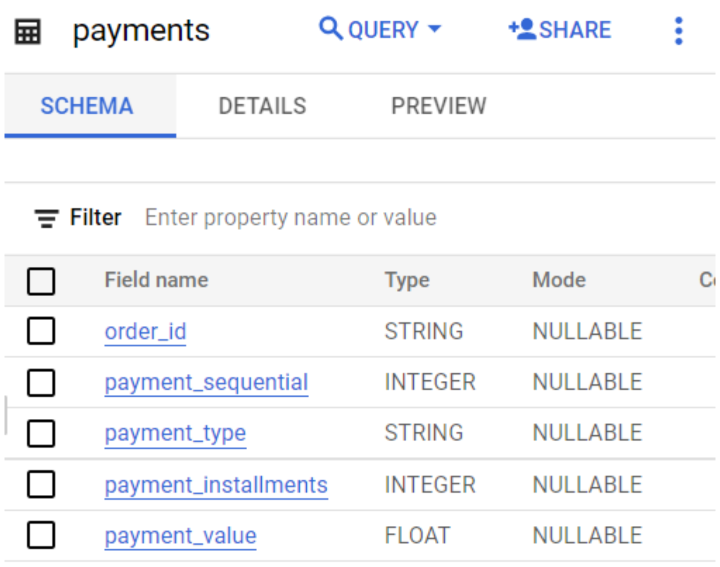
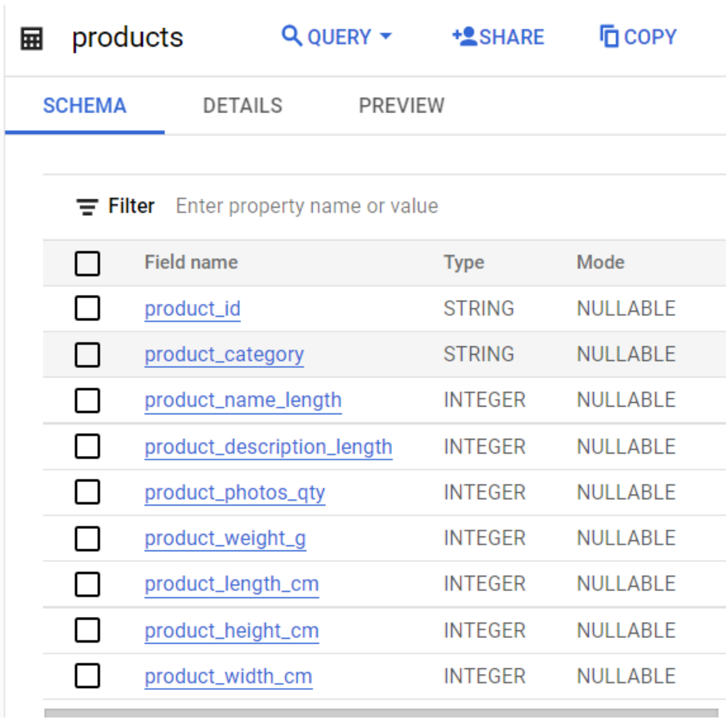
1) Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset

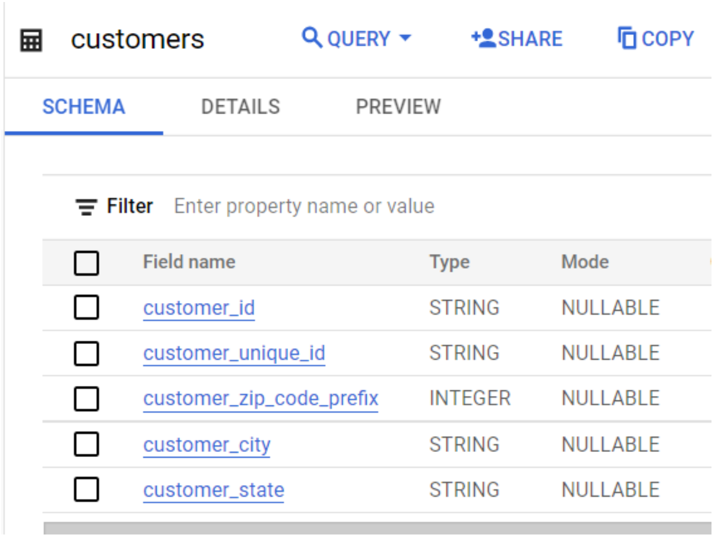
1.1 Data type of columns in a table



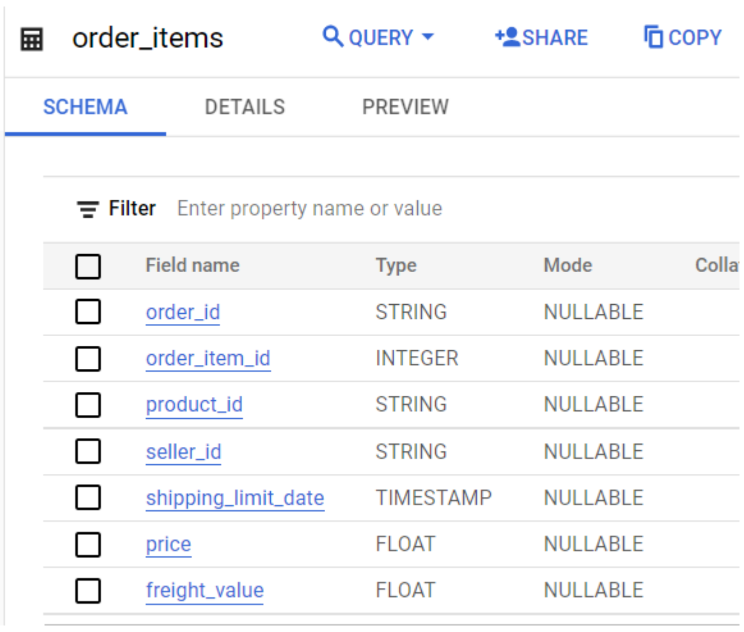
Field name	Type	Mode	C
order_id	STRING	NULLABLE	
payment_sequential	INTEGER	NULLABLE	
payment_type	STRING	NULLABLE	
payment_installments	INTEGER	NULLABLE	
payment_value	FLOAT	NULLABLE	



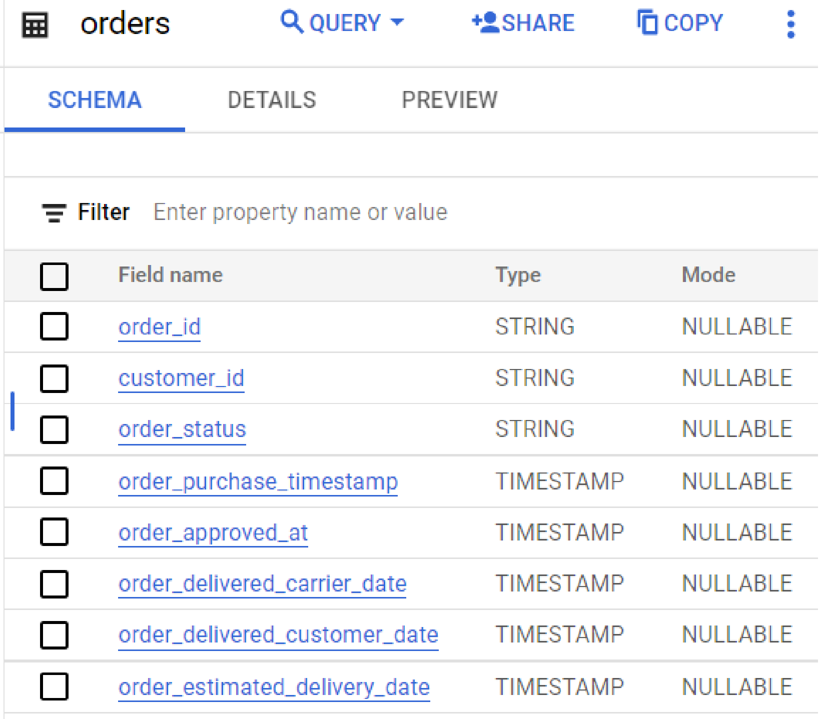
Field name	Type	Mode
product_id	STRING	NULLABLE
product_category	STRING	NULLABLE
product_name_length	INTEGER	NULLABLE
product_description_length	INTEGER	NULLABLE
product_photos_qty	INTEGER	NULLABLE
product_weight_g	INTEGER	NULLABLE
product_length_cm	INTEGER	NULLABLE
product_height_cm	INTEGER	NULLABLE
product_width_cm	INTEGER	NULLABLE



