

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;

The screenshot displays the MySQL Workbench interface. The main editor window shows the following SQL queries:

```
1. CREATE DATABASE library;
2. USE library;
3. CREATE TABLE Branch (
4.     Branch_no INT PRIMARY KEY,
5.     Manager_Id INT,
6.     Branch_address VARCHAR(255),
7.     Contact_no VARCHAR(15)
8. );
9. select* from Branch;
```

The left sidebar shows the 'SCHEMAS' panel with a tree view of databases and tables. The 'Output' panel at the bottom shows the execution results:

Action	Time	Message	Duration / Fetch
1	13:26:38	CREATE DATABASE library	1 row(s) affected / 0.000 sec
2	13:26:40	USE library	0 row(s) affected / 0.000 sec
3	13:26:53	CREATE TABLE Branch (Branch_no INT PRIMARY KEY, Manager_Id INT, Branch_address VARCHA...	0 row(s) affected / 0.032 sec
4	13:27:42	select* from Branch LIMIT 0, 1000	0 row(s) returned / 0.000 sec / 0.000 sec

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;
```

The screenshot displays the MySQL Workbench interface. The SQL Editor at the top contains the following SQL script:

```
11 CREATE DATABASE library;
12
13 CREATE TABLE Branch (
14     Branch_no INT PRIMARY KEY,
15     Manager_id INT,
16     Branch_address VARCHAR(100),
17     Branch_type VARCHAR(50),
18 );
19
20 CREATE TABLE Employee (
21     Emp_Id INT PRIMARY KEY,
22     Emp_name VARCHAR(100),
23     Position VARCHAR(50),
24     Salary DECIMAL(10, 2),
25     Branch_no INT,
26     FOREIGN KEY (Branch_no) REFERENCES Branch(Branch_no)
27 );
28
29 select* from Employee;
```

The Output window at the bottom shows the execution results of these commands:

#	Time	Action	Message	Duration / Fetch
1	13:26:38	CREATE DATABASE library	1 row(s) affected	0.000 sec
2	13:26:40	USE library	0 row(s) affected	0.000 sec
3	13:26:53	CREATE TABLE Branch (Branch_no INT PRIMARY KEY, Manager_id INT, Branch_address VARCHA...	0 row(s) affected	0.032 sec
4	13:27:42	select* from Branch LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
5	13:28:24	CREATE TABLE Employee (Emp_Id INT PRIMARY KEY, Emp_name VARCHAR(100), Position VARCH...	0 row(s) affected	0.047 sec
6	13:28:42	select* from Employee LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

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;

The screenshot shows the MySQL Workbench interface. The main editor window displays the following SQL code:

```
22 ISBN VARCHAR(13) PRIMARY KEY,  
23 Book_title VARCHAR(255),  
24 Category VARCHAR(50),  
25 Rental_Price DECIMAL(10, 2),  
26 Status ENUM('yes', 'no'),  
27 Author VARCHAR(100),  
28 Publisher VARCHAR(100)  
29 );  
30 select* from Books;
```

The left sidebar shows the 'SCHEMAS' panel with a tree view of databases and tables. The 'Result Grid' panel at the bottom shows the output of the query, which is empty. The 'Output' panel at the bottom shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
3	13:26:53	CREATE TABLE Branch (Branch_no INT PRIMARY KEY, Manager_id INT, Branch_address VARCHAR(100));	0 row(s) affected	0.032 sec
4	13:27:42	select* from Branch LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
5	13:28:24	CREATE TABLE Employee (Emp_id INT PRIMARY KEY, Emp_name VARCHAR(100), Position VARCHAR(50));	0 row(s) affected	0.047 sec
6	13:28:42	select* from Employee LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
7	13:29:11	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));	0 row(s) affected	0.047 sec
8	13:29:25	select* from Books LIMIT 0, 1000	0 row(s) returned	0.016 sec / 0.000 sec

Table 4

CREATE TABLE Customer (

Customer_Id INT PRIMARY KEY,

Customer_name VARCHAR(100),

Customer_address VARCHAR(255),

Reg_date DATE

);

select* from Customer;

The screenshot displays the MySQL Workbench interface. The left sidebar shows the 'SCHEMAS' panel with a tree view of databases including 'product', 'richard', 'sales', 'school', 'sys', 'testvk', 'vipin', and 'vk'. The main editor window shows a SQL script with the following content:

```
30 • select* from Books;
31
32 • CREATE TABLE Customer (
33   Customer_Id INT PRIMARY KEY,
34   Customer_name VARCHAR(100),
35   Customer_address VARCHAR(255),
36   Reg_date DATE
37 );
38 • select* from Customer;
```

Below the script, the 'Result Grid' is visible, showing the columns: Customer_Id, Customer_name, Customer_address, and Reg_date. The bottom panel shows the 'Output' tab with a table of execution results:

#	Time	Action	Message	Duration / Fetch
5	13:28:24	CREATE TABLE Employee (Emp_Id INT PRIMARY KEY, Emp_name VARCHAR(100), Position VAR...	0 row(s) affected	0.047 sec
6	13:28:42	select* from Employee LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
7	13:29:11	CREATE TABLE Books (ISBN VARCHAR(13) PRIMARY KEY, Book_title VARCHAR(255), Category V...	0 row(s) affected	0.047 sec
8	13:29:25	select* from Books LIMIT 0, 1000	0 row(s) returned	0.016 sec / 0.000 sec
9	13:29:52	CREATE TABLE Customer (Customer_Id INT PRIMARY KEY, Customer_name VARCHAR(100), Custo...	0 row(s) affected	0.016 sec
10	13:30:11	select* from Customer LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

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;
```

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with various databases like 'product', 'richard', 'sales', 'school', 'sys', 'testvk', and 'vipin'. The main editor window shows the SQL script for creating the 'IssueStatus' table, including primary keys and foreign key constraints. Below the script, the 'Result Grid' shows the output of the 'select* from IssueStatus;' query, which is currently empty. The bottom panel shows the 'Output' tab with a log of database actions, including the successful creation of the 'IssueStatus' table and the execution of the select query.

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

#	Time	Action	Message	Duration / Fetch
7	13:29:11	CREATE TABLE Books (ISBN VARCHAR(13) PRIMARY KEY, Book_title VARCHAR(255), Category V...	0 row(s) affected	0.047 sec
8	13:29:25	select* from Books LIMIT 0, 1000	0 row(s) returned	0.016 sec / 0.000 sec
9	13:29:52	CREATE TABLE Customer (Customer_Id INT PRIMARY KEY, Customer_name VARCHAR(100), Custo...	0 row(s) affected	0.016 sec
10	13:30:11	select* from Customer LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
11	13:30:36	CREATE TABLE IssueStatus (Issue_Id INT PRIMARY KEY, Issued_cust INT, Issued_book_name VA...	0 row(s) affected	0.062 sec
12	13:30:54	select* from IssueStatus LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

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;
```

The screenshot shows the MySQL Workbench interface. The main editor window displays the SQL script for creating the `ReturnStatus` table and a subsequent `select* from ReturnStatus;` query. The script includes the following SQL statements:

```
52 Return_Id INT PRIMARY KEY,  
53 Return_cust INT,  
54 Return_book_name VARCHAR(255),  
55 Return_date DATE,  
56 Isbn_book2 VARCHAR(13),  
57 FOREIGN KEY (Return_cust) REFERENCES Customer(Customer_Id),  
58 FOREIGN KEY (Isbn_book2) REFERENCES Books(ISBN)  
59 );  
60 select* from ReturnStatus;
```

The left sidebar shows the Schemas pane with a tree view of the database structure. The bottom pane shows the Output window with the following table:

#	Time	Action	Message	Duration / Fetch
9	13:29:52	CREATE TABLE Customer (Customer_Id INT PRIMARY KEY, Customer_name VARCHAR(100), Cus...	0 row(s) affected	0.016 sec
10	13:30:11	select* from Customer LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
11	13:30:36	CREATE TABLE IssueStatus (Issue_Id INT PRIMARY KEY, Issued_cust INT, Issued_book_name VA...	0 row(s) affected	0.062 sec
12	13:30:54	select* from IssueStatus LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
13	13:31:22	CREATE TABLE ReturnStatus (Return_Id INT PRIMARY KEY, Return_cust INT, Return_book_name ...	0 row(s) affected	0.063 sec
14	13:31:37	select* from ReturnStatus LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

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';
```

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

```
92  
93 • INSERT INTO ReturnStatus (Return_Id, Return_cust, Return_book_name, Return_date, Isbn_book2) VALUES  
94 (1, 1, 'Data Structures', '2023-06-30', '978-3-16-0');  
95  
96  
97  
98 • SELECT Book_title, Category, Rental_Price  
99 FROM Books  
100 WHERE Status = 'yes';
```

