TERRIFIC TRAVEL AGENCY

Name: Salim Tobi Ayinde

Table Of Content

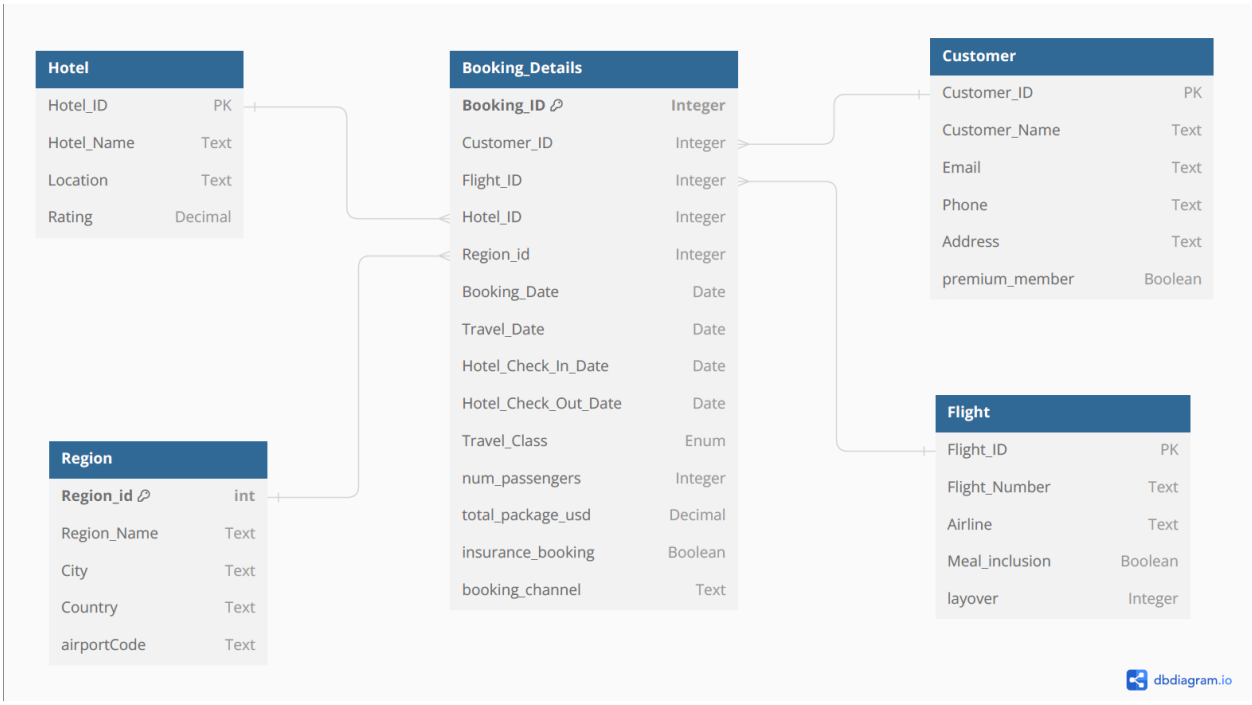
Reasons For Creating the Traveling Dataset

The dataset for the travel agency is designed to provide valuable insights and support various aspects of the agency's operations. Here are some reasons for having such a dataset and the objectives it aims to achieve:

- **Customer Understanding:**
  - **Questions:** Who are our customers? What are their preferences? Where are they located?
  - **Objective:** Gain insights into customer demographics, preferences, and behaviors to tailor services and enhance customer experiences.
- **Revenue Optimization:**
  - **Questions:** What are our revenue trends? How does revenue vary across regions and services?
  - **Objective:** Identify revenue patterns, optimize pricing strategies, and allocate resources effectively for increased profitability.
- **Operational Efficiency:**
  - **Questions:** How efficiently are we utilizing resources? What are the seasonal demand variations?
  - **Objective:** Enhance operational efficiency by analyzing demand fluctuations, optimizing resource allocation, and minimizing costs.
- **Risk Assessment:**
  - **Questions:** What are the risk factors? How do customer segments impact revenue and insurance bookings?
  - **Objective:** Evaluate risk factors, assess the impact of customer segments on revenue, and enhance risk management strategies.
- **Customer Segmentation:**
  - **Questions:** How can customers be categorized based on booking frequency and spending?
  - **Objective:** Segment customers for targeted marketing, personalized offerings, and improved customer relationship management.
- **Regional Analysis:**
  - **Questions:** Which regions contribute the most to revenue? What are the key destinations?
  - **Objective:** Analyze regional revenue contributions, identify popular destinations, and tailor services based on regional demand.
- **Service Comparison:**
  - **Questions:** How do flight and hotel services contribute to overall revenue?
  - **Objective:** Compare the revenue generated by different services, optimize service offerings, and refine business strategies.
- **Seasonal Demand Variation:**
  - **Questions:** What are the seasonal booking trends? How does revenue vary during peak seasons?
  - **Objective:** Understand seasonal variations, plan for peak demand, and optimize marketing strategies accordingly.

Showing the Relational schema Based on Plan of Project

Creating the schema marked the initial phase of the project, requiring the establishment of a structured framework to illustrate the logical representation of the database. This involved defining relationships among entities and mapping attributes to ensure a coherent foundation for the subsequent stages of the project.



Code can be access here.

Creation of the Tables

Data sets used can be found below:

a. **Booking Details**: Contains comprehensive booking information, including customer details, flight and hotel IDs, booking and travel dates, travel class, and financial details.
b. **Customer Table**: Provides customer-specific data such as customer ID, name, contact information, address, and premium membership status, crucial for understanding customer behavior.
c. **Flight Table**: Includes details about flights, such as flight ID, flight number, airline, departure, and arrival airport codes, essential for analyzing flight-related patterns.
d. **Hotel Table**: Encompasses hotel-related information, including hotel ID, name, location, rating, and type, facilitating analysis of hotel-specific trends and preferences.
e. **Region Table**: Offers regional data, including region ID, name, city, country, and airport code, supporting insights into geographical revenue distribution and regional trends.

Two types of tables were created, and these include below:

Dimension tables: These are tables that contain descriptive, textual, or categorical information, which is typically the entry points to data.

Facts tables: These are tables that contain the performance metrics of specific business processes.

Tables Created can be seen below:

- **Booking Details:**

QUERIES USED:

```
CREATE TABLE Booking_Details
(booking_id serial PRIMARY KEY, customer_id int, flight_id int, hotel_id int, region_id int,
booking_date date, travel_date date, hotel_check_in_date date, hotel_check_out_date date,
travel_class varchar (50), num_passengers int, total_package_usd decimal, insurance_booking text,
booking_channel varchar(20),
FOREIGN KEY (customer_id) REFERENCES Customer(customer_id) ON UPDATE CASCADE ON DELETE CASCADE,
FOREIGN KEY (flight_id) REFERENCES Flight(flight_id) ON UPDATE CASCADE ON DELETE CASCADE,
FOREIGN KEY (hotel_id) REFERENCES Hotel(hotel_id) ON UPDATE CASCADE ON DELETE CASCADE,
FOREIGN KEY (region_id) REFERENCES Region(region_id) ON UPDATE CASCADE ON DELETE CASCADE);
```

- **Customer Table:**

QUERIES USED:

```
CREATE TABLE Customer
(customer_id serial PRIMARY KEY, customer_name varchar (100), email text, phone varchar (100),
address varchar (100), premium_member boolean);
```

- **Flight Table:**

```
QUERIES USED:

CREATE TABLE Flight
(flight_id serial PRIMARY KEY, flight_number varchar (100), airline text, meal_inclusion boolean,
layover int );
```

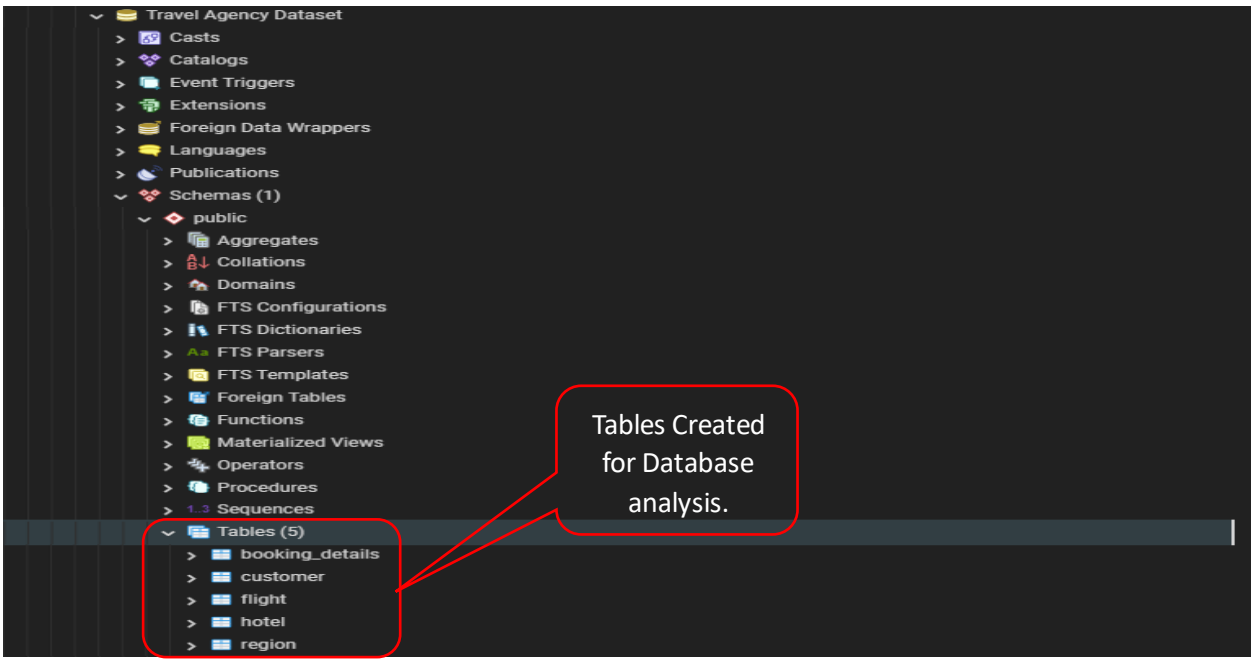
- **Hotel Table:**

```
QUERIES USED:

CREATE TABLE Hotel
(hotel_id serial PRIMARY KEY, hotel_name varchar (100), location varchar (100), rating decimal);
```

- **Region Table:**

```
QUERIES USED:

CREATE TABLE Region
(region_id serial PRIMARY KEY, region_name varchar (100), city varchar (100), country varchar (100),
airport_code text);
```

Kindly find a visual representation of the tables created below.

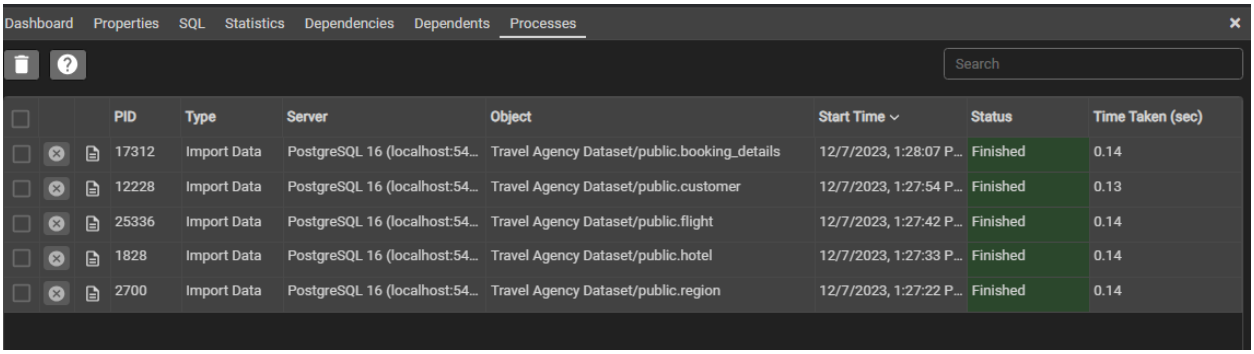


Importing of data into Tables

We utilized the import function to populate each table with data, capturing extensive information on bookings, customer details, flights, hotels, and regions. The following exemplifies how each import function was employed, including instances of the inserted data for selected records within each table.

The import function provided an efficient and organized means to insert data into each table, ensuring accuracy and completeness. The selected records within each table offer illustrative examples of the inserted data, contributing to a comprehensive dataset for subsequent analysis and insights.

Below shows the visual display of imported data into tables for analysis.



