



Project Description

This MySQL project is a small scale model of Library Management systems and its database name is **library** its six table are:

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

Table Attributes for each table are given as:

<u>1. Branch</u> Branch_no Set as PRIMARY KEY Manager_Id Branch_address Contact_no	<u>2. Employee</u> Emp_Id Set as PRIMARY KEY Emp_name Position Salary Branch_no Set as FOREIGN KEY and it refer <u>Branch_no</u> in Branch table	<u>3. Books</u> ISBN Set as PRIMARY KEY Book_title Category Rental_Price Status [Give yes if book available and no if book not available] Author Publisher
<u>4. Customer</u> Customer_Id Set as PRIMARY KEY Customer_name Customer_address Reg_date	<u>5. IssueStatus</u> Issue_Id Set as PRIMARY KEY Issued_cust Set as FOREIGN KEY and it refer <u>customer_id</u> in CUSTOMER table Issued_book_name Issue_date Isbn_book Set as FOREIGN KEY and it should refer isbn in <u>BOOKS</u> table	<u>6. ReturnStatus</u> 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 <u>BOOKS</u> table



Syntax

```
SELECT Book_title AS 'BOOKS',Category AS 'CATOGRY',Rental_Price AS 'PRICE' FROM Books;
```

MySQL Query Exercise

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

MySQL Workbench interface showing a query execution result.

Query Editor:

```
-- (1)Retrieving the book title,category, and Rental price of all available books.
SELECT Book_title AS 'BOOKS',Category AS 'CATOGRY',Rental_Price AS 'PRICE' FROM Books;
```

Result Grid:

BOOKS	CATOGRY	PRICE
Getting Started with MATLAB	Mechanical Engineering	30
A Mathematical Dictionary For Schools	Mathamatics	10
The Rise and Fall of the Third Reich: A History o...	History	30
1776	History	30
Quantum Mechanics	Physics	25
Configuring Windows Server 2008 Network Infr...	Information Technology	10
An Anthology Of Short Stories	English	6

Output:

#	Time	Action	Message
1	11:00:47	SELECT Book_title AS 'BOOKS',Category AS 'CATOGRY',...	22 row(s) returned



Syntax

```
SELECT Emp_name AS 'NAME', salary AS 'SALARY' FROM Employee ORDER BY salary DESC;
```

MySQL Query Exercise

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

```
169 • SELECT * FROM RETURNSTATUS;
170
171
172 -- (1)Retrieving the book title,category, and Rental price of all available books.
173 • SELECT Book_title AS 'BOOKS',Category AS 'CATOGRY',Rental_Price AS 'PRICE' FROM Book
174
175 -- (2)Listing the employee names and their respective salaries in descending order of
176 • SELECT Emp_name AS 'NAME', salary AS 'SALARY' FROM Employee ORDER BY salary DESC;
177
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	NAME	SALARY
▶	Preveen	55000
	Aravind	52000
	Balachandran	51000
	Pradeep	51000
	Umesh	51000
	Devid	50000
	Emanuel	40500

Employee 18 x

Output

Action Output

#	Time	Action	Message
✓ 1	11:00:47	SELECT Book_title AS 'BOOKS',Category AS 'CATOGRY',...	22 row(s) returned
✓ 2	11:01:21	SELECT Emp_name AS 'NAME', salary AS 'SALARY' FRO	32 row(s) returned



Syntax

```
SELECT I.issued_book_name AS 'BOOK TITLE',C.customer_name AS 'CUSTOMER  
NAME' FROM issuestatus I  
LEFT JOIN Customer C ON C.Customer_id =I.Issued_cust;
```

MySQL Query Exercise

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

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'library' selected. The main editor shows a SQL query with line numbers 173 to 183. The query is a LEFT JOIN between 'issuestatus' and 'Customer' tables. The 'Result Grid' shows the output of the query, displaying book titles and customer names. The 'Output' pane at the bottom shows the execution log with two entries: one for the 'Employee' query (32 rows returned) and one for the current 'issuestatus' query (10 rows returned).

