MYSQL PROJECT

Topic: 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

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

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

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

4. Customer

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

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

Queries

```
CREATE DATABASE library;

USE library;
```

Table 1

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

select* from Branch;
```

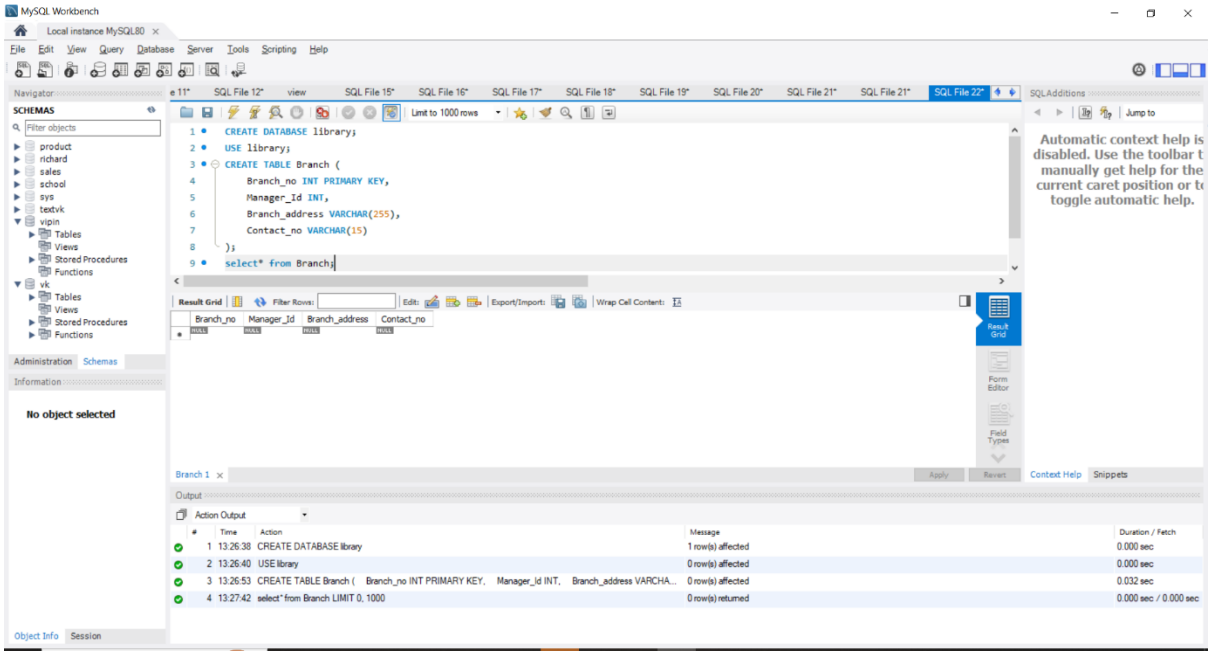


Table 2

```
CREATE TABLE Employee (
    Emp_Id INT PRIMARY KEY,
    Emp_name VARCHAR(100),
    Position VARCHAR(50),
    Salary DECIMAL(10, 2),
    Branch_no INT,
    FOREIGN KEY (Branch_no) REFERENCES Branch(Branch_no)
);
select* from Employee;
```

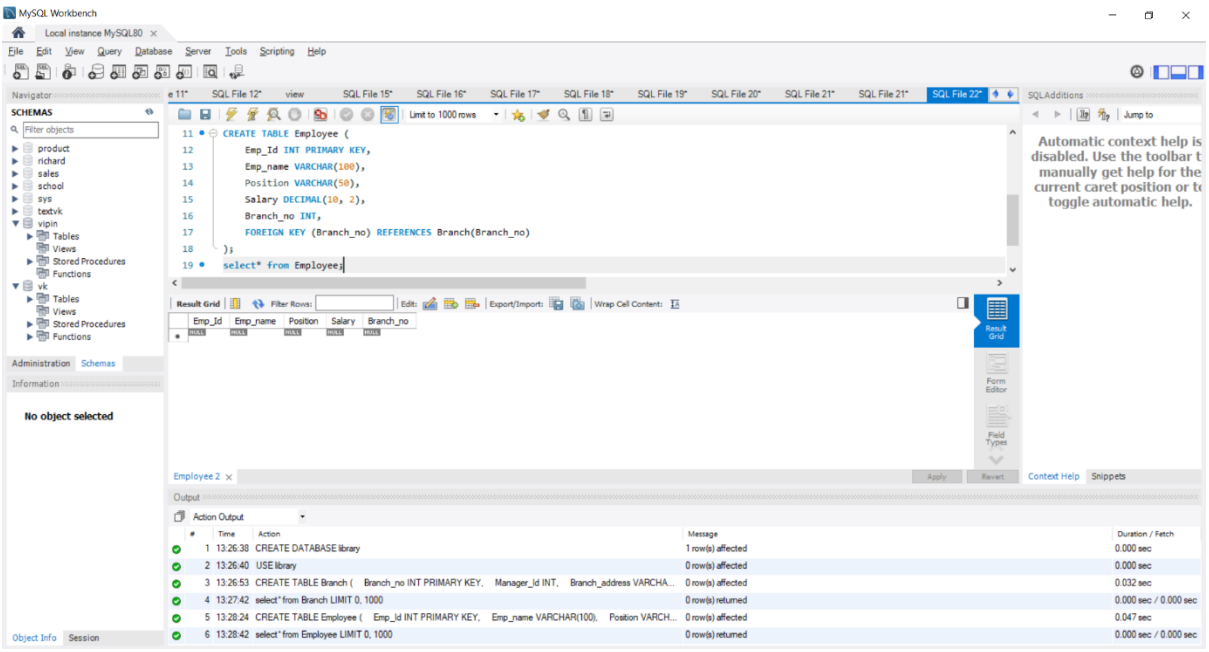


Table 3

