Mysql Comprehensive Assessment

Topic : Library Management System  
You are going to build a project based on Library Management System. It keeps track of all information about books in the library, their cost, status and total number of books available in the library

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

create DATABASE library ;

USE library;

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

CREATE TABLE Branch(

Branch\_no INT NOT NULL auto\_increment,

Manager\_Id INT NOT NULL,

Branch\_address VARCHAR(100) NOT NULL,

Contact\_no VARCHAR(20) NOT NULL,

primary key(Branch\_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

CREATE TABLE Employee(

Emp\_Id INT NOT NULL auto\_increment,

Emp\_name VARCHAR(50) NOT NULL,

Position VARCHAR(50),

Salary DECIMAL(10,2),

Branch\_no INT,

primary key(Emp\_Id),

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 NOT NULL auto\_increment,

Book\_title VARCHAR(50) NOT NULL,

Category VARCHAR(50) NOT NULL,

Rental\_Price DECIMAL(10,2) NOT NULL,

Status VARCHAR(10) NOT NULL COMMENT 'Give yes if book available and no if book not available',

Author varchar(50) NOT NULL,

Publisher varchar(50) NOT NULL,

primary key(ISBN)

);

4. Customer    
  
• Customer\_Id Set as PRIMARY KEY    
• Customer\_name    
• Customer\_address    
• Reg\_date

create table Customer(

Customer\_Id int not null auto\_increment,

Customer\_name varchar(50) not null,

Customer\_address varchar(100) not null,

Reg\_date date not null,

primary key(Customer\_Id)

);

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 not null auto\_increment,

Issued\_cust int not null,

Issue\_date date not null,

Isbn\_book int not null,

primary key(Issue\_Id),

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 not null auto\_increment,

Return\_cust int,

Return\_book\_name varchar(50),

Return\_date date not null,

Isbn\_book2 int not null,

primary key(Return\_Id),

foreign key(Isbn\_book2) references Books(ISBN),

foreign key(Return\_cust) references Customer(Customer\_Id)

);

**Insert values into the table**

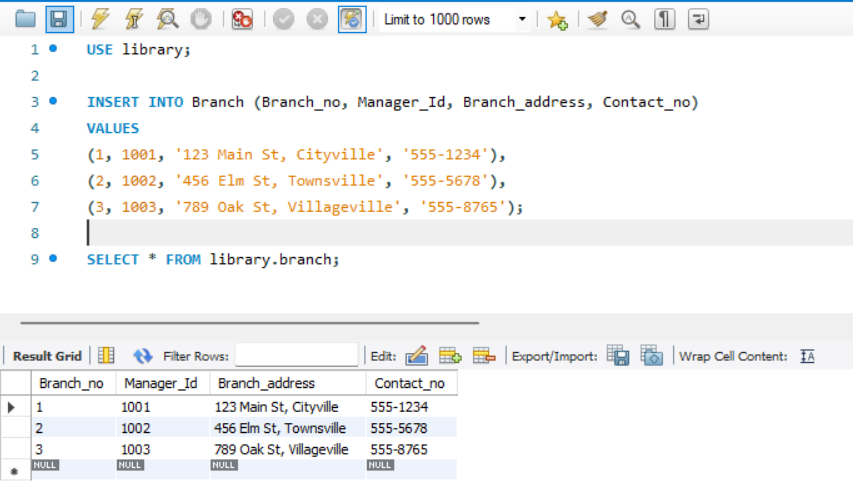
INSERT INTO Branch (Branch\_no, Manager\_Id, Branch\_address, Contact\_no)

VALUES

(1, 1001, '123 Main St, Cityville', '555-1234'),

(2, 1002, '456 Elm St, Townsville', '555-5678'),

(3, 1003, '789 Oak St, Villageville', '555-8765');



INSERT INTO Employee (Emp\_Id, Emp\_name, Position, Salary, Branch\_no)

