Problem Statement

- Goal: A business manager of a chain of hotel would like to request help from the data analytics team to develop a dashboard to analyze & visualize hotel booking data

- The business manager is interested if there's any patterns/ seasonality with respect to:
    1. Guests
    2. Revenue

Requirement: Build a Dashboard using PowerBI to provide at-a-glance information about features that are relevant to revenue

Dataset Background

- The dataset used in this analysis can be found in my GitHub

  Repository:

  - Github Repository Page

- This data set contains booking information for a city hotel and a resort hotel

- In the following slide, further elaboration into the fieldname, as well as the definition of each field will be shown.

Dataset glossary in final SQL query [1/4]

Original Columns Present in Dataset

Field Name	Description
Hotel	*Hotel (H1 = Resort Hotel or H2 = City Hotel)*
is_canceled	*Value indicating if the booking was canceled (1) or not (0)*
lead_time	*Number of days that elapsed between the entering date of the booking into the PMS and the arrival date*
Arrival_date_year	*Year of arrival date*
Arrival_date_month	*Month of arrival date*
Arrival_date_week_number	*Week number of year for arrival date*
Arrival_date_day_of_month	*Day of arrival date*
Stays_in_weekend_nights	*Number of weekend nights (Saturday or Sunday) the guest stayed or booked to stay at the hotel*
Stays_in_week_nights	*Number of week nights (Monday to Friday) the guest stayed or booked to stay at the hotel*
Adults	*Number of adults*
Children	*Number of children*
Babies	*Number of babies*
Cost	*Cost incurred by the hotel company related to the associated meal package provided to the customer*
Meal	*Type of meal booked. Categories are presented in standard hospitality meal packages: Undefined/SC – no meal package; BB – Bed & Breakfast; HB – Half board (breakfast and one other meal – usually dinner); FB – Full board (breakfast, lunch and dinner)*
country	*Country of origin. Categories are represented in the ISO 3155–3:2013 format*

Dataset glossary in final SQL query [2/4]

Original Columns Present in Dataset

Field Name	Description
Market_segment	Market segment designation. In categories, the term “TA” means “Travel Agents” and “TO” means “Tour Operators”
Distribution_channel	Booking distribution channel. The term “TA” means “Travel Agents” and “TO” means “Tour Operators”
Is_repeated_guest	Value indicating if the booking name was from a repeated guest (1) or not (0)
Previous_cancellations	Number of previous bookings that were cancelled by the customer prior to the current booking
Previous_bookings_not_cancelled	Number of previous bookings not cancelled by the customer prior to the current booking
Reserved_room_type	Code of room type reserved. Code is presented instead of designation for anonymity reasons.
Assigned_room_type	Code for the type of room assigned to the booking. Sometimes the assigned room type differs from the reserved room type due to hotel operation reasons (e.g. overbooking) or by customer request. Code is presented instead of designation for anonymity reasons.
Booking_changes	Number of changes/amendments made to the booking from the moment the booking was entered on the PMS until the moment of check-in or cancellation
Deposit_type	Indication on if the customer made a deposit to guarantee the booking. This variable can assume three categories: No Deposit – no deposit was made; Non Refund – a deposit was made in the value of the total stay cost; Refundable – a deposit was made with a value under the total cost of stay.
Agent	ID of the travel agency that made the booking
Company	ID of the company/entity that made the booking or responsible for paying the booking. ID is presented instead of designation for anonymity reasons

Dataset glossary in final SQL query [3/4]

Original Columns Present in Dataset

Field Name	Description
days_in_waiting_list	Number ofdays thebooking was inthe waiting list beforeit was confirmed to thecustomer
Customer_type	Type of booking, assuming one of four categories: Contract - when the bookinghas anallotment or other type of contract associated to it; Group – when the booking is associated to a group; Transient – when the booking is not part of a group or contract, and is not associated to other transient booking; Transient-party – when the booking is transient, but is associated to at least other transient booking
Adr	Average Daily Rate as defined by dividing the sum of all lodging transactions by the total number of staying nights
Required_car_parking_spaces	Number of car parking spaces required by the customer
total_of_special_requests	Number of special requests made by the customer (e.g. twin bed or high floor)
reservation_status	Reservation last status, assuming one of three categories: Canceled – booking was canceled by the customer; Check-Out – customer has checked in but already departed; No-Show – customer did not check-in and did inform the hotel of the reason why
Reservation_status_date	Date at which the last status was set. This variable can be used in conjunction with the ReservationStatus to understand when was the booking canceled or when did the customer checked-out of the hotel
Discount	*Float value ranging from 0 to 1. Contains the associated discount provided to the customer belonging to the market_segment*
Market_segment	*Type of market_segment that the customer belongs to. Can be classified as:* *Undefined, Direct, Groups, Corporate, Aviation, Offline TA/TO, Online TA, Complementary*

Dataset glossary in final SQL query [4/4]

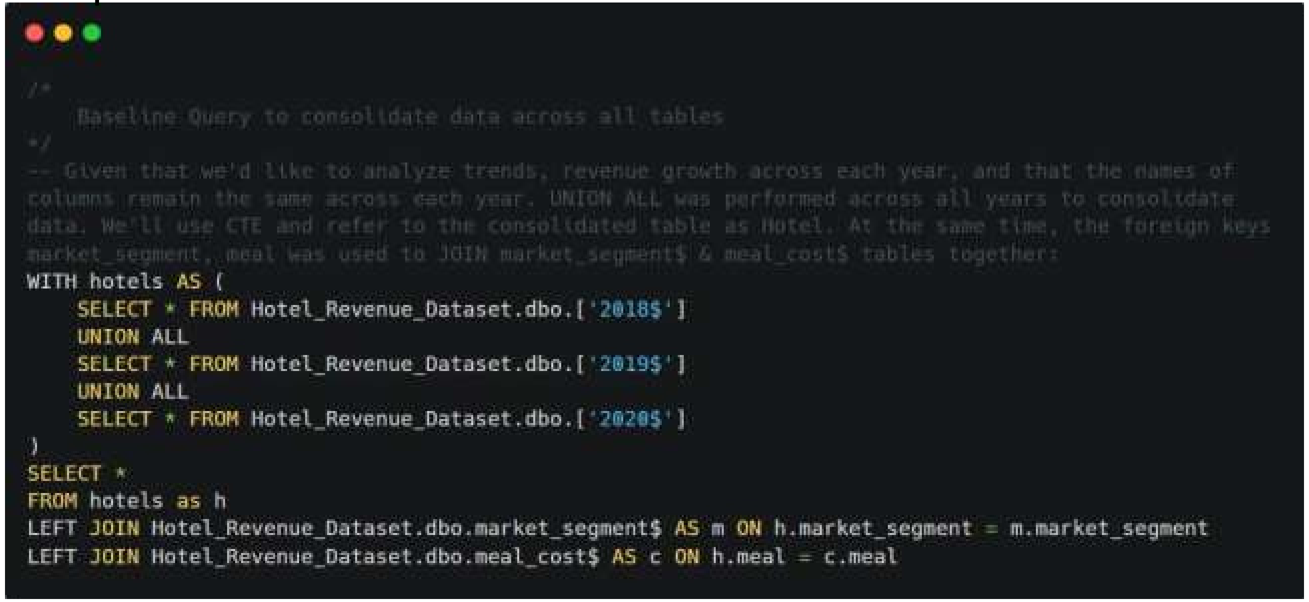
Self-Calculated Columns

Field Name	Description
Revenue	*Self-Calculated column based on the factors:* • *Stays_in_week_nights* • *Stays_in_weekend_nights* • *ADR • Cost* *• Discount*
Guest_type	*Self-Calculated column based on the factors:* *• # of adults* *• # of children* *• # of babies* *To categorize guest type into 4 categories: Single, Couple, Group, Family*
Region	*Self-Calculated column based on the guest's country code, which is represented in the ISO 3155–3:2013 format*
Seasons	*Self-Calculated column based on the 'arrival_date_month' of guests*
Day_of_week	*Self-Calculated column based on the 'arrival_date_day_of_month', 'arrival_date_month', 'arrival_date_year' of guests*

Dataset Preparation in SQL [1/7]

Creation of Baseline Query

- Raw .csvfile is ingestedinto Microsoft SQL Server ManagementStudio (SSMS) and SQL queries were iteratively built upon the baseline query below to obtain the final query to import into SSMS
- The baseline query utilizes UNION ALL, CTE & LEFT JOINs to aggregate all sheets within the excel spreadsheet



```sql
/*
    Baseline Query to consolidate data across all tables
*/
-- Given that we'd like to analyze trends, revenue growth across each year, and that the names of
columns remain the same across each year. UNION ALL was performed across all years to consolidate
data. We'll use CTE and refer to the consolidated table as Hotel. At the same time, the foreign keys
market_segment, meal was used to JOIN market_segment$ & meal_cost$ tables together:
WITH hotels AS (
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2018$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2019$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2020$']
)
SELECT *
FROM hotels as h
LEFT JOIN Hotel_Revenue_Dataset.dbo.market_segment$ AS m ON h.market_segment = m.market_segment
LEFT JOIN Hotel_Revenue_Dataset.dbo.meal_cost$ AS c ON h.meal = c.meal
```

- Following which, SQL queries were iteratively built upon for the creation of revenue & categorical columns

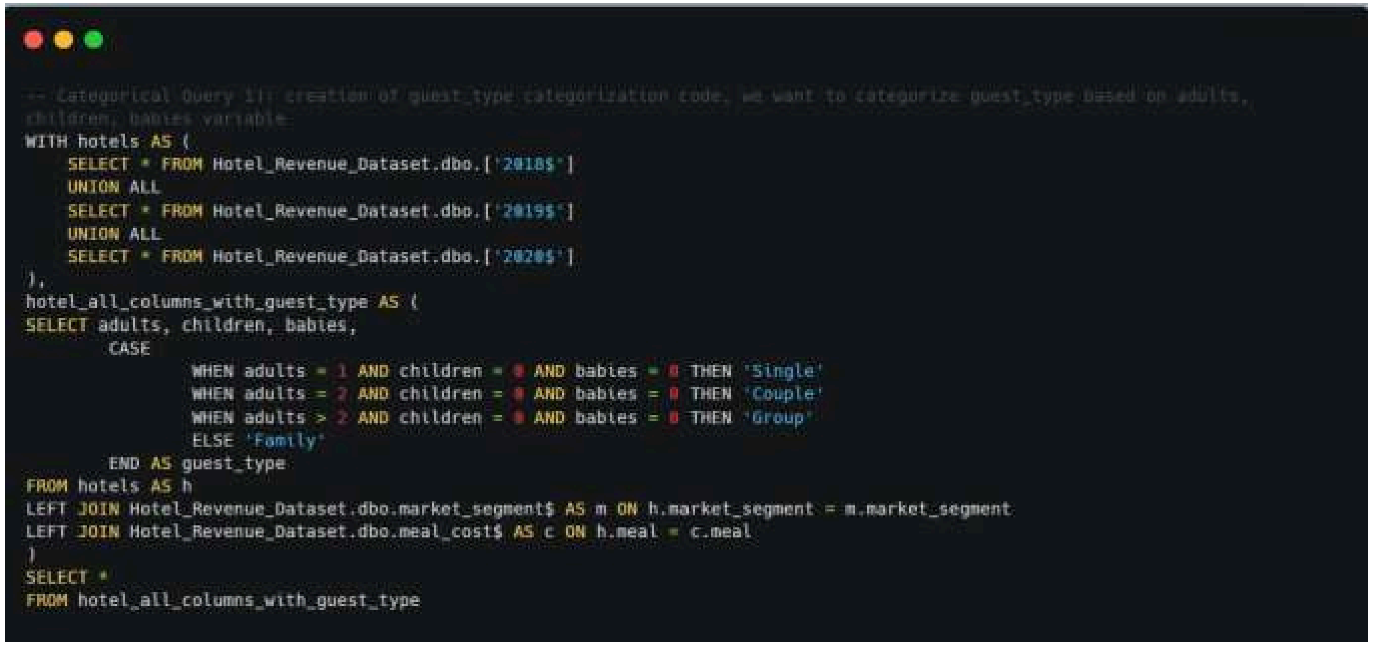
Dataset Preparation in SQL [2/7]

