Module 2: Library Management System

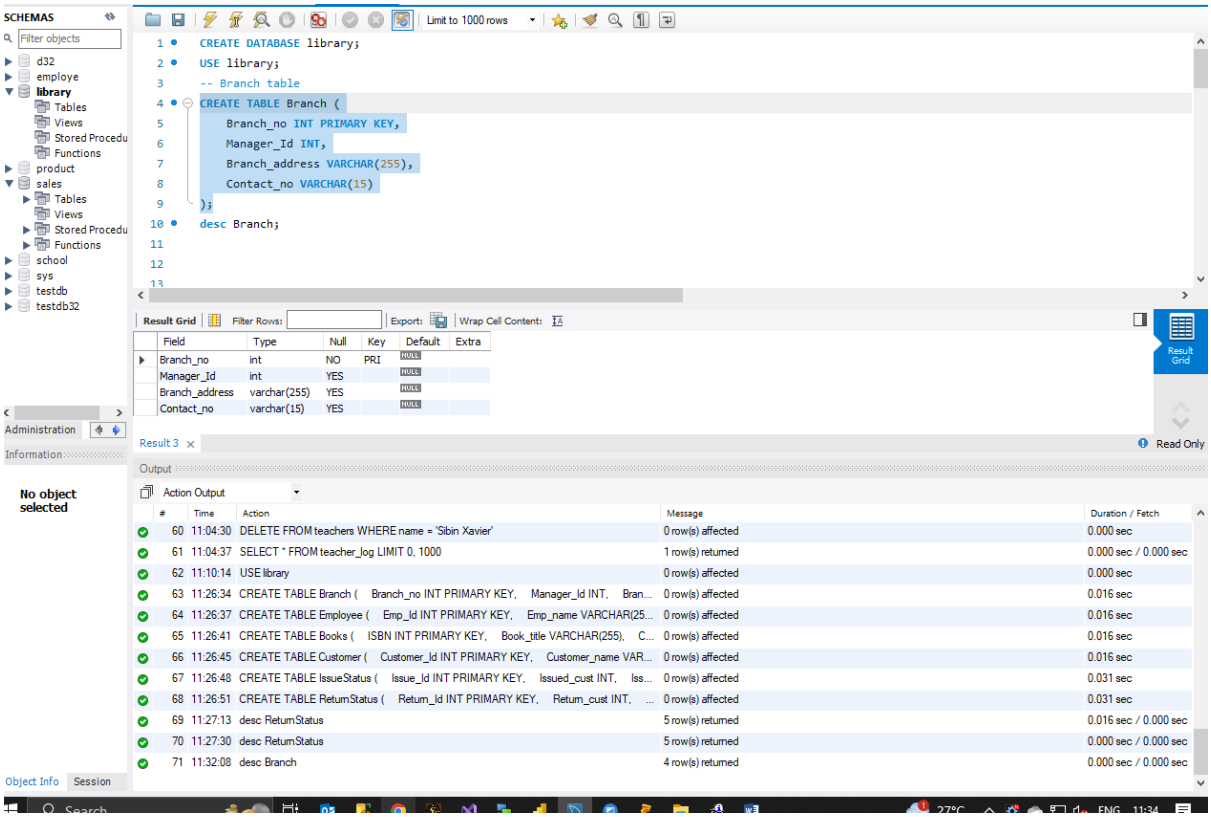
Create a database named library and following TABLES in the database:

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

Attributes for the tables:
1.Branch

a. Branch_no - Set as PRIMARY KEY
b. Manager_Id
c. Branch_address
d. Contact_no

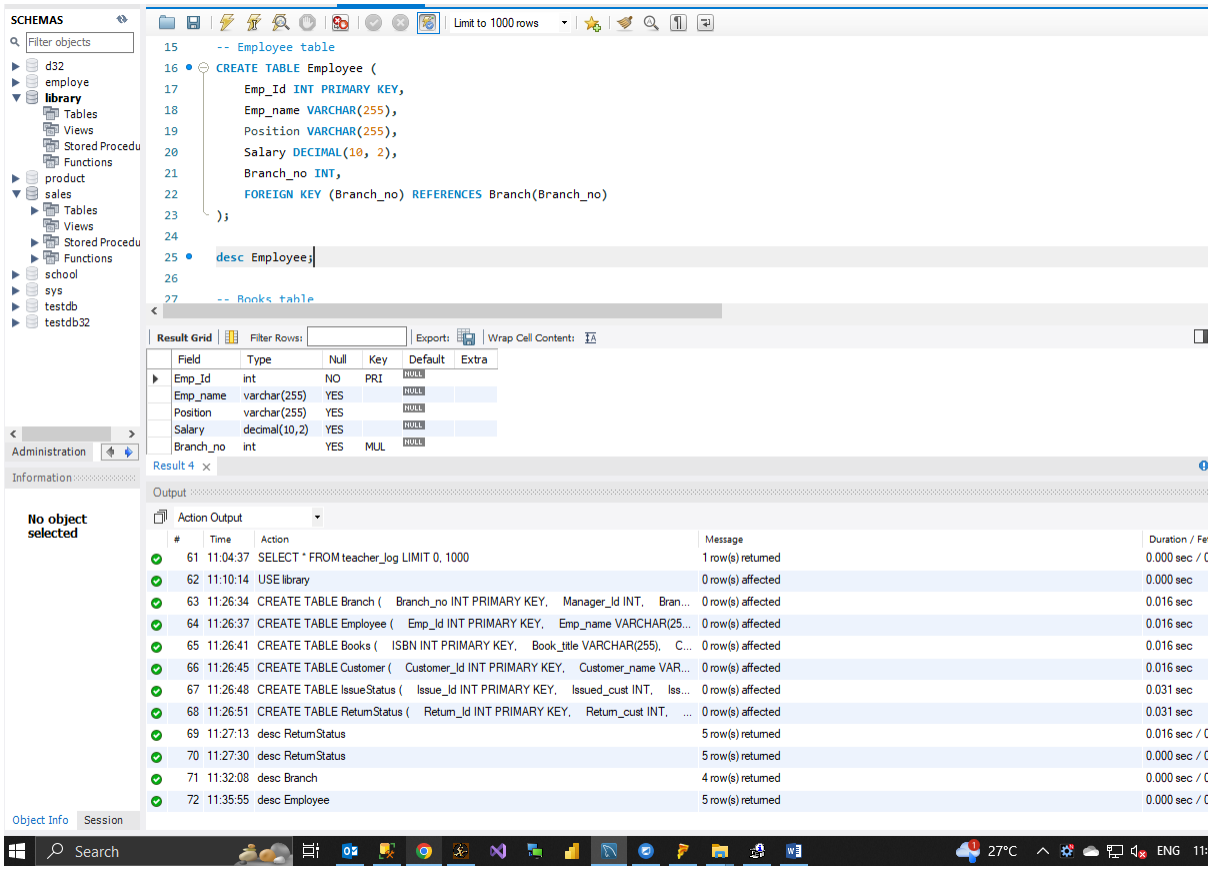
```
CREATE TABLE Branch (
Branch_no INT PRIMARY KEY,
 Manager_Id INT,
Branch_address VARCHAR(255),
Contact_no VARCHAR(15)
);
```



