

```
-- from, where, group by, having, select order by , limit
SELECT *
FROM target.customers
LIMIT 10;
```

```
SELECT *
FROM target.order_items
LIMIT 10;
```

```
SELECT *
FROM
target.orders
LIMIT 10;
```

```
SELECT *
FROM
target.products
LIMIT 10;
```

```
SELECT *
FROM
target.payments
LIMIT 10;
```

```
--LEVEL OF THE TABLE
```

```
select count(*) as all_count from target.order_items;
```

```
select count(*) from
(select order_id, order_item_id from target.order_items group by 1,2);
```

```
/*
Data type of columns in a table
*/
DESCRIBE TABLE target.customers;
```

```

/*
Get the time period for which the data is given
*/
SELECT
MIN(order_purchase_timestamp) AS first_order,
MAX(order_purchase_timestamp) AS last_order
FROM target.orders;

/*
Number of cities and states in our dataset
*/
SELECT
COUNT(DISTINCT (geolocation_city)) AS city_count,
COUNT(DISTINCT (geolocation_state)) AS state_count
FROM target.geolocations;

/*
Is there a growing trend in e-commerce in Brazil? How can we describe a
complete scenario?
*/
--chart
SELECT
EXTRACT(year FROM timestamp(order_purchase_timestamp)) AS year,
EXTRACT(month FROM timestamp(order_purchase_timestamp)) AS month,
COUNT(1) AS num_orders
FROM target.orders GROUP BY year, month ORDER BY year, month;

/*
Can we see some seasonality with peaks at specific months?
*/
--chart
SELECT
EXTRACT(month FROM timestamp(order_purchase_timestamp)) AS month,
COUNT(1) AS num_orders

FROM target.orders
GROUP BY 1
ORDER BY 1;

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

SELECT CASE

```

```

WHEN EXTRACT(hour FROM timestamp(order_purchase_timestamp)) BETWEEN 0
AND 6
THEN 'dawn'
WHEN EXTRACT(hour FROM timestamp(order_purchase_timestamp)) BETWEEN 7
AND 12
THEN 'morning' WHEN
EXTRACT(hour FROM timestamp(order_purchase_timestamp)) BETWEEN 13
AND 18
THEN 'afternoon' WHEN
EXTRACT(hour FROM timestamp(order_purchase_timestamp)) BETWEEN 19
AND 23
THEN 'night'
END AS time_of_day,
COUNT(DISTINCT order_id) AS counter
FROM target.orders
GROUP BY 1 ORDER BY 2 DESC;

```

```

/*
Get month on month orders by region/ state.
*/
--chart
--focus on tables to join
SELECT
EXTRACT(month FROM timestamp(order_purchase_timestamp)) AS month,
g.geolocation_state,

COUNT(1) AS num_orders
FROM target.orders o
INNER JOIN target.customers c
ON o.customer_id = c.customer_id
INNER JOIN target.geolocations g
ON c.customer_zip_code_prefix = g.geolocation_zip_code_prefix
GROUP BY g.geolocation_state, month
ORDER BY geolocation_state DESC, month ASC;

```

```

/*
How are customers distributed in Brazil
*/
SELECT
g.geolocation_state, COUNT(DISTINCT (c.customer_unique_id)) AS
num_customers
FROM target.customers c
INNER JOIN target.geolocations g
ON c.customer_zip_code_prefix = g.geolocation_zip_code_prefix
GROUP BY g.geolocation_state

