Shape

Description automatically generated with medium confidence



**Essay / Assignment Title:** **Database System Design for a Library**

**Programme title: Enterprise Data Warehouses and Database Management Systems**

**Name: JASMEET SINGH**

**Year: 2024**

# CONTENTS

[CONTENTS 2](#_Toc170552147)

[1. INTRODUCTION 4](#_Toc170552148)

[2. (ERD DIAGRAM -ENTITY & RELATIONSHIP) 5](#_Toc170552149)

[3 . Database Implementation 9](#_Toc170552150)

[4. SQL QUERIES AND STORED PROCEDURES 13](#_Toc170552151)

[5. CAP THEOREM 20](#_Toc170552152)

[BIBLIOGRAPHY 22](#_Toc170552153)

[APPENDIX (if necessary) 23](#_Toc170552154)

**Statement of compliance with academic ethics and the avoidance of plagiarism**

I honestly declare that this dissertation is entirely my own work and none of its part has been copied from printed or electronic sources, translated from foreign sources and reproduced from essays of other researchers or students. Wherever I have been based on ideas or other people texts I clearly declare it through the good use of references following academic ethics.

(In the case that is proved that part of the essay does not constitute an original work, but a copy of an already published essay or from another source, the student will be expelled permanently from the postgraduate program).

Name and Surname (Capital letters):

JASMEET SINGH

.......................................................................................................................................

Date: ........................./........../.........

# INTRODUCTION

In this assignment I am going to design a database for the library. Library database is used in order to manage all the records of the library using Entity Relationship diagram. This database includes wide range of entities such as books, membership, library branch, borrower, staff, authors, publisher, issue details.

Let us discuss how this library database works. The library has a wide range of books so it needs to be maintained properly and in the efficient way so it will be easy for everyone to access books and manage library. It contains many books, magazines and newspaper which is written by various authors and can be published by many publishers. Each books have a unique ISBN no. to find the book from the book rags and reports are prepared of the issued books and return date of the books. It has a proper database of the borrower and their details who has issued/or returned the books which is maintained by the staff members. In order to access full library facility like free internet, unlimited books issue, access computers for information, library has different types of membership like monthly, quarterly, annually which are charged accordingly to the facilities provided in each. Moreover, this library has a various branches available in the city so the record is maintained, maybe the one book let us take a example of ‘MY FAULT’ is not available or issued by someone else in the particular branch but a borrower want to issue that book so by its ISBN no the staff member can find that which other branch has same book available In the nearest branch so person can borrow from there.

# 2. (ERD DIAGRAM -ENTITY & RELATIONSHIP)

ERD(Entity relationship diagram ) is a pictorial representation of the relationship between various entites(people, concepts and objects) and how they interact in the system or database. Here are the key components of ERD

1. **Entities**- people, conecpts or objects that are distinguishable and have data stored about them. For example I have given to make a database for library so it has entities like books, authors, publisher, issue details, borrower, library branch, membership, staff.
2. **Attributes**- details or properties of entites.for example books entity have attributes like book title, book edition etc.
3. **Relationships**- these are the association between different enties which are typically represented by lines. There are various relations such as one to one (1:1) , one to many (1:M), many to many (M:M)
4. **Primary keys**- these are the key assigned in entity to identify uniquely. For example in book entity ‘bookid’ is as primary key.
5. **Foreign keys**- these are the column used in relational database to link data between tables without altering the links between the tables.

Entities

* **Books** – it contains details of all the available books in the library and this entity contains attributes like **Id** (primary key), **title**, **Edition** (particular book is of which edition), **genre** (types of books that fall into which category like horror, religious, romantic, thriller)
* **Authors**- it contains details of the authors who wrote the books and contain attributes **id, name , address**
* **Publisher**- it contains details of publisher and attributes are **id, name, address, contact**
* **Issue\_details** -contains attributes like **ISBN\_no** (which is a book number), **issue\_date**, **return\_date**
* **Borrower** contains details of the book issued and has attributes **id, contact, name, email.**
* **Library\_branch**- since there are many branches for a library for my point of view it is necessary to add this entity which will help to find some books which are not available in one branch but accessible in other branch. This entity has attributes such as **id, location, contact**
* **Membership**- this entity contains which person had taken a membership for the library it can be monthly subscription, annual subscription and this thing will fall in **id ,** **membership\_type** attribute, other attribute like **payment\_method** ( how the subscription is paid either by card, by cash, online banking) and **date\_of\_payment**
* **Staff**- it has all the information regarding the staff of library. Attributes includes **id , first\_name,last\_name, position**

RELATIONSHIPS

**Books**-**Borrower** (1:N) one person can borrow many books .

- one book should be borrowed by one reader.

