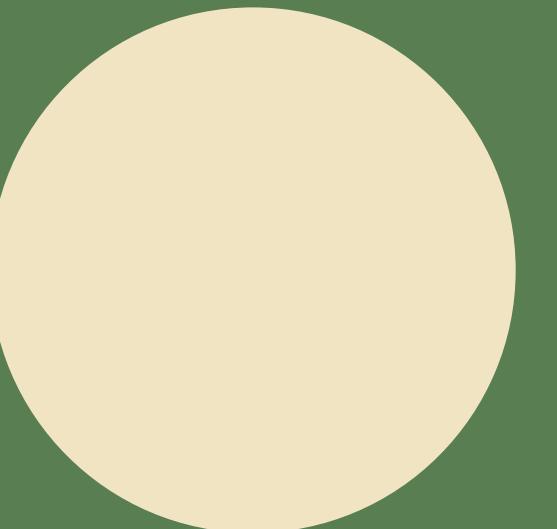


Book Store Sales SQL Project

June 2025

HELLO!



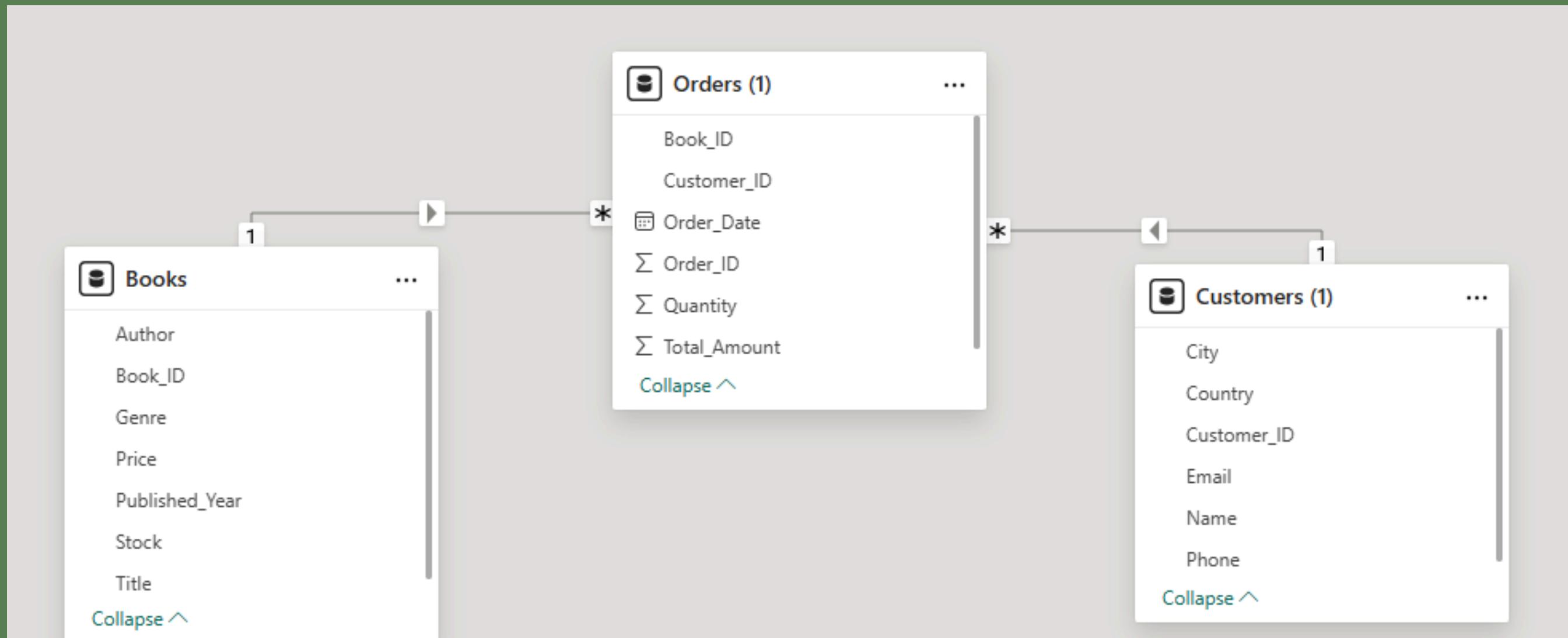
My name is Darshan Dobariya, I used SQL queries in this Project to find Meaningfull insights and Trends for a book store to grow business.

