

SQL Business Case

I. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset.

A. Data type of all columns in the “customers” table.

Describe target.customers ;

B. Get the time range between which the orders were placed.

```
select max(order_purchase_timestamp) as last_order,  
min(order_purchase_timestamp) as first_order  
from `target.orders`;
```

last_order	first_order
2018-10-17 17:30:18.000000	2016-09-04 21:15:19.000000
UTC	UTC

C. Count the number of Cities and States in our dataset.

```
select  
count(distinct(geolocation_city)) as unique_cities,  
count(distinct(geolocation_state)) as unique_states  
from `target.geolocation`;
```

unique_cities	unique_states
8011	27

II. In-depth Exploration:

A. Is there a growing trend in the no. of orders placed over the past years ?

```
select
extract (year from order_purchase_timestamp) as years,
extract (month from order_purchase_timestamp) as months ,
count(*) as total
from target.orders
group by years, months
order by 1 ,2, 3
```

years	months	total
2016	9	4
2016	10	324
2016	12	1
2017	1	800
2017	2	1780
2017	3	2682
2017	4	2404
2017	5	3700
2017	6	3245
2017	7	4026
2017	8	4331
2017	9	4285
2017	10	4631
2017	11	7544
2017	12	5673
2018	1	7269
2018	2	6728

B. Can we see some kind of monthly seasonality in terms of the no. of orders being placed?

years	months	total
2017	11	7544
2018	1	7269

2018	3	7211
2018	4	6939
2018	5	6873
2018	2	6728
2018	8	6512
2018	7	6292
2018	6	6167
2017	12	5673
2017	10	4631
2017	8	4331
2017	9	4285
2017	7	4026
2017	5	3700
2017	6	3245

Yes, we see some kind of monthly seasonality in terms of the no. of orders being at the year end and new year start we can see it

C. During what time of the day, do the Brazilian customers mostly place their

orders? (Dawn, Morning, Afternoon or Night)

- **0-6 hrs : Dawn**
- **7-12 hrs : Mornings**
- **13-18 hrs : Afternoon**
- **19-23 hrs : Night**

select
case
when

```

extract (hour from order_purchase_timestamp) >= 0 and extract (hour from
order_purchase_timestamp) <= 6 then 'Dawn'
when
extract (hour from order_purchase_timestamp) >= 7 and extract (hour from
order_purchase_timestamp) <= 12 then 'Morning'
when
extract (hour from order_purchase_timestamp) >= 13 and extract (hour from
order_purchase_timestamp) <= 18 then 'Afternoon'
when
extract (hour from order_purchase_timestamp) >= 19 and extract (hour from
order_purchase_timestamp) <= 23 then 'Night'
end as
time_intervel
from target.orders

```

Clearly we can see that every transaction was done by Dawn timeing means early in the morning .

III. Evolution of E-commerce orders in the Brazil region:

A. Get the month on month no. of orders placed in each state.

```

select
extract (year from o.order_purchase_timestamp) as years ,
extract (month from o.order_purchase_timestamp) as months ,
c.customer_state,
count(c.customer_id) as total
from target.customers c join target.orders o on
c.customer_id = o.customer_id
group by years,months,c.customer_state
order by 1,2,3,4

```

years	months	customer_state	total
2016	9	RR	1
2016	9	RS	1
2016	9	SP	2
2016	10	AL	2
2016	10	BA	4
2016	10	CE	8
2016	10	DF	6
2016	10	ES	4

2016	10	GO	9
2016	10	MA	4
2016	10	MG	40
2016	10	MT	3
2016	10	PA	4
2016	10	PB	1
2016	10	PE	7
2016	10	PI	1
2016	10	PR	19
2016	10	RJ	56
2016	10	RN	4
2016	10	RR	1
2016	10	RS	24
2016	10	SC	11
2016	10	SE	3
2016	10	SP	113

B. How are the customers distributed across all the states?

```
select
c.customer_state,
count(distinct c.customer_id) as total
from target.customers c join target.orders o on
c.customer_id = o.customer_id
group by c.customer_state
order by 1,2 desc
```

customer_state	total
AC	81
AL	413
AM	148
AP	68
BA	3380
CE	1336
DF	2140
ES	2033
GO	2020
MA	747
MG	11635
MS	715
MT	907

PA	975
PB	536

Impact on Economy: Analyze the money movement by e-commerce by looking at

order prices, freight and others.

