## 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 • GREATE TABLE Branch (
             Branch_no INT PRIMARY KEY,
9836
9837
             Manager_Id INT,
             Branch address VARCHAR(100),
9838
             Contact_no VARCHAR(15)
9839
       );
9840
9841
9842 ● ⊖ CREATE TABLE Employee (
            Emp Id INT PRIMARY KEY,
9843
            Emp name VARCHAR(50),
9844
             Position VARCHAR(50),
9845
9846
             Salary DECIMAL(10, 2),
9847
             Branch_no INT,
             FOREIGN KEY (Branch_no) REFERENCES Branch(Branch_no)
9848
9849
       - );
9850
9851 ● ⊖ CREATE TABLE Books (
9852
             ISBN VARCHAR(20) PRIMARY KEY,
            Book title VARCHAR(100),
9853
            Category VARCHAR(50),
9854
             Rental_Price DECIMAL(10, 2),
9855
            Status ENUM('yes', 'no'),
9856
9857
            Author VARCHAR(50),
             Publisher VARCHAR(50)
9858
9859
       );
```

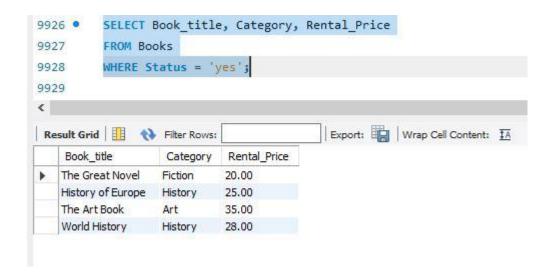
```
9861 • @ CREATE TABLE Customer (
             Customer Id INT PRIMARY KEY,
9862
             Customer name VARCHAR(50),
9863
             Customer_address VARCHAR(100),
9864
             Reg_date DATE
9865
9866
         );
9867
9868 • GREATE TABLE IssueStatus (
             Issue_Id INT PRIMARY KEY,
9869
             Issued cust INT,
9870
9871
             Issued_book_name VARCHAR(100),
             Issue date DATE,
9872
9873
             Isbn_book VARCHAR(20),
9874
             FOREIGN KEY (Issued_cust) REFERENCES Customer(Customer_Id),
             FOREIGN KEY (Isbn book) REFERENCES Books(ISBN)
9875
9876
       );
9877
9878 • CREATE TABLE ReturnStatus (
             Return_Id INT PRIMARY KEY,
9879
9880
             Return cust INT,
             Return book name VARCHAR(100),
9881
             Return_date DATE,
9882
9883
             Isbn_book2 VARCHAR(20),
             FOREIGN KEY (Isbn_book2) REFERENCES Books(ISBN)
9884
9885
       );
```

2. Inserting values into the table

