MySQL PROJECT

Topic : 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

Attributes for the tables:

1. Branch

Branch_no - Set as PRIMARY KEY
Manager_Id
Branch_address
Contact_no

```sql
);

-- Insert 10 rows into the branch table
INSERT INTO Branch (Branch_no, Manager_Id, Branch_address, Contact_no)
VALUES
(1, 101, '123_Main_St','555-1234'),
(2, 102, '456 Elm St','555-5678'),
(3, 103, '789 Oak St','555-9012'),
(4, 104, '321 Pine St','555-3456'),
(5, 105, '654 Maple St','555-7890'),
(6, 106, '987 Cedar St','555-2345'),
(7, 107, '654 Birch St','555-6789'),
(8, 108, '321 Walnut St','555-1234'),
(9, 109, '789 Cherry St','555-5678'),
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

Branch_no	Manager_Id	Branch_address	Contact_no
1	101	123_Main_St	555-1234
2	102	456 Elm St	555-5678
3	103	789 Oak St	555-9012
4	104	321 Pine St	555-3456
5	105	654 Maple St	555-7890
6	106	987 Cedar St	555-2345

Branch 1

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

```
INSERT INTO library.Employee (Emp_Id, Emp_name, Position, Salary, Branch_no)
VALUES
    (1, 'John_Davis', 'Librarian', 50000.00, 1),
    (2, 'Willow_Smith', 'Assistant_Librarian', 40000.00, 2),
    (3, 'Alice_Johnson', 'Cataloger', 35000.00, 3),
    (4, 'Bob_Williams', 'Reference_Librarian', 45000.00, 1),
    (5, 'Emily_Brown', 'Library_Assistant', 30000.00, 2),
    (6, 'Michael_Davis', 'Archivist', 40000.00, 3),
    (7, 'Jessica_Wilson', 'Library_Director', 60000.00, 1),
    (8, 'David_Martinez', 'Children\'s_Librarian', 45000.00, 2),
    (9, 'Sarah_Anderson', 'Acquisitions_Librarian', 45000.00, 3),
    (10, 'Christopher_Taylor', 'Systems_Librarian', 50000.00, 2);
SELECT * FROM Employee;
```

Result Grid

Emp_Id	Emp_name	Position	Salary	Branch_no
1	John_Davis	Librarian	50000.00	1
2	Willow_Smith	Assistant_Librarian	40000.00	2
3	Alice_Johnson	Cataloger	35000.00	3
4	Bob_Williams	Reference_Librarian	45000.00	1
5	Emily_Brown	Library_Assistant	30000.00	2

Employee 2

Output

Action Output

#	Time	Action	Message
17	16:21:05	INSERT INTO library.Employee (Emp_Id, Emp_name, Position, Salary, Branch_no) VALU...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0
18	16:21:11	SELECT * FROM Employee LIMIT 0, 1000	10 row(s) returned

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

Views | Mysql task 4.* | mysql task.5* | mysql task 6* | mysql task 7 | stored procedures* | Triggers* | mysql task 8* | mysql task 9* | mysql task 10* | mysql pr*

Limit to 1000 rows

```
VALUES
    ('9780679601085', 'The Prophet', 'Fiction', 10.99, 'yes', 'Kahlil Gibran', 'Alfred A. Knopf'),
    ('9780141439662', 'Sense and Sensibility', 'Classic', 8.99, 'yes', 'Jane Austen', 'Penguin Classics'),
    ('9780062316110', 'Sapiens: A Brief History of Humankind', 'Non-fiction', 12.99, 'yes', 'Yuval Noah Harari', 'Harper'),
    ('9788172235652', 'God of Small Things', 'Fiction', 11.99, 'yes', 'Arundhati Roy', 'IndiaInk'),
    ('9788170289186', 'Gitanjali', 'Poetry', 9.99, 'yes', 'Rabindranath Tagore', 'Rupa & Co.'),
    ('9780307947055', '1984', 'Classic', 9.99, 'yes', 'George Orwell', 'Vintage'),
    ('9780061120084', 'To Kill a Mockingbird', 'Classic', 10.99, 'yes', 'Harper Lee', 'Harper Perennial Modern Classics'),
    ('9780307277671', 'The Catcher in the Rye', 'Fiction', 8.99, 'yes', 'J.D. Salinger', 'Little, Brown and Company'),
    ('9780307477729', 'The Girl with the Dragon Tattoo', 'Mystery', 11.99, 'yes', 'Stieg Larsson', 'Vintage Crime/Black Lizard'),
    ('9781400031702', 'The Da Vinci Code', 'Mystery', 10.99, 'yes', 'Dan Brown', 'Anchor');

