LIBRARY MANAGEMENT SYSTEM

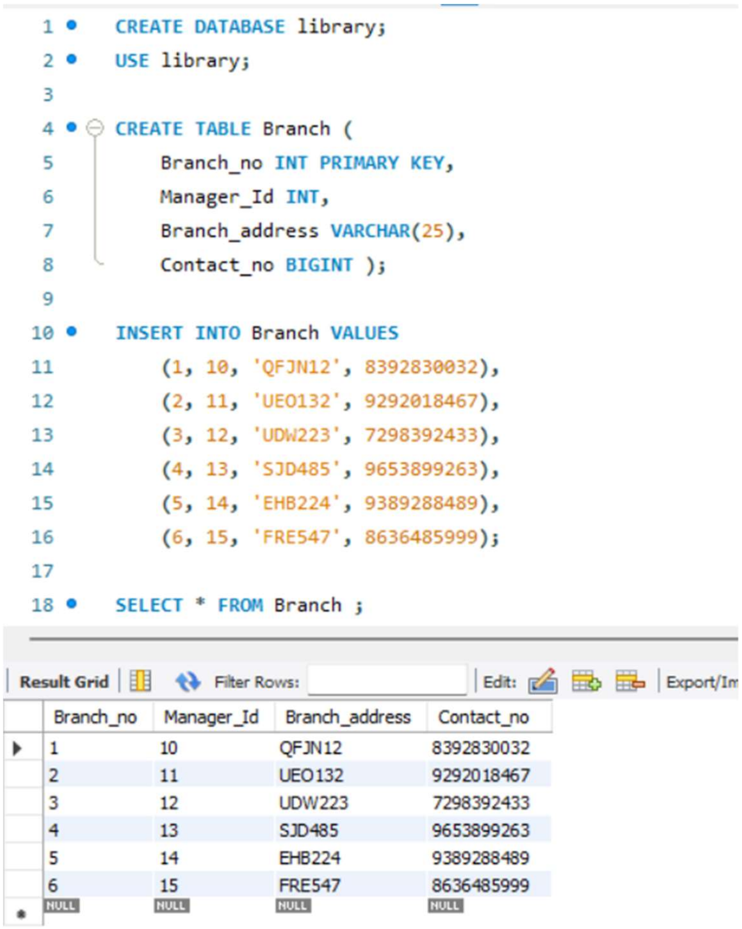
Create a database named library and following TABLES in the database:

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

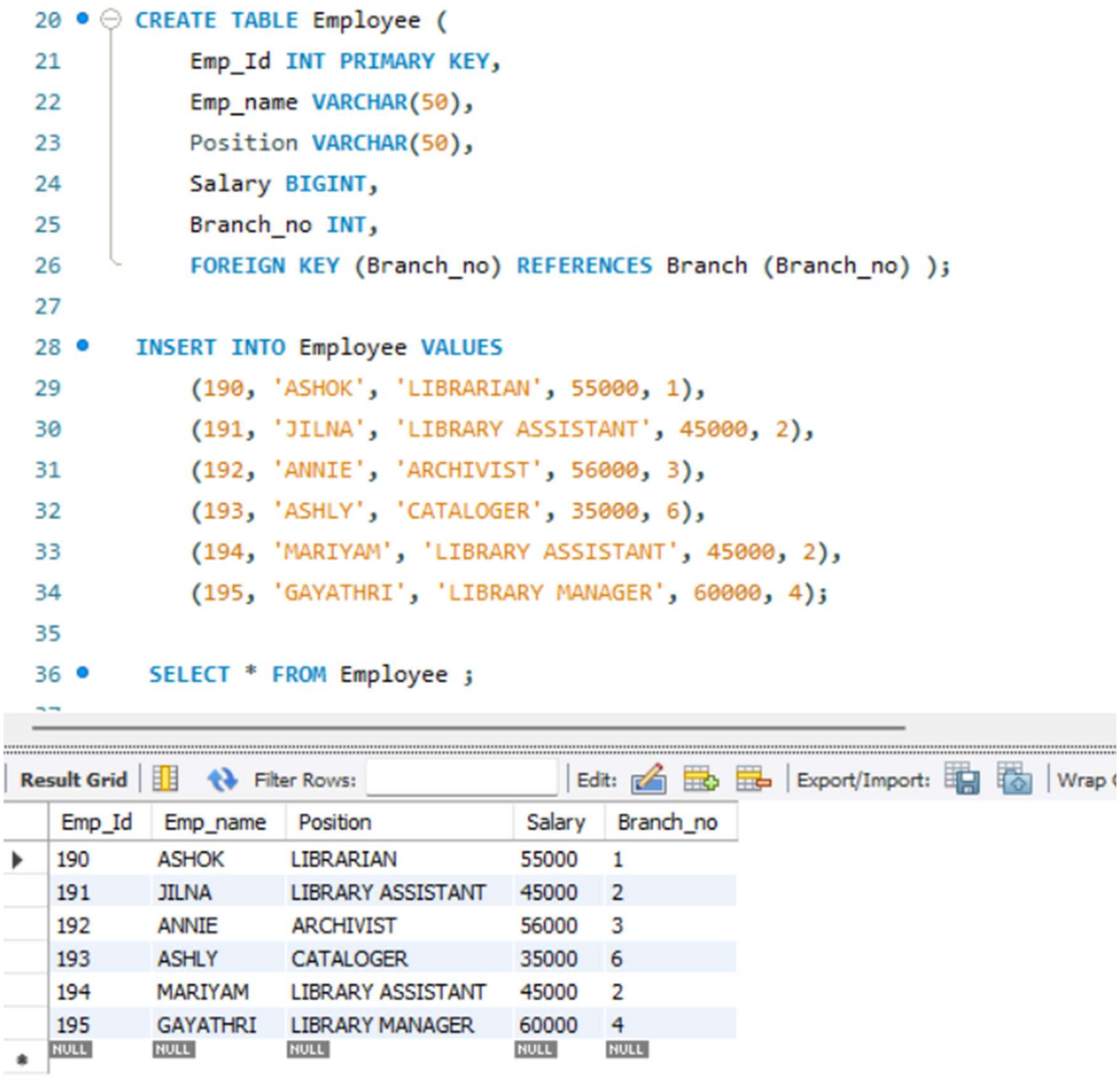


```
CREATE DATABASE library;
USE library;

CREATE TABLE Branch (
    Branch_no INT PRIMARY KEY,
    Manager_Id INT,
    Branch_address VARCHAR(25),
    Contact_no BIGINT );

INSERT INTO Branch VALUES
    (1, 10, 'QFJN12', 8392830032),
    (2, 11, 'UEO132', 9292018467),
    (3, 12, 'UDW223', 7298392433),
    (4, 13, 'SJD485', 9653899263),
    (5, 14, 'EHB224', 9389288489),
    (6, 15, 'FRE547', 8636485999);

SELECT * FROM Branch ;
```