VALUES

(1, 'Alice Smith', 'Manager', 60000, 1),

(2, 'Bob Johnson', 'Assistant', 40000, 1),

(3, 'Carol White', 'Manager', 65000, 2),

(4, 'David Brown', 'Clerk', 35000, 2),

(5, 'Eve Davis', 'Assistant', 42000, 3),

(6, 'Frank Miller', 'Clerk', 37000, 3),

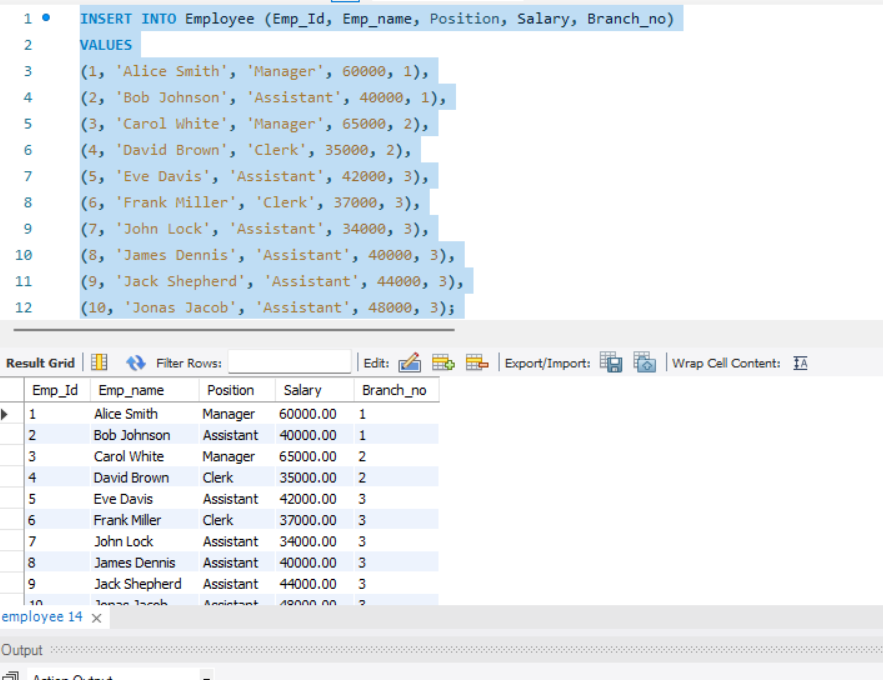
(7, 'John Lock', 'Assistant', 34000, 3),

(8, 'James Dennis', 'Assistant', 40000, 3),

(9, 'Jack Shepherd', 'Assistant', 44000, 3),

(10, 'Jonas Jacob', 'Assistant', 48000, 3);

SELECT \* FROM employee;



INSERT INTO Books (Book\_title, Category, Rental\_Price, Status, Author, Publisher)

VALUES

('The Great Gatsby', 'Fiction', 30.00, 'yes', 'F. Scott Fitzgerald', 'Scribner'),

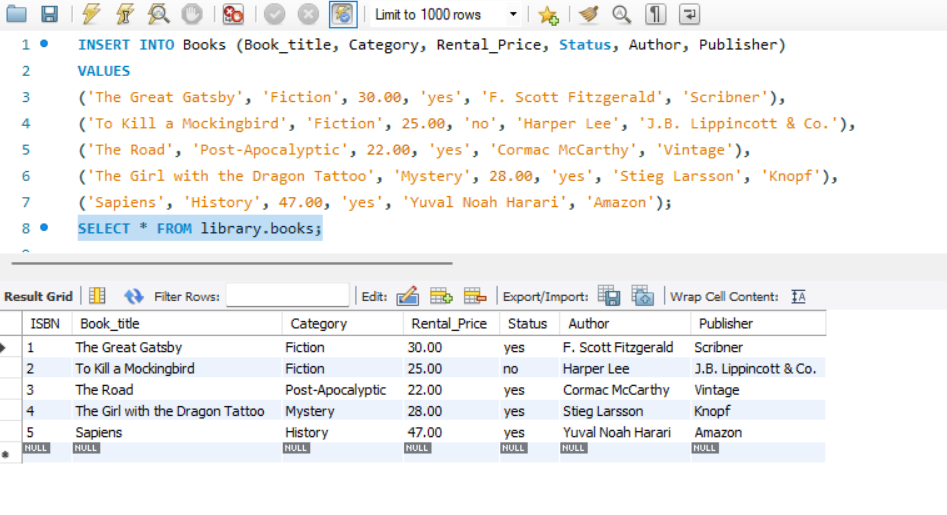
('To Kill a Mockingbird', 'Fiction', 25.00, 'no', 'Harper Lee', 'J.B. Lippincott & Co.'),