select * from books;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

ISBN	Book_title	Category	Rental_Price	Status	Author	Publisher
9780061120084	To Kill a Mockingbird	Classic	10.99	yes	Harper Lee	Harper Perennial Modern Classics
9780062316110	Sapiens: A Brief History of Humankind	Non-fiction	12.99	yes	Yuval Noah Harari	Harper
9780141439662	Sense and Sensibility	Classic	8.99	yes	Jane Austen	Penguin Classics
9780307277671	The Catcher in the Rye	Fiction	8.99	yes	J.D. Salinger	Little, Brown and Company
9780307477729	The Girl with the Dragon Tattoo	Mystery	11.99	yes	Stieg Larsson	Vintage Crime/Black Lizard

books 4 | Apply | Reve

Output

Action Output

#	Time	Action	Message	Duration / Fetch
23	16:24:20	INSERT INTO library.Books (ISBN, Book_title, Category, Rental_Price, Status, Author, Pu...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0	0.015 sec
24	16:24:39	select * from books LIMIT 0, 1000	10 row(s) returned	0.000 sec / 0.000 se

4. Customer

Customer_Id - Set as PRIMARY KEY
Customer_name
Customer_address
Reg_date

```
VALUES
    (1, 'Homer Simpson', '742 Evergreen Terrace, Springfield', '2023-01-15'),
    (2, 'Phoebe Buffay', 'Apartment 20, 495 Grove Street, New York', '2023-02-28'),
    (3, 'Barney Stinson', 'Apartment 4C, 79th Street, New York', '2023-03-10'),
    (4, 'Chandler Bing', 'Apartment 19, 15 Yemen Road, Yemen', '2023-04-05'),
    (5, 'Michael Scott', 'Dunder Mifflin Paper Company, Scranton', '2023-05-20'),
    (6, 'Leslie Knope', 'City Hall, Pawnee', '2023-06-12'),
    (7, 'Sheldon Cooper', 'Apartment 4A, 2311 Los Robles Avenue, Pasadena', '2023-07-18'),
    (8, 'Phyllis Vance', 'Dunder Mifflin Paper Company, Scranton', '2023-08-02'),
    (9, 'Ron Swanson', 'City Hall, Pawnee', '2023-09-09'),
    (10, 'Monica Geller', 'Apartment 20, 495 Grove Street, New York', '2023-10-30');

select * from customer;
```

Customer_Id	Customer_name	Customer_address	Reg_date
1	Homer Simpson	742 Evergreen Terrace, Springfield	2023-01-15
2	Phoebe Buffay	Apartment 20, 495 Grove Street, New York	2023-02-28
3	Barney Stinson	Apartment 4C, 79th Street, New York	2023-03-10
4	Chandler Bing	Apartment 19, 15 Yemen Road, Yemen	2023-04-05
5	Michael Scott	Dunder Mifflin Paper Company, Scranton	2023-05-20

#	Time	Action	Message
20	16:22:14	INSERT INTO library.Customer (Customer_Id, Customer_name, Customer_address, Reg_...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0
21	16:23:15	select * from customer LIMIT 0, 1000	10 row(s) returned