**Publisher**- **Books**- **(1:N)** one publisher can publish many books

-one book should be published by one publisher

**Author**-**Books** **(1:N)** one author can write many books

- one book should be written by one author

**Library branch**-**Staff** **(1:N)** one branch has many staff members

-one staff member should work at one branch

**Membership**-**Borrower** **(1:N)** type of membership can be used by many persons

-one membership should be used by one borrower only

**Staff** -**Issue details**- **(M:N)** one staff member can maintain many issues and return report

- these reports can be maintained by many staff members

- ne staff person can supervise many staff members

- one staff person can be supervised

**Borrower- Issue details**- **(1:N)**one borrower can issue/return many books

-one book should be returned/issued by one borrower.

SCHEMAS

In the entity relationship digram, schemas is a framework that represents how data is organized and how relationships between the enitites are enact within database. It act as a blueprint for managing and designing database, ensuring that data is logically structured and related to each other .

* **Books** – (ID, title, edition, genre, publisher\_id, borrow\_id)
* **Publisher** – (ID, name, address, contact)
* **Author** – (ID, name, address)
* **Author\_fk** – (write, written\_by)
* **Library\_branch** – (ID, location, contact)
* **Membership** – (ID, membership type, payment\_method, date\_of\_payment)
* **Staff** – (ID, first\_name,last\_name, position, supervise\_id, library\_id)
* **Maintain\_fk** –(maintain, maintained\_by)
* **Borrower** – (ID, name, contact, email, membership\_id)
* **Issuedetails** – (ISBN no, issue date, return date, borrow\_id)