Creation of ‘guest_type’ column

Oneoftherequirementswas toinvestigateif there’sany guest-relatedseasonality or trends

Noticing that in the dataset, there are ‘adults’, ‘children’ & ‘babies’ information for each booking made

- I want to create a single column that categorizes each booking by the type of guests, they fall into 4 categories
  - Single
  - Couple
  - Group
  - Family



```sql
-- Categorical Query 1): creation of guest_type categorization code, we want to categorize guest_type based on adults,
children, babies variable
WITH hotels AS (
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2018$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2019$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2020$']
),
hotel_all_columns_with_guest_type AS (
SELECT adults, children, babies,
        CASE
                WHEN adults = 1 AND children = 0 AND babies = 0 THEN 'Single'
                WHEN adults = 2 AND children = 0 AND babies = 0 THEN 'Couple'
                WHEN adults > 2 AND children = 0 AND babies = 0 THEN 'Group'
                ELSE 'Family'
        END AS guest_type
FROM hotels AS h
LEFT JOIN Hotel_Revenue_Dataset.dbo.market_segment$ AS m ON h.market_segment = m.market_segment
LEFT JOIN Hotel_Revenue_Dataset.dbo.meal_cost$ AS c ON h.meal = c.meal
)
SELECT *
FROM hotel_all_columns_with_guest_type
```

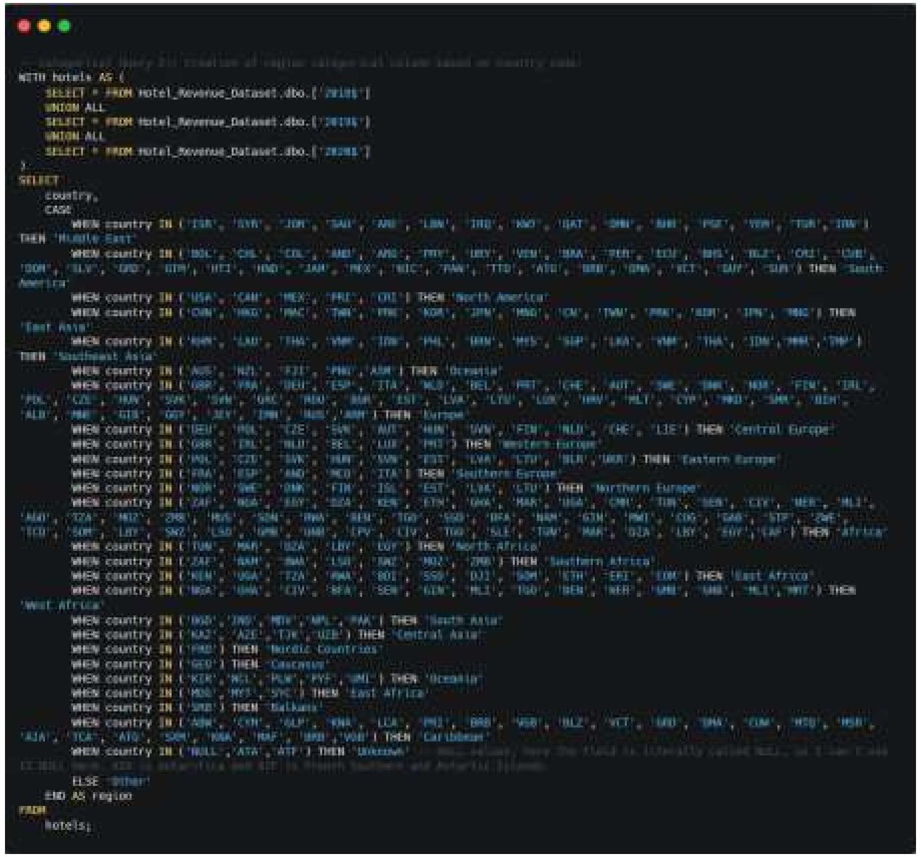
Dataset Preparation in SQL [3/7]

Creation of ‘region’ column

Next, I noticedthat thecountriesinwhichtheguests belong to are storedin athree lettercountry code format.

I’d like to create a categorical column called ‘region’ to see if we can get any insights related to the data with regards to the customer’s region

- The somewhat lengthy SQL code below utilizes multiple CASE WHEN and IN statements to return the corresponding regions of the guests

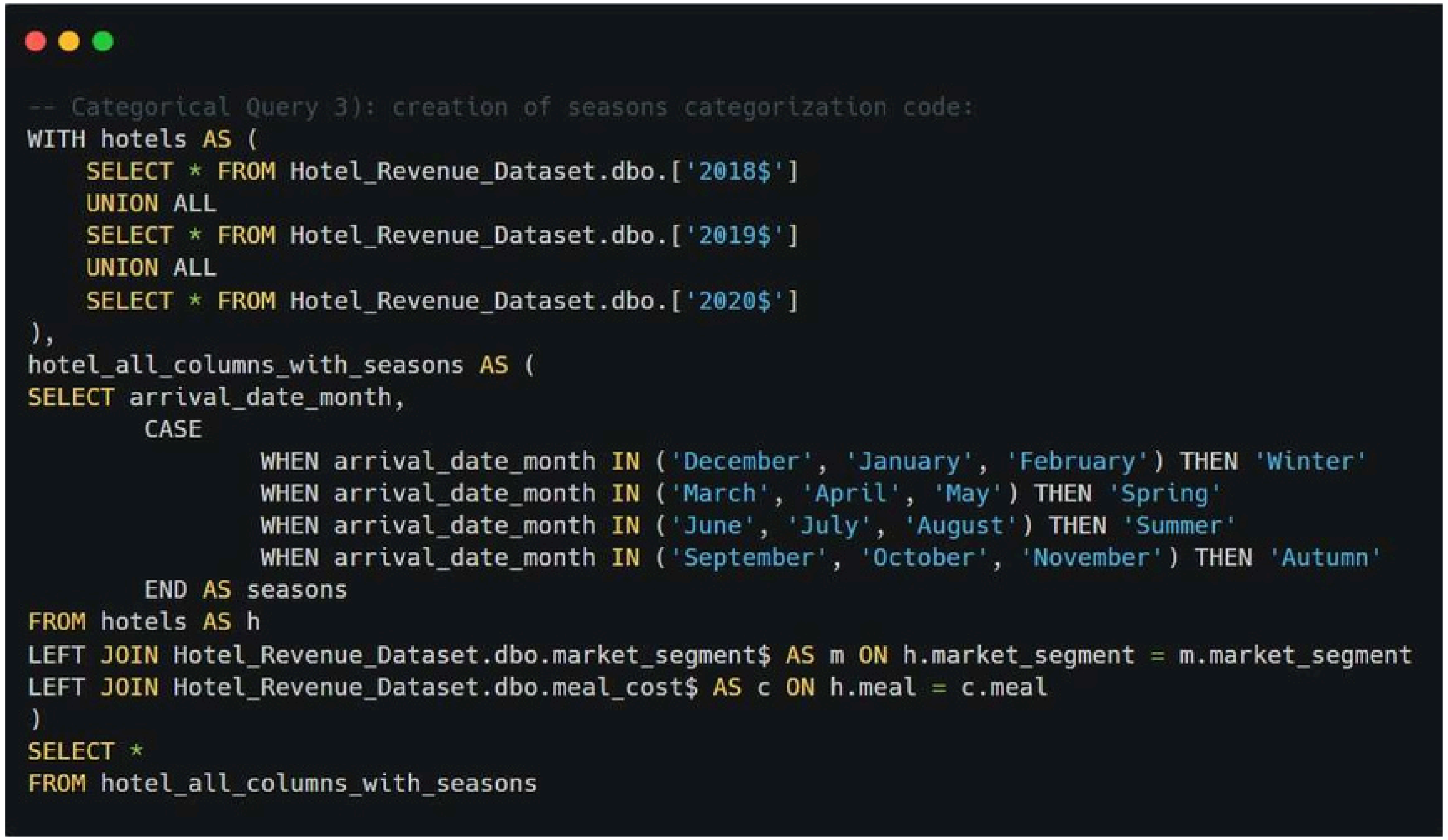


```sql
-- (categorical Query 3): Creation of region categorical column based on country code
WITH hotels AS (
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2018$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2019$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2020$']
)
SELECT
    country,
    CASE
        WHEN country IN ('ISR', 'SYR', 'JOR', 'SAU', 'ARE', 'LBN', 'IRQ', 'KWT', 'QAT', 'OMN', 'BHR', 'PSE', 'YEM', 'TUR','IRN')
THEN 'Middle East'
        WHEN country IN ('BOL', 'CHL', 'COL', 'AND', 'ARG', 'PRY', 'URY', 'VEN', 'BRA', 'PER', 'ECU', 'BHS', 'BLZ', 'CRI', 'CUB',
'DOM', 'SLV', 'GRD', 'GTM', 'HTI', 'HND', 'JAM', 'MEX', 'NIC', 'PAN', 'TTO', 'ATG', 'BRB', 'DMA', 'VCT', 'GUY', 'SUR') THEN 'South
America'
        WHEN country IN ('USA', 'CAN', 'MEX', 'PRI', 'CRI') THEN 'North America'
        WHEN country IN ('CHN', 'HKG', 'MAC', 'TWN', 'PRK', 'KOR', 'JPN', 'MNG', 'CN', 'TWN', 'PRK', 'KOR', 'JPN', 'MNG') THEN
'East Asia'
        WHEN country IN ('KHM', 'LAO', 'THA', 'VNM', 'IDN', 'PHL', 'BRN', 'MYS', 'SGP', 'LKA', 'VNM', 'THA', 'IDN','MMR','TMP')
THEN 'Southeast Asia'
        WHEN country IN ('AUS', 'NZL', 'FJI', 'PNG', 'ASM') THEN 'Oceania'
        WHEN country IN ('GBR', 'FRA', 'DEU', 'ESP', 'ITA', 'NLD', 'BEL', 'PRT', 'CHE', 'AUT', 'SWE', 'DNK', 'NOR', 'FIN', 'IRL',
'POL', 'CZE', 'HUN', 'SVK', 'SVN', 'GRC', 'ROU', 'BGR', 'EST', 'LVA', 'LTU', 'LUX', 'HRV', 'MLT', 'CYP', 'MKD', 'SMR', 'BIH',
'ALB', 'MNE', 'GIB', 'GGY', 'JEY', 'IMN', 'RUS','ARM') THEN 'Europe'
        WHEN country IN ('DEU', 'POL', 'CZE', 'SVK', 'AUT', 'HUN', 'SVN', 'FIN', 'NLD', 'CHE', 'LIE') THEN 'Central Europe'
        WHEN country IN ('GBR', 'IRL', 'NLD', 'BEL', 'LUX', 'PRT') THEN 'Western Europe'
        WHEN country IN ('POL', 'CZE', 'SVK', 'HUN', 'SVN', 'EST', 'LVA', 'LTU', 'BLR','UKR') THEN 'Eastern Europe'
        WHEN country IN ('FRA', 'ESP', 'AND', 'MCO', 'ITA') THEN 'Southern Europe'
        WHEN country IN ('NOR', 'SWE', 'DNK', 'FIN', 'ISL', 'EST', 'LVA', 'LTU') THEN 'Northern Europe'
        WHEN country IN ('ZAF', 'NGA', 'EGY', 'DZA', 'KEN', 'ETH', 'GHA', 'MAR', 'UGA', 'CMR', 'TUN', 'SEN', 'CIV', 'NER', 'MLI',
'AGO', 'TZA', 'MOZ', 'ZMB', 'MUS', 'SDN', 'RWA', 'BEN', 'TGO', 'SSD', 'BFA', 'NAM', 'GIN', 'MWI', 'COG', 'GAB', 'STP', 'ZWE',
'TCD', 'SOM', 'LBY', 'SWZ', 'LSO', 'GMB', 'GNB', 'LBR', 'CPV', 'CIV', 'TGO', 'SLE', 'TUN', 'MAR', 'DZA', 'LBY', 'EGY','CAF') THEN 'Africa'
        WHEN country IN ('TUN', 'MAR', 'DZA', 'LBY', 'EGY') THEN 'North Africa'
        WHEN country IN ('ZAF', 'NAM', 'BWA', 'LSO', 'SWZ', 'MOZ', 'ZMB') THEN 'Southern Africa'
        WHEN country IN ('KEN', 'UGA', 'TZA', 'RWA', 'BDI', 'SSD', 'DJI', 'SOM', 'ETH', 'ERI', 'COM') THEN 'East Africa'
        WHEN country IN ('NGA', 'GHA', 'CIV', 'BFA', 'SEN', 'GIN', 'MLI', 'TGO', 'BEN', 'NER', 'GMB', 'GNB', 'MLI','MRT') THEN
'West Africa'
        WHEN country IN ('BGD','IND','MDV','NPL','PAK') THEN 'South Asia'
        WHEN country IN ('KAZ', 'AZE','TJK','UZB') THEN 'Central Asia'
        WHEN country IN ('FRO') THEN 'Nordic Countries'
        WHEN country IN ('GEO') THEN 'Caucasus'
        WHEN country IN ('KIR','NCL','PLW','PYF','UMI') THEN 'Oceania'
        WHEN country IN ('MDG','MYT','SYC') THEN 'East Africa'
        WHEN country IN ('SRB') THEN 'Balkans'
        WHEN country IN ('ABW', 'CYM', 'GLP', 'KNA', 'LCA', 'PRI', 'BRB', 'VGB', 'BLZ', 'VCT', 'GRD', 'DMA', 'CUW', 'MTQ', 'MSR',
'AIA', 'TCA', 'ATG', 'SXM', 'KNA', 'MAF', 'BRB','VGB') THEN 'Caribbean'
        WHEN country IN ('NULL','ATA','ATF') THEN 'Unknown' -- NULL values; here the field is literally called NULL, so I can't use
IS NULL. Here, ATA is Antarctica and ATF is French Southern and Antarctic Islands
        ELSE 'Other'
    END AS region
FROM
    hotels;
```

