جامعة أم القرى
UMM AL-QURA UNIVERSITY

Project Course

Database II

23165306-3

LITERARY LOUNGE

September,2023

Abstract

This report provides a comprehensive overview of the library management system. The system aims to efficiently manage the huge amount of data associated with the library's daily operations. To effectively represent these data relationships, an Enhanced Entity Relationship Diagram (EERD) was developed, which illustrates the complex interactions within the system. EERD serves as a basis for creating a well-organized database schema, ensuring data integrity and efficiency. Normalization techniques were applied to the chart, adhering to normalization principles to eliminate redundancy and improve data consistency. In addition, the report demonstrates the practical implementation of the database and its tables using MySQL, demonstrating the systematic and efficient management of library resources and operations.

Table of Contents

Introduction

1.1 Literary Lounge Library

The "Literary Lounge Library". This space is a treasure trove of literary delights, offering an extensive collection of books spanning various genres and interests. Whether you're into classic novels, contemporary fiction, non-fiction, or even niche subjects, the Literary Lounge Library has something to satisfy every bibliophile's cravings. You can immerse yourself in captivating stories, gain knowledge from informative texts, or simply find a quiet corner to enjoy a good book.

1.2 Literary Lounge description

The Literary Lounge Database Management System is a comprehensive and highly efficient tool meticulously designed to streamline the intricate operations of a modern library. This versatile system accommodates a wide array of users, including staff, library members, and publishers. Within this system, individuals are systematically registered and cataloged, significantly enhancing the functionality and accessibility of the library's vast resources.

Library member are seamlessly integrated into the system, with each member assigned a unique user ID and their personal information, such as their name and contact details, meticulously recorded. At the core of our database are the books, each assigned a distinct Book ID and furnished with vital information like the title, author, price, and details concerning their availability on library shelves.

Publishers, a crucial component of the literary ecosystem, receive their due recognition within our system. Each publisher is allotted a Publisher ID, with their contact information, including their address, carefully documented to accurately acknowledge their contributions to the library.

Library members can effortlessly borrow books, generating precise Return/Issue records for every transaction. These records encompass comprehensive details about the borrowed book, the responsible member, the issue date, and the return date. The system also facilitates financial transactions, allowing members to make payments for various library services. These payment records are meticulously maintained, complete with a unique Payment ID, associated Member ID, and the amount paid.

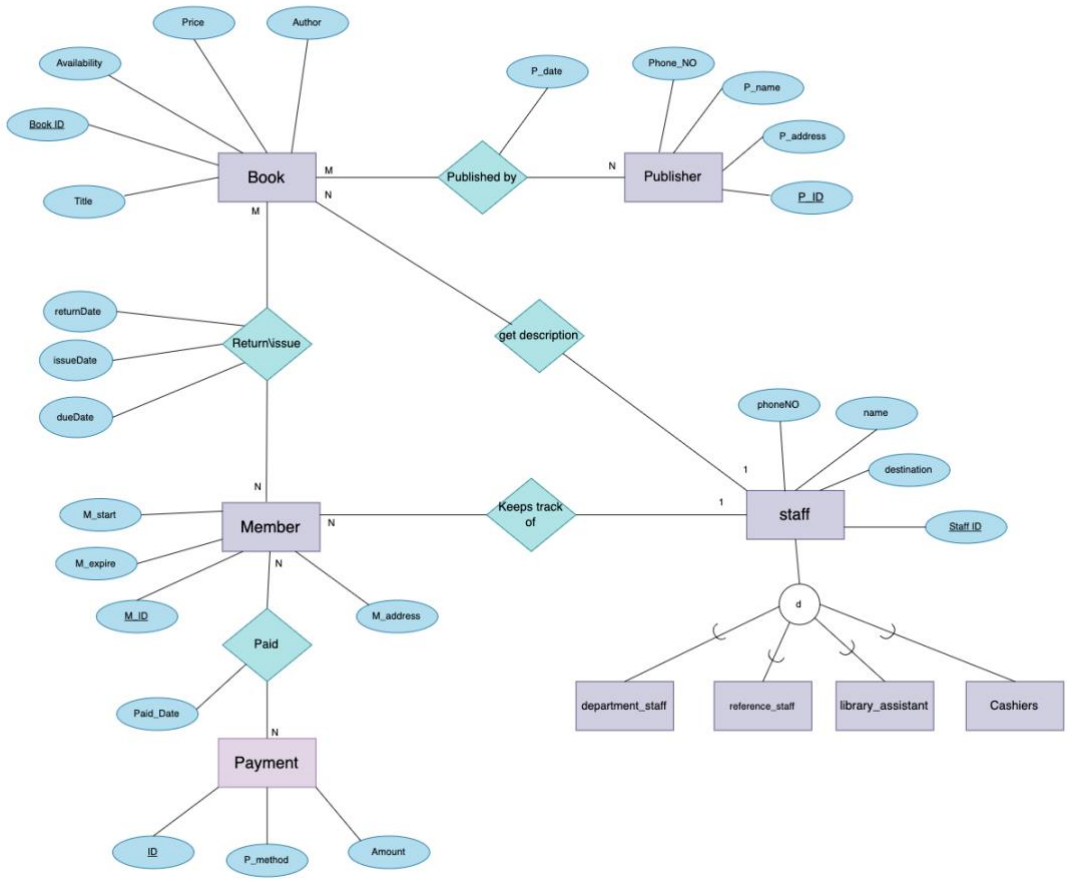
Library staff, another integral aspect of the system, are classified into different roles, including reference staff, assistant librarians, department staff, and cashiers. Each employee is assigned a unique employee ID and a comprehensive profile,

which includes a wide range of personal details such as their name, address, salary, and phone number.

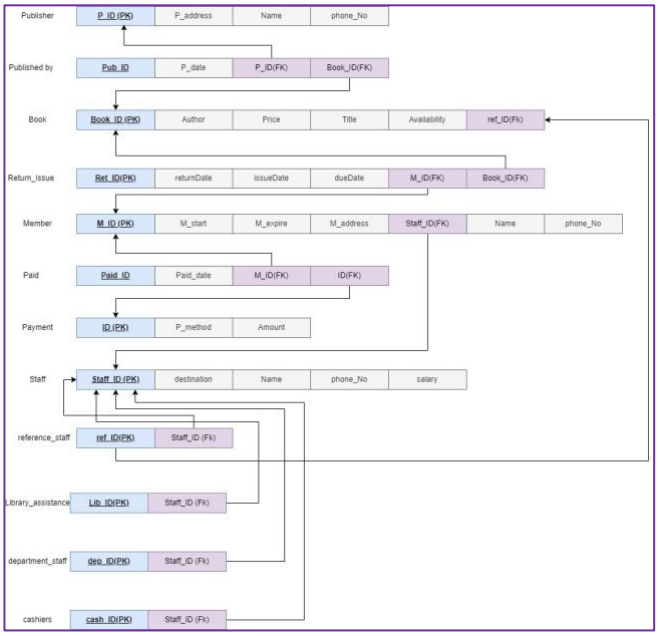
Furthermore, the system maintains essential information about published books, including the Publisher ID, Book ID, and publication date. This feature ensures an efficient tracking and management of library resources, strengthening the overall operation of the library.