The Results tab shows the output of the SELECT query:

Book_title	Category	Rental_Price
Artificial Intelligence	Computer Science	25.00
Introduction to Algorithms	Computer Science	30.00
History of the World	History	18.00
Computer Networks	Computer Science	28.00
Data Structures	Computer Science	20.00

The Output tab shows the execution log:

#	Time	Action	Message	Duration / Fetch
17	13:36:57	INSERT INTO Books (ISBN, Book_title, Category, Rental_Price, Status, Author, Publisher) VALUES (978-3-16...	Error Code: 1406: Data too long for column 'ISBN' at row 1	0.000 sec
18	13:38:00	INSERT INTO Books (ISBN, Book_title, Category, Rental_Price, Status, Author, Publisher) VALUES (978-3-16...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.000 sec
19	13:40:05	INSERT INTO Customer (Customer_Id, Customer_name, Customer_address, Reg_date) VALUES (1, Neeraj K...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
20	13:40:49	INSERT INTO IssueStatus (Issue_Id, Issued_cust, Issued_book_name, Issue_date, Isbn_book) VALUES (1, 1...	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.000 sec
21	13:41:09	INSERT INTO ReturnStatus (Return_Id, Return_cust, Return_book_name, Return_date, Isbn_book2) VALUE...	1 row(s) affected	0.000 sec
22	13:41:59	SELECT Book_title, Category, Rental_Price FROM Books WHERE Status = 'yes' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

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

```
SELECT Emp_name, Salary
```

```
FROM Employee
```

```
ORDER BY Salary DESC;
```

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

```
96  
97  
98 • SELECT Book_title, Category, Rental_Price  
99 FROM Books  
100 WHERE Status = 'yes';  
101  
102 • SELECT Emp_name, Salary  
103 FROM Employee  
104 ORDER BY Salary DESC;
```

The Results Grid shows the output of the query:

Emp_name	Salary
Grace	62000.00
Thomas	60000.00
Arun Varma	45000.00
Yesudas	35000.00
Alexander	32000.00

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
18	13:38:00	INSERT INTO Books (ISBN, Book_title, Category, Rental_Price, Status, Author, Publisher) VALUES (978-3-16...	8 row(s) affected Records: 8 Duplicates: 0 Warnings: 0	0.000 sec
19	13:40:05	INSERT INTO Customer (Customer_id, Customer_name, Customer_address, Reg_date) VALUES (1, Neeraj K...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
20	13:40:49	INSERT INTO IssueStatus (Issue_id, Issue_cust, Issue_book_name, Issue_date, Issue_book) VALUES (1, 1...	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.000 sec
21	13:41:09	INSERT INTO ReturnStatus (Return_id, Return_cust, Return_book_name, Return_date, Return_book2) VALUE...	1 row(s) affected	0.000 sec
22	13:41:59	SELECT Book_title, Category, Rental_Price FROM Books WHERE Status = 'yes' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
23	13:42:31	SELECT Emp_name, Salary FROM Employee ORDER BY Salary DESC LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

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;
```

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

```
182 SELECT Emp_name, Salary  
183 FROM Employee  
184 ORDER BY Salary DESC;  
185  
186 SELECT Books.Book_title, Customer.Customer_name  
187 FROM Books  
188 JOIN IssueStatus ON Books.ISBN = IssueStatus.Isbn_book  
189 JOIN Customer ON IssueStatus.Issued_cust = Customer.Customer_Id;  
190
```

The Results tab shows the output of the query:

Book_title	Customer_name
Data Structures	Neeraj Kumar
Introduction to Algorithms	Anul Das

The Output tab shows the execution log:

#	Time	Action	Message	Duration / Fetch
19	13:40:05	INSERT INTO Customer (Customer_Id, Customer_name, Customer_address, Reg_date) VALUES (1, 'Neeraj K...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
20	13:40:49	INSERT INTO IssueStatus (Issue_Id, Issued_cust, Issued_book_name, Issue_date, Isbn_book) VALUES (1, 1...	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.000 sec
21	13:41:09	INSERT INTO ReturnStatus (Return_Id, Return_cust, Return_book_name, Return_date, Isbn_book2) VALUE...	1 row(s) affected	0.000 sec
22	13:41:59	SELECT Book_title, Category, Rental_Price FROM Books WHERE Status = 'yes' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
23	13:42:31	SELECT Emp_name, Salary FROM Employee ORDER BY Salary DESC LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
24	13:43:00	SELECT Books.Book_title, Customer.Customer_name FROM Books JOIN IssueStatus ON Books.ISBN = Issue...	2 row(s) returned	0.000 sec / 0.000 sec

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

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

