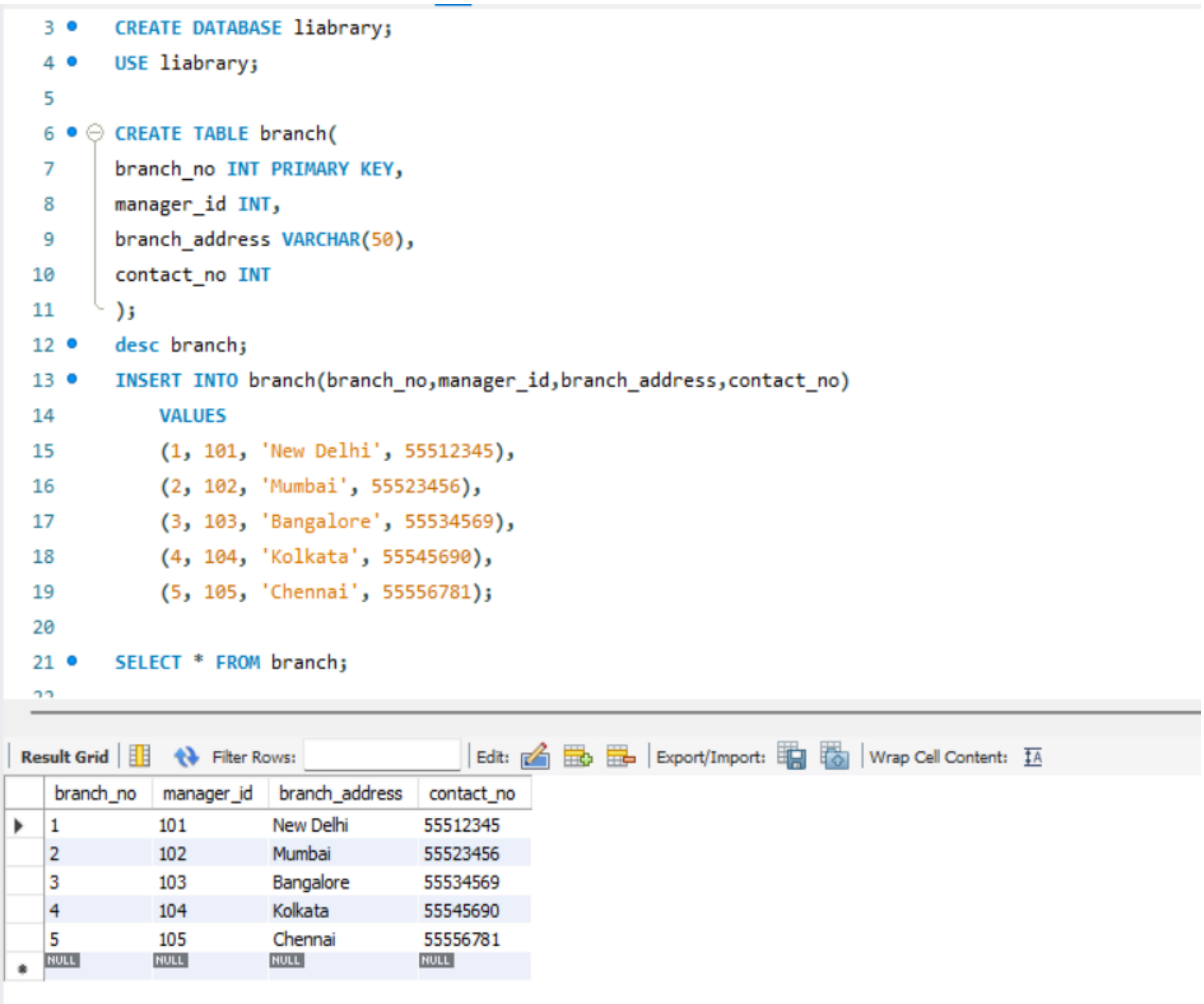
MySQL- Library Management system

Create a database named library and following TABLES in the database:
1. Branch
2. Employee
3. Books
4. Customer
5. IssueStatus
5. ReturnStatus

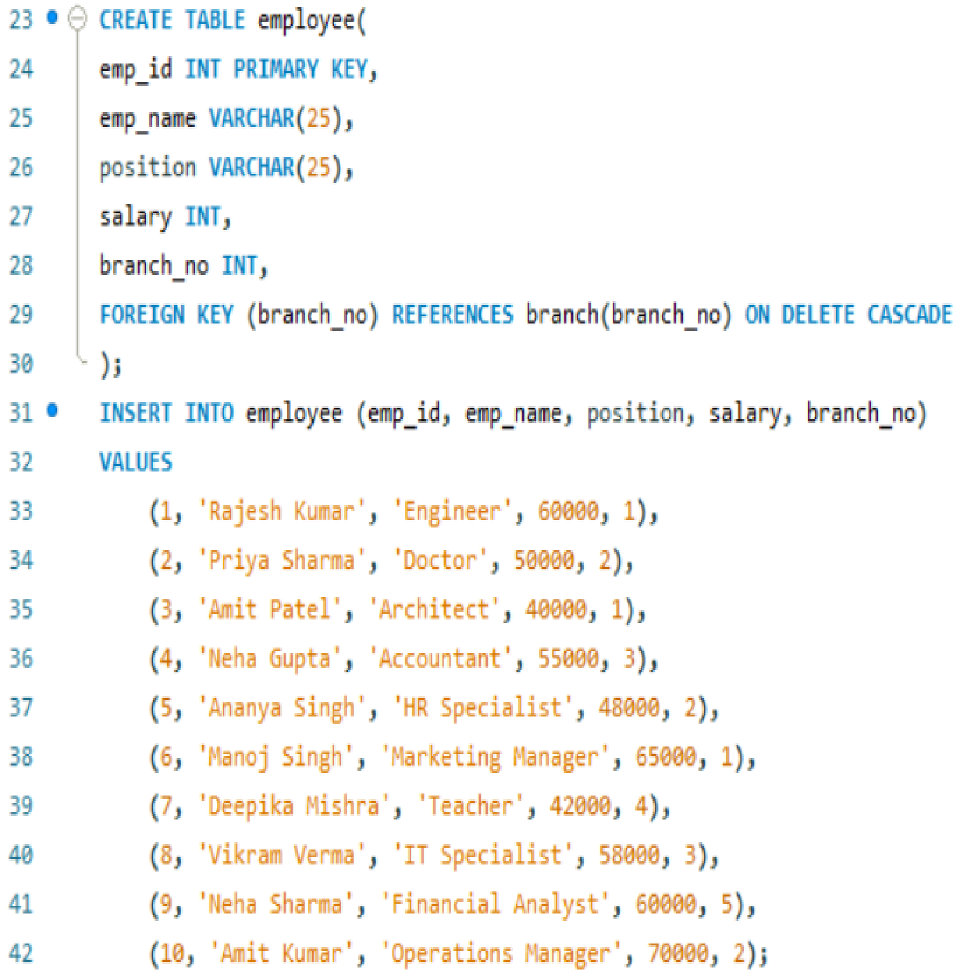
1. Branch
-Branch_no - Set as PRIMARY KEY
-Manager_Id
-Branch_address
-Contact_no



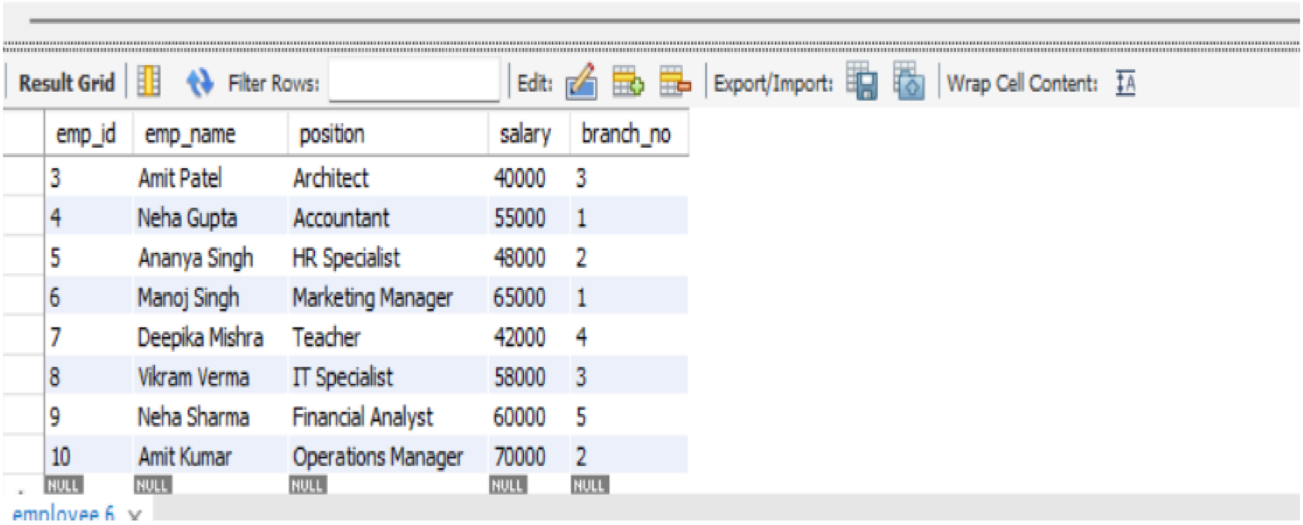
```sql
CREATE DATABASE liabrary;
USE liabrary;

CREATE TABLE branch(
branch_no INT PRIMARY KEY,
manager_id INT,
branch_address VARCHAR(50),
contact_no INT
);
desc branch;
INSERT INTO branch(branch_no,manager_id,branch_address,contact_no)
    VALUES
    (1, 101, 'New Delhi', 55512345),
    (2, 102, 'Mumbai', 55523456),
    (3, 103, 'Bangalore', 55534569),
    (4, 104, 'Kolkata', 55545690),
    (5, 105, 'Chennai', 55556781);

SELECT * FROM branch;
```

branch_no	manager_id	branch_address	contact_no
1	101	New Delhi	55512345
2	102	Mumbai	55523456
3	103	Bangalore	55534569
4	104	Kolkata	55545690
5	105	Chennai	55556781
NULL	NULL	NULL	NULL