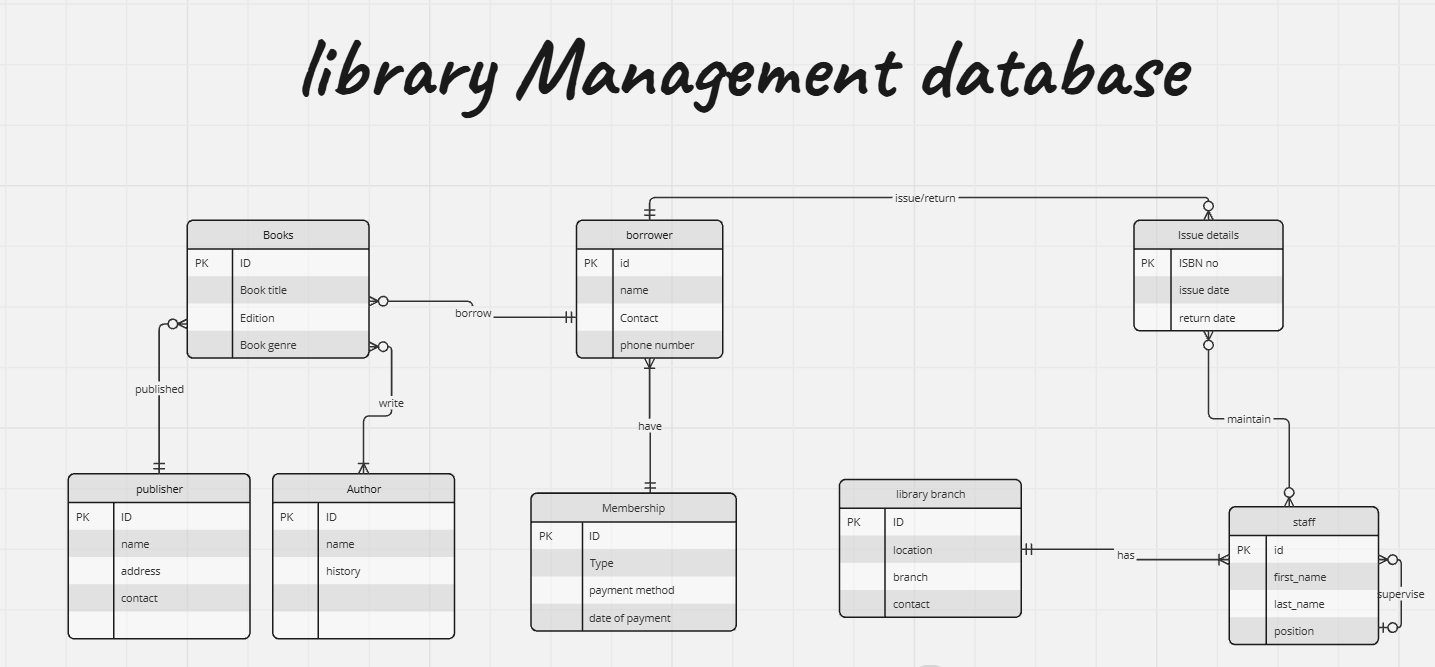


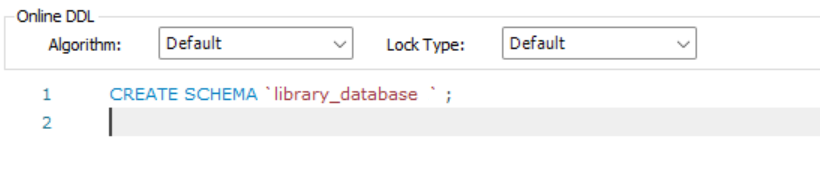
Fig 1: a manually created ERD in miro

# 3 . Database Implementation

Implementing a library database involves creating a structured system to manage various aspects of library operations such as tracking books, users, checkouts, returns, and overdue itemss. Below is a mentioned on how to implement a simple library database using MYSQL

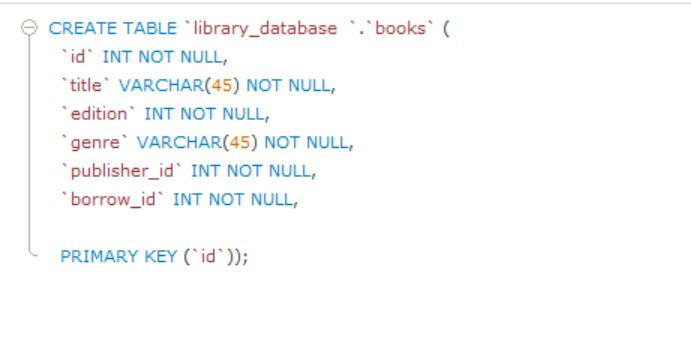
To ensure that my library database has efficient data management and retrieval in MySQL Workbench, so it is populated with data using structured MYSQL queries. Below is the detailed queries:-

1. To create schema : ‘library\_database’

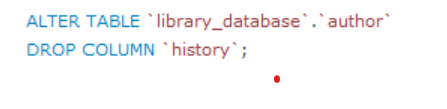


1. To create table- The table contains many entites and each entities has multiple attributes

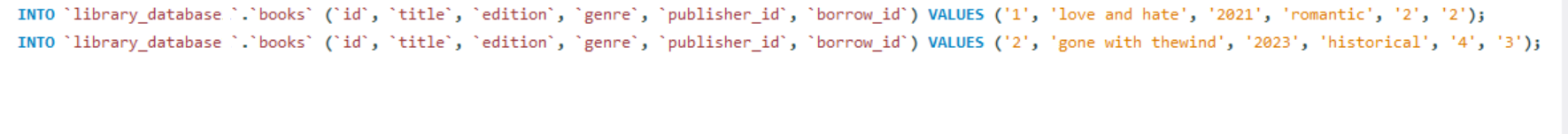
Let us take a example of entity Books and the attributes in this are id (primary key), title, edition, genre, publisher-id, borrow\_id.



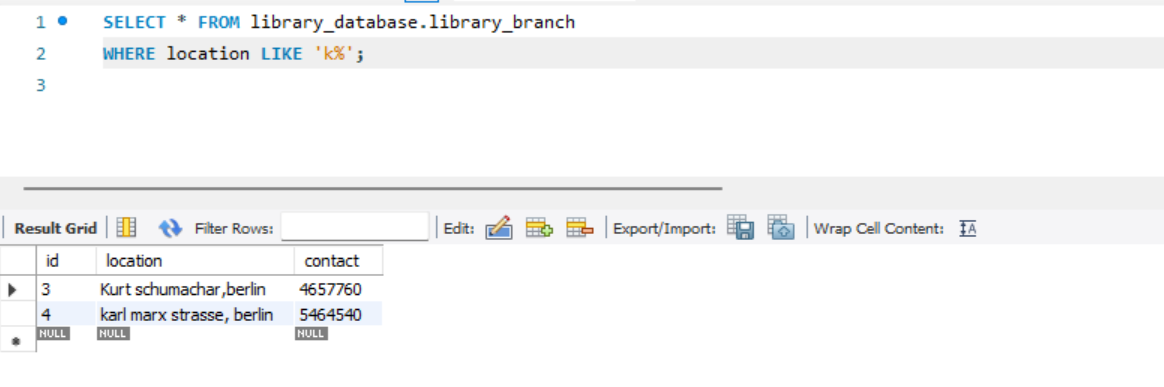
1. To delete a column from a table- I have mistakenly added a new column(history) in a author table so I run this command to delete the entire column



