# Hotel Booking And Revenue Insights using SQL
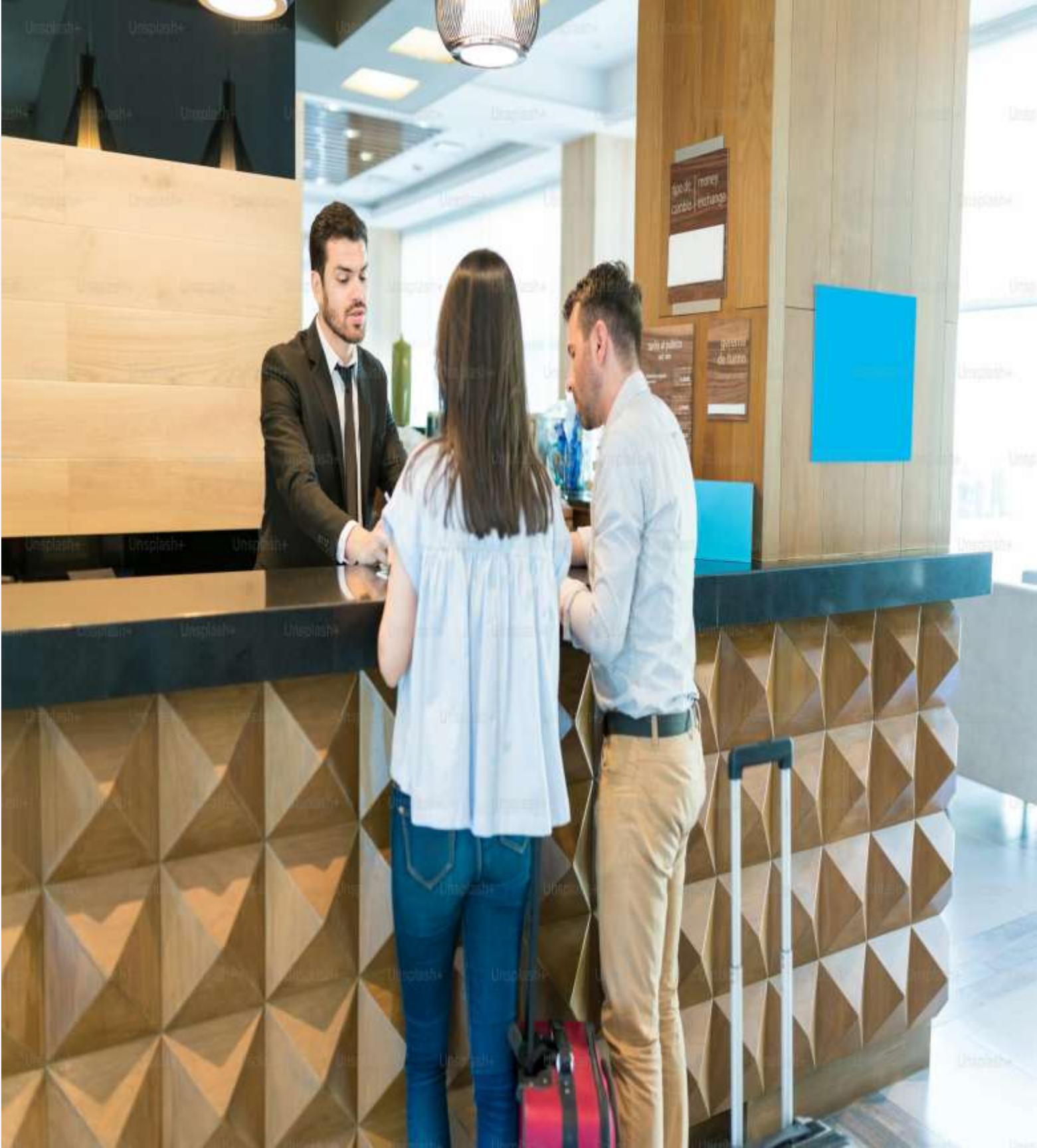
Presented by: Prathamesh Sonawane

Course - DSDA

Introduction

A smart, centralized solution to manage **customers, rooms, bookings, and reviews** seamlessly.Empowers hotels to analyze **revenue, occupancy, and guest satisfaction** efficiently.Delivers powerful insights using **advanced SQL queries for data-driven decision-making**. Provides insights using SQL queries for better decision-making.