Branch_no	Manager_Id	Branch_address	Contact_no
1	10	QFJN12	8392830032
2	11	UEO132	9292018467
3	12	UDW223	7298392433
4	13	SJD485	9653899263
5	14	EHB224	9389288489
6	15	FRE547	8636485999
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

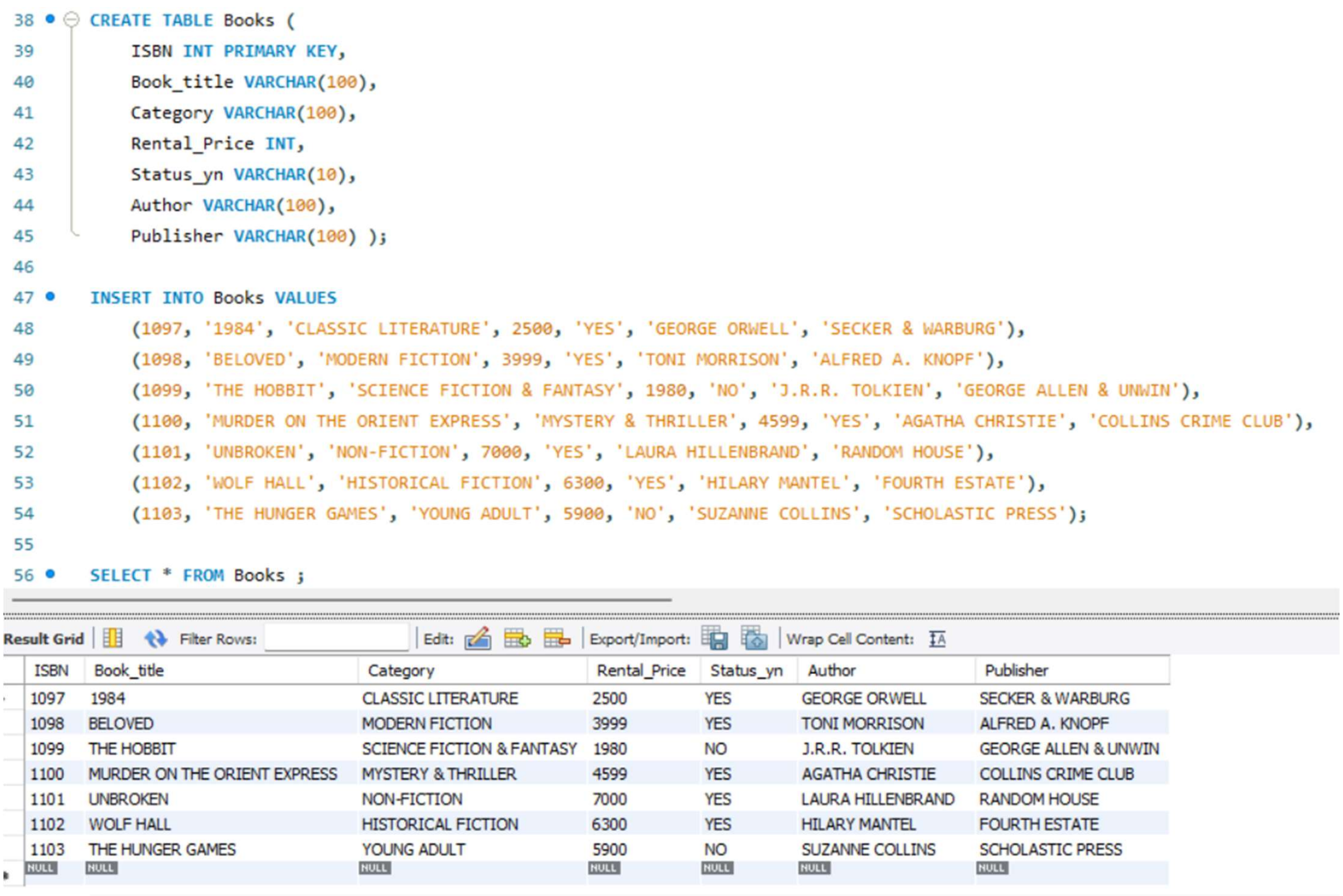


```sql
CREATE TABLE Employee (
    Emp_Id INT PRIMARY KEY,
    Emp_name VARCHAR(50),
    Position VARCHAR(50),
    Salary BIGINT,
    Branch_no INT,
    FOREIGN KEY (Branch_no) REFERENCES Branch (Branch_no) );

INSERT INTO Employee VALUES
    (190, 'ASHOK', 'LIBRARIAN', 55000, 1),
    (191, 'JILNA', 'LIBRARY ASSISTANT', 45000, 2),
    (192, 'ANNIE', 'ARCHIVIST', 56000, 3),
    (193, 'ASHLY', 'CATALOGER', 35000, 6),
    (194, 'MARIYAM', 'LIBRARY ASSISTANT', 45000, 2),
    (195, 'GAYATHRI', 'LIBRARY MANAGER', 60000, 4);

 SELECT * FROM Employee ;
```

Emp_Id	Emp_name	Position	Salary	Branch_no
190	ASHOK	LIBRARIAN	55000	1
191	JILNA	LIBRARY ASSISTANT	45000	2
192	ANNIE	ARCHIVIST	56000	3
193	ASHLY	CATALOGER	35000	6
194	MARIYAM	LIBRARY ASSISTANT	45000	2
195	GAYATHRI	LIBRARY MANAGER	60000	4
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

