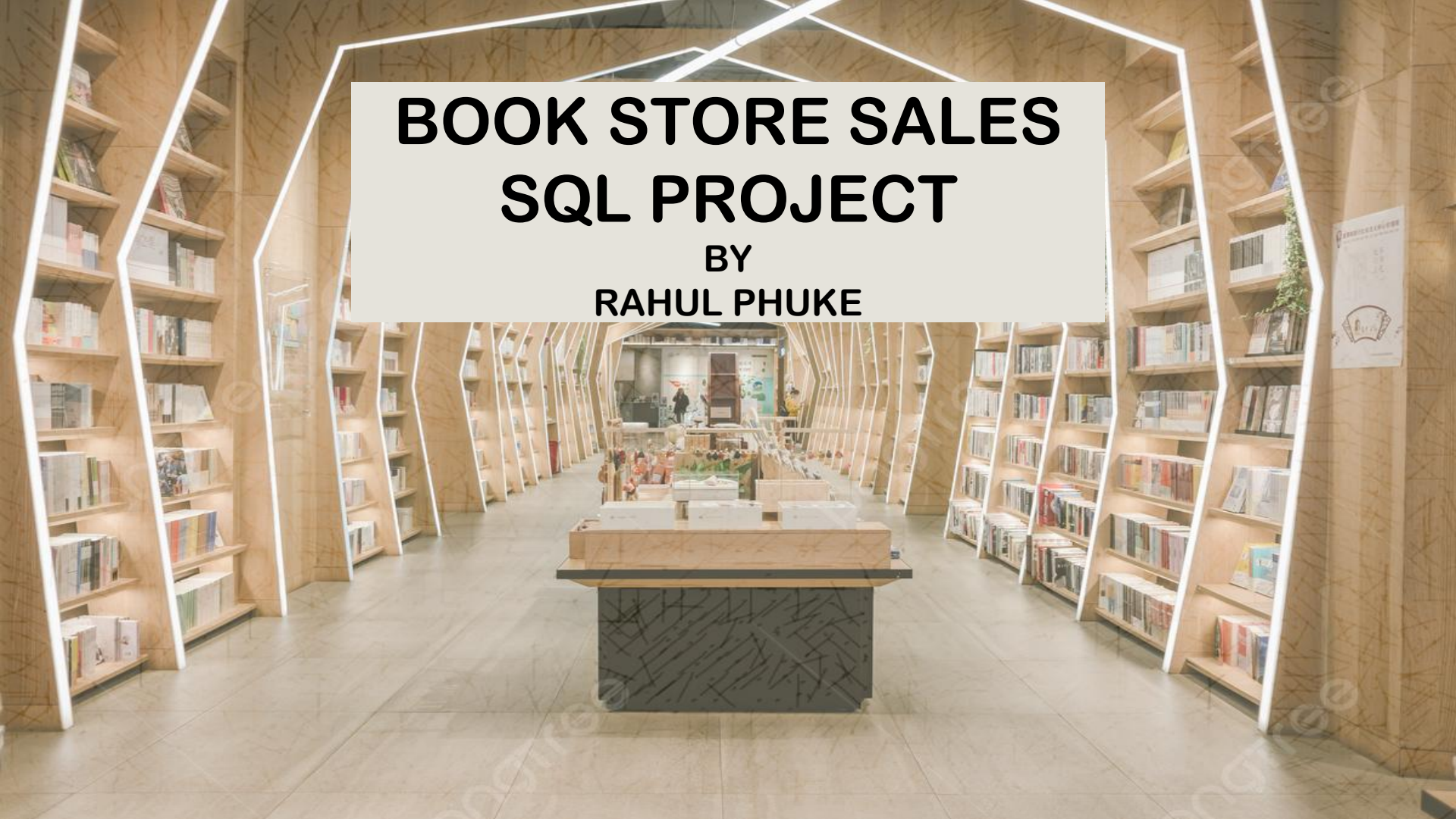


BOOK STORE SALES SQL PROJECT

BY

RAHUL PHUKE

Bookstore Sales Analysis Overview

A Data Analyst's Perspective

Objective
Analyze sales trends, customer behavior, and inventory performance using SQL-based data from a bookstore database.