- **DDL Commands** → CREATE Database & Tables (Customers, Rooms, Bookings, Reviews)
- **DML Commands** → SELECT, INSERT for data operations
- **Clauses** → WHERE, GROUP BY, ORDER BY, HAVING
- **Aggregate Functions** → SUM, COUNT, AVG, MIN, MAX, ROUND
- **Joins** → INNER JOIN, LEFT JOIN for relational queries
- **Date Functions** → YEAR(), MONTH(), DATEDIFF() for time-based analysis
- **Other Functions** → DISTINCT, LIMIT for unique & top records
- **Views** → BookingSummary (monthly bookings & revenue)
- **Stored Procedures** → GetBookingsByYear (automated yearly analysis)
- **Business Insights** → Revenue Analysis, Occupancy Trends, Customer Insights

What is Hotel Booking and Revenue Insights ?

1. **Booking Pattern Analysis**

   Understand daily, monthly, and seasonal booking trends to optimize occupancy.

2. **Room Performance Evaluation**

   identify popular room types and peak booking seasons for better resource allocation.

3. **Revenue Trend Insights**

   Track total, monthly, and yearly revenue to support strategic pricing decisions.
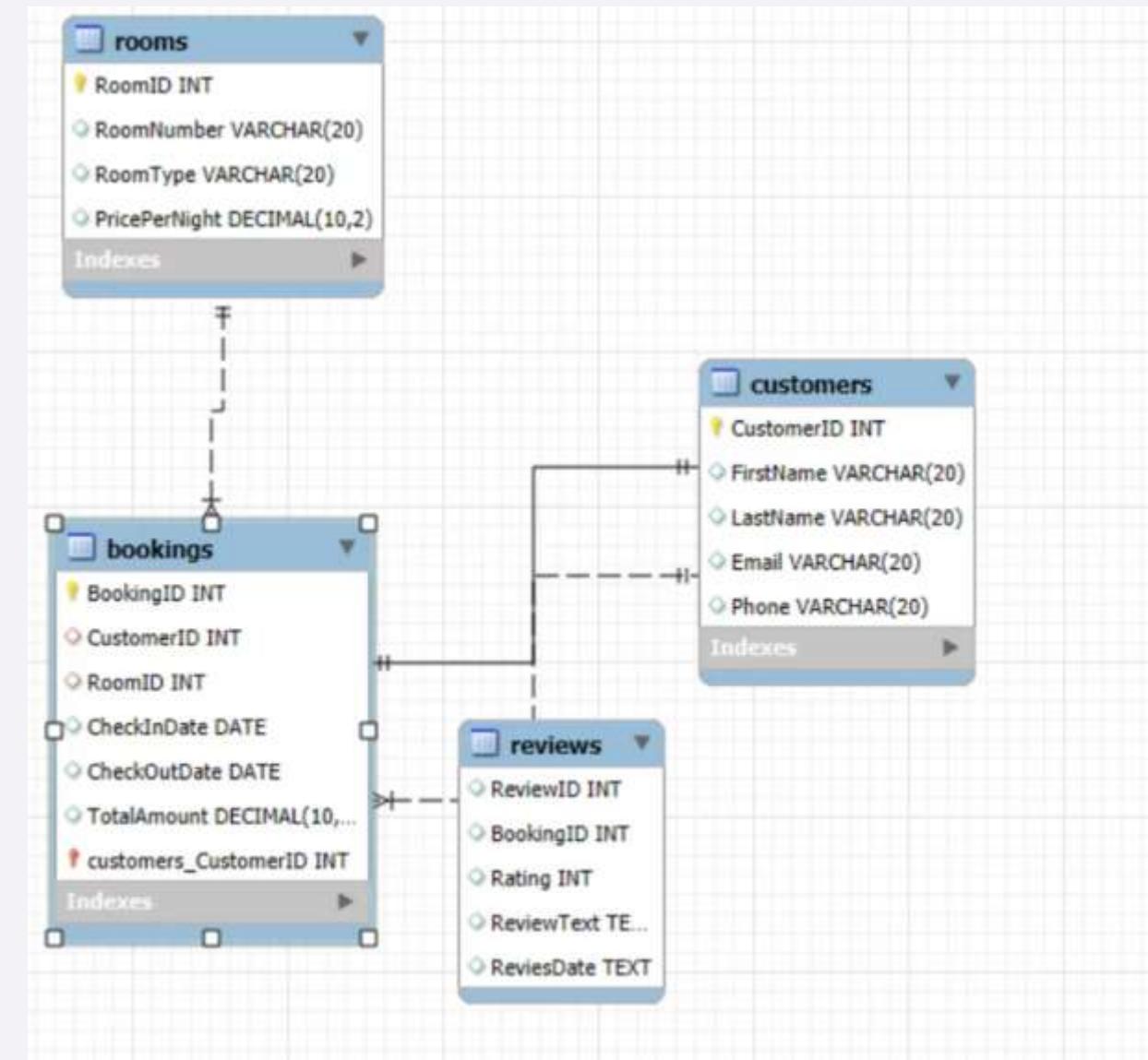
4. **Guest Behavior Understanding**