```
CREATE TABLE Books (
    ISBN VARCHAR(13) PRIMARY KEY,
    Book_title VARCHAR(255),
    Category VARCHAR(50),
    Rental_Price DECIMAL(10, 2),
    Status ENUM('yes', 'no'),
    Author VARCHAR(100),
    Publisher VARCHAR(100)
);
select* from Books;
```

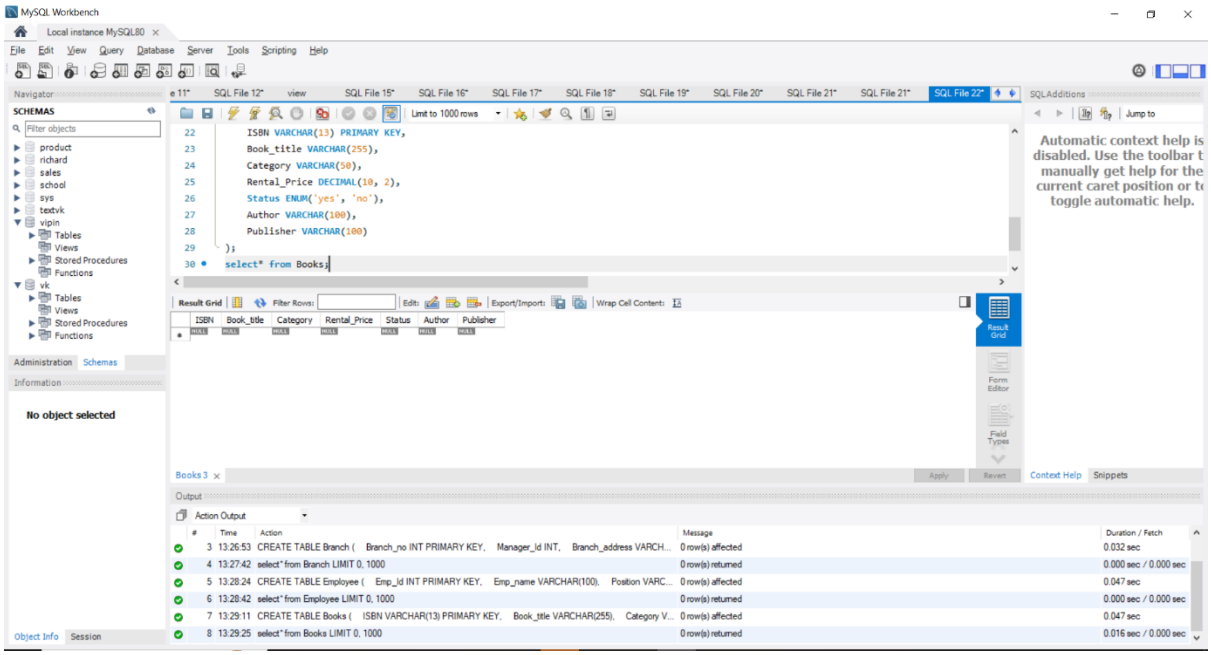


Table 4

```
CREATE TABLE Customer (
   Customer_Id INT PRIMARY KEY,
   Customer_name VARCHAR(100),
   Customer_address VARCHAR(255),
   Reg_date DATE
);
select* from Customer;
```

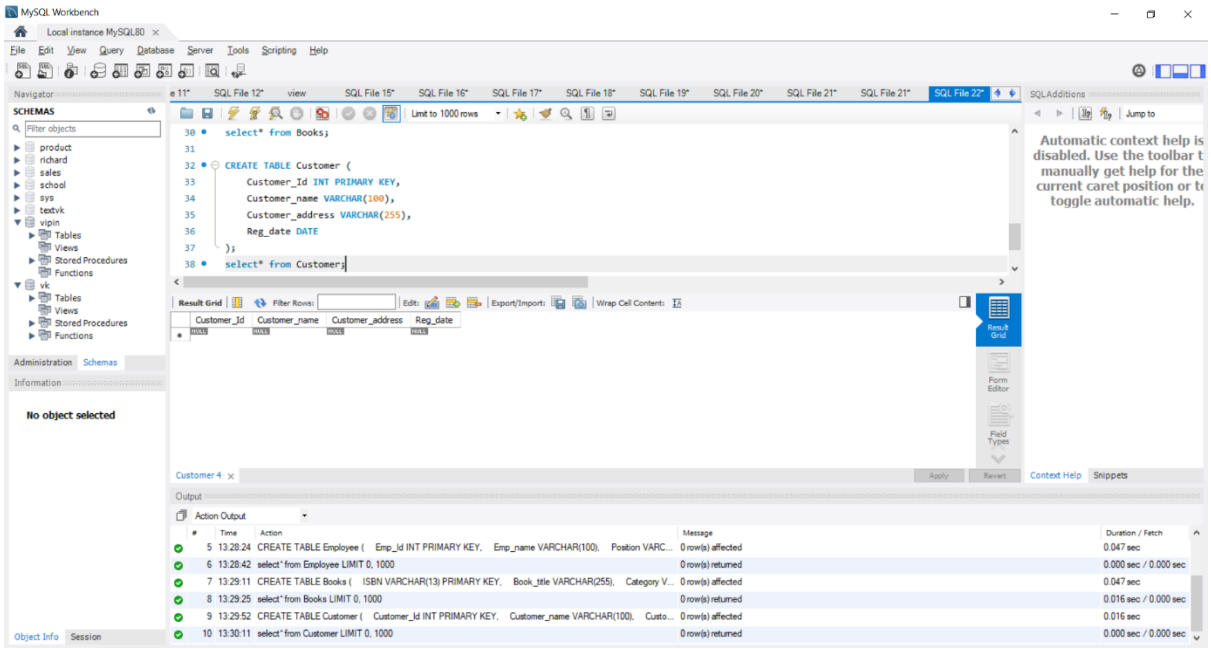


Table 5