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

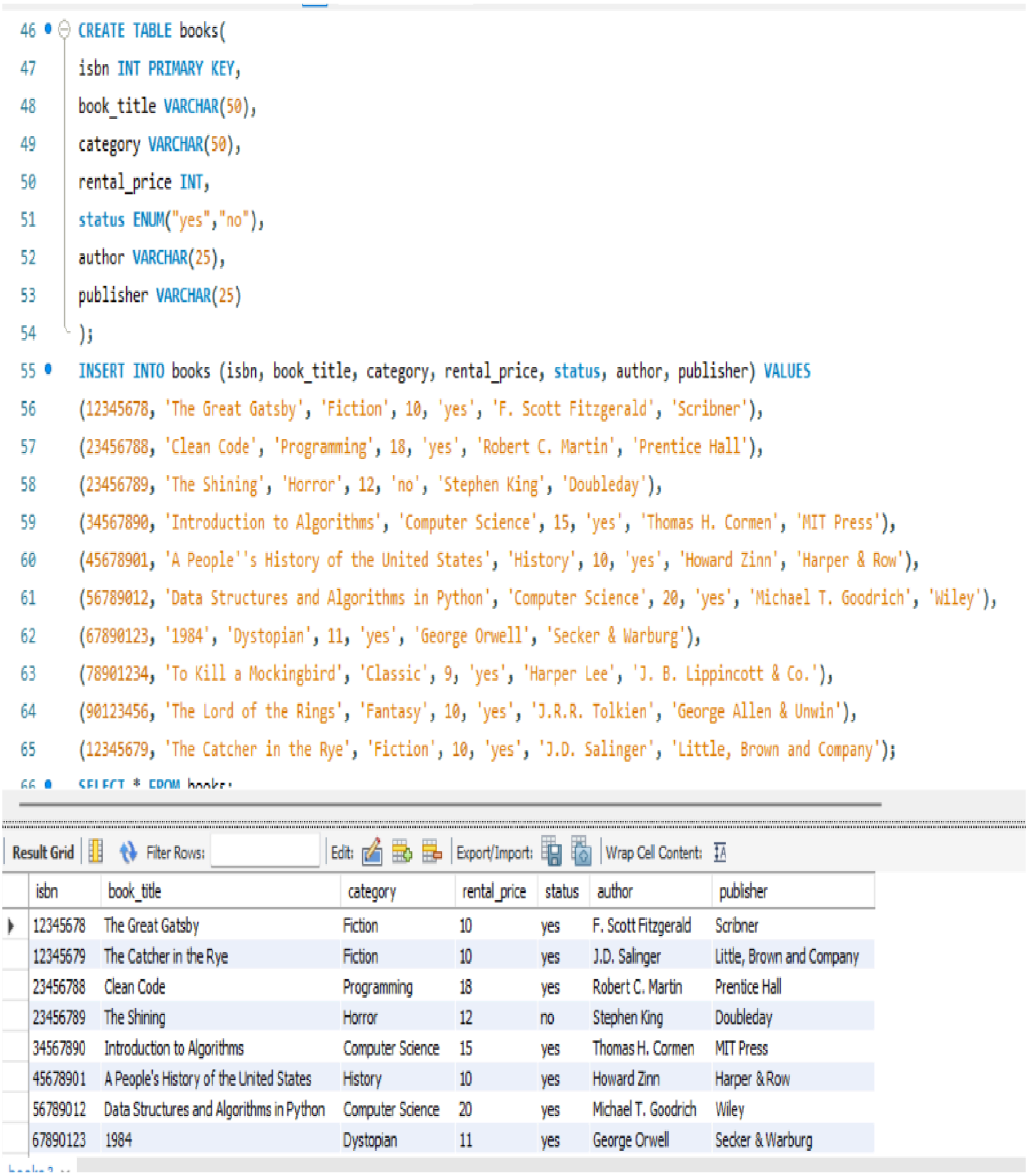


```sql
CREATE TABLE employee(
emp_id INT PRIMARY KEY,
emp_name VARCHAR(25),
position VARCHAR(25),
salary INT,
branch_no INT,
FOREIGN KEY (branch_no) REFERENCES branch(branch_no) ON DELETE CASCADE
);
INSERT INTO employee (emp_id, emp_name, position, salary, branch_no)
VALUES
    (1, 'Rajesh Kumar', 'Engineer', 60000, 1),
    (2, 'Priya Sharma', 'Doctor', 50000, 2),
    (3, 'Amit Patel', 'Architect', 40000, 1),
    (4, 'Neha Gupta', 'Accountant', 55000, 3),
    (5, 'Ananya Singh', 'HR Specialist', 48000, 2),
    (6, 'Manoj Singh', 'Marketing Manager', 65000, 1),
    (7, 'Deepika Mishra', 'Teacher', 42000, 4),
    (8, 'Vikram Verma', 'IT Specialist', 58000, 3),
    (9, 'Neha Sharma', 'Financial Analyst', 60000, 5),
    (10, 'Amit Kumar', 'Operations Manager', 70000, 2);
```



emp_id	emp_name	position	salary	branch_no
3	Amit Patel	Architect	40000	3
4	Neha Gupta	Accountant	55000	1
5	Ananya Singh	HR Specialist	48000	2
6	Manoj Singh	Marketing Manager	65000	1
7	Deepika Mishra	Teacher	42000	4
8	Vikram Verma	IT Specialist	58000	3
9	Neha Sharma	Financial Analyst	60000	5
10	Amit Kumar	Operations Manager	70000	2
NULL	NULL	NULL	NULL	NULL

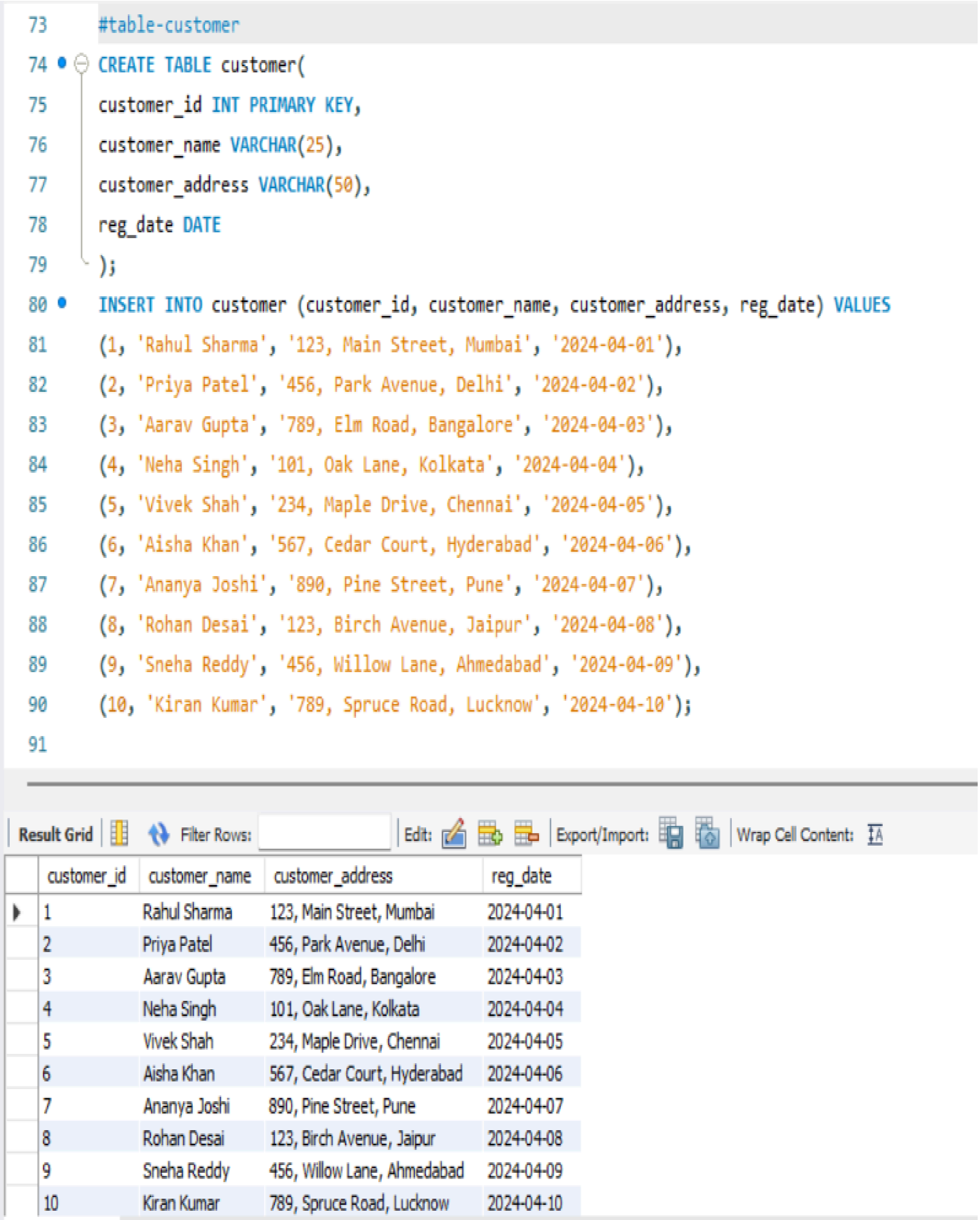
3. Books
-ISBN - Set as PRIMARY KEY
-Book_title
-Category
-Rental_Price
-Status [Give yes if book available and no if book not available]
-Author
-Publisher



```
CREATE TABLE books(
isbn INT PRIMARY KEY,
book_title VARCHAR(50),
category VARCHAR(50),
rental_price INT,
status ENUM("yes","no"),
author VARCHAR(25),
publisher VARCHAR(25)
);
INSERT INTO books (isbn, book_title, category, rental_price, status, author, publisher) VALUES
(12345678, 'The Great Gatsby', 'Fiction', 10, 'yes', 'F. Scott Fitzgerald', 'Scribner'),
(23456788, 'Clean Code', 'Programming', 18, 'yes', 'Robert C. Martin', 'Prentice Hall'),
(23456789, 'The Shining', 'Horror', 12, 'no', 'Stephen King', 'Doubleday'),
(34567890, 'Introduction to Algorithms', 'Computer Science', 15, 'yes', 'Thomas H. Cormen', 'MIT Press'),
(45678901, 'A People''s History of the United States', 'History', 10, 'yes', 'Howard Zinn', 'Harper & Row'),
(56789012, 'Data Structures and Algorithms in Python', 'Computer Science', 20, 'yes', 'Michael T. Goodrich', 'Wiley'),
(67890123, '1984', 'Dystopian', 11, 'yes', 'George Orwell', 'Secker & Warburg'),
(78901234, 'To Kill a Mockingbird', 'Classic', 9, 'yes', 'Harper Lee', 'J. B. Lippincott & Co.'),
(90123456, 'The Lord of the Rings', 'Fantasy', 10, 'yes', 'J.R.R. Tolkien', 'George Allen & Unwin'),
(12345679, 'The Catcher in the Rye', 'Fiction', 10, 'yes', 'J.D. Salinger', 'Little, Brown and Company');
SELECT * FROM books;
```

