P0

PROJECT ON

Simple Library Management System

Prepared By:

Thorat Rani

Guided By:

Zakir Hussain

TABLE OF CONTENTS

Sr.	Content	Page No
1.	Description about Simple Library Management	3
2.	Databases used in this project	4
3.	Tables used in each databases	5
4.	Queries that can be identified	14
5.	Final Goal of the project	16

Simple Library Management

A Simple Library Management System using MySQL is a database-driven application designed to help manage a library's basic operations such as tracking books, borrowers, and lending records. It facilitates organized storage and retrieval of information regarding books, members, and borrowing transactions. A Library Management System (LMS) in MySQL is a database solution for managing library resources and operations efficiently. It tracks books, members, transactions (borrowing and returning), and fines. The system includes tables such as Books(storing book details), Members (library user details), Transactions (borrow/return records), and Fines(overdue penalties). With basic SQL operations like INSERT, UPDATE, and DELETE, librarians can manage book availability, track user activity, and maintain organized records. The system improves library management by automating manual tasks, ensuring data accuracy, and providing easy access to important information.

Databases used in this Project

- Created five databases:

Syntax: create database [name of database];

DATABASES:

- Database 1: Book Management
- Database 2: Patron Management
- Database 3: Circulation Management
- Database 4: Acquisitions Management
- Database 5: Reporting and Analytics

Tables used in each of the Databases

1.Book Management Databases

Book Table

This table stores information about the books in the library or bookstore.

Columns

id: A unique identifier for each book (Primary Key).
title: The title of the book.
author: The author of the book (name).
publication_date: The date when the book was first published.
ISBN: The unique International Standard Book Number for identifying the book.

Authors Table

This table contains information about the authors of the books.

Columns:
id: A unique identifier for each author (Primary Key).
name: The name of the author.
biography: A brief biography or description of the author's background and work

Publishers Table

This table stores information about the publishing companies that published the books.

Columns:
id: A unique identifier for each publisher (Primary Key).
name: The name of the publishing company.
address: The location or address of the publisher.

Genres Table

This table categorizes books into different genres or types of literature.

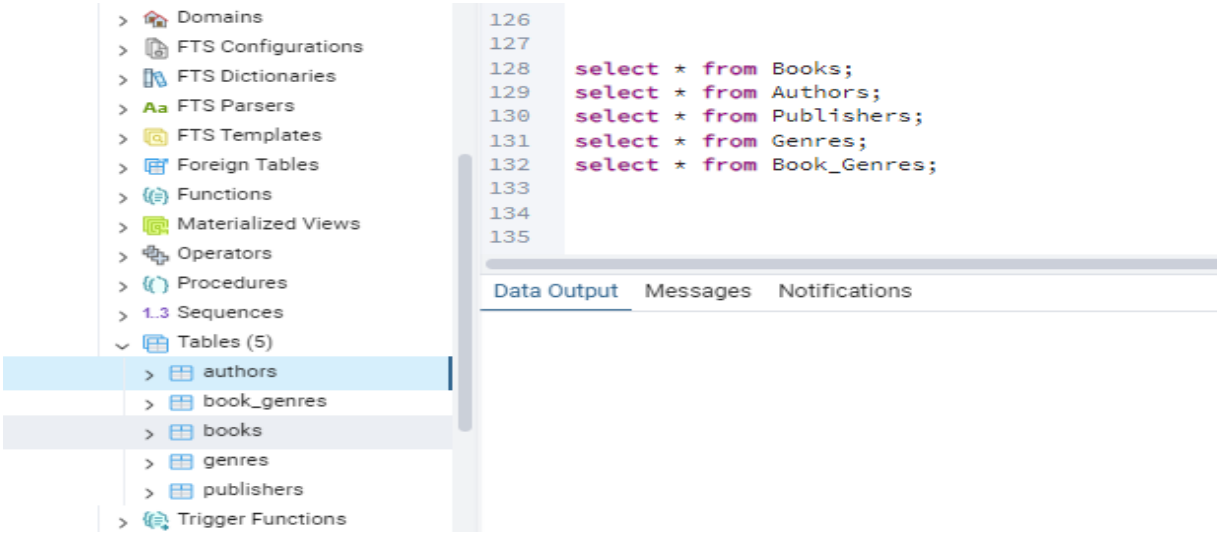
Columns:
id: A unique identifier for each genre (Primary Key).
name: The name of the genre (e.g., Fiction, Classics).
description: A brief description of the genre.

