# **Library Management System**

This is a project based on the 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. Creation of a DATABASE named library and following TABLES in the database:

#### **Tables and their attributes**:

#### 1. Branch

Branch no - Set as PRIMARY KEY

Manager\_Id

Branch\_address

Contact no

#### 2. Employee

Emp Id – Set as PRIMARY KEY

Emp\_name

Emp\_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

Stock 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 -Set as FOREIGN KEY and it refer customer id in CUSTOMER table

Return\_book\_name

Return date

Isbn book2 - Set as FOREIGN KEY and it should refer isbn in BOOKS table

# Create the library database and use the library database :

```
-- Create the library database

14 • create database library;

15

16 -- Use the library database

17 • use library;
```

Creation of the required tables and inserting values to the corresponding tables :

#### 1. Table Branch

```
-- Create the Branch table
19
20 • ⊖ create table Branch ( Branch_no int PRIMARY KEY,
                         Manager Id int,
21
                         Branch address varchar(100),
22
23
                         Contact_no varchar(10)
24
25
        -- Insert data into Branch table
        insert into Branch values
        (1, 1001, 'Main Street', '1234567890'),
        (2, 1002, 'Elm Street', '9876543210'),
28
        (3, 1003, 'Oak Avenue', '1112223333'),
29
        (4, 1004, 'Pink Street', '4445556666'),
30
        (5, 1005, 'Maple Avenue', '7778889999'),
31
        (6, 1006, 'Cedar Street', '0001112222'),
        (7, 1007, 'Birch Avenue', '3334445555'),
33
        (8, 1008, 'Walnut Street', '6667778888'),
34
        (9, 1009, 'Cherry Avenue', '9990001111'),
35
        (10, 1010, 'Pinecone Street', '2223334444');
36
```

### 2. Table Employee

```
41
       -- Create the Employee table
42 • ⊖ create table Employee ( Emp_Id int PRIMARY KEY,
43
                                Emp Name varchar(20),
44
                                Emp Position varchar(25),
45
                                Salary decimal(10,2),
46
                                Branch_no int,
                                FOREIGN KEY (Branch_no) REFERENCES Branch(Branch_no)
47
48
                             );
49
       -- Insert data into Employee table
50
51 •
       insert into Employee values
       (1, 'John Michael', 'Librarian', 55000.00, 1),
52
       (2, 'Smith Jane', 'Assistant Librarian', 48000.00, 1),
53
54
       (3, 'Johnson Doe', 'Cataloguer', 60000.00, 2),
       (4, 'Emily Brown', 'Cataloguer', 40000.00, 8),
55
       (5, 'David Wilson', 'Assistant Librarian', 38000.00, 3),
56
57
       (6, 'Sarah Miller', 'Shelver', 35000.00, 3),
       (7, 'Robert Jones', 'Assistant Librarian', 38000.00, 1),
58
       (8, 'Jennifer Davis', 'Librarian', 45000.00, 4),
59
       (9, 'William Taylor', 'Cataloguer', 40000.00, 4),
60
61
       (10, 'Jessica Martinez', 'Assistant Manager', 50000.00, 5),
62
       (11, 'Alexander Smith', 'Manager', 60000.00, 1),
       (12, 'Erick Doe', 'Assistant Manager', 50000.00, 1),
63
       (13, 'Michael Johnson', 'Librarian', 45000.00, 7),
64
       (14, 'Kiran Susmith', 'Manager', 60000.00, 1),
65
       (15, 'Farhan Khan', 'Assistant Manager', 50000.00, 1);
66
```

