

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

The screenshot shows the MySQL Workbench interface. In the left sidebar, under the 'Schemas' section, there is a tree view. A node labeled 'library' is expanded, revealing several tables: books, branch, customer, employee, requeststatus, and returnstatus. Below the schema tree, there are sections for 'Views', 'Stored Procedures', and 'Functions'. The main workspace contains the following SQL code:

```
1. create database library;
2.
3. use library;
4.
5.
6.
7.
```

Below the code, the 'Table: employee' is selected. The 'Columns' section shows the following structure:

Column	Type	Properties
Emp_Id	int PK	
Emp_name	varchar(30)	
Position	varchar(30)	
Salary	int	
Branch_no	int	

The 'Output' pane at the bottom displays the execution history of the commands:

#	Time	Action	Message	Duration / Fetch
13	11:54:19	insert into branch (Branch_no, Manager_Id, Branch_address, Contact_no) VALUES (105,1006,600 Maple ...)	Error Code: 1062. Duplicate entry '105' for key 'branch.PRIMARY'	0.047 sec
14	11:54:30	insert into branch (Branch_no, Manager_Id, Branch_address, Contact_no) VALUES (105,1006,600 Maple ...)	1 row(s) affected	0.063 sec
15	11:57:06	INSERT INTO Employee (Grp_Id, Emp_name, Position, Salary, Branch_no) VALUES (2006, 'John Doe', 'Man...')	6 row(s) affected Records: 6 Duplicates: 0 Warnings: 0	0.265 sec
16	11:57:12	select b branch_no from branch b join employee e on b branch_no = e branch_no group by e branch_no hav... 1 row(s) returned		0.000 sec / 0.000 sec
17	11:57:23	select b branch_no from branch b left join employee e on b branch_no = e branch_no group by e branch_no ... Error Code: 1055. Expression #1 of SELECT list is not in GROUP BY clause and contains nonaggregated colu... 1 row(s) returned		0.000 sec
18	11:57:40	select b branch_no from branch b left join employee e on b branch_no = e branch_no group by b branch_no ... 1 row(s) returned		0.015 sec / 0.000 sec
19	11:59:59	use library	0 row(s) affected	0.000 sec
20	12:00:04	use library	0 row(s) affected	0.000 sec

TABLE 1. Branch

Branch_no - Set as PRIMARY KEY

Manager_Id

Branch_address

Contact_no

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** library
- Table:** Branch
- SQL Editor Content:**

```
L0· create table Branch(
L1  Branch_no int primary key,
L2  Manager_id int not null,
L3  Branch_address varchar(100) not null,
L4  Contact_no int not null);
L5
```

- Result Grid:** Shows the structure of the Branch table.

Field	Type	Null	Key	Default	Extra
Branch_no	int	NO	PRI	NULL	
Manager_id	int	NO		NULL	
Branch_address	varchar(100)	NO		NULL	
Contact_no	bigint	YES		NULL	

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Schemas: coapps_task-06-02-2024 joins views* subquery* SQL_enriched_project* joins stored_procedure* views_task Triggersql tcl* dcl

Navigator: 18

```
19 • INSERT INTO Branch (Branch_no, Manager_Id, Branch_address, Contact_no)
20   VALUES
21     (101, 1001, '123 Main Street', 1234567890),
22     (102, 1002, '456 Elm Street', 2345678901),
23     (103, 1003, '789 Oak Street', 3456789012),
24     (104, 1004, '321 Pine Street', 4567890123),
25     (105, 1005, '654 Maple Street', 5678901234);
26
27 • insert into branch (Branch_no, Manager_Id, Branch_address, Contact_no)
28   VALUES
29     (106,1006,'600 Maple Street', 5678904321);
```