Book_Genres Table

This is a junction table that defines the many-to-many relationship between books and genres. It links specific books to one or more genres.

Columns:
id: A unique identifier for each entry (Primary Key).
book_id: A foreign key that references the id of a book in the books table.
genre_id: A foreign key that references the id of a genre in the genres table

Books can have one or more Genres, and a Genre can be assigned to multiple Books. The book_genres table facilitates this many-to-many relationship by associating each book with its respective genre(s).Authors and Publishers are directly linked to the Books table through author names (in this case) and separate records in their respective tables. Each book has one author and is published by one publisher, but each author can write multiple books, and publishers can publish multiple books.



2.Tables used in Patron Management database:

Patron Table

This table stores information about library patrons (members).

Columns:
id: A unique identifier for each patron (Primary Key).
name: The name of the patron.

email: The patron's email address.
phone_number: The contact number of the patron.
address: The residential address of the patron.

Membership Types table

This table defines different types of memberships available to patrons.

Columns:
id: A unique identifier for each membership type (Primary Key).
name: The name of the membership type (e.g., Basic, Premium).
description: A brief description of the membership type and its benefits.
duration: The duration of the membership in days (e.g., 365 days for a year).

Patron Membership Table

This table links patrons to their respective membership types and tracks the membership period.

Columns:
id: A unique identifier for each membership record (Primary Key).
patron_id: A foreign key referencing the id of the patron in the patrons table.
membership_type_id: A foreign key referencing the id in the membership_types table.
start_date: The date when the membership started.
end_date: The date when the membership ends or expires

Patron Fines Table

This table records any fines imposed on patrons, typically for overdue books or other infractions.

Columns
id: A unique identifier for each fine (Primary Key).
patron_id: A foreign key referencing the id of the patron in the patrons table.
fine_amount: The amount of the fine imposed on the patron.
fine_date: The date when the fine was imposed.

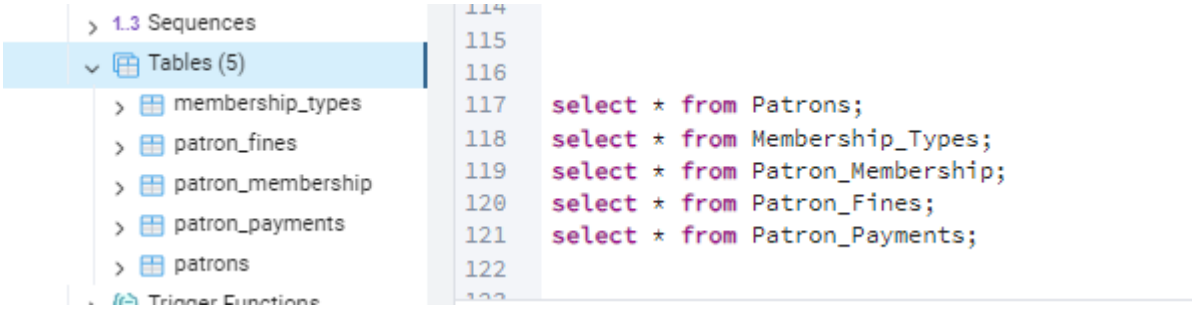
Patron Payments Table

This table tracks payments made by patrons, which could include membership fees, fines, or other charges.

Columns

id: A unique identifier for each payment (Primary Key).
patron_id: A foreign key referencing the id of the patron in the patrons table.
payment_amount: The amount of the payment made by the patron.
payment_date: The date when the payment was made.

Patrons have memberships, which are tracked in the patron_membership table. Each patron can accumulate fines, which are recorded in the patron_fines table. Patrons can make payments for memberships, fines, or other fees, tracked in the patron_payments table.