```
FROM Books
```

```
GROUP BY Category;
```

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

```
106 SELECT Books.Book_title, Customer.Customer_name
107 FROM Books
108 JOIN IssueStatus ON Books.ISBN = IssueStatus.Isbn_book
109 JOIN Customer ON IssueStatus.Issued_cust = Customer.Customer_Id;
110
111 SELECT Category, COUNT(*) AS Total_Books
112 FROM Books
113 GROUP BY Category;
114
```

The Results window displays the output of the second query:

Category	Total_Books
Computer Science	5
Science	1
History	2

The Output window shows the execution log:

#	Time	Action	Message	Duration / Fetch
20	13:40:49	INSERT INTO IssueStatus (Issue_Id, Issued_cust, Issued_book_name, Issue_date, Isbn_book) VALUES (1, 1, '...', '...', '...')	2 row(s) affected Records: 2 Duplicates: 0 Warnings: 0	0.000 sec
21	13:41:09	INSERT INTO ReturnStatus (Return_Id, Return_cust, Return_book_name, Return_date, Isbn_book2) VALUES (1, 1, '...', '...', '...')	1 row(s) affected	0.000 sec
22	13:41:59	SELECT Book_title, Category, Rental_Price FROM Books WHERE Status = 'yes' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
23	13:42:31	SELECT Emp_name, Salary FROM Employee ORDER BY Salary DESC LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
24	13:43:00	SELECT Books.Book_title, Customer.Customer_name FROM Books JOIN IssueStatus ON Books.ISBN = Issue...	2 row(s) returned	0.000 sec / 0.000 sec
25	13:43:33	SELECT Category, COUNT(*) AS Total_Books FROM Books GROUP BY Category LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

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;
```

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

```
110
111 SELECT Category, COUNT(*) AS Total_Books
112 FROM Books
113 GROUP BY Category;
114
115 SELECT Emp_name, Position
116 FROM Employee
117 WHERE Salary > 50000;
118
```

The Results tab shows the output of the second query:

Emp_name	Position
Thomas	Manager
Grace	Manager

The Output tab shows the execution log:

#	Time	Action	Message	Duration / Fetch
21	13:41:09	INSERT INTO ReturnStatus (Return_id, Return_cust, Return_book_name, Return_date, isbn_book2) VALUES...	1 row(s) affected	0.000 sec
22	13:41:59	SELECT Book_title, Category, Rental_Price FROM Books WHERE Status = 'yes' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
23	13:42:31	SELECT Emp_name, Salary FROM Employee ORDER BY Salary DESC LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
24	13:43:00	SELECT Books.Book_title, Customer.Customer_name FROM Books JOIN IssueStatus ON Books.ISBN = Issue...	2 row(s) returned	0.000 sec / 0.000 sec
25	13:43:33	SELECT Category, COUNT(*) AS Total_Books FROM Books GROUP BY Category LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
26	13:43:59	SELECT Emp_name, Position FROM Employee WHERE Salary > 50000 LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

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);
```

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

```
115 SELECT Emp_name, Position
116 FROM Employee
117 WHERE Salary > 50000;
118
119 SELECT Customer_name
120 FROM Customer
121 WHERE Reg_date < '2022-01-01'
122 AND Customer_Id NOT IN (SELECT Issued_cust FROM IssueStatus);
123
```

The Results pane shows the output of the query, displaying a single column labeled 'Customer_name'.

The Output pane shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fresh
22	13:41:59	SELECT Book_title, Category, Rental_Price FROM Books WHERE Status = 'yes' LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
23	13:42:31	SELECT Emp_name, Salary FROM Employee ORDER BY Salary DESC LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
24	13:43:00	SELECT Books.Book_title, Customer.Customer_name FROM Books JOIN IssueStatus ON Books.ISBN = Issue...	2 row(s) returned	0.000 sec / 0.000 sec
25	13:43:33	SELECT Category, COUNT(*) AS Total_Books FROM Books GROUP BY Category LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
26	13:43:59	SELECT Emp_name, Position FROM Employee WHERE Salary > 50000 LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
27	13:44:34	SELECT Customer_name FROM Customer WHERE Reg_date < '2022-01-01' AND Customer_Id NOT IN (SE...	0 row(s) returned	0.000 sec / 0.000 sec

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;
```