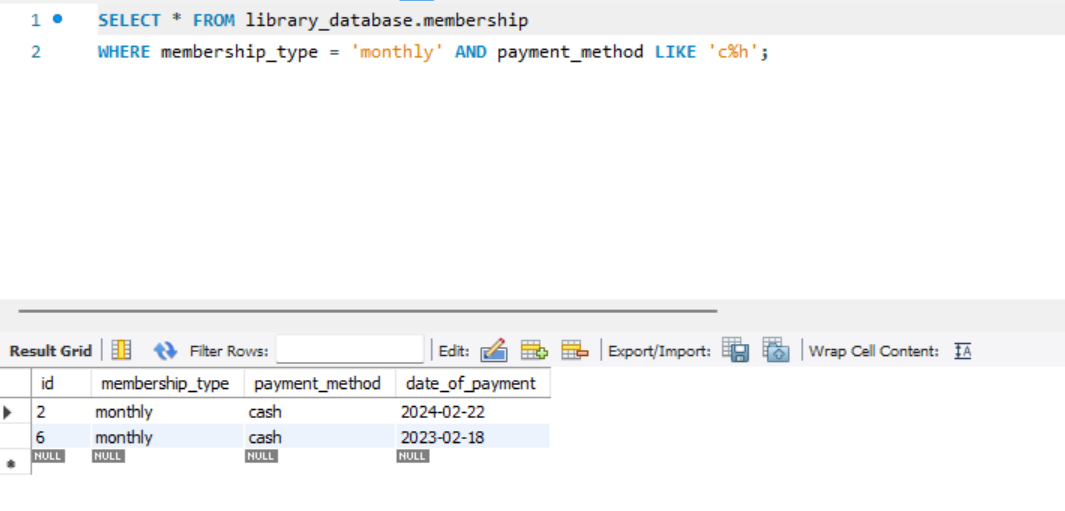
1. To insert data into the tables-



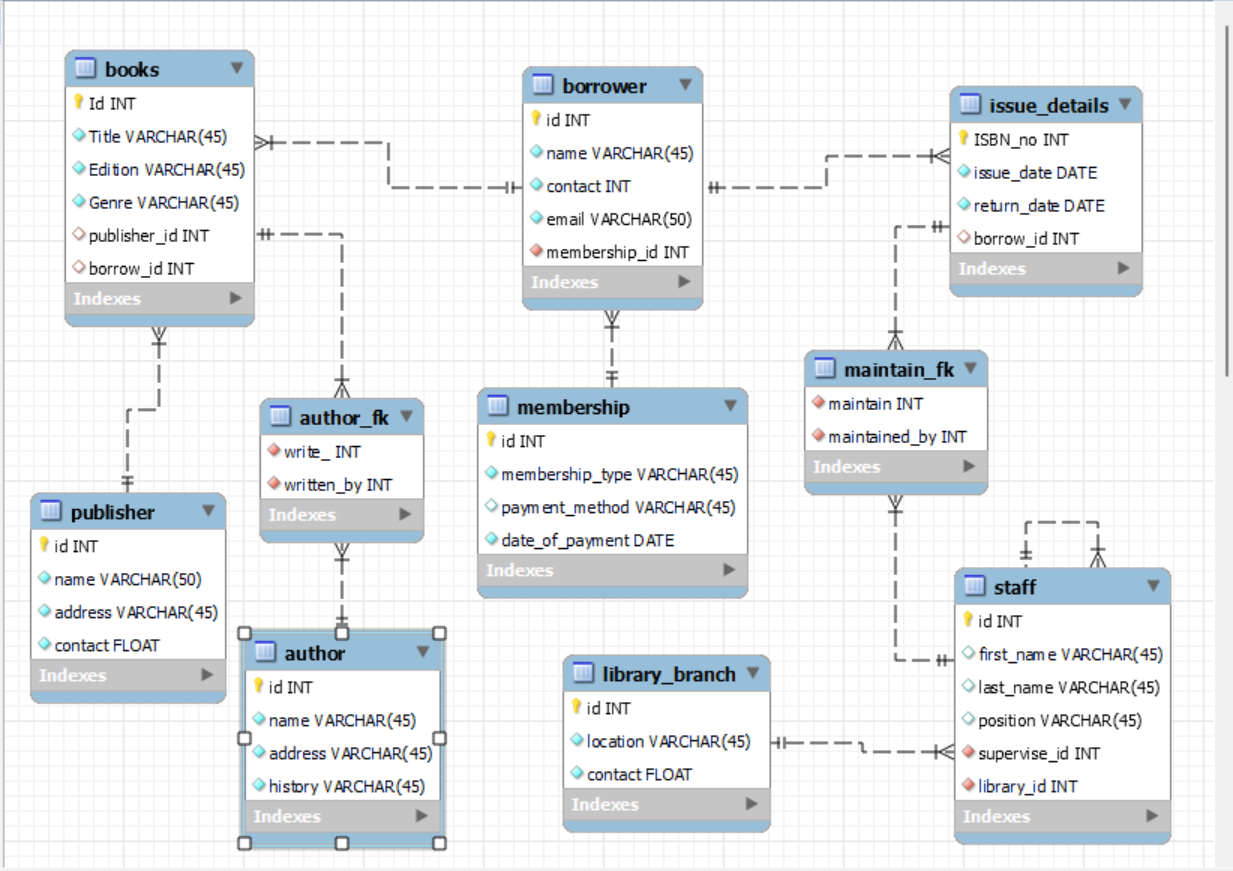
1. To use the wild card- A wild card is used to substitute one or more charcter in a string. They are used with LIKE operator. The LIKE charcter is used in where clause to search for specified pattern in column. For example in this example;- from library\_databse in table library\_branch it search location which start with charcter ‘K’.