('The Road', 'Post-Apocalyptic', 22.00, 'yes', 'Cormac McCarthy', 'Vintage'),

('The Girl with the Dragon Tattoo', 'Mystery', 28.00, 'yes', 'Stieg Larsson', 'Knopf'),

('Sapiens', 'History', 47.00, 'yes', 'Yuval Noah Harari', 'Amazon');

SELECT \* FROM library.books;



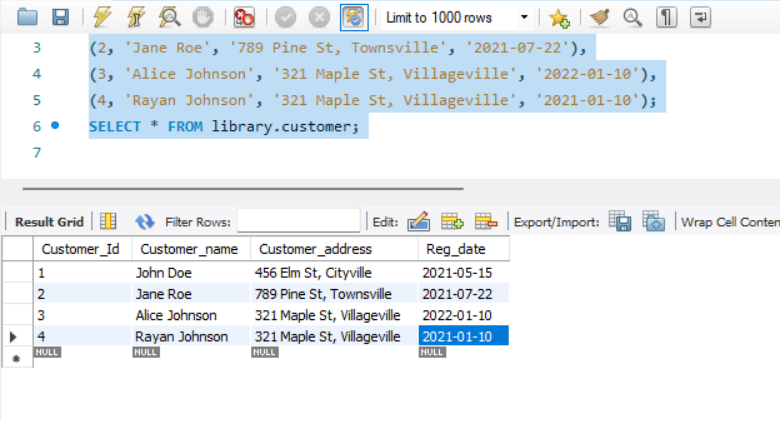
INSERT INTO Customer (Customer\_Id, Customer\_name, Customer\_address, Reg\_date)

VALUES (1, 'John Doe', '456 Elm St, Cityville', '2021-05-15'),

(2, 'Jane Roe', '789 Pine St, Townsville', '2021-07-22'),

(3, 'Alice Johnson', '321 Maple St, Villageville', '2022-01-10'),

(4, 'Rayan Johnson', '321 Maple St, Villageville', '2021-01-10');

SELECT \* FROM library.customer; 

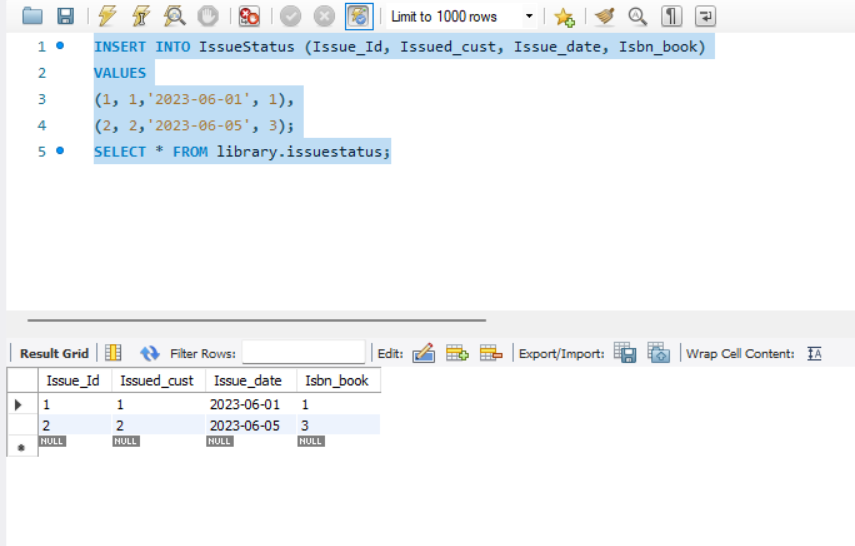
INSERT INTO IssueStatus (Issue\_Id, Issued\_cust, Issue\_date, Isbn\_book)

VALUES

(1, 1,'2023-06-01', 1),

(2, 2,'2023-06-05', 3);

SELECT \* FROM library.issuestatus;