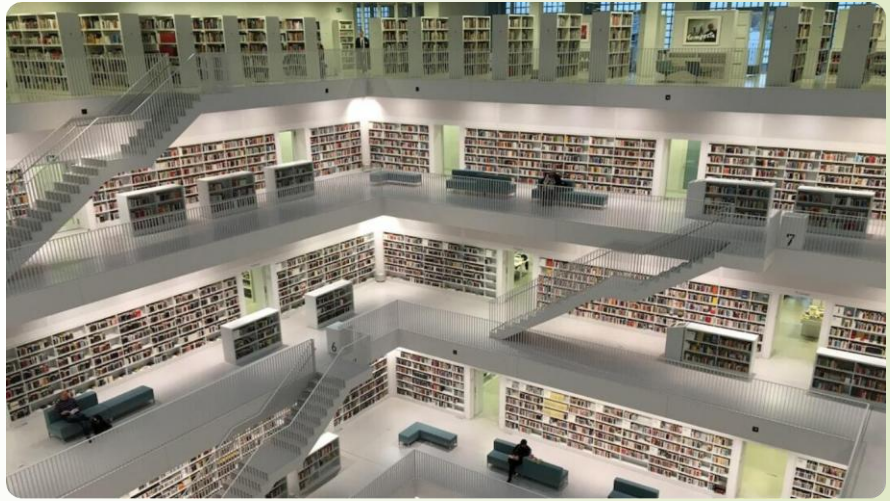
Tools Used
MySQL for querying data; Excel for exporting results; SlideGPT for presentation generation.

Data Source
Relational database with transactional data, customer profiles, and inventory records from a retail bookstore.

Database Schema Overview

Understanding the Structure Behind Bookstore Sales

- **Tables and Entities:** Key tables include Sales, Customers, Books, Inventory, and Authors with unique identifiers and foreign key relations.
- **Schema Logic:** Sales table links Customers and Books; Inventory tracks available units per title; Authors tied to Books metadata.
- **Analytical Relevance:** Normalized schema enables efficient querying and integrity across sales, customer profiles, and product data.



Key Sales Metrics

Revenue, Bestsellers & Trends

Total Revenue
Calculated by summing prices from the Sales table. Q1 showed highest revenue spike with $78,300.

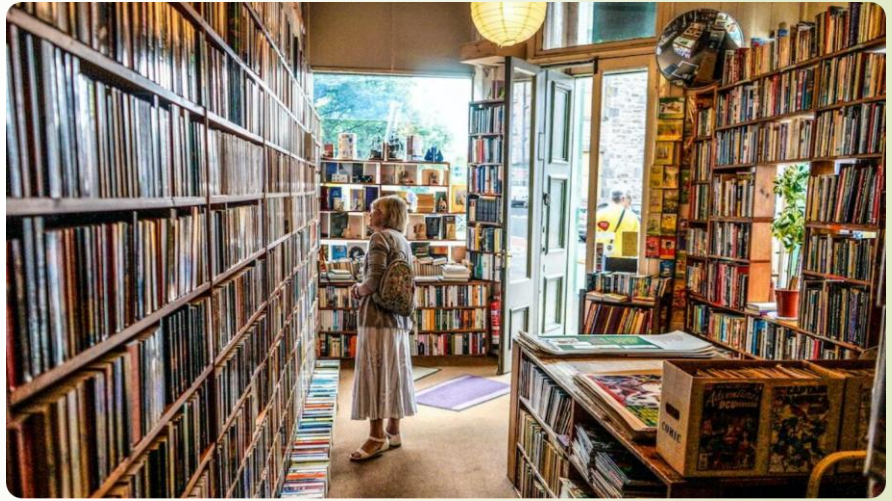
Top-Selling Books
'Atomic Habits', 'Sapiens', and 'The Midnight Library' topped sales in volume and value.

Monthly Trends
Sales peaked in March and November, suggesting seasonal promotions and holiday effects.

Customer Insights

Behavioral Patterns & Segmentation

- **Customer Segments:** Majority of customers are one-time buyers (62%), while 38% are repeat customers with higher basket values.
- **Average Order Value:** Repeat customers spend an average of $42/order vs. $27/order for one-time buyers.
- **Engagement Patterns:** High-value customers show preference for non-fiction and purchase during promotional periods.