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

```
VALUES
    (1, 1, 'The Prophet', '2024-02-01', '9780679601085'),
    (2, 2, 'Sense and Sensibility', '2024-02-02', '9780141439662'),
    (3, 3, 'Sapiens: A Brief History of Humankind', '2024-02-03', '9780062316110'),
    (4, 4, 'God of Small Things', '2024-02-04', '9788172235652'),
    (5, 5, 'Gitanjali', '2024-02-05', '9788170289186'),
    (6, 6, '1984', '2024-02-06', '9780307947055'),
    (7, 7, 'To Kill a Mockingbird', '2024-02-07', '9780061120084'),
    (8, 8, 'The Catcher in the Rye', '2024-02-08', '9780307277671'),
    (9, 9, 'The Girl with the Dragon Tattoo', '2024-02-09', '9780307477729'),
    (10, 10, 'The Da Vinci Code', '2024-02-10', '9781400031702');

select * from issuestatus;
```

Result Grid

Issue_Id	Issued_cust	Issued_book_name	Issue_date	Isbn_book
1	1	The Prophet	2024-02-01	9780679601085
2	2	Sense and Sensibility	2024-02-02	9780141439662
3	3	Sapiens: A Brief History of Humankind	2024-02-03	9780062316110
4	4	God of Small Things	2024-02-04	9788172235652
5	5	Gitanjali	2024-02-05	9788170289186

Output

Action Output

#	Time	Action	Message
26	16:25:36	INSERT INTO library.IssueStatus (Issue_Id, Issued_cust, Issued_book_name, Issue_date...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0
27	16:26:01	select * from issuestatus LIMIT 0, 1000	10 row(s) returned

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

```
);

-- Insert 6 rows into the ReturnStatus table
INSERT INTO library.ReturnStatus (Return_Id, Return_cust, Return_book_name, Return_date, Isbn_book2)
VALUES
    (1, 1, 'The Prophet', '2024-02-11', '9780679601085'),
    (2, 2, 'Sense and Sensibility', '2024-02-12', '9780141439662'),
    (3, 3, 'Sapiens: A Brief History of Humankind', '2024-02-13', '9780062316110'),
    (4, 4, 'God of Small Things', '2024-02-14', '9788172235652'),
    (5, 5, 'Gitanjali', '2024-02-15', '9788170289186'),
    (6, 6, '1984', '2024-02-16', '9780307947055');

    select * from returnstatus;
```

Result Grid

Return_Id	Return_cust	Return_book_name	Return_date	Isbn_book2
1	1	The Prophet	2024-02-11	9780679601085
2	2	Sense and Sensibility	2024-02-12	9780141439662
3	3	Sapiens: A Brief History of Humankind	2024-02-13	9780062316110
4	4	God of Small Things	2024-02-14	9788172235652
5	5	Gitanjali	2024-02-15	9788170289186

returnstatus 6

Output

Action Output

#	Time	Action	Message
29	16:26:57	INSERT INTO library.ReturnStatus (Return_Id, Return_cust, Return_book_name, Return...	6 row(s) affected Records: 6 Duplicates: 0 Warnings: 0
30	16:27:18	select * from returnstatus LIMIT 0, 1000	6 row(s) returned

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

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

Limit to 1000 rows

Execute the statement under the keyboard cursor

```
-- and rental price of all available books.
SELECT Book_title, Category, Rental_Price FROM library.Books WHERE Status = 'yes';

-- 2. List the employee names and their respective salaries in descending order of salary.
SELECT Emp_name, Salary FROM library.Employee ORDER BY Salary DESC;

-- 3. Retrieve the book titles and the corresponding customers who have issued those books.
SELECT Books.Book_title, Customer.Customer_name
FROM library.BOOKS
JOIN library.Issuestatus ON Books.ISBN = Issuestatus.Isbn_book
JOIN library.Customer ON Customer.Customer_id = Issuestatus.Issued_cust;
```