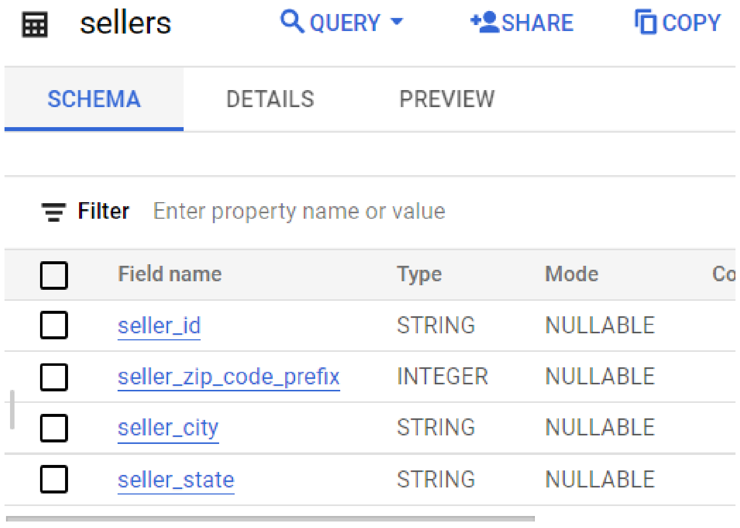
Field name	Type	Mode
customer_id	STRING	NULLABLE
customer_unique_id	STRING	NULLABLE
customer_zip_code_prefix	INTEGER	NULLABLE
customer_city	STRING	NULLABLE
customer_state	STRING	NULLABLE



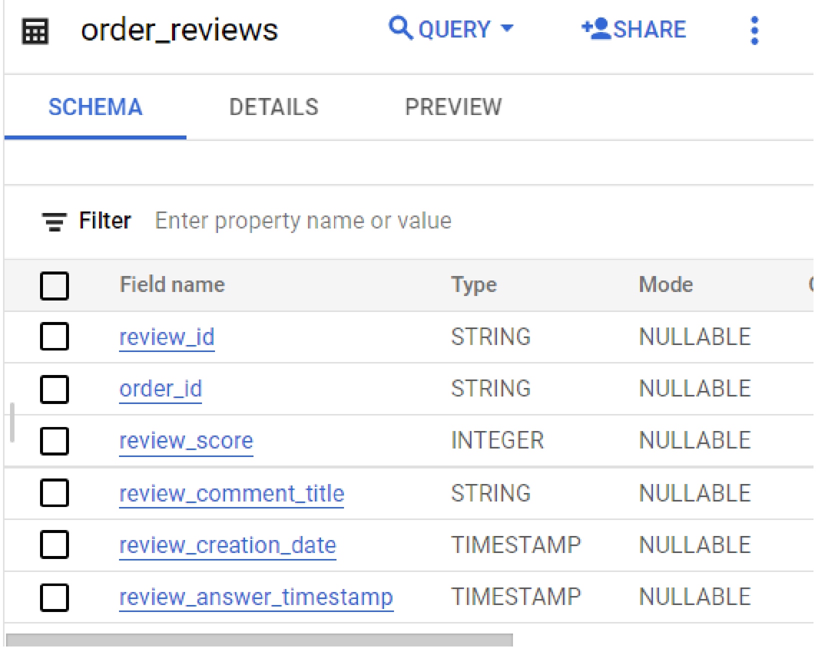
Field name	Type	Mode	Colla
order_id	STRING	NULLABLE	
order_item_id	INTEGER	NULLABLE	
product_id	STRING	NULLABLE	
seller_id	STRING	NULLABLE	
shipping_limit_date	TIMESTAMP	NULLABLE	
price	FLOAT	NULLABLE	
freight_value	FLOAT	NULLABLE	



□	Field name	Type	Mode
□	order_id	STRING	NULLABLE
□	customer_id	STRING	NULLABLE
□	order_status	STRING	NULLABLE
□	order_purchase_timestamp	TIMESTAMP	NULLABLE
□	order_approved_at	TIMESTAMP	NULLABLE
□	order_delivered_carrier_date	TIMESTAMP	NULLABLE
□	order_delivered_customer_date	TIMESTAMP	NULLABLE
□	order_estimated_delivery_date	TIMESTAMP	NULLABLE



□	Field name	Type	Mode	Co
□	seller_id	STRING	NULLABLE	
□	seller_zip_code_prefix	INTEGER	NULLABLE	
□	seller_city	STRING	NULLABLE	
□	seller_state	STRING	NULLABLE	



□	Field name	Type	Mode
□	review_id	STRING	NULLABLE
□	order_id	STRING	NULLABLE
□	review_score	INTEGER	NULLABLE
□	review_comment_title	STRING	NULLABLE
□	review_creation_date	TIMESTAMP	NULLABLE
□	review_answer_timestamp	TIMESTAMP	NULLABLE

We can see the details of the columns using the Information schema and alos we can see the details in bigquery after clicking on details