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,
    Emp_name VARCHAR(255),
    Position VARCHAR(255),
    Salary DECIMAL(10, 2),
    Branch_no INT,
    FOREIGN KEY (Branch_no) REFERENCES Branch(Branch_no)
);
```

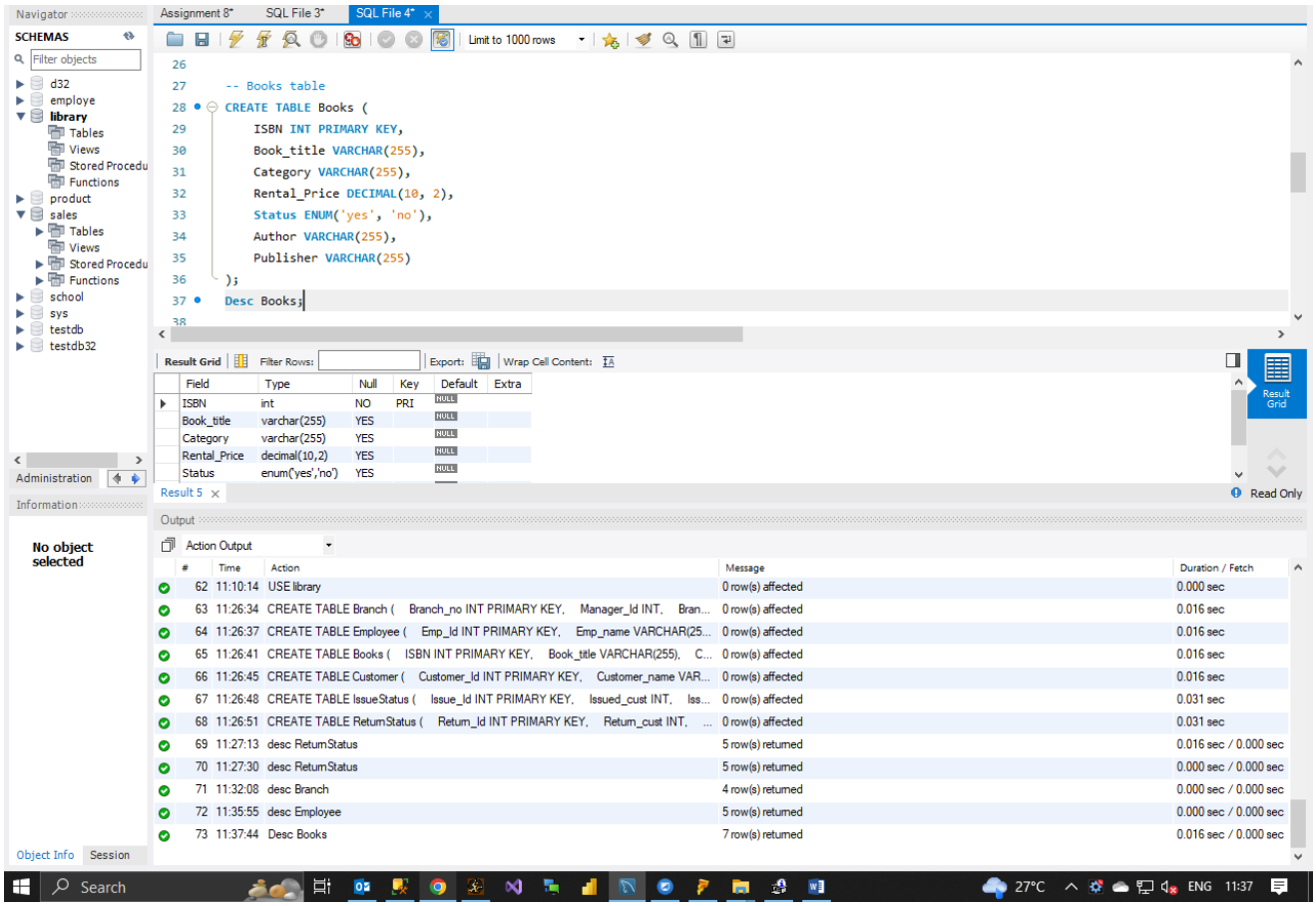


3. Books

- ISBN - Set as PRIMARY KEY
- Book_title
- Category

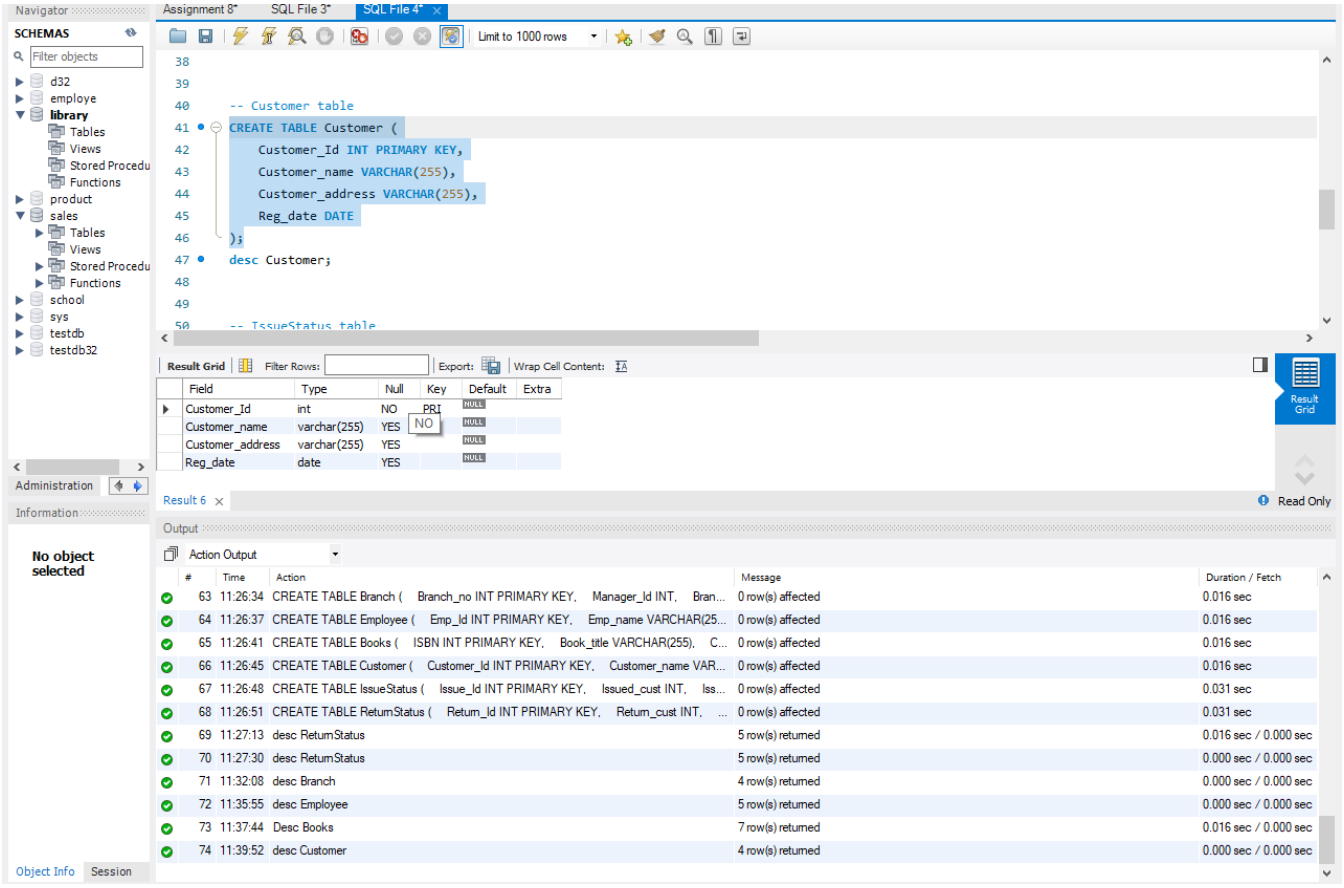
- Rental_Price
- Status [Give yes if book available and no if book not available]
- Author
- Publisher