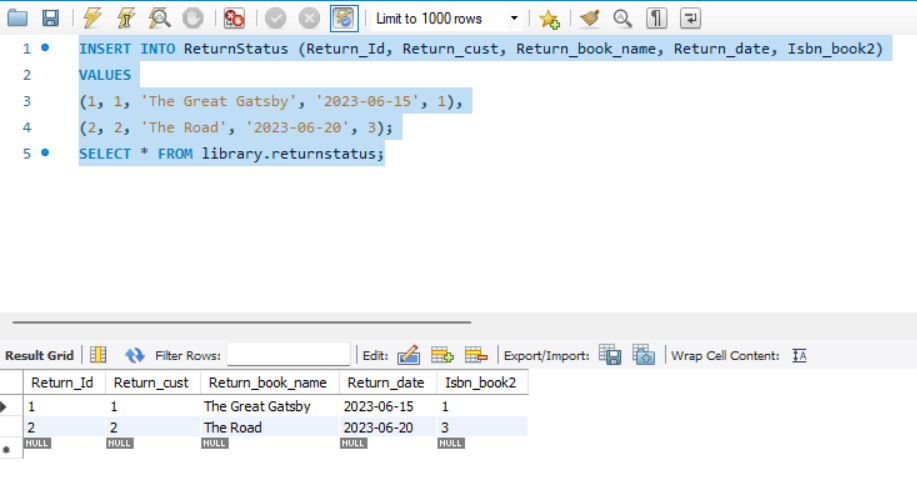
INSERT INTO ReturnStatus (Return\_Id, Return\_cust, Return\_book\_name, Return\_date, Isbn\_book2)

VALUES

(1, 1, 'The Great Gatsby', '2023-06-15', 1),

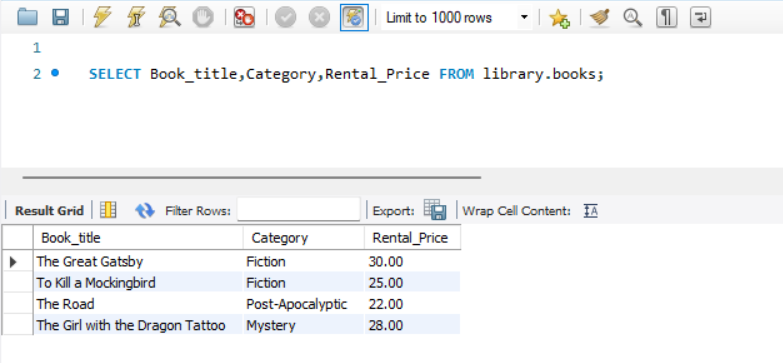
(2, 2, 'The Road', '2023-06-20', 3);

SELECT \* FROM library.returnstatus;



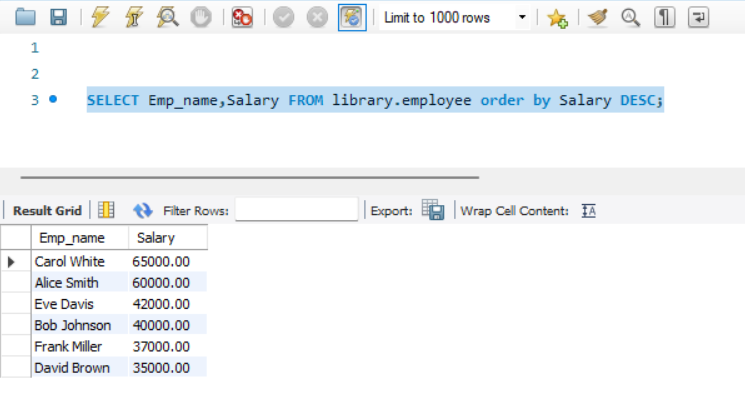
1. **Retrieve the book title, category, and rental price of all available books**.

SELECT Book\_title,Category,Rental\_Price FROM library.books;



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

SELECT Emp\_name,Salary FROM library.employee order by Salary DESC;

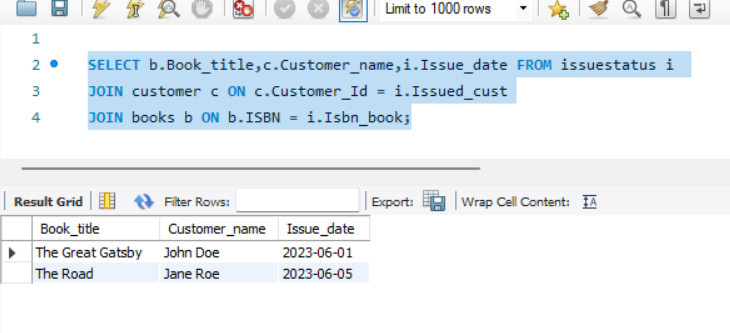


1. **Retrieve the book titles and the corresponding customers who have issued those books.**