isbn	book_title	category	rental_price	status	author	publisher
12345678	The Great Gatsby	Fiction	10	yes	F. Scott Fitzgerald	Scribner
12345679	The Catcher in the Rye	Fiction	10	yes	J.D. Salinger	Little, Brown and Company
23456788	Clean Code	Programming	18	yes	Robert C. Martin	Prentice Hall
23456789	The Shining	Horror	12	no	Stephen King	Doubleday
34567890	Introduction to Algorithms	Computer Science	15	yes	Thomas H. Cormen	MIT Press
45678901	A People's History of the United States	History	10	yes	Howard Zinn	Harper & Row
56789012	Data Structures and Algorithms in Python	Computer Science	20	yes	Michael T. Goodrich	Wiley
67890123	1984	Dystopian	11	yes	George Orwell	Secker & Warburg

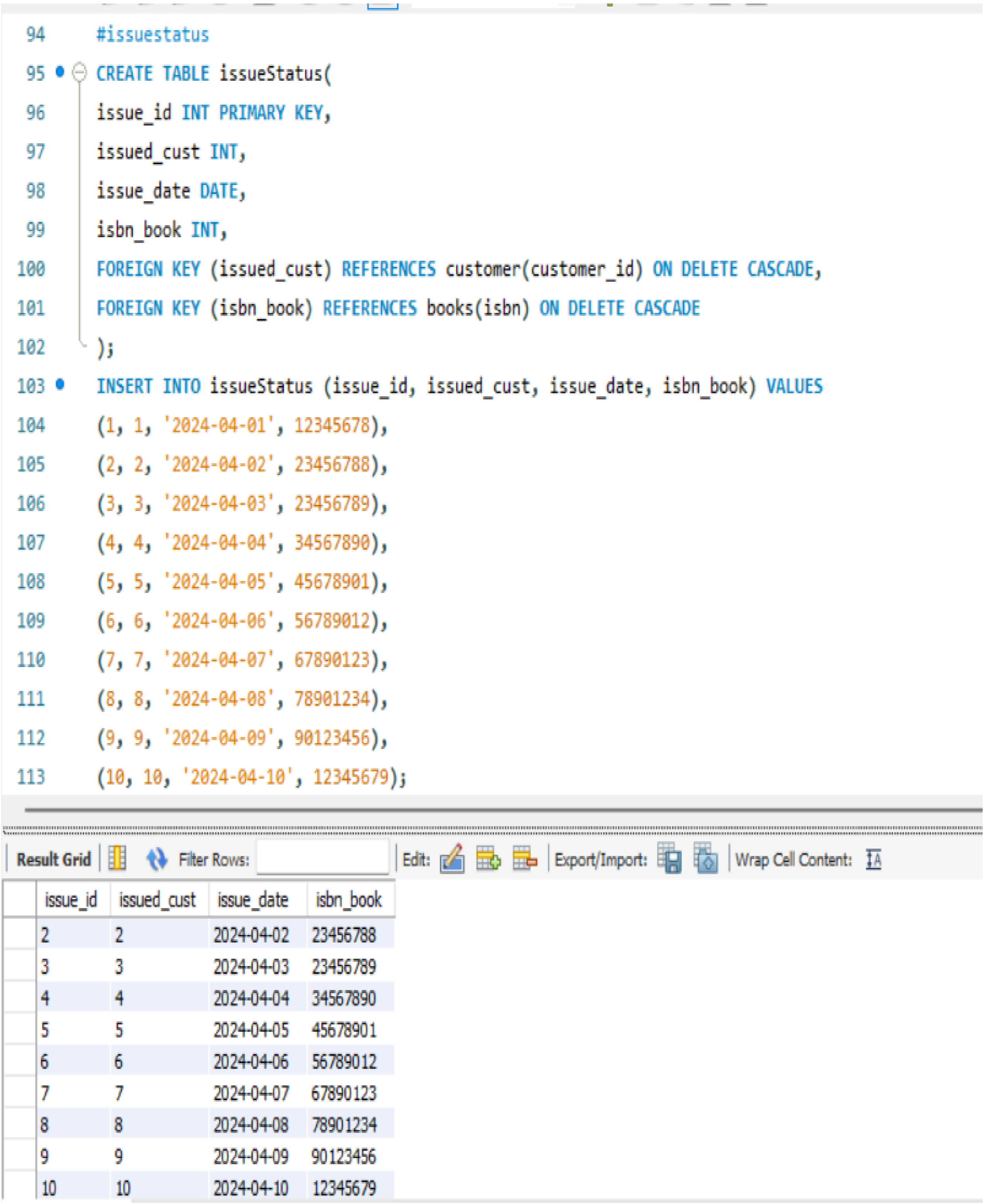
4. Customer
-Customer_Id - Set as PRIMARY KEY
-Customer_name
-Customer_address
-Reg_date



```sql
#table-customer
CREATE TABLE customer(
customer_id INT PRIMARY KEY,
customer_name VARCHAR(25),
customer_address VARCHAR(50),
reg_date DATE
);
INSERT INTO customer (customer_id, customer_name, customer_address, reg_date) VALUES
(1, 'Rahul Sharma', '123, Main Street, Mumbai', '2024-04-01'),
(2, 'Priya Patel', '456, Park Avenue, Delhi', '2024-04-02'),
(3, 'Aarav Gupta', '789, Elm Road, Bangalore', '2024-04-03'),
(4, 'Neha Singh', '101, Oak Lane, Kolkata', '2024-04-04'),
(5, 'Vivek Shah', '234, Maple Drive, Chennai', '2024-04-05'),
(6, 'Aisha Khan', '567, Cedar Court, Hyderabad', '2024-04-06'),
(7, 'Ananya Joshi', '890, Pine Street, Pune', '2024-04-07'),
(8, 'Rohan Desai', '123, Birch Avenue, Jaipur', '2024-04-08'),
(9, 'Sneha Reddy', '456, Willow Lane, Ahmedabad', '2024-04-09'),
(10, 'Kiran Kumar', '789, Spruce Road, Lucknow', '2024-04-10');
```

customer_id	customer_name	customer_address	reg_date
1	Rahul Sharma	123, Main Street, Mumbai	2024-04-01
2	Priya Patel	456, Park Avenue, Delhi	2024-04-02
3	Aarav Gupta	789, Elm Road, Bangalore	2024-04-03
4	Neha Singh	101, Oak Lane, Kolkata	2024-04-04
5	Vivek Shah	234, Maple Drive, Chennai	2024-04-05
6	Aisha Khan	567, Cedar Court, Hyderabad	2024-04-06
7	Ananya Joshi	890, Pine Street, Pune	2024-04-07
8	Rohan Desai	123, Birch Avenue, Jaipur	2024-04-08
9	Sneha Reddy	456, Willow Lane, Ahmedabad	2024-04-09
10	Kiran Kumar	789, Spruce Road, Lucknow	2024-04-10