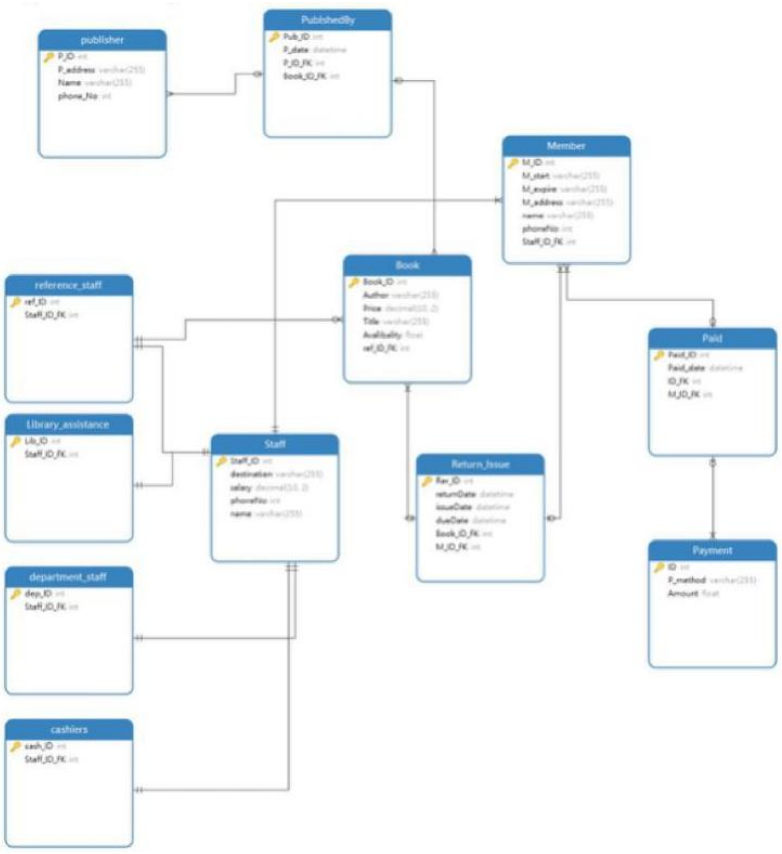
Our Library Database Management System serves as the linchpin for facilitating seamless interactions between all stakeholders. It provides a well-organized repository of books, effectively manages library resources, and fosters an environment that supports both library staff and members. With this system in place, libraries can operate effortlessly and offer the highest level of service to their patrons.

Enhanced Entity Relationship Diagram (EERD)



Database Schema





Normalization

As we notice our system is in 3NF as there is partial or transitive dependency

Publisher(**P_rID**, P_address, User_ID)

P_ID(PK)	P_address	name	phoneNo

Published by(**Pub_ID**, P_date, P_ID,Book_ID)

UserID(PK)	P_date	P_ID(FK)	Book_ID(FK)

Book(**Book_ID**, Author, price,Title, Availability)

Book_ID (PK)	Author	price	Title	Availability	Ref_ID(FK)

Return_Issue(**Rer_ID**, returnDate, issueDate, dueDate, M_ID, Book_ID)

Rer_ID(PK)	returnDate	issueDate	dueDate	M_ID(FK)	Book_ID(FK)

Member(**M_ID**, M_start,M_expire, M_address,staff_ID,User_ID)

M_ID (PK)	M_start	M_expire	M_address	name	phoneNo	staff_ID(FK)

Paid(**Paid_ID**, Paid_date, M_ID, ID)

Paid_ID (PK)	Paid_date	M_ID(FK)	ID(FK)

Payment(**ID**, P_method, Amount)

ID(PK)	P_method	Amount

Staff(**Staff_ID**, destination, User_ID)

Staff_ID (PK)	destination	salary	name	phoneNo

reference_staff (**ref_ID**, Staff_ID)

ref_ID (PK)	Staff_ID (FK)

Library_assistance (**Lib_ID**, Staff_ID)

Lib_ID (PK)	Staff_ID (FK)

department_staff (**dep_ID**, Staff_ID)

dep_ID (PK)	Staff_ID (FK)

cashiers (**cash_ID**, Staff_ID)

cash_ID (PK)	Staff_ID (FK)

Implementation

In this section we will explain our implementation of the Literary Lounge Library database on MySQL.

Step 1: we created a database called: Literary lounge_Project:

Using:

```
CREATE DATABASE Literary_lounge_Project;

USE Literary_lounge_Project;
```

Step 2: We created all the tables we need:

1- `Book` table

Book_ID	Author	Price	Title	Availability	ref_ID_FK
22	John Ronald Tolkien	19.99	the Lord of the Rings	Out of Stock	202
24	Antoine de Saint-Exupéry	39.99	The little prince	In Stock	44
31	Brothers Grimm	29.99	Grimms fairy tales	In Stock	21
36	Rowling	14.99	Harry Potter and the Philosopher Stone	In Stock	3
58	Agatha Christie	24.99	And then there Were None	Out of Stock	523
76	Cao Xueqin	19.99	Dream of the red room	In Stock	202
NULL	NULL	NULL	NULL	NULL	NULL

2- `Member` table :

M_ID	M_start	M_expire	M_address	name	phoneNo	Staff_ID_FK
13	2023-01-01	2023-12-31	London,Uk	John Doe	12346890	1
22	2023-02-01	2023-11-30	Paris,Fr	Alice Smith	23458901	2
37	2023-03-01	2023-10-31	London,Uk	Bob Johnson	34569012	3
42	2023-04-01	2023-09-30	Dubai,Uae	Eva Brown	45690123	4
52	2023-05-01	2023-08-31	Dubai,Uae	Michael Davis	56701234	5
NULL	NULL	NULL	NULL	NULL	NULL	NULL

3-`Payment` table :

ID	P_method	Amount
1001	Credit Card	100
2002	Cash	50
3003	PayPal	75
4004	Credit Card	60
5005	Cash	40
NULL	NULL	NULL

4-`Paid` table :

Paid_ID	Paid_date	ID_FK	M_ID_FK
1	2023-10-01 10:00:00	1001	13
2	2023-10-02 11:30:00	2002	22
3	2023-10-03 12:45:00	3003	37
4	2023-10-04 09:15:00	4004	42
5	2023-10-05 14:20:00	5005	52
NULL	NULL	NULL	NULL

5-`publisher` table:

P_ID	P_address	Name	phone_No
322	London, UK	Sophia Wilson	1535736
534	Jeddah, KSA	Khakid Ahmed	966546477
634	Paris, FR	Bob Black	34552765
641	Jeddah, KSA	Ahmed Ali	966537573
783	Paris, FR	Michael Davis	3264723
865	London, UK	Alice Smith	6286391
NULL	NULL	NULL	NULL

