

# **DATA WAREHOUSING PROJECT**

## **Mastering Data Management for Hotel Booking**

Sai Ramakanth Kodi (U06591564), Sai Nithin Bharadwaj Govind (U56893364)

### **Executive Summary**

"DataStay," a revolutionary hotel booking platform, is set to redefine the hospitality industry through the power of data warehousing. This innovative system leverages the latest in data warehousing to consolidate and analyze extensive datasets, revolutionizing hotel booking, management, and the overall guest experience. It empowers hoteliers with data-driven insights for maximizing revenue and streamlining operations. DataStay prioritizes the needs and preferences of each guest by offering personalized services, customized promotions, and an efficient booking interface. Additionally, it enhances inventory control, enabling hotels to optimize occupancy and adapt pricing strategies dynamically. The platform ensures real-time access to room availability and rates, while implementing advanced security measures to prevent fraud. DataStay's adaptability, effortless integration, and contributions to increased profitability, operational excellence, market leadership, and enhanced guest loyalty firmly establish it as an innovative leader in transforming hotel booking systems.

### **Problem Statement**

The current landscape of hotel management reveals a critical void: the absence of advanced data warehousing solutions, which has resulted in notable operational inefficiencies and untapped potential for revenue enhancement. Traditional systems fall short in their ability to aggregate and interpret large-scale data, leading to decisions that don't fully leverage available information. The hospitality sector faces significant hurdles in delivering personalized experiences to guests, hindered by the lack of deep, data-driven insights that only a comprehensive data warehousing approach can offer. Moreover, existing inventory management methods struggle to adapt to market demand variations promptly, causing resource underutilization and lost revenue opportunities. The absence of instant access to up-to-date availability and pricing information further challenges the transparency of the booking experience. Additionally, the industry's ongoing battle with fraud calls for a more advanced security strategy. This context highlights the urgent necessity for an innovative data warehousing solution like DataStay, which promises to

usher the hotel industry into a new age of operational effectiveness, revenue growth, and customized guest services.

## **Literature Review:**

The primary objective of our project was to design and develop a specialized data warehouse model tailored for constructing a comprehensive hotel booking database. This endeavor necessitated the integration of multiple key tables, including Bookings, Hotel, Location, and Room, utilizing a star join technique as the primary method for merging this data. Our literature review delves into the methodologies employed in existing warehousing projects, underscoring the critical importance of meticulous table linkage to enhance the overall effectiveness and efficiency of data management within the hospitality sector. The research questions we aimed to address through this project include:

1. Which properties boast the largest number of rooms available?
2. Which hotels are distinguished by the highest average customer ratings?
3. What is the cumulative number of bookings made by the state over the period from 2015 to 2022?
4. How has the total expenditure on hotel bookings evolved from 2015 through 2022?
5. What category of rooms contributes the most to hotel revenue?
6. In which state do hotels, on average, command the highest booking prices?

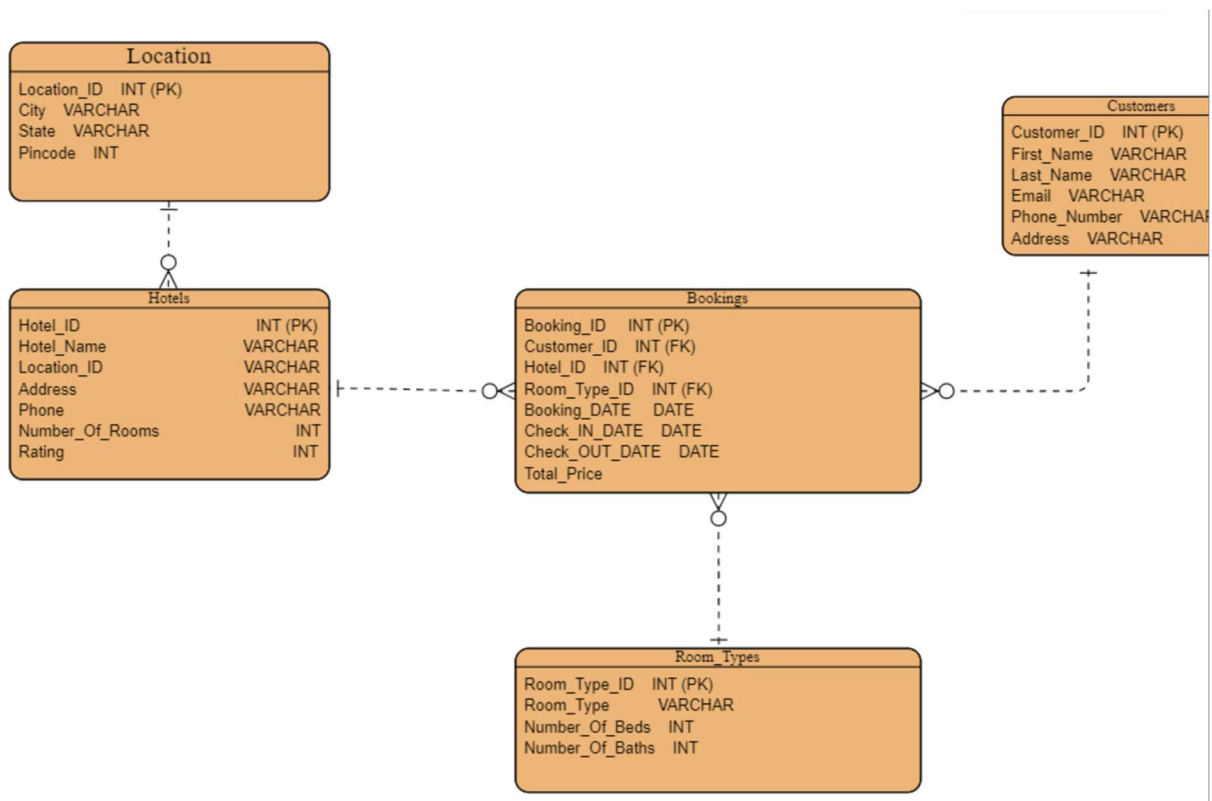
## **Data collection and Preparation:**

Data for this project was meticulously collected from a variety of platforms, including Kaggle, and other hotel booking datasets, followed by a thorough process of extraction, cleansing, preprocessing, and analysis to tackle the issue at hand. This involved the careful formatting and importing of data into Oracle SQL, whereupon tables were constructed and a series of queries executed to derive meaningful insights. The datasets utilized in our study include:

- **Hotels Dataset:** Contains detailed records of hotels, encompassing hotel ID, name, ratings, address, number of available rooms, and location ID.
- **Location Data:** Provides insights into the geographic distribution of hotels across India, featuring data points such as location ID, state name, city, and pin code.

- **Customer Data:** Encompasses a wealth of information on individual customers, including customer ID, first and last names, email addresses, phone numbers, and payment details.
- **Rooms Dataset:** Includes specifics on the various types of rooms available, identified by room ID and room type.