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

```
119 SELECT Customer_name
120 FROM Customer
121 WHERE Reg_date < '2022-01-01'
122 AND Customer_Id NOT IN (SELECT Issued_cust FROM IssueStatus);
123
124 SELECT Branch_no, COUNT(*) AS Total_Employees
125 FROM Employee
126 GROUP BY Branch_no;
127
```

The Results window displays the output of the second query:

Branch_no	Total_Employees
1	2
2	3

The Output window shows the execution log:

#	Time	Action	Message	Duration / Fetch
23	13:42:31	SELECT Emp_name, Salary FROM Employee ORDER BY Salary DESC LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
24	13:43:00	SELECT Books.Book_title, Customer.Customer_name FROM Books JOIN IssueStatus ON Books.ISBN = Issue...	2 row(s) returned	0.000 sec / 0.000 sec
25	13:43:33	SELECT Category, COUNT(*) AS Total_Books FROM Books GROUP BY Category LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
26	13:43:59	SELECT Emp_name, Position FROM Employee WHERE Salary > 50000 LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
27	13:44:34	SELECT Customer_name FROM Customer WHERE Reg_date < '2022-01-01' AND Customer_Id NOT IN (SE...	0 row(s) returned	0.000 sec / 0.000 sec
28	13:45:02	SELECT Branch_no, COUNT(*) AS Total_Employees FROM Employee GROUP BY Branch_no LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

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';
```

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

```
124 • SELECT Branch_no, COUNT(*) AS Total_Employees  
125 FROM Employee  
126 GROUP BY Branch_no;  
127  
128 • SELECT Customer.Customer_name  
129 FROM Customer  
130 JOIN IssueStatus ON Customer.Customer_Id = IssueStatus.Issued_cust  
131 WHERE Issue_date BETWEEN '2023-06-01' AND '2023-06-30';  
132
```

The Results tab shows the following data:

Customer_name
Neeraj Kumar
Anil Das

The Action Output tab shows the following log:

#	Time	Action	Message	Duration / Fetch
24	13:43:00	SELECT Books.Book_title, Customer.Customer_name FROM Books JOIN IssueStatus ON Books.ISBN = Issue...	2 row(s) returned	0.000 sec / 0.000 sec
25	13:43:33	SELECT Category, COUNT(*) AS Total_Books FROM Books GROUP BY Category LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
26	13:43:59	SELECT Emp_name, Position FROM Employee WHERE Salary > 50000 LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
27	13:44:34	SELECT Customer_name FROM Customer WHERE Reg_date < '2022-01-01' AND Customer_Id NOT IN (SE...	0 row(s) returned	0.000 sec / 0.000 sec
28	13:45:02	SELECT Branch_no, COUNT(*) AS Total_Employees FROM Employee GROUP BY Branch_no LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
29	13:45:29	SELECT Customer.Customer_name FROM Customer JOIN IssueStatus ON Customer.Customer_Id = IssueStat...	2 row(s) returned	0.000 sec / 0.000 sec

9. Retrieve book title from book table containing history.

```
SELECT Book_title
```

```
FROM Books
```

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

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

```
128 SELECT Customer.Customer_name
129 FROM Customer
130 JOIN IssueStatus ON Customer.Customer_Id = IssueStatus.Issued_cust
131 WHERE Issue_date BETWEEN '2023-06-01' AND '2023-06-30';
132
133 SELECT Book_title
134 FROM Books
135 WHERE Book_title LIKE '%history%';
136
```

The Results window shows the output of the second query, displaying a table with one row:

Book_title
A Brief History of Time

The Output window shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
25	13.43.33	SELECT Category.COUNT() AS Total_Books FROM Books GROUP BY Category LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
26	13.43.59	SELECT Emp_name, Position FROM Employee WHERE Salary > 50000 LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
27	13.44.34	SELECT Customer_name FROM Customer WHERE Reg_date < '2022-01-01' AND Customer_Id NOT IN (SE...	0 row(s) returned	0.000 sec / 0.000 sec
28	13.45.02	SELECT Branch_no, COUNT() AS Total_Employees FROM Employee GROUP BY Branch_no LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
29	13.45.29	SELECT Customer.Customer_name FROM Customer JOIN IssueStatus ON Customer.Customer_Id = IssueStat...	2 row(s) returned	0.000 sec / 0.000 sec
30	13.45.56	SELECT Book_title FROM Books WHERE Book_title LIKE '%history%'; LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

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;
```