1. To use AND operator- The **AND** operator is used to filter records based on more than one condition, like if you want to return all membership type from ‘monthly’ that starts with the letter 'C' and end with ‘D”



**ERD DIAGRAM AFTER DATA IMPLEMENTATION AND ADDING FOREIGN KEYS**



# 4. SQL QUERIES AND STORED PROCEDURES

Stored procedure- its is a SQl code that is used in the database. It use encapsulation function which means that it can be used over and over again, making it easier to manage and maintain the logic in one place.

KEY FEATURES;

**-Precomplied excuetion**- this code is complied once and stores in database so that the database engine can execute them more quickly.

-**Security**- It provides a layer of security by allowing user to excute without interfering and getting direct access to the underlying code

-**Performance**- Stored procedures can improve performance by reducing network traffic between the application and the database server. Instead of sending multiple SQL queries, an application can call a single procedure.

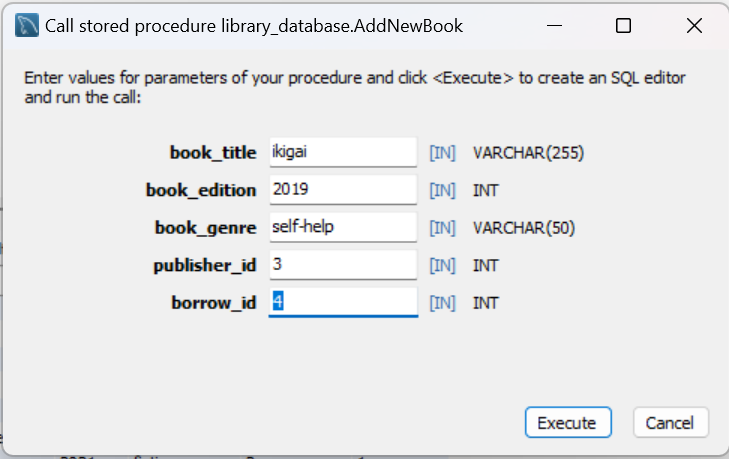
Stored procedures are powerful tools in SQL databases that encapsulate logic, optimize performance, and enhance security. They are essential for developing robust, scalable, and maintainable database applications.

SQL queries for Stored procedure

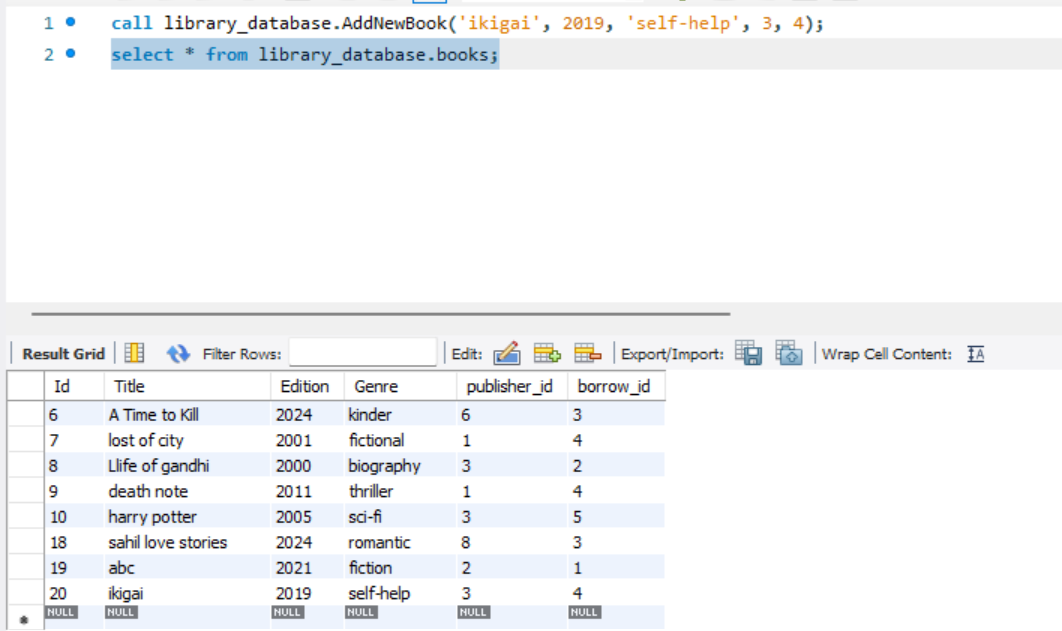
1. **Stored procedure to add new book in the library database-** In this stored procedure we will create a stored procedure named as ‘AddnewBook' and various input parameters are added in store procedure. After this a new row is inserted into books table.the columns are being inserted are ‘title’, ‘edition’, ‘genre’, publisher\_id’ and ‘borrow\_id’ as shown in in figure below