The Data types of all the columns are

1.2 Time period for which the data is given

Query

```
SELECT

MIN(order_purchase_timestamp) AS oldest_order,

MAX(order_purchase_timestamp) AS latest_order,
FROM river-clover-360718.ecommerce.orders;
```

Output:

oldest_order	latest_order
2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC

So we can see we have data from

September 4th 2016

to

October 17th 2018

We have data for a time period just above 2 years

1.3 Cities and States covered in the dataset

Query:

```
SELECT

COUNT(DISTINCT customer_state) AS Total_no_of_states

FROM river-clover-360718.ecommerce.customers;
```

Output:

Total_no_of_states
27

There are total 27 States in the given data

Now let's look at the number of cities

Query:

```
SELECT

COUNT(DISTINCT customer_city) AS Total_no_of_cities

FROM river-clover-360718.ecommerce.customers;
```

Output:

Total_no_of_cities
4119

Query:

```
SELECT
DISTINCT customer_state FROM river-clover-360718.ecommerce.customers;
```

Output:

customer_state
RN
CE
RS
SC
SP
MG
BA
RJ
GO
MA
PE
PB
ES
PR
RO
MS
PA
TO
MT
PI
AL
AM
DF
SE
RR
AP
AC

Let’s limit the output to first 10 cities as we cannot list out the all 4119 cities in this doc

Query:

```
SELECT

DISTINCT customer_city

FROM river-clover-360718.ecommerce.customers

ORDER BY customer_city

LIMIT 10;
```

customer_city
abadia dos dourados
abadiania
abaete
abaetetuba
abaiara
abaira
abare
abatia
abdon batista
abelardo luz

2) In-depth Exploration:

2.1 Is there a growing trend on e-commerce in Brazil? How can we describe a complete scenario? Can we see some seasonality with peaks at specific months?

So let’s take the month on month sales data for the given time period

Query:

```
SELECT EXTRACT(MONTH FROM order_purchase_timestamp) AS Month,

EXTRACT(YEAR FROM order_purchase_timestamp)AS Year,

COUNT(order_id) AS no_of_orders

FROM river-clover-360718.ecommerce.orders

GROUP BY Month, Year

ORDER BY Year, Month;
```

Output:

Month	Year	no_of_orders
9	2016	4
10	2016	324
12	2016	1
1	2017	800
2	2017	1780
3	2017	2682
4	2017	2404
5	2017	3700
6	2017	3245
7	2017	4026
8	2017	4331
9	2017	4285
10	2017	4631
11	2017	7544
12	2017	5673
1	2018	7269
2	2018	6728
3	2018	7211
4	2018	6939
5	2018	6873
6	2018	6167
7	2018	6292
8	2018	6512
9	2018	16
10	2018	4

2.2 What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

Here I considered Dawn as 1 am to 6 am,
Morning as 7 am to 12 pm
Afternoon as 1pm to 6 pm
Night as 7 pm to 12 am

We can see the same in Query as well

Query:

```
SELECT time_of_day, COUNT(order_id) AS no_of_orders FROM

        (SELECT order_id, order_purchase_timestamp,

        CASE WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 1 AND 6 THEN "Dawn"

        WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 7 AND 12 THEN "Morning"

        WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 13 AND 18 THEN "Afternoon"

        WHEN EXTRACT(HOUR FROM order_purchase_timestamp) IN (19,20,21,22,23,0) THEN "Night"

        END

        AS time_of_day

        FROM river-clover-360718.ecommerce.orders)

GROUP BY time_of_day;
```

Output:

time_of_day	no_of_orders
Dawn	2848
Morning	27733
Afternoon	38135
Night	30725

As we can see, Dawn is time where least number of orders takes place and afternoon I.e 1 pm to 6pm is the time where most orders take place.

So we can recommend to target that they can keep their sales or offeres according to this insight and maximize orders.

So best time to keep an sale or an offer is 1 pm to 6 pm of the day

3) Evolution of E-commerce orders in the Brazil region:

3.1 Get month on month orders by region, states

Query

```
SELECT

CONCAT(

EXTRACT(YEAR FROM order_purchase_timestamp),

"-",

RIGHT(CONCAT(00,EXTRACT(MONTH FROM order_purchase_timestamp)),2))

AS period,

customer_state,

COUNT(DISTINCT order_id) AS no_of_orders

 FROM river-clover-360718.ecommerce.orders o

 JOIN river-clover-360718.ecommerce.customers c

 ON o.customer_id = c.customer_id

 GROUP BY period, customer_state

ORDER BY period
```

This will give month on month orders for different states, we can add where condition to see the specific state output