Dataset Preparation in SQL [4/7]

Creation of ‘seasons’ column

Giventhatthemonthofarrivalofguestsareprovidedinthedataset,I’dliketocategorizethembyseasons:

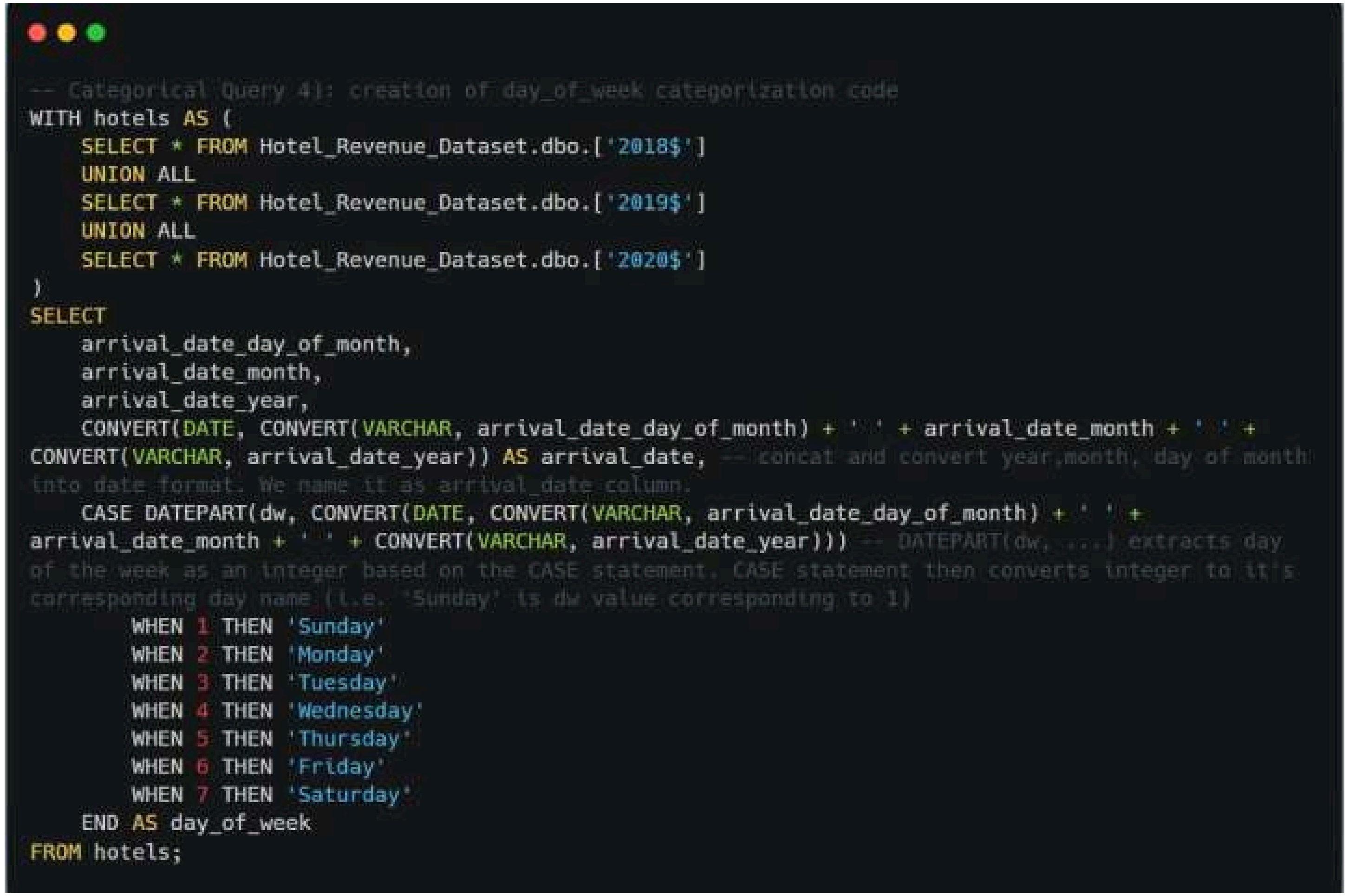


```sql
-- Categorical Query 3): creation of seasons categorization code:
WITH hotels AS (
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2018$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2019$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2020$']
),
hotel_all_columns_with_seasons AS (
SELECT arrival_date_month,
        CASE
                WHEN arrival_date_month IN ('December', 'January', 'February') THEN 'Winter'
                WHEN arrival_date_month IN ('March', 'April', 'May') THEN 'Spring'
                WHEN arrival_date_month IN ('June', 'July', 'August') THEN 'Summer'
                WHEN arrival_date_month IN ('September', 'October', 'November') THEN 'Autumn'
        END AS seasons
FROM hotels AS h
LEFT JOIN Hotel_Revenue_Dataset.dbo.market_segment$ AS m ON h.market_segment = m.market_segment
LEFT JOIN Hotel_Revenue_Dataset.dbo.meal_cost$ AS c ON h.meal = c.meal
)
SELECT *
FROM hotel_all_columns_with_seasons
```

Dataset Preparation in SQL [5/7]

Creation of ‘day_of_week’ column

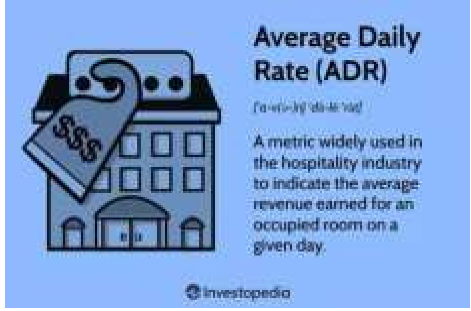
- TheSQLquerybelowutilizesDATEPART&CASEWHENstatements,aswellasconvertingandconcatenating each columns from the arrival date to extractout the corresponding day of theweek inwhich the guests arrive in.



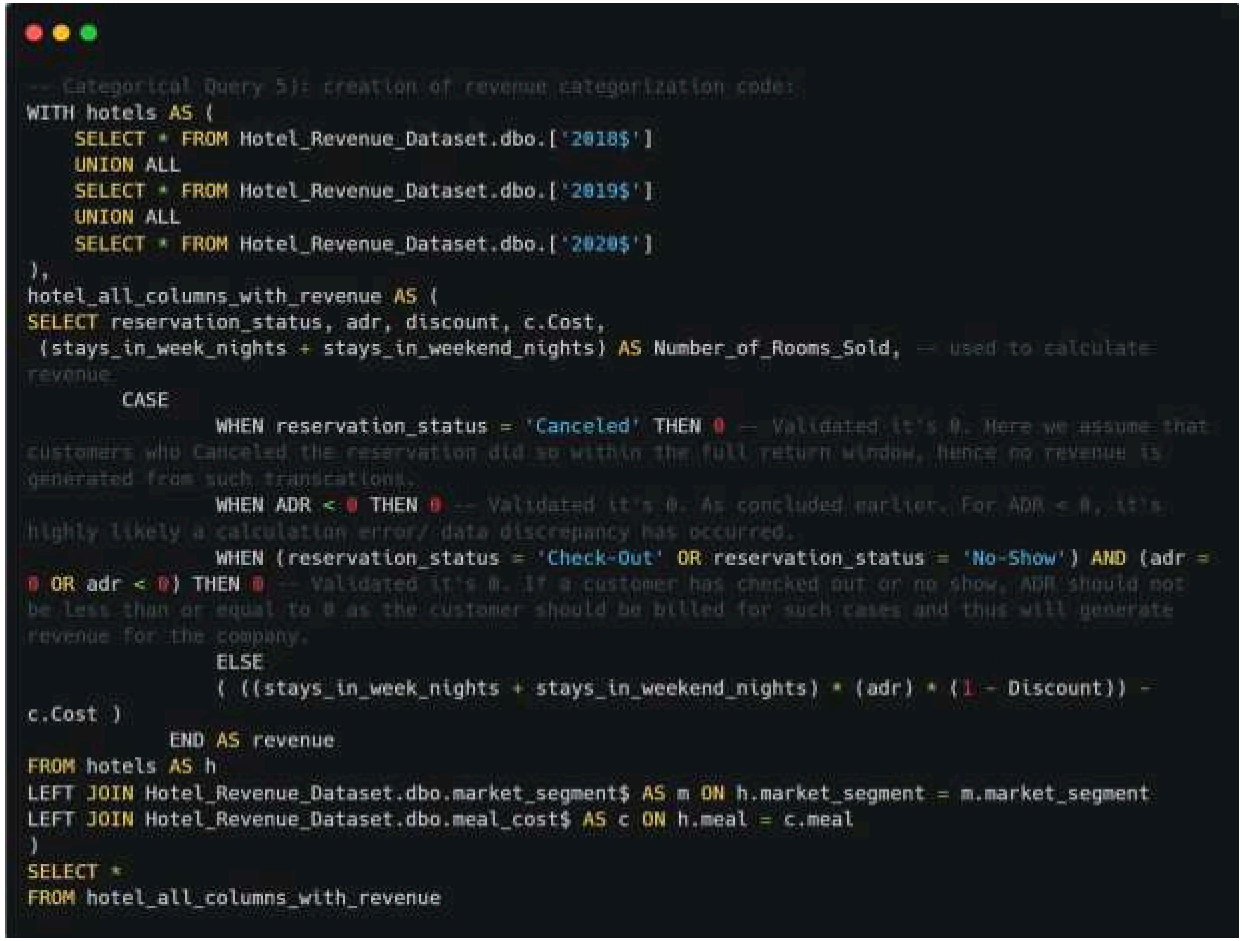
```sql
-- Categorical Query 4): creation of day_of_week categorization code
WITH hotels AS (
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2018$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2019$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2020$']
)
SELECT
    arrival_date_day_of_month,
    arrival_date_month,
    arrival_date_year,
    CONVERT(DATE, CONVERT(VARCHAR, arrival_date_day_of_month) + ' ' + arrival_date_month + ' ' +
CONVERT(VARCHAR, arrival_date_year)) AS arrival_date, -- concat and convert year,month, day of month
into date format. We name it as arrival_date column.
    CASE DATEPART(dw, CONVERT(DATE, CONVERT(VARCHAR, arrival_date_day_of_month) + ' ' +
arrival_date_month + ' ' + CONVERT(VARCHAR, arrival_date_year))) -- DATEPART(dw, ...) extracts day
of the week as an integer based on the CASE statement. CASE statement then converts integer to it's
corresponding day name (i.e. 'Sunday' is dw value corresponding to 1)
        WHEN 1 THEN 'Sunday'
        WHEN 2 THEN 'Monday'
        WHEN 3 THEN 'Tuesday'
        WHEN 4 THEN 'Wednesday'
        WHEN 5 THEN 'Thursday'
        WHEN 6 THEN 'Friday'
        WHEN 7 THEN 'Saturday'
    END AS day_of_week
FROM hotels;
```