#### 3. Table Books

```
-- Create the Books table
71
72 • ⊖ create table Books ( ISBN bigint PRIMARY KEY,
                            Book title varchar(100),
73
74
                            Category varchar(20),
                            Rental Price decimal(10,2),
75
                            Stock_Status_enum('Yes','No'),
76
77
                            Author varchar(25),
78
                            Publisher varchar(50)
79
                        );
80
       -- Insert data into Books table
81
82
       INSERT INTO Books (ISBN, Book title, Category, Rental Price, Stock Status, Author, Publisher)
83
       VALUES
       (2345678912, 'Sapiens: A Brief History of Humankind', 'History', 14.99, 'Yes', 'Yuval Noah Harari', 'Harper'),
84
       (3456789123, 'Guns, Germs, and Steel: The Fates of Human Societies', 'History', 16.99, 'Yes', 'Jared Diamond', 'W. W. Norton & Company'),
85
       (4567891234234, 'The Sixth Extinction: An Unnatural History', 'Science', 15.99, 'Yes', 'Elizabeth Kolbert', 'Henry Holt and Company'),
86
       (5678912345, 'A Short History of Nearly Everything', 'Science', 17.99, 'Yes', 'Bill Bryson', 'Broadway Books'),
87
       (1234567890, 'The Great Gatsby', 'Fiction', 15.99, 'Yes', 'F. Scott Fitzgerald', 'Scribner'),
88
       (2345678901567, 'To Kill a Mockingbird', 'Fiction', 12.99, 'Yes', 'Harper Lee', 'J. B. Lippincott & Co.'),
89
90
       (3456789012, '1984', 'Fiction', 14.99, 'No', 'George Orwell', 'Secker & Warburg'),
91
       (4567890123539, 'The Catcher in the Rye', 'Fiction', 11.99, 'Yes', 'J.D. Salinger', 'Little, Brown and Company'),
92
       (5678901234, 'Pride and Prejudice', 'Fiction', 13.99, 'Yes', 'Jane Austen', 'T. Egerton, Whitehall'),
       (6789012345176, 'To the Lighthouse', 'Fiction', 16.99, 'No', 'Virginia Woolf', 'The Hogarth Press'),
93
       (7890123456, 'Brave New World', 'Fiction', 17.99, 'Yes', 'Aldous Huxley', 'Chatto & Windus'),
94
95
       (8901234567456, 'Animal Farm', 'Fiction', 10.99, 'Yes', 'George Orwell', 'Secker & Warburg'),
       (9012345678, 'The Hobbit', 'Fantasy', 18.99, 'Yes', 'J.R.R. Tolkien', 'Allen & Unwin'),
96
97
       (1234567891, 'Harry Potter and the Philosopher''s Stone', 'Fantasy', 19.99, 'Yes', 'J.K. Rowling', 'Bloomsbury');
```

#### 4. Table Customer

```
-- Create the Customer table
102
103 • ⊖ create table Customer ( Customer Id int PRIMARY KEY,
104
                               Customer_Name varchar(20),
                               Customer Address varchar(100),
105
                               Reg date date
106
                            );
107
108
        -- Insert data into Customer table
109
        insert into Customer values
110
        (1, 'Jerin Johnson', 'Oak Avenue', '2021-12-01'),
111
112
        (2, 'Ishoya Smith', '567 Pine Street', '2022-02-15'),
        (3, 'Jennifer Justin', 'Main Street, Cityville, USA', '2023-01-15'),
113
        (4, 'Idom Smith', 'Elm Street, Townsville, USA', '2022-11-30'),
114
        (5, 'Charlie Brown', 'Oak Avenue, Villageton, USA', '2023-03-10'),
115
        (6, 'Diana Martinez', '101 Maple Lane, Hamletville, USA', '2023-02-05'),
116
        (7, 'Emily Wilson', 'Pine Street, Countryside, USA', '2022-09-20'),
117
        (8, 'Frank Miller', 'Cedar Avenue, Lakeside, USA', '2022-12-18'),
118
        (9, 'Grace Taylor', '404 Oak Street, Hilltop, USA', '2023-04-25'),
119
        (10, 'Henry Clark', '505 Elm Avenue, Riversend, USA', '2023-05-12'),
120
        (11, 'Isabel Garcia', 'Birch Lane, Mountainview, USA', '2020-10-08'),
121
        (12, 'Jack Harris', 'Pine Avenue, Lakeshore, USA', '2022-08-15');
122
```