So Instead we will order by number of orders, so we will get the period and state where maximum order took place

These are the top 10 maximum orders based on period and state

period	customer_state	no_of_orders
2018-08	SP	3253
2018-05	SP	3207
2018-04	SP	3059
2018-01	SP	3052
2018-03	SP	3037
2017-11	SP	3012
2018-07	SP	2777
2018-06	SP	2773
2018-02	SP	2703
2017-12	SP	2357

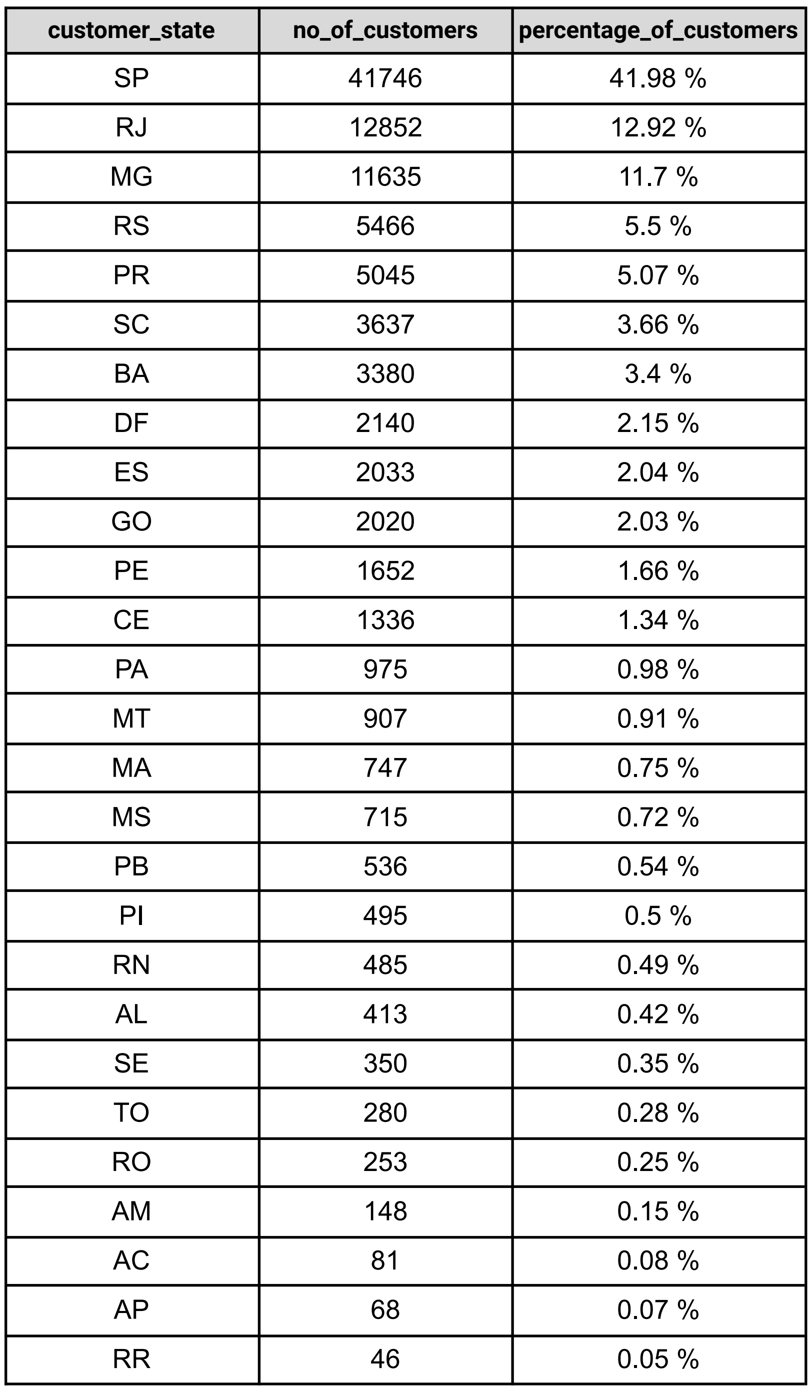
As we can see all the orders are from sau paulo state, And the maximum orders took place in the month of august