6- `PublishedBy` table :

Pub_ID	P_date	P_ID_FK	Book_ID_FK
5604	2023-05-01 12:20:00	634	58
10601	2023-01-01 08:00:00	641	31
26002	2023-02-01 09:30:00	322	22
30704	2023-03-01 10:45:00	783	24
47004	2023-04-01 07:15:00	865	36
NULL	NULL	NULL	NULL

7-`Return_Issue` table:

Rer_ID	returnDate	issueDate	dueDate	Book_ID_FK	M_ID_FK
1	2023-10-15 14:30:00	2023-09-15 10:00:00	2023-10-15 10:00:00	31	13
2	2023-10-10 16:45:00	2023-09-10 09:30:00	2023-10-10 09:30:00	22	22
3	2023-10-12 18:15:00	2023-09-12 11:20:00	2023-10-12 11:20:00	36	37
4	2023-10-09 13:55:00	2023-09-09 08:45:00	2023-10-09 08:45:00	24	42
5	2023-10-13 12:30:00	2023-09-13 07:15:00	2023-10-13 07:15:00	58	52
NULL	NULL	NULL	NULL	NULL	NULL

8- `Staff` table:

Staff_ID	destination	salary	phoneNo	name
1	Manager	55000.00	12346890	John Smith
2	Cashier	35000.50	98763210	Alice Johnson
3	Librarian	45000.75	55555555	David Lee
4	Assistant	30000.25	66666666	Emily Davis
5	Clerk	32000.00	77777777	Michael Brown
NULL	NULL	NULL	NULL	NULL

9-`reference_staff` table :

ref_ID	Staff_ID_FK
21	1
202	2
3	3
44	4
523	5
NULL	NULL

10-`Library_assistance` table:

Lib_ID	Staff_ID_FK
1647	3
2753	4
NULL	NULL

11-`department_staff` table:

dep_ID	Staff_ID_FK
953	1
243	3
342	5
NULL	NULL

12-`cashiers` table :

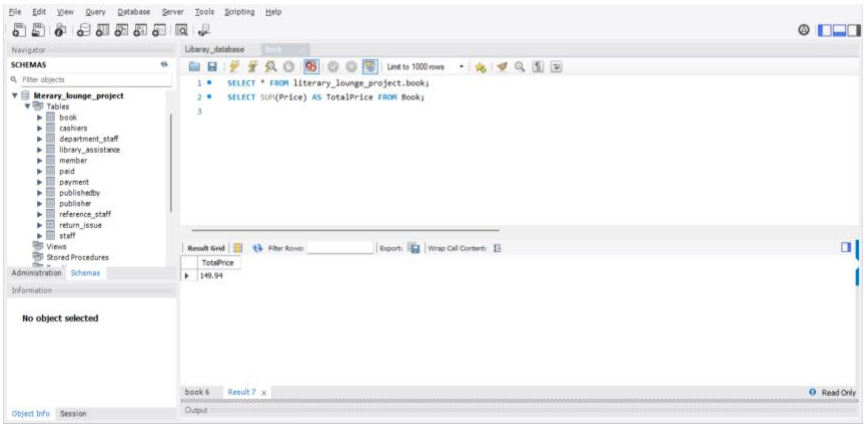
cash_ID	Staff_ID_FK
651	2
632	5
NULL	NULL

Step 3: We implemented the query on the tables:

a. Aggregation function.

Calculate the total price of all books

SELECT SUM(Price) AS TotalPrice FROM Book;

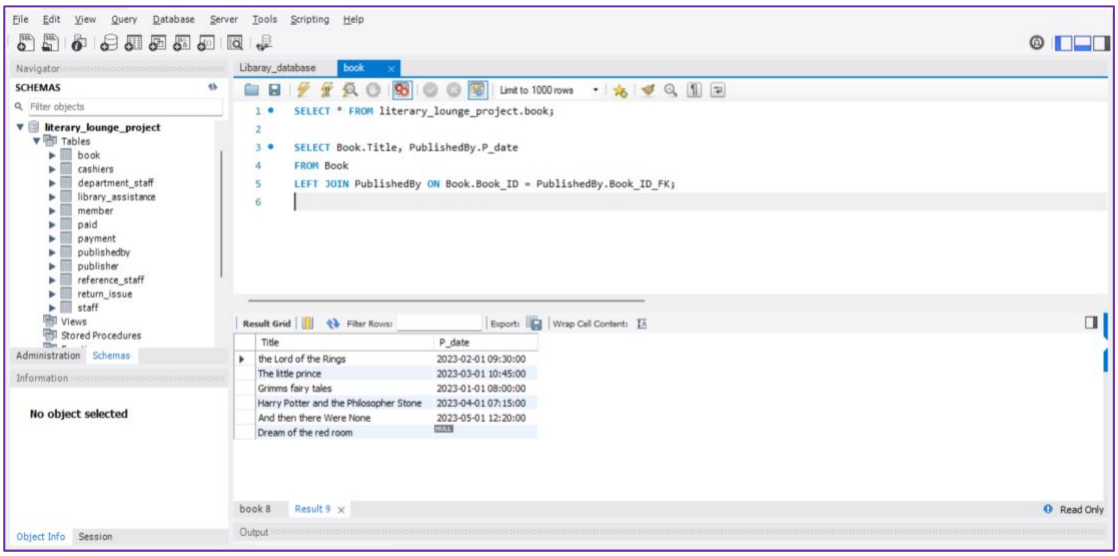


b. Outer join, Cross Join, Natural Join, Left Join, Right Join, Full

Join:

Left Join between `Book` and `PublishedBy`

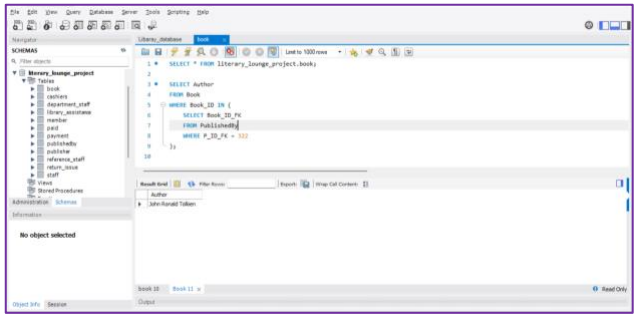
```
SELECT Book.Title, PublishedBy.P_date
FROM Book
LEFT JOIN PublishedBy ON Book.Book_ID = PublishedBy.Book_ID_FK;
```



c. Nested query.