```
CREATE TABLE IssueStatus (
    Issue_Id INT PRIMARY KEY,
    Issued_cust INT,
    Issued_book_name VARCHAR(255),
    Issue_date DATE,
    Isbn_book VARCHAR(13),
    FOREIGN KEY (Issued_cust) REFERENCES Customer(Customer_Id),
    FOREIGN KEY (Isbn_book) REFERENCES Books(ISBN)
);
select* from IssueStatus;
```

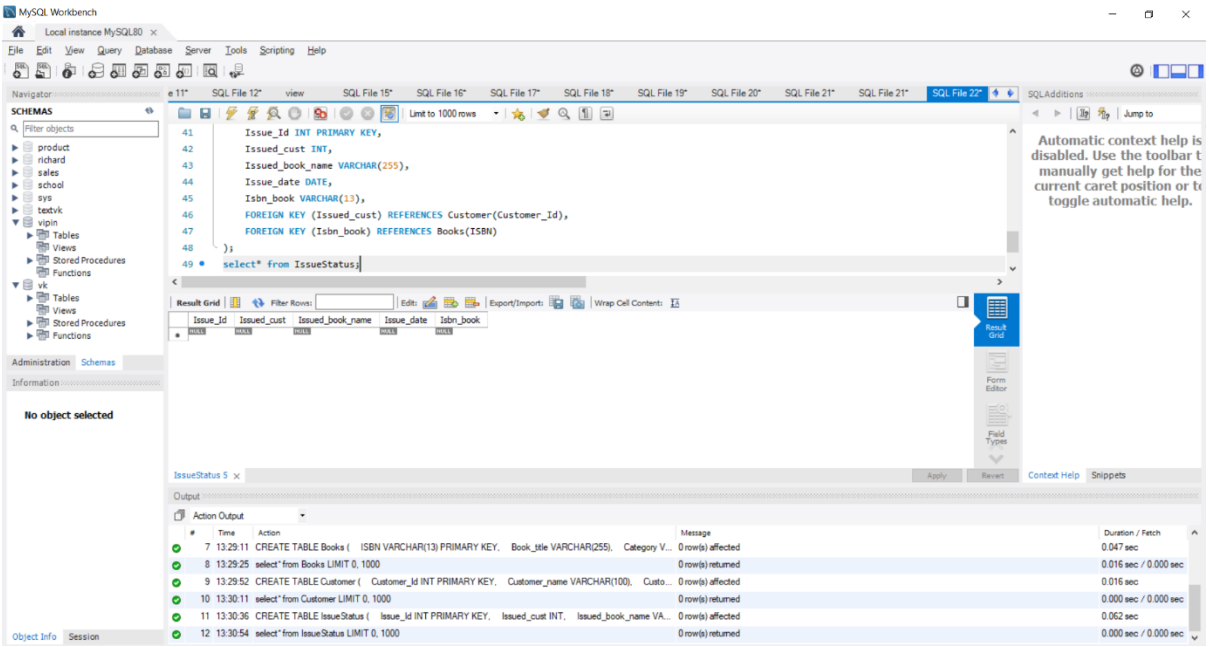
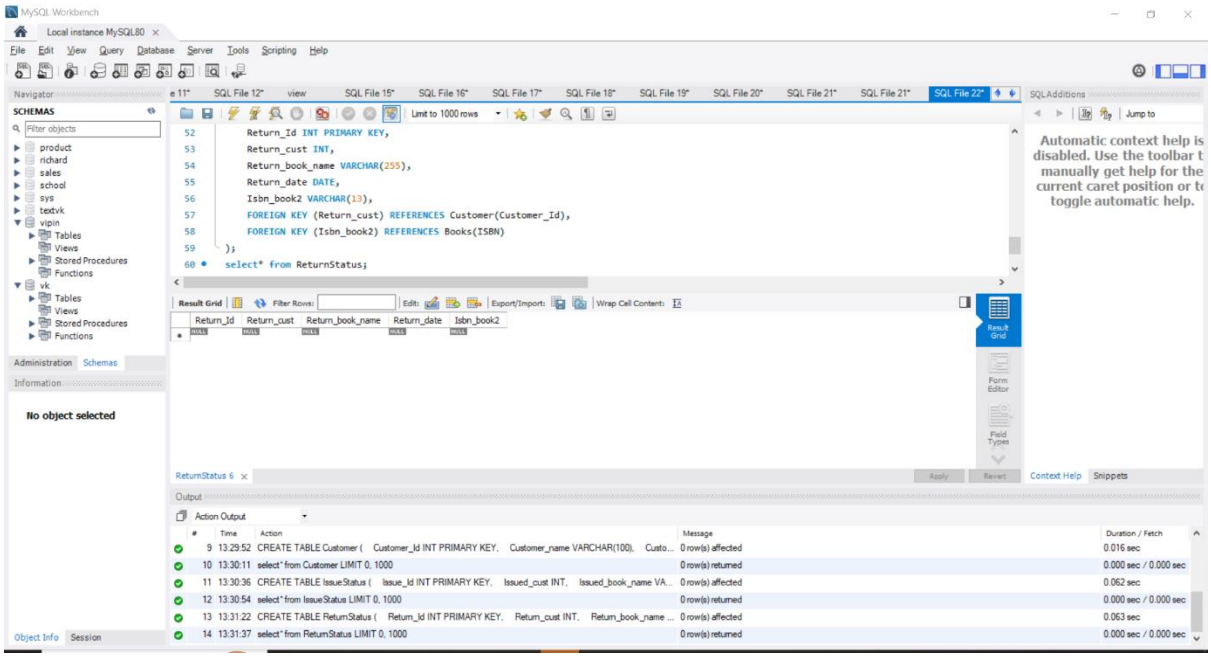


Table 6

```
CREATE TABLE ReturnStatus (
    Return_Id INT PRIMARY KEY,
    Return_cust INT,
    Return_book_name VARCHAR(255),
    Return_date DATE,
    Isbn_book2 VARCHAR(13),
    FOREIGN KEY (Return_cust) REFERENCES Customer(Customer_Id),
    FOREIGN KEY (Isbn_book2) REFERENCES Books(ISBN)
);
select* from ReturnStatus;
```

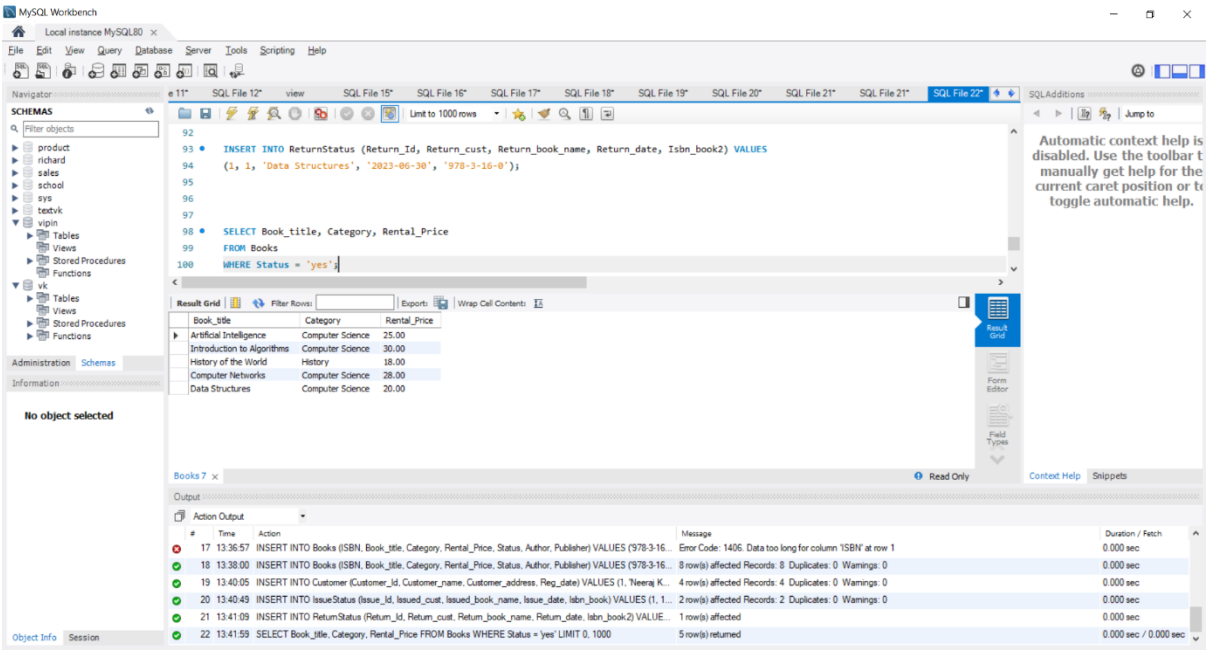


Inserting Values in the tables