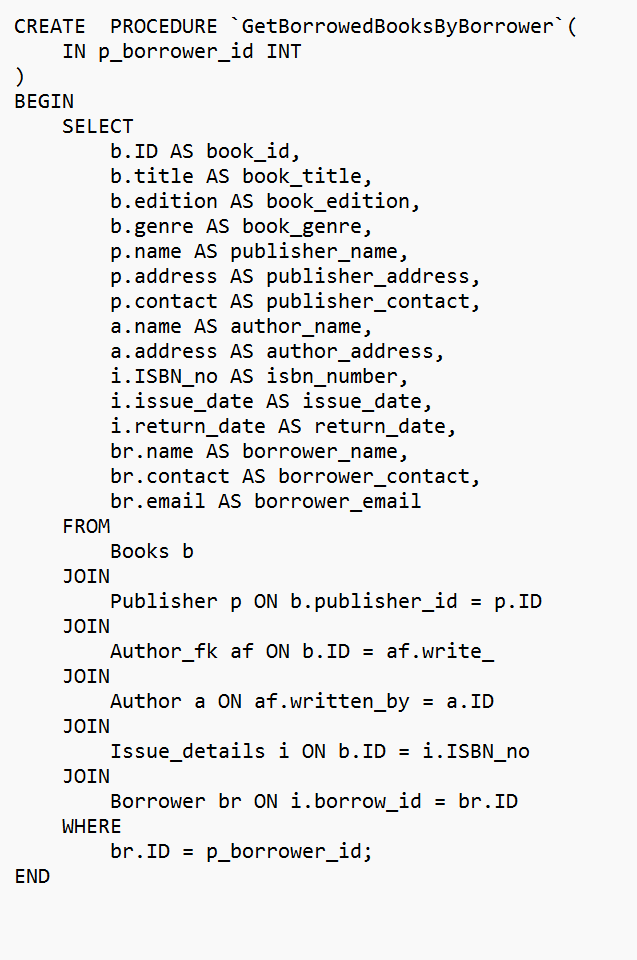
after calling the procedure we input values in these parameters to add a new book . By adding parameter as ( ‘ikigai’, 2019, ‘self-help’, 3, 4)



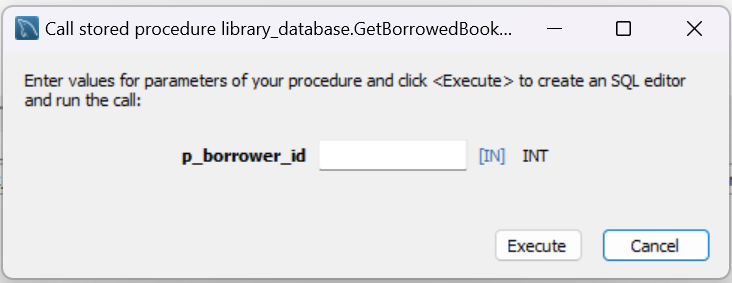
The output is generated a new book is added in a books table