```
173 • SELECT Book_title AS 'BOOKS',Category AS 'CATOGRY',Rental_Price AS 'PRICE' FROM Books;  
174  
175 -- (2)Listing the employee names and their respective salaries in descending order of salary  
176 • SELECT Emp_name AS 'NAME', salary AS 'SALARY' FROM Employee ORDER BY salary DESC;  
177  
178 -- (3)Retrievig the book titles and the corresponding customers who have issued those books.  
179 • SELECT I.issued_book_name AS 'BOOK TITLE',C.customer_name AS 'CUSTOMER NAME' FROM issuestatus I  
180 LEFT JOIN Customer C ON C.Customer_id =I.Issued_cust;  
181 /* Since the Primary Key on Customer table is same as Foring Key on issuestatus table  
182 we can retrive information from boath table using joins */  
183
```

BOOK TITLE	CUSTOMER NAME
Classical Mechanics Of Particles And Rigid Bodies	Balu S
Configuring Windows Server 2008 Active Direct...	Bavana
A Mathematical Dictionary For Schools	Dinakar G
Illustrated Dictionary of Mathematics	Divya M
Hindi Grammer & Composition	Fathima
Microprocessors & Microcontrollers	Faizal

Result 19 x

Output

Action Output

#	Time	Action	Message
✓ 2	11:01:21	SELECT Emp_name AS 'NAME', salary AS 'SALARY' FR...	32 row(s) returned
✓ 3	11:01:49	SELECT I.issued_book_name AS 'BOOK TITLE',C.custo...	10 row(s) returned



Syntax

```
SELECT Category AS 'CATEGORY',COUNT(Category) AS 'COUNT' FROM Books  
GROUP BY Category;
```

MySQL Query Exercise

Display the total count of books in each category.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'library' selected. The main editor shows a SQL query: `SELECT Category AS 'CATEGORY',COUNT(Category) AS 'COUNT' FROM Books GROUP BY Category;`. The 'Result Grid' at the bottom displays the query results in a table format.

CATEGORY	COUNT
Mechanical Engineering	1
Mathamatics	2
History	2
Physics	8
Information Technology	2
English	3
Chemistry	2

The 'Output' pane at the bottom shows the execution log with two entries, both indicating '9 row(s) returned'.

#	Time	Action	Message
4	11:02:55	SELECT Category,COUNT(Category) FROM Books GRO...	9 row(s) returned
5	11:04:11	SELECT Category AS 'CATEGORY',COUNT(Category) A...	9 row(s) returned



Syntax

```
SELECT Emp_name AS 'NAME',Position AS 'POSITION' FROM Employee  
WHERE Salary>50000;
```

MySQL Query Exercise

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

The screenshot shows the MySQL Workbench interface. The 'Query' tab is active, displaying the following SQL query:

```
180 • SELECT I.issued_book_name AS 'BOOK TITLE',C.customer_name AS 'CUSTOMER NAME' FROM issuestatus I  
181 LEFT JOIN Customer C ON C.Customer_id =I.Issued_cust;  
182 /* Since the Primary Key on Customer table is same as Foring Key on issuestatus table  
183 we can retrive information from boath table using joins */  
184  
185 -- (4) Displaying tottal count of book in each catogory  
186 • SELECT Category AS 'CATEGORY',COUNT(Category) AS 'COUNT' FROM Books GROUP BY Category;  
187  
188 /* (5) Retrieving the employee name and their posistions for the employees whose  
189 salaries are above Rs.50,000 */  
190 • SELECT Emp_name AS 'NAME',Position AS 'POSITION' FROM Employee WHERE Salary>50000;
```

The 'Result Grid' shows the output of the query:

	NAME	POSITION
▶	Aravind	Librarian
	Balachandran	Manager
	Preveen	Librarian
	Pradeep	Manager
	Umesh	Manager

The 'Output' tab shows the execution details:

#	Time	Action	Message
✓ 1	21:20:42	SELECT Emp_name AS 'NAME',Position AS 'POSITION' FROM Employee WHERE Salary>50...	5 row(s) returned



Syntex