```
INSERT INTO Branch (Branch_no, Manager_Id, Branch_address, Contact_no) VALUES
(1, 101, '36 Anna Nagar', '1234567890'),
(2, 102, '456 Thiruvananthapuram', '9876543210');

INSERT INTO Employee (Emp_Id, Emp_name, Position, Salary, Branch_no) VALUES
(1, 'Thomas', 'Manager', 60000.00, 1),
(2, 'Arun Varma', 'Assistant Manager', 45000.00, 1),
(3, 'Yesudas', 'Clerk', 35000.00, 2),
(4, 'Alexander', 'Clerk', 32000.00, 2),
(5, 'Grace', 'Manager', 62000.00, 2);

INSERT INTO Books (ISBN, Book_title, Category, Rental_Price, Status, Author, Publisher) VALUES
('978-3-16-0', 'Data Structures', 'Computer Science', 20.00, 'yes', 'Seymour Lipschutz', 'McGraw-Hill'),
('978-0-13-7', 'C Programming Language', 'Computer Science', 15.00, 'no', 'Brian W. Kernighan', 'Prentice Hall'),
('978-0-07-6', 'Artificial Intelligence', 'Computer Science', 25.00, 'yes', 'Stuart Russell', 'Pearson'),
('978-0-262-8', 'Introduction to Algorithms', 'Computer Science', 30.00, 'yes', 'Thomas H. Cormen', 'MIT Press'),
('978-1-4088-1', 'History of the World', 'History', 18.00, 'yes', 'J. M. Roberts', 'Penguin Books'),
('978-0-521-2', 'A Brief History of Time', 'Science', 22.00, 'no', 'Stephen Hawking', 'Bantam Dell'),
('978-1-56619-4', 'Computer Networks', 'Computer Science', 28.00, 'yes', 'Andrew S. Tanenbaum', 'Pearson'),
('978-0-7432-5', 'The Art of War', 'History', 15.00, 'no', 'Sun Tzu', 'Shambhala');

INSERT INTO Customer (Customer_Id, Customer_name, Customer_address, Reg_date) VALUES
(1, 'Neeraj Kumar', 'Thirumangalam', '2021-12-15'),
(2, 'Arul Das', 'Kattakada', '2022-01-10'),
(3, 'Sandeep', 'Kattapana', '2023-03-05'),
(4, 'Sunitha', 'Ashok Nagar', '2024-06-20');

INSERT INTO IssueStatus (Issue_Id, Issued_cust, Issued_book_name, Issue_date, Isbn_book) VALUES
(1, 1, 'Data Structures', '2023-06-15', '978-3-16-0'),
(2, 2, 'Introduction to Algorithms', '2023-06-20', '978-0-262-8');

INSERT INTO ReturnStatus (Return_Id, Return_cust, Return_book_name, Return_date, Isbn_book2) VALUES
(1, 1, 'Data Structures', '2023-06-30', '978-3-16-0');
```

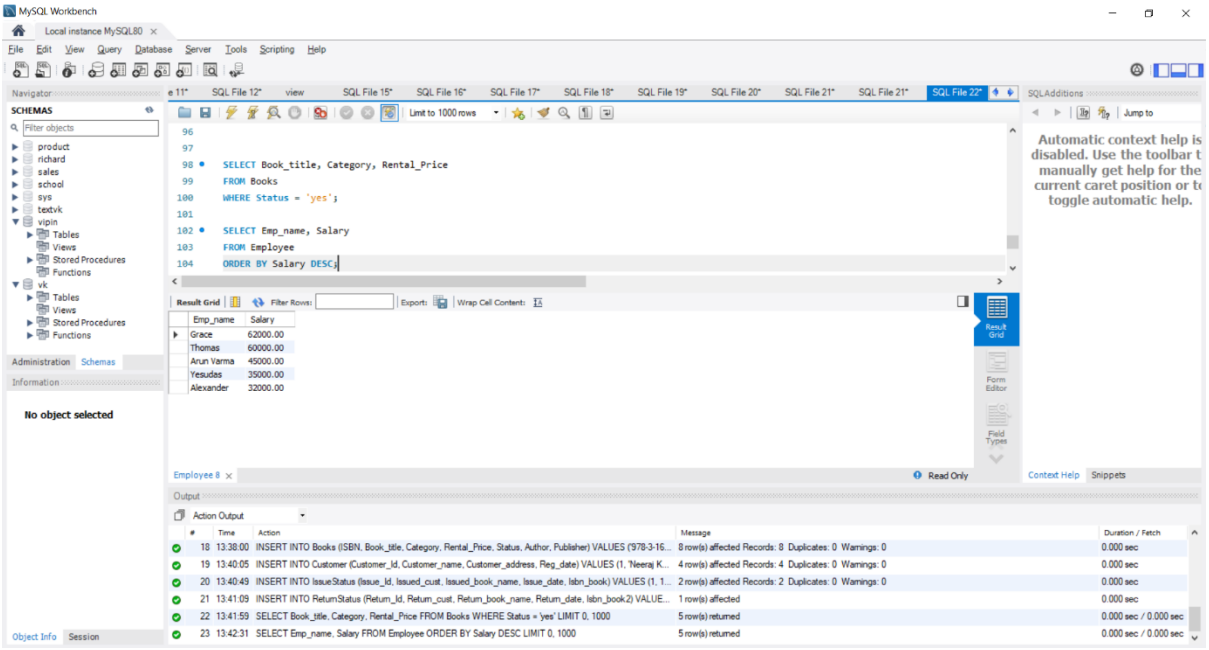
1. Retrieve the book title, category, and rental price of all available books.