```
CREATE TABLE Books (
    ISBN INT PRIMARY KEY,
    Book_title VARCHAR(255),
    Category VARCHAR(255),
    Rental_Price DECIMAL(10, 2),
    Status ENUM('yes', 'no'),
    Author VARCHAR(255),
    Publisher VARCHAR(255)
);
```



4. **Customer**
    - **a.** Customer_Id - Set as PRIMARY KEY
    - b. Customer_name
    - c. Customer_address
    - d. Reg_date

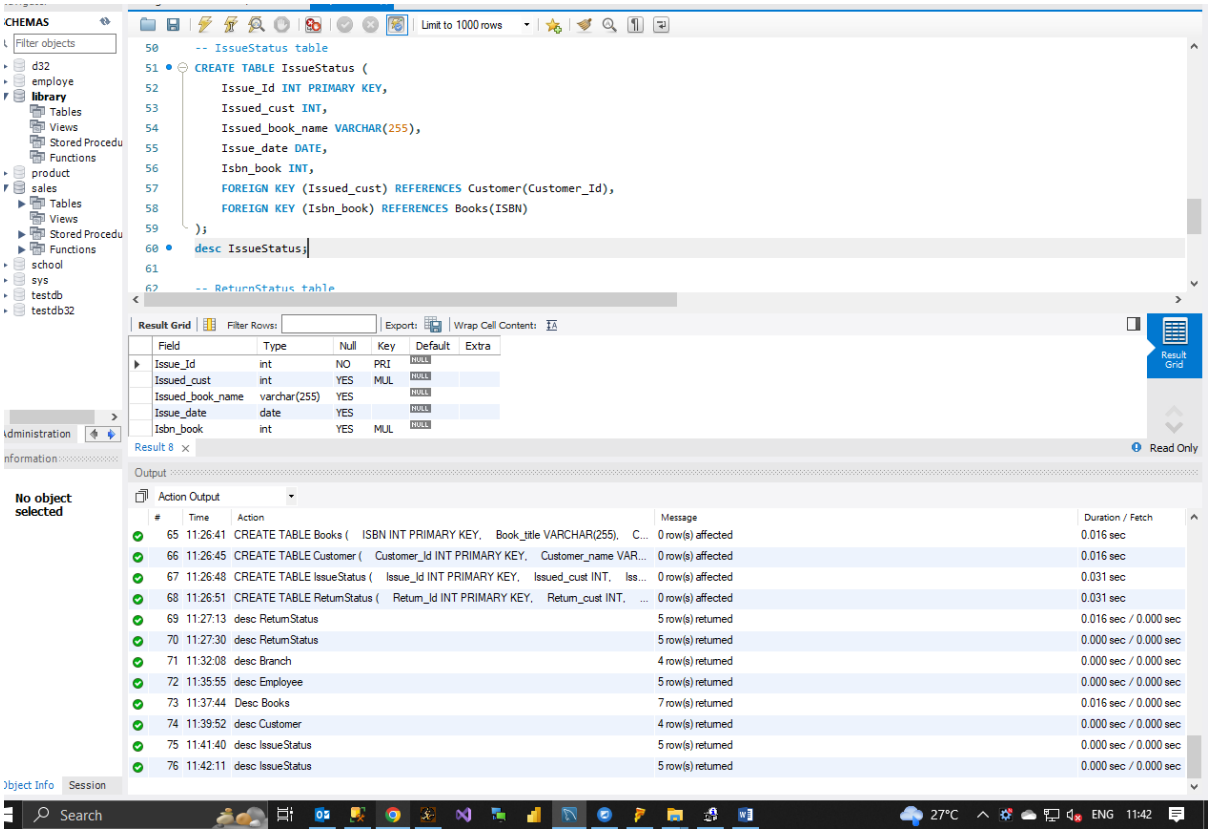
```
CREATE TABLE Customer (
    Customer_Id INT PRIMARY KEY,
    Customer_name VARCHAR(255),
    Customer_address VARCHAR(255),
    Reg_date DATE
);
```



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

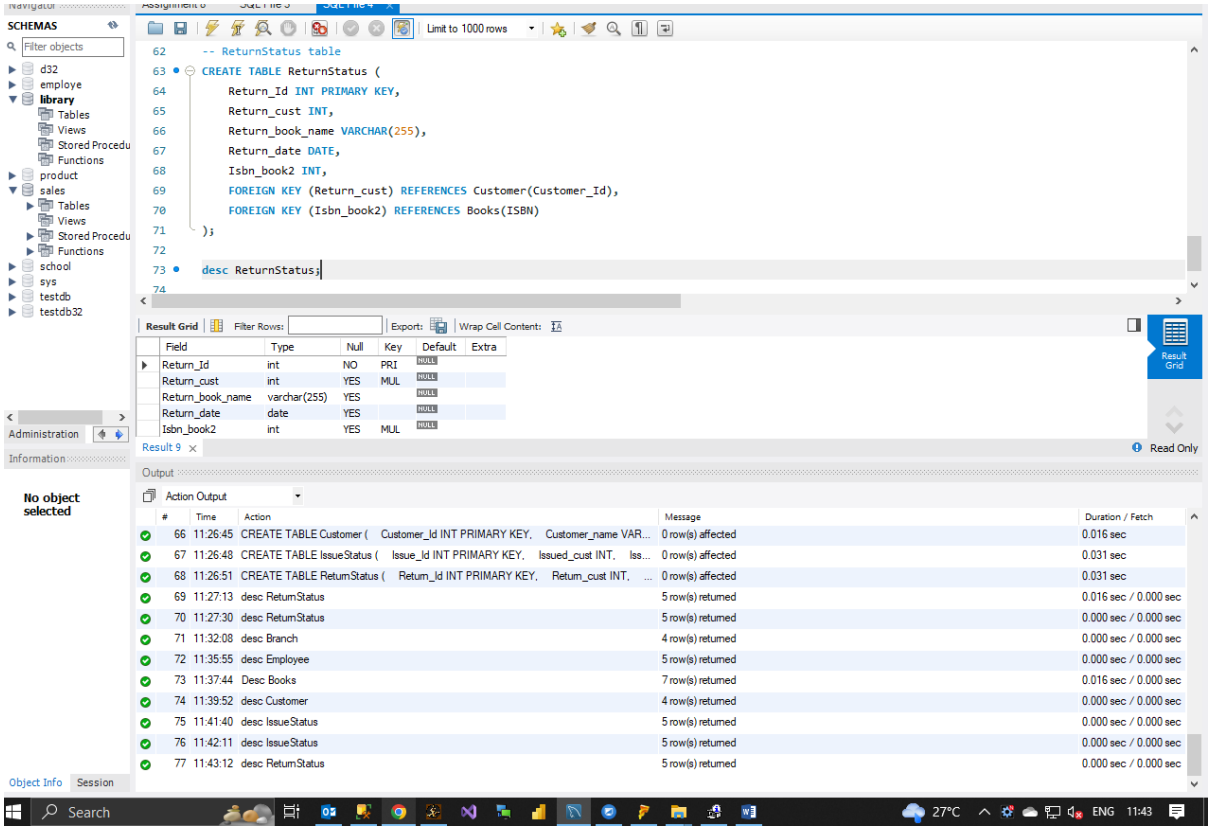
```
CREATE TABLE IssueStatus (
    Issue_Id INT PRIMARY KEY,
    Issued_cust INT,
    Issued_book_name VARCHAR(255),
    Issue_date DATE,
    Isbn_book INT,
    FOREIGN KEY (Issued_cust) REFERENCES Customer(Customer_Id),
    FOREIGN KEY (Isbn_book) REFERENCES Books(ISBN));
```



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

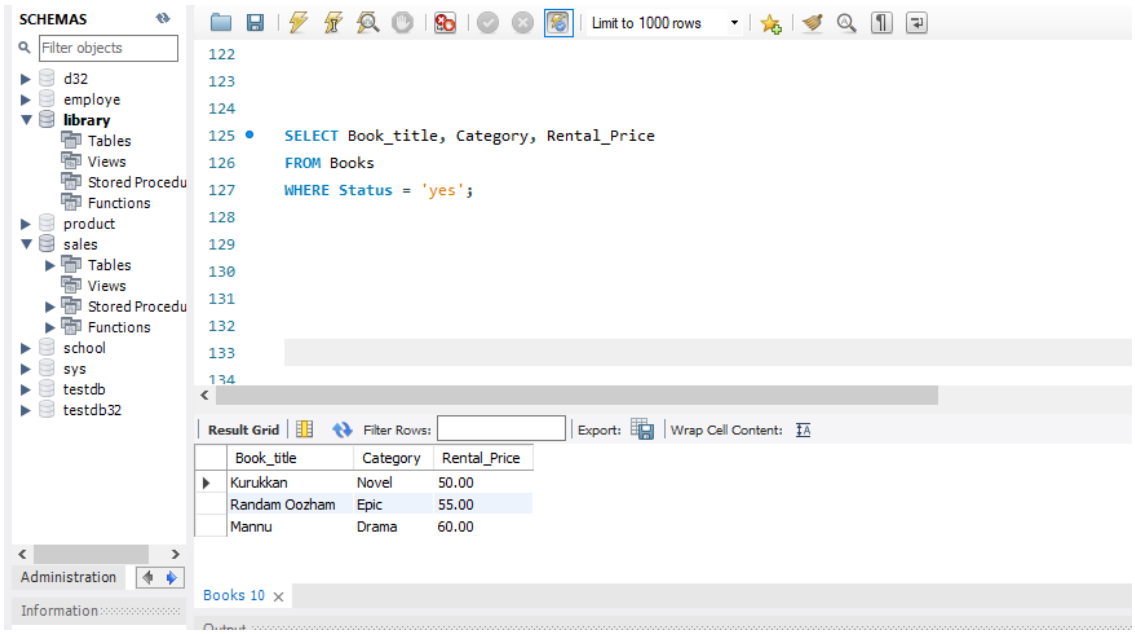
```
CREATE TABLE ReturnStatus (
    Return_Id INT PRIMARY KEY,
    Return_cust INT,
    Return_book_name VARCHAR(255),
    Return_date DATE,
    Isbn_book2 INT,
    FOREIGN KEY (Return_cust) REFERENCES Customer(Customer_Id),
    FOREIGN KEY (Isbn_book2) REFERENCES Books(ISBN)
);
```



Display all the tables and Write the queries for the following :

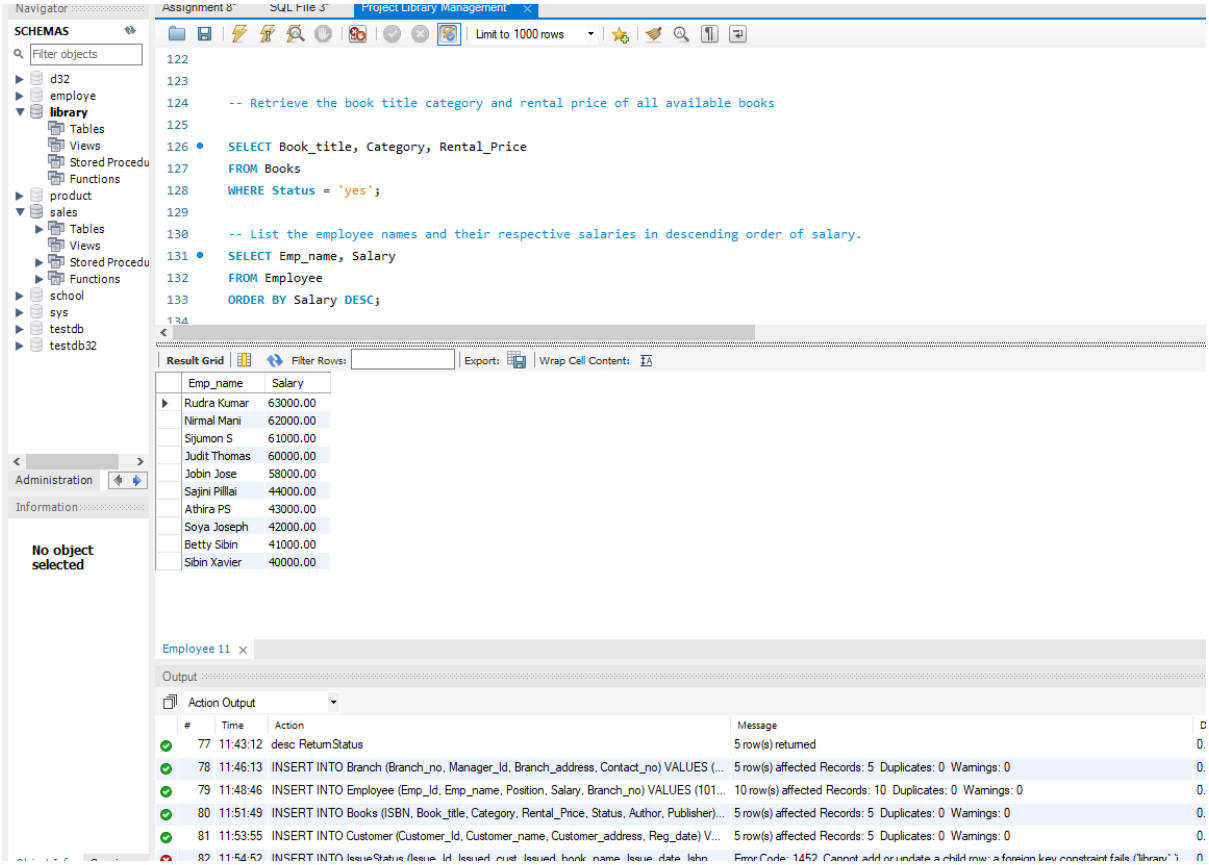
1. Retrieve the book title, category, and rental price of all available books.