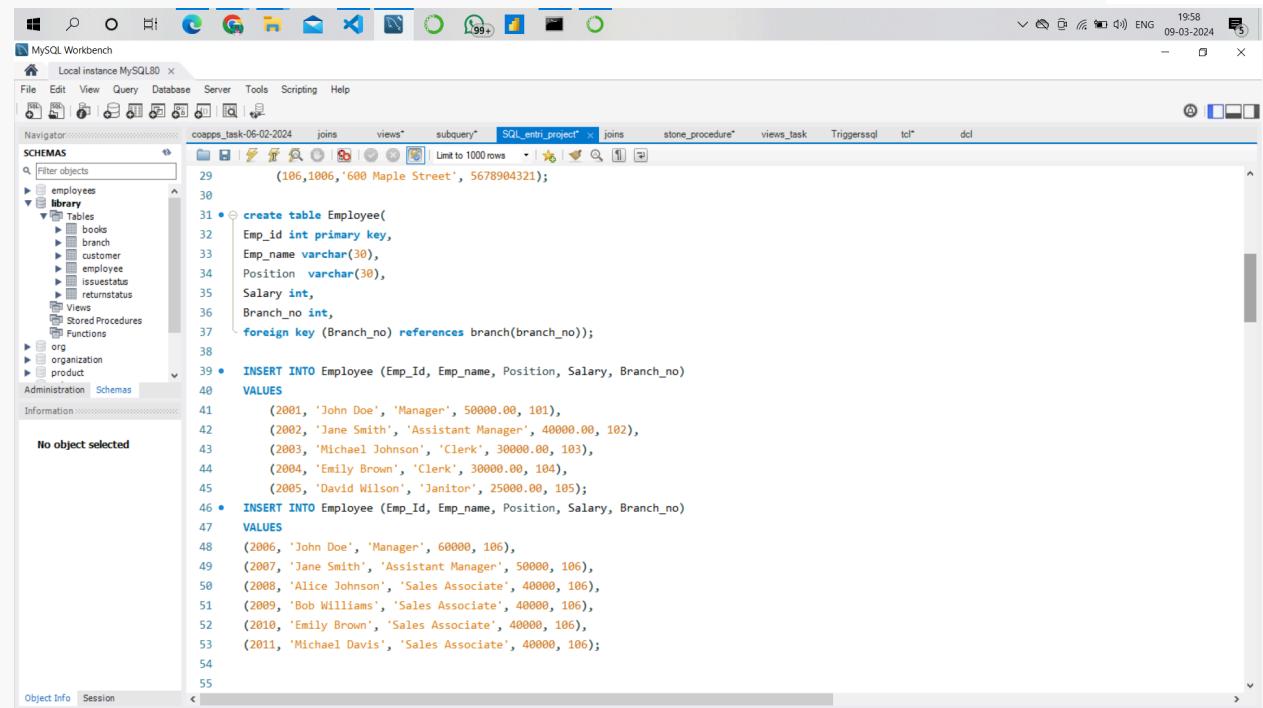
Table: employee

Columns:

Field	Type	Null	Key	Default	Extra
Emp_id	int PK				
Fname	varchar(30)				
Middle_name	varchar(30)				
Position	varchar(30)				
Salary	int				
Branch_no	int				