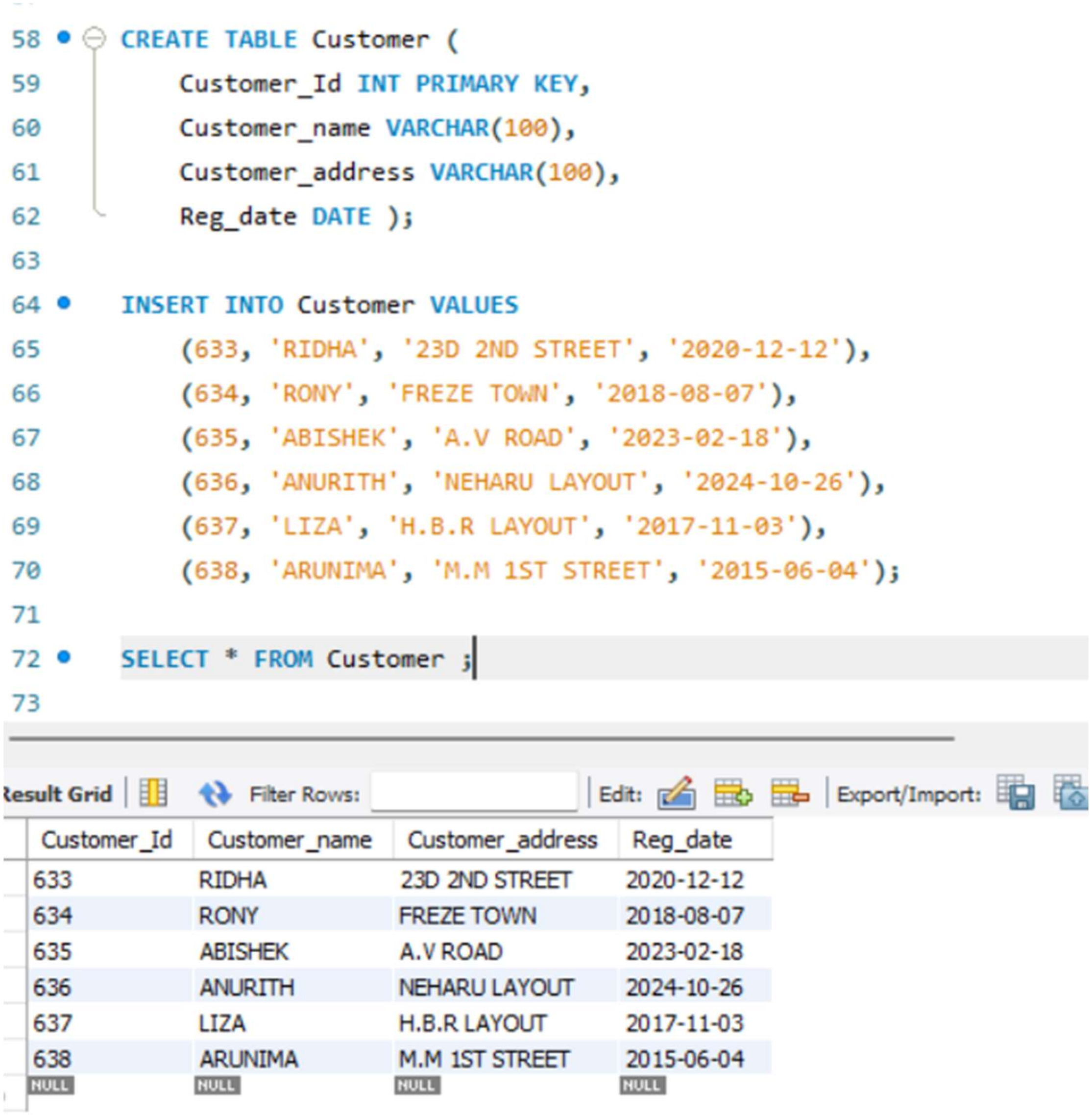


```
CREATE TABLE Books (
    ISBN INT PRIMARY KEY,
    Book_title VARCHAR(100),
    Category VARCHAR(100),
    Rental_Price INT,
    Status_yn VARCHAR(10),
    Author VARCHAR(100),
    Publisher VARCHAR(100) );

INSERT INTO Books VALUES
    (1097, '1984', 'CLASSIC LITERATURE', 2500, 'YES', 'GEORGE ORWELL', 'SECKER & WARBURG'),
    (1098, 'BELOVED', 'MODERN FICTION', 3999, 'YES', 'TONI MORRISON', 'ALFRED A. KNOPF'),
    (1099, 'THE HOBBIT', 'SCIENCE FICTION & FANTASY', 1980, 'NO', 'J.R.R. TOLKIEN', 'GEORGE ALLEN & UNWIN'),
    (1100, 'MURDER ON THE ORIENT EXPRESS', 'MYSTERY & THRILLER', 4599, 'YES', 'AGATHA CHRISTIE', 'COLLINS CRIME CLUB'),
    (1101, 'UNBROKEN', 'NON-FICTION', 7000, 'YES', 'LAURA HILLENBRAND', 'RANDOM HOUSE'),
    (1102, 'WOLF HALL', 'HISTORICAL FICTION', 6300, 'YES', 'HILARY MANTEL', 'FOURTH ESTATE'),
    (1103, 'THE HUNGER GAMES', 'YOUNG ADULT', 5900, 'NO', 'SUZANNE COLLINS', 'SCHOLASTIC PRESS');

SELECT * FROM Books ;
```

ISBN	Book_title	Category	Rental_Price	Status_yn	Author	Publisher
1097	1984	CLASSIC LITERATURE	2500	YES	GEORGE ORWELL	SECKER & WARBURG
1098	BELOVED	MODERN FICTION	3999	YES	TONI MORRISON	ALFRED A. KNOPF
1099	THE HOBBIT	SCIENCE FICTION & FANTASY	1980	NO	J.R.R. TOLKIEN	GEORGE ALLEN & UNWIN
1100	MURDER ON THE ORIENT EXPRESS	MYSTERY & THRILLER	4599	YES	AGATHA CHRISTIE	COLLINS CRIME CLUB
1101	UNBROKEN	NON-FICTION	7000	YES	LAURA HILLENBRAND	RANDOM HOUSE
1102	WOLF HALL	HISTORICAL FICTION	6300	YES	HILARY MANTEL	FOURTH ESTATE
1103	THE HUNGER GAMES	YOUNG ADULT	5900	NO	SUZANNE COLLINS	SCHOLASTIC PRESS
NULL	NULL	NULL	NULL	NULL	NULL	NULL