PID	Type	Server	Object	Start Time	Status	Time Taken (sec)
17312	Import Data	PostgreSQL 16 (localhost:54...	Travel Agency Dataset/public.booking_details	12/7/2023, 1:28:07 P...	Finished	0.14
12228	Import Data	PostgreSQL 16 (localhost:54...	Travel Agency Dataset/public.customer	12/7/2023, 1:27:54 P...	Finished	0.13
25336	Import Data	PostgreSQL 16 (localhost:54...	Travel Agency Dataset/public.flight	12/7/2023, 1:27:42 P...	Finished	0.14
1828	Import Data	PostgreSQL 16 (localhost:54...	Travel Agency Dataset/public.hotel	12/7/2023, 1:27:33 P...	Finished	0.14
2700	Import Data	PostgreSQL 16 (localhost:54...	Travel Agency Dataset/public.region	12/7/2023, 1:27:22 P...	Finished	0.14

Booking Details: Employed the import function to seamlessly insert comprehensive booking information, encompassing customer details, flight and hotel IDs, booking and travel dates, travel class, and financial details.

Customer Table: Utilized the import function to populate the table with essential customer-specific data, including customer ID, name, contact information, address, and premium membership status, facilitating a deeper understanding of customer behavior.

Flight Table: Leveraged the import function to insert details about flights, including flight ID, flight number, airline, departure, and arrival airport codes. This approach ensured a streamlined process for analyzing flight-related patterns.

Hotel Table: Executed the import function to input hotel-related information, such as hotel ID, name, location, rating, and type. This allowed for thorough analysis of hotel-specific trends and preferences.

Region Table: Applied the import function to add regional data, encompassing region ID, name, city, country, and airport code. This was instrumental in gaining insights into geographical revenue distribution and regional trends.

Data Cleaning for Analysis

Cleaning operations contribute to data consistency, accuracy, and improved analysis capabilities, aligning the dataset with the project's objectives and ensuring meaningful insights.

Below were the steps taken to clean the dataset:

Correcting Travel Class:

- **Reason**: Ensuring standardized and consistent representation of travel classes for accurate analysis.
- **Description**: Updated the travel_class column in the booking_details table, replacing variations like 'frst' and '1st' with 'First Class', 'busi' with 'Business', and 'econ' with 'Economy'.

QUERIES USED:

```sql
-- Update 'First Class'
UPDATE booking_details
SET travel_class = 'First Class'
WHERE LOWER(travel_class) ILIKE '%frst%' OR LOWER(travel_class) LIKE '%1st%';

-- Update 'Business'
UPDATE booking_details
SET travel_class = 'Business'
WHERE LOWER(travel_class) = 'busi';

-- Update 'Economy'
UPDATE booking_details
SET travel_class = 'Economy'
WHERE LOWER(travel_class) = 'econ';
```

Updating Insurance Booking:

- **Reason**: Normalizing boolean values for insurance booking to 'Yes' or 'No' for clarity and consistency.
- **Description**: Modified the insurance_booking column in the booking_details table, updating values like 'Yyess' and 'Y' to 'Yes', and 'N' and 'None' to 'No'.

QUERIES USED:

```sql
-- Update 'Yyess' and 'Y' to 'Yes'
UPDATE booking_details
SET insurance_booking = 'Yes'
WHERE (insurance_booking) IN ('Yyees', 'Y');

-- Update 'N' and 'None' to 'No'
UPDATE booking_details
SET insurance_booking = 'No'
WHERE (insurance_booking) IN ('N', 'None');
```

Updating Email with Placeholder

- **Reason**: Ensuring no NULL or empty values in the email column, maintaining data integrity.
- **Description**: Set a placeholder email ('placeholder@example.com') for records

```
QUERIES USED:

UPDATE customer SET email = 'placeholder@example.com' WHERE email IS NULL OR email
= '';
```

Deleting Duplicate Customer Records:

- **Reason**: Eliminating redundancy and maintaining a clean customer dataset.
- **Description**: Removed duplicate records from the customer table, keeping only the one with the minimum customer_id for each unique phone number.

```
QUERIES USED:

DELETE FROM customer
WHERE customer_id NOT IN (
    SELECT MIN(customer_id) AS min_id
    FROM customer
    GROUP BY phone
);
```

Formatting Phone Numbers:

- **Reason**: Standardizing the format of phone numbers for consistency and better presentation.
- **Description**: Updated the phone column in the customer table to follow the format **(**'###-###-####'.)

```
QUERIES USED:

UPDATE customer
SET phone = CONCAT(
    SUBSTRING(phone FROM 1 FOR 3),
    '-',
    SUBSTRING(phone FROM 4 FOR 3),
    '-',
    SUBSTRING(phone FROM 7 FOR 4)
);
```

Changing Abbreviated Country Names:

- **Reason**: Enhancing clarity by replacing abbreviated country names with full names.
- **Description**: Altered the country column in the region table, replacing 'USA' with 'United States' and 'UK' with 'United Kingdom'.

QUERIES USED:

```
UPDATE region
SET country =
    CASE
        WHEN country = 'USA' THEN 'United States'
        WHEN country = 'UK' THEN 'United Kingdom'
        ELSE country
    END;
```

Analysis, Visualization & Insights

1. Revenue Analysis:

1.1 Revenue Trends Analysis (Evaluate the overall revenue trends over time)

Query:

```
QUERIES USED:

WITH CTE_Monthly_Revenue AS (
SELECT EXTRACT(YEAR FROM booking_date) AS Sale_Year,EXTRACT(MONTH FROM booking_date) AS
Sale_Month,
TO_CHAR(booking_date, 'Month') AS Month_Name, SUM(Total_Package_USD) AS Total_Revenue,
ROW_NUMBER() OVER (PARTITION BY EXTRACT(YEAR FROM booking_date),
EXTRACT(MONTH FROM booking_date) ORDER BY SUM(Total_Package_USD) DESC) AS Month_Rank
FROM Booking_Details
GROUP BY EXTRACT(YEAR FROM booking_date), EXTRACT(MONTH FROM booking_date),
TO_CHAR(booking_date, 'Month')
)
SELECT Sale_Year,Month_Name,Total_Revenue,
COALESCE(ROUND(((Total_Revenue - LAG(Total_Revenue, 1) OVER (ORDER BY Sale_Year, Sale_Month)) /
LAG(Total_Revenue, 1) OVER (ORDER BY Sale_Year, Sale_Month)) * 100, 2), 0) AS Percent_Change
FROM CTE_Monthly_Revenue
ORDER BY Sale_Year, Sale_Month;
```

Output:

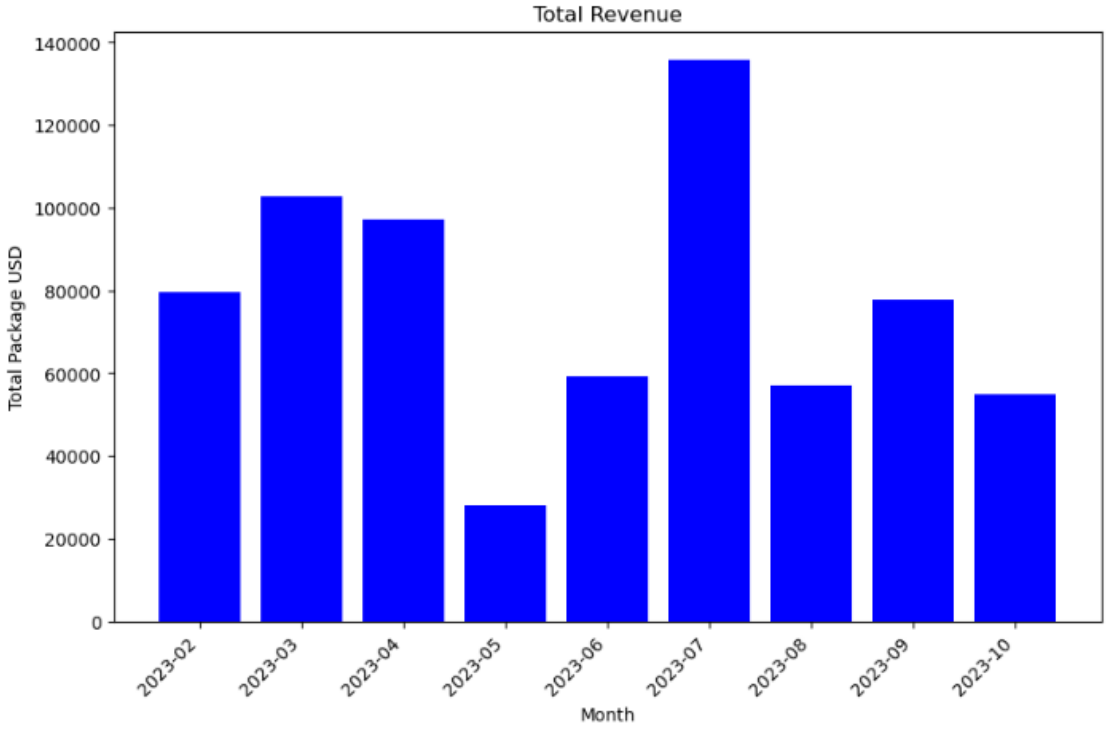
Data Output | Messages | Notifications

	sale_year numeric	month_name text	total_revenue numeric	percent_change numeric
1	2023	February	79600	0
2	2023	March	102800	29.15
3	2023	April	97200	-5.45
4	2023	May	27900	-71.30
5	2023	June	59100	111.83
6	2023	July	135900	129.95
7	2023	August	57100	-57.98
8	2023	September	77600	35.90
9	2023	October	54800	-29.38

Summary

Utilizing the window function and organizing data by year and month to calculate total revenue per month and determine the percentage change in revenue from previous month, and finally sorting by year and month.

Visualization:



Insights:

The revenue trends observed during 2023 demonstrate varying patterns in monthly revenues. Months such as July and March have significant increases in revenue, maybe suggesting peak travel seasons or promotional activities. In contrast, notable decreases in May, August, and October indicate a reduction in travel demand or less profitable seasons. The variations in client behavior might be attributed to seasonal travel preferences, as well as exogenous variables like economic swings or marketing tactics.

1.2 Geographical Revenue Distribution Analysis:

Query:

```
QUERIES USED:

SELECT RG.country AS Country, RG.region_name AS Region,SUM(BD.Total_Package_USD) AS Total_Revenue,
ROUND(AVG_Region.Avg_Revenue_Per_Region, 2) AS Avg_Revenue_Per_Region
FROM Booking_Details BD
LEFT JOIN Region RG ON BD.region_id = RG.region_id
LEFT JOIN (SELECT region_name AS Region_Name,
AVG(Total_Package_USD) AS Avg_Revenue_Per_Region
FROM Booking_Details BD
JOIN Region ON BD.region_id = Region.region_id
GROUP BY region_name
) AS AVG_Region ON RG.region_name = AVG_Region.Region_Name
GROUP BY RG.country, RG.region_name, AVG_Region.Avg_Revenue_Per_Region
ORDER BY Total_Revenue DESC;
```

Output:

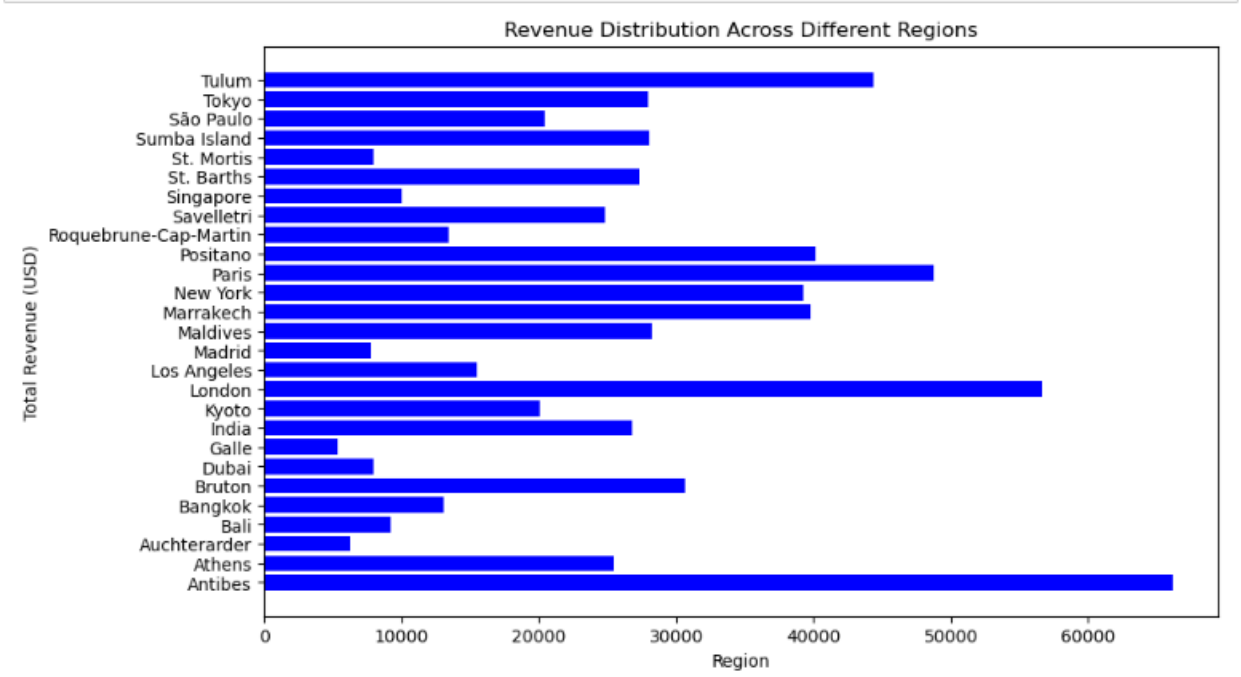
Data Output | Messages | Notifications

	country character varying (100)	region character varying (100)	total_revenue numeric	avg_revenue_per_region numeric
1	France	Antibes	66200	7355.56
2	United Kingdom	London	56700	5670.00
3	France	Paris	48800	6100.00
4	Mexico	Tulum	44400	5550.00
5	Italy	Positano	40200	5742.86
6	Morocco	Marrakech	39800	6633.33
7	United States	New York	39300	4366.67
8	United Kingdom	Bruton	30700	6140.00
9	Maldives	Maldives	28200	7050.00
10	Indonesia	Sumba Island	28100	9366.67
11	Japan	Tokyo	28000	4666.67
12	Saint Barthelemy	St. Barths	27300	5460.00
13	India	India	26800	5360.00
14	Greece	Athens	25500	8500.00
15	Italy	Savelletri	24800	6200.00
16	Brazil	São Paulo	20400	6800.00
17	Japan	Kyoto	20100	6700.00
18	United States	Los Angeles	15500	5166.67
19	France	Roquebrune-Cap-Martin	13400	3350.00
20	Thailand	Bangkok	13100	6550.00
21	Singapore	Singapore	10000	10000.00
22	Indonesia	Bali	9200	3066.67
23	Switzerland	St. Mortis	8000	4000.00
24	United Arab Emirates	Dubai	8000	4000.00
25	Spain	Madrid	7800	3900.00
26	United Kingdom	Auchterarder	6300	6300.00
27	Sri Lanka	Galle	5400	5400.00

Summary:

Using aggregation functions and grouping the data by country and region to derive the total and average revenue per region and country, showcasing revenue distribution.

Visualization:



Insights:

The Geographical Revenue Distribution demonstrates the allocation of revenue across several nations and their corresponding regions. The data reveals significant disparities in revenue across different regions, highlighting locations such as Antibes in France and London in the United Kingdom as major contributors to high revenue. This suggests that there are varying average revenue levels within regions of the same nation. Possible factors contributing to these variances include regional demand drivers, unique attractions, cultural events, or marketing strategies.

1.3 Seasonal Revenue Fluctuations (Identify revenue fluctuations based on seasons):

Query:

```
QUERIES USED:

WITH CTE_Seasonal_Revenue AS (SELECT
EXTRACT(YEAR FROM booking_date) AS Sale_Year,
    CASE
        WHEN EXTRACT(MONTH FROM booking_date) IN (3, 4, 5) THEN 'Spring'
        WHEN EXTRACT(MONTH FROM booking_date) IN (6, 7, 8) THEN 'Summer'
        WHEN EXTRACT(MONTH FROM booking_date) IN (9, 10, 11) THEN 'Fall'
        ELSE 'Winter'
 END AS Season,
 SUM(Total_Package_USD) AS Total_Revenue
 FROM Booking_Details
 GROUP BY Sale_Year, Season
)
SELECT Sale_Year,Season,Total_Revenue
FROM CTE_Seasonal_Revenue
ORDER BY Sale_Year, Season;
```

Output:

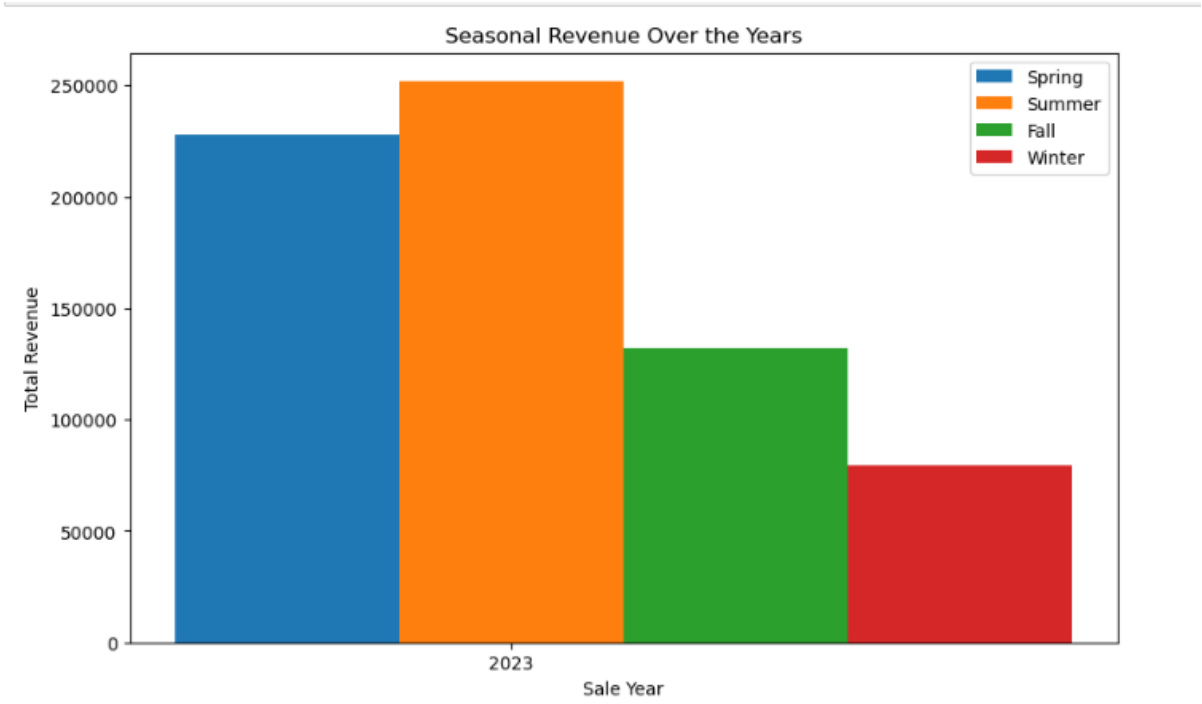
Data Output | Messages | Notifications

	sale_year numeric	season text	total_revenue numeric
1	2023	Fall	132400
2	2023	Spring	227900
3	2023	Summer	252100
4	2023	Winter	79600

Summary:

Applying conditional functions and aggregation based on season categorization (Spring, Summer, Fall, Winter), to illustrate revenue fluctuations across different seasons and grouping it by year and season to highlight seasonal revenue patterns.

Visualization:



Insights:

The Seasonal Revenue Fluctuations Analysis illustrates the variations in revenue throughout different seasons, highlighting significant increases in revenue during the Summer and Spring seasons in comparison to the Fall and Winter seasons. This implies a seasonal pattern in which the months with higher temperatures tend to generate greater revenue, maybe due to an increase in travel during holiday times.

1.4 Monthly Revenue Forecasting and Deviation Analysis:

Query

```
QUERIES USED:

WITH CTE_Monthly_Revenue_Forecasting AS (SELECT
EXTRACT(YEAR FROM booking_date) AS Sale_Year, EXTRACT(MONTH FROM booking_date) AS Sale_Month,
SUM(total_package_usd) AS Total_Revenue
FROM Booking_Details
GROUP BY EXTRACT(YEAR FROM booking_date), EXTRACT(MONTH FROM booking_date)
)

SELECT Sale_Year,Sale_Month,Total_Revenue,
ROUND(AVG(Total_Revenue) OVER (ORDER BY Sale_Year,
Sale_Month ROWS BETWEEN 3 PRECEDING AND 1 FOLLOWING), 2) AS Forecasted_Revenue,
ROUND((Total_Revenue - AVG(Total_Revenue) OVER (ORDER BY Sale_Year,
Sale_Month ROWS BETWEEN 3 PRECEDING AND 1 FOLLOWING)) / AVG(Total_Revenue)
OVER (ORDER BY Sale_Year, Sale_Month ROWS BETWEEN 3 PRECEDING AND 1 FOLLOWING) * 100, 2) AS
Deviation_Percentage
FROM CTE_Monthly_Revenue_Forecasting
ORDER BY Sale_Year, Sale_Month;
```

Output:

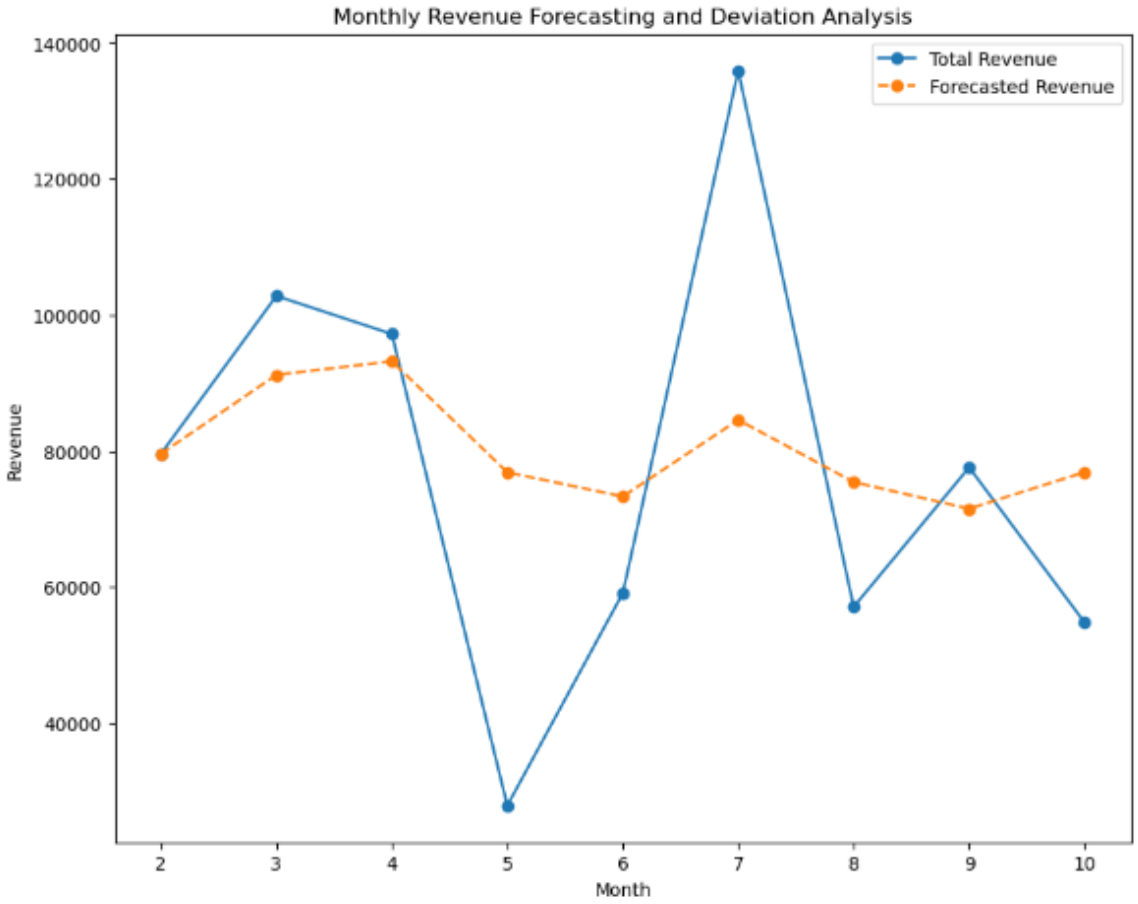
Data Output | Messages | Notifications

	sale_year numeric	sale_month numeric	total_revenue numeric	forecasted_revenue numeric	deviation_percentage numeric
1	2023	2	79600	91200.00	-12.72
2	2023	3	102800	93200.00	10.30
3	2023	4	97200	76875.00	26.44
4	2023	5	27900	73320.00	-61.95
5	2023	6	59100	84580.00	-30.13
6	2023	7	135900	75440.00	80.14
7	2023	8	57100	71520.00	-20.16
8	2023	9	77600	76900.00	0.91
9	2023	10	54800	81350.00	-32.64

Summary:

Created CTE and used window function to calculate monthly revenue, moving average revenue, percentage difference between the monthly revenue and forecasted revenue, and ordering it by Sale year and month.