```
SELECT Book_title, Category, Rental_Price
FROM Books
WHERE Status = 'yes';
```



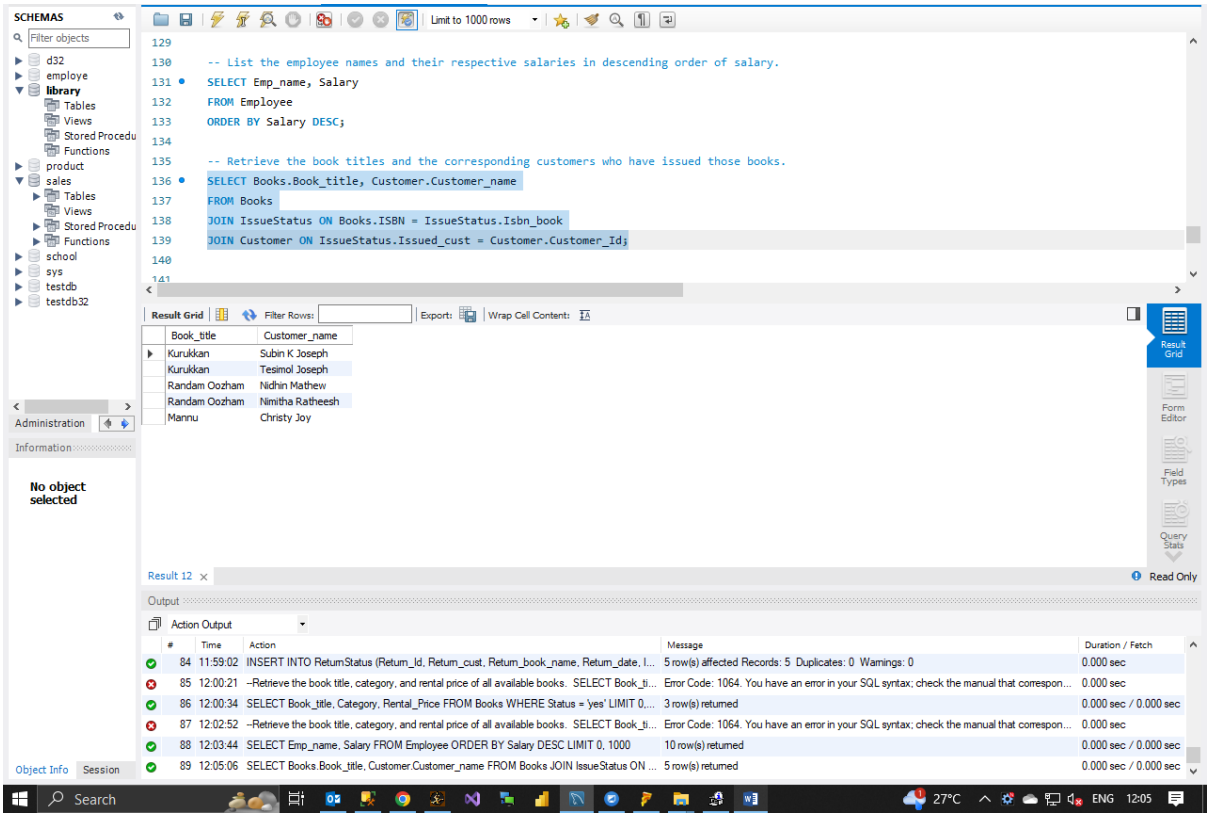
2. List the employee names and their respective salaries in descending order of salary.