## Schema:



Below are the queries that are used to create the above tables in the oracle database:

### Tables & Structures:

#### 1. Location Table:

```

CREATE TABLE Location (
  location_id INT PRIMARY KEY,
  city VARCHAR(255),
  State VARCHAR(255),

```

```

pincode INT UNIQUE
);

```

	LOCATION_ID	CITY	STATE	PINCODE
1	1699	BULANDSHAHR	Uttar Pradesh	203205
2	1700	BULANDSHAHR	Uttar Pradesh	203206
3	1701	GAUTAM BUDDHA NAGAR	Uttar Pradesh	203207
4	1702	GAUTAM BUDDHA NAGAR	Uttar Pradesh	203209
5	1703	BULANDSHAHR	Uttar Pradesh	203389
6	1704	BULANDSHAHR	Uttar Pradesh	203390
7	1705	BULANDSHAHR	Uttar Pradesh	203391
8	1706	BULANDSHAHR	Uttar Pradesh	203392
9	1707	BULANDSHAHR	Uttar Pradesh	203393
10	1708	BULANDSHAHR	Uttar Pradesh	203394

## 2. Hotels table:

```

CREATE TABLE Hotels (
hotel_id INT PRIMARY KEY,
hotel_name VARCHAR(255),
location_id INT,
address VARCHAR(255),
phone VARCHAR(20),
number_of_rooms INT,
Rating INT,
FOREIGN KEY (location_id) REFERENCES Location(location_id)
);

```

HOTEL_ID	HOTEL_NAME	LOCATION_ID	ADDRESS	PHONE	NUMBER_OF_ROOMS	RATING
2	87 Subh Laxmi P'aying Guest House	283035/13	Khari Kuan ,	(null)	297	1
3	88 Lotus Beach Resort-Murud Hornai	8861	At Post - Murud,Taluka - Dapoli, district - ...	(null)	72	3
4	89 HOTEL RUDRAKSH INN	17113	Beltola - Basistha Rd , Maidamgaon, Krishnapur	(null)	78	4
5	90 Hotel Bhagsu View	14795	Bhagsunag Dharamsala Kangra ,	(null)	114	3
6	91 The Hidden Valley Camps and Resorts	3989	Pangot , Nainital	(null)	153	2
7	92 Meraki Inn	18199	flat no - 201202 2floor, NV villa, 17-A, Sil...	(null)	88	2
8	93 KAMLA HOTEL	11277	Jalasar road, Deoghar ,	(null)	75	2
9	94 Hotel Holiday Home	850	Lohna Hills , Opposite Hotel TeaBud	(null)	222	3
10	95 Tex Woods Resort	8573	Tex woods sheegekhan estate, next to muthodi...	(null)	133	5

## 3. Customers Table:

```

CREATE TABLE Customers (
customer_id INT PRIMARY KEY,
first_name VARCHAR(255),
last_name VARCHAR(255),
email VARCHAR(255),
phone_number VARCHAR(20),
address VARCHAR(255)
);

```

	CUSTOMER_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	ADDRESS
1	121	Alexander	Bass	Bass_Alexander65@zoho.com	378-238-0167	(null)
2	122	Leslie	Wilson	LeslieWilson@comcast.net	954-271-5882	(null)
3	123	Emily	Pennington	EPennington@protonmail.com	468-989-6508	(null)
4	124	Mary	Mendez	MaryMendez@protonmail.com	939-016-1514	(null)
5	125	Rebecca	Price	Rebecca.Price@zoho.com	805-825-8710	(null)
6	126	Diana	Mcdowell DDS	DDDS@comcast.net	552-624-5025	(null)
7	127	Jennifer	Terry	Jennifer_Terry58@outlook.com	831-324-4638	(null)
8	128	Andrew	Barnett	Andrew.B@yahoo.com	387-017-4651	(null)
9	129	Evelyn	Day	Day.Evelyn@outlook.com	909-666-8383	(null)
10	130	Natalie	Anderson	Anderson_Natalie@gmail.com	109-008-2223	(null)

#### 4.Rooms Table:

```
CREATE TABLE Room_Types (
    Room_Type_id INT PRIMARY KEY,
    Room_Type VARCHAR(50),
    Number_of_Beds INT,
    Number_of_Baths INT
);
```

	ROOM_TYPE_ID	ROOM_TYPE	NUMBER_OF_BEDS	NUMBER_OF_BATHS
1	1	Non A/C Standard Room	1	1
2	2	A/C Standard Room	2	1
3	3	Deluxe Room	2	2
4	4	Suite	3	2
5	5	Luxury Suite	4	3

#### 5. Bookings Table:

```
CREATE TABLE Bookings (
    booking_id INT PRIMARY KEY,
    customer_id INT,
    hotel_id INT,
    room_type_id INT,
    booking_date DATE,
    check_in_date DATE,
    check_out_date DATE,
    total_price DECIMAL(10,2),

    FOREIGN KEY (customer_id) REFERENCES Customers(customer_id),
    FOREIGN KEY (hotel_id) REFERENCES Hotels(hotel_id),
    FOREIGN KEY (room_type_id) REFERENCES Room_Types(room_type_id)
);
```

	BOOKING_ID	CUSTOMER_ID	HOTEL_ID	ROOM_TYPE_ID	BOOKING_DATE	CHECK_IN_DATE	CHECK_OUT_DATE	TOTAL_PRICE
1	153	49	54	3	25-FEB-15	03-MAR-15	05-MAR-15	1108.27
2	154	109	226	4	30-JAN-18	06-FEB-18	10-FEB-18	2391.37
3	155	225	154	5	16-JAN-22	19-JAN-22	23-JAN-22	2848.79
4	156	13	226	1	14-FEB-18	14-FEB-18	25-FEB-18	1331.4
5	157	140	178	2	10-NOV-15	18-NOV-15	26-NOV-15	2029.98
6	158	78	233	3	14-JAN-15	15-JAN-15	22-JAN-15	3023.31
7	159	147	156	4	10-SEP-16	10-SEP-16	13-SEP-16	1748.39
8	160	52	155	5	05-APR-21	09-APR-21	12-APR-21	2194.96
9	161	152	9	1	22-MAY-20	31-MAY-20	10-JUN-20	1305.76
10	162	144	235	2	24-DEC-20	25-DEC-20	31-DEC-20	1688.85

We carefully specified the structure and links between tables using primary and foreign keys in our SQL database design for this project, ensuring a strong foundation for our study. This is a summary of the connections between our tables made by these keys:

- **Location Table:** Acts as a foundational table with **location\_id** serving as its primary key. This table does not reference any foreign keys.
- **Hotels Table:** Identified by **hotel\_id** as its primary key, it links to the Location Table through **location\_id** as a foreign key, establishing a geographical relationship.
- **Customers Table:** Centers around **customer\_id** as its primary key, standing independently without referencing any foreign keys.
- **Room\_Types Table:** Defined by **Room\_Type\_id** as its primary key, this table also does not have foreign keys, focusing solely on categorizing room types.
- **Bookings Table:** This table is the core of our database schema, with **booking\_id** as its primary key. It interconnects with the Customers, Hotels, and Room\_Types tables through foreign keys: **customer\_id**, **hotel\_id**, and **room\_type\_id** respectively. These foreign keys create a comprehensive linkage back to their respective primary keys in the dimension tables.