```
INSERT INTO Branch (Branch no, Manager Id, Branch address, Contact no) VALUES
9887 •
         (1, 101, '123 Main St, City A', '555-0101'),
9888
         (2, 102, '456 Oak Rd, City B', '555-0102'),
9889
         (3, 103, '789 Pine Ave, City C', '555-0103');
9890
9891
9892 •
         INSERT INTO Employee (Emp_Id, Emp_name, Position, Salary, Branch_no) VALUES
9893
         (101, 'John Doe', 'Manager', 60000, 1),
         (102, 'Jane Smith', 'Manager', 62000, 2),
9894
         (103, 'Bob Johnson', 'Manager', 61000, 3),
9895
9896
         (104, 'Alice Brown', 'Librarian', 45000, 1),
9897
         (105, 'Charlie Davis', 'Librarian', 44000, 2),
         (106, 'Eve Wilson', 'Clerk', 35000, 3);
9898
9899
9900 •
         INSERT INTO Books (ISBN, Book_title, Category, Rental_Price, Status, Author, Publisher) VALUES
         ('ISBN001', 'The Great Novel', 'Fiction', 20, 'yes', 'Author A', 'Publisher X'),
9901
9902
         ('ISBN002', 'History of Europe', 'History', 25, 'yes', 'Author 8', 'Publisher Y'),
         ('ISBN003', 'Science Explained', 'Science', 30, 'no', 'Author C', 'Publisher Z'),
9903
         ('ISBN004', 'The Art Book', 'Art', 35, 'yes', 'Author D', 'Publisher X'),
9904
         ('ISBN005', 'World History', 'History', 28, 'yes', 'Author E', 'Publisher Y');
9905
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'),
         (2, 'Sarah Lee', '654 Maple Ave, City B', '2022-03-20'),
9909
         (3, 'Tom Brown', '987 Oak Rd, City C', '2023-01-10'),
9910
9911
         (4, 'Emily Davis', '246 Pine St, City A', '2021-11-30'),
         (5, 'David Wilson', '135 Cedar Ln, City B', '2022-09-05');
9912
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'),
         (2, 2, 'History of Europe', '2023-06-15', 'ISBN002'),
9916
         (3, 3, 'The Art Book', '2023-07-01', 'ISBN004'),
         (4, 4, 'World History', '2023-06-20', 'ISBN005');
9918
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'),
         (2, 2, 'History of Europe', '2023-06-30', 'ISBN002');
9922
```

## 3. Queries

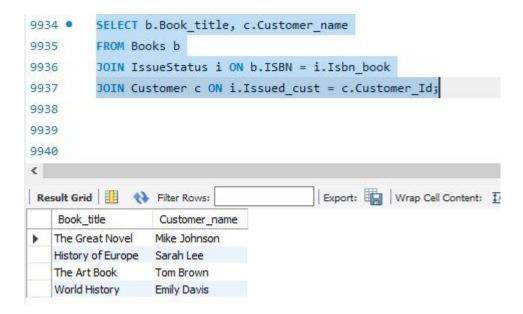
i. Retrieve the book title, category, and rental price of all available books.



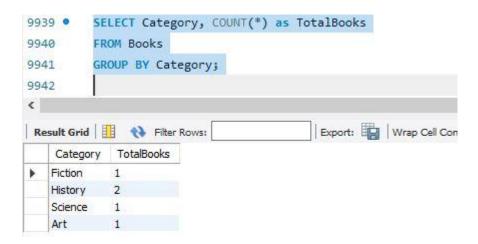
ii.List the employee names and their respective salaries in descending order of salary.



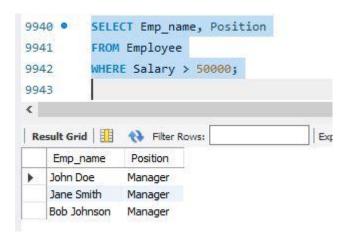
iii.Retrieve the book titles and the corresponding customers who have issued those books.



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



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

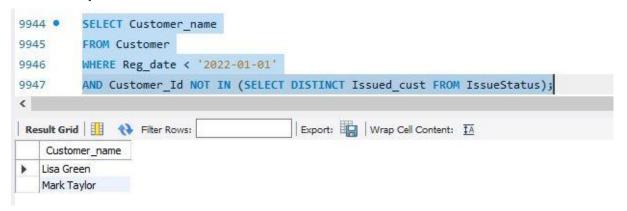


vi . List the customer names who registered before 2022-01-01 and have not issued any books yet.

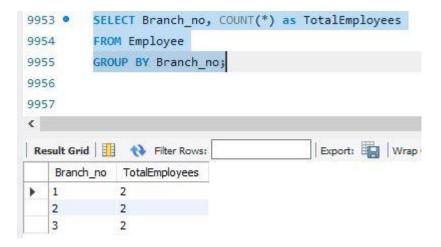
Here the query has no output, so adding additional customer data into the database

```
• 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');
```

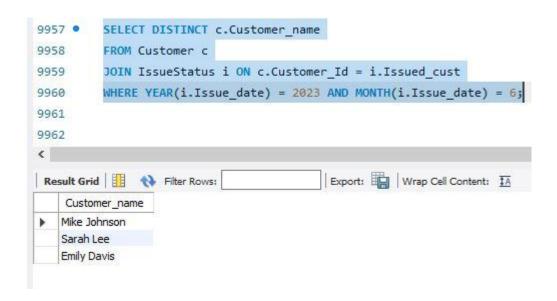
## After update the result is



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



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



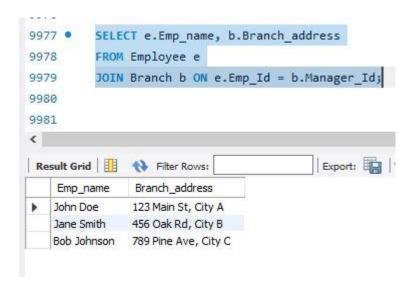
ix.Retrieve book\_title from book table containing history.



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



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



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

```
SELECT DISTINCT c.Customer_name
9978 •
9979
        FROM Customer c
         JOIN IssueStatus i ON c.Customer_Id = i.Issued_cust
9980
         JOIN Books b ON i.Isbn_book = b.ISBN
9981
        WHERE b.Rental_Price > 25;
9982
9983
9984
<
                                      Export: Wrap Cell Conte
Customer_name
   Tom Brown
   Emily Davis
```