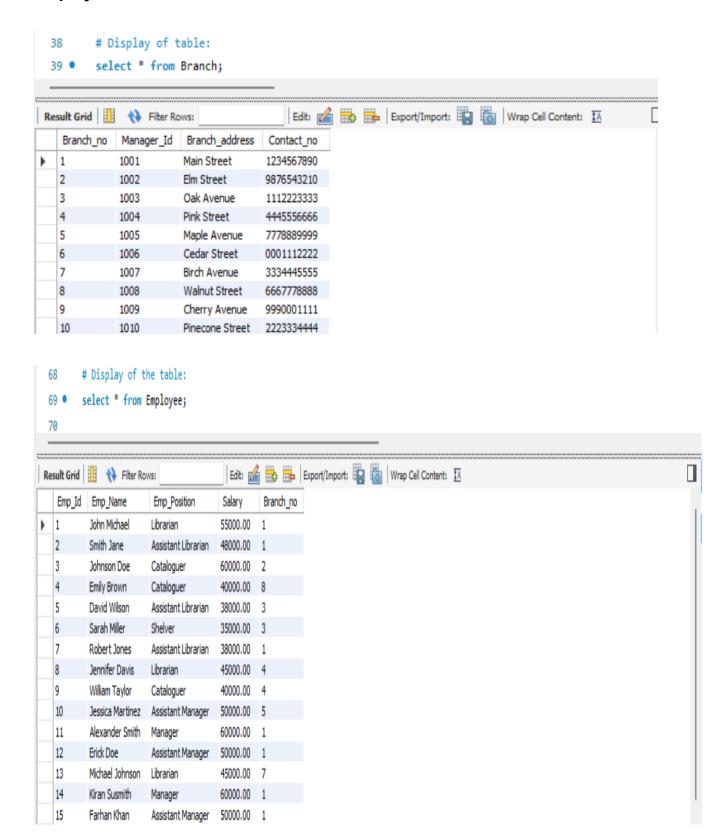
# 5. Table IssueStatus

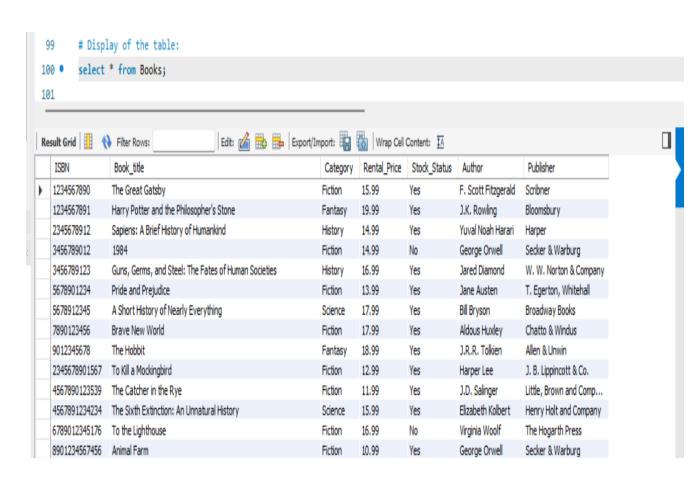
```
127
        -- Create the IssueStatus table
128 • ○ create table IssueStatus ( Issue Id int PRIMARY KEY,
                                  Issued cust int,
129
                                  Issued_book_name varchar(100),
130
                                  Issue date date,
131
132
                                  Isbn book bigint,
                                  FOREIGN KEY(Issued_cust) REFERENCES customer(Customer_Id),
133
                                  FOREIGN KEY(Isbn book) REFERENCES books(ISBN)
134
135
                                  );
136
137
        -- Insert data into IssueStatus table
        INSERT INTO IssueStatus (Issue Id, Issued cust, Issued book name, Issue date, Isbn book)
138
139
        VALUES
        (1, 10, 'The Great Gatsby', '2023-01-20', 1234567890),
140
        (2, 3, 'To Kill a Mockingbird', '2023-02-05', 2345678901567),
141
        (3, 2, 'Sapiens: A Brief History of Humankind', '2023-03-15', 2345678912),
142
        (4, 4, '1984', '2023-04-10', 3456789012),
143
        (5, 12, 'The Hobbit', '2023-05-20', 9012345678),
144
        (6, 7, 'To the Lighthouse', '2023-06-05', 6789012345176),
145
        (7, 6, 'Pride and Prejudice', '2023-07-10', 5678901234),
146
        (8, 9, 'Brave New World', '2023-08-15', 7890123456),
147
        (9, 5, 'The Catcher in the Rye', '2023-09-20', 4567890123539),
148
        (10, 1, 'Harry Potter and the Philosopher''s Stone', '2023-10-25', 1234567891);
149
150
```