Dataset Preparation in SQL [6/7]

Creation of ‘revenue’ column



Average Daily Rate (ADR)
A metric widely used in the hospitality industry to indicate the average revenue earned for an occupied room on a given day.
Investopedia



```sql
-- Categorical Query 5): creation of revenue categorization code:
WITH hotels AS (
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2018$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2019$']
    UNION ALL
    SELECT * FROM Hotel_Revenue_Dataset.dbo.['2020$']
),
hotel_all_columns_with_revenue AS (
SELECT reservation_status, adr, discount, c.Cost,
 (stays_in_week_nights + stays_in_weekend_nights) AS Number_of_Rooms_Sold, -- used to calculate
revenue
        CASE
                WHEN reservation_status = 'Canceled' THEN 0 -- Validated it's 0. Here we assume that
customers who Canceled the reservation did so within the full return window, hence no revenue is
generated from such transcations.
                WHEN ADR < 0 THEN 0 -- Validated it's 0. As concluded earlier. For ADR < 0, it's
highly likely a calculation error/ data discrepancy has occurred.
                WHEN (reservation_status = 'Check-Out' OR reservation_status = 'No-Show') AND (adr =
0 OR adr < 0) THEN 0 -- Validated it's 0. If a customer has checked out or no show, ADR should not
be less than or equal to 0 as the customer should be billed for such cases and thus will generate
revenue for the company.
                ELSE
                ( ((stays_in_week_nights + stays_in_weekend_nights) * (adr) * (1 - Discount)) -
c.Cost )
            END AS revenue
FROM hotels AS h
LEFT JOIN Hotel_Revenue_Dataset.dbo.market_segment$ AS m ON h.market_segment = m.market_segment
LEFT JOIN Hotel_Revenue_Dataset.dbo.meal_cost$ AS c ON h.meal = c.meal
)
SELECT *
FROM hotel_all_columns_with_revenue
```

Sincewe’regiven the:

1. ADR value
2. # of weekdays and weekends stayed from a particular booking
3. % Discount corresponding for guests belong to a particular market_segment
4. Meal Cost of food provided to the guest

We can manipulate the ADR formula to obtain the corresponding revenue generated by each booking, this is given by:

$$Revenue\ per\ booking$$
$$= [(stays\ in\ week\ nights + stays\ in\ weekend\ nights) \times ADR \times (1 - Discount\,]) - Meal\ Cost$$

Assumptions:

1. Weassume that customers who canceled booking did so within the full return window
2. Customers that have made the booking, did not cancel in time, or show up on the appointment date, will still be charged and thus generate revenue

Dataset Preparation in SQL [7/7]

Amalgating all previous queries into a single giant query in SSMS to export into PowerBI

After validating the calculations ofeach individualquery created,wenow combineallqueries into a single giant query to bring in the necessary columns and features for data analysis and visualization

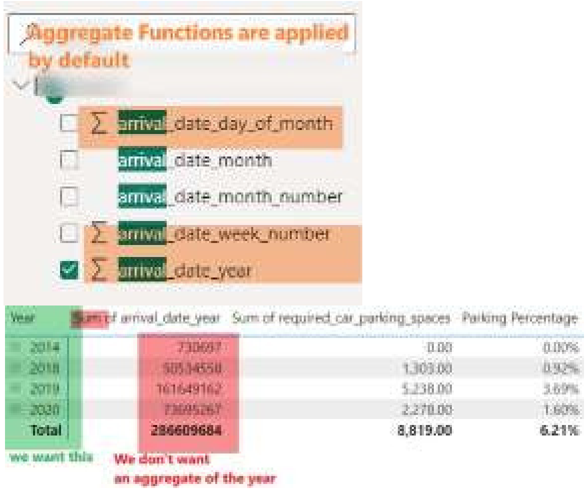
PowerBI Data Pre-Processing [1/3]

Combining“arrival_date_year”,“arrival_date_month”,“arrival_date_day_of_month”intoasinglecolumn for continuous time axis plottingwithhierarchicalsorting

The dataset ported into PowerBI has separate columns for arrival information for guests, given by:

- “arrival_date_year”,
- “arrival_date_month”,
- “arrival_date_day_of_month”

This makes it difficult to hierarchically display arrival date information of guests in a continuous time axis



Aggregate Functions are applied by default

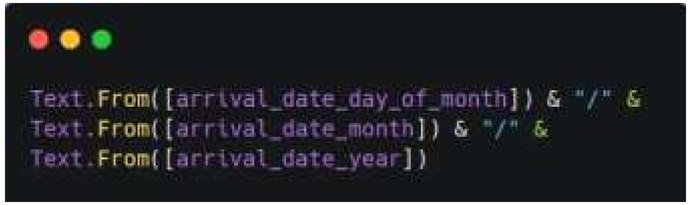
☐ ∑ arrival_date_day_of_month
☐ arrival_date_month
☐ arrival_date_month_number
☐ ∑ arrival_date_week_number
☑ ∑ arrival_date_year

Year	Sum of arrival_date_year	Sum of required_car_parking_spaces	Parking Percentage
2014	730697	0.00	0.00%
2018	50534558	1,303.00	0.92%
2019	161649162	5,238.00	3.69%
2020	73695267	2,278.00	1.60%
Total	**286609684**	**8,819.00**	**6.21%**

we want this

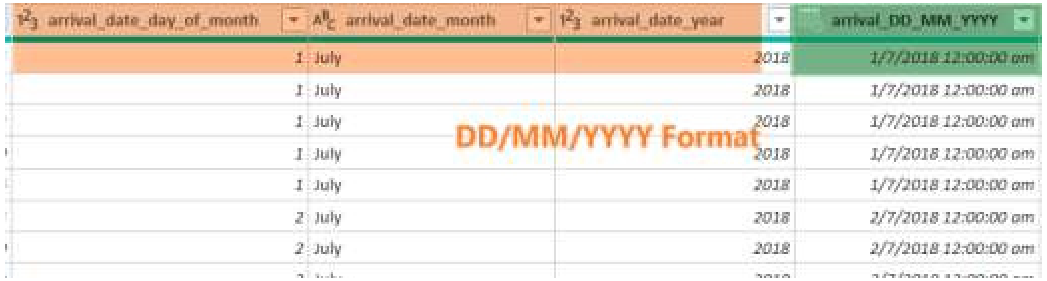
We don't want an aggregate of the year

A custom column named “arrival_DD_MM_YYYY” was created in PowerBI with the formula below:



```
Text.From([arrival_date_day_of_month]) & "/" &
Text.From([arrival_date_month]) & "/" &
Text.From([arrival_date_year])
```

The new column was converted to DateTime Format:



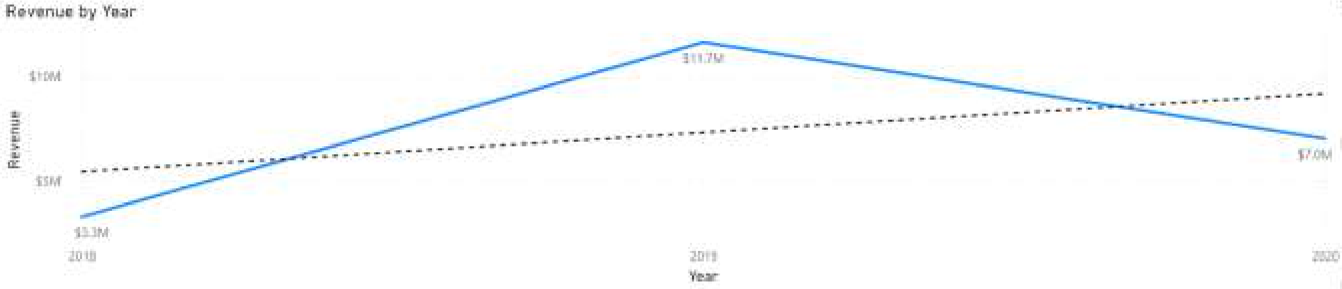
arrival_date_day_of_month	arrival_date_month	arrival_date_year	arrival_DD_MM_YYYY
1	July	2018	1/7/2018 12:00:00 am
1	July	2018	1/7/2018 12:00:00 am
1	July	2018	1/7/2018 12:00:00 am
1	July	2018	1/7/2018 12:00:00 am
1	July	2018	1/7/2018 12:00:00 am
2	July	2018	2/7/2018 12:00:00 am
2	July	2018	2/7/2018 12:00:00 am

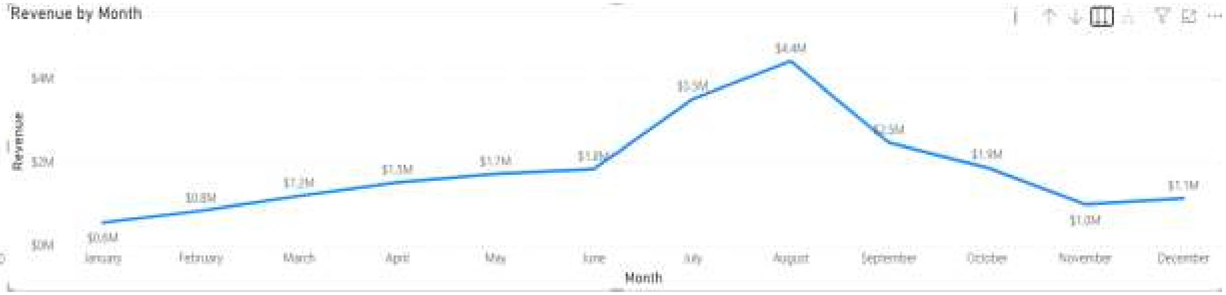
DD/MM/YYYY Format

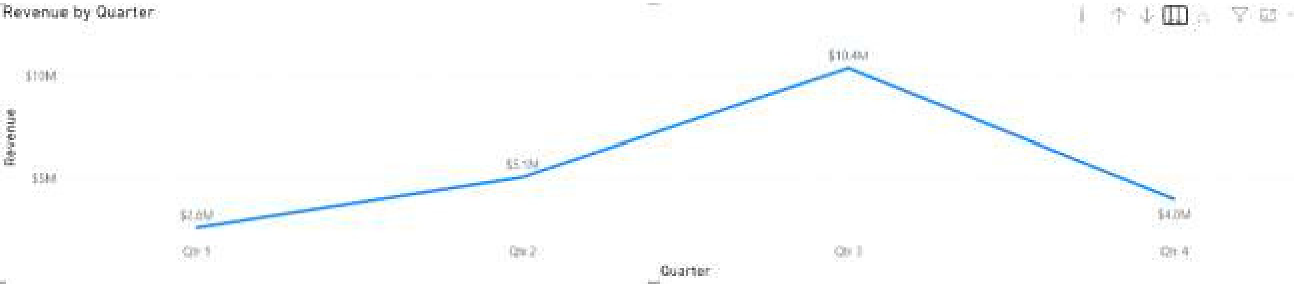
PowerBI Data Pre-Processing [2/3]