```
SELECT Book_title, Category, Rental_Price

FROM Books

WHERE Status = 'yes';
```



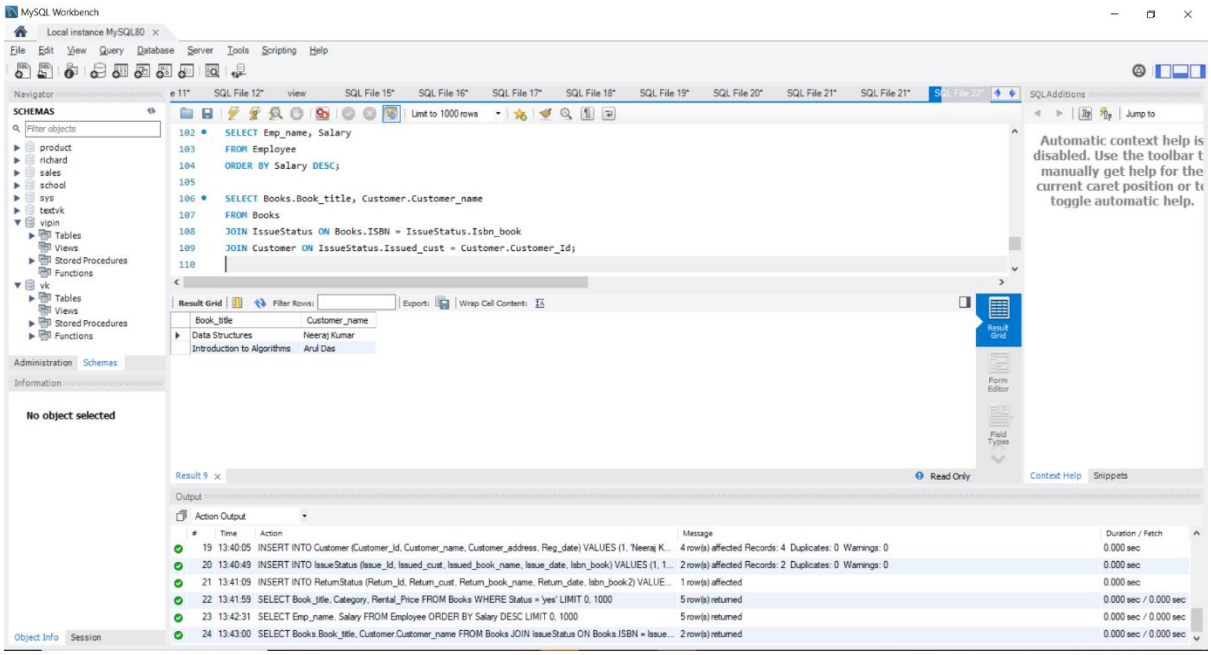
2. List the employee names and their respective salaries in descending order of salary.

```
SELECT Emp_name, Salary

FROM Employee

ORDER BY Salary DESC;
```



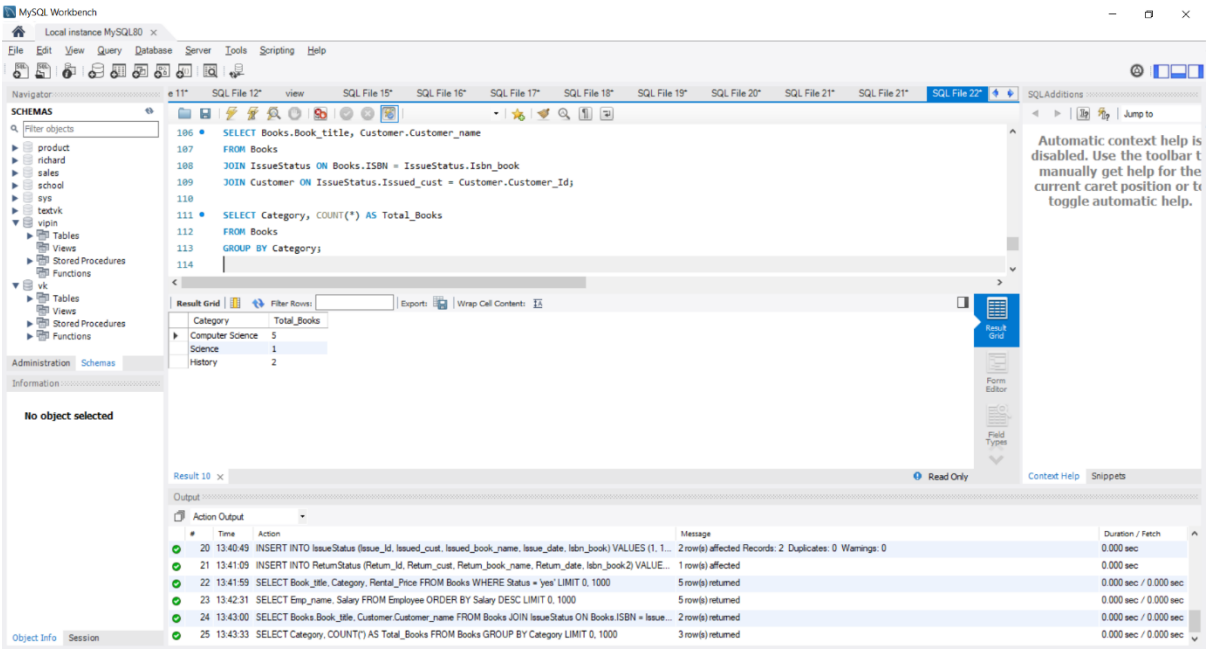
3. Retrieve the book titles and the corresponding customers who have issued those books.