   Analyze guest frequency, average stay duration, and cancellation rates to improve satisfaction.

The Relational Foundation: Entity-Relationship Model

ER Diagram And Database Schema

The architecture of the system is built around four primary entities, connected by crucial one-to-many relationships that ensure data integrity and traceability.

Database Schema and Data Integrity

We have successfully defined four normalized tables, establishing clear primary and foreign key constraints to enforce referential integrity across the system.

Customers Table

Stores personal details of guests (CustomerID, FirstName ,LastName , Email , Phone)

Rooms Table

Catalog of all physical rooms (RoomID , RoomNumber , RoomType , PricePerNight)

Bookings Table

Transaction records (BookingID , CustomerID , RoomID , CheckInDate , CheckOutDate , TotalAmount)

Reviews Table

Guest feedback and ratings.(ReviewID , BookingID , Rating , ReviewText , ReviesDate)

Hotel Overview

1.All Over Booking List

```
select * from Customers;
```

2.Total Rooms in the Hotel

```
SELECT COUNT(Distinct RoomNumber) AS TotalRooms FROM Rooms;
```

BookingID	CustomerID	RoomID	CheckInDate	CheckOutDate	TotalAmount
1	1036	249	2025-05-20	2025-06-01	53364.00
2	326	68	2025-01-28	2025-02-03	26316.00
3	207	365	2025-07-07	2025-07-10	14391.00
5	1287	334	2024-09-30	2024-10-06	15816.00
7	1087	359	2025-03-18	2025-03-21	14139.00
11	360	5	2025-02-20	2025-02-21	4364.00
14	3215	335	2024-12-17	2024-12-21	14392.00
15	781	417	2024-10-30	2024-11-01	3472.00
16	522	301	2024-10-29	2024-11-01	9738.00
22	2458	15	2024-10-03	2024-10-14	34727.00
24	630	75	2024-10-16	2024-10-18	5240.00
28	1661	461	2024-10-14	2024-10-17	8817.00
34	351	358	2025-01-24	2025-01-30	12144.00

Result Grid | Filter Rows:

TotalRooms
40

3. Available Room Categories

```
53
54     --  3.  Available Room Categories
55 •   select Distinct roomtype from Rooms;
```

Result Grid	Filter Rows:	Export:	Wrap Cel

	roomtype
▶	Double
	Single
	Suite

Room Analysis

4. Popular Room Type Analysis by RoomType

```
57     -- 4.    Room Price Pernight by RoomType
58 •   SELECT RoomType, MIN(PricePerNight) AS MinPrice, MAX(PricePerNight) AS MaxPrice, AVG(PricePerNight) AS AvgPrice
59     FROM Rooms
60     GROUP BY RoomType;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:

	RoomType	MinPrice	MaxPrice	AvgPrice
▶	Double	1017.00	4985.00	3124.124138
	Single	1036.00	4982.00	2886.751381
	Suite	1001.00	4972.00	2983.316092

5.Room Booking Count by RoomNumber

```
62
63      -- 5.Room Booking Count by RoomNumber (more time)
64 •    SELECT r.RoomNumber, COUNT(b.BookingID) AS BookingCount
65      FROM Rooms r
66      LEFT JOIN Bookings b ON r.RoomID = b.RoomID
67      GROUP BY r.RoomNumber
68      ORDER BY BookingCount desc;
```

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

RoomNumber	BookingCount
1	231
17	199
5	176
33	171
2	171
26	169
10	167
38	165
25	164
39	152
11	150
19	150
4	149
30	138
23	137

6. Popular Room Type Analysis By RoomType