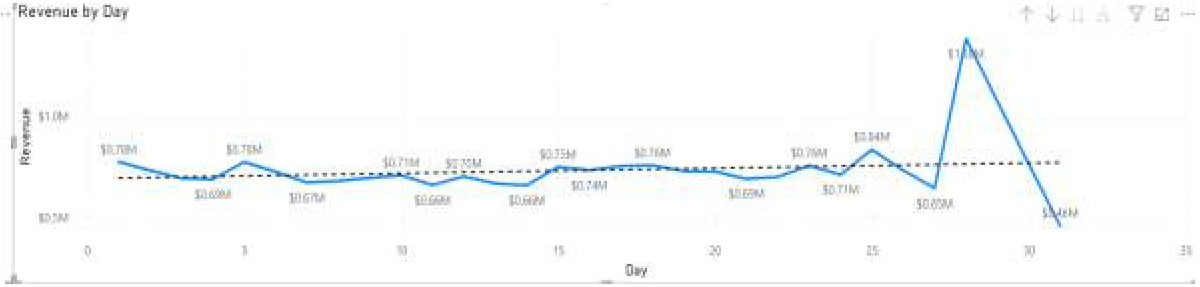
Combining“arrival_date_year”,“arrival_date_month”,“arrival_date_day_of_month”intoasinglecolumn for continuous time axis plottingwithhierarchicalsorting

- Withthe new“arrival_DD_MM_YYYY”column,wecanfinallydisplayguests’arrivaltime&datedetailsinahierarchical format:
    - By Year
    - By Quarter
    - By Month
    - By Day







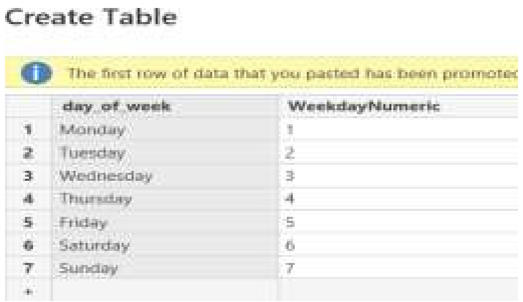


PowerBI Data Pre-Processing [3/3]

Creating‘WeekdayNumericTable’ to sort guests booking by day_of_week

PowerBIsortsdaysofthe weekin alphabetical order by default

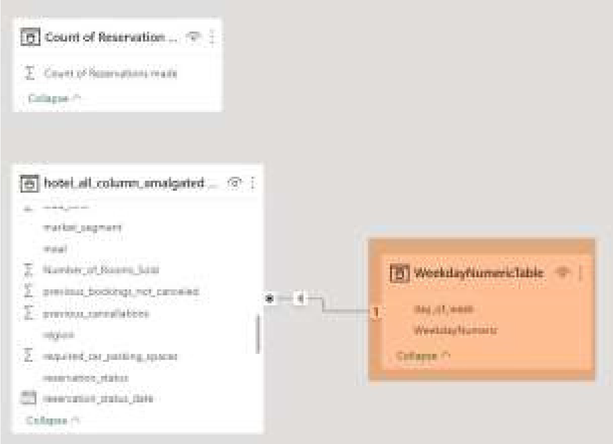
- To logically sort ‘day_of_week’, wecreatea table with the same columnname ‘day_of_week’
- Now, ‘day_of_week’ from the original query is able to act as a foreign key and establish a relationship with ‘day_of_week’



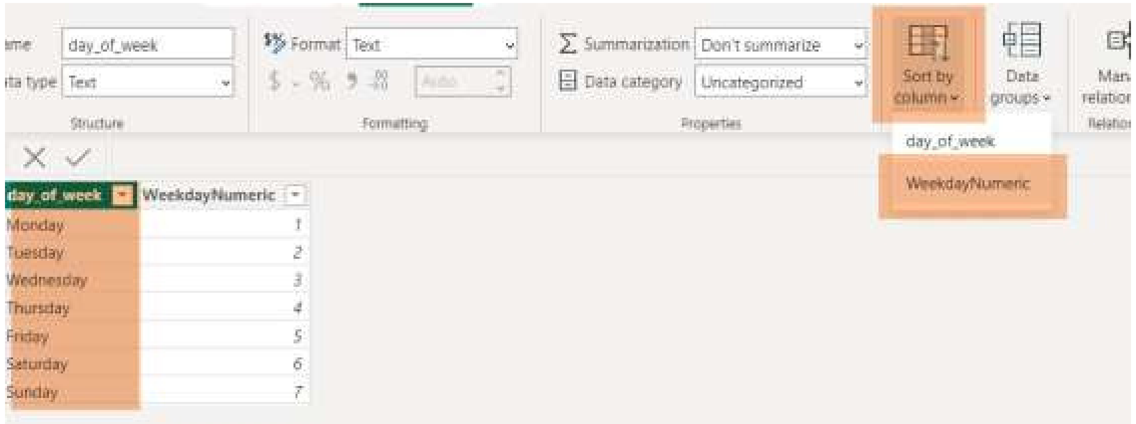
Create Table

The first row of data that you pasted has been promoted

	day_of_week	WeekdayNumeric
1	Monday	1
2	Tuesday	2
3	Wednesday	3
4	Thursday	4
5	Friday	5
6	Saturday	6
7	Sunday	7
+		



- Finally, ‘day_of_week’ was assigned to sort by column, using the ‘WeekdayNumeric’ column, to obtain the weekday-sorted plot



ame day_of_week

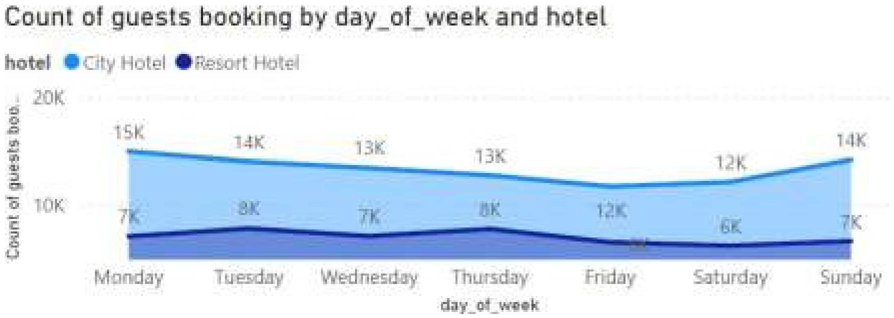
ta type Text

Format Text

Summarization Don't summarize

Data category Uncategorized

day_of_week	WeekdayNumeric
Monday	1
Tuesday	2
Wednesday	3
Thursday	4
Friday	5
Saturday	6
Sunday	7



Regarding when "reservation_status_date" & "arrival_DD_MM_YYYY" is used during visualization

Field Name	Description
reservation_status	*Reservation last status, assuming one of three categories: Canceled – booking was canceled by the customer; Check-Out – customer has checked in but already departed; No-Show – customer did not check-in and did inform the hotel of the reason why*
Reservation_status_date	***Date at which the last status was set. This variable can be used in conjunction with the ReservationStatus to understand when was the booking canceled or when did the customer checked-out of the hotel***

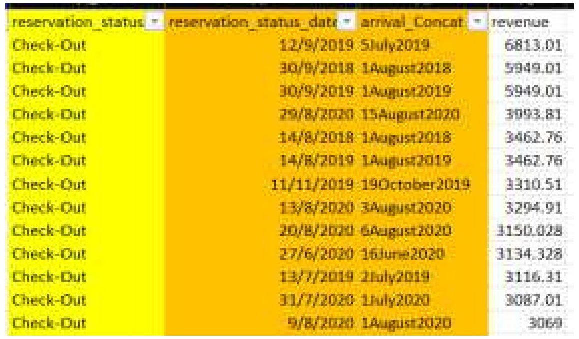
- If we want to answer revenue-related questions with plots, we'll use "reservation_status_date" as the x-axis. This is because as defined in our SQL query, revenue is only generated when reservation_status is either "Check-Out" or "No-Show" having an adr >= 0.
  - For a revenue generating entry, since the "reservation_status_date" date value will always update to the latest date in which the customer has "check-out" or "No-Show", it is a more accurate representation of when revenue is generated as compared to using "arrival_DD_MM_YYYY" as the x-axis
  - Conversely, this also assumes that the hotel's revenue is generated when the guests physically check-out from the hotel itself
  - This also means that we assume that all revenue in this data analysis is not generated from online booking

Field Name	Description
Arrival_date_year	*Year of arrival date*
Arrival_date_month	*Month of arrival date*
Arrival_date_week_number	*Week number of year for arrival date*
Arrival_date_day_of_month	*Day of arrival date*

- Conversely, if we want to answer questions such as "Which are the most busy month?" & parking-related questions, We'll use "arrival_DD_MM_YYYY" column as it'll be a more accurate time representation of when the guests have physically "checked-in" into the hotel

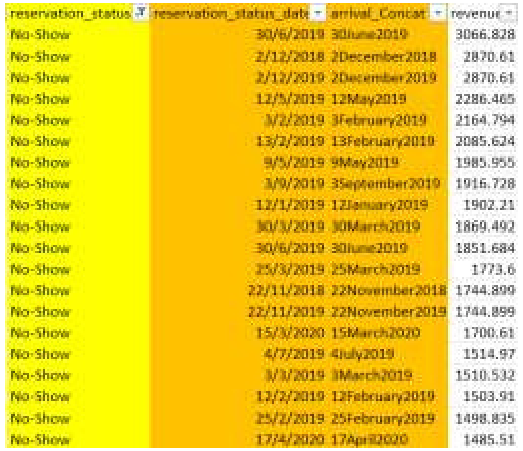
Elaborating on ‘reservation_status_date’

- For “Check-Out”, we see that “reservation_status_date” is >= arrival date.
    - This means that the date listed in reservation_status_date is updated at the same time when the customer has checked out and thus the revenue is generated



reservation_status	reservation_status_dat	arrival_Concat	revenue
Check-Out	12/9/2019	5July2019	6813.01
Check-Out	30/9/2018	1August2018	5949.01
Check-Out	30/9/2019	1August2019	5949.01
Check-Out	29/8/2020	15August2020	3993.81
Check-Out	14/8/2018	1August2018	3462.76
Check-Out	14/8/2019	1August2019	3462.76
Check-Out	11/11/2019	19October2019	3310.51
Check-Out	13/8/2020	3August2020	3294.91
Check-Out	20/8/2020	6August2020	3150.028
Check-Out	27/6/2020	16June2020	3134.328
Check-Out	13/7/2019	2July2019	3116.31
Check-Out	31/7/2020	1July2020	3087.01
Check-Out	9/8/2020	1August2020	3069