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

```
133 SELECT Book_title
134 FROM Books
135 WHERE Book_title LIKE '%History%';
136
137 SELECT Branch_no, COUNT(*) AS Total_Employees
138 FROM Employee
139 GROUP BY Branch_no
140 HAVING COUNT(*) > 2;
141
```

The Results window shows the output of the second query:

Branch_no	Total_Employees
2	3

The Action Output window shows the execution log:

#	Time	Action	Message	Duration / Fetch
2	11:18:40	USE library	0 row(s) affected	0.000 sec
3	11:18:45	SELECT E.Emp_name, B.Branch_address FROM Employee E JOIN Branch B ON E.Emp_Id = B.Manager_Id ...	0 row(s) returned	0.047 sec / 0.000 sec
4	11:21:35	SELECT DISTINCT C.Customer_name FROM Customer C JOIN IssueStatus I ON C.Customer_Id = I.Issued_C...	1 row(s) returned	0.015 sec / 0.000 sec
5	11:39:35	SELECT Branch_no, COUNT(*) AS Total_Employees FROM Employee GROUP BY Branch_no HAVING COU...	0 row(s) returned	0.000 sec / 0.000 sec
6	11:39:49	SELECT Branch_no, COUNT(*) AS Total_Employees FROM Employee GROUP BY Branch_no HAVING COU...	0 row(s) returned	0.000 sec / 0.000 sec
7	11:39:55	SELECT Branch_no, COUNT(*) AS Total_Employees FROM Employee GROUP BY Branch_no HAVING COU...	1 row(s) returned	0.000 sec / 0.000 sec

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;
```

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

```
137 SELECT Branch_no, COUNT(*) AS Total_Employees
138 FROM Employee
139 GROUP BY Branch_no
140 HAVING COUNT(*) > 5;
141
142 SELECT E.Emp_name, B.Branch_address
143 FROM Employee E
144 JOIN Branch B ON E.Emp_Id = B.Manager_Id;
145
```

The Results window shows the execution of the second query. The output is as follows:

Emp_name	Branch_address
----------	----------------

The Action Output window shows the following messages:

#	Time	Action	Message	Duration / Fetch
1	11:18:22	SELECT E.Emp_name, B.Branch_address FROM Employee E JOIN Branch B ON E.Emp_Id = B.Manager_Id LL...	Errr Code: 1146. Table 'vipin.employee' doesn't exist	0.000 sec
2	11:18:40	USE library	0 row(s) affected	0.000 sec
3	11:18:45	SELECT E.Emp_name, B.Branch_address FROM Employee E JOIN Branch B ON E.Emp_Id = B.Manager_Id LL...	0 row(s) returned	0.047 sec / 0.000 sec

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;
```

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

```
143 FROM Employee E  
144 JOIN Branch B ON E.Emp_Id = B.Manager_Id;  
145  
146 SELECT DISTINCT C.Customer_name  
147 FROM Customer C  
148 JOIN IssueStatus I ON C.Customer_Id = I.Issued_cust  
149 JOIN Books B ON I.Isbn_book = B.ISBN  
150 WHERE B.Rental_Price > 25;  
151
```

The Results tab shows the following data:

Customer_name
Anul Des

The Output tab shows the execution log:

#	Time	Action	Message	Duration / Fetch
1	11:18:22	SELECT E.Emp_name, B.Branch_address FROM Employee E JOIN Branch B ON E.Emp_Id = B.Manager_Id LI...	Error Code: 1146. Table 'vpin employee' doesn't exist	0.000 sec
2	11:18:40	USE library	0 row(s) affected	0.000 sec
3	11:18:45	SELECT E.Emp_name, B.Branch_address FROM Employee E JOIN Branch B ON E.Emp_Id = B.Manager_Id LI...	0 row(s) returned	0.047 sec / 0.000 sec
4	11:21:35	SELECT DISTINCT C.Customer_name FROM Customer C JOIN IssueStatus I ON C.Customer_Id = I.Issued_cust...	1 row(s) returned	0.015 sec / 0.000 sec