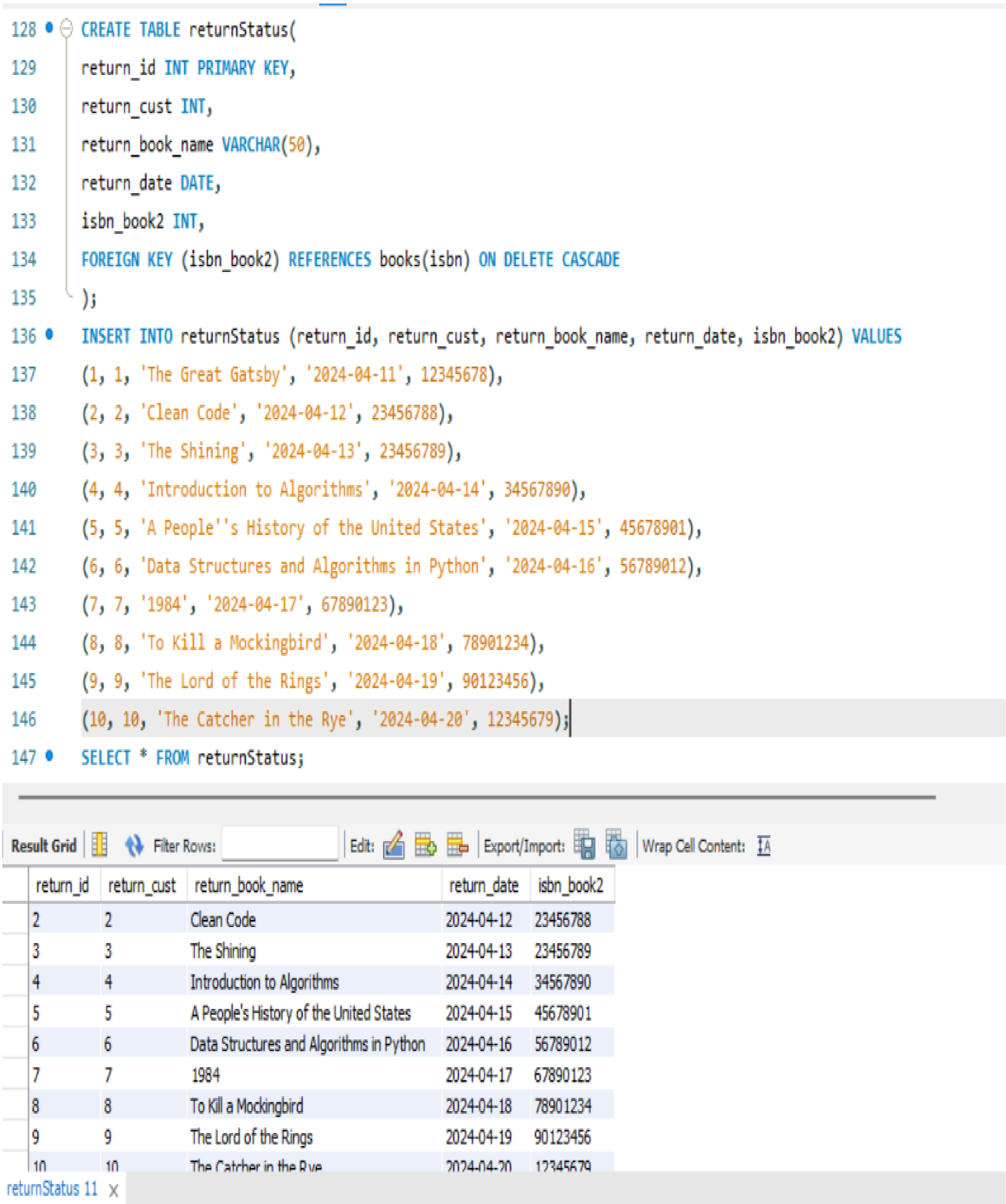
5. IssueStatus
-Issue_Id - Set as PRIMARY KEY
-Issued_cust – Set as FOREIGN KEY and it refer customer_id in CUSTOMER table Issued_book_name
-Issue_date
-Isbn_book – Set as FOREIGN KEY and it should refer isbn in BOOKS table



```
#issuestatus
CREATE TABLE issueStatus(
issue_id INT PRIMARY KEY,
issued_cust INT,
issue_date DATE,
isbn_book INT,
FOREIGN KEY (issued_cust) REFERENCES customer(customer_id) ON DELETE CASCADE,
FOREIGN KEY (isbn_book) REFERENCES books(isbn) ON DELETE CASCADE
);
INSERT INTO issueStatus (issue_id, issued_cust, issue_date, isbn_book) VALUES
(1, 1, '2024-04-01', 12345678),
(2, 2, '2024-04-02', 23456788),
(3, 3, '2024-04-03', 23456789),
(4, 4, '2024-04-04', 34567890),
(5, 5, '2024-04-05', 45678901),
(6, 6, '2024-04-06', 56789012),
(7, 7, '2024-04-07', 67890123),
(8, 8, '2024-04-08', 78901234),
(9, 9, '2024-04-09', 90123456),
(10, 10, '2024-04-10', 12345679);
```

issue_id	issued_cust	issue_date	isbn_book
2	2	2024-04-02	23456788
3	3	2024-04-03	23456789
4	4	2024-04-04	34567890
5	5	2024-04-05	45678901
6	6	2024-04-06	56789012
7	7	2024-04-07	67890123
8	8	2024-04-08	78901234
9	9	2024-04-09	90123456
10	10	2024-04-10	12345679

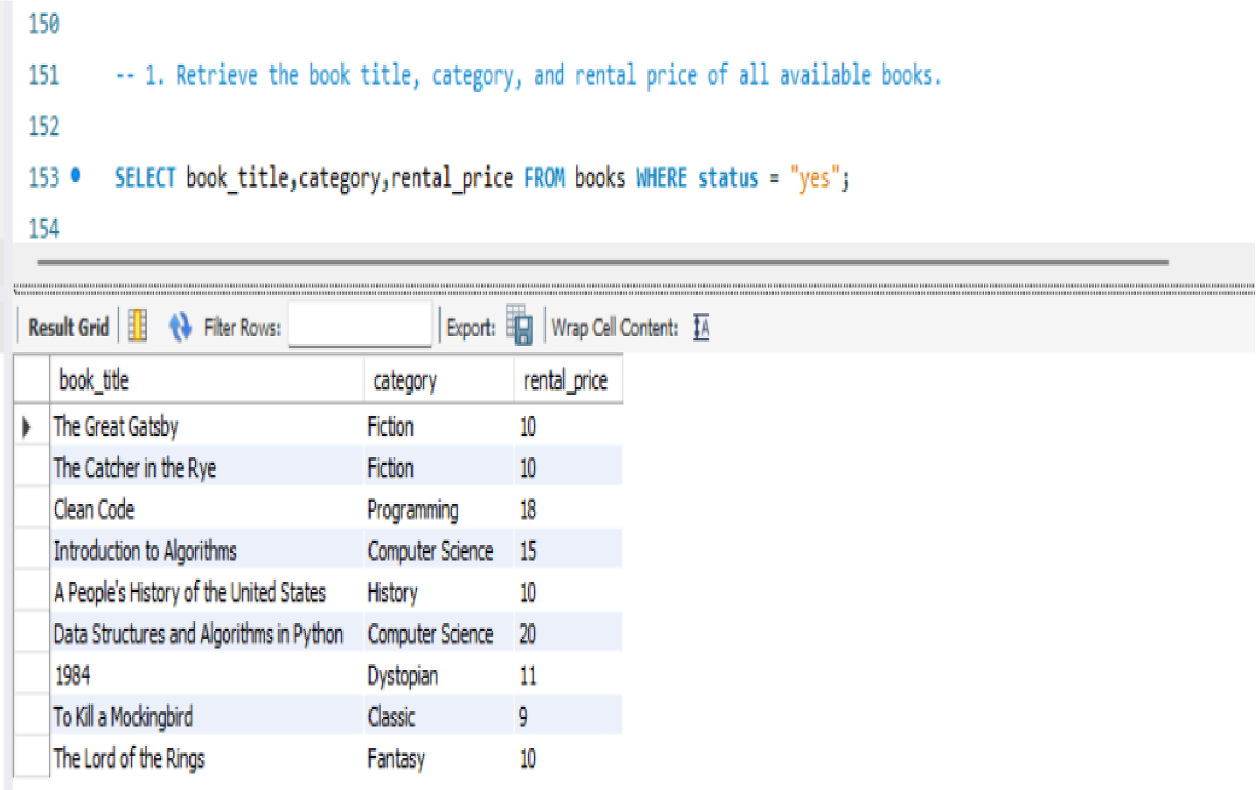
6. ReturnStatus
-Return_Id - Set as PRIMARY KEY
-Return_cust
-Return_book_name
-Return_date
-Isbn_book2 - Set as FOREIGN KEY and it should refer isbn in BOOKS table



```sql
CREATE TABLE returnStatus(
return_id INT PRIMARY KEY,
return_cust INT,
return_book_name VARCHAR(50),
return_date DATE,
isbn_book2 INT,
FOREIGN KEY (isbn_book2) REFERENCES books(isbn) ON DELETE CASCADE
);
INSERT INTO returnStatus (return_id, return_cust, return_book_name, return_date, isbn_book2) VALUES
(1, 1, 'The Great Gatsby', '2024-04-11', 12345678),
(2, 2, 'Clean Code', '2024-04-12', 23456788),
(3, 3, 'The Shining', '2024-04-13', 23456789),
(4, 4, 'Introduction to Algorithms', '2024-04-14', 34567890),
(5, 5, 'A People''s History of the United States', '2024-04-15', 45678901),
(6, 6, 'Data Structures and Algorithms in Python', '2024-04-16', 56789012),
(7, 7, '1984', '2024-04-17', 67890123),
(8, 8, 'To Kill a Mockingbird', '2024-04-18', 78901234),
(9, 9, 'The Lord of the Rings', '2024-04-19', 90123456),
(10, 10, 'The Catcher in the Rye', '2024-04-20', 12345679);
SELECT * FROM returnStatus;
```

return_id	return_cust	return_book_name	return_date	isbn_book2
2	2	Clean Code	2024-04-12	23456788
3	3	The Shining	2024-04-13	23456789
4	4	Introduction to Algorithms	2024-04-14	34567890
5	5	A People's History of the United States	2024-04-15	45678901
6	6	Data Structures and Algorithms in Python	2024-04-16	56789012
7	7	1984	2024-04-17	67890123
8	8	To Kill a Mockingbird	2024-04-18	78901234
9	9	The Lord of the Rings	2024-04-19	90123456
10	10	The Catcher in the Rye	2024-04-20	12345679

