📊 Online Bookstore Project - Data Insights

This document provides SQL-driven insights into the performance of an online bookstore using three key datasets: **Books**, **Customers**, and **Orders**.

🔗 Summary of Tables

- **Books:** Contains book details like title, author, genre, price, stock.
- **Customers:** Contains customer details such as name, location, and contact.
- **Orders:** Contains information on transactions including quantity, date, and total amount.

📈 1. Total Revenue Generated

```sql
SELECT SUM(total_amount) AS Revenue FROM Orders;
```

Insight: Total revenue generated from all book sales.

📘 2. Most Frequently Ordered Book

```sql
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;
```

Insight: Book with the highest number of orders.

🕐 3. Top 3 Most Expensive Books (Fantasy Genre)

```sql
SELECT title, genre, price FROM books
WHERE genre = 'Fantasy'
ORDER BY price DESC
LIMIT 3;
```

Insight: Highlights premium offerings in the fantasy genre.

4. Repeat Customers (2+ Orders)

```sql
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(o.order_id) >= 2;
```

Insight: Customers with loyalty or strong purchase interest.

🧾 5. Revenue by Genre

```sql
SELECT b.genre, SUM(o.total_amount) AS revenue
FROM orders o
JOIN books b ON o.book_id = b.book_id
GROUP BY b.genre;
```

Insight: Helps understand which genres generate the most revenue.

💰 6. Highest Spending Customer

```sql
SELECT 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;
```

Insight: Identifies the top-value customer.

📅 7. Orders in November 2023

```sql
SELECT * FROM Orders
WHERE order_date BETWEEN '2023-11-01' AND '2023-11-30'
ORDER BY order_date ASC;
```

Insight: Useful for seasonal demand tracking.

💴8. Cities With High Spending Customers

```sql
SELECT c.city, SUM(o.total_amount) AS city_spend
FROM customers c
JOIN orders o ON o.customer_id = c.customer_id
GROUP BY c.city
HAVING SUM(o.total_amount) > 30
ORDER BY city_spend DESC;
```

Insight: Identify regional demand for marketing strategy.

🖊9. Total Quantity Sold by Author

```sql
SELECT b.author, SUM(o.quantity) AS total_quantity
FROM orders o
JOIN books b ON b.book_id = o.book_id
GROUP BY b.author
ORDER BY total_quantity DESC;
```

Insight: See which authors are best sellers.

📦10. Remaining Stock After Sales

```sql
SELECT b.title, b.stock, COALESCE(SUM(o.quantity), 0) AS order_quantity,
       b.stock - COALESCE(SUM(o.quantity), 0) AS remaining_stock
FROM books b
LEFT JOIN orders o ON b.book_id = o.book_id
GROUP BY b.book_id, b.title, b.stock
ORDER BY remaining_stock ASC;
```

Insight: Helps manage inventory and restocking needs.

11. List All Customers From Canada

```sql
SELECT * FROM Customers WHERE country = 'Canada';
```

Insight: Analyze region-specific customer base.

🧳 12. Books Published After 1950

```sql
SELECT * FROM Books WHERE published_year > 1950 ORDER BY published_year ASC;
```

Insight: Discover newer books in the collection.

🧮 13. Average Price of Fantasy Books

```sql
SELECT AVG(price) AS avg_price FROM Books WHERE genre = 'Fantasy';
```

Insight: Helps price competitively in that genre.

🕐 14. Orders with Quantity > 1

```sql
SELECT * FROM Orders WHERE quantity > 1;
```

Insight: Indicates bulk or high-interest purchases.

💳 15. Orders with Amount > ₹20

```sql
SELECT * FROM Orders WHERE total_amount > 20;
```

Insight: Find higher-value transactions.

---

16. Book With Lowest Stock

```sql
SELECT * FROM Books ORDER BY stock ASC LIMIT 1;
```

Insight: Suggests which book may soon be out of stock.

📖17. Available Genres

```sql
SELECT DISTINCT genre FROM Books;
```

Insight: Gives an overview of all genres offered.

📚18. Total Available Stock

```sql
SELECT SUM(stock) AS total_stock FROM Books;
```

Insight: Gives a sense of inventory scale.

🔍19. Total Quantity Sold by Each Book

```sql
SELECT b.title, SUM(o.quantity) AS quantity_sold
FROM Orders o
JOIN Books b ON o.book_id = b.book_id
GROUP BY b.title
ORDER BY quantity_sold DESC;
```

Insight: Reveals best-performing books.

🔷20. Top Countries by Customer Count

```sql
SELECT country, COUNT(*) AS customer_count FROM Customers GROUP BY country
ORDER BY customer_count DESC;
```

Insight: Understand global customer distribution.

This insights report will be included in the project folder and linked via `README.md` for easy access.