```
69
70      -- 6.  Popular Room Type Analysis  by RoomType  ( more time )
71 •    SELECT r.RoomType, COUNT(b.BookingID) AS BookingCount
72      FROM Rooms r
73      JOIN Bookings b ON r.RoomID = b.RoomID
74      GROUP BY r.RoomType
75      ORDER BY BookingCount DESC;
```

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

RoomType	BookingCount
Single	1796
Suite	1738
Double	1505

Booking Trends

7.Month Booking Trends By Year

8. Booking Season

```
77        --  7. Monthly Booking Trends by Year
78  ●     SELECT YEAR(CheckInDate) AS Year, MONTH(CheckInDate) AS Month, COUNT(*) AS BookingCount
79        FROM Bookings
80        GROUP BY Year, Month
```

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

Year	Month	BookingCount
2024	9	85
2024	10	427
2024	11	403
2024	12	436
2025	1	447
2025	2	373
2025	3	413
2025	4	395
2025	5	422
2025	6	445
2025	7	439
2025	8	407
2025	9	347

```
200       --  25.Peak Booking Season (year + month)
201  ●    SELECT YEAR(CheckInDate) AS Year, MONTH(CheckInDate) AS Month, COUNT(*) AS Bookings
202       FROM Bookings
203       GROUP BY Year, Month
204       ORDER BY Year, Month desc;
```

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

Year	Month	Bookings
2024	12	436
2024	11	403
2024	10	427
2024	9	85
2025	9	347
2025	8	407
2025	7	439
2025	6	445
2025	5	422
2025	4	395
2025	3	413
2025	2	373
2025	1	447

Guest Analysis

9.Average Stay Duration

10. Top Guest by Booking frequency

```
102    -- 11.  Top Guests by Booking Frequency  ( More Time Booked guest)
103 •  SELECT c.CustomerID, c.FirstName, c.LastName, COUNT(b.BookingID) AS BookingCount
104    FROM Customers c
105    JOIN Bookings b ON c.CustomerID = b.CustomerID
106    GROUP BY c.CustomerID
107    ORDER BY BookingCount DESC
108    LIMIT 10;
```

CustomerID	FirstName	LastName	BookingCount
2297	Suzanne	Dorsey	9
2843	Thomas	Blair	8
52	Thomas	Freeman	8
2594	Crystal	Brock	8
1900	Monica	Henderson	8
4526	David	Moon	8
4055	William	Williams	8
169	Todd	Wilson	7
1760	Dawn	James	7
162	Tara	Ray	7

```
96    -- 10.  Average Stay Duration
97 •  SELECT AVG(DATEDIFF(CheckOutDate, CheckInDate)) AS AvgStayDuration
98    FROM Bookings;
99
```

AvgStayDuration
7.4628

Revenue Trends

11.Total Revenue

```
114
115    -- 13.  Total Revenue Generated  (income)
116  ●  SELECT SUM(TotalAmount) AS TotalRevenue
117    FROM Bookings;
118
```

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

TotalRevenue
▶ 112141629.00

12.Annual Revenue Analysis

```
20    -- 14.  Annual Revenue Analysis ( yearly income)
21  ●  SELECT YEAR(CheckInDate) AS Year,
22         SUM(TotalAmount) AS YearlyRevenue
23    FROM Bookings
24    GROUP BY Year
25    ORDER BY Year desc;
```

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

Year	YearlyRevenue
2025	81639927.00
2024	30501702.00

13.Monthly Revenue Trends

```
126
127        -- 15.Monthly Revenue  Trends (per year + month)
128 •     SELECT YEAR(CheckInDate) AS Year,
129             MONTH(CheckInDate) AS Month,
130             SUM(TotalAmount) AS MonthlyRevenue
131       FROM Bookings
132       GROUP BY Year, Month
133       ORDER BY Year, Month desc;
```

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

Year	Month	MonthlyRevenue
2024	12	9846608.00
2024	11	8948027.00
2024	10	9488401.00
2024	9	2218666.00
2025	9	8179228.00
2025	8	8870825.00
2025	7	9772499.00
2025	6	10009335.00
2025	5	9261861.00
2025	4	8577801.00
2025	3	9140958.00
2025	2	8137498.00
2025	1	9689922.00

Revenue by Room And Top Customers

14.Roomtype Revenue

