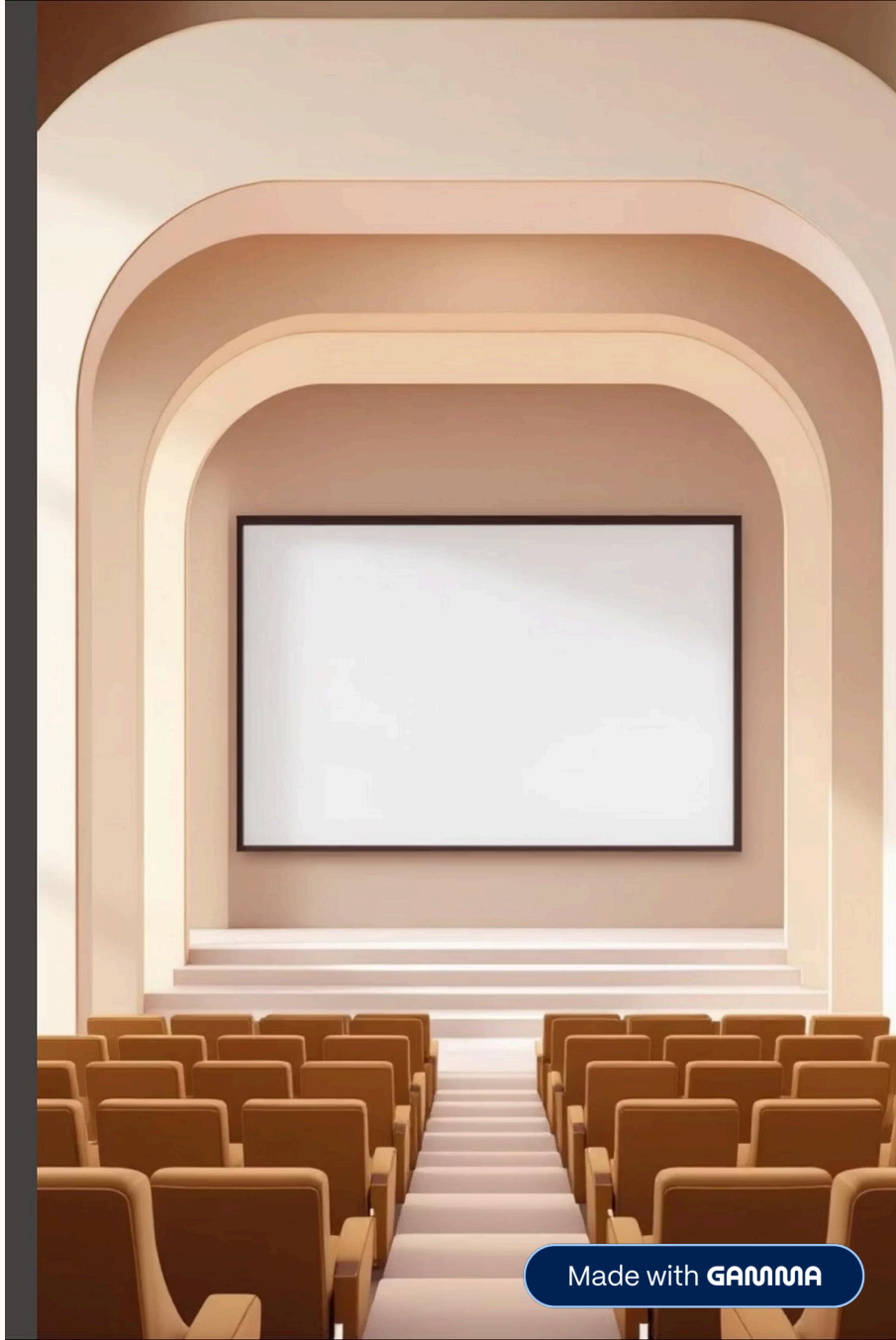
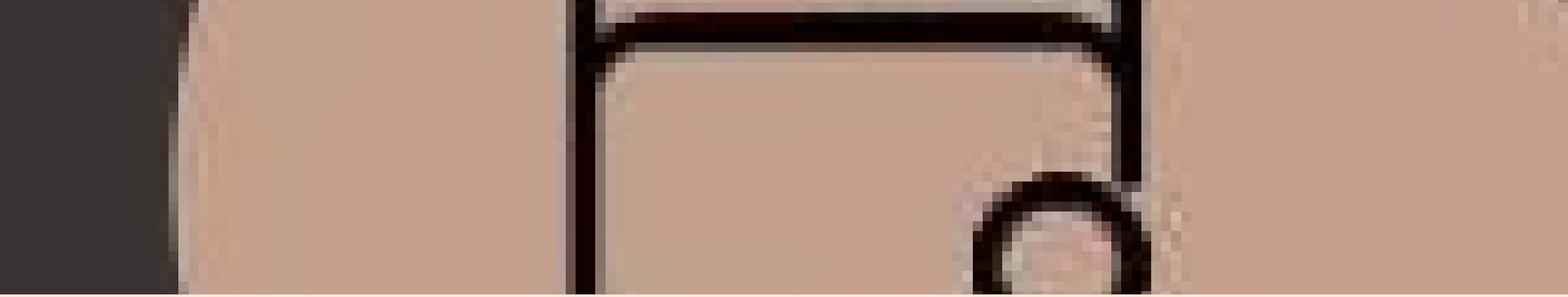
Movie Booking SQL Project