- For “No-Show”, we see that that “reservation_status_date” = arrival date

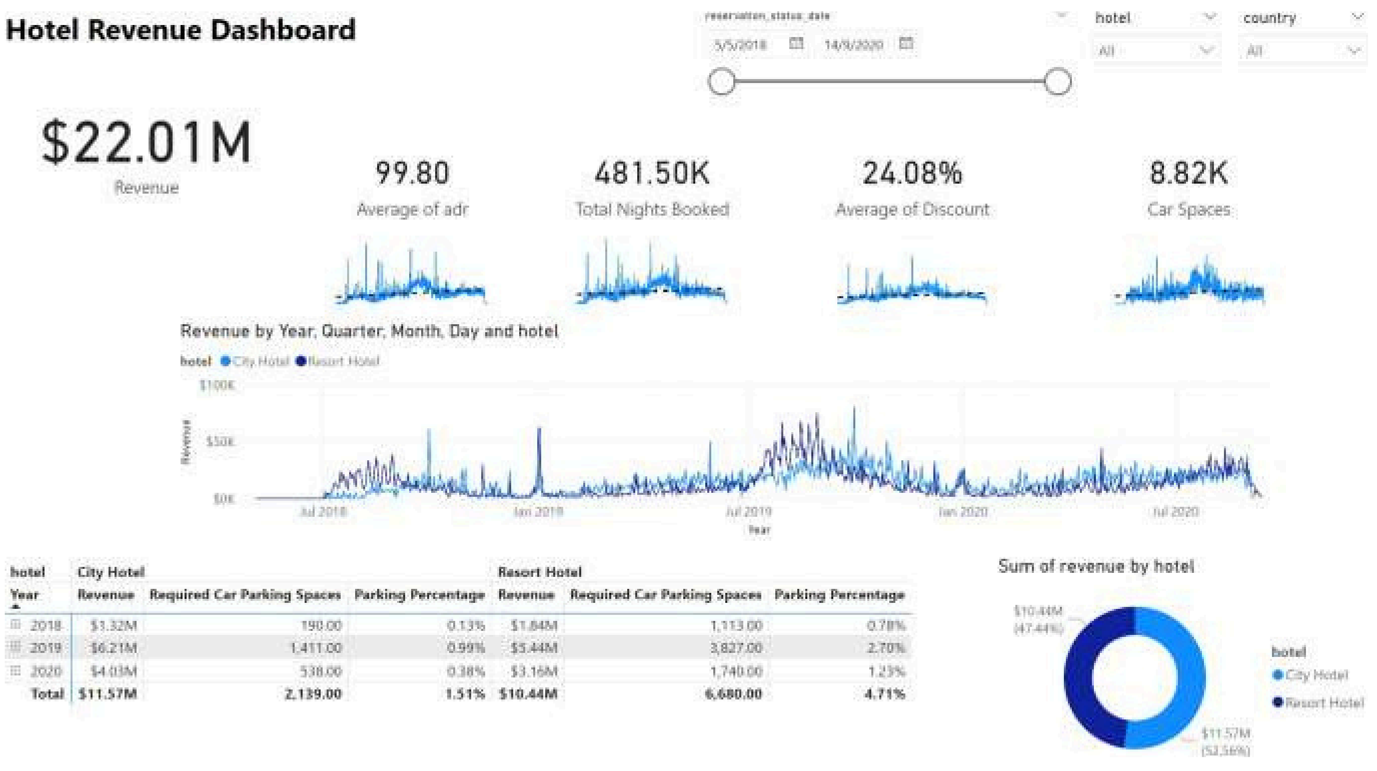


reservation_status	reservation_status_dat	arrival_Concat	revenu
No-Show	30/6/2019	30June2019	3066.828
No-Show	2/12/2018	2December2018	2870.61
No-Show	2/12/2019	2December2019	2870.61
No-Show	12/5/2019	12May2019	2286.465
No-Show	3/2/2019	3February2019	2164.794
No-Show	13/2/2019	13February2019	2085.624
No-Show	9/5/2019	9May2019	1985.955
No-Show	3/9/2019	3September2019	1916.728
No-Show	12/1/2019	12January2019	1902.21
No-Show	30/3/2019	30March2019	1869.492
No-Show	30/6/2019	30June2019	1851.684
No-Show	25/3/2019	25March2019	1773.6
No-Show	22/11/2018	22November2018	1744.899
No-Show	22/11/2019	22November2019	1744.899
No-Show	15/3/2020	15March2020	1700.61
No-Show	4/7/2019	4July2019	1514.97
No-Show	3/3/2019	3March2019	1510.532
No-Show	12/2/2019	12February2019	1503.91
No-Show	25/2/2019	25February2019	1498.835
No-Show	17/4/2020	17April2020	1485.51

<u>Takeaway reminder:</u>

Reservation_status_date IS NOT THE DATE where the customer has reserved the booking prior to arriving at the hotel

Hotel Revenue Dashboard



Hotel Revenue Dashboard

reservation_status_date: 5/5/2018 – 14/9/2020

hotel: All

country: All

$22.01M Revenue

99.80 Average of adr

481.50K Total Nights Booked

24.08% Average of Discount

8.82K Car Spaces

Revenue by Year, Quarter, Month, Day and hotel

hotel	City Hotel			Resort Hotel		
Year	Revenue	Required Car Parking Spaces	Parking Percentage	Revenue	Required Car Parking Spaces	Parking Percentage
2018	$1.32M	190.00	0.13%	$1.84M	1,113.00	0.78%
2019	$6.21M	1,411.00	0.99%	$5.44M	3,827.00	2.70%
2020	$4.03M	538.00	0.38%	$3.16M	1,740.00	1.23%
Total	**$11.57M**	**2,139.00**	**1.51%**	**$10.44M**	**6,680.00**	**4.71%**

Sum of revenue by hotel

It provides at-a-glance information about features that are relevant to revenue generation

It also provides a comparison of

revenue generated by different hotels, and parking space utilization information

Hotel guest-related seasonality EDA

This section will mainly focus on relevant visualizations that will aid in answering the following questions:

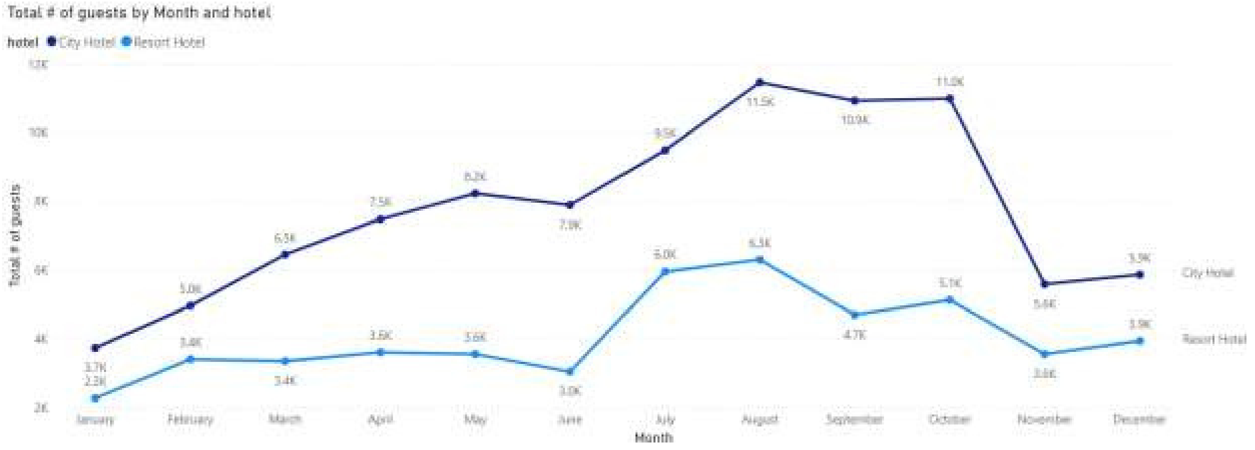
- “Which are the busiest months?”
- “On average, how long do guests stay at the hotel?”
- “Which day of the week do different types of guests, typically like to book on?”

Breakdown on the type of guests that are booking the hotel will be provided

Relevant recommendations will be provided for the hotel manager to improve staffing allocation based on guest booking arrival data

Hotel guest-related seasonality EDA

“Which are the busiest months?”

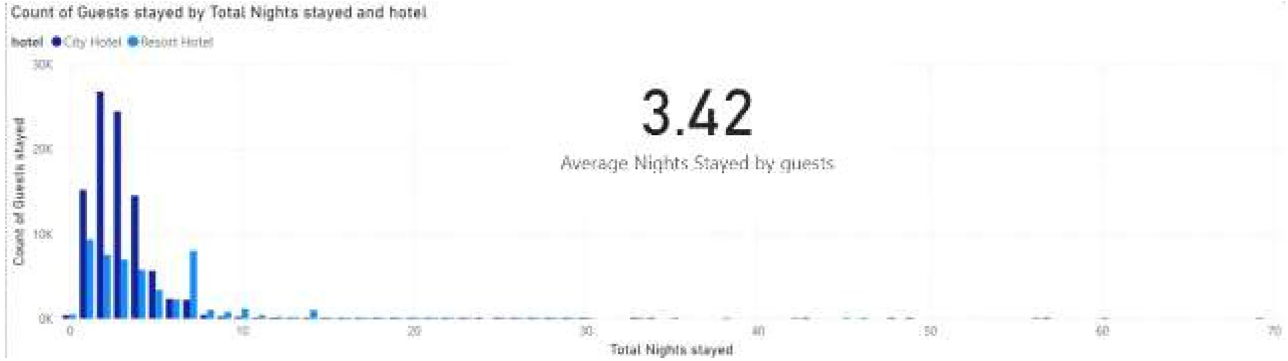


- From theline chart, we observe that the busiest months across all years are:
  - July & August for Resort Hotels
  - July to October for City Hotels

<u>Recommendation:</u> The hotel manager

should take note to plan service staff shift accordingly to allocate more hotel staffing during those critical high volume months

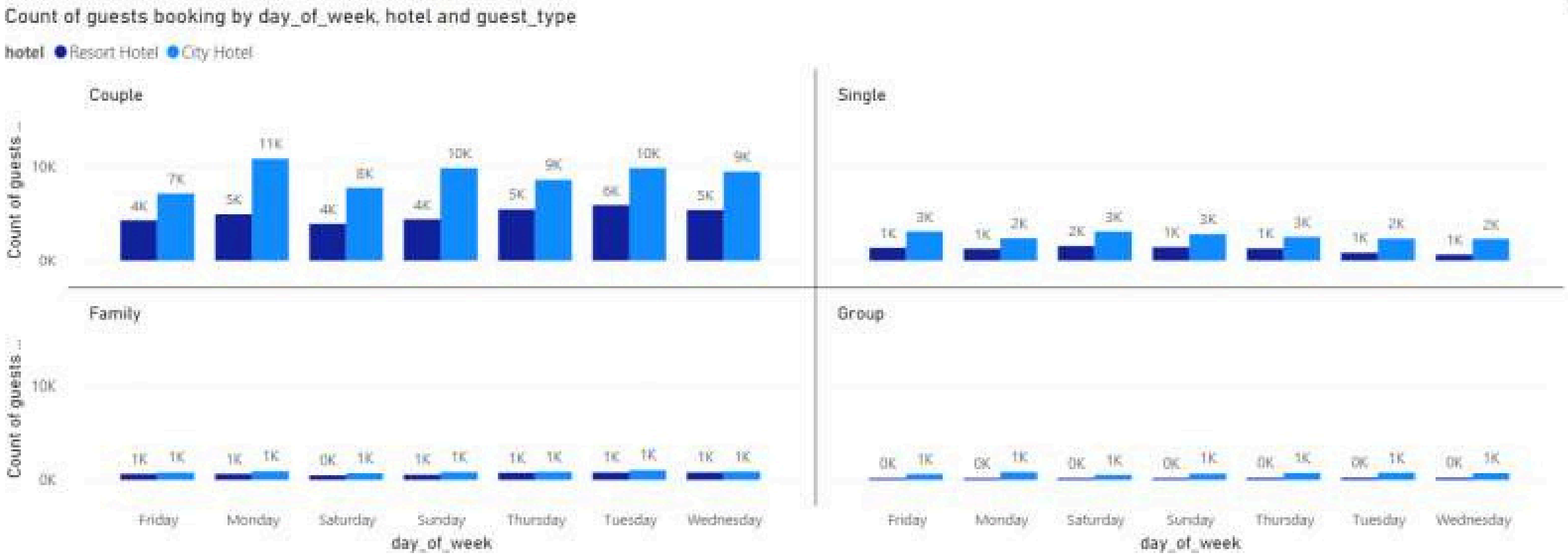
On average, how long do guests stay at the hotel?