```
142
143      -- 17    RoomType Revenue  (income)
144 •    SELECT r.RoomType, SUM(b.TotalAmount) AS Revenue
145      FROM Bookings b
146      JOIN Rooms r ON b.RoomID = r.RoomID
147      GROUP BY r.RoomType;
```

RoomType	Revenue
Double	35144715.00
Single	38959695.00
Suite	38037219.00

15. Top 5 Spending Customer

```
154      -- 19. Top 5 Spending Customers (paying)
155 •    SELECT c.CustomerID, c.FirstName, c.LastName, SUM(b.TotalAmount) AS TotalSpent
156      FROM Customers c
157      JOIN Bookings b ON c.CustomerID = b.CustomerID
158      GROUP BY c.CustomerID
159      ORDER BY TotalSpent DESC
160      LIMIT 5;
161
```

CustomerID	FirstName	LastName	TotalSpent
4131	Robert	Beck	264945.00
3235	Daniel	Jones	252671.00
2843	Thomas	Blair	242918.00
4526	David	Moon	242715.00
3681	Cory	Reed	236398.00

Ratings And Reviews

16.Total Guest Rating

```
162
163        -- 20. Total Guest Ratings
164  •     SELECT SUM(Rating) AS TotalRating
165        FROM Reviews;
```

Result Grid | Filter Rows: | Export: | Wrap Ce

	TotalRating
▶	11818

17.RoomType Total Rating

```
166
167        -- 21. RoomType ToTal Rating
168  •     SELECT r.RoomType, SUM(rv.Rating) AS TotalRating
169        FROM Reviews rv
170        JOIN Bookings b ON rv.BookingID = b.BookingID
171        JOIN Rooms r ON b.RoomID = r.RoomID
172        GROUP BY r.RoomType;
173
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: IA
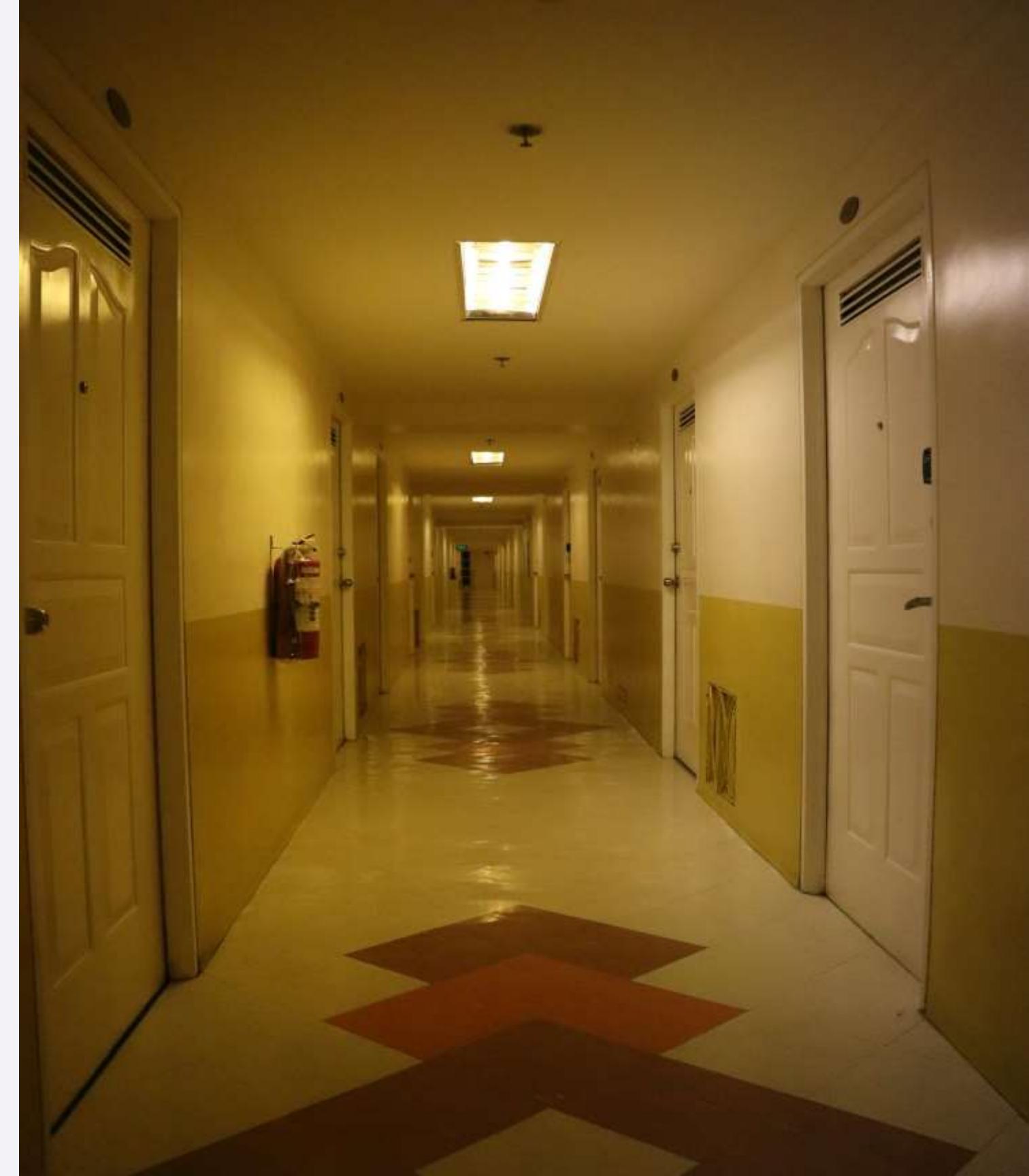
	RoomType	TotalRating
▶	Suite	1816
	Single	1659
	Double	1458