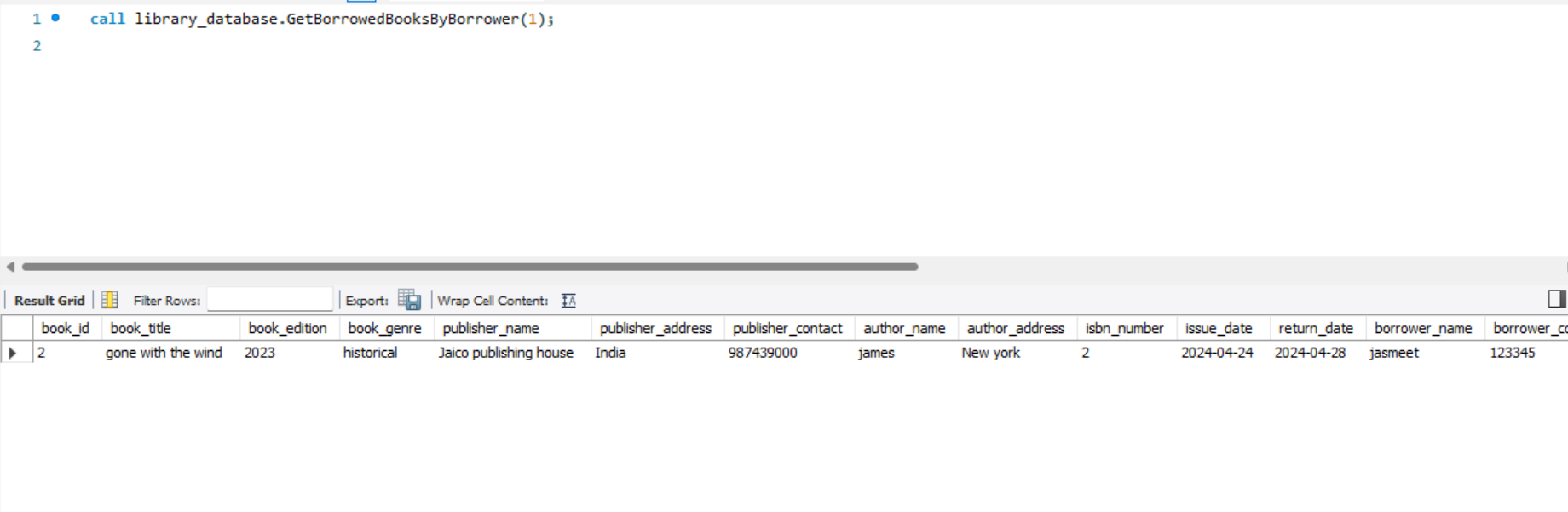
1. Stored procedure to ‘ get borrowed books by borrower’ – In this stored procedure there is one input parameter which is p\_borrower\_id which take input as a integer. The select statement retrives all the information based on borrower id and the columns are retrieved like book\_id as b.id, book\_title as b.title, book\_edition as b.edition, book\_genre as b.genre, publisher\_name as p.name etc. after that the publisher table is joined with books table, author table is joined based on written\_by which is a foreign key as shown in figure below



After executing this stored procedure the input parameter is added in P\_borrower\_id as a integer and retrives information about borrowed book by specific borrower. It joins several tables (‘books’, ‘publihser’, ‘author\_fk’, ‘author’, ‘issue-details’, ‘borrower’) to gather comprehensive information about books, their publisher, authors, issue details and borrower details

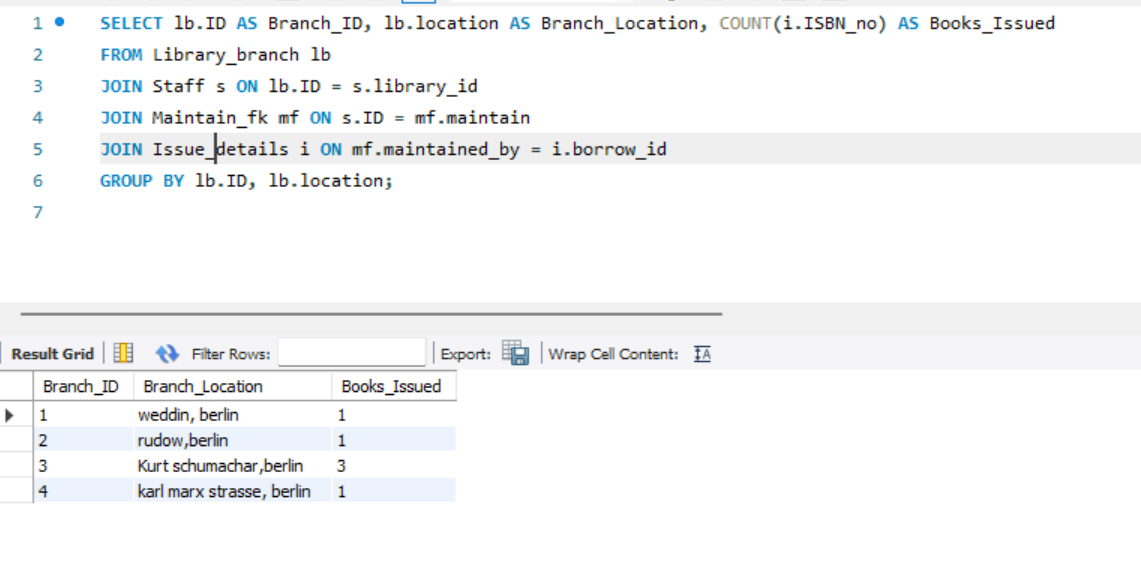


Now the result is shown as under for the borrower id ‘1’