Visualization:



Insights:

Monthly Revenue Forecasting and Deviation Analysis predicts how much revenue will be made each month in 2023 and figures out what percentage of that revenue will be different from what was predicted. There are changes, especially in May and June, when real sales are very different from what was expected. This could lead to questions about how accurate the forecasting models were during those times, possibly because of outside factors or changes in the way the market works that affected the revenue forecasts.

2. Customer Behavior Analysis:

2.1 Customer Spending Analysis:

Query:

```
QUERIES USED:

SELECT Customer_ID, customer_name, email AS Email,Total_Package_Value,
RANK() OVER (ORDER BY Total_Package_Value DESC) AS Spending_Rank
FROM (SELECT CS.customer_id AS Customer_ID, CS.customer_name, CS.email,
SUM(BD.Total_Package_Usd) AS Total_Package_Value
FROM Booking_Details BD
LEFT JOIN Customer CS ON BD.customer_id = CS.customer_id
GROUP BY CS.customer_id, CS.customer_name, CS.email
) AS CustomerSpending
ORDER BY  Total_Package_Value DESC
LIMIT 20;
```

Output:

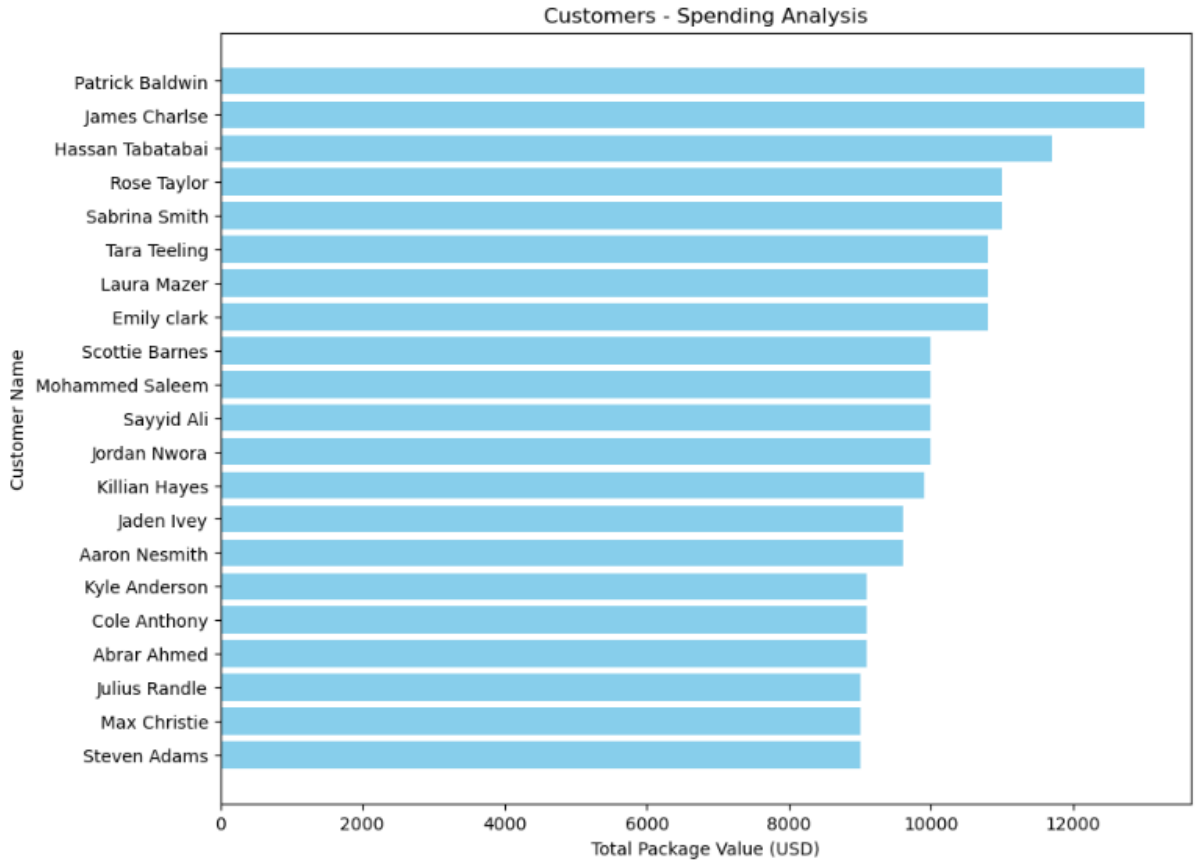
Data Output | Messages | Notifications

	customer_id [PK] integer	customer_name character varying (100)	email text	total_package_value numeric	spending_rank bigint
1	209	James Charlse	james@yahoo.com	13000	1
2	121	Patrick Baldwin	Patrick@gmail.com	13000	1
3	168	Hassan Tabatabai	Hassan@gmail.com	11700	3
4	207	Rose Taylor	Taylor@iCloud.com	11000	4
5	230	Sabrina Smith	smithS@yahoo.com	11000	4
6	142	Tara Teeling	Tara@gmail.com	10800	6
7	204	Emily clark	Emily87@conestoga.ca	10800	6
8	145	Laura Mazer	Laura@gmail.com	10800	6
9	198	Jordan Nwora	Jordan@gmail.com	10000	9
10	129	Scottie Barnes	Scottie@gmail.com	10000	9
11	154	Sayyid Ali	Sayyid@gmail.com	10000	9
12	136	Mohammed Saleem	Mohammed@gmail.com	10000	9
13	183	Killian Hayes	Killian@gmail.com	9900	13
14	197	Aaron Nesmith	Aaron@gmail.com	9600	14
15	184	Jaden Ivey	Jaden@gmail.com	9600	14
16	166	Abrar Ahmed	Abrar@gmail.com	9100	16
17	116	Kyle Anderson	Kyle@gmail.com	9100	16
18	117	Cole Anthony	placeholder@example.com	9100	16
19	103	Julius Randle	Julius@gmail.com	9000	19
20	111	Steven Adams	Steven@gmail.com	9000	19

Summary:

Retrieving details of the top 20 customers by their total spending and ranking based on their spending by using left join and ranking function.

Visualization:



Insights:

Based on the overall package values of their clients, the Customer Spending study ranks the highest-spending consumers. The two highest-spending clients on the list, James Charlse and Patrick Baldwin, are noteworthy for their equal expenditure. Many consumers in the same spending range point to a group of consumers with comparable spending capacities, pointing to possible markets for focused advertising campaigns or service offers to keep or grow these consumers' spending and engagement levels.

2.2 Feedback-based Preference Analysis:

Query:

```
QUERIES USED:

SELECT HT.hotel_id AS Hotel_ID,HT.hotel_name AS Hotel_Name,ROUND(AVG(HT.rating), 1) AS
Average_hotel_rating,
COUNT(BD.booking_id) AS Total_Bookings,
CASE
    WHEN AVG(HT.rating) >= 4.5 THEN 'Highly Preferred'
    WHEN AVG(HT.rating) >= 3.8 THEN 'Moderately Preferred'
    ELSE 'Low Preferred'
END AS Preference_Category
FROM Hotel HT
join Booking_Details BD ON HT.hotel_id = BD.hotel_id
GROUP BY HT.hotel_id, HT.hotel_name
ORDER BY Average_Hotel_Rating DESC, Total_Bookings DESC;
```

Output:

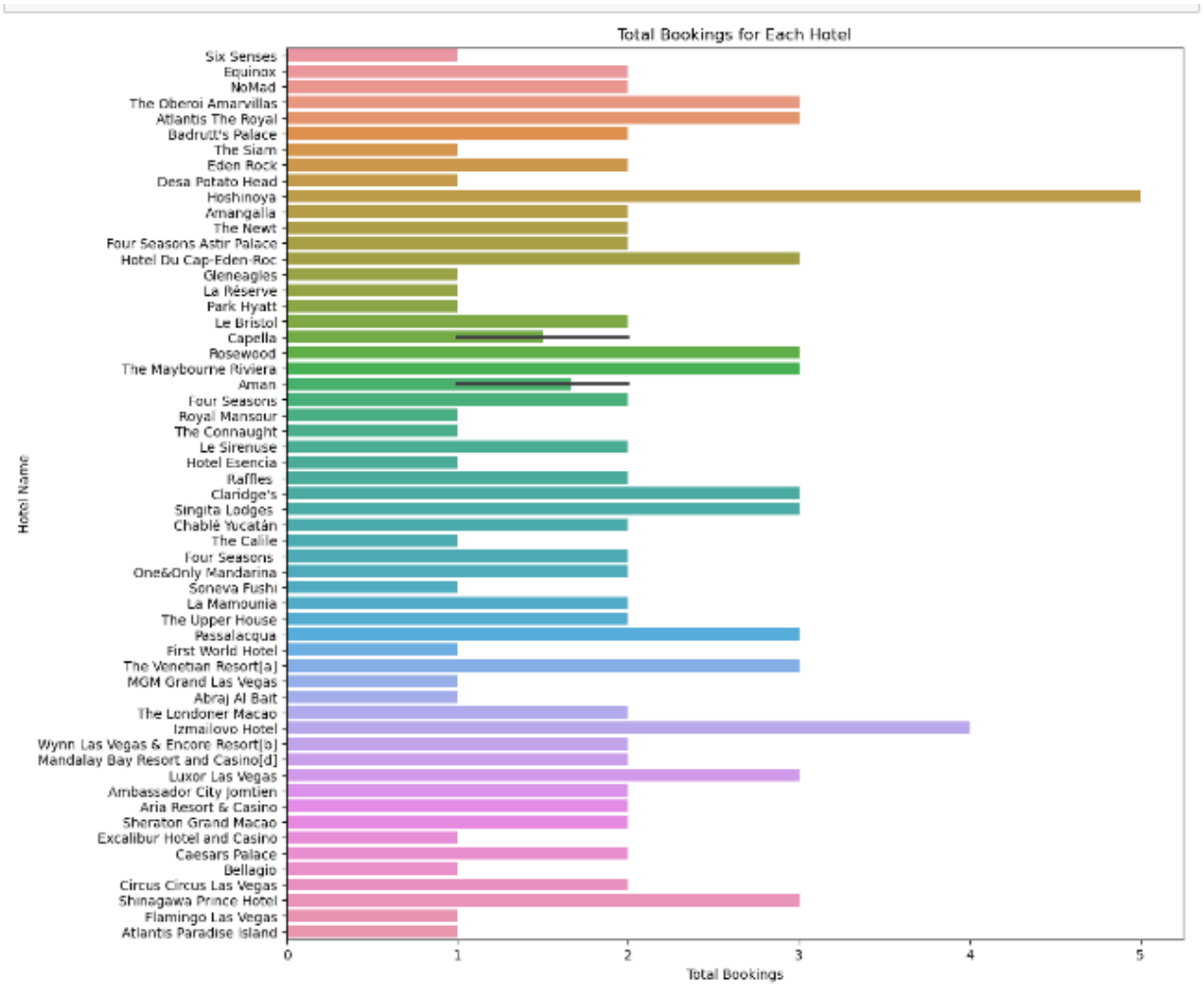
Data Output | Messages | Notifications

	hotel_id [PK] integer	hotel_name character varying (100)	average_hotel_rating numeric	total_bookings bigint	preference_category text
1	560	Passalacqua	5.0	3	Highly Preferred
2	540	Le Sirenuse	5.0	2	Highly Preferred
3	513	Equinox	5.0	2	Highly Preferred
4	563	MGM Grand Las Vegas	5.0	1	Highly Preferred
5	531	Park Hyatt	5.0	1	Highly Preferred
6	577	Circus Circus Las Vegas	4.7	2	Highly Preferred
7	553	One&Only Mandarina	4.7	2	Highly Preferred
8	573	Sheraton Grand Macao	4.7	2	Highly Preferred
9	554	Soneva Fushi	4.7	1	Highly Preferred
10	580	Flamingo Las Vegas	4.7	1	Highly Preferred
11	576	Bellagio	4.7	1	Highly Preferred
12	550	Capella	4.7	1	Highly Preferred
13	521	Desa Potato Head	4.7	1	Highly Preferred
14	529	Gleneagles	4.7	1	Highly Preferred
15	530	La Réserve	4.7	1	Highly Preferred
16	566	Izmailovo Hotel	4.6	4	Highly Preferred
17	545	Claridge's	4.6	3	Highly Preferred
18	536	Aman	4.6	2	Highly Preferred
19	535	The Maybourne Riviera	4.5	3	Highly Preferred
20	517	Atlantis The Royal	4.5	3	Highly Preferred
21	526	Four Seasons Astir Palace	4.5	2	Highly Preferred
22	537	Four Seasons	4.5	2	Highly Preferred
23	512	Six Senses	4.5	1	Highly Preferred
24	559	Rosewood	4.4	3	Moderately Preferred
25	565	The Londoner Macao	4.4	2	Moderately Preferred
26	524	The Newt	4.4	2	Moderately Preferred
27	567	Wynn Las Vegas & Encore Resort[b]	4.4	2	Moderately Preferred
28	544	Raffles	4.4	2	Moderately Preferred
29	564	Abraj Al Bait	4.4	1	Moderately Preferred

Summary:

Calculating the average hotel rating and total bookings for each hotel, categorizing them into Highly Preferred/Moderately Preferred/Low Preferred based on their average ratings to estimate most popular and preferred hotels among customers.