Tables used in Circulation Management database

Checkouts Table

This table records the details of books that have been checked out by patrons.

Columns

id: A unique identifier for each checkout record (Primary Key).

book_id: A foreign key referencing the id of the book in the books table.

patron_id: A foreign key referencing the id of the patron in the patrons table.

checkout_date: The date when the book was checked out.

due_date: The date by which the book should be returned.

Returns Table

This table records the details of books that have been returned.

Columns:

id: A unique identifier for each return record (Primary Key).

checkout_id: A foreign key referencing the id of the checkout record in the checkouts table.

return_date: The date when the book was returned

Holds Table

This table tracks books that patrons have placed a hold on, indicating they want to borrow it once it becomes available.

Columns:

id: A unique identifier for each hold record (Primary Key).

book_id: A foreign key referencing the id of the book in the books table.

patron_id: A foreign key referencing the id of the patron in the patrons table.

hold_date: The date when the hold was placed.

expiration_date: The date when the hold expires if the book is not yet available.

Waitlists Table

This table records the waitlists for books, indicating patrons who are waiting for a book that is currently checked out.

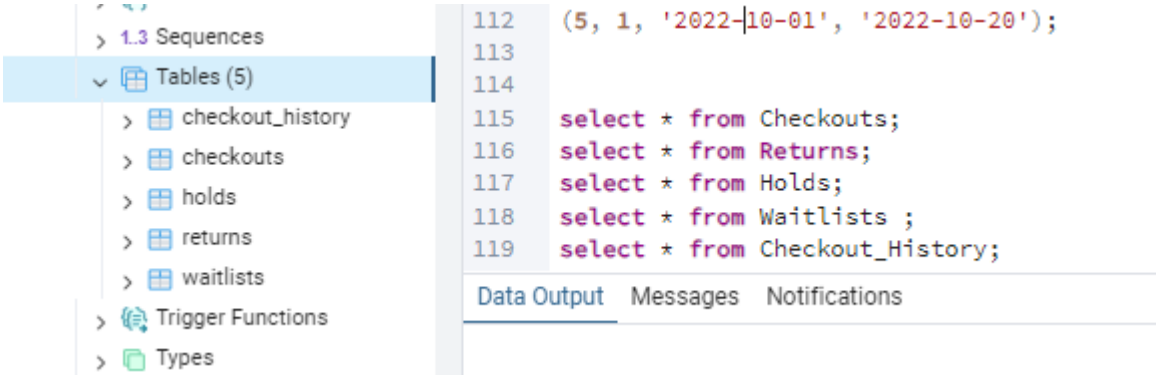
Columns:

id: A unique identifier for each waitlist record (Primary Key).

book_id: A foreign key referencing the id of the book in the books table.

patron_id: A foreign key referencing the id of the patron in the patrons table.

waitlist_date: The date when the patron was added to the waitlist.



Tables used in Acquisitions Management database:

Orders Table

This table tracks the details of book orders placed with vendors.

Columns:
id: A unique identifier for each order (Primary Key).
book_id: A foreign key referencing the id of the book in the books table.
quantity: The number of copies of the book ordered.
order_date: The date when the order was placed.
total_cost: The total cost of the order.

Vendors Table

This table stores information about vendors from whom books are purchased.

Columns:
id: A unique identifier for each vendor (Primary Key).
name: The name of the vendor.
address: The address of the vendor.
contact_info: The contact information for the vendor.

Order Items Table

This table provides detailed information about the individual items included in each order.

Columns:

id: A unique identifier for each order item (Primary Key).
order_id: A foreign key referencing the id of the order in the orders table.
book_id: A foreign key referencing the id of the book in the books table.
quantity: The number of copies of the book ordered in this item.
cost: The cost per copy of the book.