```
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;
```



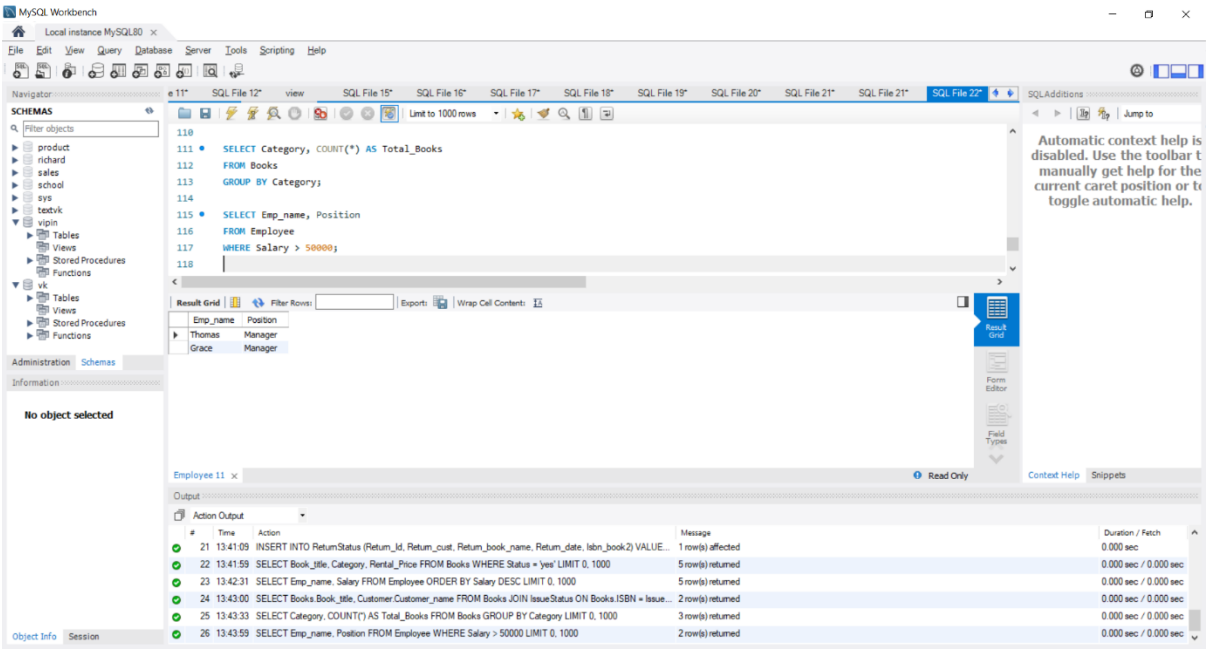
4. Display the total count of books in each category.

```
SELECT Category, COUNT(*) AS Total_Books

FROM Books

GROUP BY Category;
```



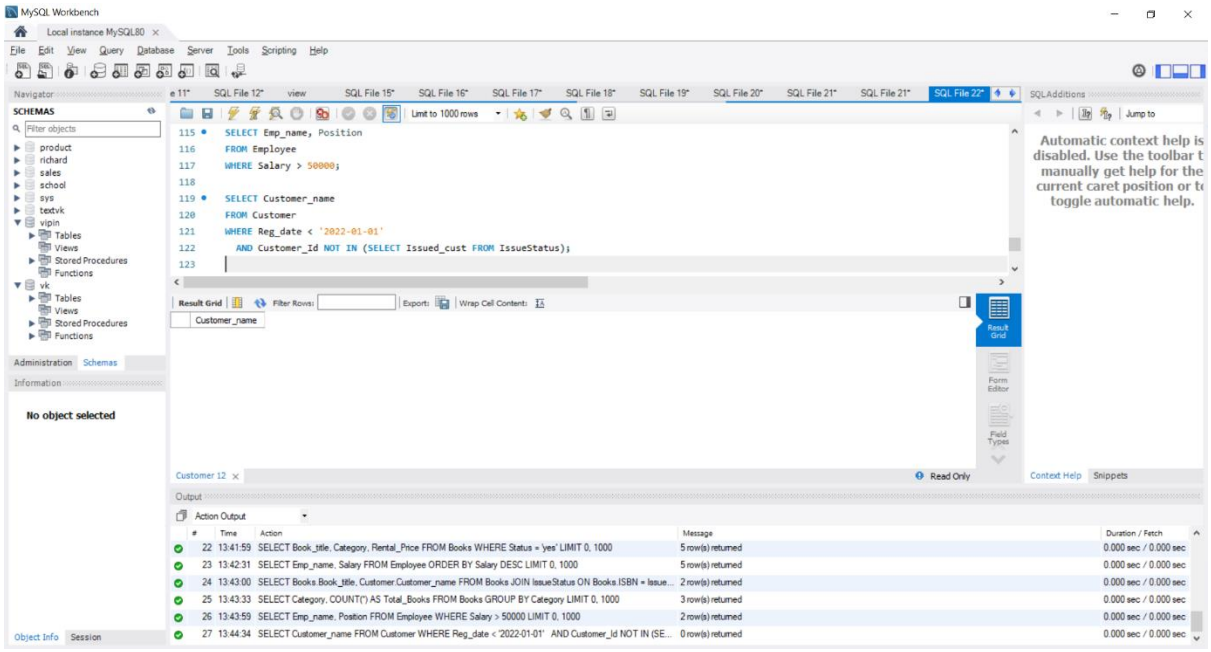
5. Retrieve the employee names and their positions for the employees whose salaries are above Rs.50,000.

```
SELECT Emp_name, Position

FROM Employee

WHERE Salary > 50000;
```



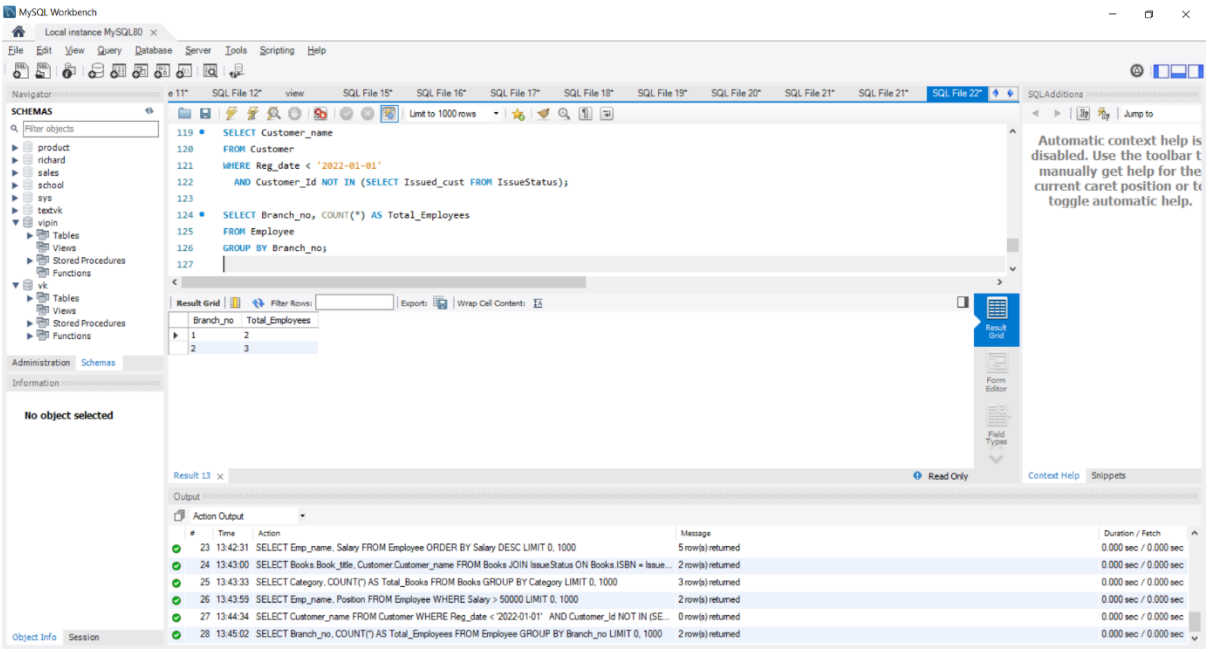
6. List the customer names who registered before 2022-01-01 and have not issued any books yet.