Analyzed book store sales data using SQL to uncover key business insights such as top-selling books, customer purchase trends, revenue by category, and monthly performance. Focused on data cleaning, aggregation, and optimization through SQL queries.

• • • •
• • • •
• • • •
• • • •
• • • •
• • • •
• • • •



Model



Retrieve the total number of books sold for each genre

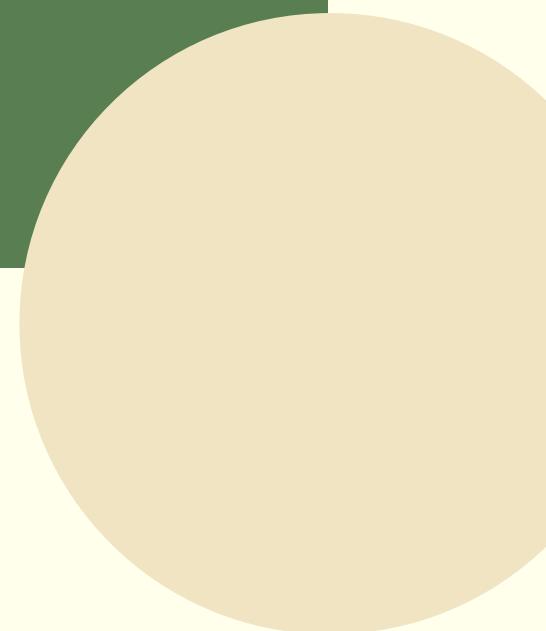
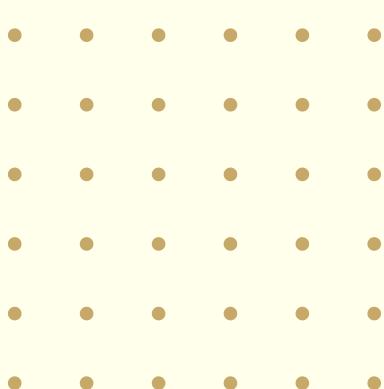
```
select books.Genre, sum(orders.Quantity) as total_quantity  
from books join orders  
on books.Book_id = orders.Book_ID  
group by books.Genre  
order by total_quantity;
```

	Genre	total_quantity
▶	Fiction	225
	Biography	285
• • • • •	Non-Fiction	351
• • • • •	Romance	439
• • • • •	Fantasy	446
• • • • •	Science Fiction	447
• • • • •	Mystery	504

Find the average price of books in the "Fantasy" genre

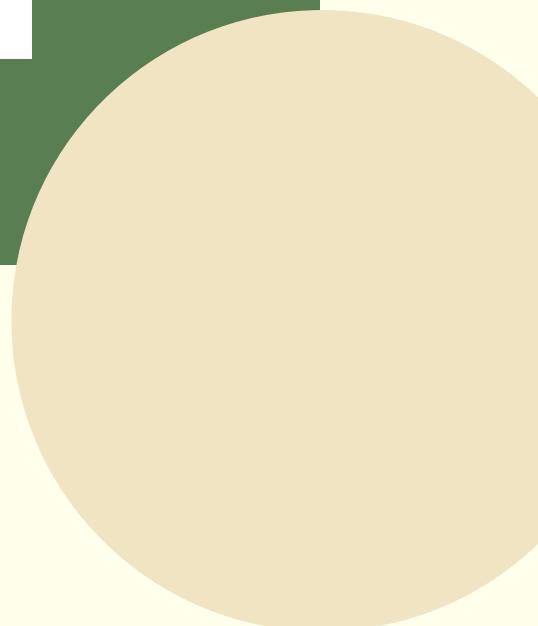
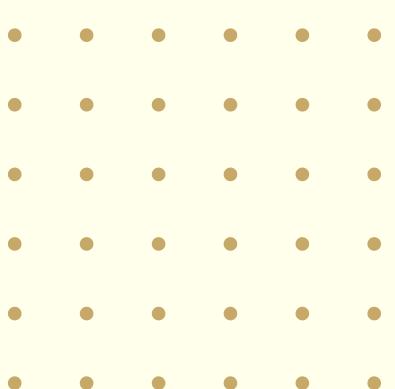
```
SELECT  
    ROUND(AVG(price), 2) AS avg_price  
FROM  
    books  
WHERE  
    Genre = 'Fantasy';
```