```
SELECT Customer_name AS 'NAME OF NOT ISSUED CUSTOMER',Reg_date AS 'REGISTERD DATE'  
FROM customer WHERE Reg_date <= '2022-01-01'  
AND NOT EXISTS (SELECT return_book_name FROM returnstatus WHERE return_cust = customer.customer_Id);
```

MySQL Query Exercise

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

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'SCHEMAS' tree with 'library' selected. The main editor shows a SQL query with line numbers 187 to 197. The query is:
187
188 /* (5) Retriving the employee name and their posistions for the employees whose
189 salaries are above Rs.50,000 */
190 • SELECT Emp_name AS 'NAME',Position AS 'POSITION' FROM Employee WHERE Salary>50000;
191
192 /* (6)Listing the customer names who registered before 2022-01-01 and have not issued any books yet */
193 • SELECT Customer_name AS 'NAME OF NOT ISSUED CUSTOMER',Reg_date AS 'REGISTERD DATE'
194 FROM customer WHERE Reg_date <= '2022-01-01'
195 AND NOT EXISTS (SELECT return_book_name FROM returnstatus WHERE return_cust = customer.customer_Id);
196 /* For date referance extra query steps included for reg_date colum and retruns as REGISTERD DATE */
197
The 'Result Grid' shows the output of the query, with columns 'NAME OF NOT ISSUED CUSTOMER' and 'REGISTERD DATE'. The results are:

NAME OF NOT ISSUED CUSTOMER	REGISTERD DATE
Ajeesh Kumar	1955-10-22
Ashok R	1988-05-03
Benny T	1979-04-25
Balu S	1998-03-08
Bavana	1992-12-08
Charulatha V	1963-01-27

 The 'Output' pane at the bottom shows the execution log, indicating that the query was successful and returned 13 rows.



Syntax

```
SELECT Branch_no AS 'BRANCH',COUNT(Emp_id) AS 'TOTLA NUMBER OF  
EMPLOYEE' FROM employee  
GROUP BY Branch_no;
```

MySQL Query Exercise

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

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

- Navigator:** Shows a tree view of schemas. The 'library' schema is selected, showing 'Tables', 'Views', and 'Stored Procedures'.
- Query Editor:** Contains the following SQL query:

```
190 • SELECT Emp_name AS 'NAME',Position AS 'POSITION' FROM Employee WHERE Salary>50000;  
191  
192 /* (6)Listing the customer names who registered before 2022-01-01 and have not issued any books yet */  
193 • SELECT Customer_name AS 'NAME OF NOT ISSUED CUSTOMER',Reg_date AS 'REGISTERD DATE'  
194 FROM customer WHERE Reg_date <= '2022-01-01'  
195 AND NOT EXISTS (SELECT return_book_name FROM returnstatus WHERE return_cust = customer.customer_Id);  
196 /* For date reference extra query steps included for reg_date colum an retruns as REGISTERD DATE */  
197  
198 -- (7)Displaying the branch numbers and the total count of employees in each branch  
199 • SELECT Branch_no AS 'BRANCH',COUNT(Emp_id) AS 'TOTLA NUMBER OF EMPLOYEE' FROM employee  
200 GROUP BY Branch_no;
```
- Result Grid:** Displays the results of the query in a table with two columns: 'BRANCH' and 'TOTLA NUMBER OF EMPLOYEE'.

BRANCH	TOTLA NUMBER OF EMPLOYEE
101	3
102	3
103	6
104	3
105	3
106	3
- Output:** Shows the execution log with two entries:
 - 5 17:32:40 SELECT Customer_name AS 'CUSTOMER NAME' FROM customer WHERE EXISTS(SELE... 1 row(s) returned
 - 6 17:33:48 SELECT Branch_no AS 'BRANCH',COUNT(Emp_id) AS 'TOTLA NUMBER OF EMPLOYEE' ... 8 row(s) returned



Syntex

```
SELECT Customer_name AS 'CUSTOMER NAME' FROM customer WHERE  
EXISTS(SELECT Issued_cust FROM IssueStatus WHERE  
Issued_cust = customer.customer_Id AND YEAR(Issue_date)=2023 AND MONTH(Issue_date)=06);
```

MySQL Query Exercise

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

The screenshot shows the MySQL Workbench interface. The 'Navigator' pane on the left displays the 'library' database schema, including tables, views, and stored procedures. The 'Query' pane shows the following SQL query:

```
195 AND NOT EXISTS (SELECT return_book_name FROM returnstatus WHERE return_cust = customer.customer_Id);  
196 /* For date reference extra query steps included for reg_date column an retruns as REGISTERED DATE */  
197  
198 -- (7)Displaying the branch numbers and the total count of employees in each branch  
199 • SELECT Branch_no AS 'BRANCH',COUNT(Emp_id) AS 'TOTLA NUMBER OF EMPLOYEE' FROM employee  
200 GROUP BY Branch_no;  
201  
202 -- (8)Displaying the names of customers who have issued books in the month of june 2023  
203 • SELECT Customer_name AS 'CUSTOMER NAME' FROM customer WHERE  
204 EXISTS(SELECT Issued_cust FROM IssueStatus WHERE  
205 Issued_cust = customer.customer_Id AND YEAR(Issue_date)=2023 AND MONTH(Issue_date)=06);
```

The 'Result Grid' pane shows the results of the query, displaying a single row with the customer name 'Fathima'.

CUSTOMER NAME
Fathima

The 'Output' pane shows the 'Action Output' table, which displays the execution details of the query:

#	Time	Action	Message
4	17:31:11	INSERT INTO IssueStatus(Issue_Id,Issued_cust,Issued_book_name,Issue_date,lsbn_book)...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0
5	17:32:40	SELECT Customer name AS 'CUSTOMER NAME' FROM customer WHERE EXISTS(SELE...	1 row(s) returned



Syntax

```
SELECT Book_title AS 'HISTORY BOOKS' FROM Books WHERE Category = 'history';
```

MySQL Query Exercise

Retrieve book_title from book table containing history.

The screenshot displays the MySQL Workbench environment. The 'Navigator' pane on the left shows the 'library' schema selected, with 'Tables' expanded. The 'Query' tab is active, showing a SQL script with three queries. The third query, starting at line 207, is selected and highlighted:

```
-- (9) Retrieving book_title from book table containing history
207
208 • SELECT Book_title AS 'HISTORY BOOKS' FROM Books WHERE Category = 'history';
```

Below the query editor, the 'Result Grid' shows the output of the selected query:

	HISTORY BOOKS
▶	The Rise and Fall of the Third Reich: A History o...
	1776

The 'Output' pane at the bottom shows the 'Action Output' for the query, indicating that 2 row(s) were returned.

#	Time	Action	Message
✓ 10	18:56:49	SELECT Branch_no AS 'BRANCH NUMBER', COUNT(Emp_name) AS 'COUNT OF EMPL...	2 row(s) returned
✓ 11	18:57:31	SELECT Book_title AS 'HISTORY BOOKS' FROM Books WHERE Category = 'history' LIMIT...	2 row(s) returned



Syntax

```
SELECT Branch_no AS 'BRANCH NUMBER', COUNT(Emp_name) AS 'COUNT OF EMPLOYEE'  
FROM employee GROUP BY branch_no HAVING count(Emp_name)>5;
```

MySQL Query Exercise

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

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

```
203 SELECT Customer_name AS 'CUSTOMER NAME' FROM customer WHERE  
204 EXISTS(SELECT Issued_cust FROM IssueStatus WHERE  
205 Issued_cust = customer.customer_Id AND YEAR(Issue_date)=2023 AND MONTH(Issue_date)=06);  
206  
207 -- (9) Retrieving book_title from book table containing history  
208 SELECT Book_title AS 'HISTORY BOOKS' FROM Books WHERE Category = 'history';  
209  
210 /*(10) Retrieving the branch numbers along with the count of  
211 employees for braches having more than 5 employee*/  
212 SELECT Branch_no AS 'BRANCH NUMBER', COUNT(Emp_name) AS 'COUNT OF EMPLOYEE'  
213 FROM employee GROUP BY branch_no HAVING count(Emp_name)>5;
```

The results of the query are displayed in the Result Grid:

	BRANCH NUMBER	COUNT OF EMPLOYEE
▶	103	6
	108	6

The bottom of the screenshot shows the Output pane with the following messages:

#	Time	Action	Message
9	17:39:35	SELECT Book_title AS 'HISTORY BOOKS' FROM Books WHERE Category = 'history' LIMIT...	2 row(s) returned
10	18:56:49	SELECT Branch_no AS 'BRANCH NUMBER', COUNT(Emp_name) AS 'COUNT OF EMPL...	2 row(s) returned



Syntax