```
SELECT Emp_name, Salary
FROM Employee
ORDER BY Salary DESC;
```



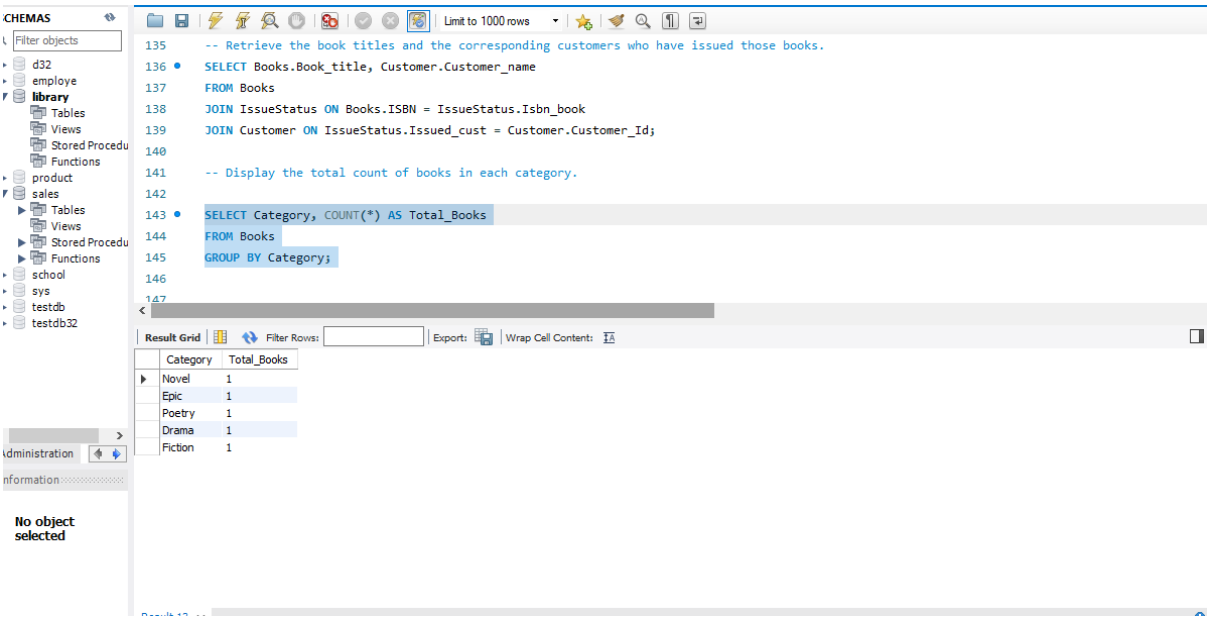
3. Retrieve the book titles and the corresponding customers who have issued those books.