# 6. Table ReturnStatus

```
-- Create the ReturnStatus table
154
155 • ⊖ create table ReturnStatus ( Return Id int PRIMARY KEY,
156
                                    Return cust int,
                                    Return_book_name_varchar(100),
157
                                    Return date date,
158
                                    Isbn book2 bigint,
159
                                    FOREIGN KEY (Return_cust) REFERENCES customer(Customer_Id),
160
                                    FOREIGN KEY (Isbn_book2) REFERENCES books(ISBN)
161
162
                                    );
163
        -- Insert data into ReturnStatus table
164
        insert into ReturnStatus values
165
        (1, 10, 'The Great Gatsby', '2023-01-25', 1234567890),
166
        (2, 3, 'To Kill a Mockingbird', '2023-02-15', 2345678901567),
167
        (3, 2, 'Sapiens: A Brief History of Humankind', '2023-03-25', 2345678912),
168
        (4, 4, '1984', '2023-04-20', 3456789012),
169
        (5, 12, 'The Hobbit', '2023-05-25', 9012345678),
170
        (6, 7, 'To the Lighthouse', '2023-06-15', 6789012345176),
171
        (7, 6, 'Pride and Prejudice', '2023-07-20', 5678901234),
172
173
        (8, 9, 'Brave New World', '2023-08-25', 7890123456),
        (9, 5, 'The Catcher in the Rye', '2023-09-30', 4567890123539),
174
        (10, 1, 'Harry Potter and the Philosopher''s Stone', '2023-10-30', 1234567891);
175
176
```