4. Customer

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

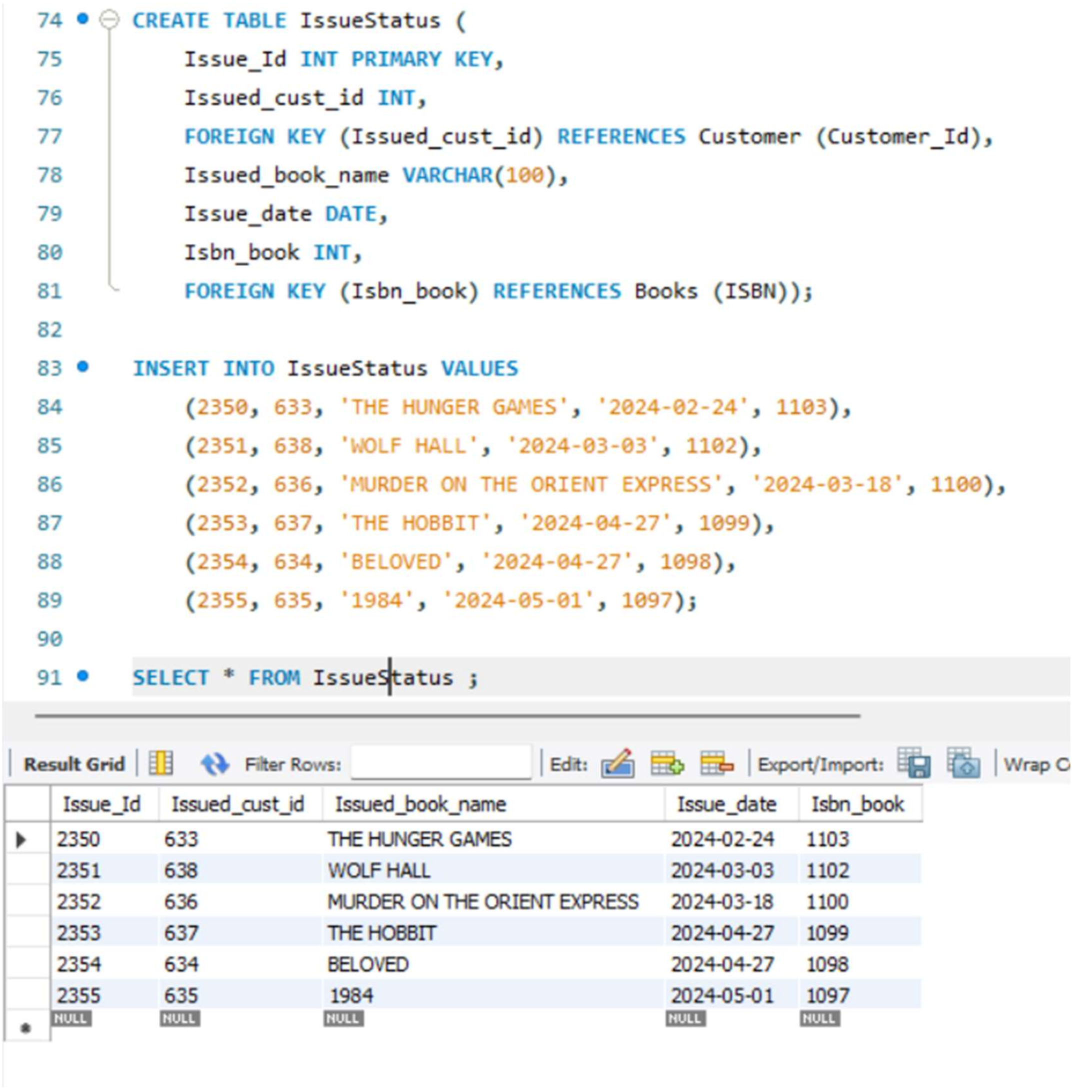


```
CREATE TABLE Customer (
    Customer_Id INT PRIMARY KEY,
    Customer_name VARCHAR(100),
    Customer_address VARCHAR(100),
    Reg_date DATE );

INSERT INTO Customer VALUES
    (633, 'RIDHA', '23D 2ND STREET', '2020-12-12'),
    (634, 'RONY', 'FREZE TOWN', '2018-08-07'),
    (635, 'ABISHEK', 'A.V ROAD', '2023-02-18'),
    (636, 'ANURITH', 'NEHARU LAYOUT', '2024-10-26'),
    (637, 'LIZA', 'H.B.R LAYOUT', '2017-11-03'),
    (638, 'ARUNIMA', 'M.M 1ST STREET', '2015-06-04');

SELECT * FROM Customer ;
```

Customer_Id	Customer_name	Customer_address	Reg_date
633	RIDHA	23D 2ND STREET	2020-12-12
634	RONY	FREZE TOWN	2018-08-07
635	ABISHEK	A.V ROAD	2023-02-18
636	ANURITH	NEHARU LAYOUT	2024-10-26
637	LIZA	H.B.R LAYOUT	2017-11-03
638	ARUNIMA	M.M 1ST STREET	2015-06-04
NULL	NULL	NULL	NULL

5. IssueStatus

- Issue_Id - Set as PRIMARY KEY
- Issued_cust_id – 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