Inventory & Stock Analysis

Managing Book Turnover and Shelf Space

Unsold Inventory
23% of stocked titles had zero sales, mostly academic or niche-interest books.

Top-Moving Titles
Best-performing books turned over stock every 3–4 weeks, requiring restocking twice a quarter.

Stock Efficiency
Sales-to-stock ratio highest in fiction and self-help; lowest in reference categories.

Seasonal Sales Patterns

Understanding Timing and Promotional Peaks

Peak Sales Periods
Sales spike in March and November, aligning with academic terms and holiday shopping.

Promotional Impact
Sales uplift of 18–22% during targeted discounts and campaigns.

Category Variation
Fiction and self-help books surged during holidays; academic titles peaked in August–September.

Strategic Recommendations

Data-Driven Improvements for Sales and Operations

Inventory Optimization
Phase out unsold titles and increase stock of high-turnover categories like fiction and self-help.

Targeted Campaigns
Focus promotions during known peak months and personalize offers based on customer behavior.

Customer Retention
Launch loyalty programs and re-engagement email campaigns for high-value repeat buyers.

Conclusion

Final Takeaways from the Bookstore Sales Analysis

Data-Driven Insights
Sales patterns, stock performance, and customer behavior were all quantifiably assessed through SQL analysis.

Seasonal & Genre Trends
March, August, and November saw revenue peaks; fiction and self-help genres led performance.

Strategic Direction
Optimize inventory, personalize marketing, and retain top-tier customers through loyalty initiatives.

Retrieve all books in the "Fiction" genre

```
SELECT BOOK_ID,TITLE,GENRE FROM BOOKS WHERE GENRE = "Fiction";
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

BOOK_ID	TITLE	GENRE
4	Customizable 24hour product	Fiction
22	Multi-layered optimizing migration	Fiction
28	Expanded analyzing portal	Fiction
29	Quality-focused multi-tasking challenge	Fiction
31	Implemented encompassing conglomeration	Fiction
39	Optimized national process improvement	Fiction
40	Adaptive didactic interface	Fiction
47	Reverse-engineered directional conglomeration	Fiction
62	Re-contextualized real-time strategy	Fiction
63	Polarized heuristic database	Fiction
100	Synchronized client-server service-desk	Fiction
116	Multi-tiered foreground contingency	Fiction
125	Public-key analyzing Graphic Interface	Fiction
130	Realigned context-sensitive pricing structure	Fiction
134	Polarized bandwidth-monitored throughput	Fiction
142	Multi-tiered responsive parallelism	Fiction
143	Networked multimedia support	Fiction
144	Future-proofed scalable software	Fiction
156	Synergistic grid-enabled website	Fiction

Find books published after the year 1950

SELECT BOOK_ID,TITLE,PUBLISHED_YEAR FROM BOOKS WHERE PUBLISHED_YEAR > 1950 order by PUBLISHED_YEAR;

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

BOOK_ID	TITLE	PUBLISHED_YEAR
166	Customizable discrete Graphical User Interface	1951
174	Pre-emptive executive knowledge user	1951
432	Horizontal disintermediate alliance	1951
43	Function-based zero-defect initiative	1952
150	Phased logistical open system	1952
62	Re-contextualized real-time strategy	1953
156	Synergistic grid-enabled website	1953
243	Automated systemic toolset	1953
457	Configurable disintermediate extranet	1953
167	User-friendly radical standardization	1954
193	Customer-focused tertiary methodology	1954
242	Business-focused responsive parallelism	1954
37	Up-sized tertiary archive	1955
49	Robust attitude-oriented attitude	1955
102	Ergonomic stable open architecture	1955
134	Polarized bandwidth-monitored throughput	1955
293	Secured object-oriented secured line	1955
365	Synchronized impactful synergy	1955
5	Adaptive 5thgeneration encoding	1956

1955

Basic SQL Queries

List all customers from the Canada