```
SELECT Books.Book_title, Customer.Customer_name
FROM Books
JOIN IssueStatus ON Books.ISBN = IssueStatus.Isbn_book
JOIN Customer ON IssueStatus.Issued_cust = Customer.Customer_Id;
```



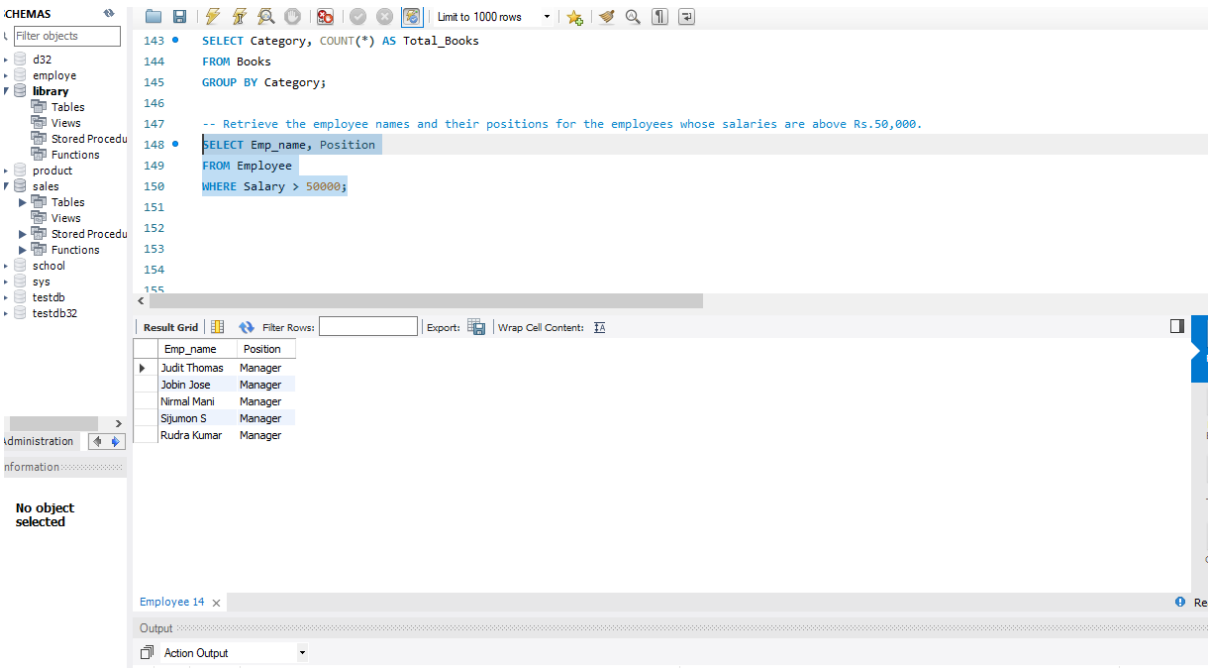
4. Display the total count of books in each category.

```
SELECT Category, COUNT(*) AS Total_Books
FROM Books
GROUP BY Category;
```



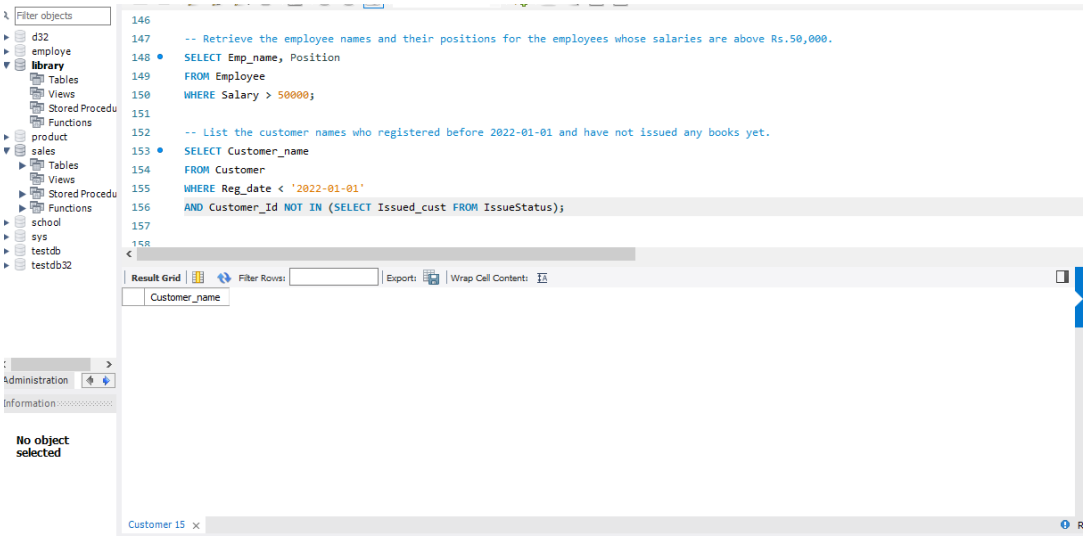
5. Retrieve the employee names and their positions for the employees whose salaries are above Rs.50,000.