```
SELECT E.emp_name AS 'MANGER NAME', B.brach_address AS 'BRACH ASSRESS'  
FROM employee E  
LEFT JOIN branch B ON B.Branch_no = E.Branch_no WHERE E.Position = 'Manager';
```

MySQL Query Exercise

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

The screenshot shows the MySQL Query Editor interface. The left sidebar displays the 'SCHEMAS' tree with the 'library' database selected. The main window shows a SQL query being executed. The query is as follows:

```
208 SELECT Book_title AS 'HISTORY BOOKS' FROM Books WHERE Category = 'history';  
209  
210 /*(10) Retrieving the branch numbers along with the count of  
211 employees for braches having more than 5 employee*/  
212 SELECT Branch_no AS 'BRANCH NUMBER', COUNT(Emp_name) AS 'COUNT OF EMPLOYEE'  
213 FROM employee GROUP BY branch_no HAVING count(Emp_name)>5;  
214  
215 -- (11) Retrieving the names of employees who manage branches and their respective branch addresses  
216 SELECT E.emp_name AS 'MANGER NAME', B.brach_address AS 'BRACH ASSRESS'  
217 FROM employee E  
218 LEFT JOIN branch B ON B.Branch_no = E.Branch_no WHERE E.Position = 'Manager';
```

The 'Result Grid' shows the output of the query:

MANGER NAME	BRACH ASSRESS
Balachandran	E Padmnabhan Memorial Library Vinayaka Colony Palakkad - 678014...
Emanuel	Kakkanni Vikasana Samithi & APJ Abdulkalam Memorial Library 40, K...
Hakeem	Kala Samskarika Samithi Library Ithingaparambu Street, Ithingapara...
Kannan	Palakkad Medical College Library Government Medical College, Palak...
Narayanan	Koppam Paurasamgham Vayanasala & Library Puthur Road, Shesha...
Pradeep	BHSS Library Sahyadri Colony, Chandranagar Colony Palakkad - 678...
Shameer	Thondikulam Public Library 276 Single Street New Extension, Thon...

The 'Output' section shows the execution message:

```
# Time Action Message  
1 21:18:56 SELECT E.emp_name AS'MANGER NAME',B.brach_address AS'BRACH ASSRESS' FROM e... 8 row(s) returned
```




Syntex

```
SELECT customer_name AS 'CUSTOMER NAME', Rental_Price AS 'PRICE' FROM IssueStatus  
INNER JOIN customer ON customer.customer_id= IssueStatus.Issued_cust  
INNER JOIN books ON IssueStatus.isbn_book = books.isbn WHERE Rental_Price>25;
```

MySQL Query Exercise

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

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

- Navigator:** Shows a tree view of schemas. The 'library' schema is selected, showing tables, views, and stored procedures.
- Query Editor:** Contains the SQL query:

```
213 FROM employee GROUP BY branch_no HAVING count(Emp_name)>5;  
214  
215 -- (11) Retrieving the names of employees who manage branches and their respective branch addresses  
216 • SELECT E.emp_name AS'MANGER NAME',B.brach_address AS'BRACH ASSRESS'  
217 FROM employee E  
218 LEFT JOIN branch B ON B.Branch_no = E.Branch_no WHERE E.Position = 'Manager';  
219  
220 -- (12) Displaying the names of customers who have issued books with a rental price higher than Rs.25  
221 • SELECT customer_name AS'CUSTOMER NAME', Rental_Price AS 'PRICE' FROM IssueStatus  
222 INNER JOIN customer ON customer.customer_id= IssueStatus.Issued_cust  
223 INNER JOIN books ON IssueStatus.isbn_book = books.isbn WHERE Rental_Price>25;
```
- Result Grid:** Displays the results of the query in a table:

	CUSTOMER NAME	PRICE
▶	Lavanya A	30
	Balu S	28
	Hameedh	26
- Output:** Shows the execution log with two entries:

#	Time	Action	Message
✓ 12	20:26:25	SELECT E.emp_name AS'MANGER NAME',B.brach_address AS'BRACH ASSRESS' FROM...	8 row(s) returned
✓ 13	21:17:51	SELECT customer_name AS'CUSTOMER NAME', Rental_Price AS 'PRICE' FROM IssueSta...	3 row(s) returned

THANK YOU

*By
Sreepathap*



[linkedin.com/in/sreepathap](https://www.linkedin.com/in/sreepathap)



github.com/sreepathap