	avg_price
▶	25.98



List customers who have placed at least 2 orders.

```
select o.Customer_Id, c.Name, count(o.Order_ID) as orders  
from orders o  
join customers c  
on o.Customer_Id = c.Customer_ID  
group by o.Customer_Id, c.Name  
having orders >= 2;
```



Find the most frequently ordered book

```
select orders.Book_ID, books.title, count(orders.Order_ID) as order_count
from orders
join books
on orders.Book_id = books.Book_id
group by orders.Book_id, books.title
order by order_count desc
limit 1;
```

	Book_ID	title	order_count
▶	88	Robust tangible hardware	4

• • • • •
• • • • •
• • • • •
• • • • •
• • • • •
• • • • •
• • • • •

Show the top 3 most expensive books of 'Fantasy' Genre

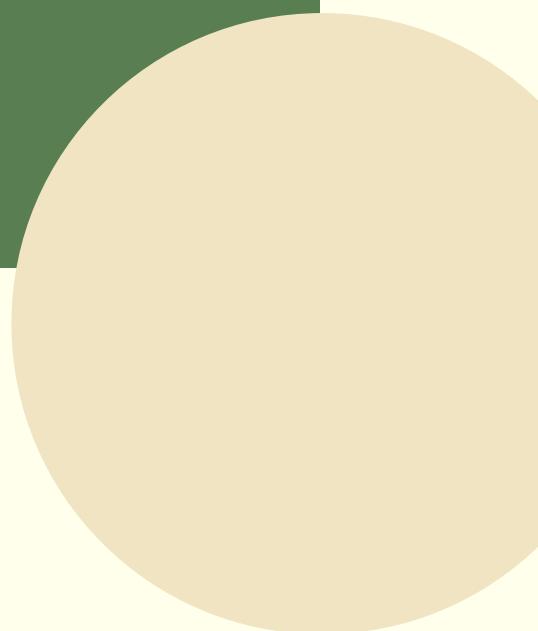
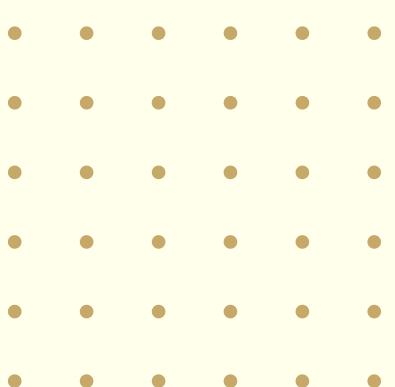
```
select title, Genre, price from books
where Genre = "fantasy"
order by price desc
limit 3;
```

	title	Genre	price
▶	Stand-alone content-based hub	Fantasy	49.90
	Innovative 3rdgeneration database	Fantasy	49.23
	Optimized even-keeled analyzer	Fantasy	48.97

• • • • •
• • • • •
• • • • •
• • • • •
• • • • •
• • • • •
• • • • •

Retrieve the total quantity of books sold by each author

```
select books.Author, sum(orders.Quantity) as total_quantity  
from books  
join orders  
on books.Book_id = orders.Book_ID  
group by books.Author  
order by total_quantity;
```