Result Grid | Filter Rows: Export: Wrap Cell Content: Result 6 x Read Only

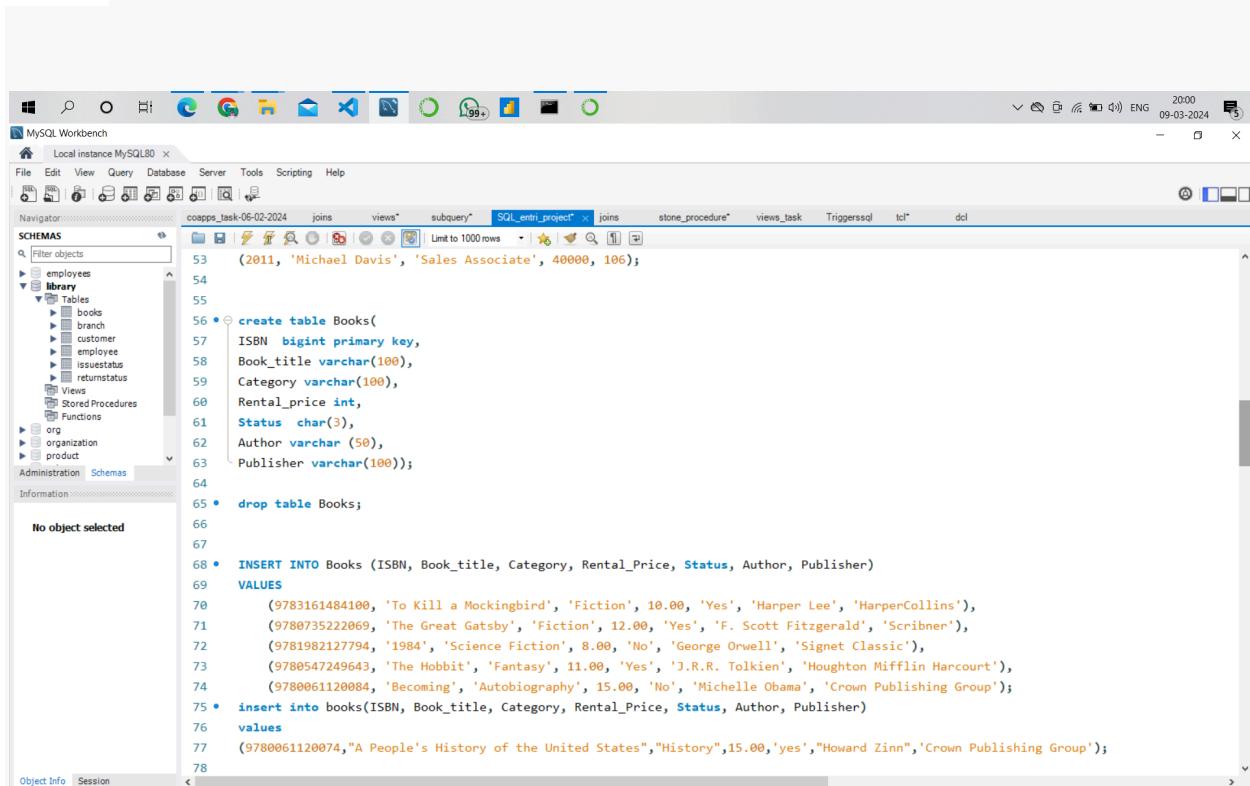
Result Grid | Form Editor | Field Types

TABLE 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**

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** The current schema is "library".
- Tables:** Inside "library", there are tables: books, branch, customer, issuestatus, returnstatus.
- SQL Editor:** The query tab contains the following SQL code:

```
29      (106,1006,'600 Maple Street', 5678904321);
30
31 • create table Employee(
32     Emp_Id int primary key,
33     Emp_name varchar(30),
34     Position varchar(30),
35     Salary int,
36     Branch_no int,
37     foreign key (Branch_no) references branch(branch_no));
38
39 • INSERT INTO Employee (Emp_Id, Emp_name, Position, Salary, Branch_no)
40     VALUES
41     (2001, 'John Doe', 'Manager', 50000.00, 101),
42     (2002, 'Jane Smith', 'Assistant Manager', 40000.00, 102),
43     (2003, 'Michael Johnson', 'Clerk', 30000.00, 103),
44     (2004, 'Emily Brown', 'Clerk', 30000.00, 104),
45     (2005, 'David Wilson', 'Janitor', 25000.00, 105);
46 • INSERT INTO Employee (Emp_Id, Emp_name, Position, Salary, Branch_no)
47     VALUES
48     (2006, 'John Doe', 'Manager', 60000, 106),
49     (2007, 'Jane Smith', 'Assistant Manager', 50000, 106),
50     (2008, 'Alice Johnson', 'Sales Associate', 40000, 106),
51     (2009, 'Bob Williams', 'Sales Associate', 40000, 106),
52     (2010, 'Emily Brown', 'Sales Associate', 40000, 106),
53     (2011, 'Michael Davis', 'Sales Associate', 40000, 106);
54
55
```

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**

The screenshot shows the MySQL Workbench interface with the 'SQL' tab selected. The code pane displays the creation of a 'Books' table and its insertion of data. The table structure includes columns for ISBN (primary key), Book_title, Category, Rental_Price, Status, Author, and Publisher. The data inserted includes various books like 'To Kill a Mockingbird', 'The Great Gatsby', 'The Hobbit', and 'A People's History of the United States'.

```
53     (2011, 'Michael Davis', 'Sales Associate', 40000, 106);
54
55
56 •  create table Books(
57     ISBN bigint primary key,
58     Book_title varchar(100),
59     Category varchar(100),
60     Rental_price int,
61     Status char(3),
62     Author varchar (50),
63     Publisher varchar(100));
64
65 •  drop table Books;
66
67
68 •  INSERT INTO Books (ISBN, Book_title, Category, Rental_Price, Status, Author, Publisher)
69     VALUES
70         (9783161484100, 'To Kill a Mockingbird', 'Fiction', 10.00, 'Yes', 'Harper Lee', 'HarperCollins'),
71         (9780735222069, 'The Great Gatsby', 'Fiction', 12.00, 'Yes', 'F. Scott Fitzgerald', 'Scribner'),
72         (9781982127794, '1984', 'Science Fiction', 8.00, 'No', 'George Orwell', 'Signet Classic'),
73         (9780547249643, 'The Hobbit', 'Fantasy', 11.00, 'Yes', 'J.R.R. Tolkien', 'Houghton Mifflin Harcourt'),
74         (9780061120084, 'Becoming', 'Autobiography', 15.00, 'No', 'Michelle Obama', 'Crown Publishing Group');
75 •  insert into books(ISBN, Book_title, Category, Rental_Price, Status, Author, Publisher)
76     values
77     (9780061120074,"A People's History of the United States","History",15.00,'yes',"Howard Zinn",'Crown Publishing Group');
```