```

```
ORDER BY num_customers DESC;
```

```
/*
```

```
Analyze the money movement by e-commerce by looking at order prices,  
freight and others.
```

```
Create CTE Table and new columns:
```

```
price_per_order = sum(price)/count(order_id) freight_per_order=  
sum(freight_value)/count(order_id) Group the data on yearly and monthly  
level
```

```
--chart
```

```
*/
```

```
WITH
```

```
cte_table AS (
```

```
SELECT
```

```
EXTRACT(month FROM timestamp(o.order_purchase_timestamp)) AS month,  
EXTRACT(year FROM timestamp(o.order_purchase_timestamp)) AS year,  
(sum(price) / COUNT( distinct o.order_id)) AS price_per_order,  
(sum(freight_value) / COUNT(distinct o.order_id)) AS
```

```
freight_per_order
```

```
FROM target.orders o
```

```
INNER JOIN target.order_items i
```

```
ON o.order_id = i.order_id
```

```
GROUP BY year, month
```

```
)
```

```
SELECT (price_per_order), (freight_per_order), month, year
```

```
FROM cte_table
```

```
order by year asc, month asc ;
```

```
/*
```

```
Total amount sold in 2017 between Jan to august (Jan to Aug because data  
is available starting 2017 01 to 2018 08) and we can only compare cycles  
with cycles
```

```
Compare YoY at a monthly level
```

```
*/
```

```
WITH
```

```
cte_table AS (
```

```
SELECT
```

```
EXTRACT(month FROM timestamp(order_purchase_timestamp)) AS month,  
EXTRACT(year FROM timestamp(order_purchase_timestamp)) AS year,  
sum(price) AS total_price,
```

```
sum(freight_value) AS total_freight
```

```
FROM target.orders o
```

```
INNER JOIN target.order_items i
```

```
ON o.order_id = i.order_id
```

```

GROUP BY year, month
ORDER BY year ASC, month ASC
)
SELECT
month, price_2017, price_2018,
round((price_2018 - price_2017) / price_2017 * 100, 2) AS yoy
FROM
(
SELECT
month,
sum(CASE WHEN year = 2017 THEN total_price ELSE 0 END) AS
price_2017,
sum(CASE WHEN year = 2018 THEN total_price ELSE 0 END) AS
price_2018
-- sum(total_price) as total_transaction_amt

FROM cte_table
WHERE (year = 2017 OR year = 2018) AND month BETWEEN 1 AND 8
GROUP BY month
order by month
);

/*
MoM increase for year 2017
*/

SELECT
month, orders, lagged_orders,
(orders - coalesce(lagged_orders, 0)) / coalesce(lagged_orders, 1) *
100
AS difference
FROM
(
SELECT *, lag(orders, 1) OVER (ORDER BY month ASC) AS lagged_orders
FROM
(
month,

```

2017

```
SELECT
EXTRACT(month FROM timestamp(a.order_purchase_timestamp)) AS

COUNT(DISTINCT a.order_id) AS orders,
COUNT(DISTINCT b.customer_unique_id) AS customers
FROM target.orders a
LEFT JOIN target.customers b
ON a.customer_id = b.customer_id
WHERE EXTRACT(year FROM timestamp(a.order_purchase_timestamp)) =

GROUP BY 1

) base
) base_2
ORDER BY month ASC;
```

```
/*
Sum and mean price by customer state
```

It's very interesting to see how some states have a high total amount sold and a low price per order.

If we look at SP (São Paulo) for example, it's possible to see that it is the state with most valuable state for e-commerce (5202955 sold) but it is also where customers pay less per order (125.75 per order)

```
*/
with cte_table as (
select
c.customer_state as state,
sum(price) as total_price, count(distinct(o.order_id)) as num_orders
from target.orders o
inner join target.order_items i
on o.order_id= i.order_id
inner join target.customers c on o.customer_id=c.customer_id group by
state
)
select state, total_price, num_orders, (total_price/num_orders) as
avg_price
from cte_table
order by total_price desc;
```

```

/*
Analysis on sales, freight and delivery time
create new columns for time to delivery and difference in estimated vs
actual delivery
*/

```

```

--some data type issue with order_delivered_customer_date
SELECT
order_id,
date_DIFF( date(order_estimated_delivery_date),
date(order_purchase_timestamp), DAY) AS time_to_del,
-- TIMESTAMP_DIFF(
-- timestamp(order_delivered_customer_date),
-- timestamp(order_estimated_delivery_date),

-- DAY) AS diff_estimated_dil
FROM target.orders
WHERE order_status = 'delivered';

```

```

/*Top 5 states with highest/lowest average time to delivery
*/

```

```

select g.geolocation_state as state,
SUM(TIMESTAMP_DIFF(
timestamp(order_estimated_delivery_date),timestamp(order_purchase_timest
amp), DAY))/COUNT(ORDER_ID) AS avg_dil_time,
from target.orders o
inner join target.customers c
on o.customer_id=c.customer_id
inner join target.geolocations g
on c.customer_zip_code_prefix=g.geolocation_zip_code_prefix
where order_status='delivered' group by state
order by avg_dil_time
limit 5;

```