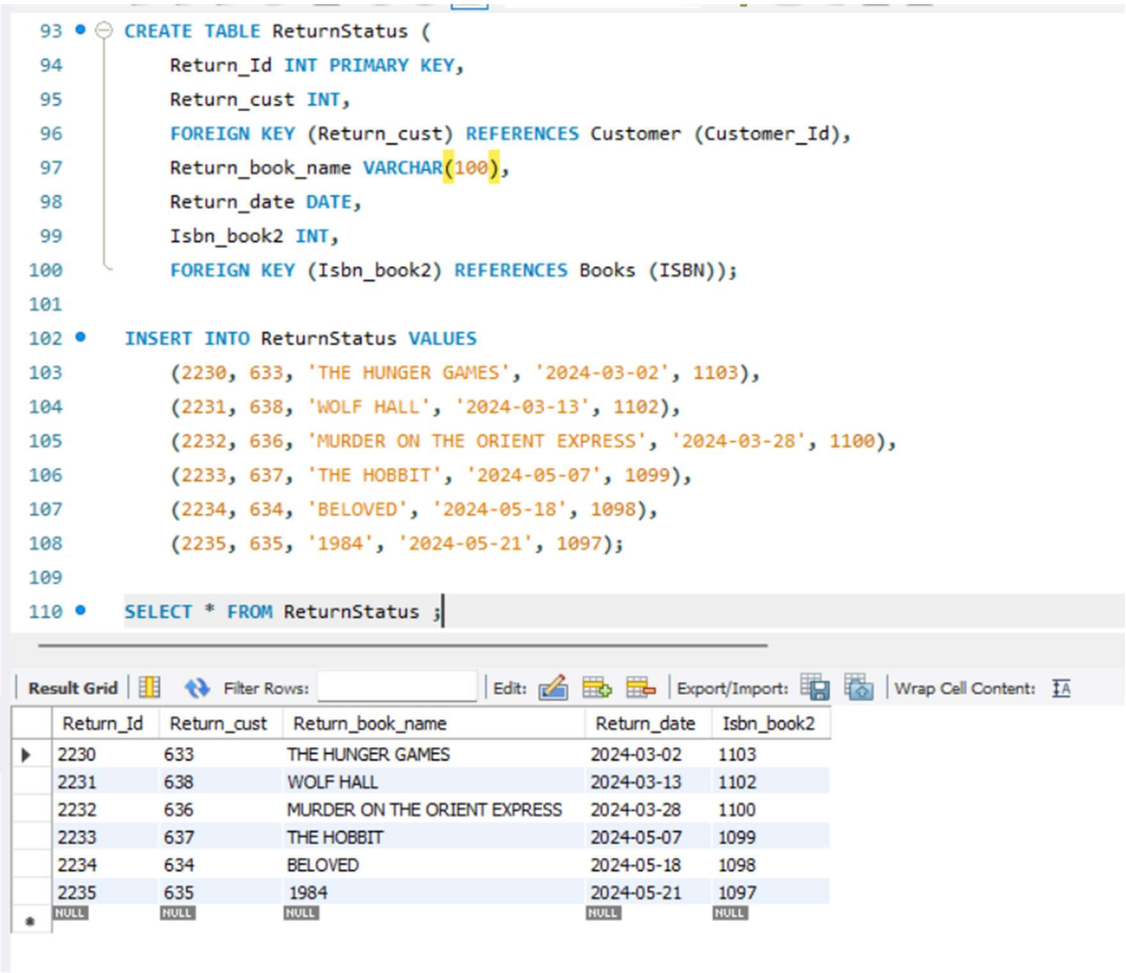


```
CREATE TABLE IssueStatus (
    Issue_Id INT PRIMARY KEY,
    Issued_cust_id INT,
    FOREIGN KEY (Issued_cust_id) REFERENCES Customer (Customer_Id),
    Issued_book_name VARCHAR(100),
    Issue_date DATE,
    Isbn_book INT,
    FOREIGN KEY (Isbn_book) REFERENCES Books (ISBN));

INSERT INTO IssueStatus VALUES
    (2350, 633, 'THE HUNGER GAMES', '2024-02-24', 1103),
    (2351, 638, 'WOLF HALL', '2024-03-03', 1102),
    (2352, 636, 'MURDER ON THE ORIENT EXPRESS', '2024-03-18', 1100),
    (2353, 637, 'THE HOBBIT', '2024-04-27', 1099),
    (2354, 634, 'BELOVED', '2024-04-27', 1098),
    (2355, 635, '1984', '2024-05-01', 1097);

SELECT * FROM IssueStatus ;
```

Issue_Id	Issued_cust_id	Issued_book_name	Issue_date	Isbn_book
2350	633	THE HUNGER GAMES	2024-02-24	1103
2351	638	WOLF HALL	2024-03-03	1102
2352	636	MURDER ON THE ORIENT EXPRESS	2024-03-18	1100
2353	637	THE HOBBIT	2024-04-27	1099
2354	634	BELOVED	2024-04-27	1098
2355	635	1984	2024-05-01	1097
NULL	NULL	NULL	NULL	NULL

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

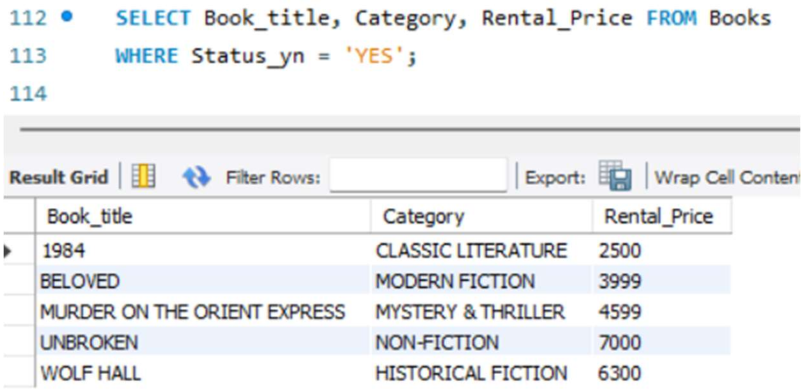


```sql
CREATE TABLE ReturnStatus (
    Return_Id INT PRIMARY KEY,
    Return_cust INT,
    FOREIGN KEY (Return_cust) REFERENCES Customer (Customer_Id),
    Return_book_name VARCHAR(100),
    Return_date DATE,
    Isbn_book2 INT,
    FOREIGN KEY (Isbn_book2) REFERENCES Books (ISBN));

INSERT INTO ReturnStatus VALUES
    (2230, 633, 'THE HUNGER GAMES', '2024-03-02', 1103),
    (2231, 638, 'WOLF HALL', '2024-03-13', 1102),
    (2232, 636, 'MURDER ON THE ORIENT EXPRESS', '2024-03-28', 1100),
    (2233, 637, 'THE HOBBIT', '2024-05-07', 1099),
    (2234, 634, 'BELOVED', '2024-05-18', 1098),
    (2235, 635, '1984', '2024-05-21', 1097);

SELECT * FROM ReturnStatus ;
```