4. Customer

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

The screenshot shows the MySQL Workbench interface with a query editor window. The code is as follows:

```
78
79 • create table Customer(
80     Customer_id int primary key,
81     Customer_name varchar(50),
82     Customer_address varchar(100),
83     Reg_date Date
84 );
85
86 • INSERT INTO Customer (Customer_Id, Customer_name, Customer_address, Reg_date)
87     VALUES
88         (3001, 'Alice Johnson', '123 Main Street', '2023-01-15'),
89         (3002, 'Bob Smith', '456 Elm Street', '2023-02-20'),
90         (3003, 'Charlie Brown', '789 Oak Street', '2023-03-10'),
91         (3004, 'Diana Wilson', '321 Pine Street', '2023-04-05'),
92         (3005, 'Eve Taylor', '654 Maple Street', '2023-05-12');
93
94 • create table IssueStatus(
```

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

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL script:

```
93
94 • create table IssueStatus(
95     Issue_id int primary key,
96     issued_cust int,
97     issued_book_name varchar(100),
98     Issue_date date,
99     isbn_book bigint,
100    foreign key (issued_cust) references Customer(customer_id) ON Delete cascade On update Cascade,
101    foreign key (isbn_book) references Books(ISBN) on delete cascade on update cascade);
102
103 • drop table IssueStatus;
104
105 • INSERT INTO IssueStatus (Issue_Id, Issued_cust, Issued_book_name, Issue_date, Isbn_book)
106     VALUES
107         (4001, 3001, 'To Kill a Mockingbird', '2023-01-20', 9783161484100),
108         (4002, 3002, 'The Great Gatsby', '2023-02-25', 9780735222069),
109         (4003, 3003, '1984', '2023-03-15', 9781982127794),
110         (4004, 3004, 'The Hobbit', '2023-04-10', 9780547249643),
111         (4005, 3005, 'Becoming', '2023-05-05', 9780061120084);
112
```

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

```

L12
L13 • create table ReturnStatus(
L14     Return_id int primary key,
L15     Return_cust int ,
L16     Return_book_name varchar(100),
L17     Return_date date,
L18     Isbn_book2 bigint,
L19         Foreign key (Isbn_book2) references Books(ISBN) on delete cascade on update cascade);
L20
L21 • drop table ReturnStatus;
L22 • INSERT INTO ReturnStatus (Return_Id, Return_cust, Return_book_name, Return_date, Isbn_book2)
L23     VALUES
L24         (5001, 3001, 'To Kill a Mockingbird', '2023-02-10', 9783161484100),
L25         (5002, 3002, 'The Great Gatsby', '2023-03-15', 9780735222069),
L26         (5003, 3003, '1984', '2023-04-20', 9781982127794),
L27         (5004, 3004, 'The Hobbit', '2023-05-25', 9780547249643),
L28         (5005, 3005, 'Becoming', '2023-06-10', 9780061120084);
L29
L30
L31
L32
L33
L34
L35
L36
L37
L38
L39
L40

```

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

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

```

L29
L30
L31
L32 #1. Retrieve the book title, category, and rental price of all available books
L33
L34 • select book_title,category,rental_price from books where status = 'yes';
L35
L36
L37
L38
L39
L40

```

book_title	category	rental_price
A People's History of the United States	History	15
The Hobbit	Fantasy	11
The Great Gatsby	Fiction	12
To Kill a Mockingbird	Fiction	10

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

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard MySQL icons for connection, schema selection, and query execution.
- Navigator:** Shows the database schema with the **library** schema expanded, containing tables like `books`, `branch`, `customer`, `employee`, `issuestatus`, and `returnstatus`.
- Query Editor:** A multi-line text area with numbered lines (L41-L51). Line L42 contains the query: `#2. List the employee names and their respective salaries in descending order of salary.`. Line L44 contains the SQL command: `select emp_name ,salary from employee order by salary desc;`.
- Result Grid:** A table titled "Result Grid" showing the results of the query. The columns are `emp_name` and `salary`. The data is as follows:

emp_name	salary
John Doe	60000
John Doe	50000
Jane Smith	50000
Jane Smith	40000
Alice Johnson	40000
Bob Williams	40000
Emily Brown	40000
Michael Davis	40000
Michael Johnson	30000
Emily Brown	30000
David Wilson	25000

At the bottom, tabs for "Object Info", "Session", and "employee 2" are visible, along with a "Read Only" button.