List the cities where customers who spent over \$30 are located

```
select customers.City, sum(orders.Total_Amount) as total_amt  
from customers  
join orders  
on customers.Customer_ID = orders.Customer_Id  
group by customers.City  
having total_amt > 30  
order by total_amt;
```

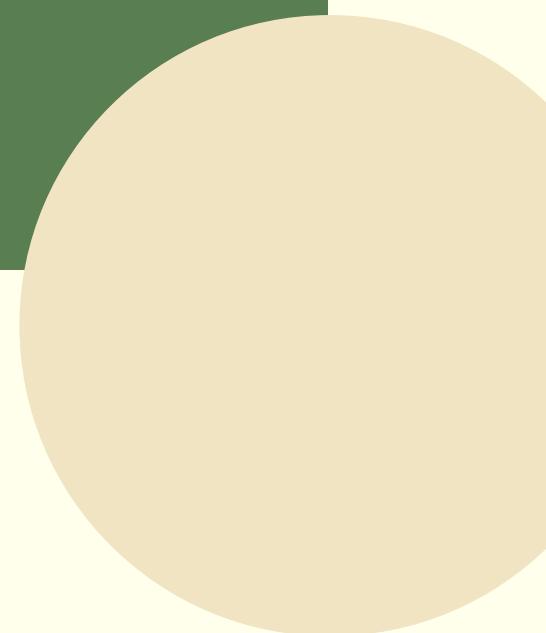
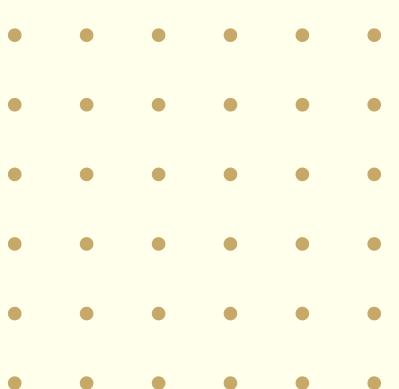
OR

```
select distinct customers.City, orders.Total_Amount  
from customers  
join orders  
on customers.Customer_ID = orders.Customer_Id  
where orders.Total_Amount > 30;
```

Find the customer who spent the most on orders

```
select customers.name, sum(orders.Total_Amount) as total_amt  
from customers  
join orders  
on customers.Customer_ID = orders.Customer_Id  
group by customers.name  
order by total_amt desc  
limit 1;
```

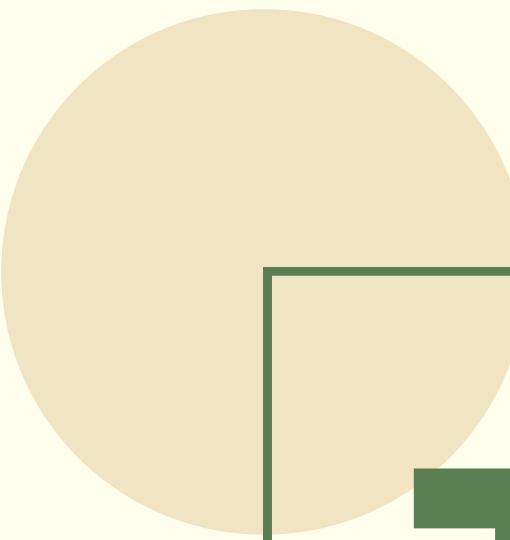
	name	total_amt
▶	Kim Turner	1398.90



Calculate the stock remaining after fulfilling all orders

```
SELECT  
    books.Book_id,  
    books.title,  
    books.Stock,  
    COALESCE(SUM(orders.quantity), 0) AS order_quantity,  
    books.Stock - COALESCE(SUM(orders.quantity), 0)  
FROM  
    books  
        LEFT JOIN  
    orders ON books.Book_id = orders.Book_ID  
GROUP BY books.Book_id;
```

	name	total_amt
▶	Kim Turner	1398.90



Thank You

FOR YOUR ATTENTION

June 2025