```
SELECT Customer_name

FROM Customer

WHERE Reg_date < '2022-01-01'

  AND Customer_Id NOT IN (SELECT Issued_cust FROM IssueStatus);
```



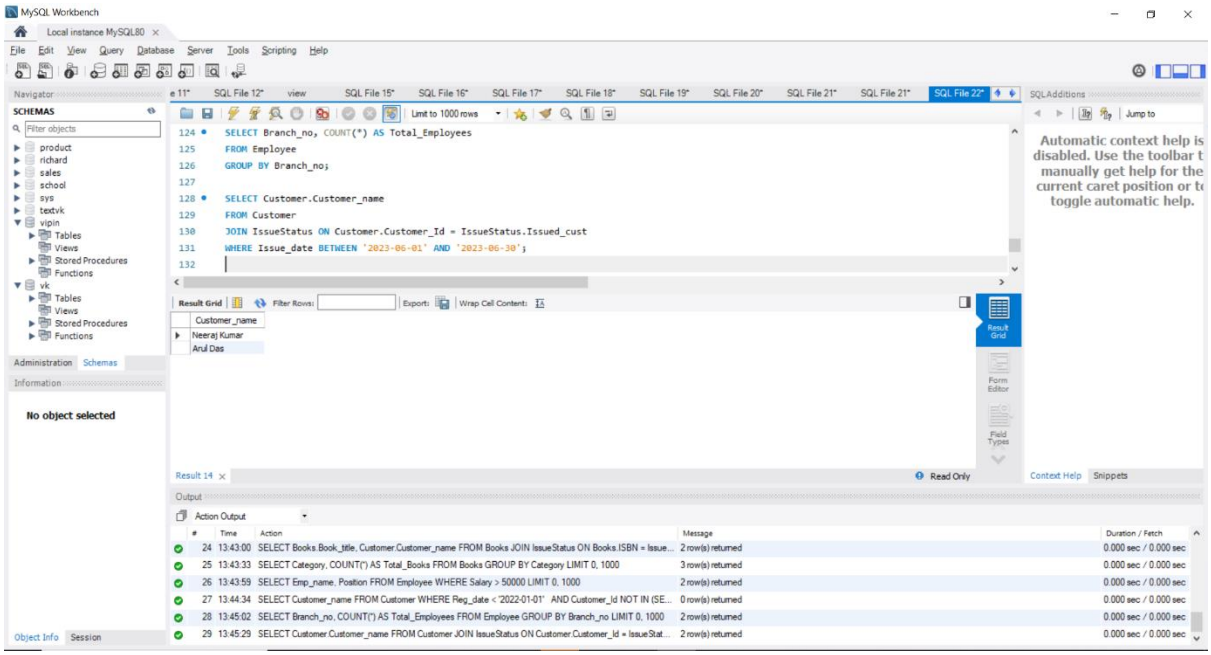
7. Display the branch numbers and the total count of employees in each branch.

```
SELECT Branch_no, COUNT(*) AS Total_Employees

FROM Employee

GROUP BY Branch_no;
```



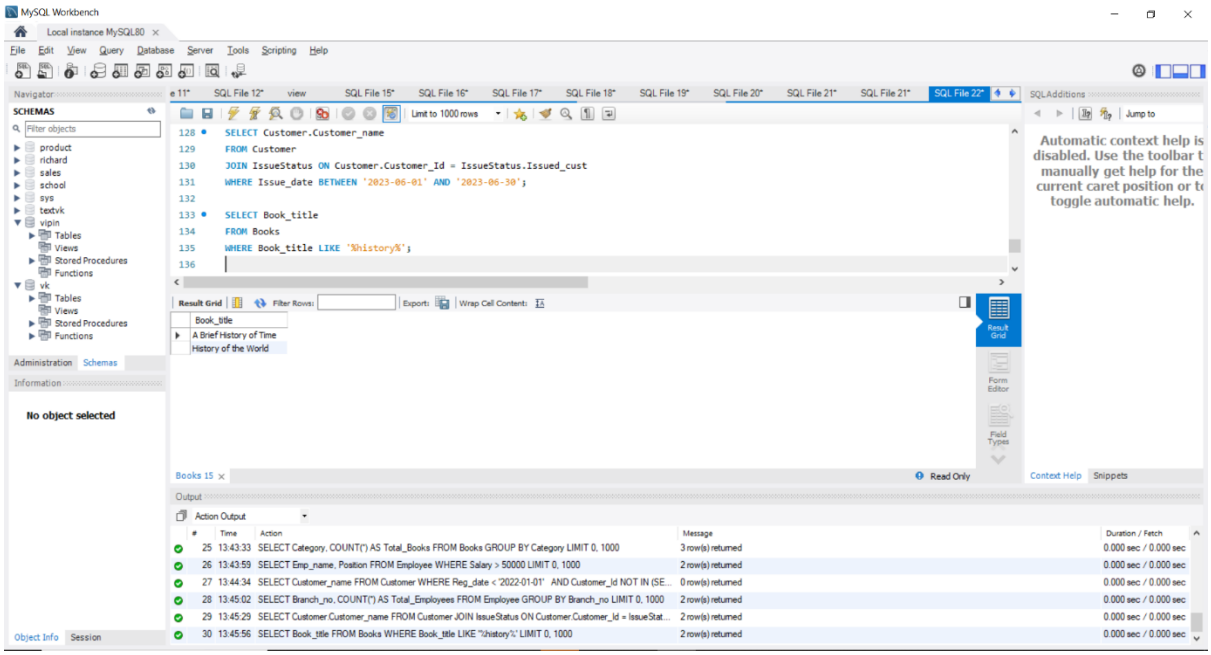
8. Display the names of customers who have issued books in the month of June 2023.