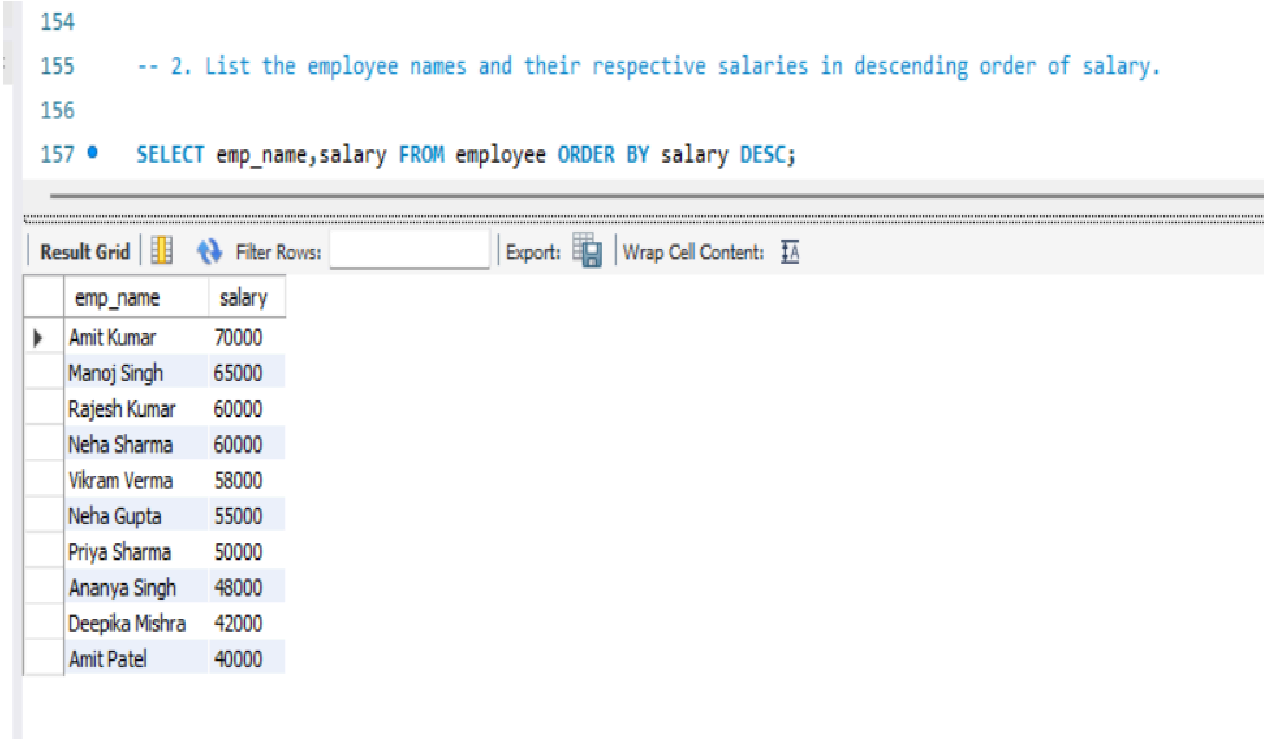
1. Retrieve the book title, category, and rental price of all available books.



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

SELECT book_title,category,rental_price FROM books WHERE status = "yes";

```

book_title	category	rental_price
The Great Gatsby	Fiction	10
The Catcher in the Rye	Fiction	10
Clean Code	Programming	18
Introduction to Algorithms	Computer Science	15
A People's History of the United States	History	10
Data Structures and Algorithms in Python	Computer Science	20
1984	Dystopian	11
To Kill a Mockingbird	Classic	9
The Lord of the Rings	Fantasy	10

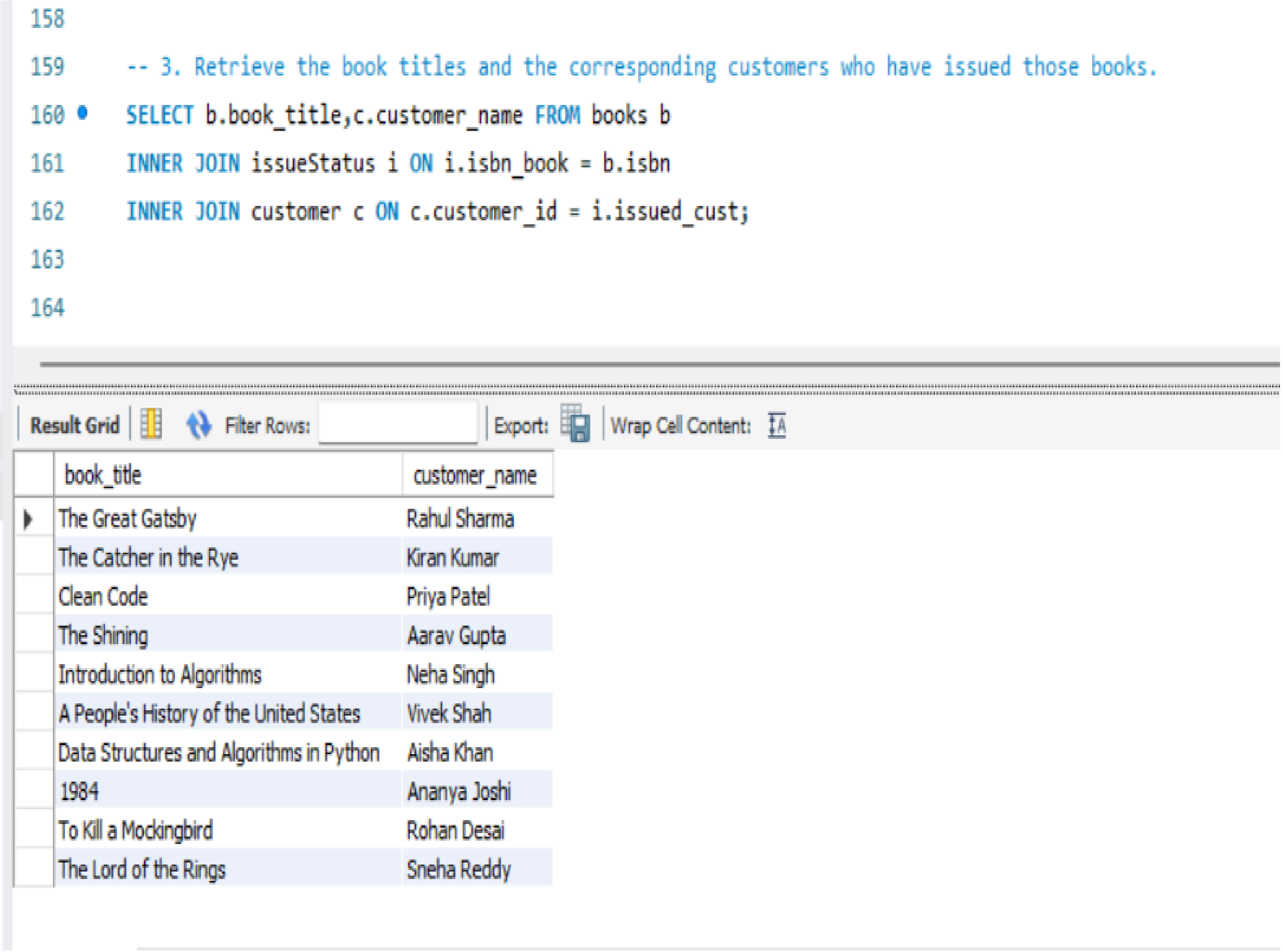
2. List the employee names and their respective salaries in descending order of salary.



```

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

SELECT emp_name,salary FROM employee ORDER BY salary DESC;
```

emp_name	salary
Amit Kumar	70000
Manoj Singh	65000
Rajesh Kumar	60000
Neha Sharma	60000
Vikram Verma	58000
Neha Gupta	55000
Priya Sharma	50000
Ananya Singh	48000
Deepika Mishra	42000
Amit Patel	40000

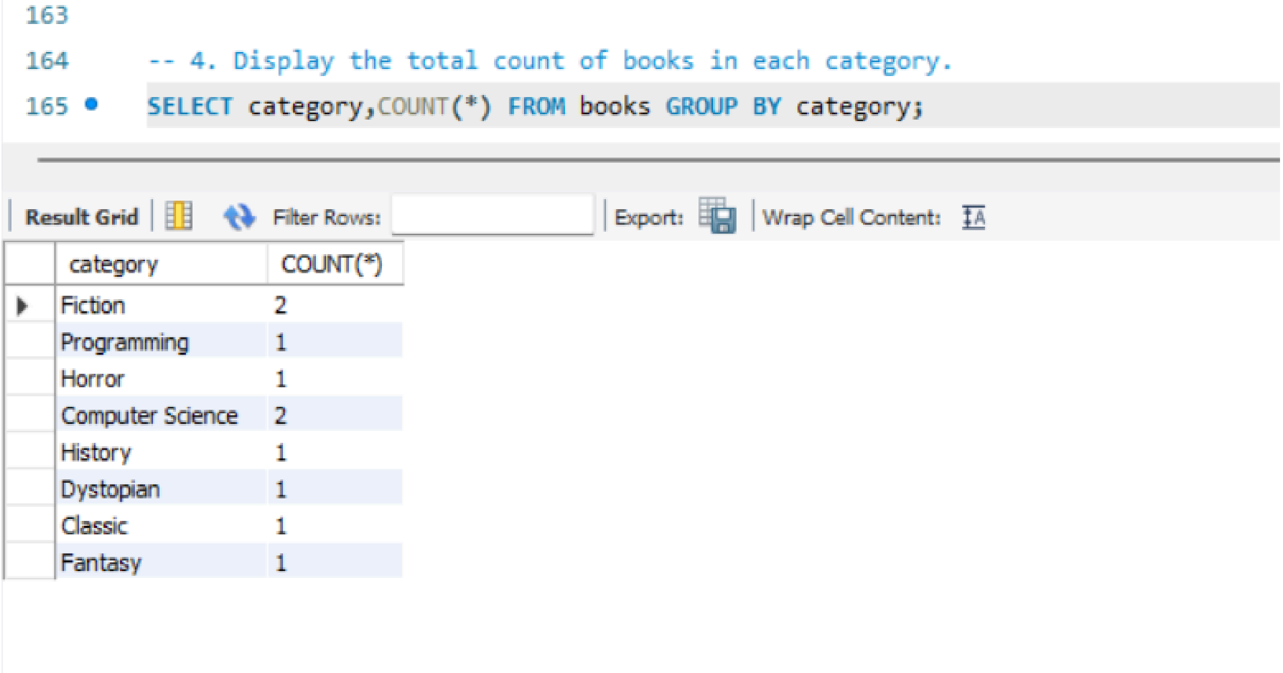
3. Retrieve the book titles and the corresponding customers who have issued those books.