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

Book_title	Category	Rental_Price
To Kill a Mockingbird	Classic	10.99
Sapiens: A Brief History of Humankind	Non-fiction	12.99
Sense and Sensibility	Classic	8.99
The Catcher in the Rye	Fiction	8.99
The Girl with the Dragon Tattoo	Mystery	11.99

Books 7

Output

Action Output

#	Time	Action	Message
30	16:27:18	select * from returnstatus LIMIT 0, 1000	6 row(s) returned

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

```sql
-- 1.Retrieve the book title, category, and rental price of all available books.
SELECT Book_title, Category, Rental_Price FROM library.Books WHERE Status = 'yes';

-- 2. List the employee names and their respective salaries in descending order of salar
SELECT Emp_name, Salary FROM library.Employee ORDER BY Salary DESC;

-- 3. Retrieve the book titles and the corresponding customers who have issued those bool
SELECT Books.Book_title, Customer.Customer_name
FROM library.BOOKS
JOIN library.Issuestatus ON Books.ISBN = Issuestatus.Isbn_book
JOIN library.Customer ON Customer.Customer_id = Issuestatus.Issued_cust;
```

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

Emp_name	Salary
Jessica_Wilson	60000.00
John_Davis	50000.00
Christopher_Taylor	50000.00
Bob_Williams	45000.00
David_Martinez	45000.00

plovee 8

put

Action Output

#	Time	Action	Message
31	16:28:04	SELECT Book_title, Category, Rental_Price FROM library.Books WHERE Status = 'yes' L...	10 row(s) returned
32	16:28:50	SELECT Emp_name, Salary FROM library.Employee ORDER BY Salary DESC LIMIT 0, 1...	10 row(s) returned

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

```
149
150         -- Execute the statement under the keyboard cursor and rental price of all available books.
151 •   SELECT Book_title, Category, Rental_Price FROM library.Books WHERE Status = 'yes';
152
153     -- 2. List the employee names and their respective salaries in descending order of salary.
154 •   SELECT Emp_name, Salary FROM library.Employee ORDER BY Salary DESC;
155
156     -- 3. Retrieve the book titles and the corresponding customers who have issued those books.
157 •   SELECT Books.Book_title, Customer.Customer_name
158     FROM library.BOOKS
159     JOIN library.Issuestatus ON Books.ISBN = Issuestatus.Isbn_book
160     JOIN library.Customer ON Customer.Customer_id = Issuestatus.Issued_cust;
161
```

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

Book_title	Customer_name
To Kill a Mockingbird	Sheldon Cooper
Sapiens: A Brief History of Humankind	Barney Stinson
Sense and Sensibility	Phoebe Buffay
The Catcher in the Rye	Phyllis Vance
The Girl with the Dragon Tattoo	Ron Swanson

Result 9

Output

Action Output

#	Time	Action	Message
32	16:28:50	SELECT Emp_name, Salary FROM library.Employee ORDER BY Salary DESC LIMIT 0, 1...	10 row(s) returned
33	16:30:03	SELECT Books.Book_title, Customer.Customer_name FROM library.BOOKS JOIN library.I...	10 row(s) returned

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

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

-- 5. Retrieve the employee names and their positions for the employees whose salaries are abov
SELECT Emp_name, Position FROM library.Employee WHERE Salary > 50000;
```

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

Category	Total_Count
Classic	3
Non-fiction	1
Fiction	3
Mystery	2
Poetry	1

ult 10

put

Action Output

#	Time	Action	Message
33	16:30:03	SELECT Books.Book_title, Customer.Customer_name FROM library.BOOKS JOIN library.l...	10 row(s) returned

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.50,0
SELECT Emp_name, Position FROM library.Employee WHERE Salary > 50000;

-- 6. List the customer names who registered before 2022-01-01 and have not issued any books yet.
SELECT Customer_name FROM library.Customer WHERE Reg_date < '2022-01-01' AND Customer_Id NOT IN
(SELECT Issued_cust FROM library.IssueStatus);

-- 7. Display the branch numbers and the total count of employees in each branch.
SELECT Branch_no, COUNT(*) AS Total_Employees
FROM library.Employee GROUP BY Branch_no;
```