Receipts Table

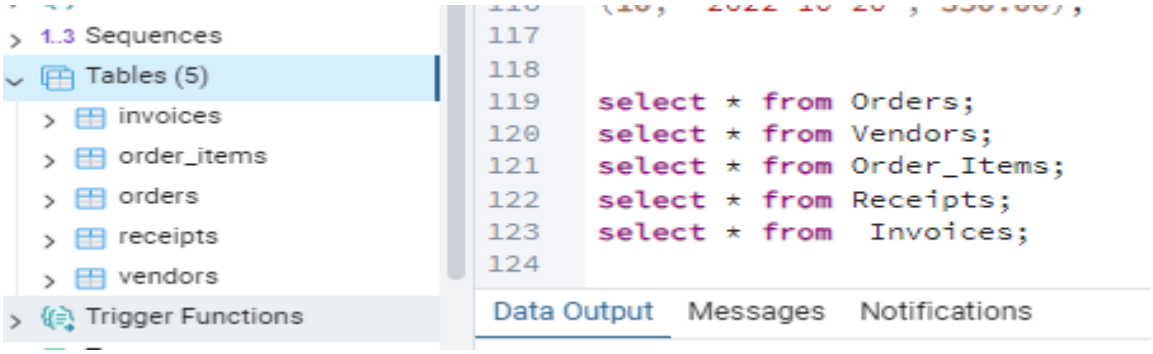
This table records the receipts issued for payments made for orders.

Columns:
id: A unique identifier for each receipt (Primary Key).
order_id: A foreign key referencing the id of the order in the orders table.
receipt_date: The date when the receipt was issued.
total_cost: The total amount reflected on the receipt.

Invoices Table

This table tracks the invoices issued for orders.

Columns:
id: A unique identifier for each invoice (Primary Key).
order_id: A foreign key referencing the id of the order in the orders table.
invoice_date: The date when the invoice was issued.
total_cost: The total amount listed on the invoice.



Tables used in Reporting and Analytics database:

Circulation Stats Table

Tracks the circulation metrics of the library, including checkouts and returns.

Columns:
id: A unique identifier for each record (Primary Key).
date: The date for which the statistics are reported.
total_checkouts: The total number of books checked out on that date.
total_returns: The total number of books returned on that date

Collection Stats Table

Provides information about the library's collection, including the number of books and authors.

Columns:
id: A unique identifier for each record (Primary Key).
date: The date for which the statistics are reported.
total_books: The total number of books in the collection on that date.
total_authors: The total number of authors represented in the collection on that date

Patron Stats Table

Tracks statistics related to library patrons and memberships.

Columns:
id: A unique identifier for each record (Primary Key).
date: The date for which the statistics are reported.
total_patrons: The total number of patrons registered on that date.
total_membership: The total number of active memberships on that date.

Fine Stats Table

Provides information on fines and payments related to overdue items.

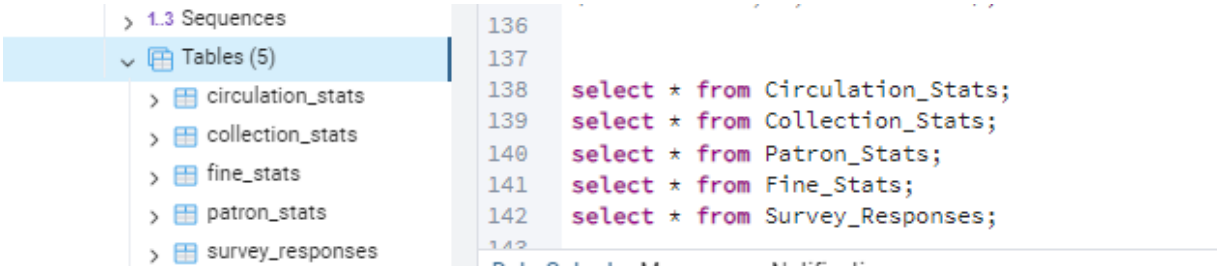
Columns:
id: A unique identifier for each record (Primary Key).

date: The date for which the statistics are reported.
total_fines: The total amount of fines accrued on that date.
total_payments: The total amount of payments made towards fines on that date.

