📊 SQL Analysis Report – Online Bookstore Project

🎯 Goal:

Perform real-time SQL analysis on bookstore data to extract business insights using filtering, joining, grouping, and aggregation techniques.

```
-- Get all books where the genre is Fiction
select * from books
where Genre = 'Fiction';

-- Get all books published after the year 1950
select * from books
where Published_Year > 1950;

-- Get all customers who are from Canada
select * from customers
where Country = 'Canada';

-- Get all orders placed between 1st Nov 2023 and 30th Nov 2023
select * from orders
where Order_Date between '2023-11-01' and '2023-11-30';

-- Calculate the total stock of all books
select sum(stock) as total_stock
from books;

-- Get all orders where quantity is more than 1
select * from orders
where Quantity > 1;

-- Get all unique book genres
select distinct Genre
from books;

-- Get the book that has the lowest stock
```

```
select * from books
order by Stock
limit 1;

-- Calculate the total revenue from all orders
select sum(Total_Amount) as Total_revenew
from orders;

-- Get total books sold by genre
select b.Genre, sum(o.Quantity) as Total_books_sold
from Orders o
join books b on o.book_id = b.book_id
group by b.Genre;

-- Get average price of books in the Fantasy genre
select Genre, avg(Price) as Avrage_price
from books
where Genre = 'Fantasy';

-- Get customers who have placed 2 or more orders
select o.Customer_ID, c.name, count(o.Order_ID) as Order_Count
from orders o
join customers c ON o.customer_id = c.customer_id
group by o.Customer_ID, c.name
having count(Order_ID) >= 2;

-- Get the most ordered book (by number of orders)
select o.Book_ID, b.title, count(o.Order_ID) as Order_count
from orders o
join books b on o.book_id = b.book_id
group by o.Book_ID, b.title
order by Order_count desc
limit 1;

-- Get top 3 most expensive Fantasy books
```

```sql
select * from books
where Genre = 'Fantasy'
order by Price desc
limit 3;

-- Get total books sold by each author
select b.Author, sum(o.Quantity) as Books_Sold
from orders o
join books b on o.book_id = b.book_id
group by b.Author;

-- Get cities (no repeat) where customers placed orders above $30
select distinct c.city, Total_Amount
from orders o
join customers c on o.customer_id = c.customer_id
where o.total_amount > 30;

-- Get the customer who spent the most money
select c.customer_id, c.name, sum(o.total_amount) as Total_spent
from orders o
join customers c on o.Customer_ID = c.customer_id
group by c.customer_id, c.name
order by Total_spent desc
limit 1;

-- Get each book's stock, order quantity, and remaining quantity
select b.book_id, b.title, b.stock, coalesce(sum(o.quantity), 0) as Order_quantity,
b.stock - coalesce(sum(o.quantity), 0) as remaining_Quantity
from books b
left join orders o on b.book_id = o.book_id
group by b.book_id;
```