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

Emp_name	Position
Jessica_Wilson	Library_Director

Employee 11

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 library.Customer WHERE Reg_date < '2022-01-01' AND Customer_Id NOT IN
(SELECT Issued_cust FROM library.IssueStatus);

-- 7. Display the branch numbers and the total count of employees in each branch.
SELECT Branch_no, COUNT(*) AS Total_Employees
FROM library.Employee GROUP BY Branch_no;
```

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

Customer_name

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

```
-- 7. Display the branch numbers and the total count of employees in each branch.
SELECT Branch_no, COUNT(*) AS Total_Employees
FROM library.Employee GROUP BY Branch_no;

-- 8.Display the names of customers who have issued books in the month of June 2023.
SELECT DISTINCT Customer_name
FROM library.Customer WHERE Customer_Id IN (
    SELECT Issued_cust
```

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

Branch_no	Total_Employees
1	3
2	4
3	3

Result 13

Output

Action Output

#	Time	Action	Message
36	16:32:30	SELECT Customer_name FROM library.Customer WHERE Reg_date < '2022-01-01' AND...	0 row(s) returned

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

```sql
-- 8.Display the names of customers who have issued books in the month of June 2023.
SELECT DISTINCT Customer_name
FROM library.Customer WHERE Customer_Id IN (
    SELECT Issued_cust
    FROM library.IssueStatus
    WHERE MONTH(Issue_date) = 6 AND YEAR(Issue_date) = 2023
);

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

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

Customer_name

Customer 14

Output

Action Output

#	Time	Action	Message
37	16:33:28	SELECT Branch_no, COUNT(*) AS Total_Employees FROM library.Employee GROUP BY...	3 row(s) returned

9. Retrieve book_title from book table containing history.

```
-- 9. Retrieve book_title from book table containing history.
SELECT Book_title from where category="History";
select book_title from books
WHERE book_title LIKE '%history%' or book_title like "Histoty";
```

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

book_title
Sapiens: A Brief History of Humankind

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(Emp_ID) AS Employee_Count
FROM library.Employee
GROUP BY Branch_no
HAVING COUNT(Emp_ID)>5;
```

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

Branch_no	Employee_Count

11.Retrieve the names of customers who have issued books authored by George Orwell.

```
-- 11.Retrieve the names of customers who have issued books authored by George Orwell.

SELECT DISTINCT c.Customer_name
FROM library.Customer c
JOIN library.IssueStatus i ON c.Customer_Id = i.Issued_cust
JOIN library.Books b ON i.Isbn_book = b.ISBN
WHERE b.Author = 'George Orwell';
```

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

Customer_name
Leslie Knope

12.Retrieve the names of customers who have issued books published by Penguin Classics

```
SELECT DISTINCT c.Customer_name
FROM library.Customer c
JOIN library.IssueStatus i ON c.Customer_Id = i.Issued_cust
JOIN library.Books b ON i.Isbn_book = b.ISBN
WHERE b.Author = 'George Orwell';

-- 12.Retrieve the names of customers who have issued books published by Penguin Classics

SELECT DISTINCT c.Customer_name
```

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

Customer_name
Leslie Knope

13.Display the total count of books issued in each month of 2024

```sql
-- 13.Display the total count of books issued in each month of 2024

SELECT MONTH(Issue_date) AS Issue_Month, COUNT(*) AS Total_Books_Issued
FROM library.IssueStatus
WHERE YEAR(Issue_date) = 2024
GROUP BY MONTH(Issue_date);
```

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

Issue_Month	Total_Books_Issued
2	10

14.Retrieve the names of customers who have issued books authored by Jane Austen and have not returned them yet

```
-- 14.Retrieve the names of customers who have issued books authored by Jane Austen and have not returned them yet

SELECT DISTINCT c.Customer_name
FROM library.Customer c
JOIN library.IssueStatus i ON c.Customer_Id = i.Issued_cust
JOIN library.Books b ON i.Isbn_book = b.ISBN
WHERE b.Author = 'Jane Austen'
AND c.Customer_Id NOT IN (SELECT Return_cust FROM library.ReturnStatus);
```

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

Customer_name

15.Retrieve the names of employees who are not managers.

