SELECT DISTINCT Issued_cust FROM IssueStatus);
9948
9949
9950
```

<

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

Customer_name

Here the query has no output,so adding additional customer data into the database

```
• INSERT INTO Customer (Customer_Id, Customer_name, Customer_address, Reg_date) VALUES
(6, 'Lisa Green', '789 Birch St, City D', '2021-08-20'),
(7, 'Mark Taylor', '321 Cedar Ave, City E', '2021-12-15');
```

After update the result is

```
9944 • SELECT Customer_name
9945 FROM Customer
9946 WHERE Reg_date < '2022-01-01'
9947 AND Customer_Id NOT IN (SELECT DISTINCT Issued_cust FROM IssueStatus);
```

<

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

Customer_name
Lisa Green
Mark Taylor

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

```

9953 • SELECT Branch_no, COUNT(*) as TotalEmployees
9954 FROM Employee
9955 GROUP BY Branch_no;
9956
9957

```

Branch_no	TotalEmployees
1	2
2	2
3	2

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

```

9957 • SELECT DISTINCT c.Customer_name
9958 FROM Customer c
9959 JOIN IssueStatus i ON c.Customer_Id = i.Issued_cust
9960 WHERE YEAR(i.Issue_date) = 2023 AND MONTH(i.Issue_date) = 6;
9961
9962

```

Customer_name
Mike Johnson
Sarah Lee
Emily Davis

ix. Retrieve book_title from book table containing history.

```

9962 • SELECT Book_title
9963 FROM Books
9964 WHERE Category LIKE '%history%' OR Book_title LIKE '%history%';
9965

```

Book_title
History of Europe
World History

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

```
9972 • SELECT Branch_no, COUNT(*) as EmployeeCount
9973 FROM Employee
9974 GROUP BY Branch_no
9975 HAVING EmployeeCount > 5;
9976
9977
```

Branch_no	EmployeeCount
1	6

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

```
9977 • SELECT e.Emp_name, b.Branch_address
9978 FROM Employee e
9979 JOIN Branch b ON e.Emp_Id = b.Manager_Id;
9980
9981
```

Emp_name	Branch_address
John Doe	123 Main St, City A
Jane Smith	456 Oak Rd, City B
Bob Johnson	789 Pine Ave, City C

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

```
9978 • SELECT DISTINCT c.Customer_name
9979 FROM Customer c
9980 JOIN IssueStatus i ON c.Customer_Id = i.Issued_cust
9981 JOIN Books b ON i.Isbn_book = b.ISBN
9982 WHERE b.Rental_Price > 25;
9983
9984
```

<
Result Grid
Filter Rows: <input type="text"/>
Export:  Wrap Cell Conte
Customer_name
▶ Tom Brown
Emily Davis