```sql

-- 3. Retrieve the book titles and the corresponding customers who have issued those books.
SELECT b.book_title,c.customer_name FROM books b
INNER JOIN issueStatus i ON i.isbn_book = b.isbn
INNER JOIN customer c ON c.customer_id = i.issued_cust;


```

book_title	customer_name
The Great Gatsby	Rahul Sharma
The Catcher in the Rye	Kiran Kumar
Clean Code	Priya Patel
The Shining	Aarav Gupta
Introduction to Algorithms	Neha Singh
A People's History of the United States	Vivek Shah
Data Structures and Algorithms in Python	Aisha Khan
1984	Ananya Joshi
To Kill a Mockingbird	Rohan Desai
The Lord of the Rings	Sneha Reddy

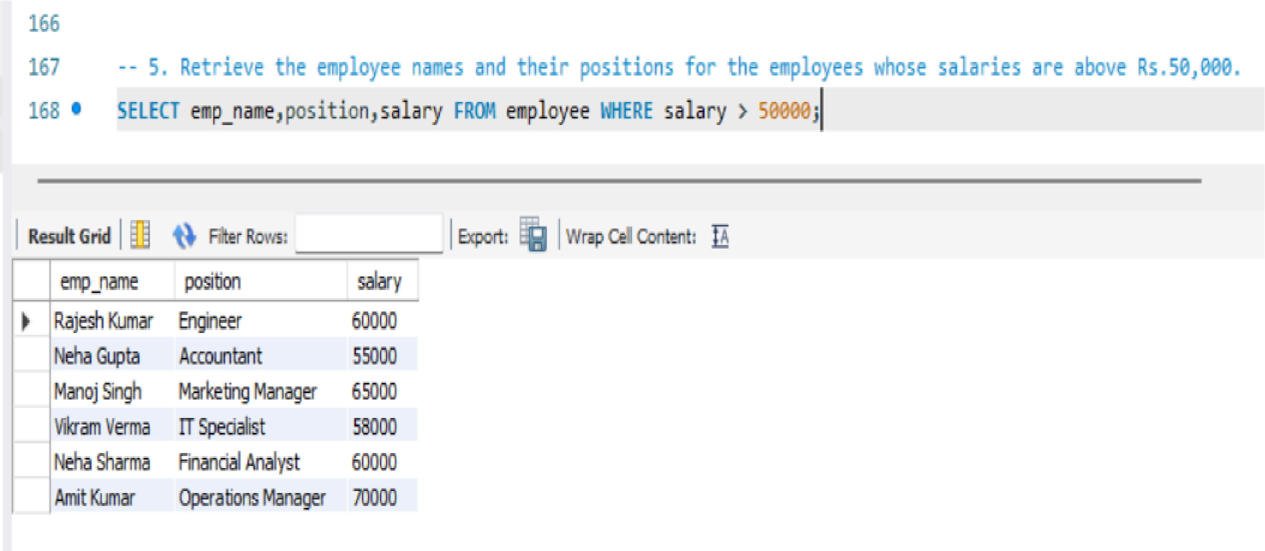
4. Display the total count of books in each category.



```sql

-- 4. Display the total count of books in each category.
SELECT category,COUNT(*) FROM books GROUP BY category;
```

category	COUNT(*)
Fiction	2
Programming	1
Horror	1
Computer Science	2
History	1
Dystopian	1
Classic	1
Fantasy	1

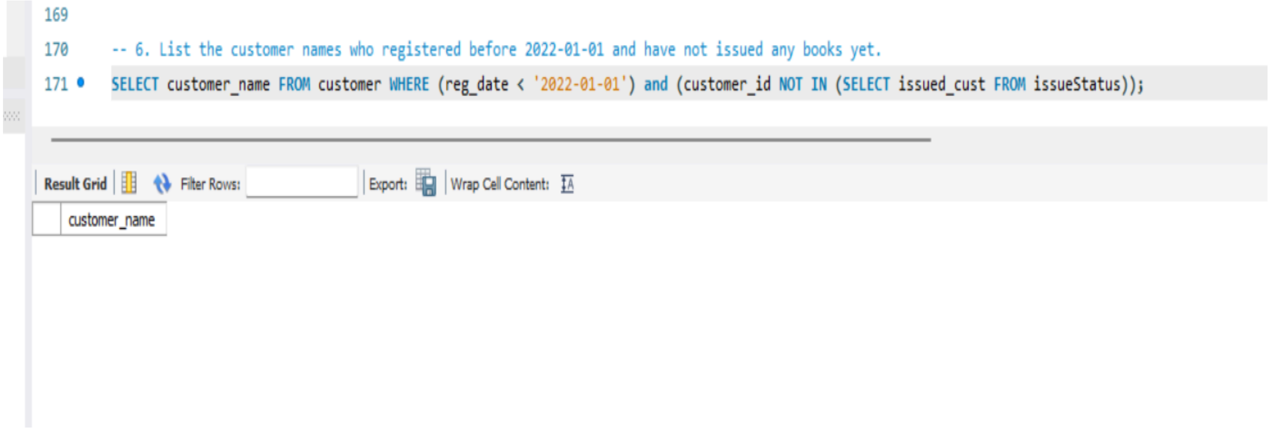
5. Retrieve the employee names and their positions for the employees whose salaries are above Rs.50,000.



```sql
-- 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;
```

emp_name	position	salary
Rajesh Kumar	Engineer	60000
Neha Gupta	Accountant	55000
Manoj Singh	Marketing Manager	65000
Vikram Verma	IT Specialist	58000
Neha Sharma	Financial Analyst	60000
Amit Kumar	Operations Manager	70000

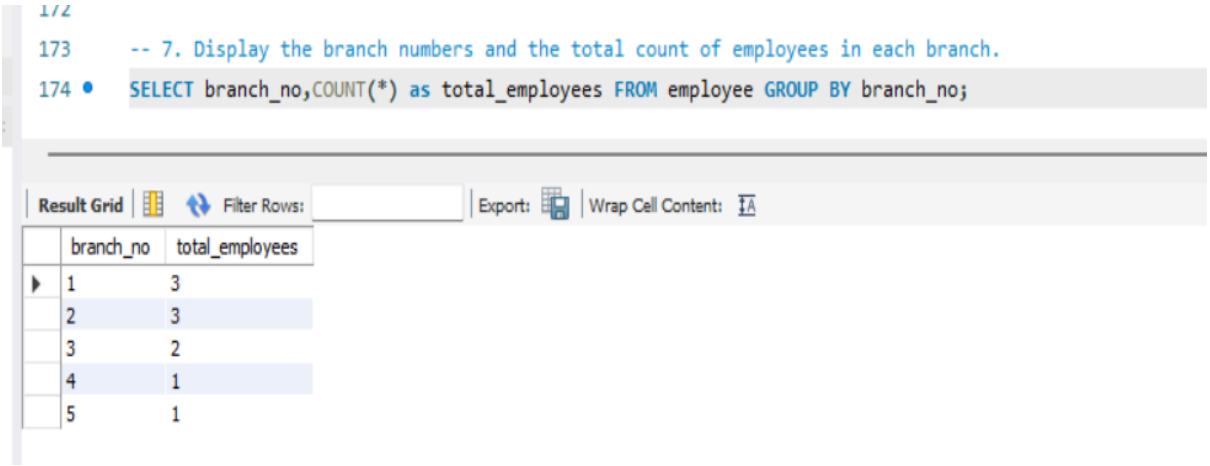
6. List the customer names who registered before 2022-01-01 and have not issued any books yet.



```sql
-- 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));
```

customer_name

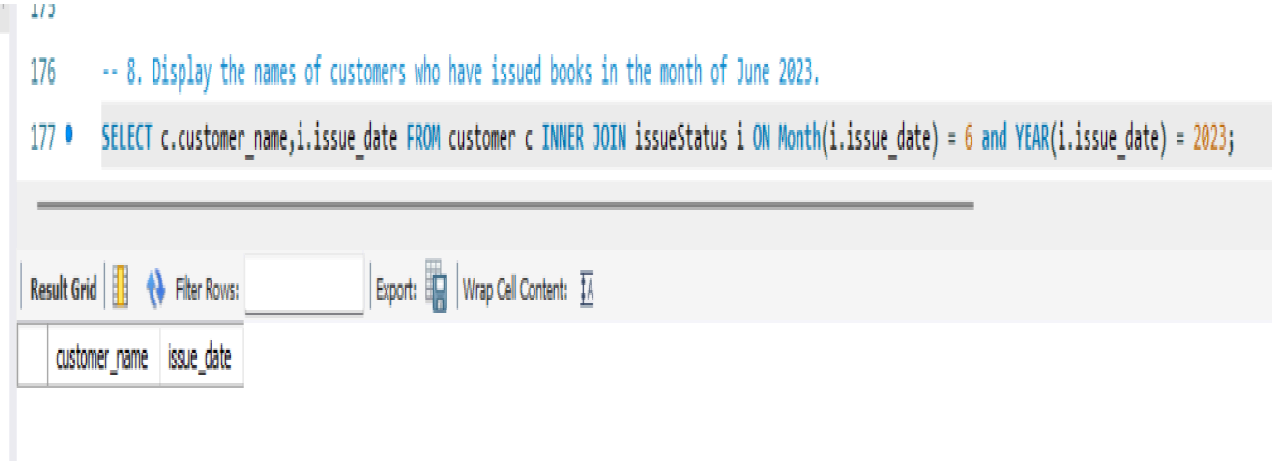
7. Display the branch numbers and the total count of employees in each branch.