Visualization:



Insights:

The hotel analysis sorts preferences according on bookings and ratings. Consistent reservations are correlated with high ratings, indicating that ratings have an effect on choice. There is a need to adjust services to meet the different demands of customers, since moderately liked hotels have inconsistent ratings but consistent reservations. There is opportunity for improvement to boost demand for hotels with lower ratings (<3.8) as they have fewer bookings.

2.3 Travel Class Preference Analysis by Expenditure:

Query:

```
QUERIES USED:

WITH CTE_Class_Spending AS (
SELECT travel_class,
COUNT(*) AS Total_Bookings,
ROUND(AVG(total_package_usd), 2) AS Avg_Spending,
SUM(total_package_usd) AS Total_Spending
FROM Booking_Details
GROUP BY travel_class
)

SELECT travel_class, Total_Bookings, Avg_Spending,
ROUND((Total_Spending / (SELECT SUM(Total_Spending) FROM CTE_Class_Spending)) * 100, 2) AS
Spending_Percentage
FROM CTE_Class_Spending
ORDER BY Total_Bookings DESC;
```

Output:

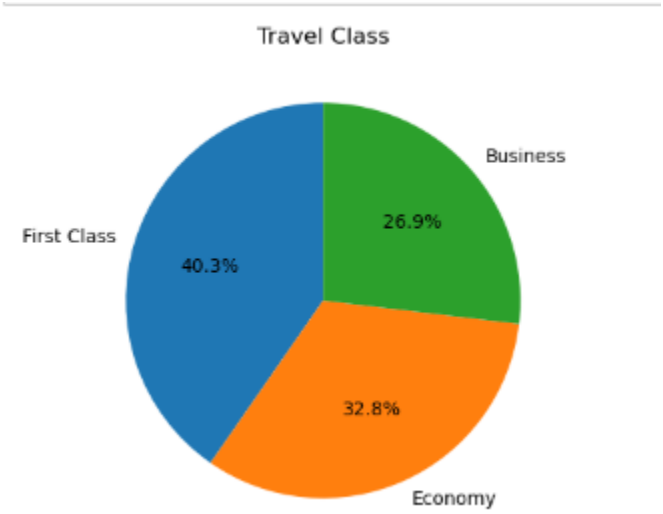
Data Output | Messages | Notifications

	travel_class character varying (50)	total_bookings bigint	avg_spending numeric	spending_percentage numeric
1	First Class	48	5956.25	41.32
2	Economy	39	5453.85	30.74
3	Business	32	6043.75	27.95

Summary:

Utilizing aggregation function to calculate total spending and average expenditure per travel class, and grouping it by the travel class to analyze the number of bookings and respective spending for each class.

Visualization:



Insights:

According to the analysis, travelers tend to choose different types of classes when booking their flights. While First Class tends to have less revenue but more bookings, Business Class has higher spending but fewer bookings, indicating a greater willingness to invest. In an effort to comprehend the factors that influence spending decisions across different types of travel, one may postulate that First-Class spending could be enhanced or that Business Class preference could be exploited to generate more revenue.

2.4 Booking Preference Analysis:

Query:

```
QUERIES USED:

SELECT booking_channel AS Booking_Channel, ROUND(COUNT(booking_id)) AS Total_Bookings,
ROUND(SUM(total_package_usd)) AS Total_Revenue, ROUND(AVG(total_package_usd)) AS
Avg_Revenue_Per_Booking,
ROUND((SUM(total_package_usd) / (SELECT SUM(total_package_usd) FROM Booking_Details)) * 100, 2) AS
Revenue_Contribution_Percentage
FROM Booking_Details
GROUP BY booking_channel
ORDER BY Total_Revenue DESC;
```

Output:

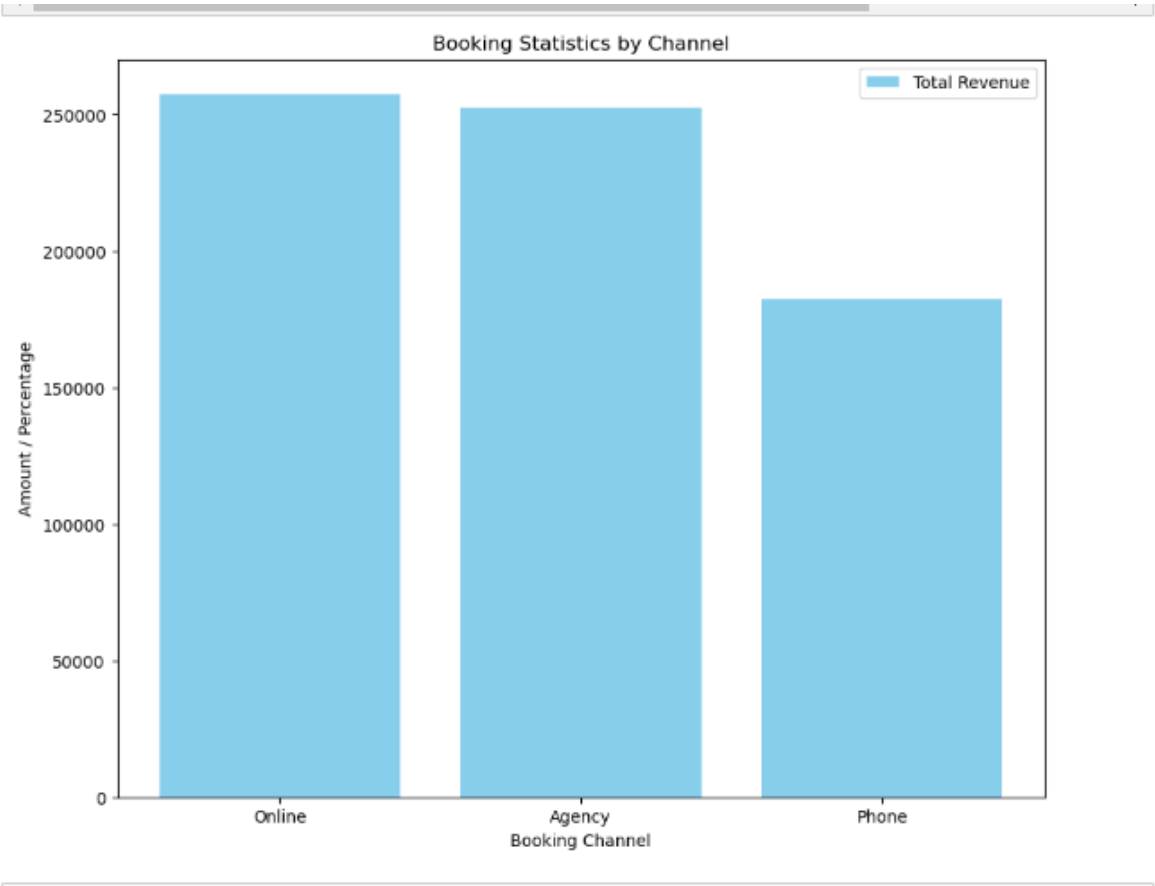
Data Output | Messages | Notifications

	booking_channel character varying (20)	total_bookings double precision	total_revenue numeric	avg_revenue_per_booking numeric	revenue_contribution_percentage numeric
1	Online	45	257200	5716	37.17
2	Agency	43	252400	5870	36.47
3	Phone	31	182400	5884	26.36

Summary:

Analyzing booking data across different channels to ascertain booking volumes and revenues per channel, along with the average revenue per booking and computing the contribution percentage of each channel to the total revenue.

Visualization:



Insights:

Despite having comparable booking numbers, the analysis shows that Online and Agency channels contribute much more to overall revenue. This might be a chance to boost revenue even more by optimizing marketing and incentives for various channels. Revenue might theoretically be amplified by concentrating on targeted promotions or improving the user experience on various platforms. Additional research may reveal ways to increase the entire revenue mix's reliance on phone bookings while keeping their revenue consistent.

3. Seasonal Demand Variation Analysis

3.1 Seasonal Booking Trends Analysis:

Query:

```
QUERIES USED:

SELECT EXTRACT(YEAR FROM booking_date) AS Sale_Year,
    CASE
    WHEN EXTRACT(MONTH FROM booking_date) IN (7, 8, 12) THEN 'Peak Season'
    ELSE 'Non-Peak Season'
END AS season_category,
SUM(total_package_usd) AS total_revenue
FROM Booking_Details
GROUP BY Sale_Year, season_category
ORDER BY Sale_Year, season_category;
```

Output:

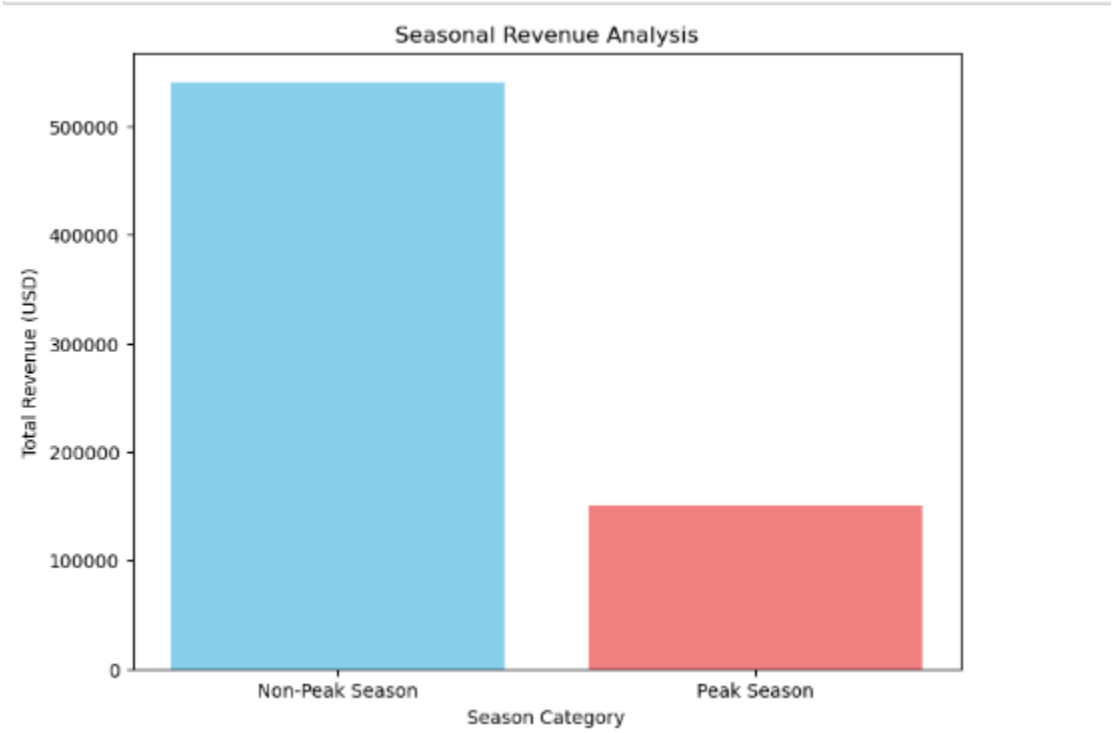
Data Output | Messages | Notifications

	sale_year numeric	season_category text	total_revenue numeric
1	2023	Non-Peak Season	499000
2	2023	Peak Season	193000

Summary:

Utilizing CASE statements to categorize booking dates into Peak Season/Non-Peak Season, to showcase total revenue during these periods and grouping it by year and season category to show revenue trends.

Visualization:



Insights:

The data shows that there are significant year-to-year differences in revenue between peak and off-peak seasons. Profitability outside of normally busy times may be indicated by the high revenue during off-peak seasons. One possible suggestion is to look at ways to generate more revenue during off-peak seasons. This may be through targeted marketing or giving discounts or deals to attract customers.