Return_Id	Return_cust	Return_book_name	Return_date	Isbn_book2
2230	633	THE HUNGER GAMES	2024-03-02	1103
2231	638	WOLF HALL	2024-03-13	1102
2232	636	MURDER ON THE ORIENT EXPRESS	2024-03-28	1100
2233	637	THE HOBBIT	2024-05-07	1099
2234	634	BELOVED	2024-05-18	1098
2235	635	1984	2024-05-21	1097
NULL	NULL	NULL	NULL	NULL

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

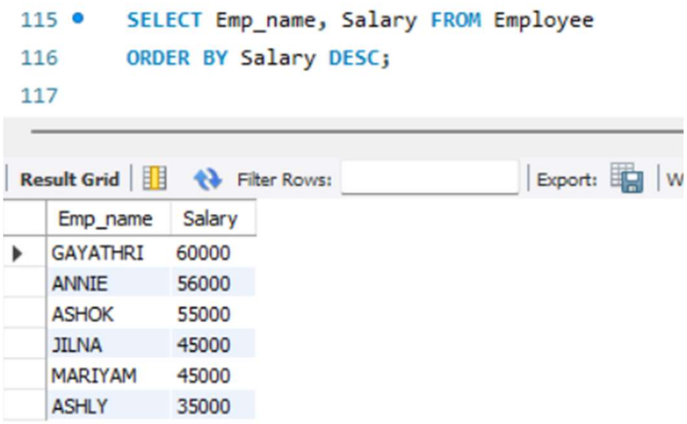
1. Retrieve the book title, category, and rental price of all available books.



```sql
SELECT Book_title, Category, Rental_Price FROM Books
WHERE Status_yn = 'YES';
```

Book_title	Category	Rental_Price
1984	CLASSIC LITERATURE	2500
BELOVED	MODERN FICTION	3999
MURDER ON THE ORIENT EXPRESS	MYSTERY & THRILLER	4599
UNBROKEN	NON-FICTION	7000
WOLF HALL	HISTORICAL FICTION	6300

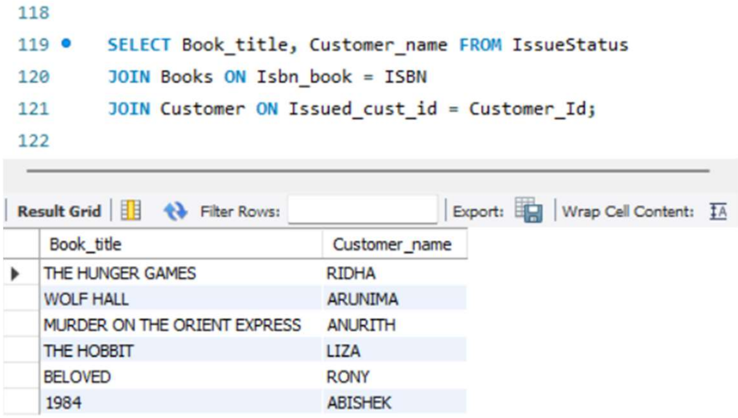
2. List the employee names and their respective salaries in descending order of salary.



```
SELECT Emp_name, Salary FROM Employee
ORDER BY Salary DESC;
```

Emp_name	Salary
GAYATHRI	60000
ANNIE	56000
ASHOK	55000
JILNA	45000
MARIYAM	45000
ASHLY	35000

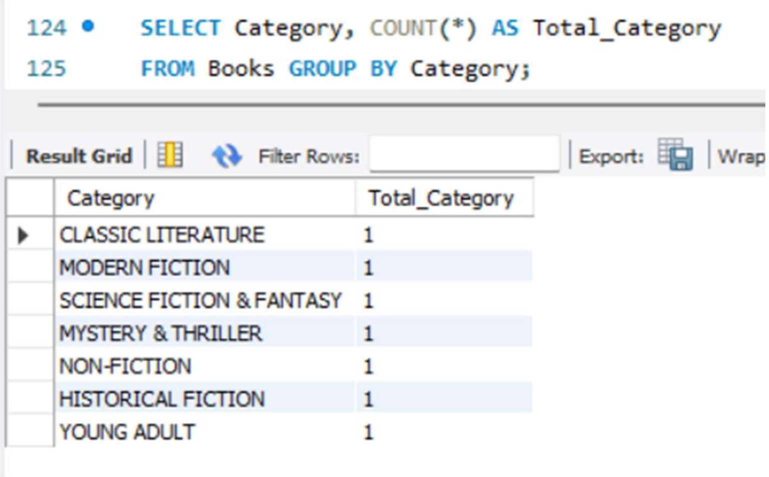
3. Retrieve the book titles and the corresponding customers who have issued those books.



```
SELECT Book_title, Customer_name FROM IssueStatus
JOIN Books ON Isbn_book = ISBN
JOIN Customer ON Issued_cust_id = Customer_Id;
```

Book_title	Customer_name
THE HUNGER GAMES	RIDHA
WOLF HALL	ARUNIMA
MURDER ON THE ORIENT EXPRESS	ANURITH
THE HOBBIT	LIZA
BELOVED	RONY
1984	ABISHEK

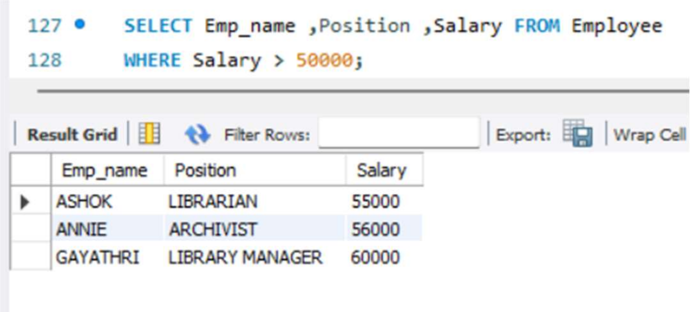
4. Display the total count of books in each category.



```
SELECT Category, COUNT(*) AS Total_Category
FROM Books GROUP BY Category;
```

Category	Total_Category
CLASSIC LITERATURE	1
MODERN FICTION	1
SCIENCE FICTION & FANTASY	1
MYSTERY & THRILLER	1
NON-FICTION	1
HISTORICAL FICTION	1
YOUNG ADULT	1