Analyzing and managing movie booking data through SQL queries to understand customer trends, payment statuses, and show performance.





Project Objective

Data Management

Analyze comprehensive movie booking data using SQL queries

Customer Insights

Understand booking trends and behavior patterns

Payment Tracking

Monitor payment statuses and transaction trends

Show Performance

Evaluate movie and show performance metrics

Dataset Overview

Data Source

Dataset: movie_booking_basic_sql.xlsx

Tables: movie_booking, booking, payments, customers, shows, theatres

Key Columns: MovieID, ShowID, CustomerID, PaymentID, BookingDate, TicketPrice

Data Coverage

- Customer information and profiles
- Movie details and metadata
- Booking records and history
- Payment transactions
- Show schedules and theatre data

SQL Operations Performed

01

SELECT Queries

Filtering and sorting movie and booking data for targeted analysis

02

GROUP BY & HAVING

Aggregation and analysis of grouped data sets

03

JOIN Operations

INNER, LEFT, and RIGHT joins for combining related tables

04

Advanced Retrieval

DISTINCT, LIMIT, and OFFSET for sophisticated data extraction

05

DELETE Queries

Cleaning outdated or invalid data from the database

Key Insights Discovered

Popular Content

Most popular genres and top-performing movies identified through booking frequency

Payment Trends

Payment trends and pending transactions tracked for financial insights

Customer Behavior

Customers with multiple bookings analyzed to understand loyalty patterns

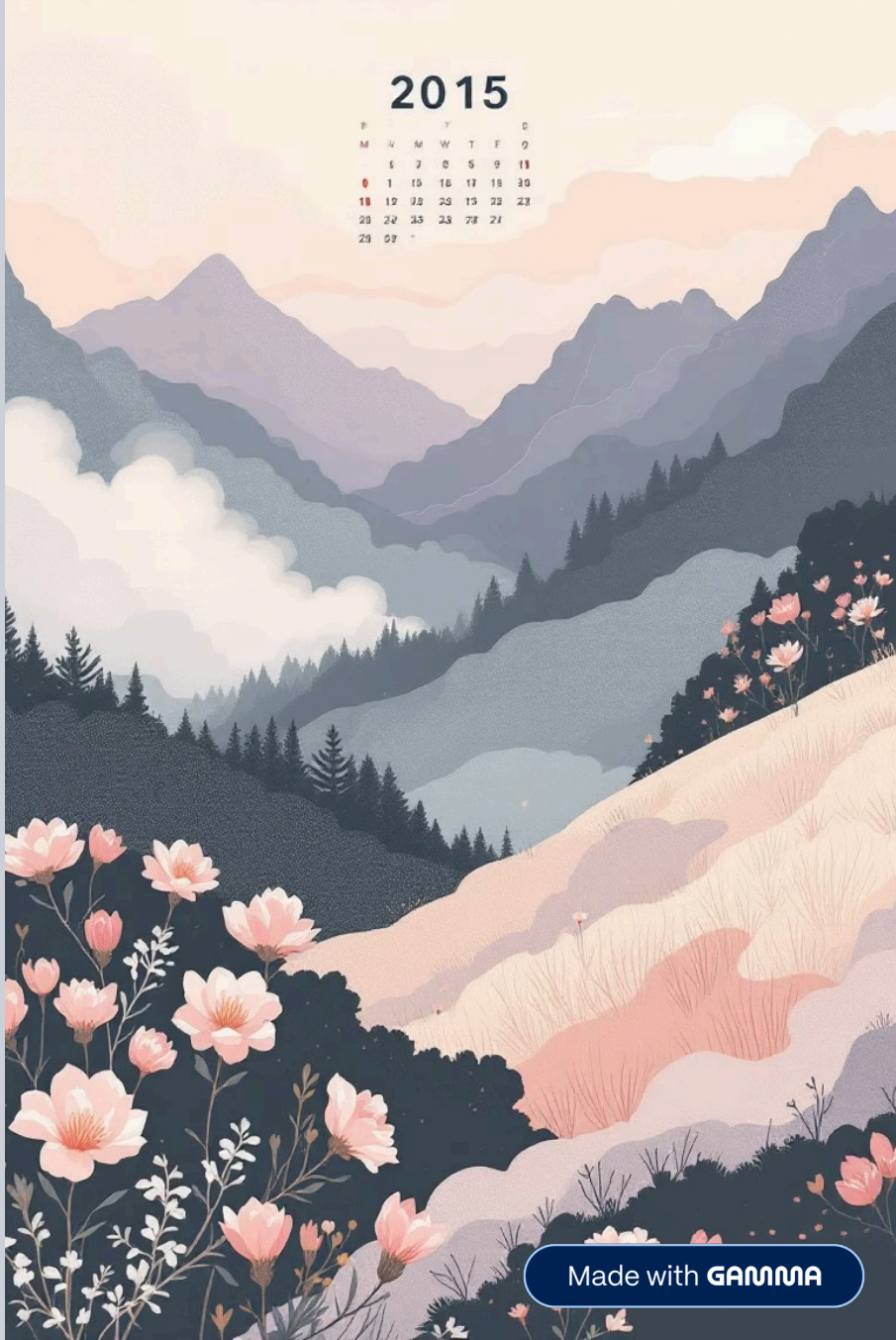
Pricing Strategy

Ticket price ranges analyzed for better pricing and revenue optimization

Sample Query: Date Range Filtering

```
SELECT * FROM movie_booking
WHERE ReleaseDate BETWEEN '2024-01-01' AND '2024-12-31';
```