```
SELECT Emp_name, Position
FROM Employee
WHERE Salary > 50000;
```



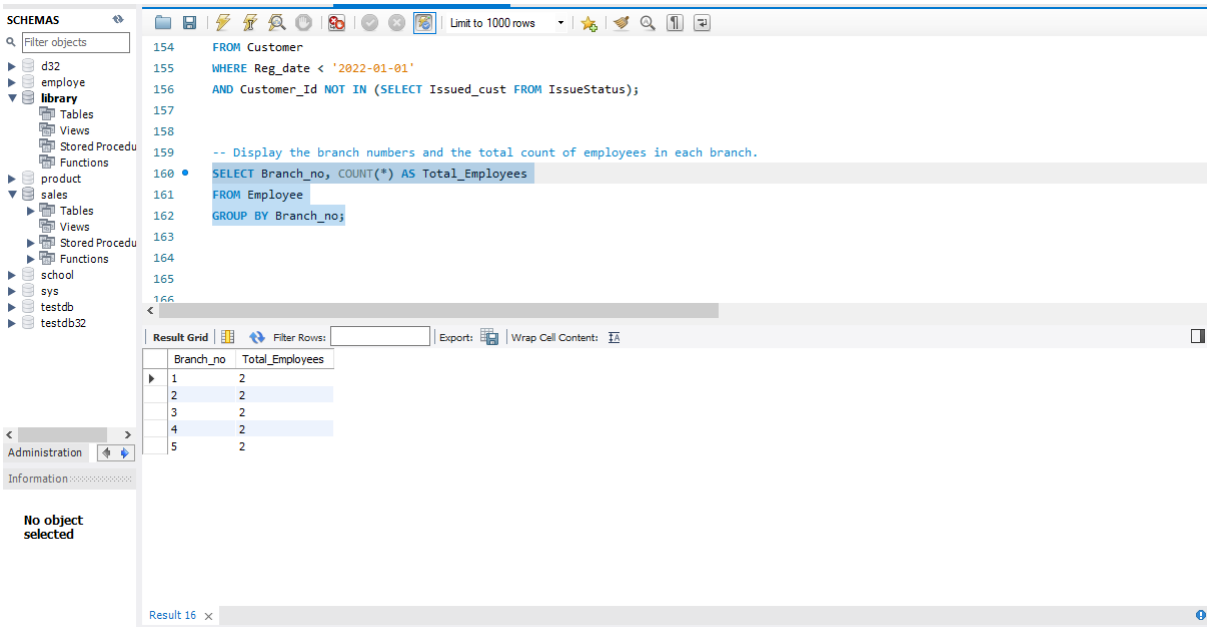
6. List the customer names who registered before 2022-01-01 and have not issued any books yet.

```
SELECT Customer_name
FROM Customer
WHERE Reg_date < '2022-01-01'
AND Customer_Id NOT IN (SELECT Issued_cust FROM IssueStatus);
```



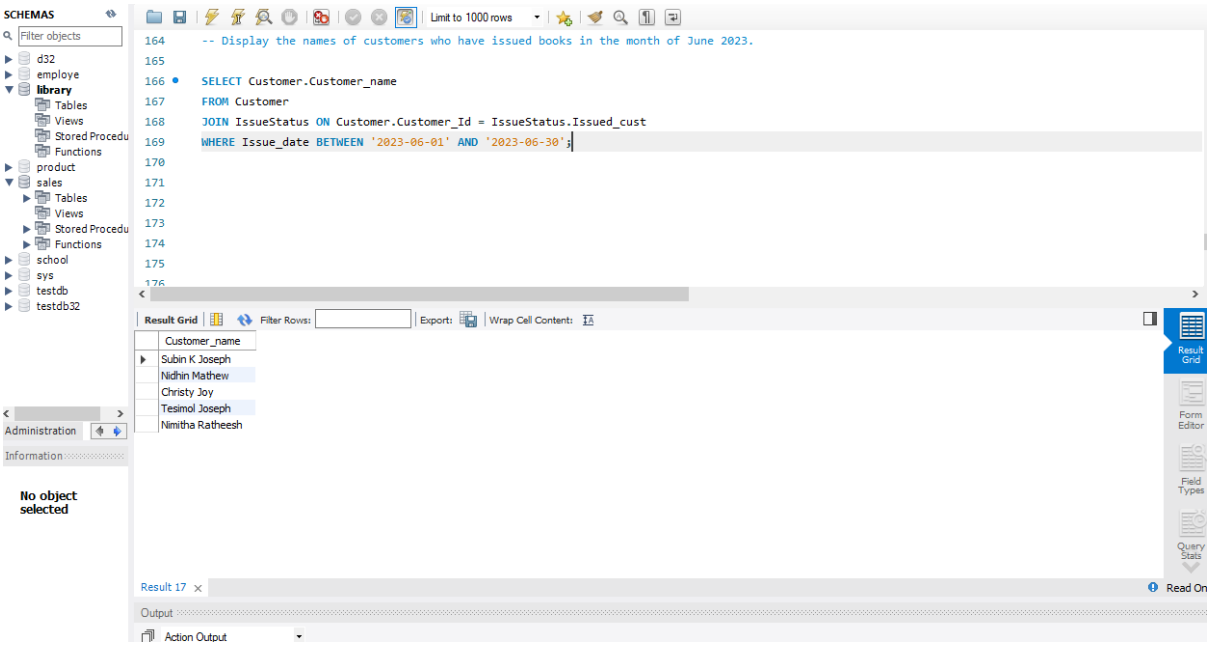
7. Display the branch numbers and the total count of employees in each branch.

```
SELECT Branch_no, COUNT(*) AS Total_Employees
FROM Employee
GROUP BY Branch_no;
```



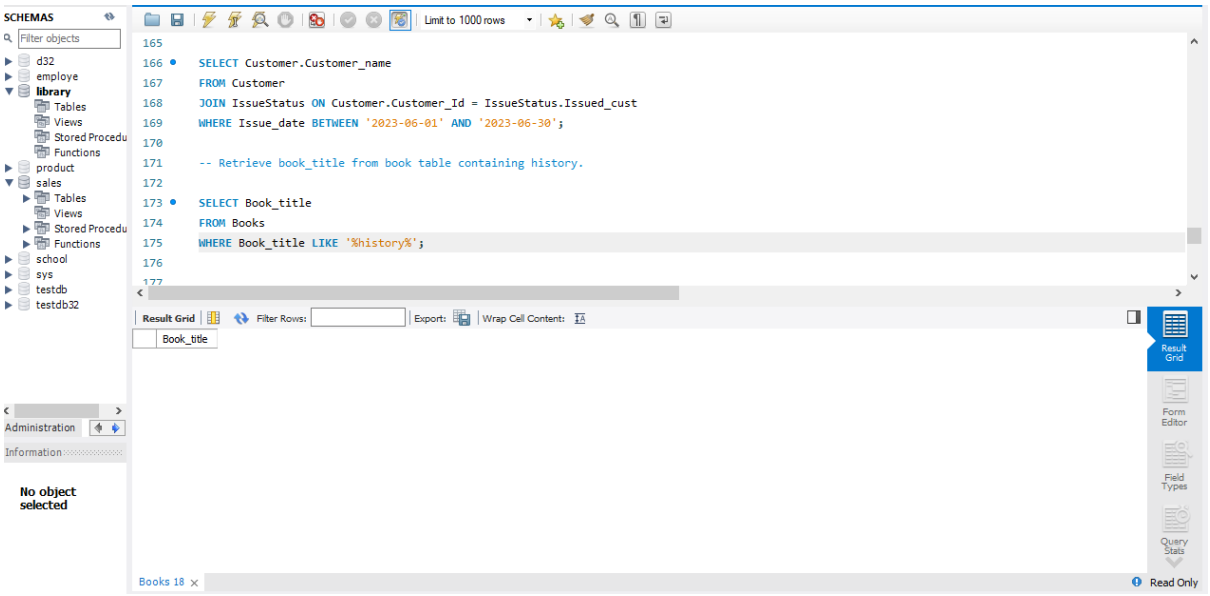
8. Display the names of customers who have issued books in the month of June 2023.