```sql
-- 15.Retrieve the names of employees who are not managers.

SELECT Emp_name
FROM library.Employee
WHERE Emp_Id NOT IN (SELECT Manager_Id FROM library.Branch);

-- 16.Display the total count of books issued by each customer.

SELECT c.Customer_name, COUNT(i.Issue_Id) AS Total_Books_Issued
```

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

Emp_name
John_Davis
Willow_Smith
Alice_Johnson
Bob_Williams
Emily_Brown

Employee 38

16.Display the total count of books issued by each customer.

```
-- 16.Display the total count of books issued by each customer.

SELECT c.Customer_name, COUNT(i.Issue_Id) AS Total_Books_Issued
FROM library.Customer c
LEFT JOIN library.IssueStatus i ON c.Customer_Id = i.Issued_cust
GROUP BY c.Customer_name;

-- 17. List the authors along with the total number of books they have written in the Mystery category.
```

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

Customer_name	Total_Books_Issued
Homer Simpson	1
Phoebe Buffay	1
Barney Stinson	1
Chandler Bing	1
Michael Scott	1

Result 39

17. List the authors along with the total number of books they have written in the Mystery category.

```
-- 17. List the authors along with the total number of books they have written in the Mystery category.

SELECT Author, COUNT(*) AS Total_Books
FROM library.Books
WHERE Category = 'Mystery'
GROUP BY Author;

-- 18.Identify the branches where the average salary of employees is below Rs. 45,000.
```

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

Author	Total_Books
Stieg Larsson	1
Dan Brown	1

18.Identify the branches where the average salary of employees is below Rs. 45,000.

```
-- 18.Identify the branches where the average salary of employees is below Rs. 45,000.

select branch_no,avg(salary) as avg_sal from employee group by branch_no having avg(salary)<45000;

-- 19. #Retrieve the details of customers who have issued books separately in the years 2021, 2022, 2023, and 2024.
create view issue_details as
```

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

branch_no	avg_sal
2	41250.000000
3	40000.000000

-- 19. #Retrieve the details of customers who have issued books separately in the years 2021, 2022, 2023, and 2024.

```
-- 19. #Retrieve the details of customers who have issued books separat
create view issue_details as
select c.customer_name,i.issued_cust,i.issue_date,i.isbn_book
from customer c join issuestatus i
on c.customer_id = i.issued_cust;

select * from issue_details;

delimiter $
create procedure issued_year(issue_year int)
begin
        select customer_name,issue_date,isbn_book from issue_details
    where year(issue_date) = issue_year
```

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

customer_name	issue_date	isbn_book
Barney Stinson	2024-02-03	9780062316110
Chandler Bing	2024-02-04	9788172235652
Homer Simpson	2024-02-01	9780679601085
Leslie Knope	2024-02-06	9780307947055
Michael Scott	2024-02-05	9788170289186

lt 41

20. dispaly name and position of employee who have the highest salary.

```
-- 20. #dispaly name and position of employee who have the highest salary.
select emp_name, position,salary from employee where salary= (select max(salary) from employee);
```

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

emp_name	position	salary
John_Davis	Librarian	50000.00
Willow_Smith	Assistant_Librarian	40000.00
Alice_Johnson	Cataloger	35000.00
Bob_Williams	Reference_Librarian	45000.00
Emily_Brown	Library_Assistant	30000.00

employee 42 x

Suchithra.p
D18