3.2 How are customers distributed in Brazil

Let’s see the state wise distribution of customers in brazil

Query:

```
WITH cte1 AS

      (SELECT customer_state, COUNT(DISTINCT customer_id) AS no_of_customers

      FROM river-clover-360718.ecommerce.customers

      GROUP BY customer_state

      ORDER BY no_of_customers DESC)


SELECT customer_state, no_of_customers,

CONCAT(ROUND((no_of_customers)*100/(SUM(no_of_customers) OVER()),2)," %") AS  percentage_of_customers

FROM cte1

ORDER BY no_of_customers DESC;
```



customer_state	no_of_customers	percentage_of_customers
SP	41746	41.98 %
RJ	12852	12.92 %
MG	11635	11.7 %
RS	5466	5.5 %
PR	5045	5.07 %
SC	3637	3.66 %
BA	3380	3.4 %
DF	2140	2.15 %
ES	2033	2.04 %
GO	2020	2.03 %
PE	1652	1.66 %
CE	1336	1.34 %
PA	975	0.98 %
MT	907	0.91 %
MA	747	0.75 %
MS	715	0.72 %
PB	536	0.54 %
PI	495	0.5 %
RN	485	0.49 %
AL	413	0.42 %
SE	350	0.35 %
TO	280	0.28 %
RO	253	0.25 %
AM	148	0.15 %
AC	81	0.08 %
AP	68	0.07 %
RR	46	0.05 %

As we can see state SP (Sao Paulo) Alone consists of almost 42% of the customers

Out of 27 states, top 5 states consists of 78% of the customers.

4) Impact on Economy: Analyze the money movemented by e-commerce by looking at order prices, freight and others.

4.1 Get % increase in cost of orders from 2017 to 2018 (include months between Jan to Aug only)

Query:

```
SELECT
EXTRACT(YEAR FROM order_purchase_timestamp) AS Year,
ROUND(SUM(payment_value),0) AS total_cost
FROM river-clover-360718.ecommerce.orders o
JOIN river-clover-360718.ecommerce.payments p
ON o.order_id = p.order_id
WHERE
EXTRACT(MONTH FROM order_purchase_timestamp) BETWEEN 1 AND 8
GROUP BY Year
ORDER BY Year;
```

Output:

Year	total_cost
2017	3669022
2018	8694734

Now let's calculate the percentage increase

Query:

```
WITH cte3 AS(
SELECT
EXTRACT(YEAR FROM order_purchase_timestamp) AS Year,
ROUND(SUM(payment_value),0) AS total_cost
FROM river-clover-360718.ecommerce.orders o
JOIN river-clover-360718.ecommerce.payments p
ON o.order_id = p.order_id
WHERE
EXTRACT(MONTH FROM order_purchase_timestamp) BETWEEN 1 AND 8
GROUP BY Year
ORDER BY Year)


SELECT Year, total_cost,
LAG(total_cost) OVER(order by Year) AS lag,
(total_cost-LAG(total_cost) OVER(order by Year))*100/LAG(total_cost) OVER(order by Year) AS per_incre
FROM cte3
```

Output:

Year	total_cost	diff	per_incre
2017	3669022		
2018	8694734	3669022	136.9768838

As we can see in the last cell, the percentage increase in total_cost from 2017 to 2018 in Jan to Aug months is almost 137%

4.2 Mean & Sum of price and freight value by customer state

Query:

```
SELECT customer_state,
ROUND(AVG(price),0) AS avg_price ,
ROUND(AVG(freight_value),0) AS avg_freight_value,
ROUND(SUM(price),0) AS total_price,
ROUND(SUM(freight_value),0) AS total_freight_value
FROM river-clover-360718.ecommerce.orders o
JOIN river-clover-360718.ecommerce.customers c ON o.customer_id = c.customer_id
JOIN river-clover-360718.ecommerce.order_items oi ON o.order_id = oi.order_id
GROUP BY customer_state
ORDER BY total_price DESC
LIMIT 10;
```

Output:

customer_state	avg_price	avg_freight_value	total_price	total_freight_value
SP	110	15	5202955	718723
RJ	125	21	1824093	305589
MG	121	21	1585308	270853
RS	120	22	750304	135523
PR	119	21	683084	117852
SC	125	21	520553	89660
BA	135	26	511350	100157
DF	126	21	302604	50625
GO	126	23	294592	53115
ES	122	22	275037	49765

As we can see the freight value is low for top cities and it increases as it goes down

5) Analysis on sales, freight and delivery time

5.1 Calculate days between purchasing, delivering and estimated delivery

5.2 Create columns:

time_to_delivery = order_purchase_timestamp-order_delivered_customer_date

diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date

5.3 Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery

The above operations are done by query below

```
SELECT
customer_state,
ROUND(
AVG(DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY)),0)
AS time_to_delivery,
ROUND(
AVG(DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY)),0) AS diff_estimated_delivery ,
ROUND(AVG(freight_value),0) AS freight_value

 FROM river-clover-360718.ecommerce.orders o
 JOIN river-clover-360718.ecommerce.customers c ON o.customer_id = c.customer_id
 JOIN river-clover-360718.ecommerce.order_items oi ON o.order_id = oi.order_id
 GROUP BY customer_state
```