Find the authors of books published by 'Sophia Wilson'

```
SELECT Author
FROM Book
WHERE Book_ID IN (
   SELECT Book_ID_FK
   FROM PublishedBy
   WHERE P_ID_FK = 322
);
```

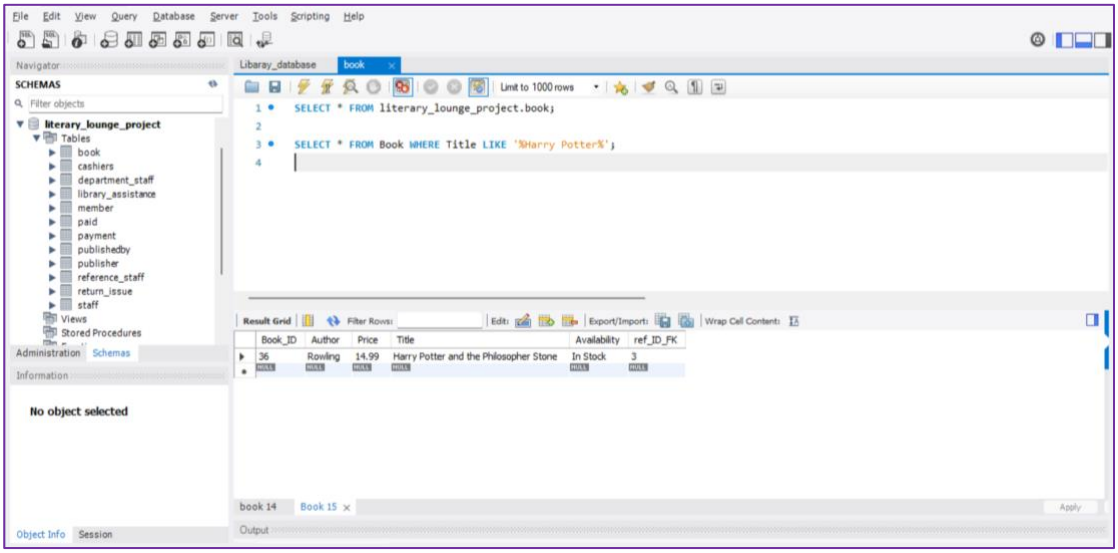


d. Use CHECK, LIKE, IN, and logical operator (<,>,>=,…..) (at least two)

1-using LIKE:

Find books with 'Harry Potter' in the title

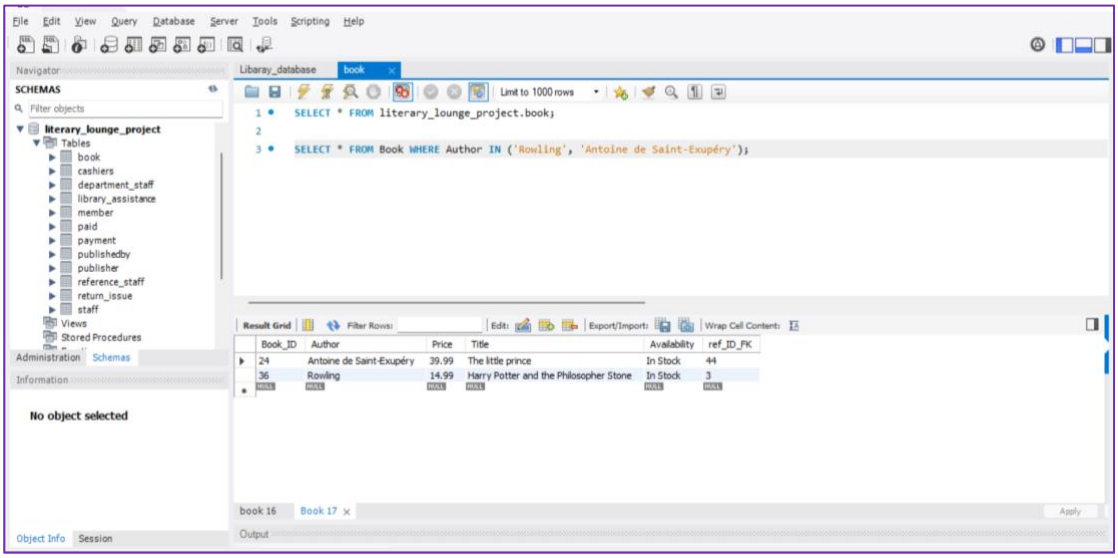
SELECT * FROM Book WHERE Title LIKE '%Harry Potter%';



2-using IN:

Find books by certain authors

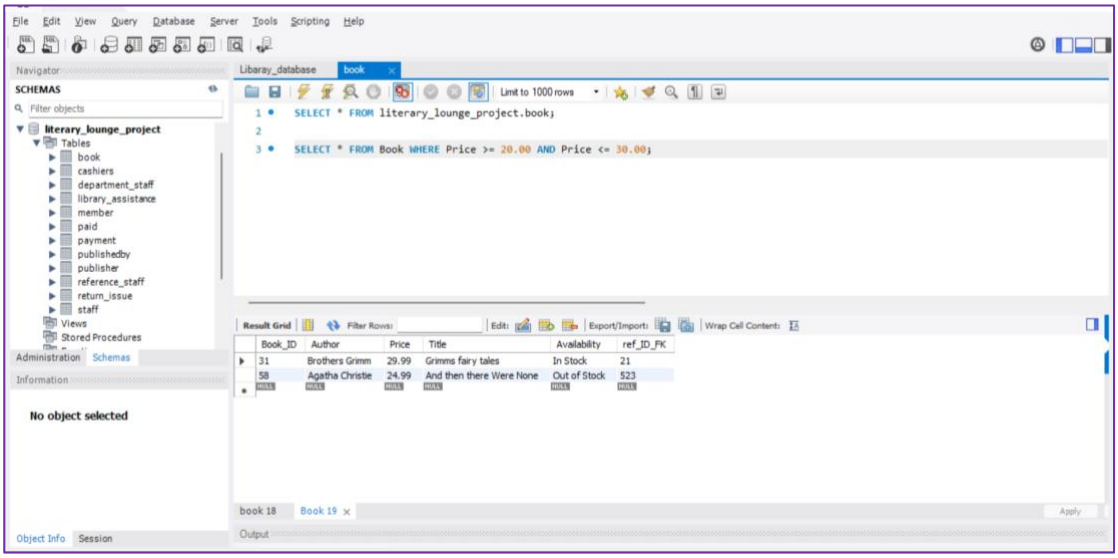
SELECT * FROM Book WHERE Author IN ('Rowling', 'Antoine de Saint-Exupéry');



3- using logical operators:

find books with a price between $20 and $30

SELECT * FROM Book WHERE Price >= 20.00 AND Price <= 30.00;



e. Link 3 or more tables.