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

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard MySQL icons for connection, schema selection, and query execution.
- Navigator:** Shows the database schema with the **library** schema expanded, containing tables like `books`, `branch`, `customer`, `employee`, `issuestatus`, and `returnstatus`.
- Query Editor:** A multi-line text area with numbered lines (L49-L59). Line L53 contains the query: `#3. Retrieve the book titles and the corresponding customers who have issued those books.`. Line L55 contains the SQL command: `select Customer.customer_name,IssueStatus.Issued_book_name from Customer join IssueStatus on customer.customer_id = IssueStatus.issued_cust;`.
- Result Grid:** A table titled "Result Grid" showing the results of the query. The columns are `customer_name` and `Issued_book_name`. The data is as follows:

customer_name	Issued_book_name
Alice Johnson	To Kill a Mockingbird
Bob Smith	The Great Gatsby
Charlie Brown	1984
Diana Wilson	The Hobbit
Eve Taylor	Becoming

At the bottom, tabs for "Object Info", "Session", and "Result 3" are visible, along with a "Read Only" button.

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

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard MySQL icons for connection, schema, table, view, stored procedure, function, and other database operations.
- Schemas Navigator:** Shows the current schema is "library".
- Query Editor:** Contains the following SQL code:

```
161
162 #4. Display the total count of books in each category.
163
164 • select category,count(*) as count_of_books_category from books group by category order by count_of_books_category desc;
```
- Result Grid:** Displays the results of the query:

category	count_of_books_category
Fiction	2
History	1
Autobiography	1
Fantasy	1
Science Fiction	1
- Status Bar:** Local instance MySQL80, 2004, 09-03-2024, ENG.

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

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard MySQL icons for connection, schema, table, view, stored procedure, function, and other database operations.
- Schemas Navigator:** Shows the current schema is "library".
- Query Editor:** Contains the following SQL code:

```
174
175 #5. Retrieve the employee names and their positions for the employees whose salaries are above Rs.50,000
176
177 • select emp_name,position,salary from employee where salary >= 50000;
```
- Result Grid:** Displays the results of the query:

emp_name	position	salary
John Doe	Manager	50000
John Doe	Manager	60000
Jane Smith	Assistant Manager	50000
- Status Bar:** Local instance MySQL80, 2004, 09-03-2024, ENG.

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

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** library
- Query Editor:** SQL_entri_project
- Code:**

```
181
182
183
184
185
186
187 #6. List the customer names who registered before 2022-01-01 and have not issued any books yet
188
189 • select customer.customer_name from customer
190 join issuestatus
191 on customer.customer_name = issuestatus.issued_cust
192 where customer.reg_date < '2022-01-01' and issued_cust is null;
193
```
- Result Grid:** Displays the column "customer_name" with no data.

There is no customer registered before 2022-01-01 ,So we can't able to see the values

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

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** library
- Query Editor:** SQL_entri_project
- Code:**

```
193
194 #7. Display the branch numbers and the total count of employees in each branch.
195
196 • select b.branch_no,count(e.emp_id) as employees_in_each_branch from branch b
197 join employee e
198 on b.branch_no = e.branch_no
199 group by b.branch_no;
```
- Result Grid:** Displays the columns "branch_no" and "employees_in_each_branch". The data is as follows:

branch_no	employees_in_each_branch
101	1
102	1
103	1
104	1
105	1
106	6

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

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard database management tools.
- Navigator:** Schemas (employees, library, org, organization, product) and Tables (books, branch, customer, employee, issuestatus, returnstatus, views).
- Query Editor:** SQL_enqri_project* tab, containing the following code:

```
213
214
215
216
217 #8. Display the names of customers who have issued books in the month of may 2023.
218
219 • select c.customer_name from customer c
220   left join issuestatus i
221   on c.customer_id = i.issued_cust
222   where monthname(i.issue_date) = 'may' and year(i.issue_date) = '2023';
223
224
225
```
- Result Grid:** Shows the result of the query with one row: "Eve Taylor".
- Status Bar:** Local instance MySQL80, 20:12, 09-03-2024, ENG.