Survey Responses Table

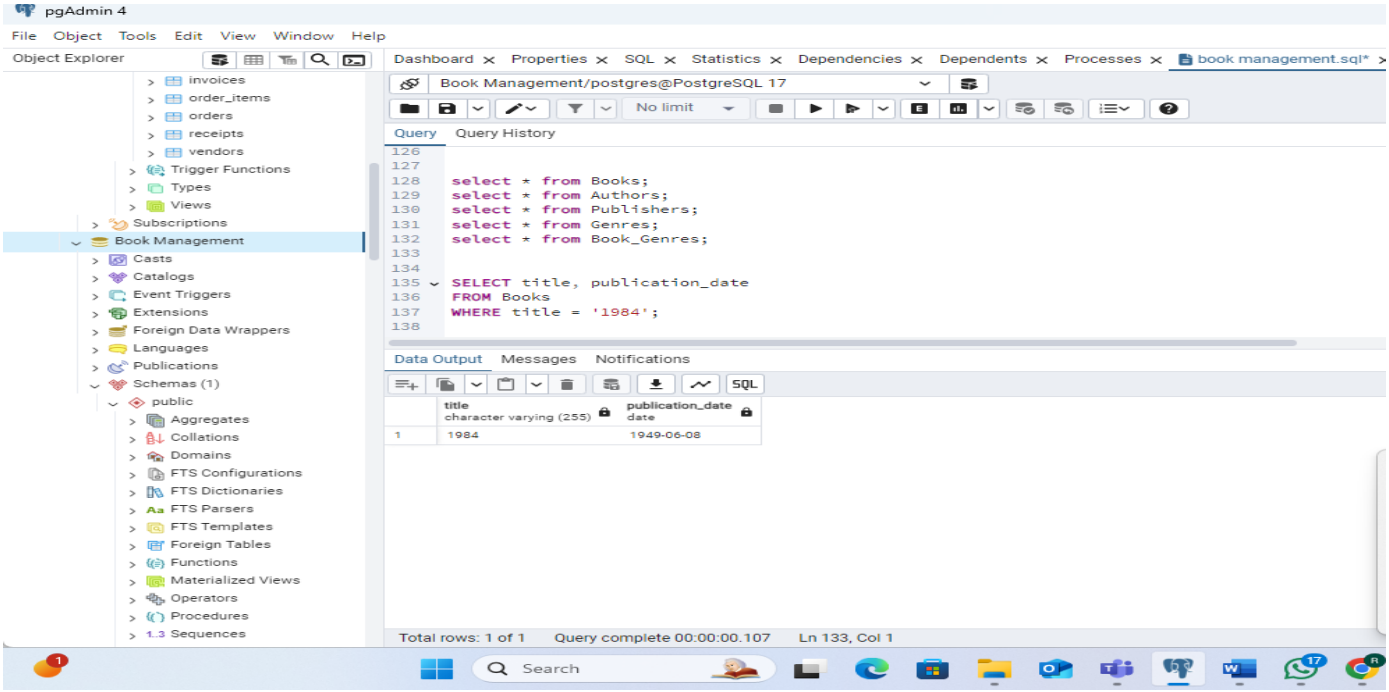
Records responses from surveys conducted for library service feedback.

Columns:
id: A unique identifier for each survey response (Primary Key).
date: The date the response was recorded.
question_id: An identifier for the survey question being answered.
response_text: The text of the patron's response to the question