A. Get the % increase in the cost of orders from year 2017 to 2018 (include

months between Jan to Aug only).

```
select
(sum(case when extract(year from o.order_purchase_timestamp) = 2018 then p.payment_value else
0 end ) -
sum(case when extract (year from o.order_purchase_timestamp) = 2017 then p.payment_value else 0
end) ) / sum(case when extract (year from o.order_purchase_timestamp) = 2017 then
p.payment_value else 0 end) * 100 as cost_inc_percentage
from target.orders o join target.payments p on
o.order_id = p.order_id
where extract(year from o.order_purchase_timestamp) in (2017,2018) and
extract(month from o.order_purchase_timestamp) between 1 and 8
```

```
cost_inc_percentage
136.9768716
```

B. Calculate the Total & Average value of order price for each state.

```
select c.customer_state,
round(sum(p.payment_value),2) as total,
round(avg(p.payment_value),2) as average
from target.customers c join target.orders o on
c.customer_id = o.customer_id join target.payments p on
o.order_id = p.order_id
```

group by c.customer_state
order by 1,2,3

customer_state	total	average
AC	19680.62	234.29
AL	96962.06	227.08
AM	27966.93	181.6
AP	16262.8	232.33
BA	616645.82	170.82
CE	279464.03	199.9
DF	355141.08	161.13
ES	325967.55	154.71
GO	350092.31	165.76
MA	152523.02	198.86
MG	1872257.26	154.71
MS	137534.84	186.87
MT	187029.29	195.23
PA	218295.85	215.92
PB	141545.72	248.33

C. Calculate the Total & Average value of order freight for each state.

```
select c.customer_state,
round(sum(p.freight_value),2) as total,
round(avg(p.freight_value),2) as average
from target.customers c join target.orders o on
c.customer_id = o.customer_id join target.order_items p on
o.order_id = p.order_id
group by c.customer_state
order by 1,2,3
```

customer_state	total	average
AC	3686.75	40.07
AL	15914.59	35.84
AM	5478.89	33.21
AP	2788.5	34.01
BA	100156.68	26.36
CE	48351.59	32.71
DF	50625.5	21.04
ES	49764.6	22.06
GO	53114.98	22.77
MA	31523.77	38.26
MG	270853.46	20.63

V. Analysis based on sales, freight and delivery time.

B. Find out the top 5 states with the highest & lowest average freight value.

```
select c.customer_state,  
avg(u.freight_value) as total  
from target.customers c join target.orders o on  
c.customer_id = o.customer_id join target.order_items u on  
o.order_id = u.order_id  
group by c.customer_state  
order by 2 desc  
limit 5
```

customer_state	total
RR	42.98442308
PB	42.72380399
RO	41.06971223
AC	40.07336957
PI	39.14797048

```
select c.customer_state,  
avg(u.freight_value) as total  
from target.customers c join target.orders o on  
c.customer_id = o.customer_id join target.order_items u on  
o.order_id = u.order_id  
group by c.customer_state  
order by 2  
limit 5
```

customer_state	total
SP	15.14727539
PR	20.53165157
MG	20.63016681
RJ	20.96092393
DF	21.04135495

VI. Analysis based on the payments:

A. Find the month on month no. of orders placed using different payment types.

```
select
extract (year from o.order_purchase_timestamp) as years ,
extract (month from o.order_purchase_timestamp) as months ,
p.payment_type,
count(*) as num_count
from target.orders o join target.payments p on
o.order_id = p.order_id
group by years , months ,payment_type
order by 1 ,2,3
```

years	months	payment_type	num_count
2016	9	credit_card	3
2016	10	UPI	63
2016	10	credit_card	254
2016	10	debit_card	2
2016	10	voucher	23
2016	12	credit_card	1
2017	1	UPI	197
2017	1	credit_card	583
2017	1	debit_card	9
2017	1	voucher	61
2017	2	UPI	398
2017	2	credit_card	1356
2017	2	debit_card	13

B. Find the no. of orders placed on the basis of the payment installments that have been paid.

```
select order_id ,
count( distinct case when payment_installments = 1 then payment_installments end ) as
successfully_paid_installments,
COUNT(*) AS total_installments
from target.payments
group by order_id
order by 3 desc ,2 desc ,1
```

order_id	successfully_paid_installments	total_installments
fa65dad1b0e818e3ccc5cb0e39231352	1	29
ccf804e764ed5650cd8759557269dc13	1	26
285c2e15bebd4ac83635ccc563dc71f4	1	22
895ab968e7bb0d5659d16cd74cd1650c	1	21
ee9ca989fc93ba09a6eddc250ce01742	1	19
fedcd9f7ccdc8cba3a18defedd1a5547	1	19
21577126c19bf11a0b91592e5844ba78	1	15
4bfcba9e084f46c8e3cb49b0fa6e6159	1	15
3c58bff7b70dcf45f12bdf66a3c215905	1	14
4689b1816de42507a7d63a4617383c59	1	14
4fb76fa13b108a0d0478483421b0992c	1	13
73df5d6adbbeea12c8ae03df93f346e86	1	13
cf101c3abd3c061ca9f78c1bbb1125af	1	13

FAQs

Q. Which platform am I supposed to use for writing queries?

You are advised to use BigQuery as the platform for solving this case study.

You are advised to use Google BigQuery as the platform for writing queries to solve the case study. Google BigQuery is a fully managed, serverless, and highly scalable enterprise data warehouse that allows you to analyze large datasets using SQL queries. It is a popular platform for performing analytics on big data and is often used for data exploration, machine learning, and business intelligence tasks.