```
SELECT CUSTOMER_ID,NAME,EMAIL,COUNTRY FROM CUSTOMERS
WHERE COUNTRY ="Canada";
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CUSTOMER_ID	NAME	EMAIL	COUNTRY
38	Nicholas Harris	christine93@perkins.com	Canada
415	James Ramirez	robert54@hall.com	Canada
468	David Hart	stokesrebecca@gmail.com	Canada

Show orders placed in November 2023

```
SELECT ORDER_ID, BOOK_ID, QUANTITY, ORDER_DATEFROM ORDERSWHERE
MONTH(ORDER_DATE) = 11 AND YEAR(ORDER_DATE) = 2023;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ORDER_ID	BOOK_ID	QUANTITY	ORDER_DATE
4	343	7	2023-11-25
19	60	9	2023-11-17
75	375	5	2023-11-30
132	333	7	2023-11-22
137	471	8	2023-11-25
163	384	3	2023-11-23
182	293	7	2023-11-01
200	303	1	2023-11-23
213	447	7	2023-11-17
231	384	1	2023-11-11
245	97	9	2023-11-01
252	387	5	2023-11-15
257	403	1	2023-11-06
288	128	1	2023-11-13
307	133	1	2023-11-17
322	112	2	2023-11-08
344	218	5	2023-11-25
389	391	2	2023-11-18
414	234	1	2023-11-10

Retrieve the total stock of books available

```
SELECT sum(STOCK) AS TOTAL_STOCK FROM BOOKS;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	TOTAL_STOCK
▶	25056

#Find the details of the most expensive book

SELECT BOOK_ID,TITLE,AUTHOR,GENRE,PUBLISHED_YEAR,PRICE FROM BOOKS order by PRICE DESC LIMIT 1;

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

BOOK_ID	TITLE	AUTHOR	GENRE	PUBLISHED_YEAR	PRICE
340	Proactive system-worthy orchestration	Robert Scott	Mystery	1907	49.98

Show all customers who ordered more than 1 quantity of a book

```
SELECT CUSTOMER_ID,QUANTITY FROM ORDERS WHERE QUANTITY >1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CUSTOMER_ID	QUANTITY
84	8
137	10
216	6
433	7
14	7
439	5
195	6
32	4
109	9
94	4
454	2
420	5
454	2
127	6
412	8
462	5
377	4
496	9
356	3

Basic SQL Queries

Retrieve all orders where the total amount exceeds $20

SELECT Order_ID,ORDER_DATE,TOTAL_AMOUNT FROM ORDERS WHERE TOTAL_AMOUNT>20 ORDER BY TOTAL_AMOUNT;

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Order_ID	ORDER_DATE	TOTAL_AMOUNT
307	2023-11-17	20.96
100	2023-07-14	22.38
260	2024-05-07	22.56
319	2024-06-27	22.98
345	2023-07-04	23.32
494	2024-11-23	23.32
288	2023-11-13	24.04
188	2023-03-13	24.23
394	2024-04-14	24.23
272	2024-11-02	24.23
415	2023-07-07	24.63
499	2023-08-26	24.7
172	2024-09-20	25.63
251	2023-12-12	25.76
187	2023-12-30	25.83
220	2023-07-07	25.83
271	2023-07-01	25.97
192	2024-06-24	26.34
129	2023-08-02	26.62

List all genres available in the Books table

SELECT distinct GENRE FROM BOOKS;

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

GENRE
Biography
Fantasy
Non-Fiction
Fiction
Romance
Science Fiction
Mystery

Find the book with the lowest stock

```
SELECT BOOK_ID,TITLE,AUTHOR,STOCK FROM BOOKS order by STOCK
LIMIT 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

BOOK_ID	TITLE	AUTHOR	STOCK
378	Future-proofed heuristic function	Samantha Mcclain	0

Basic SQL Queries

Calculate the total revenue generated from all orders

```
SELECT round(SUM(TOTAL_AMOUNT),2) AS "TOTAL_REVENUE($)" FROM
ORDERS;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	TOTAL_REVENUE($)
▶	75628.66

Advance SQL Queries

Retrieve the total number of books sold for each genre

SELECT B.GENRE,sum(O.QUANTITY) AS BOOKS_SOLDFROM ORDERS AS O JOIN BOOKS AS B ON B.BOOK_ID=O.BOOK_IDgroup by B.GENRE ORDER BY BOOKS_SOLD DESC;

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

GENRE	BOOKS_SOLD
Mystery	504
Science Fiction	447
Fantasy	446
Romance	439
Non-Fiction	351
Biography	285
Fiction	225

Advance SQL Queries

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