```sql
-- 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;
```

branch_no	total_employees
1	3
2	3
3	2
4	1
5	1

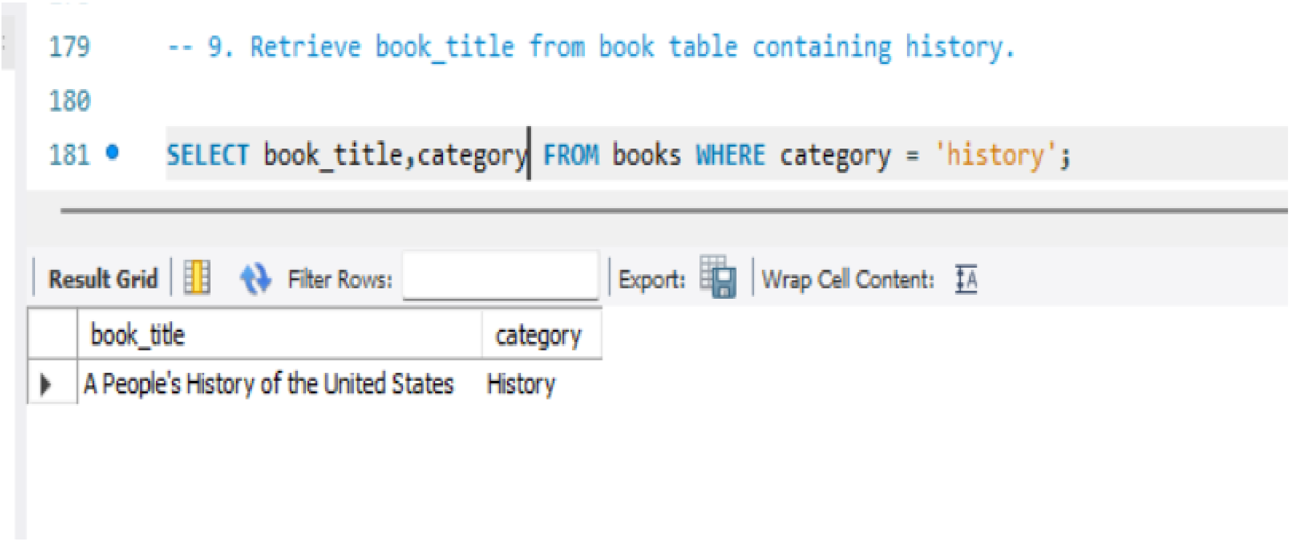
8. Display the names of customers who have issued books in the month of June 2023.



```
-- 8. Display the names of customers who have issued books in the month of June 2023.
SELECT c.customer_name,i.issue_date FROM customer c INNER JOIN issueStatus i ON Month(i.issue_date) = 6 and YEAR(i.issue_date) = 2023;
```

customer_name	issue_date

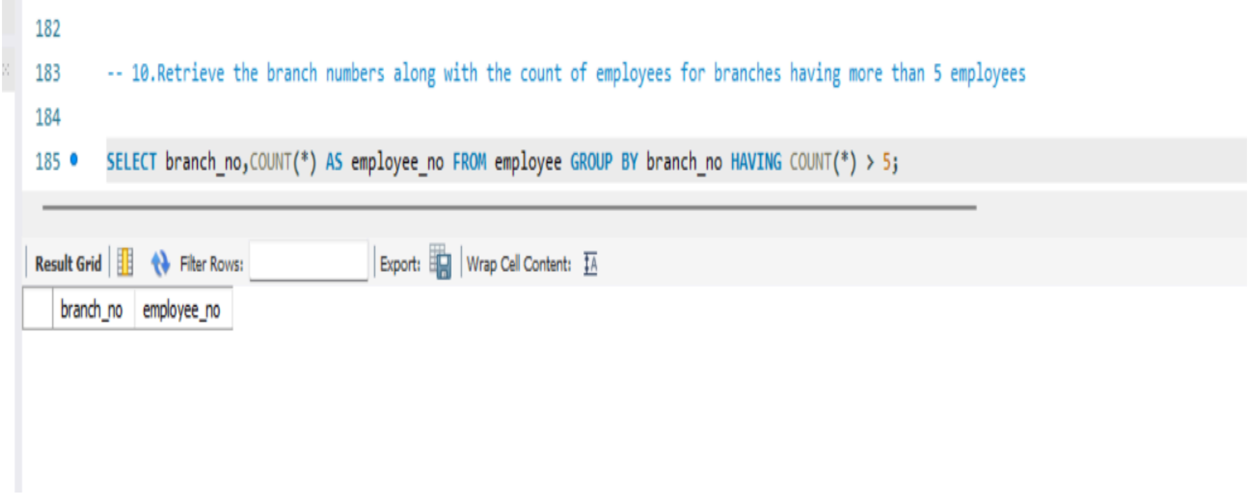
9. Retrieve book_title from book table containing history.



```
-- 9. Retrieve book_title from book table containing history.

SELECT book_title,category FROM books WHERE category = 'history';
```

book_title	category
A People's History of the United States	History

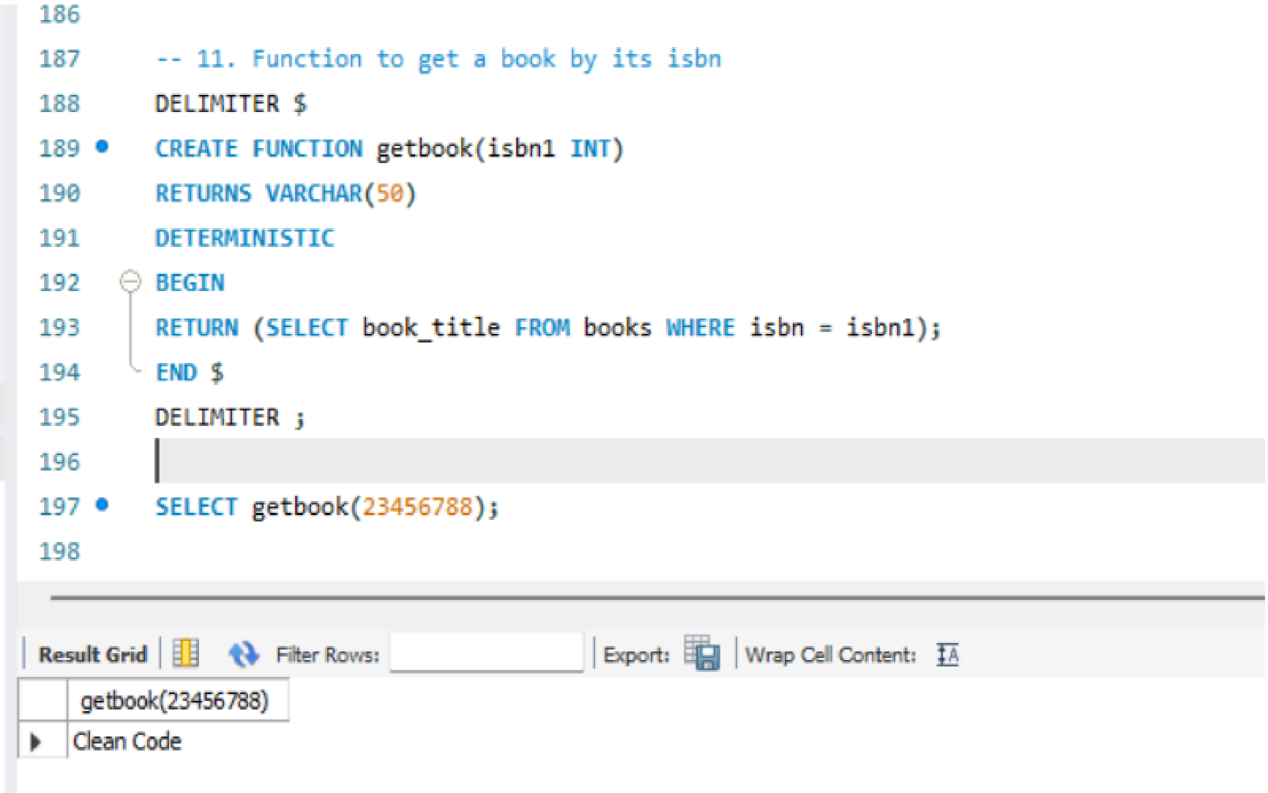
10.Retrieve the branch numbers along with the count of employees for branches having more than 5 employees



```

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

SELECT branch_no,COUNT(*) AS employee_no FROM employee GROUP BY branch_no HAVING COUNT(*) > 5;
```

branch_no	employee_no

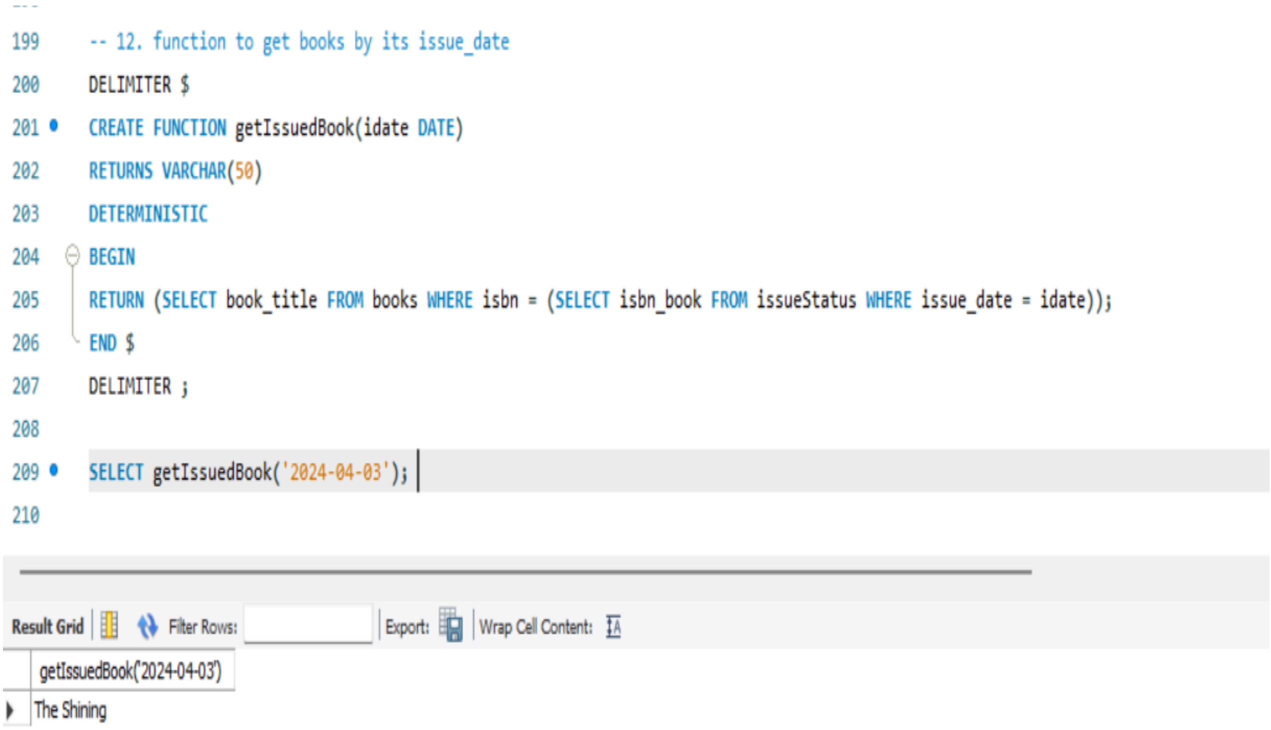
11.Get book title from its isbn