Now we will apply order by and limit clauses to answer the required questions

5.4 Sort the data to get the following:

Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

Top 5 highest freight value

Query:

```
 SELECT customer_state,
ROUND(AVG(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,DAY)),0) AS time_to_delivery,
ROUND(AVG(DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY)),0) AS diff_estimated_delivery
ROUND(AVG(freight_value),0) AS freight_value
 FROM river-clover-360718.ecommerce.orders o
 JOIN river-clover-360718.ecommerce.customers c ON o.customer_id = c.customer_id
 JOIN river-clover-360718.ecommerce.order_items oi ON o.order_id = oi.order_id
 GROUP BY customer_state
 ORDER BY freight_value DESC
 LIMIT 5;
```

Output:

customer_state	time_to_delivery	diff_estimated_delivery	freight_value
PB	20	12	43
RR	28	17	43
RO	19	19	41
AC	20	20	40
PI	19	11	39

5 lowest freight value

Just remove the DESC from above query

customer_state	time_to_delivery	diff_estimated_delivery	freight_value
SP	8	10	15
PR	11	13	21
RJ	15	11	21
DF	13	11	21
MG	12	12	21

Top 5 states with highest/lowest average time to delivery

Top 5 highest

Query

```
   SELECT customer_state,
ROUND(AVG(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,DAY)),0) AS time_to_delivery,
ROUND(AVG(DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY)),0) AS diff_estimated_delivery ,
ROUND(AVG(freight_value),0) AS freight_value

 FROM river-clover-360718.ecommerce.orders o
 JOIN river-clover-360718.ecommerce.customers c ON o.customer_id = c.customer_id
 JOIN river-clover-360718.ecommerce.order_items oi ON o.order_id = oi.order_id
 GROUP BY customer_state
 ORDER BY time_to_delivery DESC
 LIMIT 5;
```

Output;

customer_state	time_to_delivery	diff_estimated_delivery	freight_value
AP	28	17	34
RR	28	17	43
AM	26	19	33
AL	24	8	36
PA	23	13	36

5 lowest delivery

Now remove desc

customer_state	time_to_delivery	diff_estimated_delivery	freight_value
SP	8	10	15
PR	11	13	21
MG	12	12	21
DF	13	11	21
RS	15	13	22

Top 5 states where delivery is really fast/ not so fast compared to estimated date

5 highest
Query

```
    SELECT customer_state,

ROUND(AVG(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp,DAY)),0) AS time_to_delivery,

ROUND(AVG(DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY)),0) AS diff_estimated_delivery ,

ROUND(AVG(freight_value),0) AS freight_value


 FROM river-clover-360718.ecommerce.orders o

 JOIN river-clover-360718.ecommerce.customers c ON o.customer_id = c.customer_id

 JOIN river-clover-360718.ecommerce.order_items oi ON o.order_id = oi.order_id

 GROUP BY customer_state

 ORDER BY diff_estimated_delivery DESC

 LIMIT 5;
```

Output:

customer_state	time_to_delivery	diff_estimated_delivery	freight_value
AC	20	20	40
AM	26	19	33
RO	19	19	41
RR	28	17	43
AP	28	17	34

These are 5 states where delivery is fast compared to estimated delivery

Now remove DESC from the query

Output:

customer_state	time_to_delivery	diff_estimated_delivery	freight_value
AL	24	8	36
SE	21	9	37
MA	21	9	38
SP	8	10	15
BA	19	10	26

These are the top 5 states where difference between estimated delivery and

6) Payment type analysis:

6.1 Month over Month count of orders for different payment types

First let’s see which payment type is used most

Query:

```
SELECT payment_type, COUNT(DISTINCT order_id) AS no_of_orders
FROM river-clover-360718.ecommerce.payments
GROUP BY payment_type
ORDER BY no_of_orders DESC;
```

Output:

payment_type	no_of_orders
credit_card	76505
UPI	19784
voucher	3866
debit_card	1528
not_defined	3

As we can see majority of people use credit card as their payment type, It is 75% of the orders.

So Target can provide some offers on credit cards or exclusive vouchers which makes them buy again on target since it cannot be used anywhere.