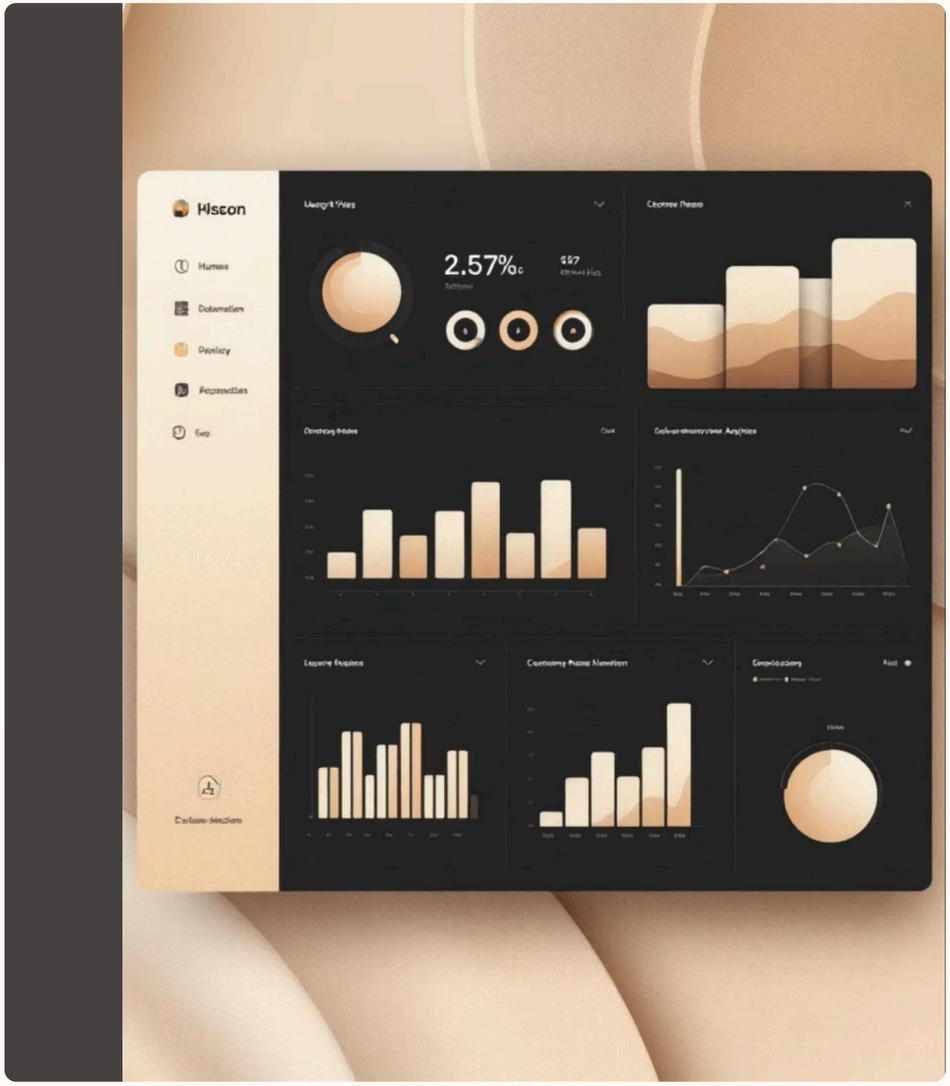
> This query retrieves all movies released in 2024, enabling year-specific analysis of new releases and their performance.



Sample Query: Customer Analysis

```
SELECT CustomerID, COUNT(*)
FROM movie_booking_booking
GROUP BY CustomerID
HAVING COUNT(*) > 2;
```

Identifies frequent customers with more than two bookings, helping target loyalty programs and understand repeat customer behavior.



Sample Query: Multi-Table JOIN

```
SELECT s.ShowID, m.Title, t.Name
FROM movie_booking_shows s
INNER JOIN movie_booking m ON s.MovieID = m.MovieID
INNER JOIN movie_booking_theatre t ON s.TheatreID = t.TheatreID;
```

Combines show, movie, and theatre data to create comprehensive reports linking movies to their screening locations.

Business Impact

Data Discovery

Uncover hidden patterns in booking behavior

Strategic Insights

Generate actionable business intelligence

Performance Growth

Optimize operations and revenue

Conclusion

1

Database Mastery

Effective use of SQL queries for complex data operations and analysis

2

Pattern Recognition

Deep analysis of booking patterns and customer behavior trends

3

Business Intelligence

Valuable insights for improving business performance and decision-making

The Movie Booking SQL Project demonstrates the power of SQL for data management, reporting, and insights generation.