The Bookings table is designated as the fact table, central to our analysis, holding the transactional data of hotel bookings. The surrounding tables — Location, Hotels, Customers, and Room\_Types — are treated as dimension tables. They provide descriptive attributes and context to the bookings, such as customer profiles, hotel specifics, room characteristics, and location details. This arrangement, with a central fact table connected to dimension tables via foreign keys, enables efficient data aggregation, analysis, and retrieval, facilitating a deeper understanding of the dynamics within hotel bookings.

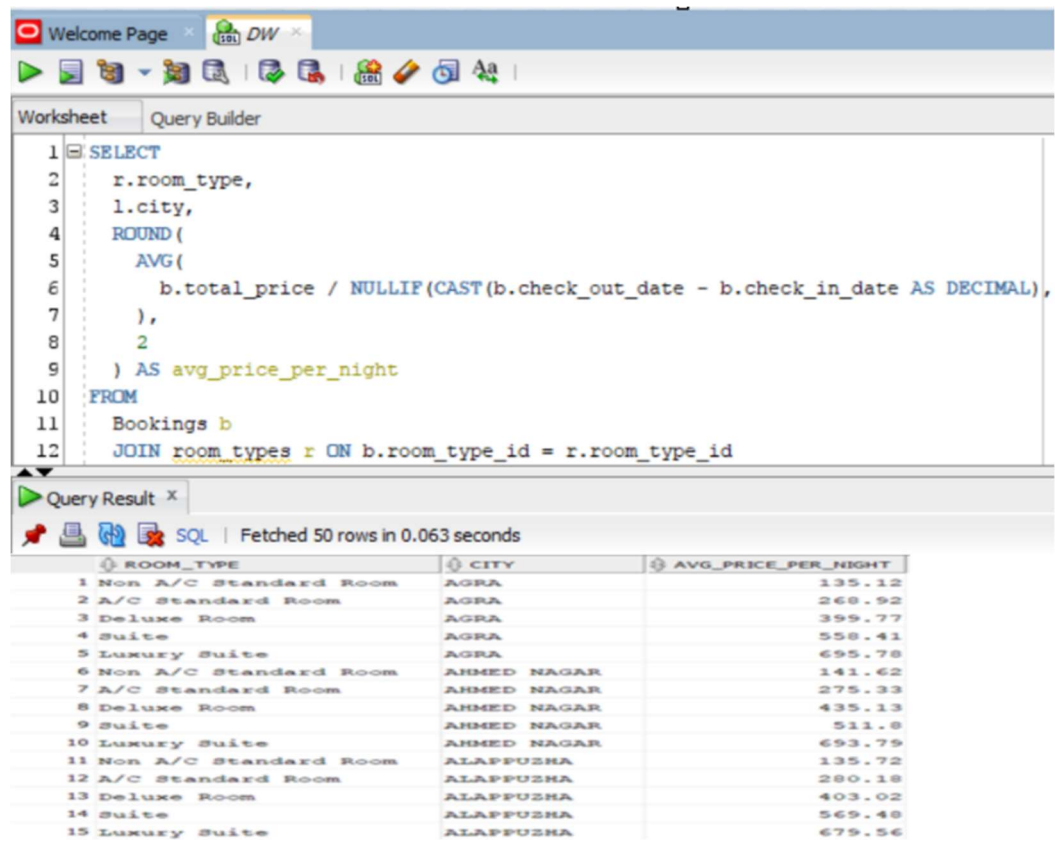
## Exploratory Data Analysis:

1.Determine the average night rate for a particular type of room in each city.

```

SELECT
    r.room_type,
    l.city,
    ROUND(
        AVG(
            b.total_price / NULLIF(CAST(b.check_out_date - b.check_in_date AS DECIMAL), 0)
        ),
        2
    ) AS avg_price_per_night
FROM
    Bookings b
    JOIN room_types r ON b.room_type_id = r.room_type_id
    JOIN Hotels h ON b.hotel_id = h.hotel_id
    JOIN Location l ON h.location_id = l.location_id
GROUP BY
    r.room_type, l.city
ORDER BY
    l.city,
    avg_price_per_night;

```



The screenshot shows a database query tool interface. The top section displays the SQL query, and the bottom section shows the query results. The query is a SELECT statement that calculates the average price per night for different room types across various cities, grouped by room type and city, and ordered by city and average price per night.

**Query Builder**

```

1 SELECT
2   r.room_type,
3   l.city,
4   ROUND(
5     AVG(
6       b.total_price / NULLIF(CAST(b.check_out_date - b.check_in_date AS DECIMAL),
7     ),
8     2
9   ) AS avg_price_per_night
10 FROM
11   Bookings b
12   JOIN room_types r ON b.room_type_id = r.room_type_id

```

**Query Result**

SQL | Fetched 50 rows in 0.063 seconds

ROOM_TYPE	CITY	AVG_PRICE_PER_NIGHT
1 Non A/C Standard Room	AGRA	135.12
2 A/C Standard Room	AGRA	268.92
3 Deluxe Room	AGRA	399.77
4 Suite	AGRA	558.41
5 Luxury Suite	AGRA	695.78
6 Non A/C Standard Room	AHMED NAGAR	141.62
7 A/C Standard Room	AHMED NAGAR	275.33
8 Deluxe Room	AHMED NAGAR	435.13
9 Suite	AHMED NAGAR	511.8
10 Luxury Suite	AHMED NAGAR	693.79
11 Non A/C Standard Room	ALAPPUZHA	135.72
12 A/C Standard Room	ALAPPUZHA	280.18
13 Deluxe Room	ALAPPUZHA	403.02
14 Suite	ALAPPUZHA	569.48
15 Luxury Suite	ALAPPUZHA	679.56

### Explanation:

This refined SQL query calculates the average nightly rate for room types across cities by joining the **Bookings**, **room\_types**, **Hotels**, and **Location** tables. It divides the total cost of each booking by the stay duration to find the average price per night, rounding to two decimal places for precision. By grouping the data by **room\_type** and **city**, the query avoids the complexities of the **CUBE** function, ensuring only relevant, non-null combinations are analyzed. It includes safeguards like **NULLIF** to prevent division by zero errors, enhancing reliability. The results are ordered by city and average price, providing a clear snapshot of pricing trends across locations. This streamlined approach ensures efficiency while maintaining the query's accuracy and relevance.