Query

```
WITH cte2 AS(
SELECT
CONCAT(
EXTRACT(YEAR FROM order_purchase_timestamp),
"-",
RIGHT(CONCAT(00,EXTRACT(MONTH FROM order_purchase_timestamp)),2))
AS period,
payment_type, COUNT(DISTINCT o.order_id) AS no_of_orders
FROM river-clover-360718.ecommerce.orders o
JOIN river-clover-360718.ecommerce.payments p
ON o.order_id = p.order_id
GROUP BY period,payment_type
ORDER BY period);
```

Output:

period	credit_card	UPI	voucher	debit_card
2016-09	3			
2016-10	253	63	11	2
2016-12	1			
2017-01	582	197	33	9
2017-02	1347	398	69	13
2017-03	2008	590	123	31
2017-04	1835	496	115	27
2017-05	2833	772	171	30
2017-06	2452	707	142	27
2017-07	3072	845	205	22
2017-08	3272	938	198	34
2017-09	3274	903	174	43
2017-10	3510	993	208	52
2017-11	5867	1509	267	70
2017-12	4363	1160	220	64
2018-01	5511	1518	304	109
2018-02	5235	1325	219	69
2018-03	5674	1352	272	78
2018-04	5441	1287	238	97
2018-05	5475	1263	203	51
2018-06	4796	1100	231	181
2018-07	4738	1229	212	242
2018-08	4963	1139	232	277
2018-09			15	
2018-10			4	

There is significant growth in the usage of credit card and UPI over the timeperiod, apart from the last

6.2 Distribution of payment installments and count of orders

Query:

```
SELECT payment_installments, COUNT(order_id) AS no_of_orders

FROM  river-clover-360718.ecommerce.payments

GROUP BY payment_installments
```

Output:

payment_installments	no_of_orders
0	2
1	52546
2	12413
3	10461
4	7098
5	5239
6	3920
7	1626
8	4268
9	644
10	5328
11	23
12	133
13	16
14	15
15	74
16	5
17	8
18	27
20	17
21	3
22	1
23	1
24	18

As we can see people tend to finish their payments on first installment only

50% of the orders are paid in 1 installment

Around 85% of the orders are paid within 6 installmnts

Insights

There is a seasonality in drop in number of orders in month of June in both the years, And there is a overall increase in trend on number of orders as a whole for the given time period

Most of the orders are placed in the time 1 PM to 6 PM, which is 38.35% of the total orders. And the least orders are places in the time period of 1 AM to 6AM which is 2.86% of the total orders

When we see customers distribution across states in brazil 42% of the customers are from SP (Sao Paulo state). Out of the 27 states in Brazil, the top 5 states consists of almost 80% of the the customers of target.

There is an almost 137% increase in the total payment value from the year 2017(Jan to Aug) to 2018(Jan to Aug) in the months of January to August as we don't have significant data after august in 2018.

Average price is highest in the state PB (**Paraíba**) along with the highest freight_value

Average price is lowest in the state SP (Sao Paulo) along with the lowest freight value

Average time to delivery is highest in the state AP

Average time to delivery is lowest in the state SP

Average difference between estimated delivery and actual delivery is highest in state AC, which means delivery is relly fast, compared to estimated delivery

Average difference between estimated delivery and actual delivery is lowest in state AL, which means delivery is not so fast, compared to estimated delivery

75% of the customers use credit card as their payment type

And 20% of the customers use UPI as their payment type

Only 3.5% of the customers use Vouchers to pay their orders

And only 1.5% of the customers use debit cars to pay their orders

Over the period there is significant growth in usage of credit card payments and UPI payments on target platform, while Debit card and Voucher payments are not increasing

Around 50 % of the orders are paid within 1 installment only

While 85% of the orders are paid within the first 6 installments of the payments, So most customers tend to pay their orders within the first 6 months

Recommendations

Target has to focus on tactics to tackle a seasonal drop in sales in month of June

Most of the orders are placed in the timeperiod of 1 PM to 6 PM of the day, so target can launch their exclusive sales offers in this time to attract most number of customers and maximize their revenue.

And 1 AM to 6 AM is not a good time to launch any products or sale offers, because only 2-3% of the orders are placed in this time period

42% of the customers are from state SP Sao Paulo, So target can focus on special regional offers for this state as that can maximize their revenue

Target can collaborate with credit card companies and offer casbacks or vouchers which bring back them to buy again on target website since 75% of the orders are paid using credit card method

Around 85% of the customers tend to pay their orders within the first 6 months, So target can consider No cost EMI’s as that will increase the number of customers finishing their payments within first 6 months even more

There is a 137% of increase in total revenue for target over the past year, So Target should continue their strategies for advertising and customer retention tactics, So that they can Increase their revenue even more in the coming years