- On average, guests stay ***<u>about 3.42 nights</u>*** for both hotels

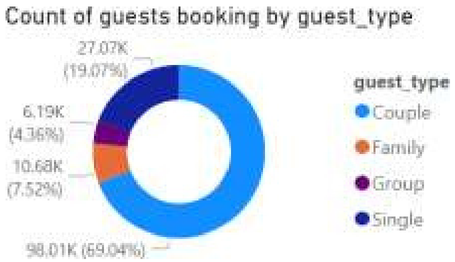
Hotel guest-related seasonality EDA

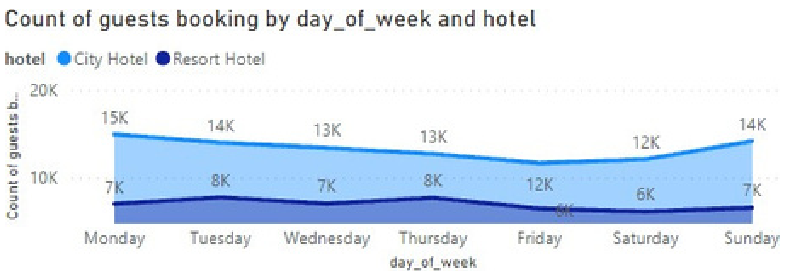
“Which day of the week do different types of guests, typically like to book on?”


Count of guests booking by day_of_week, hotel and guest_type

Based on the data, we infer that:

- Couples and single guests typically account for the majority of bookings made for both hotels
- There are significantly more bookings made by Couple & Single guests for City Hotels as compared to Resort Hotels
- The number of City Hotel bookings made by group guests are about ten-fold for every day of the week as compared to Resort Hotels


Count of guests booking by guest_type


Count of guests booking by day_of_week and hotel

- For both hotels, the count of guests arriving for each day of the week appears to be quite evenly distributed

Recommendation: Management could look into designing attractive packages tailored towards Couple & Single guests for both hotels, given that they represent the majority of the customers

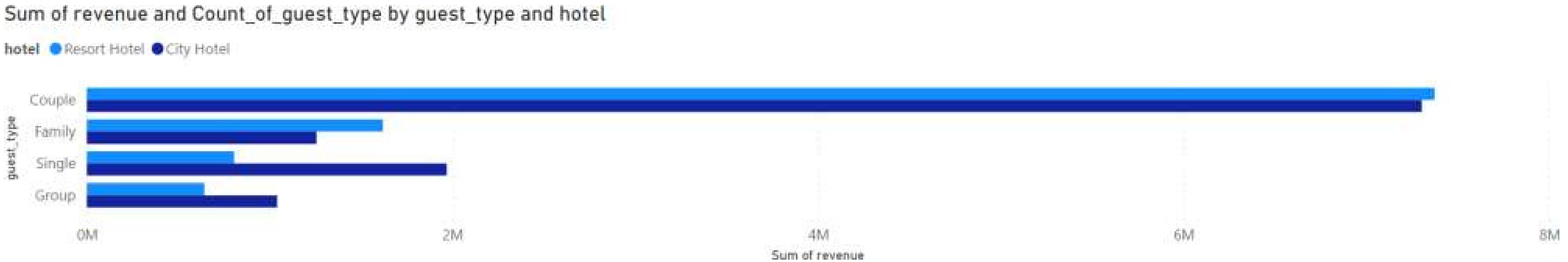
Hotel Revenue EDA

This section will mainly focus on relevant visualizations that will aid in answering the following questions:

- “Which typeof guestsarelikely to generate more revenue byhotel?”
- “Give a revenue breakdown by region. What are the top 5 countries that has generated the greatest revenue?”
- “Is our hotel revenue growing by year? If so, which hotel brings in greater revenue? Is there any observed seasonality with respect to revenue?”

Hotel Revenue EDA

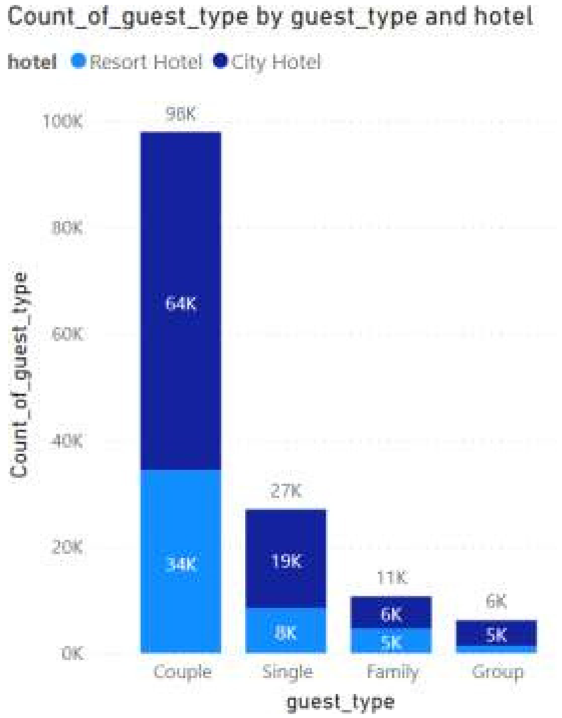
“Whichtypeofguestsarelikelytogeneratemorerevenuebyhotel?”



Sum of revenue and Count_of_guest_type by guest_type and hotel

- **Couples and family guests tend to generate more revenue in resort hotels as compared to city hotels. While single and group guests tend to generate more revenue in city hotels.**
- The revenue difference is marginal between hotels for couples. This is in spite of there being almost twice the number of couples who booked for city hotel (as indicated in the stacked bar plot to the right)
  - This marginal difference in revenue despite fewer bookings in resort hotels may be due to the presence of more expensive amenities (such as pools & spa services) packaged for couples and families in resort hotels

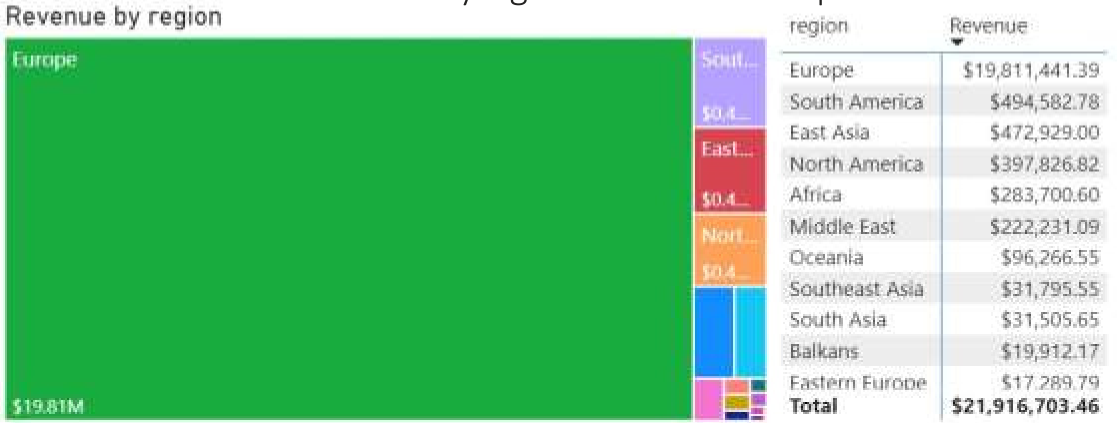
- On the other hand, **more revenue has been generated in city hotels for single and group hotel guests**
  - This is likely attributed by city hotel having more bookings for single and group guests as compared to resort hotels
    - This is in-line with what was previously mentioned, that couples and families are likely to have bundled extra amenities when they had booked for resort hotels hence attributing to a higher revenue despite fewer bookings



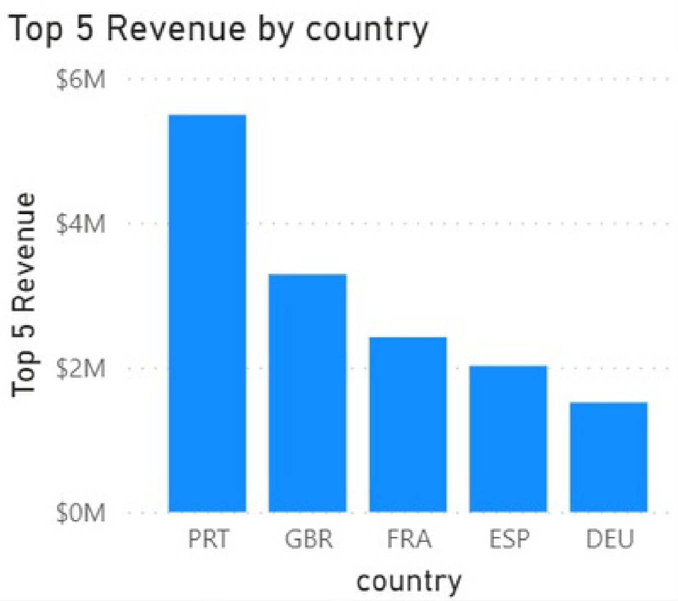
Count_of_guest_type by guest_type and hotel

Hotel Revenue EDA

“Givearevenuebreakdownbyregion.Whatarethetop5countries that has generated the greatest revenue?”



region	Revenue
Europe	$19,811,441.39
South America	$494,582.78
East Asia	$472,929.00
North America	$397,826.82
Africa	$283,700.60
Middle East	$222,231.09
Oceania	$96,266.55
Southeast Asia	$31,795.55
South Asia	$31,505.65
Balkans	$19,912.17
Eastern Europe	$17,289.79
Total	**$21,916,703.46**



IRELAND
UNITED KINGDOM
GERMANY
FRANCE
ITALY
PORTUGAL
SPAIN

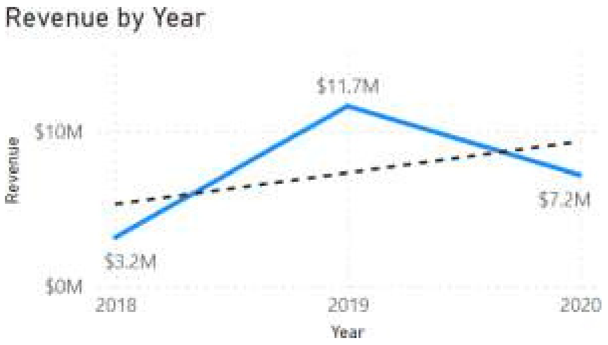
- The main bulk of revenue generated were from guests in the European region from 2018-2020

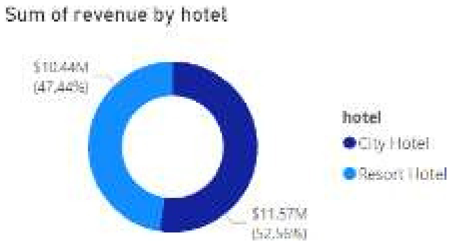
- The **corresponding top 5 countries** by revenue are:
  1. PRT – Portugal, $5.49M
  2. GBR – United Kingdom, $3.28M
  3. FRA – France, $2.42M
  4. ESP – Spain, $2.02M
  5. DEU – Germany, $1.51M

The top 5 countries by revenue are all situated in the European region.
This may suggest that the location of the hotels may likely to be situated in the European region

Hotel Revenue EDA

“Isourhotelrevenuegrowingbyyear?Ifso,whichhotelbrings in greater revenue? Is there any observed seasonality with respect to revenue?”