### 2.Rank according to hotel ratings in every city

```
SELECT
    h.hotel_id,
    h.hotel_name,
    h.rating,
    l.state,
    l.city,
    RANK() OVER (ORDER BY h.rating DESC) AS rating_rank
FROM
    hotels h
JOIN
    location l ON h.location_id = l.location_id
ORDER BY
    rating_rank, l.city, l.state;
```



The screenshot shows a SQL IDE with a 'Welcome Page' and 'DW' tabs. The 'Query Builder' tab is active, displaying a SQL query. Below the query, the 'Query Result' tab shows the results of the query, which is a table with 6 columns: HOTEL\_ID, HOTEL\_NAME, RATING, STATE, CITY, and RATING\_RANK. The results are sorted by RATING\_RANK in descending order.

```

1 SELECT
2     h.hotel_id,
3     h.hotel_name,
4     h.rating,
5     l.state,
6     l.city,
7     RANK() OVER (ORDER BY h.rating DESC) AS rating_rank
8 FROM
9     hotels h
10 JOIN
11     location l ON h.location_id = l.location_id
12 ORDER BY

```

HOTEL_ID	HOTEL_NAME	RATING	STATE	CITY	RATING_RANK
1	113 Honeyvale Homestay	5	Uttar Pradesh	AMBEDKAR NAGAR	1
2	121 Amrit Residency	5	Bihar	BEGUSARAI	1
3	141 Hotel Neelkanth	5	Himachal Pradesh	SHIMLA	1
4	167 Aangan Marraige Hall and Guest house	5	Himachal Pradesh	HAMIRPURHP	1
5	199 Singhi Villa Guest House	5	Uttar Pradesh	ALIGARH	1
6	272 Hotel K R Bhavan	5	Rajasthan	DAUSA	1
7	193 hotel avlokhan	5	Rajasthan	BHILWARA	1
8	213 Hotel Riddhi Inn	5	Rajasthan	DUNGARPUR	1
9	112 Hotel Royal Heritage	5	Rajasthan	ALWAR	1
10	231 Chira Meadows Eco Resort	5	Uttar Pradesh	SAHARANPUR	1
11	216 Kamat Lingapur Hotel	5	Uttar Pradesh	BASTI	1
12	206 Snow View Huts	5	Uttar Pradesh	AGRA	1
13	40 Hotel Castle View	5	Gujarat	ARAVALLI	1
14	205 Hotel Malji Ka Kamra	5	Maharashtra	MUMBAI SUBUEBAN	1
15	35 Hotel Grand Riviera	5	Maharashtra	MUMBAI SUBUEBAN	1

### Explanation:

The SQL query ranks hotels based on their ratings, combining hotel details with their locations. By joining the **hotels** table with the **location** table on **location\_id**, it pulls in each hotel's name, rating, state, and city. The **RANK()** function then orders hotels globally by their rating in descending order, assigning a rank that reflects their performance compared to all hotels in the dataset. The results are sorted to present hotels organized by their rank, ensuring a clear and concise overview of hotel ratings across locations. This approach offers an efficient way to evaluate and compare hotel ratings on a broad scale, suitable for analyses that span multiple cities or states.

### 3. Identify those customers who made the most overall reservation amount:

```

SELECT
    CustomerDetails.CUSTOMER_ID,
    CustomerDetails.FIRST_NAME,
    CustomerDetails.LAST_NAME,
    BookingInfo.BOOKING_YEAR,
    BookingInfo.TOTAL_BOOKING_AMOUNT,
    HotelLocation.CITY

```

FROM

(SELECT

c.CUSTOMER\_ID,

c.FIRST\_NAME,

c.LAST\_NAME

FROM

customers c) AS CustomerDetails

JOIN

(SELECT

b.CUSTOMER\_ID,

EXTRACT(YEAR FROM b.BOOKING\_DATE) AS BOOKING\_YEAR,

SUM(b.TOTAL\_PRICE) AS TOTAL\_BOOKING\_AMOUNT

FROM

bookings b

GROUP BY

b.CUSTOMER\_ID,

EXTRACT(YEAR FROM b.BOOKING\_DATE)) AS BookingInfo ON  
CustomerDetails.CUSTOMER\_ID = BookingInfo.CUSTOMER\_ID

JOIN

(SELECT

h.HOTEL\_ID,

l.CITY

FROM

HOTELS h

JOIN LOCATION l ON h.LOCATION\_ID = l.LOCATION\_ID) AS HotelLocation ON  
BookingInfo.HOTEL\_ID = HotelLocation.HOTEL\_ID

ORDER BY

BookingInfo.TOTAL\_BOOKING\_AMOUNT DESC;

The screenshot shows a SQL Query Builder window with a query and its results. The query is as follows:

```

2 CustomerDetails.CUSTOMER_ID,
3 CustomerDetails.FIRST_NAME,
4 CustomerDetails.LAST_NAME,
5 BookingInfo.BOOKING_YEAR,
6 BookingInfo.TOTAL_BOOKING_AMOUNT,
7 HotelLocation.CITY
8 FROM
9 (SELECT
10 c.CUSTOMER_ID,
11 c.FIRST_NAME,
12 c.LAST_NAME
13 FROM
14 customers c) AS CustomerDetails

```

The results are displayed in a table with 5 columns: CUSTOMER\_ID, FIRST\_NAME, LAST\_NAME, BOOKING\_YEAR, and TOTAL\_BOOKING\_AMOUNT. The results are ordered by TOTAL\_BOOKING\_AMOUNT in descending order.

CUSTOMER_ID	FIRST_NAME	LAST_NAME	BOOKING_YEAR	TOTAL_BOOKING_AMOUNT	CITY
1	49 Diane	Rose	2018	11816.31	PRAKASAM
2	90 Edwin	Norton	2020	11374	KHORDA
3	162 Virginia	Jones	2018	10060.01	MUMBAI SUBUEBAN
4	192 Mark	Cox MD	2022	9622.52	THIRUVANANTHAPURAM
5	47 Reginald	Russo	2021	9503.55	RAJGARH
6	176 Ryan	Wright	2016	9426.52	NORTH 24 PARGANAS
7	155 Victoria	Herman	2015	8872.85	AHMED NAGAR
8	151 Mary	Robinson	2017	8590.6	PATHANAMTHITTA
9	61 Allen	Hoffman	2021	8544.21	DAUSA
10	230 Craig	Oconnor	2022	8475.33	VAISHALI
11	24 Michael	Davidson	2018	8293.04	ARARIA
12	198 Brianna	Montgomery	2018	8285.85	THRISSUR
13	54 Sheri	Aguilar	2020	7963.43	Yavatmal
14	117 John	Anderson	2022	7891.91	Kolhapur
15	54 Sheri	Aguilar	2022	7811.17	DAUSA
16	208 Cynthia	Meadows	2020	7790.5	TUTICORIN
17	130 Natalie	Anderson	2017	7780.48	VADDARA
18	58 Teresa	Maldonado	2015	7775.92	VAISHALI
19	50 Nathan	Smith	2021	7773.46	ERNAKULAM
20	159 Elizabeth	Hendrix	2021	7773.01	NAGPUR
21	169 Brianna	Malone	2021	7771.66	DUNGARPUR

## Explanation:

The SQL query efficiently aggregates total booking amounts for customers by booking year and hotel city, utilizing subqueries for enhanced clarity. It breaks down the process into manageable parts: extracting customer details, summarizing booking amounts by year, and identifying hotel locations. These components are then joined to compile a comprehensive view, focusing on customer spend across different times and places. The strategy of segmenting the query improves readability and makes modifications easier, ideal for analytical tasks in a graduate-level study. The results are ordered by the total booking amount in descending order, prioritizing the analysis towards customers with higher spending.