18. Top Rated RoomTypes

```
179        -- 23.Top Rated RoomTypes
180  •     SELECT r.RoomType, ROUND(AVG(rv.Rating),2) AS AvgRating
181        FROM Reviews rv
182        JOIN Bookings b ON rv.BookingID = b.BookingID
183        JOIN Rooms r ON b.RoomID = r.RoomID
184        GROUP BY r.RoomType
185        ORDER BY AvgRating DESC;
186
```

	RoomType	AvgRating
▶	Suite	2.97
	Double	2.95
	Single	2.92

View + Stored Procedure

19. views

20. Stored Procedure


```
233     -- 31. views   (query repeat type karycha kam nahi )
234 •   CREATE VIEW BookingSummary AS
235     SELECT YEAR(CheckInDate) AS Year, MONTH(CheckInDate) AS Month,
236             COUNT(*) AS BookingCount,
237             SUM(TotalAmount) AS Revenue
238     FROM Bookings
239     GROUP BY Year, Month;
240 •   select *from BookingSummary;
```

Year	Month	BookingCount	Revenue
2025	5	422	9261861.00
2025	1	447	9689922.00
2025	7	439	9772499.00
2024	9	85	2218666.00
2025	3	413	9140958.00
2025	2	373	8137498.00
2024	12	436	9846608.00
2024	10	427	9488401.00
2025	6	445	10009335.00
2025	9	347	8179228.00
2025	4	395	8577801.00
2024	11	403	8948027.00
2025	8	407	8870825.00

```
242     -- 32.Stored Procedure
243     DELIMITER //
244 •   CREATE PROCEDURE GetBookingsByYear(IN inputYear INT)
245     BEGIN
246         SELECT RoomType, COUNT(*) AS BookingCount
247         FROM Bookings b
248         JOIN Rooms r ON b.RoomID = r.RoomID
249         WHERE YEAR(CheckInDate) = inputYear
250         GROUP BY RoomType;
251     END //
252     DELIMITER ;
253 •   CALL GetBookingsByYear(2025);
254 •   CALL GetBookingsByYear(2024);
```

RoomType	BookingCount
Double	1083
Single	1326
Suite	1279

Key Insights from SQL Analytics

•**Peak Season:** Maximum bookings recorded in festive months & weekends.

•**Revenue Drivers:** Deluxe Rooms generated the highest share of revenue.

•**Customer Insights:** Majority of guests were repeat customers from top metro cities.

•**Occupancy Trends:** Average occupancy rate consistently above 70%, with spikes during holidays.

•**Decision-Making:** Data highlights areas for optimizing pricing and improving guest satisfaction.

Conclusion

This project highlights how a smart hotel management system transforms data into actionable intelligence, enabling hotels to operate with precision and efficiency.By leveraging SQL-based analytics, hotels can not only track booking trends and revenue streams but also anticipate guest needs, improve service quality, and achieve sustainable growth.Ultimately, such a system empowers the hospitality industry to deliver superior guest experiences while maximizing operational performance and profitability.

THANK YOU