```
SELECT Customer.Customer_name
FROM Customer
JOIN IssueStatus ON Customer.Customer_Id = IssueStatus.Issued_cust
WHERE Issue_date BETWEEN '2023-06-01' AND '2023-06-30';
```



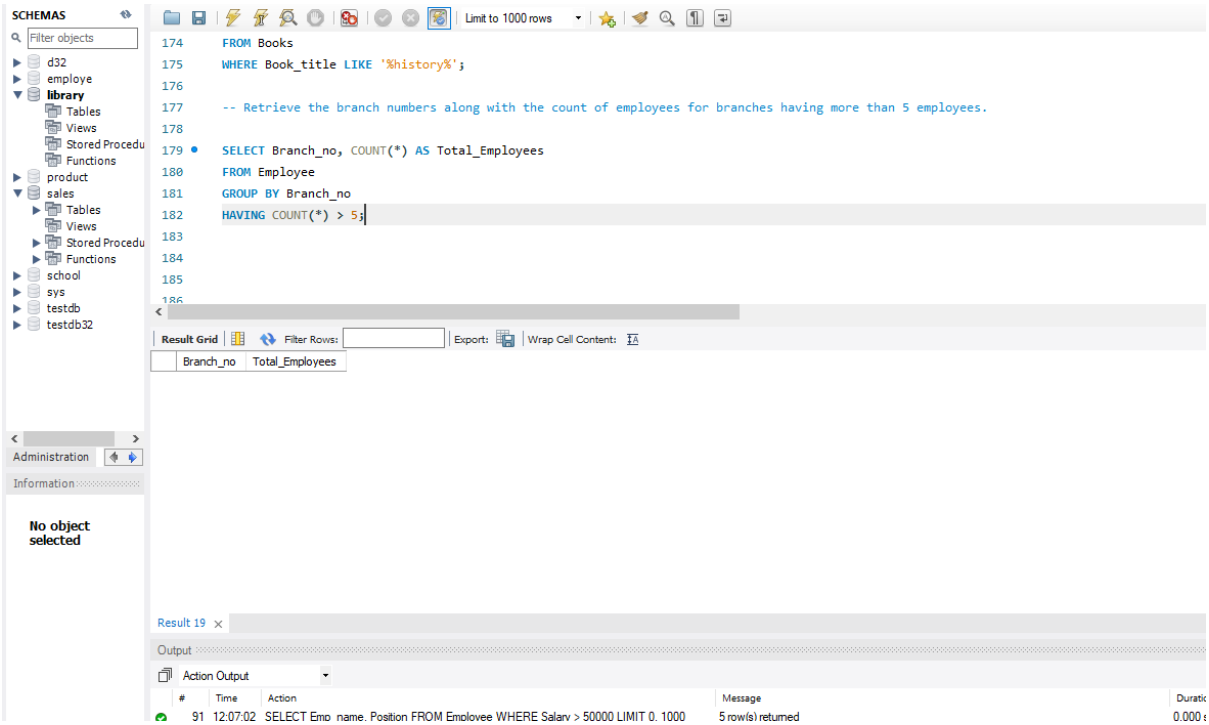
9. Retrieve book_title from book table containing history.

```
SELECT Book_title
FROM Books
WHERE Book_title LIKE '%history%';
```



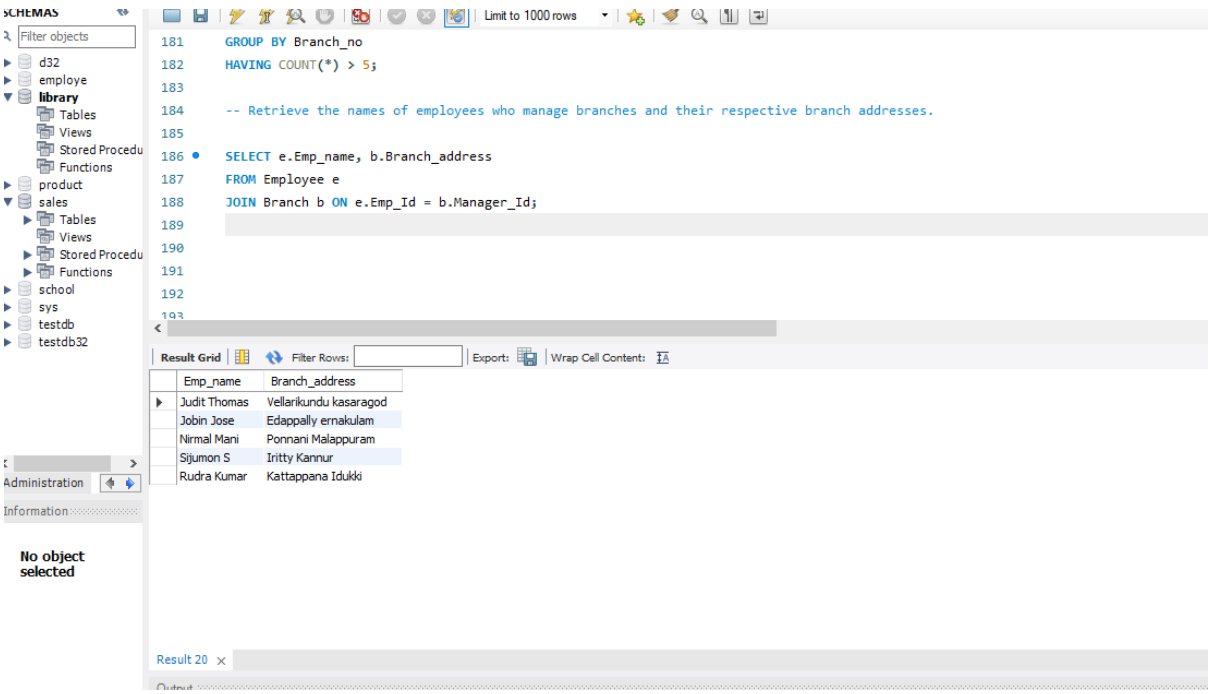
10. Retrieve the branch numbers along with the count of employees for branches having more than 5 employees.

```
SELECT Branch_no, COUNT(*) AS Total_Employees
FROM Employee
GROUP BY Branch_no
HAVING COUNT(*) > 5;
```



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

```
SELECT e.Emp_name, b.Branch_address
FROM Employee e
JOIN Branch b ON e.Emp_Id = b.Manager_Id;
```



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

```
SELECT DISTINCT Customer.Customer_name
FROM Customer
JOIN IssueStatus ON Customer.Customer_Id = IssueStatus.Issued_cust
JOIN Books ON IssueStatus.Isbn_book = Books.ISBN
WHERE Books.Rental_Price > 25;
```