## 4. Determining the ranks by total reservations

SELECT

state,

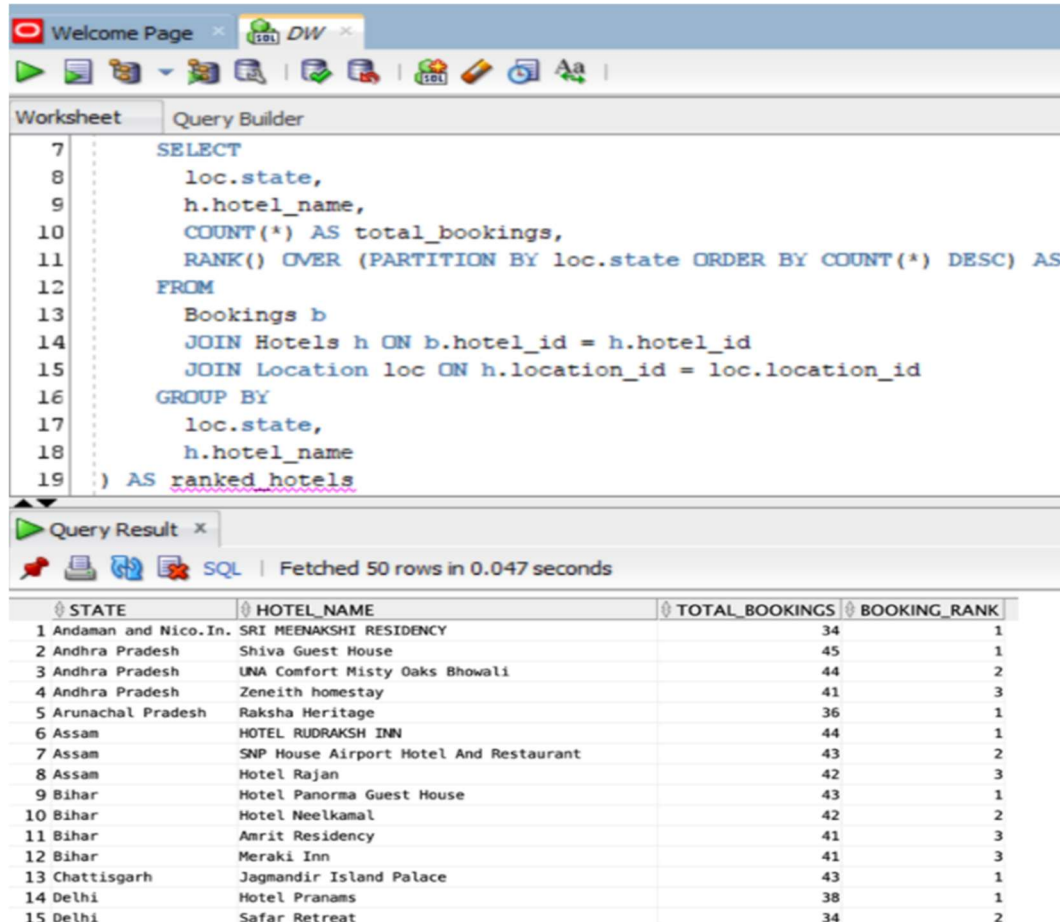
hotel\_name,

total\_bookings,

```

        booking_rank
FROM (
    SELECT
        loc.state,
        h.hotel_name,
        COUNT(*) AS total_bookings,
        RANK() OVER (PARTITION BY loc.state ORDER BY COUNT(*) DESC) AS
        booking_rank
    FROM
        Bookings b
    JOIN Hotels h ON b.hotel_id = h.hotel_id
    JOIN Location loc ON h.location_id = loc.location_id
    GROUP BY
        loc.state,
        h.hotel_name
) AS ranked_hotels
WHERE
    booking_rank <= 3;

```



The screenshot shows a SQL query editor window with a 'Query Builder' tab. The query is as follows:

```

SELECT
    loc.state,
    h.hotel_name,
    COUNT(*) AS total_bookings,
    RANK() OVER (PARTITION BY loc.state ORDER BY COUNT(*) DESC) AS
    booking_rank
FROM
    Bookings b
JOIN Hotels h ON b.hotel_id = h.hotel_id
JOIN Location loc ON h.location_id = loc.location_id
GROUP BY
    loc.state,
    h.hotel_name
) AS ranked_hotels
WHERE
    booking_rank <= 3;

```

Below the query, the 'Query Result' tab shows the results of the query. The results are displayed in a table with the following columns: STATE, HOTEL\_NAME, TOTAL\_BOOKINGS, and BOOKING\_RANK. The results are sorted by STATE and then by HOTEL\_NAME.

STATE	HOTEL_NAME	TOTAL_BOOKINGS	BOOKING_RANK
1 Andaman and Nico.In.	SRI MEENAKSHI RESIDENCY	34	1
2 Andhra Pradesh	Shiva Guest House	45	1
3 Andhra Pradesh	UNA Comfort Misty Oaks Bhowali	44	2
4 Andhra Pradesh	Zeneith homestay	41	3
5 Arunachal Pradesh	Raksha Heritage	36	1
6 Assam	HOTEL RUDRAKSH INN	44	1
7 Assam	SNP House Airport Hotel And Restaurant	43	2
8 Assam	Hotel Rajan	42	3
9 Bihar	Hotel Panorma Guest House	43	1
10 Bihar	Hotel Neelkamal	42	2
11 Bihar	Amrit Residency	41	3
12 Bihar	Meraki Inn	41	3
13 Chattisgarh	Jagmandir Island Palace	43	1
14 Delhi	Hotel Pranams	38	1
15 Delhi	Safar Retreat	34	2

**Explanation:**

This SQL query ranks hotels within each state by their total bookings and displays the top three for each. It first counts bookings per hotel, groups them by state, and ranks them in descending order of bookings within the same state. By filtering for ranks three or lower, it efficiently identifies the most booked hotels per state, offering a focused analysis of hotel popularity and demand across regions.