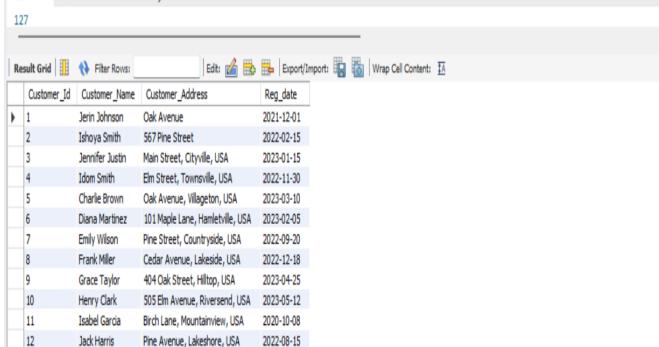
# **Display of tables:**

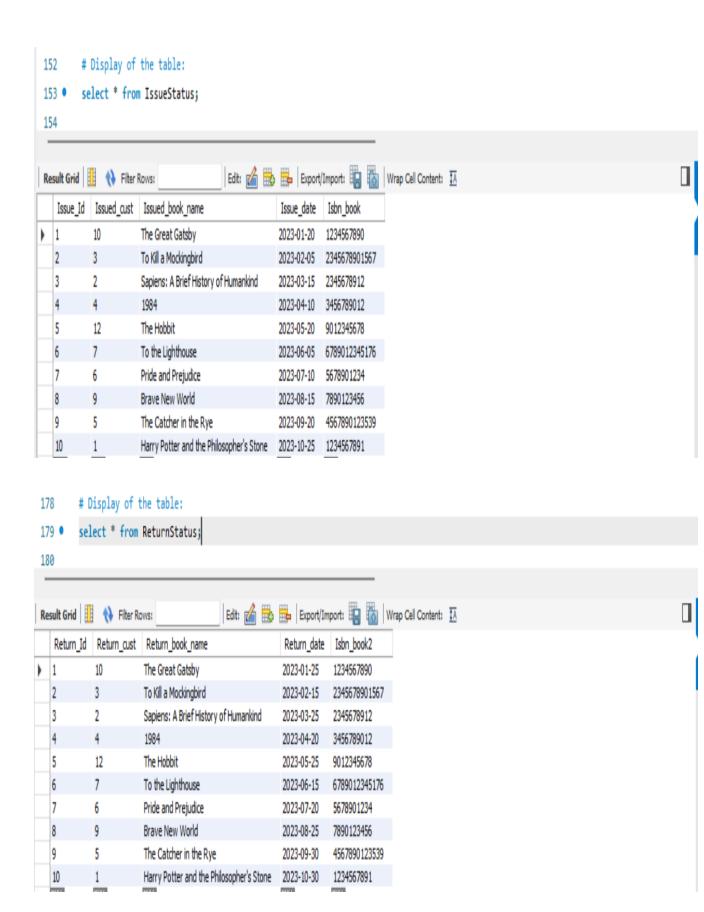






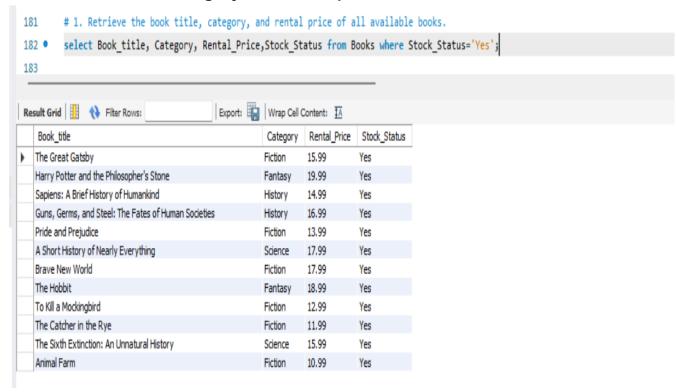
126 • select \* from Customer;



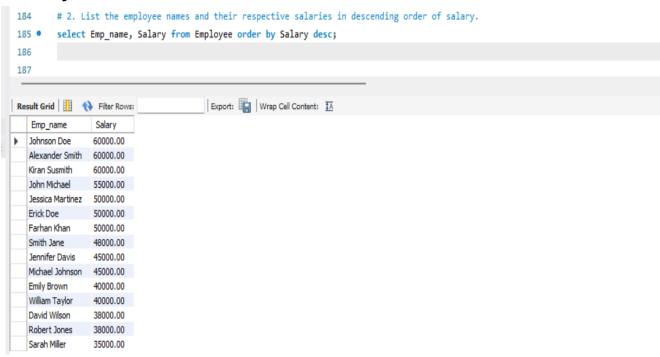


# Write the queries for the following:

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



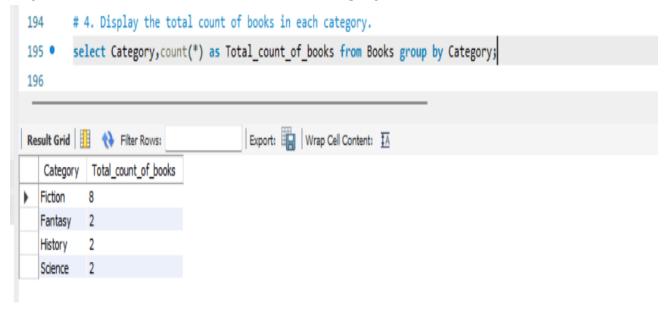
2. List the employee names and their respective salaries in descending order of salary.



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



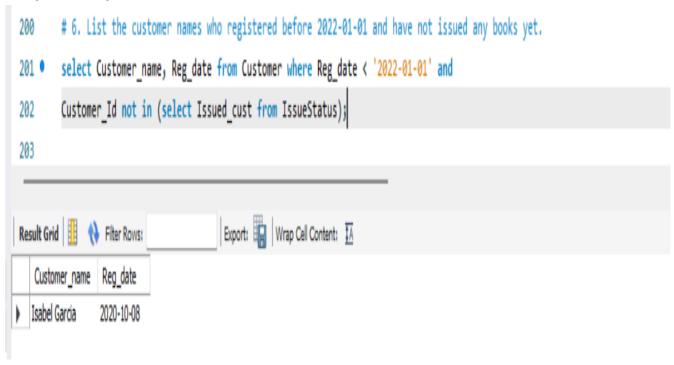
# 4. Display the total count of books in each category.