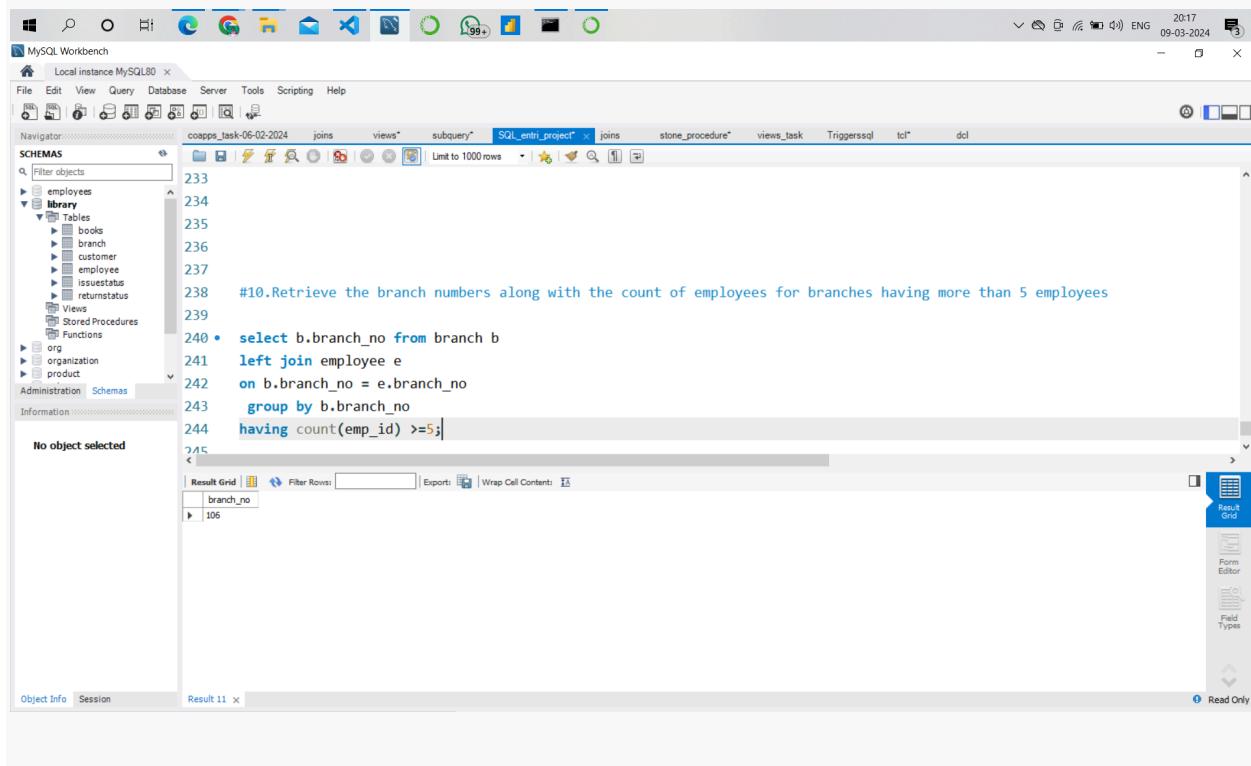
9. Retrieve book_title from book table containing history.

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard database management tools.
- Navigator:** Schemas (employees, library, org, organization, product) and Tables (books, branch, customer, employee, issuestatus, returnstatus, views).
- Query Editor:** SQL_enqri_project* tab, containing the following code:

```
225
226
227
228 #9. Retrieve book_title from book table containing history.
229
230 • select book_title from books where category = 'history';
231
232
233
234
235
236
237
```
- Result Grid:** Shows the result of the query with one row: "A People's History of the United States".
- Status Bar:** Local instance MySQL80, 20:17, 09-03-2024, ENG.

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



The screenshot shows the MySQL Workbench interface. The query editor contains the following SQL code:

```
#10.Retrieve the branch numbers along with the count of employees for branches having more than 5 employees
233
234
235
236
237
238 select b.branch_no from branch b
239 left join employee e
240 on b.branch_no = e.branch_no
241 group by b.branch_no
242 having count(emp_id) >=5;
243
244
245
```

The results grid shows the output of the query:

branch_no
106