**5. Total Reservations for each location:**

```
CREATE MATERIALIZED VIEW mv_total_bookings_by_location
AS
SELECT
    h.HOTEL_ID,
    h.HOTEL_NAME,
    l.CITY,
    l.STATE,
    COUNT(b.BOOKING_ID) AS TOTAL_BOOKINGS
FROM
    HOTELS h
INNER JOIN LOCATION l
    ON h.LOCATION_ID = l.LOCATION_ID
LEFT JOIN BOOKINGS b
    ON h.HOTEL_ID = b.HOTEL_ID
GROUP BY
    h.HOTEL_ID,
    h.HOTEL_NAME,
    l.CITY,
    l.STATE
WITH DATA;
```

Welcome PageDW

<

Explanation:

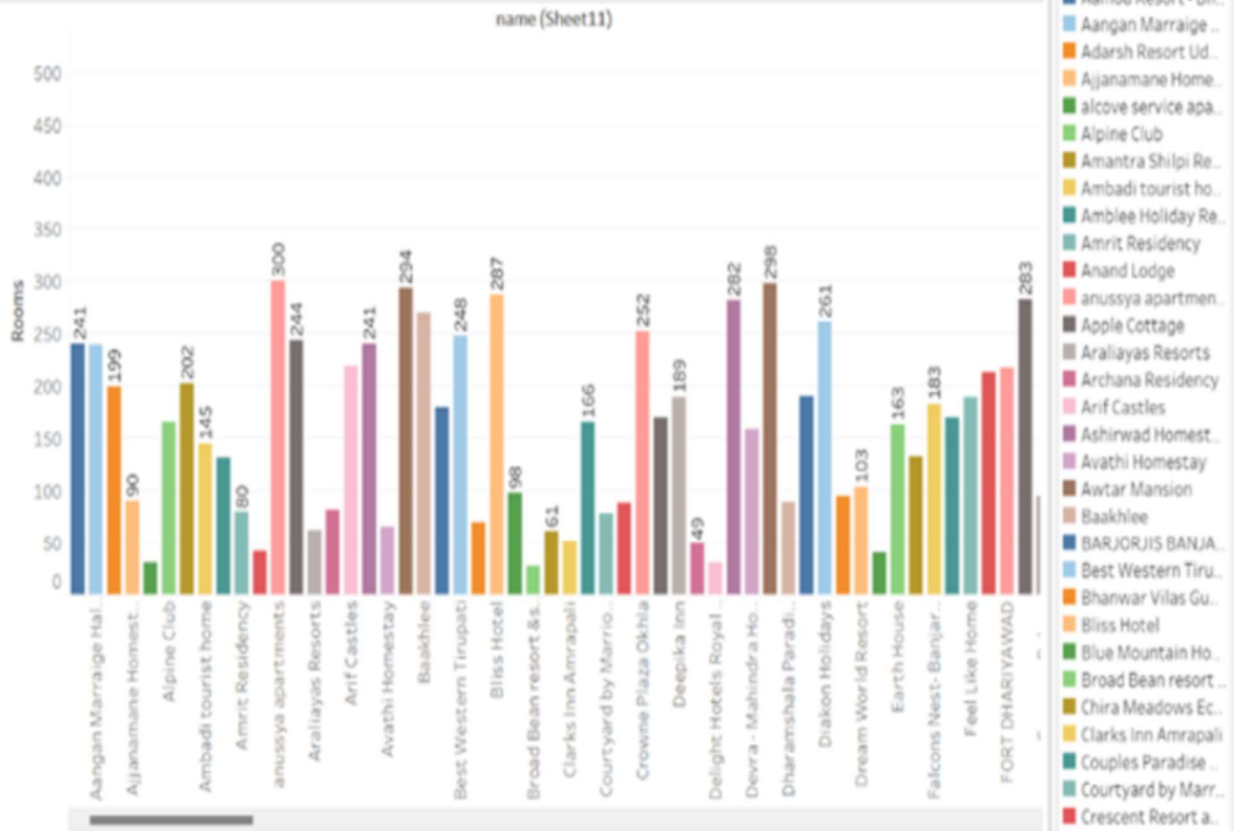
The query constructs a materialized view named **mv\_total\_bookings\_by\_location**, offering a pre-calculated summary of total bookings per hotel, categorized by each hotel's city and state. It achieves this by integrating three tables: **HOTELS**, **LOCATION**, and **BOOKINGS**, ensuring that even hotels without any bookings are counted. By grouping results by hotel ID, name, city, and state, the view efficiently organizes booking data for quick access. This setup demands manual refreshes to keep the data current, making it a convenient tool for analyzing booking trends across locations.

Reporting, Modeling and Storytelling:

1.Which properties boast the largest number of rooms available?

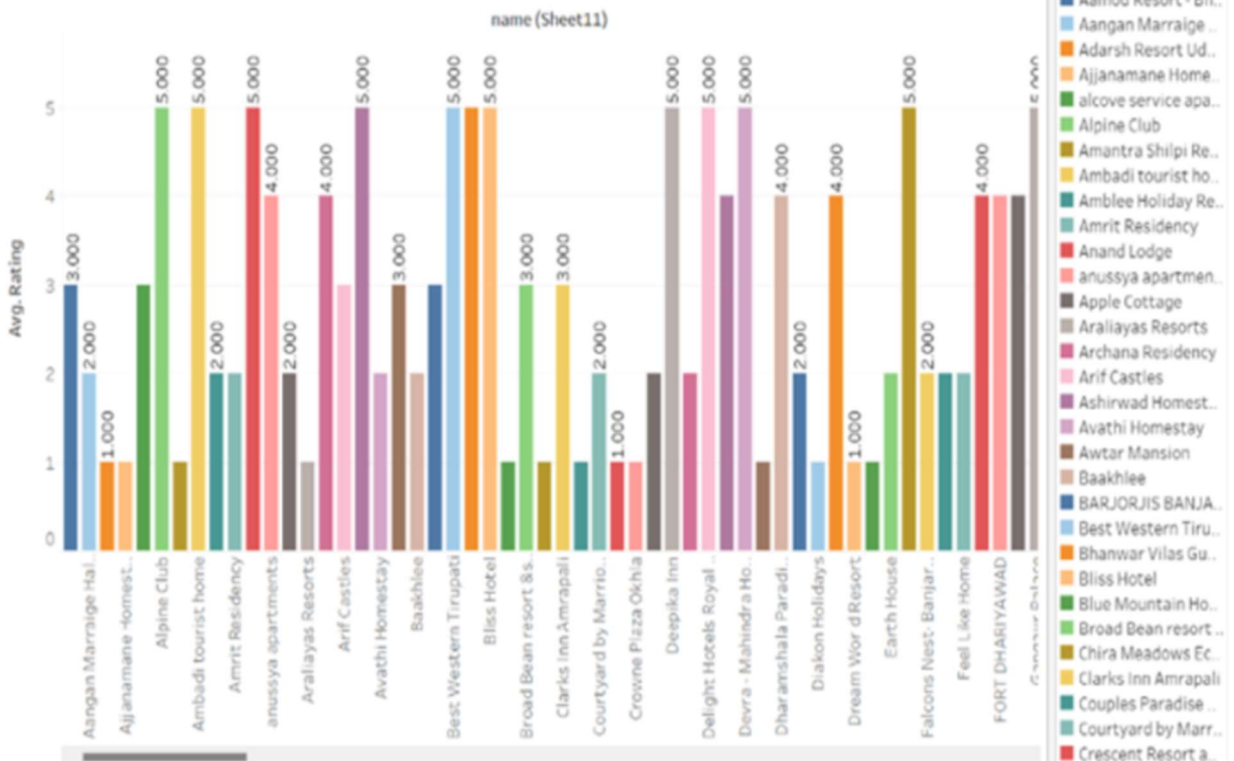
The bar graph effectively showcases the room availability across various hotels, with Anussya Apartments standing out prominently for its extensive capacity. With a total of 300 rooms, Anussya Apartments leads the pack, underscoring its superior accommodation potential compared to its competitors. This visual representation not only highlights the significant disparity in room counts among hotels but also positions Anussya Apartments as a prime choice for guests seeking plentiful lodging options.