SELECT b.Book\_title,c.Customer\_name,i.Issue\_date FROM issuestatus i

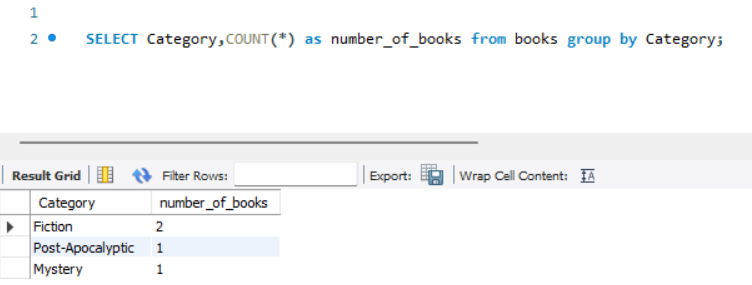
JOIN customer c ON c.Customer\_Id = i.Issued\_cust

JOIN books b ON b.ISBN = i.Isbn\_book;



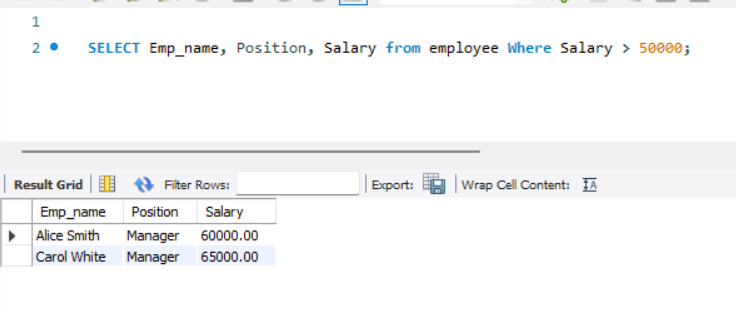
**4. Display the total count of books in each category.**

SELECT Category,COUNT(\*) as number\_of\_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, Salary from employee Where Salary > 50000;

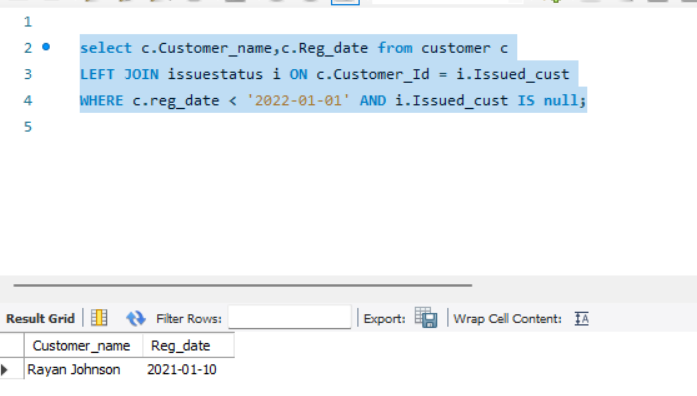


**6. List the customer names who registered before 2022-01-01 and have not issued any books yet.**

select c.Customer\_name,c.Reg\_date from customer c

LEFT JOIN issuestatus i ON c.Customer\_Id = i.Issued\_cust

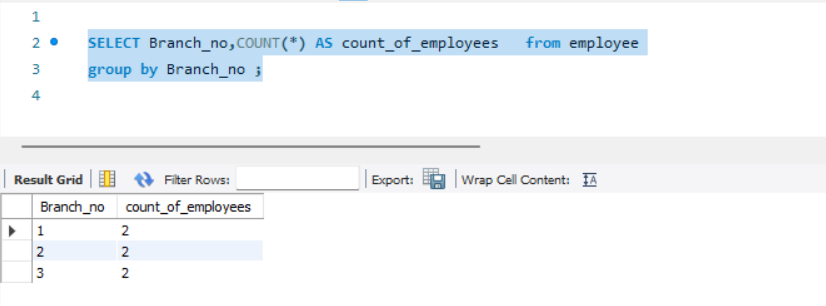
WHERE c.reg\_date < '2022-01-01' AND i.Issued\_cust IS null;



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

SELECT Branch\_no,COUNT(\*) AS count\_of\_employees from employee

group by Branch\_no ;