```
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';
```



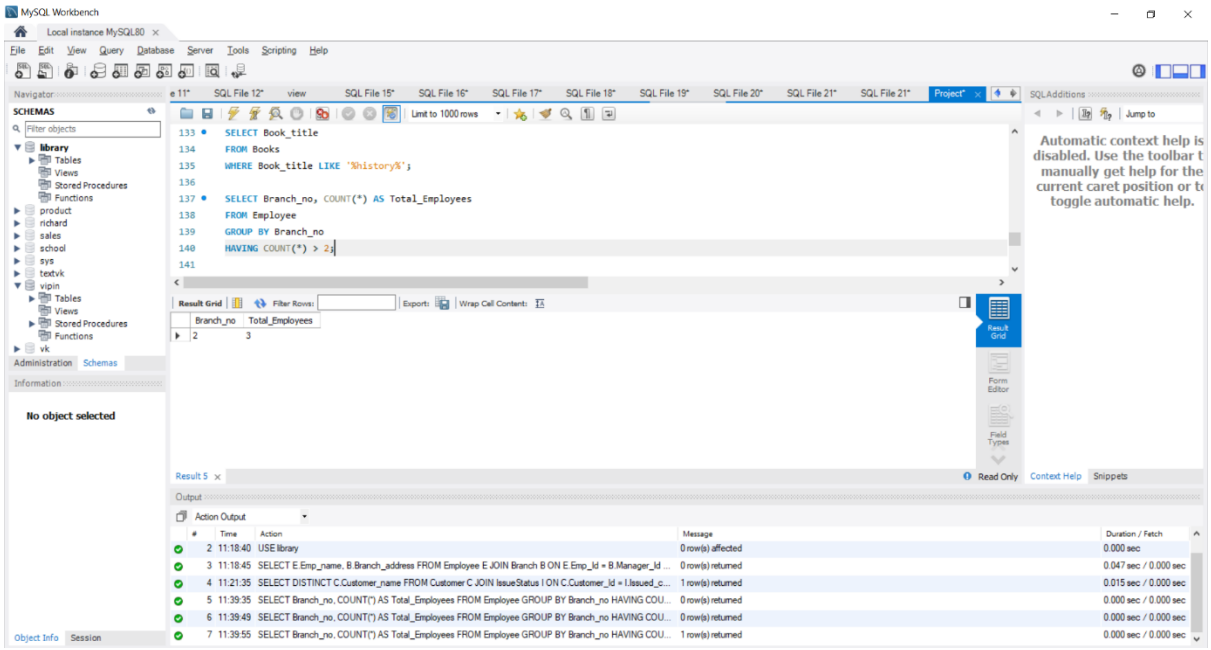
9. Retrieve book title from book table containing history.

```
SELECT Book_title

FROM Books

WHERE Book_title LIKE '%history%';
```



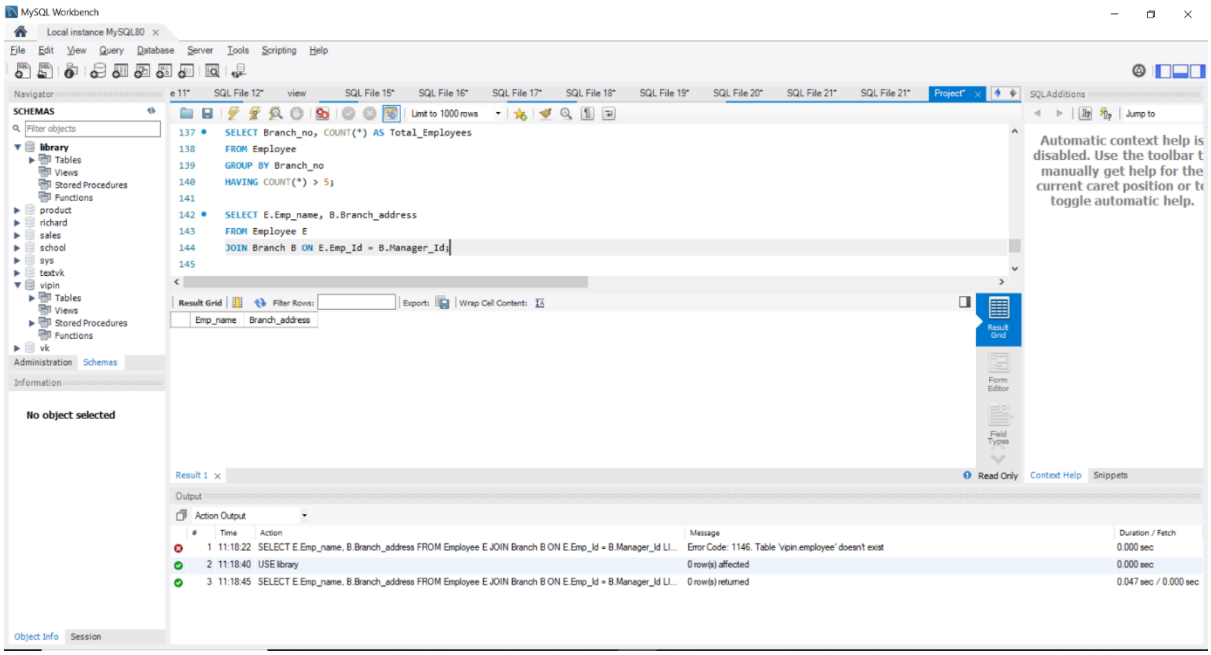
10. Retrieve the branch numbers along with the count of employees for branches having more than 2 employees

```
SELECT Branch_no, COUNT(*) AS Total_Employees

FROM Employee

GROUP BY Branch_no

HAVING COUNT(*) > 2;
```



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 C.Customer_name

FROM Customer C

JOIN IssueStatus I ON C.Customer_Id = I.Issued_cust

JOIN Books B ON I.Isbn_book = B.ISBN

WHERE B.Rental_Price > 25;
```