Available rooms in each Hotel



2.Which hotels are distinguished by the highest average customer ratings?

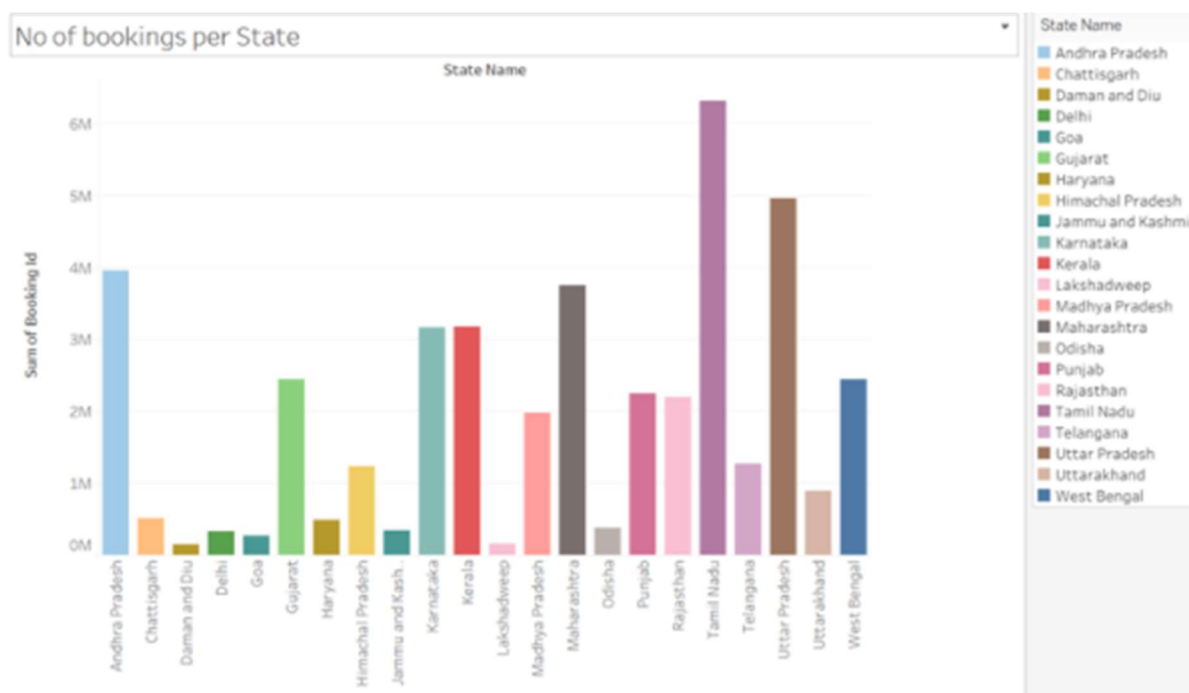
Ratings of each Hotel





The displayed bar graph provides a detailed comparison of average customer ratings received by various hotels over the period from 2015 to 2022. Within this competitive landscape, The Alpine Club emerges as a clear leader, securing the highest average rating among its peers, including several renowned establishments. This achievement underscores The Alpine Club's exceptional performance in maintaining superior customer satisfaction levels throughout these years. The graph not only illustrates the comparative success of The Alpine Club in ensuring guest contentment but also reflects its consistent dedication to quality and service excellence, setting a high standard for the hospitality industry during the observed period.

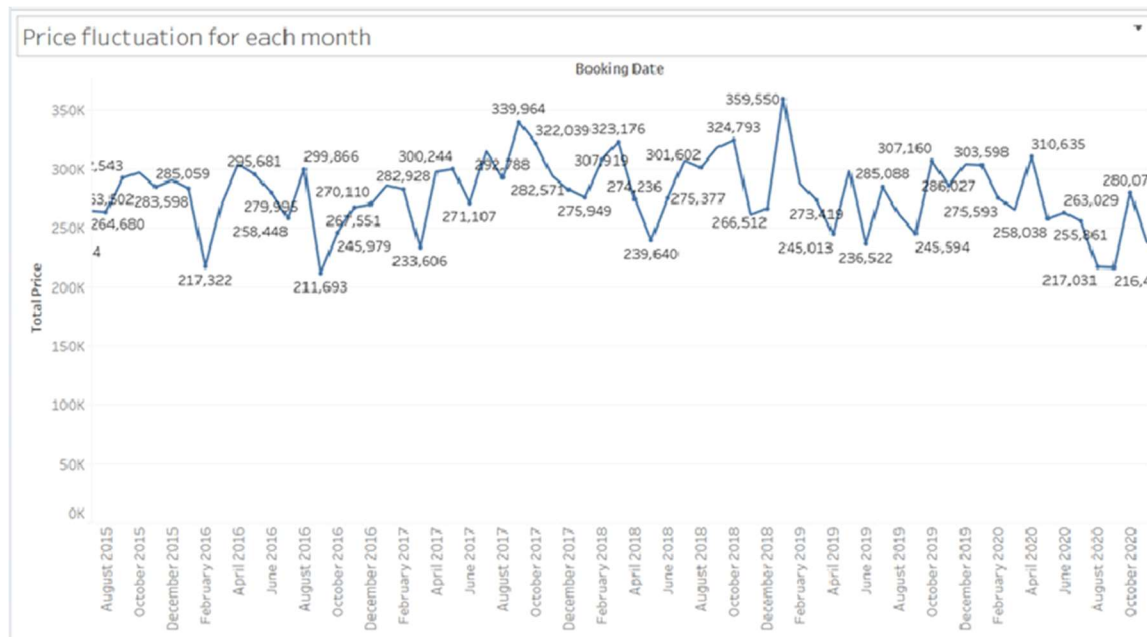
### 3.What is the cumulative number of bookings made by the state over the period from 2015 to 2022?



The bar chart provides a comprehensive overview of the total hotel bookings across different states from 2015 to 2022. Among the states analyzed, Tamil Nadu stands out significantly with the highest booking volume, exceeding 6 million bookings within the eight-year timeframe. This remarkable figure not only highlights Tamil Nadu's dominant position in attracting visitors but also suggests a robust tourism and hospitality sector within the state. On the opposite end of the spectrum, Daman and Diu recorded the lowest booking figure, with just 150,000 bookings, underscoring a stark contrast in tourism dynamics when compared to Tamil Nadu. The chart thus effectively captures the vast disparities in hotel booking trends across states, offering insights into regional preferences and the varying levels of tourist influx, which could be instrumental for stakeholders in understanding market performance and planning future strategies.