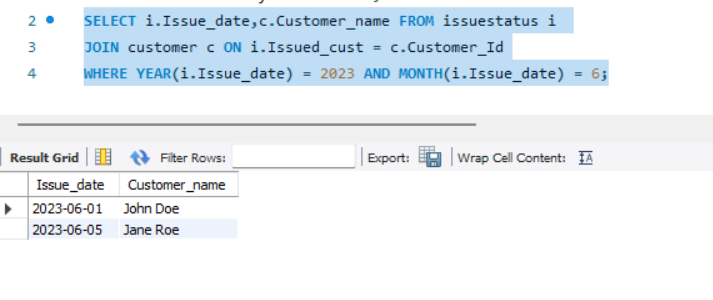


**8. Display the names of customers who have issued books in the month of June 2023**.

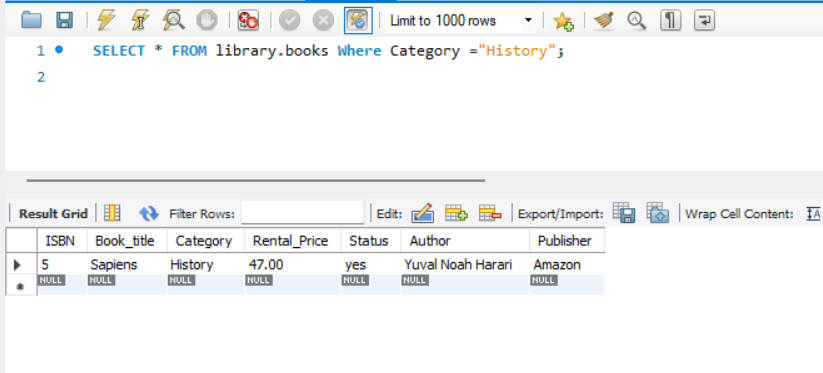
SELECT i.Issue\_date,c.Customer\_name FROM issuestatus i

JOIN customer c ON i.Issued\_cust = c.Customer\_Id

WHERE YEAR(i.Issue\_date) = 2023 AND MONTH(i.Issue\_date) = 6;



**9. Retrieve book\_title from book table containing history**.

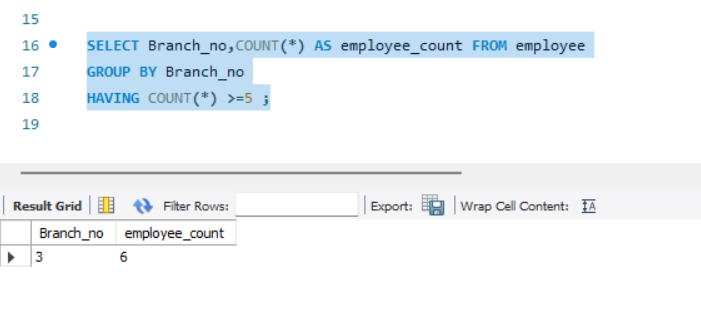


**10.Retrieve the branch numbers along with the count of employees for branches having more than 5 employees**

SELECT Branch\_no,COUNT(\*) AS employee\_count FROM employee

GROUP BY Branch\_no

HAVING COUNT(\*) >=5 ;

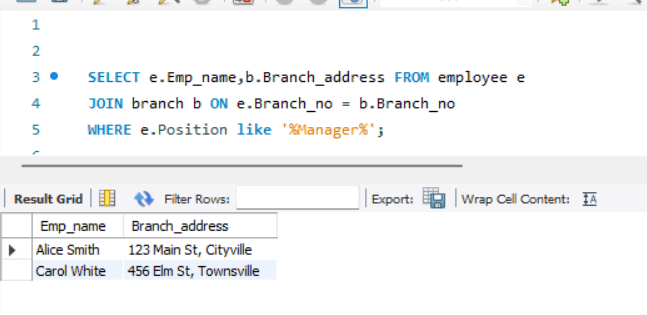
****

**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.Branch\_no = b.Branch\_no

WHERE e.Position like '%Manager%';



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

SELECT c.Customer\_name,b.Rental\_Price

FROM customer c

JOIN issuestatus i ON i.Issued\_cust = c.Customer\_Id

JOIN books b ON b.ISBN = i.Isbn\_book

WHERE b.Rental\_Price > 25 ;