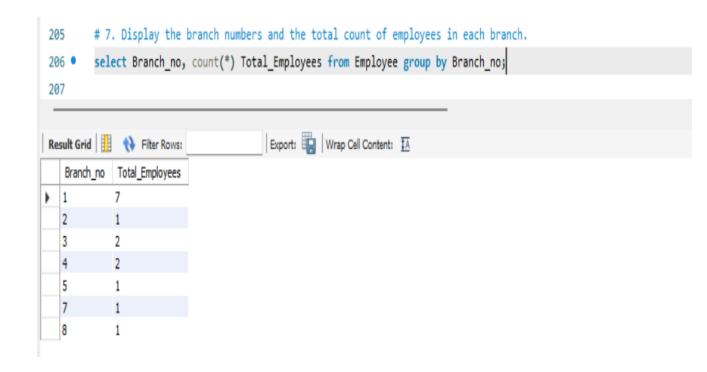
5. Retrieve the employee names and their positions for the employees whose salaries are above Rs.50,000.



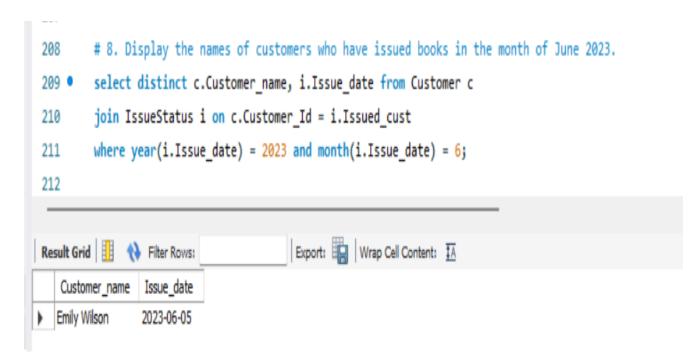
6. List the customer names who registered before 2022-01-01 and have not issued any books yet.



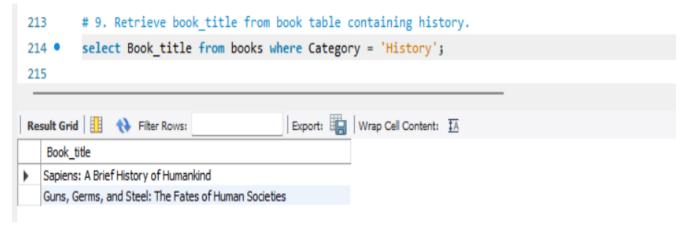
7. Display the branch numbers and the total count of employees in each branch.



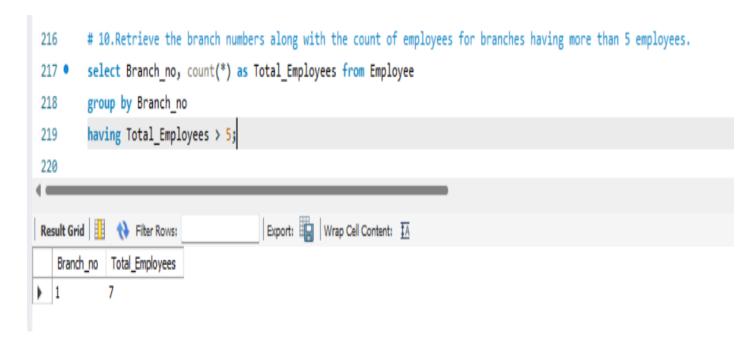
8. Display the names of customers who have issued books in the month of June 2023.



### 9. Retrieve book\_title from a book table containing history.



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



### **Conclusion:**

The system provides functionalities for book issuance, returns, employee management, and customer registration, among others. This system enables users to perform various queries and operations to efficiently manage library resources and services. The queries outlined in the project requirements allow us to retrieve valuable insights and perform necessary actions within the library management system.