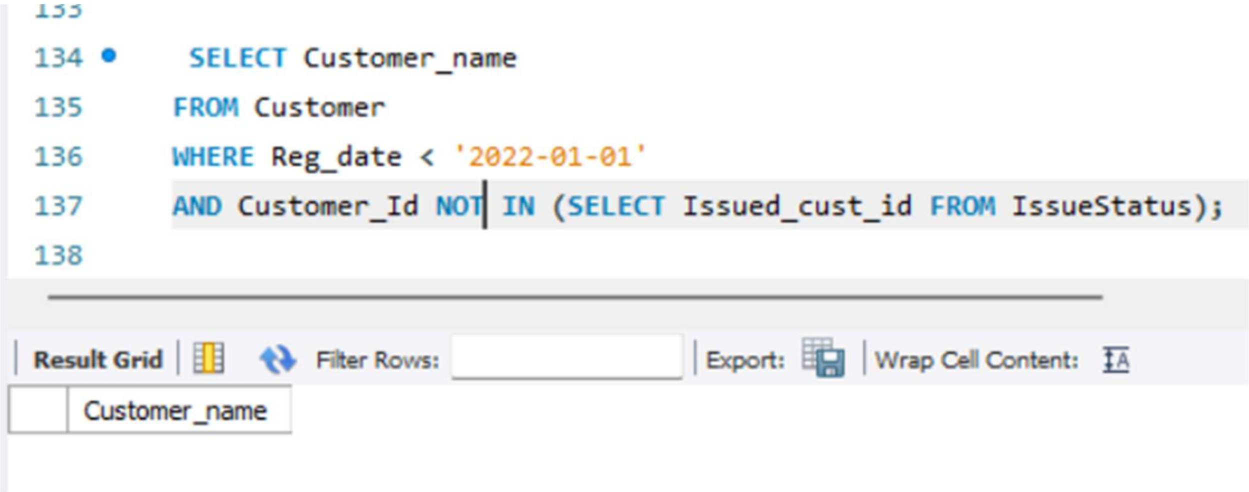
5. Retrieve the employee names and their positions for the employees whose salaries are above Rs.50,000.



```sql
SELECT Emp_name ,Position ,Salary FROM Employee
WHERE Salary > 50000;
```

Emp_name	Position	Salary
ASHOK	LIBRARIAN	55000
ANNIE	ARCHIVIST	56000
GAYATHRI	LIBRARY MANAGER	60000

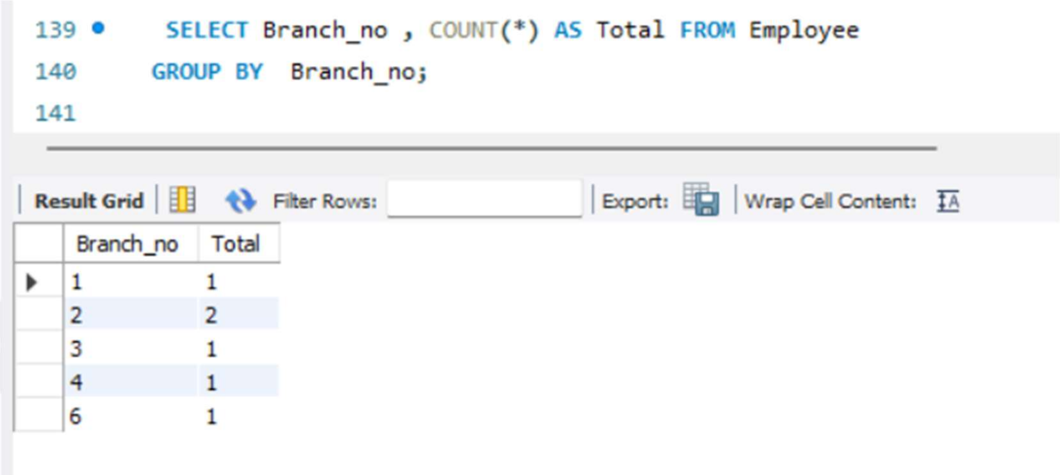
6. List the customer names who registered before 2022-01-01 and have not issued any books yet.



```sql
 SELECT Customer_name
FROM Customer
WHERE Reg_date < '2022-01-01'
AND Customer_Id NOT IN (SELECT Issued_cust_id FROM IssueStatus);
```

Customer_name

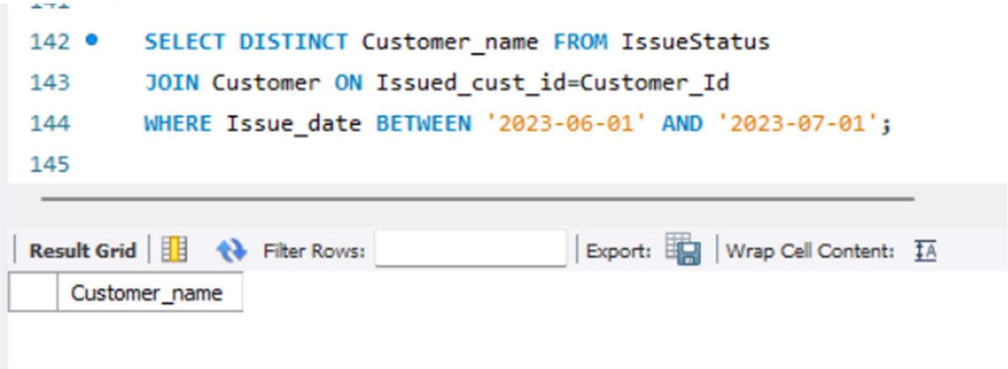
7. Display the branch numbers and the total count of employees in each branch.



```sql
 SELECT Branch_no , COUNT(*) AS Total FROM Employee
GROUP BY  Branch_no;
```

Branch_no	Total
1	1
2	2
3	1
4	1
6	1

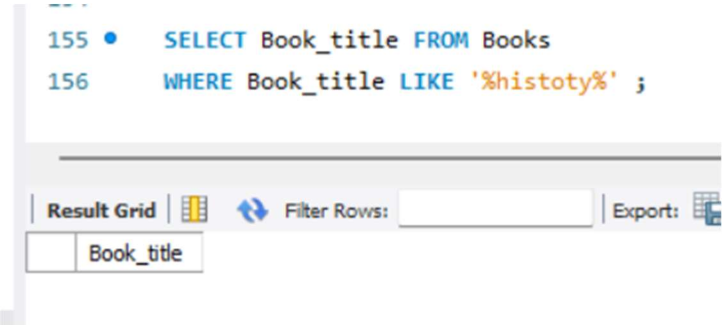
8. Display the names of customers who have issued books in the month of June 2023.



```
SELECT DISTINCT Customer_name FROM IssueStatus
JOIN Customer ON Issued_cust_id=Customer_Id
WHERE Issue_date BETWEEN '2023-06-01' AND '2023-07-01';
```