SQL QUERIES

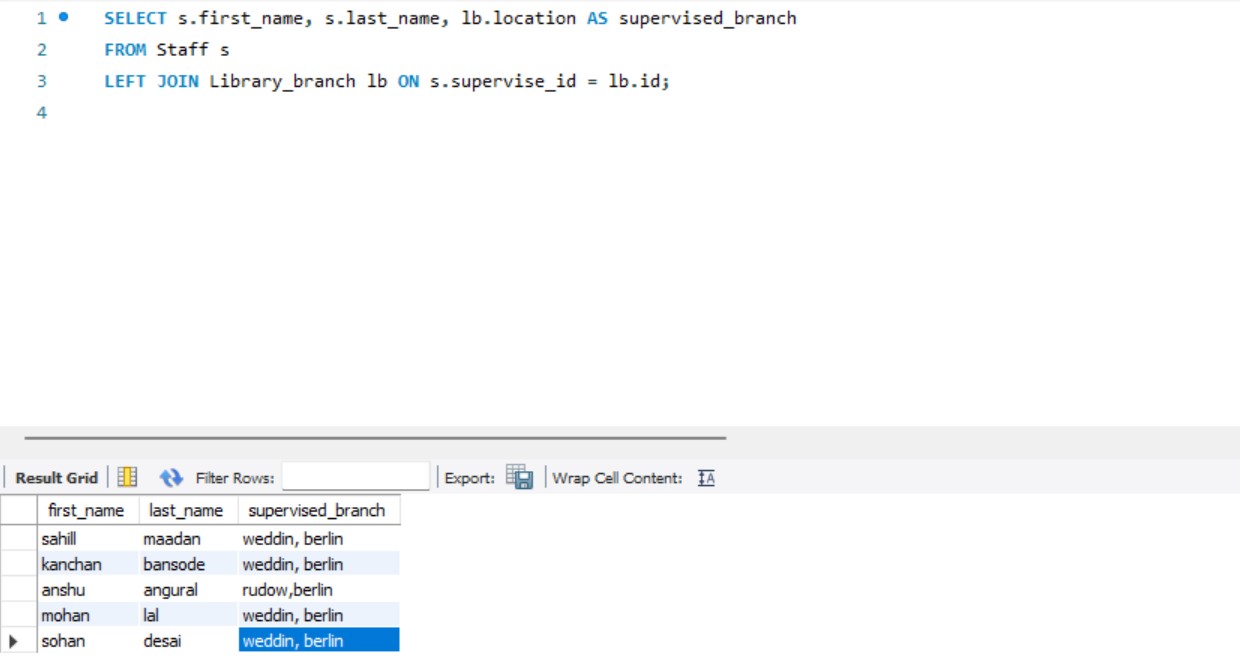
1. Count of Books Issued by Each Library Branch:



1. Find the most frequently borrowed book genre.



1. Query to find borrowers who have issued less than 3 books:



# 5. CAP THEOREM

The CAP theorem, or Brewer's theorem, posits that a distributed database system can only guarantee two of the following three properties simultaneously: Consistency, Availability, and Partition Tolerance. Nevertheless, the extension of CAP Theorem to a library database in ERP is explained as follows:

1. **Consistency**:

- Transactional Integrity: Makes sure that processes such as book borrowing, return, and others have the correct status in the system. This does away with problems such as over bookings, and lost sales transactions.

- Atomicity: Commercial exchanges are of an ‘either / or’ type. If, for instance, a user checks out a book, the operation should either go to completion or not at all given the need to avoid wrong data records.

2. **Availability**:

- High Availability: They should be able to cater for many users requesting for services at one time especially during rush time for example in the onset of a semester or whenever there is an event.

- Redundancy: It is aimed at explaining that effective availability can be reached through redundant server instances during the hardware failures.

3. **Partition Tolerance**:

- Network Issues: Whereas local library might have work stable networks, distributed branches may face problems with connectivity. Some of the considerations which are relevant for the design of the system include the ability to work under conditions where the connection is lost temporarily but data should not be lost.

- Eventual Consistency: In two cases where it is important to be partition tolerant, then it is okay to be eventually consistent. This remains bound by the fact that the system makes certain that data will agree in the long run once the partition is corrected.

**Practical Implementation**

Database sharding increases speed and reliability by distributing data across multiple hosts. Replication enhances availability and partition tolerance, while conflict resolution procedures resolve simultaneous operations.

# BIBLIOGRAPHY

# APPENDIX (if necessary)