3.2 Seasonal Revenue Analysis by Customer Premium Status

Query:

```
QUERIES USED:

WITH Seasonal_Booking AS (
SELECT
CASE
     WHEN EXTRACT(MONTH FROM booking_date) IN (7, 8, 12) THEN 'Peak Season'
     ELSE 'Non-Peak Season'
END AS season_category, bd.total_package_usd, CT.premium_member
FROM Booking_Details bd
JOIN Customer CT ON bd.customer_id = CT.customer_id
)
SELECT season_category,
CASE
    WHEN premium_member THEN 'Premium'
    ELSE 'Non-Premium'
END AS customer_status,
SUM(total_package_usd) AS total_revenue
FROM Seasonal_Booking
GROUP BY season_category, customer_status
ORDER BY season_category, total_revenue DESC;
```

Output:

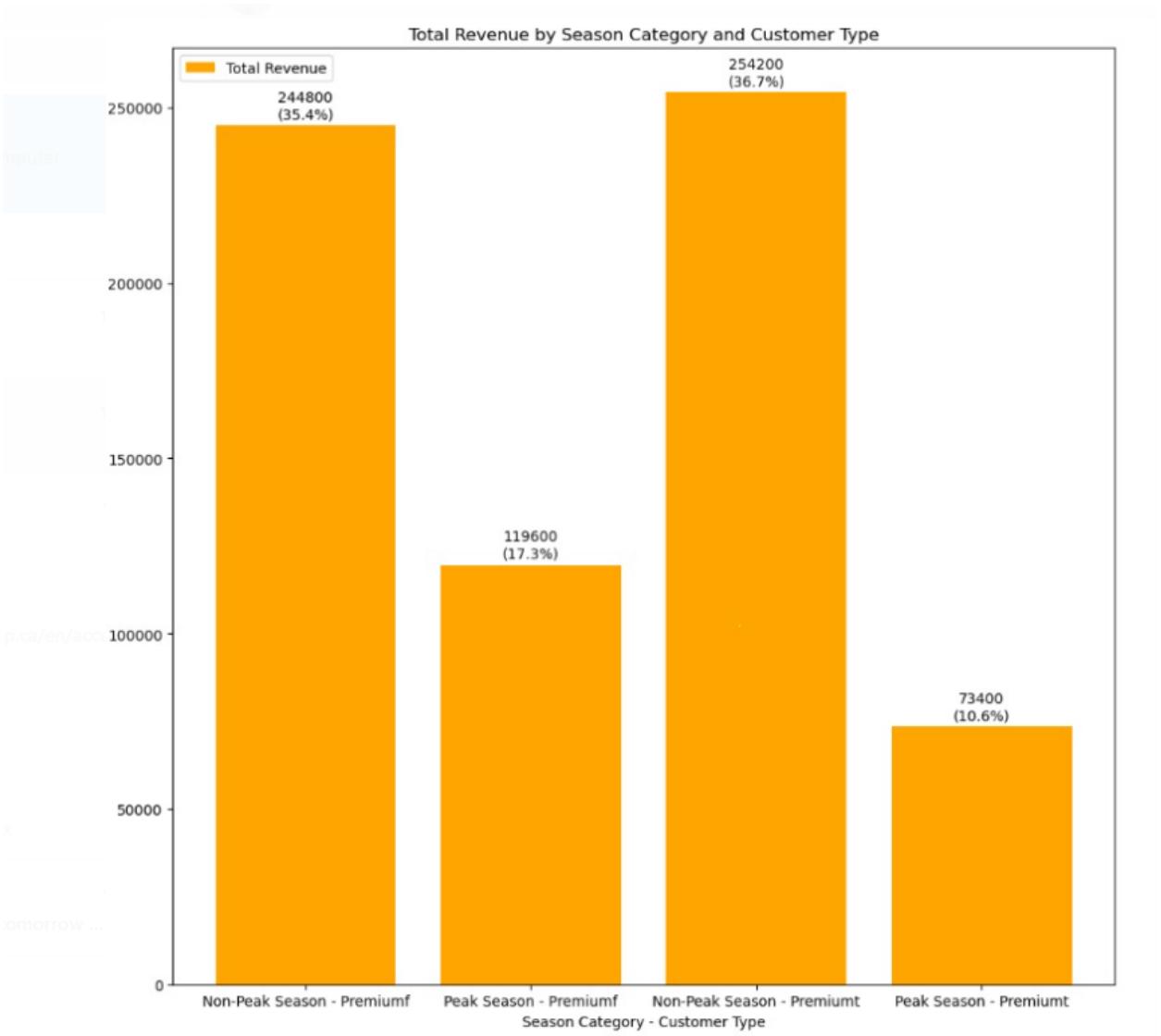
Data Output | Messages | Notifications

	season_category text	customer_status text	total_revenue numeric
1	Non-Peak Season	Premium	254200
2	Non-Peak Season	Non-Premium	244800
3	Peak Season	Non-Premium	119600
4	Peak Season	Premium	73400

Summary:

Categorizing seasonal revenue based on the premium status of customers. Utilizing CASE statements, joins, aggregators, group by clause to derive total revenue by premium status and seasons.

Visualization:



Insights:

When comparing peak and off-peak seasons, the data shows that premium consumers contribute significantly more revenue during off-peak times. Off-peak hours present a chance to plan premium-focused services that might make better use of their purchasing power. Furthermore, by customizing incentives or promotions, it may be possible to increase revenue by engaging non-premium clients during peak seasons.

3.3 Regional Revenue Comparison: Peak and Off-Peak Season Analysis

Query:

```
QUERIES USED:

SELECT region_name,
    COALESCE(SUM(CASE WHEN month_number IN (7, 8, 12) THEN total_revenue END), 0) AS
peak_season_revenue,
    COALESCE(SUM(CASE WHEN month_number NOT IN (7, 8, 12) THEN total_revenue END), 0) AS
off_peak_season_revenue,
    COALESCE(SUM(total_revenue), 0) AS total_combined_revenue
FROM (SELECT r.region_name,EXTRACT(MONTH FROM bd.travel_date) AS month_number,
      SUM(bd.total_package_usd) AS total_revenue
    FROM Booking_Details bd
    JOIN Region r ON bd.region_id = r.region_id
    GROUP BY r.region_name, EXTRACT(MONTH FROM bd.travel_date)
) AS MonthlyRegionRevenue
GROUP BY region_name
ORDER BY total_combined_revenue DESC, region_name;
```

Output:

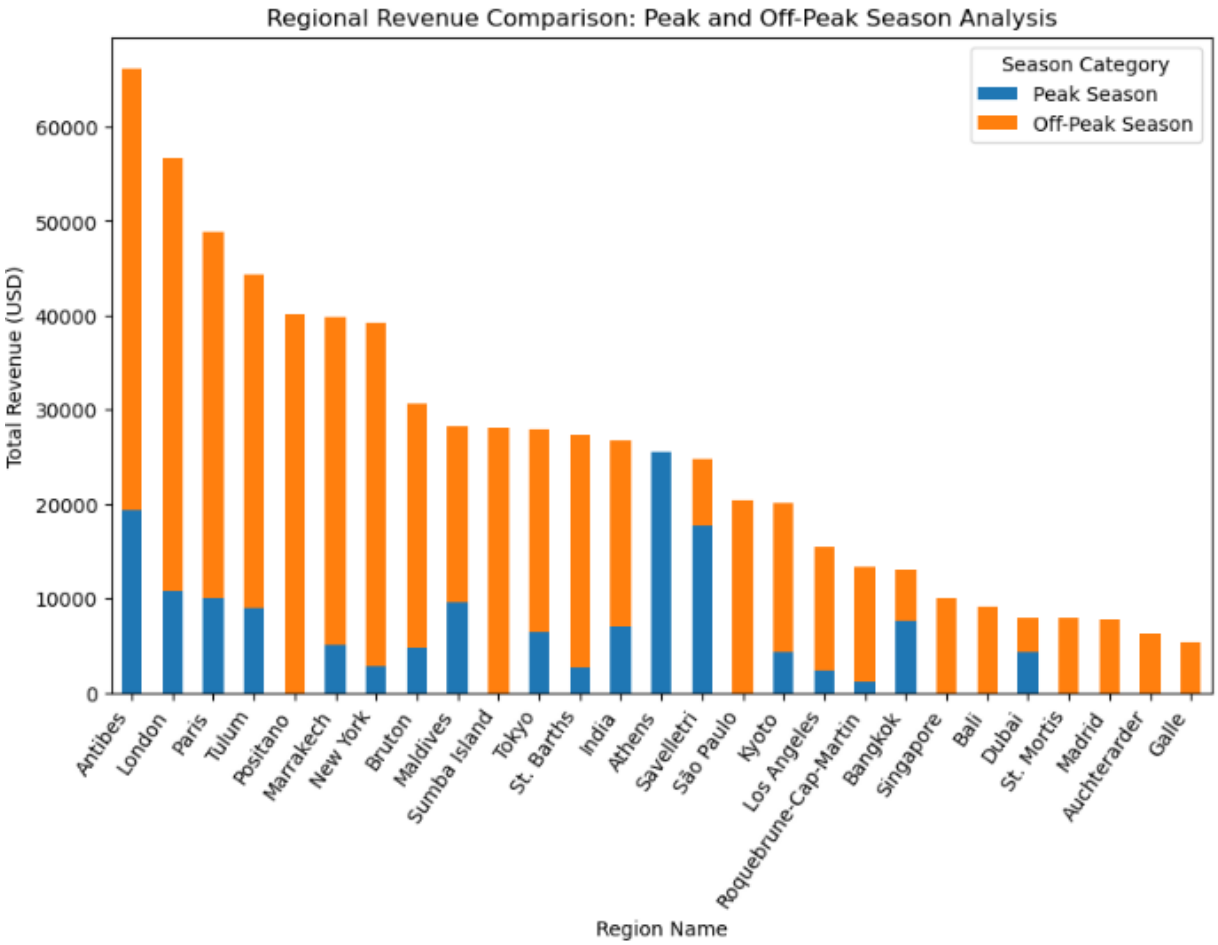
Data Output | Messages | Notifications

	region_name character varying (100)	peak_season_revenue numeric	off_peak_season_revenue numeric	total_combined_revenue numeric
1	Antibes	19400	46800	66200
2	London	10800	45900	56700
3	Paris	10000	38800	48800
4	Tulum	9000	35400	44400
5	Positano	0	40200	40200
6	Marrakech	5100	34700	39800
7	New York	2800	36500	39300
8	Bruton	4800	25900	30700
9	Maldives	9600	18600	28200
10	Sumba Island	0	28100	28100
11	Tokyo	6500	21500	28000
12	St. Barths	2700	24600	27300
13	India	7000	19800	26800
14	Athens	25500	0	25500
15	Savelletri	17800	7000	24800
16	São Paulo	0	20400	20400
17	Kyoto	4400	15700	20100
18	Los Angeles	2400	13100	15500
19	Roquebrune-Cap-Martin	1200	12200	13400
20	Bangkok	7700	5400	13100
21	Singapore	0	10000	10000
22	Bali	0	9200	9200
23	Dubai	4400	3600	8000
24	St. Mortis	0	8000	8000
25	Madrid	0	7800	7800
26	Auchterarder	0	6300	6300
27	Galle	0	5400	5400

Summary:

Categorizing seasons and analyzing total revenue for each region across both seasonal categories, using Coalesce, aggregation, case and join functions.

Visualization:



Insights:

Revenue trends vary between areas during peak and off-peak seasons. Revenue in Antibes, London, and Paris remains constant even during off-peak hours, suggesting that these cities consistently attract visitors. During off-peak months, unique destinations like Marrakech and Positano tend to do better. This might be because of seasonal events or their specific attractiveness to tourists. Investigating these patterns across the year can help identify the variables that cause these changes, which in turn can inform more precise advertising and better customer service.