Find books published by publishers in London

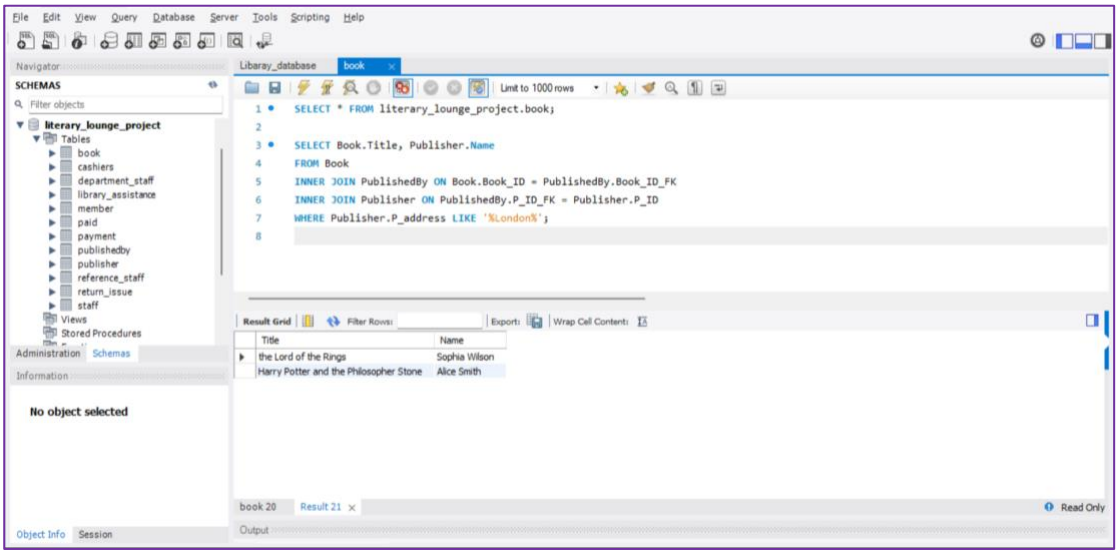
SELECT Book.Title, Publisher.Name

FROM Book

INNER JOIN PublishedBy ON Book.Book_ID = PublishedBy.Book_ID_FK

INNER JOIN Publisher ON PublishedBy.P_ID_FK = Publisher.P_ID

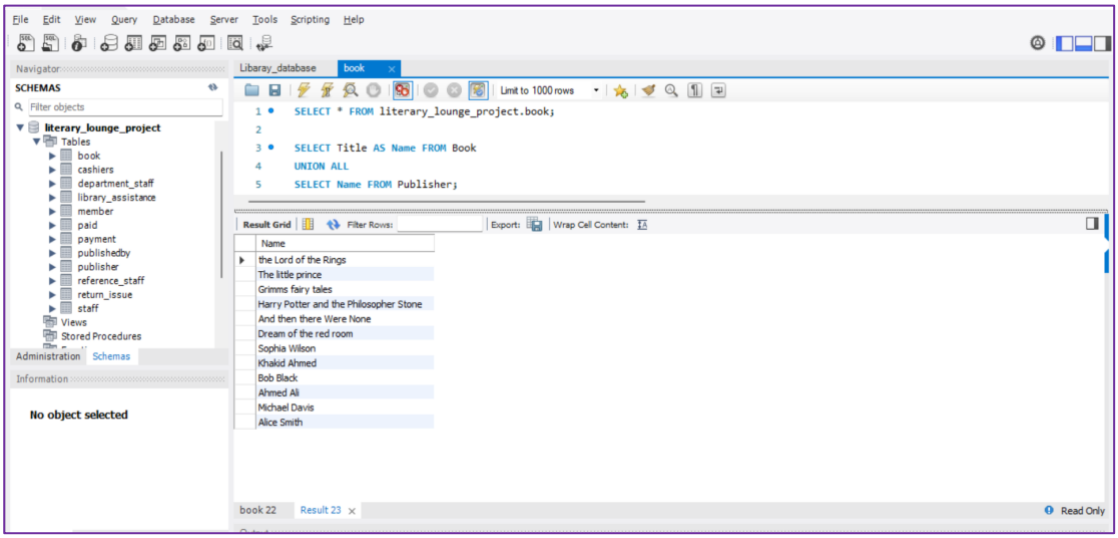
WHERE Publisher.P_address LIKE '%London%';



f. Use UNION ALL, EXCEPT ALL, INTERSECT, MINUS.

1- UNION ALL:

Combine the titles of books and the names of publishers



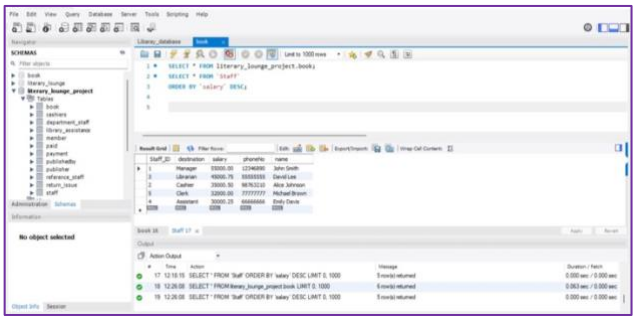
g. Use GROUP BY, ORDER BY, HAVING (at least two).

1- Using Group by :

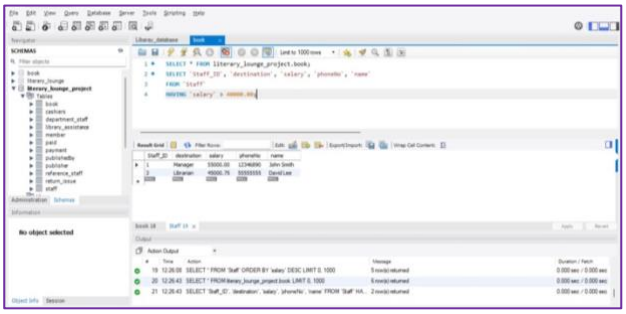
Group number of books by available

```
SELECT * FROM literary_lounge_project.book;
SELECT count(Book_ID) as "number of books",Availability
```

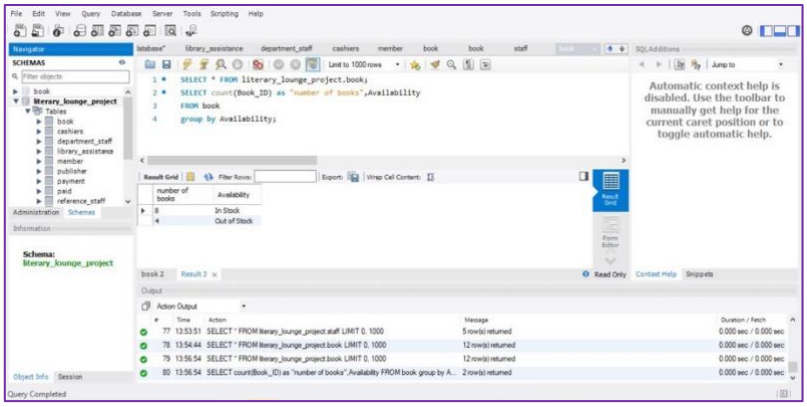
FROM book
group by Availability;

2- 

in Using order by :
Arrange staff member salaries descending order
SELECT * FROM Staff
ORDER BY salary DESC;



3- using HAVING :
select staff members who have a salary greater than a specified amount
SELECT Staff_ID, destination, salary, phoneNo, name
FROM Staff
HAVING salary > 40000.00;

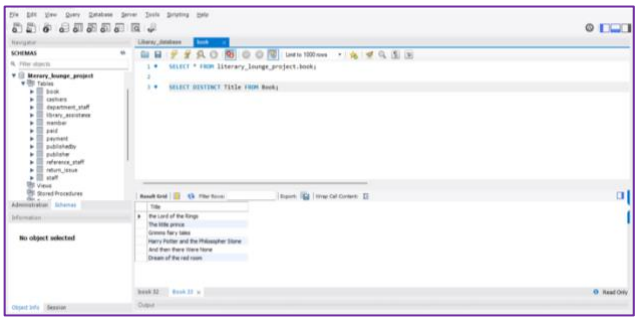


h. Use DISTINCT, ALL, AS.