```
SELECT GENRE,round(AVG(price),2) as AVG_PRICE FROM BOOKS WHERE
GENRE= "Fantasy";
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	GENRE	AVG_PRICE
▶	Fantasy	25.98

Advance SQL Queries

List customers who have placed at least 2 orders

SELECT CUSTOMER_ID, COUNT(ORDER_ID) AS NO_OF_ORDERSFROM ORDERSGROUP BY CUSTOMER_IDHAVING COUNT(ORDER_ID) >= 2;

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CUSTOMER_ID	NO_OF_ORDERS
84	2
137	2
216	2
14	2
195	3
109	2
94	3
131	2
454	2
420	3
462	3
377	2
177	2
119	3
265	3
305	2
438	2
386	3
463	3

Advance SQL Queries

Find the most frequently ordered book

SELECT B.TITLE, COUNT(O.ORDER_ID) AS ORDER_COUNTFROM ORDERS AS OJOIN BOOKS AS B ON O.BOOK_ID = B.BOOK_IDGROUP BY B.BOOK_ID, B.TITLEORDER BY ORDER_COUNT DESCLIMIT 1;

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

	TITLE	ORDER_COUNT
▶	Implemented encompassing conglomeration	4

Advance SQL Queries

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;

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

TITLE	GENRE	PRICE
Stand-alone content-based hub	Fantasy	49.9
Innovative 3rdgeneration database	Fantasy	49.23
Optimized even-keeled analyzer	Fantasy	48

49.23

Advance SQL Queries

Retrieve the total quantity of books sold by each author

select B.AUTHOR, sum(O.QUANTITY) AS BOOKS_SOLD FROM ORDERS AS OJOIN BOOKS AS B ON O.BOOK_ID=B.BOOK_IDGROUP BY B.AUTHOR order by BOOKS_SOLD DESC;

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

AUTHOR	BOOKS_SOLD
Patrick Contreras	28
Melissa Taylor	27
Thomas Trujillo	24
Emily James	24
Sheena Harris	23
Ellen Doyle	23
Erica Parker	23
Valerie Moore	23
Amanda Wilson	22
Rachel Gibbs	22
Kristi Phillips	21
Tonya Saunders	21
Anna Roberts	21
Alexander Wallace	20
Michael Hill	20
Stacy Cabrera	19
Zachary Williams	19
Paul Miles	19
Robert Bullock	19

Advance SQL Queries

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

SELECT distinct C.CITY,O.TOTAL_AMOUNT FROM ORDERS AS OJOIN CUSTOMERS AS C ON C.Customer_ID=O.Customer_IDWHERE O.TOTAL_AMOUNT > 30.0 order by O.TOTAL_AMOUNT DESC;

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

CITY	TOTAL_AMOUNT
Smithborough	491.5
Fieldsland	489.6
Thomaschester	489.6
Mendezburgh	486.7
Melissaside	480.3
Freemanland	469.7
Bakerton	469.3
West Kimberlyhaven	466.6
Elizabethshire	465.4
Lake Charleshaven	459.1
Buckbury	452
Crystalborough	449.01
Erikaberg	446.31
Lake Tyler	445.5
Jenniferfurt	442.8
South Rachelview	440.7
Angelastad	426.1
South Rachelview	421.9
South John	419.4

Advance SQL Queries


```
# Find the customer who spent the most on orders
SELECT distinct C.Customer_ID,C.Name,ROUND(sum(O.TOTAL_AMOUNT),2) AS
TOTAL_SPENTFROM ORDERS AS OJOIN CUSTOMERS AS C ON
C.Customer_ID=O.Customer_IDgroup by C.CUSTOMER_ID,C.NAMEORDER BY
TOTAL_SPENT DESC LIMIT 1;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

Customer_ID	Name	TOTAL_SPENT
457	Kim Turner	1398.9

THANK YOU….!