3.4 Revenue Breakdown of Top 10 Hotels by Ratings in Peak and Non-Peak Seasons

Query:

```
QUERIES USED:

SELECT Rating.hotel_id, Rating.rating,Rating.hotel_name,Rating.location,
    SUM(CASE WHEN EXTRACT(MONTH FROM bd.travel_date) IN (7, 8, 12) THEN bd.total_package_usd ELSE
0 END) AS Peak_Season_Revenue,
    SUM(CASE WHEN EXTRACT(MONTH FROM bd.travel_date) NOT IN (7, 8, 12) THEN bd.total_package_usd
ELSE 0 END) AS "Non-Peak_Season_Revenue",
    SUM(bd.total_package_usd) AS Total_Revenue
FROM (SELECT HT.hotel_id,HT.rating,HT.hotel_name,HT.location,
ROW_NUMBER() OVER (ORDER BY HT.rating DESC) AS rating_rank
FROM Hotel HT
) AS Rating
JOIN Booking_Details bd ON Rating.hotel_id = bd.hotel_id
WHERE Rating.rating_rank <= 10
GROUP BY Rating.hotel_id, Rating.rating, Rating.hotel_name, Rating.location
ORDER BY Total_Revenue DESC;
```

Output:

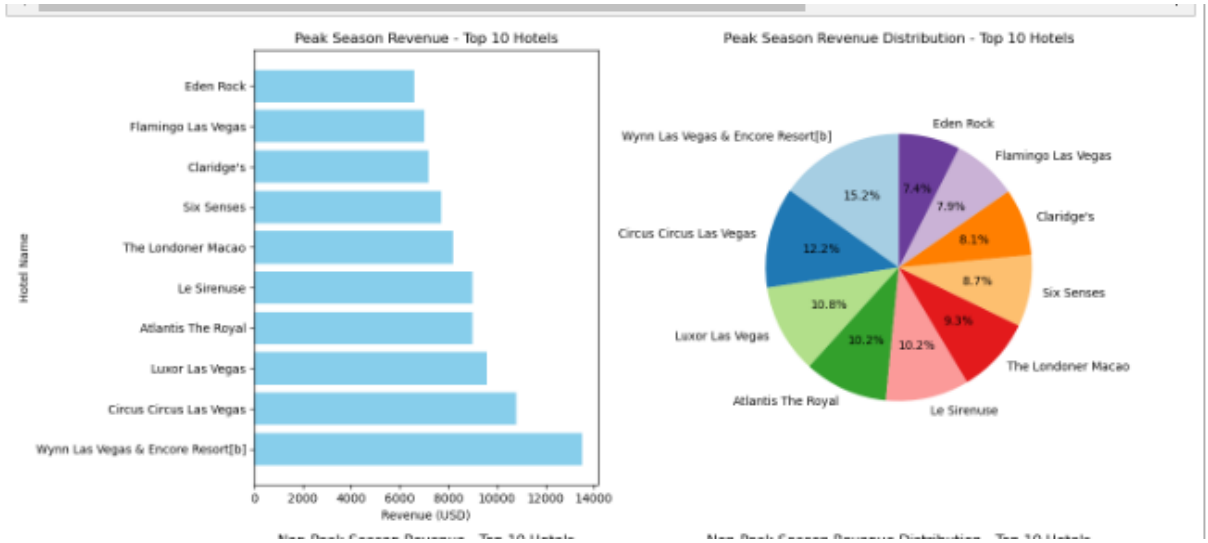
Data Output | Messages | Notifications

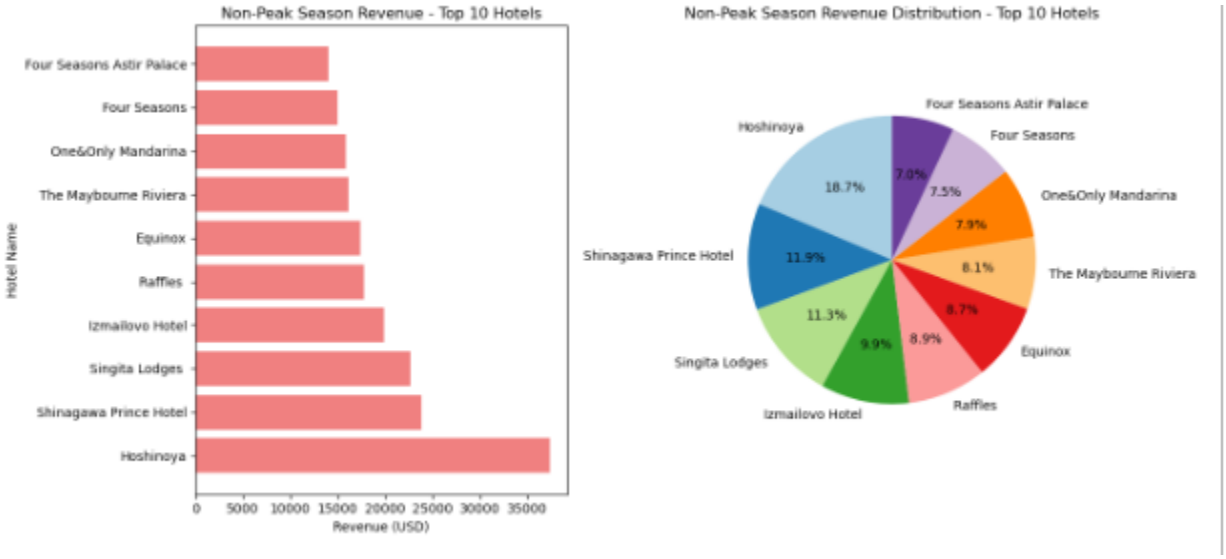
	hotel_id [PK] integer	rating numeric	hotel_name character varying (100)	location character varying (100)	peak_season_revenue numeric	Non-Peak_Season_Revenue numeric	total_revenue numeric
1	540	5	Le Sirenuse	Positano	9000	10000	19000
2	560	5	Passalacqua	Lake Como	4400	14000	18400
3	513	5	Equinox	New York	0	17400	17400
4	573	4.7	Sheraton Grand Macao	Cotai	0	13600	13600
5	550	4.7	Capella	Bangkok	0	8000	8000
6	580	4.7	Flamingo Las Vegas	Las Vegas	7000	0	7000
7	521	4.7	Desa Potato Head	Bali	0	6500	6500
8	563	5	MGM Grand Las Vegas	Las Vegas	5600	0	5600
9	530	4.7	La Réserve	Paris	0	5400	5400
10	531	5	Park Hyatt	Kyoto	0	2100	2100

Summary:

Fetching top 10 hotels based on their ratings, providing revenue breakdown for each hotel across both peak and non-peak seasons.

Visualization:





Insights:

During peak and off-peak seasons, top-rated hotels show different patterns of revenue. Equinox and Sheraton Grand Macao do very well at off-peak times, although Le Sirenuse and Passalacqua routinely generate more revenue. Gaining insight into these behaviors might lead to the development of seasonally-specific marketing tactics, which in turn could increase revenue.

4. Geographical Revenue Analysis:

4.1 Top Regions by Revenue for Each Quarter

Query:

```
QUERIES USED:

SELECT travel_quarter,region_id,region_name,total_revenue
FROM (SELECT Reg.region_id,Reg.region_name,EXTRACT(QUARTER FROM BD.travel_date) AS travel_quarter,
SUM(BD.total_package_usd) AS total_revenue,
ROW_NUMBER() OVER(PARTITION BY EXTRACT(QUARTER FROM BD.travel_date) ORDER BY
SUM(BD.total_package_usd) DESC) AS revenue_rank
FROM Region Reg
JOIN Booking_Details BD ON Reg.region_id = BD.region_id
WHERE EXTRACT(YEAR FROM BD.travel_date) = 2023
GROUP BY Reg.region_id, Reg.region_name, EXTRACT(QUARTER FROM BD.travel_date)
) AS Ranked_Regions
WHERE revenue_rank <= 3
ORDER BY travel_quarter, revenue_rank;
```

Output:

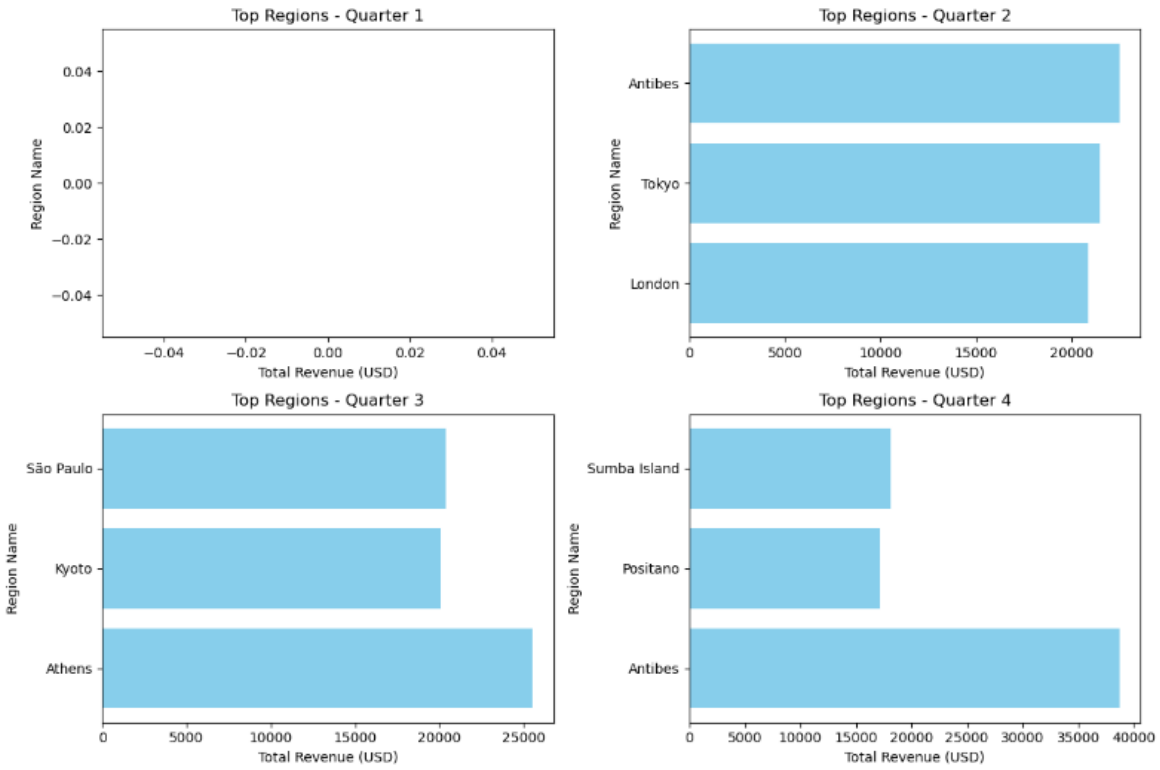
Data Output | Messages | Notifications

	travel_quarter numeric	region_id [PK] integer	region_name character varying (100)	total_revenue numeric
1	2	8905	Antibes	22500
2	2	8899	Tokyo	21500
3	2	8890	London	20900
4	3	8903	Athens	25500
5	3	8909	São Paulo	20400
6	3	8907	Kyoto	20100
7	4	8905	Antibes	38700
8	4	8917	Sumba Island	18100
9	4	8914	Positano	17200

Summary:

Using ranking functions to categorize and identify the top revenue-generating regions for each quarter and grouping data by quarter and region to determine the regions contributing the most revenue in different quarters.

Visualization



Insights:

The analysis identifies the locations that generate the most revenue for each quarter of 2023. Antibes, Tokyo, and London are the top performers in the second quarter, reflecting a significant surge in travel demand. Athens, São Paulo, and Kyoto have been identified as the most revenue generating cities in the third quarter, showcasing diverse regional preferences. During the fourth quarter, Antibes, Sumba Island, and Positano continue to demonstrate robust performance, indicating enduring appeal and possible seasonal patterns.

4.2 Top Airlines Contribution towards Regional Revenue

Query:

```
QUERIES USED:

SELECT reg.region_id AS reg_id, reg.region_name AS reg_name,FL.airline AS airline,
SUM(BD.total_package_usd) AS total_revenue
FROM Region Reg
left join Booking_Details BD ON reg.region_id = BD.region_id
left join Flight FL ON BD.flight_id = FL.flight_id
WHERE BD.flight_id IS NOT NULL AND FL.airline IS NOT NULL
GROUP BY reg.region_id, reg.region_name, FL.airline
HAVING SUM(BD.total_package_usd) = (SELECT SUM(BD.total_package_usd)
FROM Booking_Details bd_inner
WHERE BD_inner.flight_id IS NOT NULl AND BD_inner.region_id = reg.region_id
GROUP BY BD_inner.flight_id
ORDER BY SUM(BD_inner.total_package_usd) DESC
LIMIT 1)
ORDER BY reg.region_id, total_revenue DESC;
```