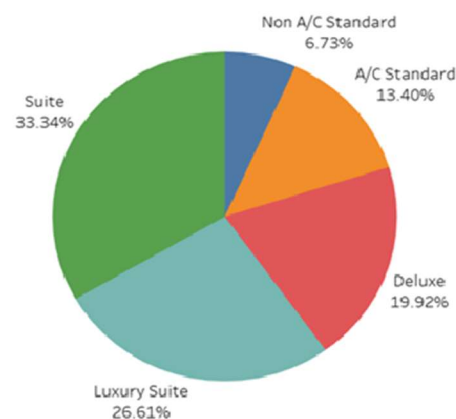
#### 4. How has the total expenditure on hotel bookings evolved from 2015 through 2022?



The line graph presented offers an insightful analysis of hotel pricing trends from 2015 to 2022, with a particular emphasis on the fluctuations observed during different months of the year. A pattern of notable significance is the consistent increase in prices during August and December, periods that align with peak travel seasons, traditionally characterized by a surge in tourist activity. This trend suggests a strategic adjustment of hotel rates in response to the heightened demand during these popular holiday times, indicating a dynamic pricing model that hotels appear to adopt to capitalize on the increased willingness of travelers to pay premium prices for accommodation. Such pricing strategies, while expected, provide a clear illustration of how external factors such as seasonality directly influence hotel pricing, reflecting the industry's responsiveness to market demands and the cyclical nature of travel trends.

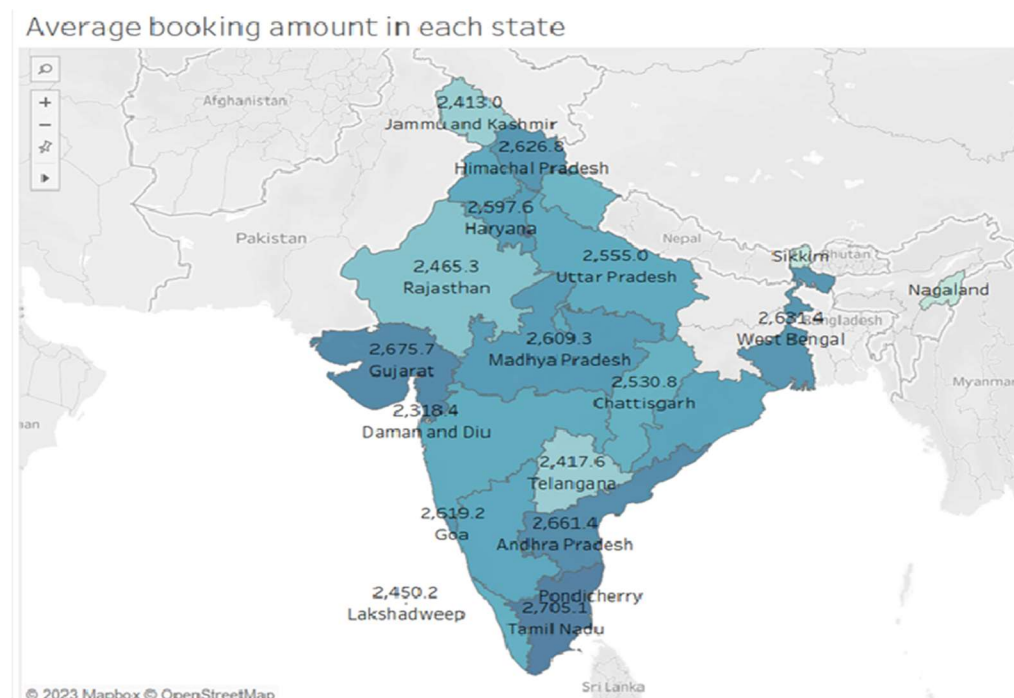
#### 5. What category of rooms contributes the most to hotel revenue?

Percentage contribution of each room type to the total booking amount



The pie chart provided delineates the revenue distribution across different room types, offering a clear visual breakdown of their respective contributions to the total income. Remarkably, Suites emerge as the most significant revenue generator, accounting for 33.34% of the total, surpassing all other room categories. Following closely are the Luxury Suites, which, despite being positioned as a high-end offering, contribute a slightly lower percentage of 26.61% to the overall revenue. This data highlights an interesting market dynamic where, although Luxury Suites are priced at a premium, General Suites lead in revenue generation. This could indicate a higher occupancy rate for General Suites or a preference among a broader segment of guests for these rooms due to their value proposition. Such insights are invaluable for understanding consumer behavior and strategizing hotel operations to maximize profitability while catering to diverse guest preferences.

## 6. In which state do hotels, on average, command the highest booking prices?



The map offers a comprehensive visualization of the disparities in average hotel room prices across various states in India, effectively mapping out the geographical pricing landscape for accommodations. It highlights Haryana as the state with the highest average room price, standing at 2754.1 rupees. This figure could suggest a high demand for accommodations in Haryana, possibly driven by its economic activities or tourist attractions, which in turn justifies the elevated room prices. On the other end of the spectrum, Daman & Diu is showcased with the lowest average price of 2318 rupees. This lower average could be indicative of either a less intense demand for hotel rooms or a more competitive market environment where hotels are incentivized to offer more attractive pricing to lure guests. This variation in room pricing across states not only sheds light on the economic and demand-driven factors influencing hotel pricing strategies but also provides valuable insights for potential travelers and stakeholders in the hospitality industry, underlining the significant impact of location on accommodation costs.

## **Conclusion:**

This in-depth analysis, covering data from 2015 to 2022, sheds light on key aspects of the hospitality sector, revealing a multifaceted and evolving industry landscape. It delves into critical areas such as hotel ratings, the availability of rooms, price dynamics, and the distribution of bookings across various Indian states. The findings highlight a competitive environment where the Alpine Club leads in terms of customer satisfaction, while Anussya Apartments stands out for its extensive room capacity. The observed price trends underscore the influence of seasonal demand on room rates, particularly noting increases in December and August. Additionally, the study maps the variance in average room prices by state, with Haryana commanding the highest prices, indicative of possibly higher demand or premium offerings, and Daman & Diu at the lower end, suggesting either a competitive pricing strategy or lesser demand. Tamil Nadu's position as a booking hotspot further enriches the analysis, illustrating its pivotal role in the industry's geographical booking patterns. This comprehensive review not only offers deep insights into the current state of the hospitality industry but also serves as a valuable tool for strategic planning and decision-making, highlighting areas of opportunity and growth within the sector.

## **References:**

### **Datasets:**

Data world, Hotel Bookings Dataset

<https://data.world/mesum/hotel-bookings-dataset>

Kaggle, Hotel booking Dataset

<https://www.kaggle.com/datasets/abdurahmankhaled1/hotel-booking-dataset/>