	Customer_name

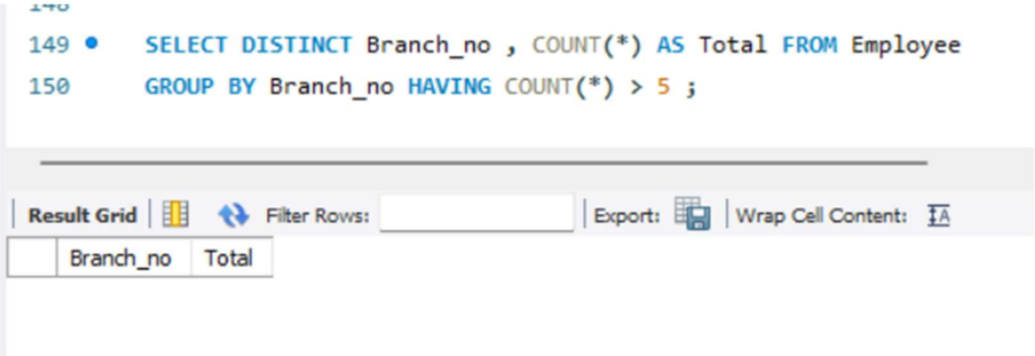
9. Retrieve book_title from book table containing history.



```
SELECT Book_title FROM Books
WHERE Book_title LIKE '%histoty%' ;
```

	Book_title

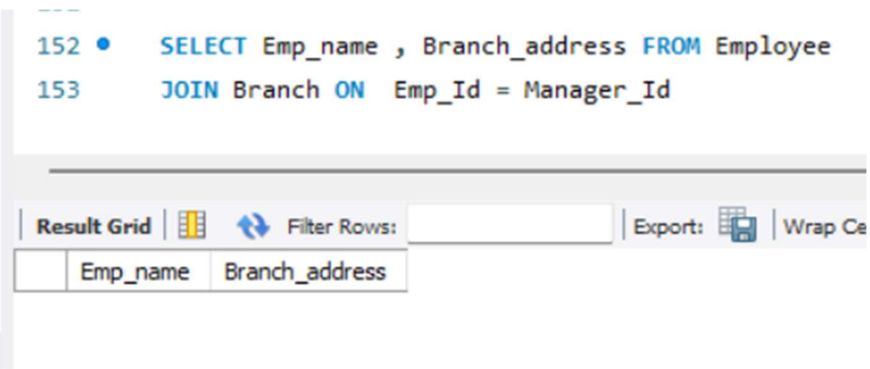
10.Retrieve the branch numbers along with the count of employees for branches having more than 5 employees



```
SELECT DISTINCT Branch_no , COUNT(*) AS Total FROM Employee
GROUP BY Branch_no HAVING COUNT(*) > 5 ;
```

	Branch_no	Total

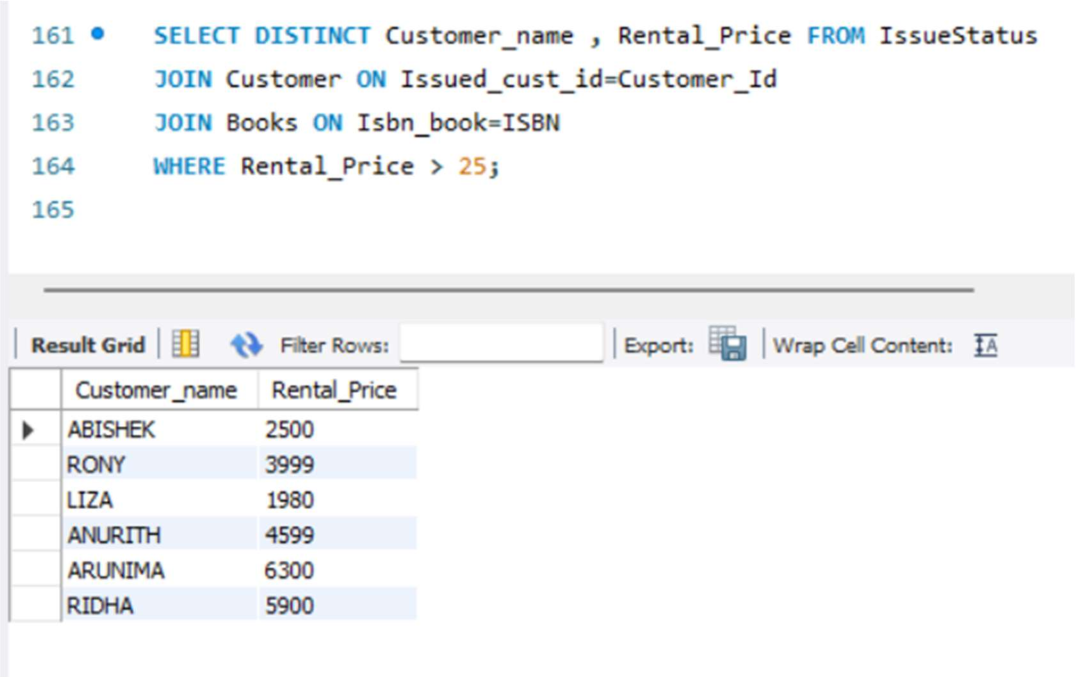
11. Retrieve the names of employees who manage branches and their respective branch addresses.



```sql
SELECT Emp_name , Branch_address FROM Employee
JOIN Branch ON  Emp_Id = Manager_Id
```

Emp_name	Branch_address

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



```sql
SELECT DISTINCT Customer_name , Rental_Price FROM IssueStatus
JOIN Customer ON Issued_cust_id=Customer_Id
JOIN Books ON Isbn_book=ISBN
WHERE Rental_Price > 25;
```

Customer_name	Rental_Price
ABISHEK	2500
RONY	3999
LIZA	1980
ANURITH	4599
ARUNIMA	6300
RIDHA	5900