

Project work:

Introduction:

The Bookstore Management System is a software application designed to simplify and automate the day-to-day operations of a bookstore. This project leverages **Structured Query Language (SQL)** to develop a database that provides a robust and scalable solution for managing inventory, sales, and customer information.

1. Database Creation:

Create the six tables (books, authors, customers, orders, orderdetails and genres) as described in the project structure.

books table column: bookid, title, authored, genrename, price, stock, quantity

authors table column: authorid, authername, city

customers table column: customerid, name, email, city

orders table column: orderid, customerid, orderdate

orderdetails table column: orderdetailid, orderid, bookid, price, quantity, orderamount

genres table column: ordered, customerid, orderdate

Database relationship:

Maintain the relationship (primary key, foreign key) between all the six tables

books : primary key (bookid)

authors: primary key (authorid)

customers: primary key (customerid)

orders: primary key (orderid)

orderdetails: primary key (orderdetailid)

genres: primary key (genrename)

Foreign key:

Books.AuthorID → Authors.AuthorID

Books.Genre → Genres.GenreName

Orders.CustomerID → Customers.CustomerID

OrderDetails.OrderID → Orders.OrderID

OrderDetails.BookID → Books.BookID

2. SQL queries:

Create a database named bookstores and import all six tables in dbeaver by connected MySQL. Maintain the data type when create six tables in dbeaver. After that set the relation between all six tables by primary key and foreign key.

Containing the queries:

Objective 1: Subqueries

- 1.1. Find books priced above the average price for their genre.
- 1.2: Identify customers with purchases exceeding a specific value using subqueries.
- 2. Display each genre with the total stock of books using a subquery.

Objective 2: Common Table Expressions (CTEs)

- 2.1: List the top 5 customers with the highest spending in a specific year.
- 2.2: Track the total number of books ordered monthly using a recursive CTE.

Objective 3: JOIN Operations

- 3.1: Display a list of books with their authors, genres, and stock quantity.
- 3.2: List orders with the total cost, including book titles and quantities.

Objective 4: Handling NULLs

- 4.1: Show all customers with NULL if they have no orders.
- 4.2: List genres with the number of books available, showing zero for genres with no books.

Objective 5: Query Optimization

- 5.1. Create and optimize queries using indexes and EXPLAIN statements.
- 5.2. Rewrite the query for the top 5 highest-priced books to improve performance.