1- Using DISTINCT:

List all distinct book titles

SELECT DISTINCT Title FROM Book;

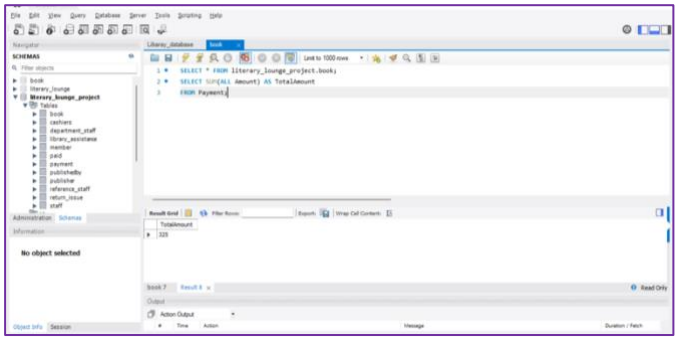
with 

2-using the ALL keyword SUM:

Calculate the total amount paid by all members

SELECT SUM(ALL Amount) AS TotalAmount

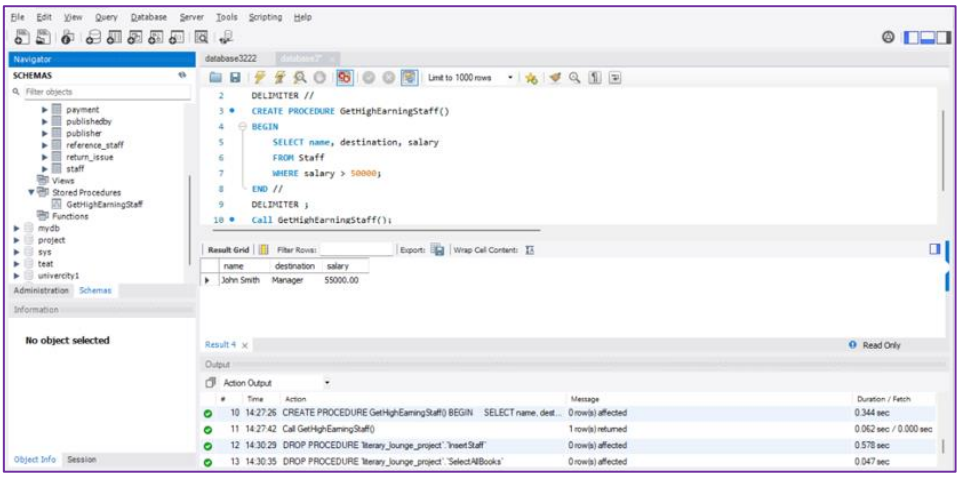
FROM Payment;



i. MySQL/ACCESS STORED PROCEDURE.

This stored procedure is named "GetHighEarningStaff" and it retrieves staff members with a salary greater than 50,000.

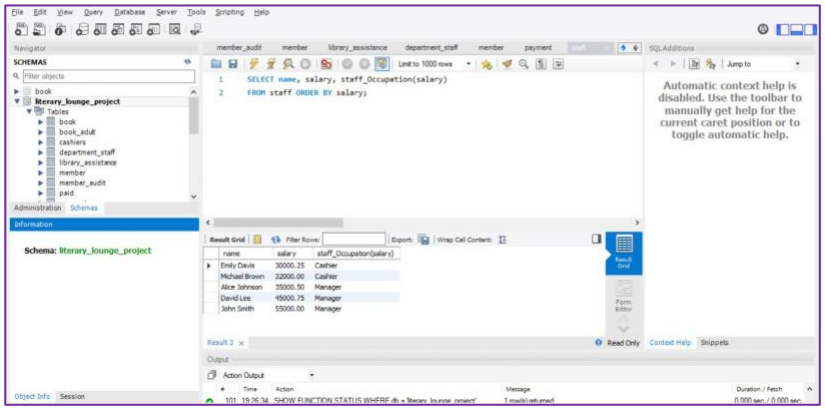
```
CREATE PROCEDURE GetHighEarningStaff()
BEGIN
   SELECT name, destination, salary
   FROM Staff
   WHERE salary > 50000;
END //
DELIMITER ;
```



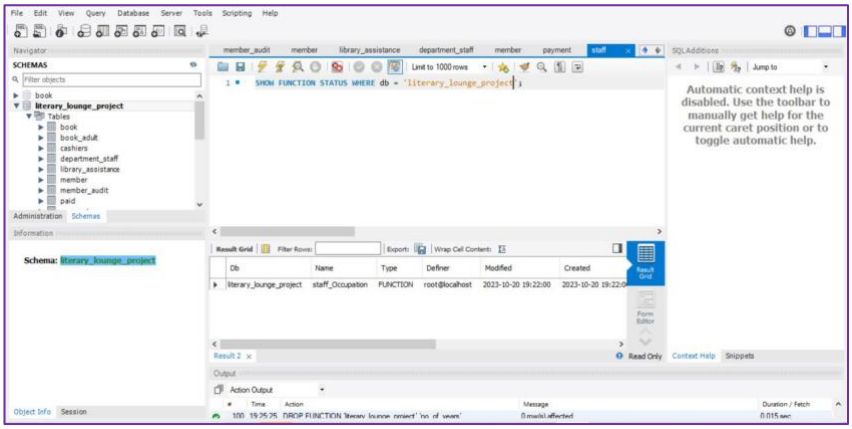
j. MySQL/ACCESS STORED FUNCTIONS.