Output

Query Query History

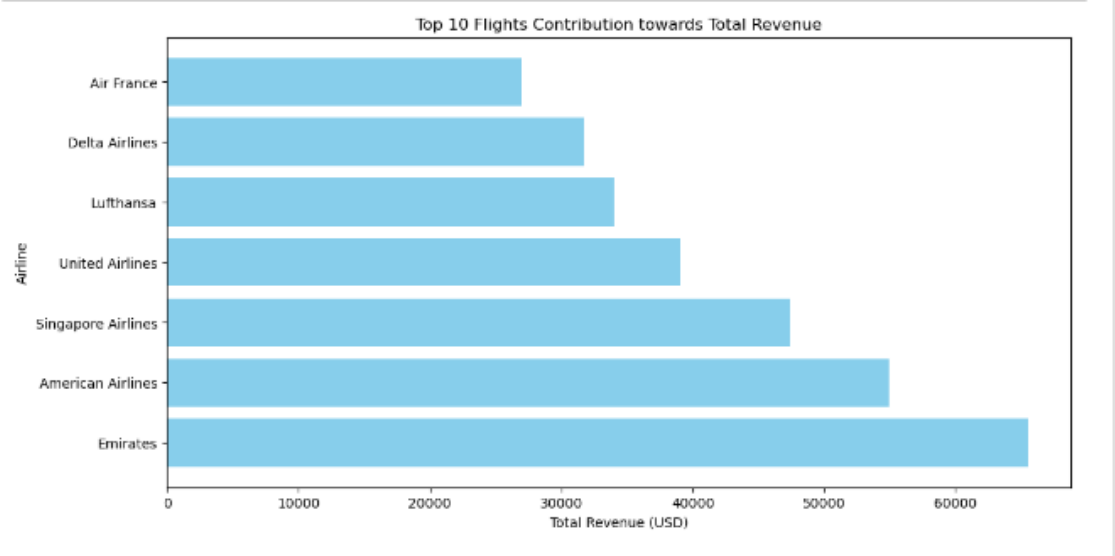
Data Output Messages Notifications

	reg_id integer	reg_name character varying (100)	airline text	total_revenue numeric
1	8888	New York	Emirates	12100
2	8888	New York	Delta Airlines	4200
3	8889	Los Angeles	Singapore Airlines	9100
4	8889	Los Angeles	British Airways	4000
5	8889	Los Angeles	American Airlines	2400
6	8890	London	United Airlines	11000
7	8890	London	Singapore Airlines	8700
8	8890	London	American Airlines	7800
9	8890	London	Qantas	6400
10	8891	Paris	Qantas	9100
11	8891	Paris	Singapore Airlines	1800
12	8891	Paris	Air France	700
13	8892	London	United Airlines	8100
14	8892	London	Singapore Airlines	6500
15	8892	London	American Airlines	4200
16	8892	London	British Airways	4000
17	8893	India	United Airlines	17700
18	8893	India	Emirates	9100
19	8894	Dubai	Singapore Airlines	4400
20	8894	Dubai	Qantas	3600
21	8895	St. Mortis	Lufthansa	6000
22	8895	St. Mortis	Qantas	2000
23	8896	Bangkok	Emirates	7700
24	8896	Bangkok	Delta Airlines	5400
25	8897	St. Barths	Emirates	11000
26	8897	St. Barths	Singapore Airlines	7000
27	8897	St. Barths	American Airlines	6600
28	8897	St. Barths	British Airways	2700
29	8898	Bali	Singapore Airlines	8000

Summary:

Evaluating revenue contribution of various airlines in each region and identifying the top three airlines, based on total booking values, using joins and aggregations and group by clause.

Visualization



Insights:

The analysis highlights the airlines that make the most significant contributions to regional revenues. Emirates distinguishes itself in several places, such as New York and Positano, by making substantial financial donations. Singapore Airlines and United Airlines have significant influences in several areas, demonstrating their extensive market penetration. The distribution indicates a wide range of airline preferences in different locations, emphasizing the many options available in profitable sectors.

5. Revenue impact based on flight service

5.1 Flight Meal Inclusion Revenue Analysis

Query:

```
QUERIES USED:

SELECT FL.meal_inclusion, COUNT(BD.booking_id) AS Total_Bookings,
ROUND(SUM(BD.total_package_usd), 2) AS Total_Revenue,
ROUND(AVG(BD.total_package_usd), 2) AS Avg_Revenue_Per_Booking,
ROUND((SUM(BD.total_package_usd) / (SELECT SUM(total_package_usd) FROM Booking_Details)) * 100, 2)
AS Revenue_Contribution_Percentage
FROM Booking_Details BD
join Flight FL ON BD.flight_id = FL.flight_id
GROUP BY FL.meal_inclusion
ORDER BY Total_Revenue DESC;
```

Output:

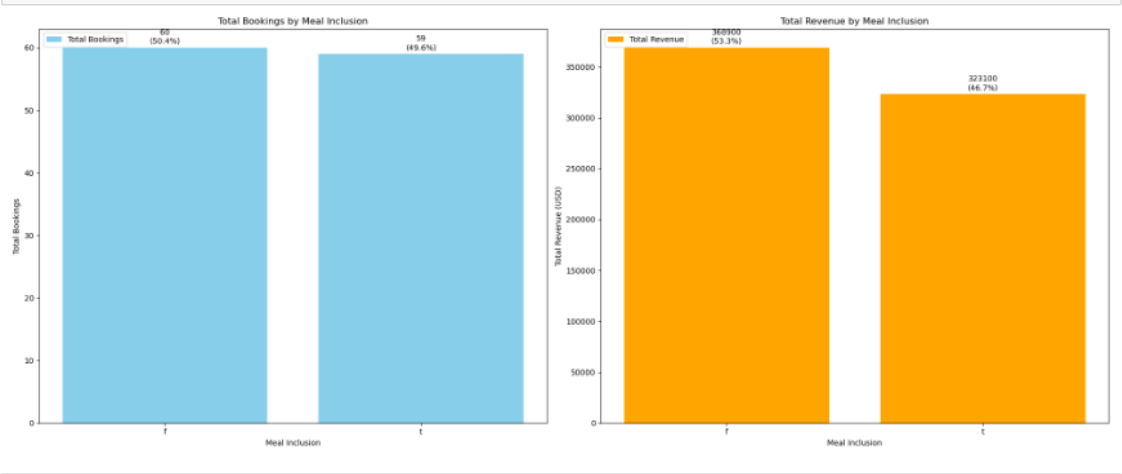
Data Output | Messages | Notifications

	meal_inclusion boolean	total_bookings bigint	total_revenue numeric	avg_revenue_per_booking numeric	revenue_contribution_percentage numeric
1	false	60	368900.00	6148.33	53.31
2	true	59	323100.00	5476.27	46.69

Summary:

Utilizing aggregators to understand the impact of in-flight meal service on bookings by categorizing booking data based on meal options, and estimating total bookings, overall revenue, and average revenue per booking for each meal inclusion type.

Visualization:



Insights:

This analysis examines the revenue and booking patterns of flights that include meals against those that do not. Flights that do not include meals generate more overall revenue, although having fewer reservations, indicating a better revenue per ticket. This indicates a possible inclination towards flights that do not include meals, which might affect how revenue is distributed and harm overall profitability.

6. Customer Revenue Segmentation

6.1Customer Revenue Segmentation Analysis

Query:

```
QUERIES USED:

SELECT
CASE
     WHEN C.premium_member THEN 'Premium Member'
     ELSE 'Non-Premium Member'
END AS Membership_Type,COUNT(*) AS Total_Customers,
ROUND((COUNT(*) * 100.0) / (SELECT COUNT(*) FROM Customer), 2) AS Percentage_of_Total,
ROUND(SUM(BD.total_package_usd), 2) AS Total_Revenue,
ROUND(AVG(BD.total_package_usd), 2) AS Avg_Revenue_Per_Customer
FROM Customer C
LEFT JOIN Booking_Details BD ON C.customer_id = BD.customer_id
GROUP BY Membership_Type;
```

Output:

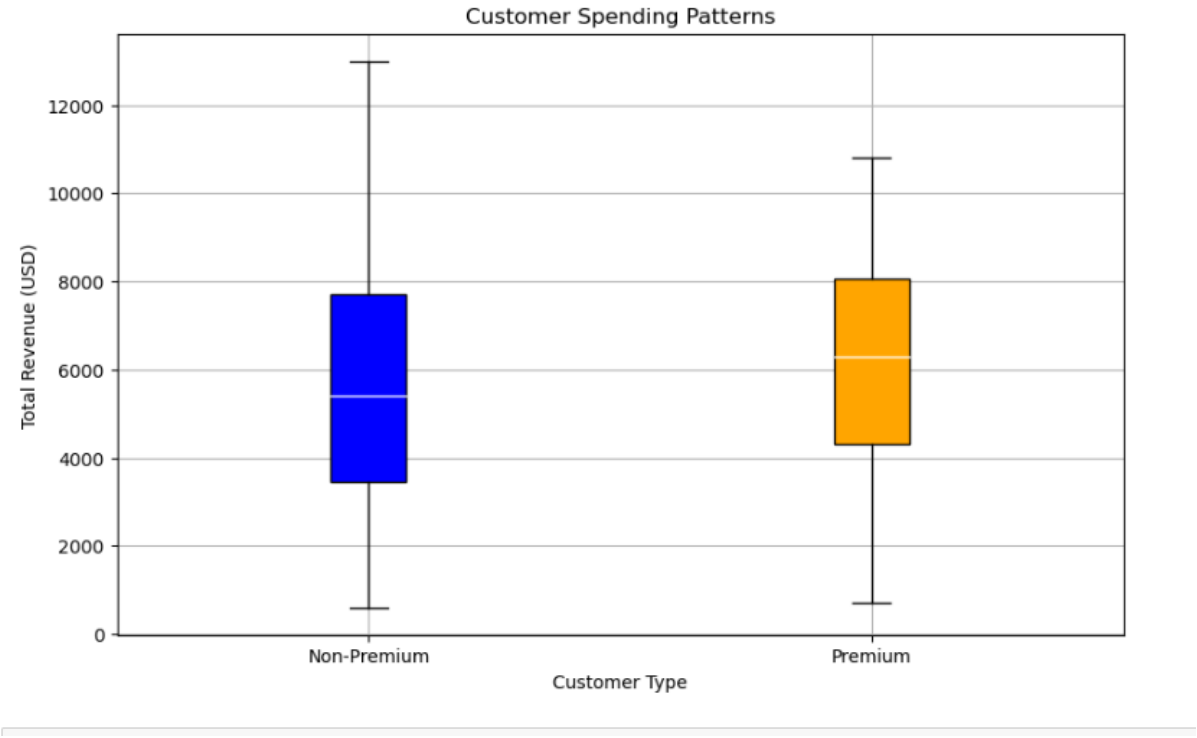
Data Output | Messages | Notifications

	membership_type text	total_customers bigint	percentage_of_total numeric	total_revenue numeric	avg_revenue_per_customer numeric
1	Premium Member	57	43.85	327600.00	5956.36
2	Non-Premium Member	73	56.15	364400.00	5693.75

Summary:

Categorizing customers into "Premium" and "Non-Premium" groups, calculating the total number of customers in each category, their percentage in relation to the total customer base, total revenue generated by each group, and the average revenue per customer in both segments.

Visualization:



Insights:

This analysis categorizes customers into two segments: premium and non-premium. It reveals that although non-premium customers outnumber premium customers, the latter contribute significantly more to the overall revenue. This suggests the need to prioritize efforts in retaining and engaging premium clientele to boost revenue.

Recommendations

Implement data-driven marketing strategies: Utilize consumer information to create focused marketing campaigns, customizing offers based on distinct demographics and interests.

Enhance Regional Strategies: Allocate resources and direct marketing strategies towards locations with strong revenue potential, to maximize profitability.

Execute seasonal marketing initiatives: Devise and implement seasonal marketing campaigns and incentives to coincide with known fluctuations in demand, with the goal of increasing revenue during periods of low customer activity.

Enhance Risk Management: Consistently evaluate risk variables derived from client segments and geographical studies to enhance risk management techniques.

Improve Customer Interaction: Create customized services and experiences by categorizing customers into segments, so promoting better ties and increasing loyalty.

Ongoing surveillance and adjustment: Continuously observe patterns in revenue and consumer actions, adjusting tactics to changing market conditions for long-term success.

Goals of the project

The overarching goal of the project is to optimize revenue streams and operational efficiency for Terrific Travel Agency. By analyzing customer behavior, revenue distribution across services and regions, cost efficiency, risk assessment, and seasonal demand variations within the travel industry, the agency aims to leverage data insights. This comprehensive approach seeks to tailor offerings, refine strategies, and enhance customer experiences, fostering sustainable growth and a competitive advantage in the market.

Table-Specific Goals and Objectives:

Booking Details:

- **Goal**: Analyze booking trends, geographical revenue distribution, seasonal fluctuations, and conduct comparative revenue analysis for different services and regions.
- **Objective**: Serve as the central repository for booking details, covering booking and travel dates, hotel and flight IDs, travel class, passenger count, and financial information.

Customer Table:

- **Goal**: Understand customer spending patterns, preferences, and segment customers based on booking frequency for targeted marketing strategies.
- **Objective**: Capture and manage detailed information about customers, including personal details, contact information, and premium membership status.

Flight Table:

- **Goal**: Evaluate revenue impact and trends associated with different flights, airlines, and departure/arrival airport codes.
- **Objective**: Record details related to flights, including flight numbers, airlines, departure, and arrival airport codes, to support flight booking services.

Hotel Table:

- **Goal**: Examine revenue patterns in key hotel destinations, analyze hotel ratings, and understand the impact of hotel-related factors on overall revenue.
- **Objective**: Store information about hotels, such as names, locations, ratings, and types, to facilitate booking and analysis of accommodation services.

Region Table:

- **Goal**: Assess regional revenue contribution, analyze seasonal trends, and understand geographical distribution patterns to optimize marketing efforts.
- **Objective**: Define and organize geographical regions, cities, and countries, including associated airport codes for efficient categorization.