```sql

-- 11. Function to get a book by its isbn
DELIMITER $
CREATE FUNCTION getbook(isbn1 INT)
RETURNS VARCHAR(50)
DETERMINISTIC
BEGIN
RETURN (SELECT book_title FROM books WHERE isbn = isbn1);
END $
DELIMITER ;

SELECT getbook(23456788);

```

getbook(23456788)
Clean Code

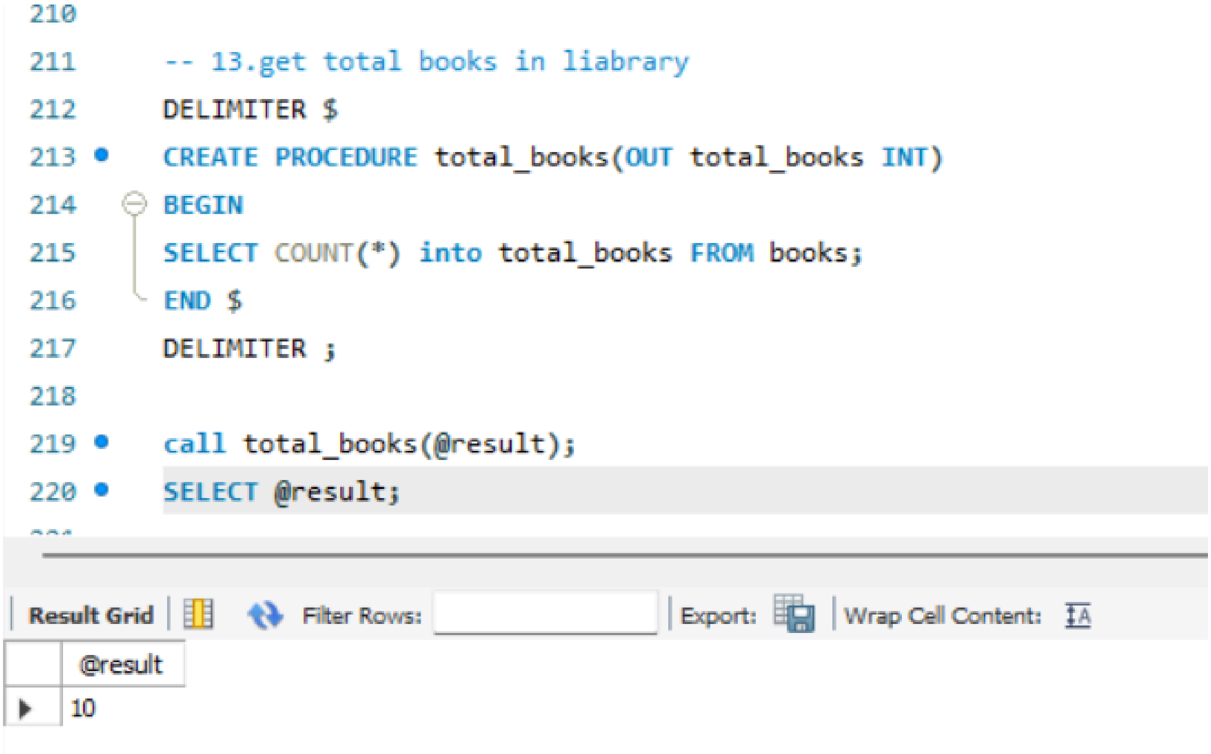
12. Get book title from issue date



```sql
-- 12. function to get books by its issue_date
DELIMITER $
CREATE FUNCTION getIssuedBook(idate DATE)
RETURNS VARCHAR(50)
DETERMINISTIC
BEGIN
RETURN (SELECT book_title FROM books WHERE isbn = (SELECT isbn_book FROM issueStatus WHERE issue_date = idate));
END $
DELIMITER ;

SELECT getIssuedBook('2024-04-03');

```

getIssuedBook('2024-04-03')
The Shining

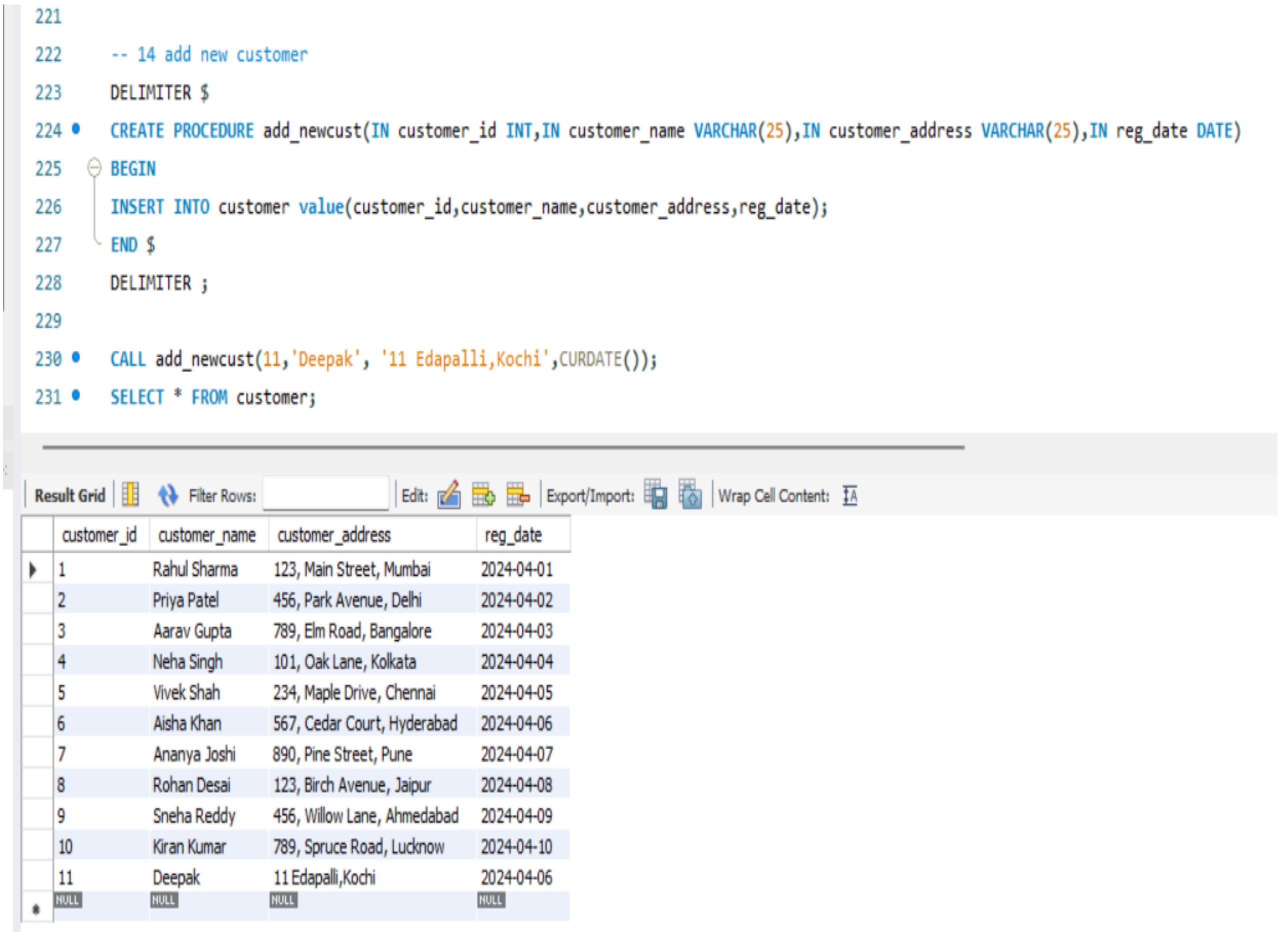
13. Total books in the liabrary



```sql
-- 13.get total books in liabrary
DELIMITER $
CREATE PROCEDURE total_books(OUT total_books INT)
BEGIN
SELECT COUNT(*) into total_books FROM books;
END $
DELIMITER ;

call total_books(@result);
SELECT @result;
```

@result
10

14.Add a new customer to customer table.



```sql
-- 14 add new customer
DELIMITER $
CREATE PROCEDURE add_newcust(IN customer_id INT,IN customer_name VARCHAR(25),IN customer_address VARCHAR(25),IN reg_date DATE)
BEGIN
INSERT INTO customer value(customer_id,customer_name,customer_address,reg_date);
END $
DELIMITER ;

CALL add_newcust(11,'Deepak', '11 Edapalli,Kochi',CURDATE());
SELECT * FROM customer;
```

customer_id	customer_name	customer_address	reg_date
1	Rahul Sharma	123, Main Street, Mumbai	2024-04-01
2	Priya Patel	456, Park Avenue, Delhi	2024-04-02
3	Aarav Gupta	789, Elm Road, Bangalore	2024-04-03
4	Neha Singh	101, Oak Lane, Kolkata	2024-04-04
5	Vivek Shah	234, Maple Drive, Chennai	2024-04-05
6	Aisha Khan	567, Cedar Court, Hyderabad	2024-04-06
7	Ananya Joshi	890, Pine Street, Pune	2024-04-07
8	Rohan Desai	123, Birch Avenue, Jaipur	2024-04-08
9	Sneha Reddy	456, Willow Lane, Ahmedabad	2024-04-09
10	Kiran Kumar	789, Spruce Road, Lucknow	2024-04-10
11	Deepak	11 Edapalli,Kochi	2024-04-06
NULL	NULL	NULL	NULL