we will create a function that returns the staff occupation based on the salary

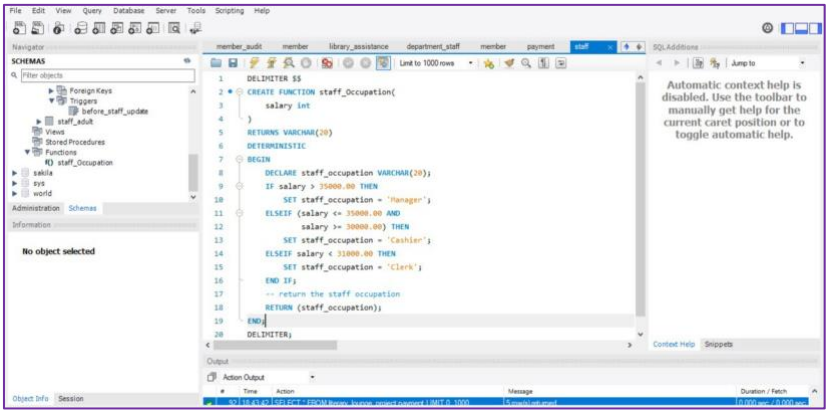
Step1:



Step2:



Step3:

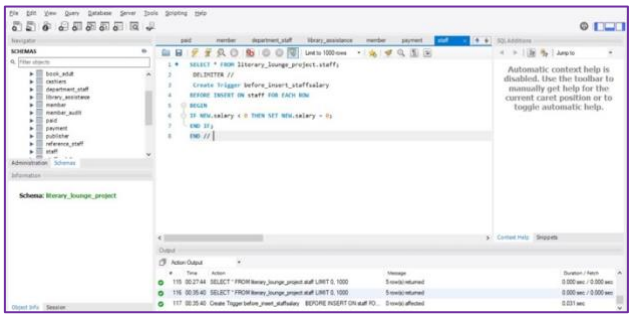


k. Triggers.

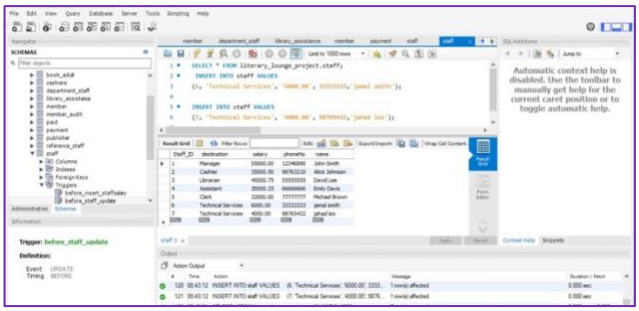
```
SELECT * FROM literary_lounge_project.staff;
 DELIMITER //
 Create Trigger before_insert_staffsalary
BEFORE INSERT ON staff FOR EACH ROW
BEGIN
IF NEW.salary < 0 THEN SET NEW.salary = 0;
END IF;
END //
 INSERT INTO staff VALUES
(6, 'Technical Services', '6000.00', 33333333,'jamal smith');

INSERT INTO staff VALUES
(7, 'Technical Services', '4000.00', 98765432,'jahad leo');
```

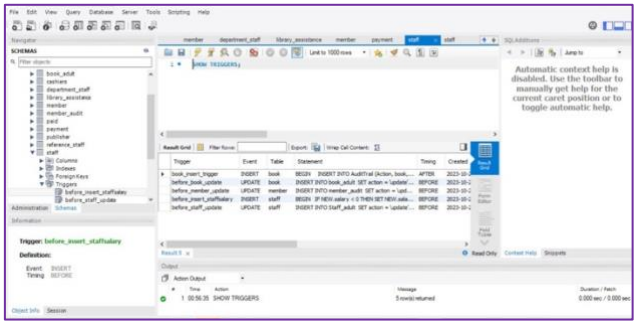
Step1:



Step2:



step3:



Code:

```
CREATE DATABASE
Literary_lounge_Project;
```

```
USE Literary_lounge_Project;

CREATE TABLE `Book` (
`Book_ID` int NOT NULL,
`Author` varchar(255) NOT NULL,
`Price` decimal(10, 2) NOT NULL,
`Title` varchar(255) NOT NULL,
`Availability` varchar(255) NOT NULL,
`ref_ID_FK` int NOT NULL,
PRIMARY KEY (`Book_ID`),
FOREIGN KEY (`ref_ID_FK`) REFERENCES `reference_staff` (`ref_ID`)
);
INSERT INTO Book (Book_ID, Author, Price, Title, Availability, ref_ID_FK)
VALUES
(31, 'Brothers Grimm', 29.99, 'Grimms fairy tales', 'In Stock',021),
(22, 'John Ronald Tolkien', 19.99, 'the Lord of the Rings', 'Out of Stock',202),
(36, 'Rowling', 14.99, 'Harry Potter and the Philosopher Stone', 'In Stock',003),
(24, 'Antoine de Saint-Exupéry', 39.99, 'The little prince', 'In Stock',044),
(58, 'Agatha Christie', 24.99, 'And then there Were None', 'Out of Stock',523),
(76, 'Cao Xueqin', 19.99, 'Dream of the red room', 'In Stock',202);

CREATE TABLE `cashiers` (
`cash_ID` int NOT NULL,
`Staff_ID_FK` int NOT NULL,
PRIMARY KEY (`cash_ID`),
FOREIGN KEY (`Staff_ID_FK`) REFERENCES `Staff` (`Staff_ID`)
);
INSERT INTO `cashiers` (`cash_ID`, `Staff_ID_FK`)
VALUES
(651, 2),
(632, 5);
```

```
CREATE TABLE `department_staff` (
`dep_ID` int NOT NULL,
`Staff_ID_FK` int NOT NULL,
PRIMARY KEY (`dep_ID`),
FOREIGN KEY (`Staff_ID_FK`) REFERENCES `Staff` (`Staff_ID`)
);
INSERT INTO `department_staff` (`dep_ID`, `Staff_ID_FK`)
VALUES
(953, 1),
(243, 3),
(342, 5);

CREATE TABLE `Library_assistance` (
`Lib_ID` int NOT NULL,
`Staff_ID_FK` int NOT NULL,
PRIMARY KEY (`Lib_ID`),
FOREIGN KEY (`Staff_ID_FK`) REFERENCES `Staff` (`Staff_ID`)
);
INSERT INTO `Library_assistance` (`Lib_ID`, `Staff_ID_FK`)
VALUES
(1647, 3),
(2753, 4);
CREATE TABLE `Member` (
`M_ID` int NOT NULL,
`M_start` varchar(255) NOT NULL,
`M_expire` varchar(255) NOT NULL,
`M_address` varchar(255) NOT NULL,
`name` varchar(255) NOT NULL,
`phoneNo` int NOT NULL,
```

```
`Staff_ID_FK` int NULL,

PRIMARY KEY (`M_ID`),

FOREIGN KEY (`Staff_ID_FK`) REFERENCES `Staff` (`Staff_ID`)

);

INSERT INTO `Member` (`M_ID`,`name`, `M_start`, `M_expire`, `M_address`,  `phoneNo`,
`Staff_ID_FK`)

VALUES

(13,'John Doe', '2023-01-01', '2023-12-31', 'London,Uk',  12346890, 1),

(22, 'Alice Smith','2023-02-01', '2023-11-30', 'Paris,Fr',  23458901, 2),

(37,  'Bob Johnson','2023-03-01', '2023-10-31', 'London,Uk', 34569012, 3),

(42, 'Eva Brown', '2023-04-01', '2023-09-30', 'Dubai,Uae', 45690123, 4),

(52,  'Michael Davis','2023-05-01', '2023-08-31', 'Dubai,Uae', 56701234, 5);

CREATE TABLE `Paid` (

`Paid_ID` int NOT NULL,

`Paid_date` datetime NOT NULL,

`ID_FK` int NOT NULL,

`M_ID_FK` int NOT NULL,

PRIMARY KEY (`Paid_ID`),

FOREIGN KEY (`ID_FK`) REFERENCES `Payment` (`ID`),

FOREIGN KEY (`M_ID_FK`) REFERENCES `Member` (`M_ID`)

);

INSERT INTO `Paid` (`Paid_ID`, `Paid_date`, `ID_FK`, `M_ID_FK`)

VALUES

(1, '2023-10-01 10:00:00', 1001, 13),

(2, '2023-10-02 11:30:00', 2002, 22),

(3, '2023-10-03 12:45:00', 3003, 37),

(4, '2023-10-04 09:15:00', 4004, 42),

(5, '2023-10-05 14:20:00', 5005, 52);


CREATE TABLE `Payment` (

`ID` int NOT NULL,
```

```
`P_method` varchar(255) NOT NULL,
`Amount` float NOT NULL,
PRIMARY KEY (`ID`)
);
INSERT INTO `Payment` (`ID`, `P_method`, `Amount`)
VALUES
(1001, 'Credit Card', 100.00),
(2002, 'Cash', 50.00),
(3003, 'PayPal', 75.00),
(4004, 'Credit Card', 60.00),
(5005, 'Cash', 40.00);
CREATE TABLE `publisher` (
`P_ID` int NOT NULL,
`P_address` varchar(255) NOT NULL,
`Name` varchar(255) NOT NULL,
`phone_No` int NULL,
PRIMARY KEY (`P_ID`)
);
INSERT INTO `publisher` (`P_ID`, `Name`, `P_address`, `phone_No`)
VALUES
(641,'Ahmed Ali' ,'Jeddah, KSA',966537573),
(322, 'Sophia Wilson','London, UK',1535736),
(783,'Michael Davis' ,'Paris, FR',3264723),
(534,'Khakid Ahmed' ,'Jeddah, KSA',966546477),
(865,'Alice Smith' ,'London, UK',6286391),
(634, 'Bob Black','Paris, FR',34552765);
CREATE TABLE `PublishedBy` (
`Pub_ID` int NOT NULL,
`P_date` datetime NOT NULL,
`P_ID_FK` int NOT NULL,
`Book_ID_FK` int NOT NULL,
```

```
PRIMARY KEY (`Pub_ID`),
FOREIGN KEY (`P_ID_FK`) REFERENCES `publisher` (`P_ID`),
FOREIGN KEY (`Book_ID_FK`) REFERENCES `Book` (`Book_ID`)
);
INSERT INTO `PublishedBy` (`Pub_ID`, `P_date`, `P_ID_FK`, `Book_ID_FK`)
VALUES
(10601, '2023-01-01 08:00:00', 641, 31),
(26002, '2023-02-01 09:30:00', 322, 22),
(30704, '2023-03-01 10:45:00', 783, 24),
(47004, '2023-04-01 07:15:00', 865, 36),
(5604, '2023-05-01 12:20:00', 634, 58);

CREATE TABLE `reference_staff` (
`ref_ID` int NOT NULL,
`Staff_ID_FK` int NOT NULL,
PRIMARY KEY (`ref_ID`),
FOREIGN KEY (`Staff_ID_FK`) REFERENCES `Staff` (`Staff_ID`)
);
INSERT INTO `reference_staff` (`ref_ID`, `Staff_ID_FK`)
VALUES
(021, 1),
(202, 2),
(003, 3),
(044, 4),
(523, 5);


CREATE TABLE `Return_Issue` (
`Rer_ID` int NOT NULL,
`returnDate` datetime NOT NULL,
`issueDate` datetime NOT NULL,
```

```
`dueDate` datetime NOT NULL,
`Book_ID_FK` int NOT NULL,
`M_ID_FK` int NOT NULL,
PRIMARY KEY (`Rer_ID`),
FOREIGN KEY (`Book_ID_FK`) REFERENCES `Book` (`Book_ID`),
FOREIGN KEY (`M_ID_FK`) REFERENCES `Member` (`M_ID`)
);
INSERT INTO `Return_Issue` (`Rer_ID`, `returnDate`, `issueDate`, `dueDate`, `Book_ID_FK`, `M_ID_FK`)
VALUES
(1, '2023-10-15 14:30:00', '2023-09-15 10:00:00', '2023-10-15 10:00:00', 31, 13),
(2, '2023-10-10 16:45:00', '2023-09-10 09:30:00', '2023-10-10 09:30:00', 22, 22),
(3, '2023-10-12 18:15:00', '2023-09-12 11:20:00', '2023-10-12 11:20:00', 36, 37),
(4, '2023-10-09 13:55:00', '2023-09-09 08:45:00', '2023-10-09 08:45:00', 24, 42),
(5, '2023-10-13 12:30:00', '2023-09-13 07:15:00', '2023-10-13 07:15:00', 58, 52);

CREATE TABLE `Staff` (
`Staff_ID` int NOT NULL,
`destination` varchar(255) NOT NULL,
`salary` decimal(10, 2) NOT NULL,
`phoneNo` int NOT NULL,
`name` varchar(255) NOT NULL,
PRIMARY KEY (`Staff_ID`)
);

INSERT INTO `Staff` (`Staff_ID`, `destination`, `salary`, `phoneNo`, `name`)
VALUES
(1, 'Manager', 55000.00, 12346890, 'John Smith'),
(2, 'Cashier', 35000.50, 98763210, 'Alice Johnson'),
(3, 'Librarian', 45000.75, 55555555, 'David Lee'),
(4, 'Assistant', 30000.25, 66666666, 'Emily Davis'),
```

(5, 'Clerk', 32000.00, 77777777, 'Michael Brown');

Conclusion

In conclusion, this database project has been a resounding success, thanks in large part to the exceptional teamwork demonstrated by all team members. Our collective efforts, collaboration, and dedication have resulted in a robust and efficient database system that meets the project's objectives and exceeds expectations. The seamless integration of our individual skills, effective communication, and the spirit of cooperation were the cornerstones of our achievement. This project not only showcases our technical expertise but also our ability to work together cohesively to overcome challenges and deliver a high-quality solution. The success of this project stands as a testament to the power of teamwork in achieving complex and ambitious goals. We look forward to applying the lessons learned from this experience to future projects and continuing to excel as a collaborative and innovative team.