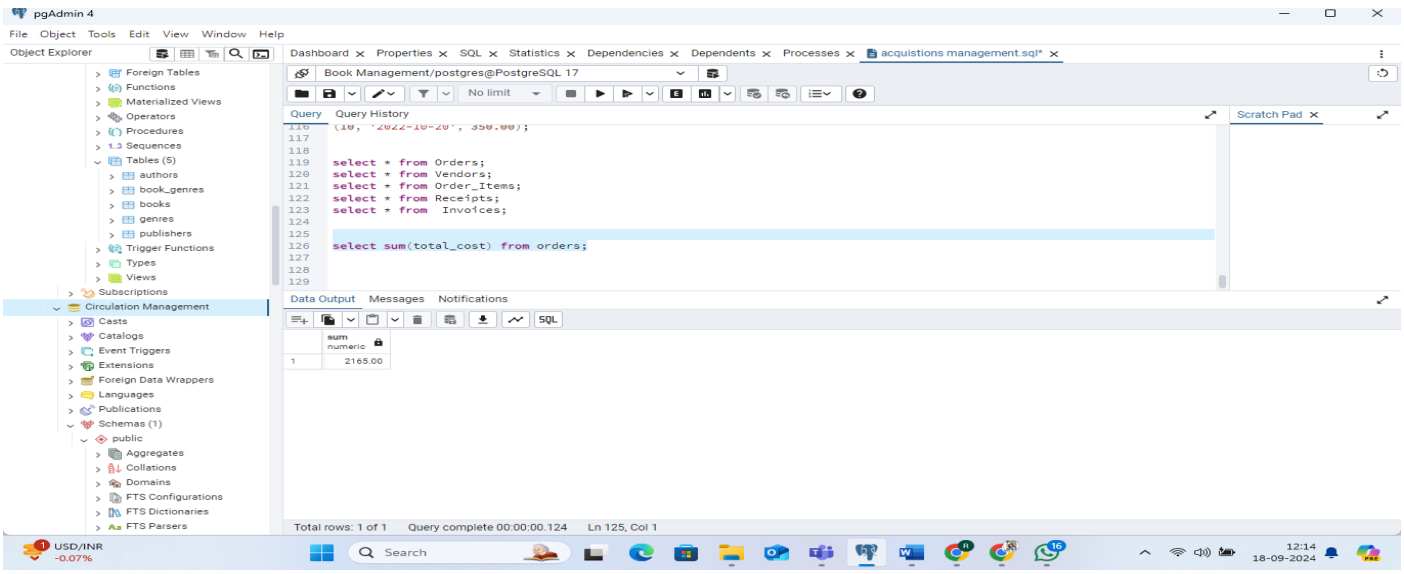


Some of the Queries

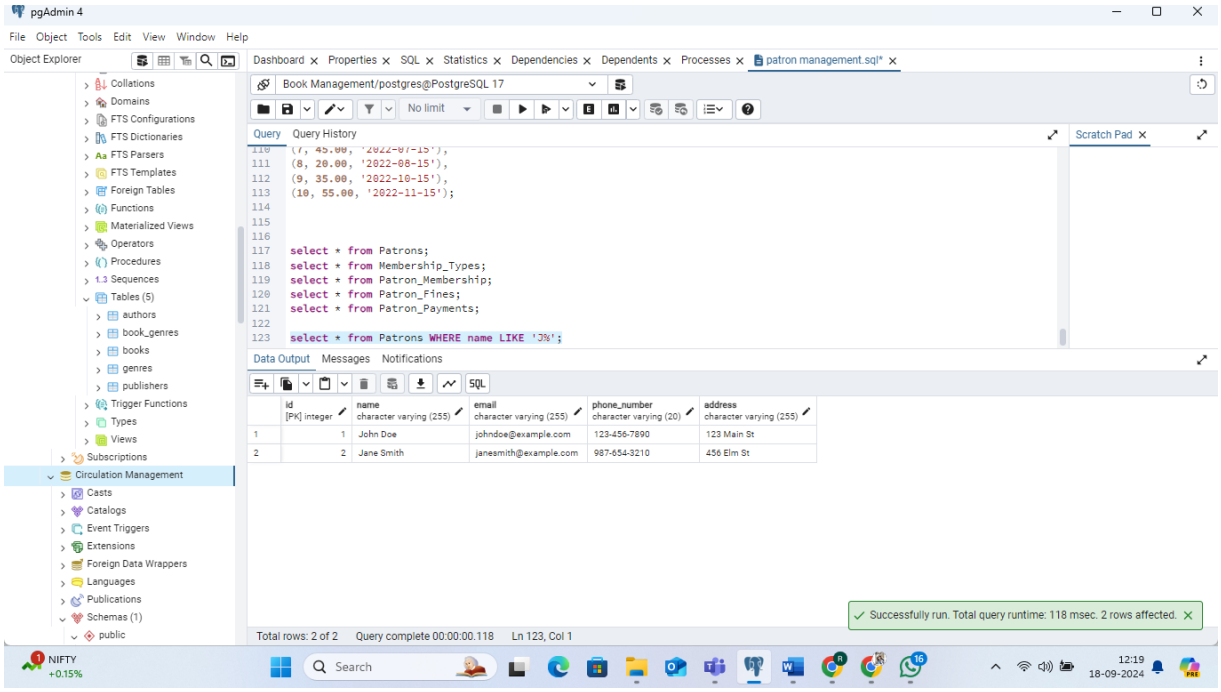
1.Retrieve record for the for the particular book title:



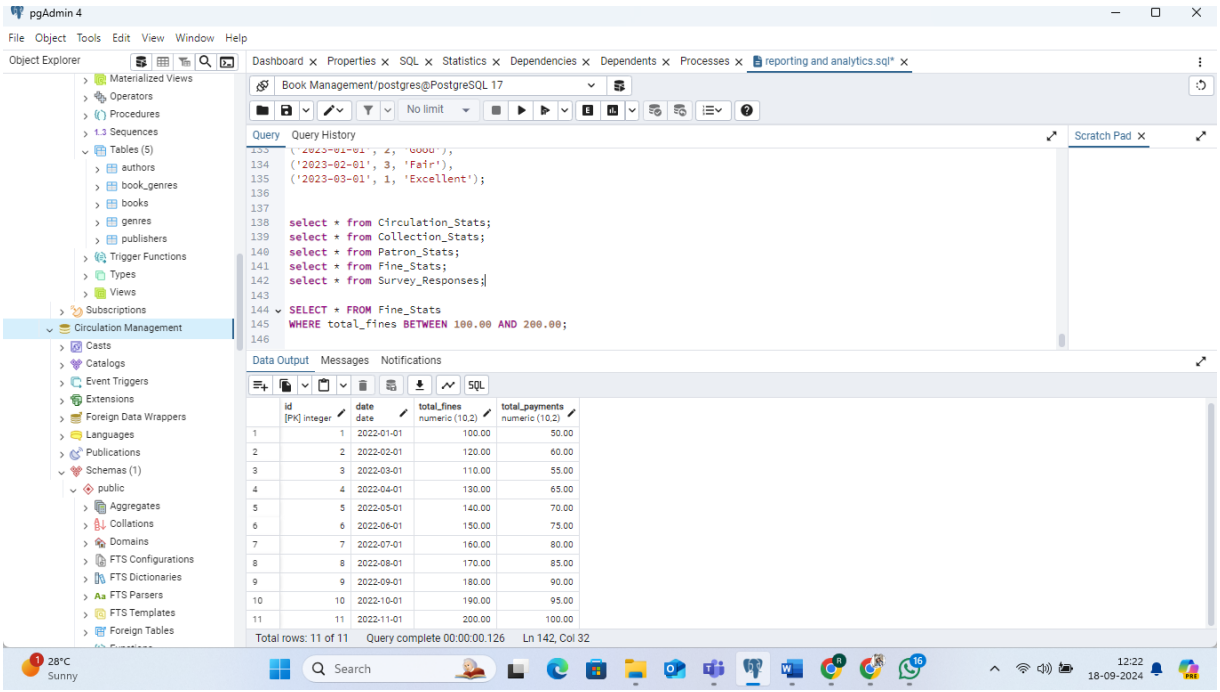
2.Get the total cost required for the all books



3.Retrieve all the names which are starting from J.



4.Retrieve the records whose total_fine date is between particular $100 and $200.



Final Goal of the Project

The goal of the Simple Library Management System project is to create a comprehensive database system that supports the efficient management of a library's operations. The project aims to

- Organize Library Data
- Streamline Library Operations
- Enhance User Experience
- Support Reporting and Analysis
- Facilitate Data Management
- Manage Acquisitions