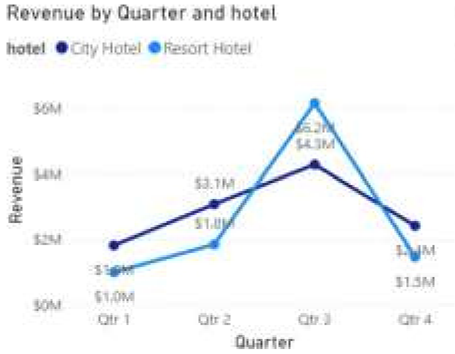


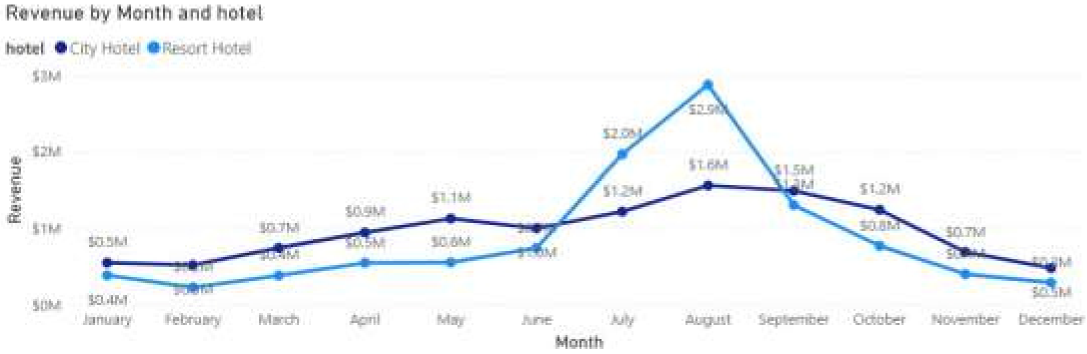


- Yes,ouroverallhotelrevenue is growing by year and there is an upwards growth trend as depicted in the trendline
  - Revenue trendedup, resulting in a **127.20%**increase between2018and 2020.
  - 2019 had the **highest combined revenue recorded at $11.7M**, which is 254.18% higher than the lowest revenue recorded at $3.2M

- From 2019 to 2020, we see a decrease in revenue from $11.7M to $7.2M.
  - This may be due to the influence of COVID-19 pandemic which may have impacted the ho recorded case of COVID-19 death in Portugal was recorded on 16 MAR 2020]

- **City Hotel brings in greater revenue**[52.56% of total revenue] as compared to Resort Hotel[47.44%oftotal revenue]





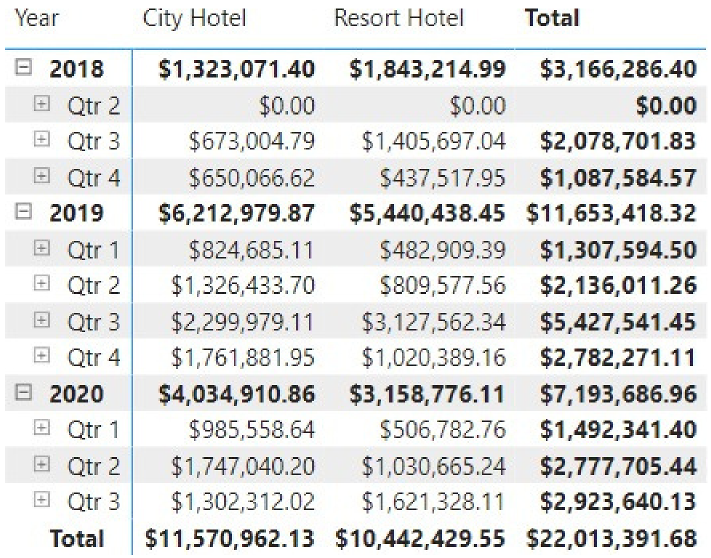
Seasonality trend based on limited dataset from 2018 -2

- Across all years, it was observed that revenue typically pick up from the end of Q2 (June), cumulating in a ma the month of August

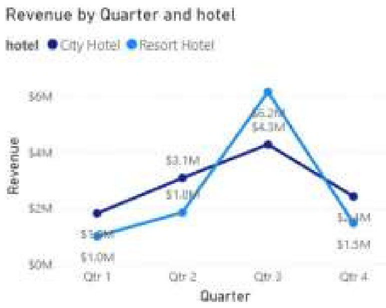
Limitations of dataset

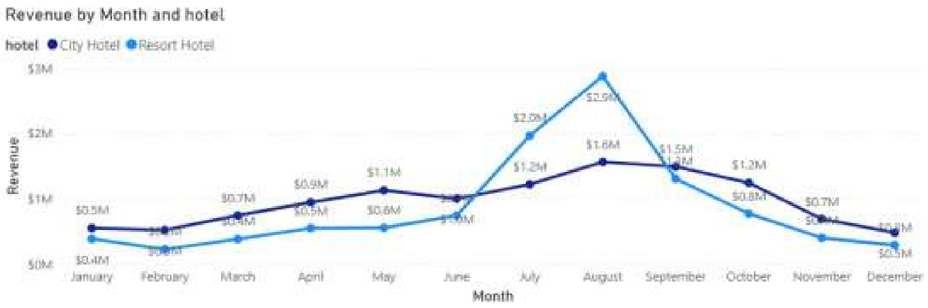
Limiteddata on2018&2020maypaintanincorrectionconclusion for revenue & guest-related seasonality reporting

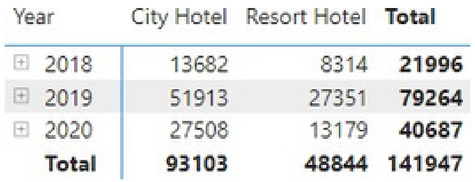
- Morecompletedatasetcontaininginformationfrom**2018 Q1,Q2 & 2020 Q4** is required for the seasonality analysis to be moreaccurate



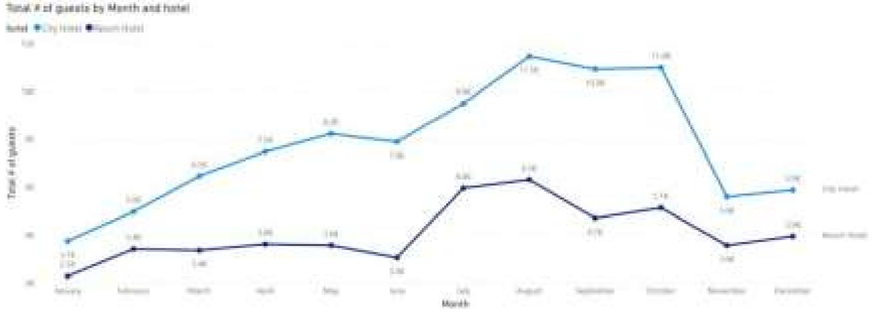
Year	City Hotel	Resort Hotel	Total
⊟ **2018**	**$1,323,071.40**	**$1,843,214.99**	**$3,166,286.40**
⊞ Qtr 2	$0.00	$0.00	**$0.00**
⊞ Qtr 3	$673,004.79	$1,405,697.04	**$2,078,701.83**
⊞ Qtr 4	$650,066.62	$437,517.95	**$1,087,584.57**
⊟ **2019**	**$6,212,979.87**	**$5,440,438.45**	**$11,653,418.32**
⊞ Qtr 1	$824,685.11	$482,909.39	**$1,307,594.50**
⊞ Qtr 2	$1,326,433.70	$809,577.56	**$2,136,011.26**
⊞ Qtr 3	$2,299,979.11	$3,127,562.34	**$5,427,541.45**
⊞ Qtr 4	$1,761,881.95	$1,020,389.16	**$2,782,271.11**
⊟ **2020**	**$4,034,910.86**	**$3,158,776.11**	**$7,193,686.96**
⊞ Qtr 1	$985,558.64	$506,782.76	**$1,492,341.40**
⊞ Qtr 2	$1,747,040.20	$1,030,665.24	**$2,777,705.44**
⊞ Qtr 3	$1,302,312.02	$1,621,328.11	**$2,923,640.13**
Total	**$11,570,962.13**	**$10,442,429.55**	**$22,013,391.68**







Year	City Hotel	Resort Hotel	Total
⊞ 2018	13682	8314	**21996**
⊞ 2019	51913	27351	**79264**
⊞ 2020	27508	13179	**40687**
Total	**93103**	**48844**	**141947**

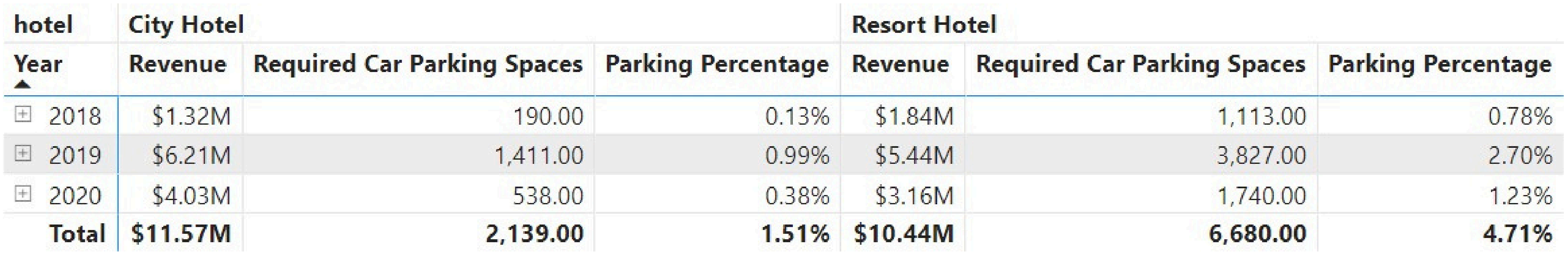


Limitations of dataset

Moreinformationabout car parking spacesisrequired to be provided into the dataset to accurately calculatetheparking percentage

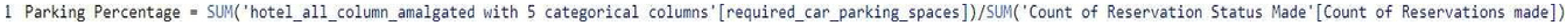
- For example, a more accurate representation of the calculated ‘Parking Percentage’ should be obtained with a column containting ‘total_car_parking_spaces_available’, where:
  - ParkingPercentage = $\frac{Required\ \ Car\ \ Parking\ Spaces}{TotalCarParkingSpaces\ Available} \times 100\%$

Field Name	Description
Required_car_parking_spaces	Number of car parking spaces requiredbythecustomer



hotel	City Hotel			Resort Hotel		
Year	Revenue	Required Car Parking Spaces	Parking Percentage	Revenue	Required Car Parking Spaces	Parking Percentage
⊞ 2018	$1.32M	190.00	0.13%	$1.84M	1,113.00	0.78%
⊞ 2019	$6.21M	1,411.00	0.99%	$5.44M	3,827.00	2.70%
⊞ 2020	$4.03M	538.00	0.38%	$3.16M	1,740.00	1.23%
Total	**$11.57M**	**2,139.00**	**1.51%**	**$10.44M**	**6,680.00**	**4.71%**

Currently, the Required Car Parking Spaces in the dashboard is calculated as such:



```
Parking Percentage = SUM('hotel_all_column_amalgated with 5 categorical columns'[required_car_parking_spaces])/SUM('Count of Reservation Status Made'[Count of Reservations made])
```

- Parking Percentage = $\frac{Required\ Car\ Parking\ Spaces}{Count\ of\ Reservation\ Status\ made} \times 100\%$

The validity of the calculation can be further improved if we can obtain a ‘total_car_parking_spaces_available’ column for management to decide if a larger parking lot is required to be built.

Conclusion / Key-Takeaways

Based onthe limited datasetprovided, below are thekey-takeawaysfrom this project:

With respect to hotel guest-related seasonality:

- The hotel manager should ensure increased hotel staffing during these months for each hotels:

o July & August for Resort Hotels

o July to October for City Hotels

- Guests on average stay 3.42 nights for both hotels
- No particular trends can be observed as to which days are more popular for guests to book on
- Couples & single guests represent the majority of bookings made for both hotels, management can look into designing attractive packages tailored towards such guests

With respect to Hotel Revenue:

- Couples & Family guests tend to generate more revenue in resort hotels as compared to city hotels, while group guests generate more revenue in city hotels
- The main bulk of revenue generated was attributed to the European region
  - Portugal corresponded to the country with the highest revenue
- Hotel revenue displays an upwards trend and is growing by year
- City Hotel brings in greater revenue as compared to Resort Hotel
- Revenue revenue typically starts to pick up from the end of Q2 (June), cumulating in a maximum in the month of August