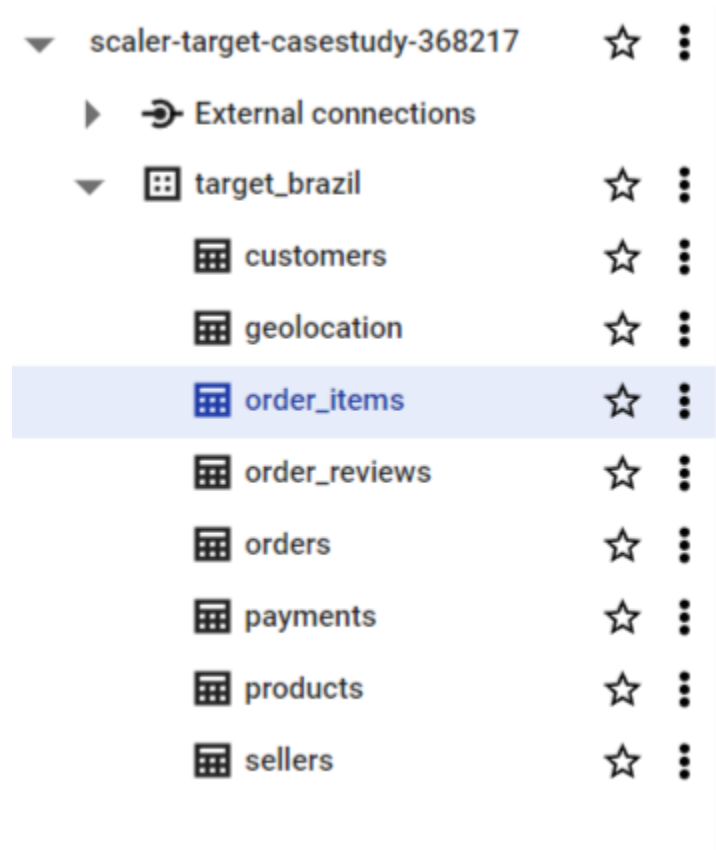


What 'good' looks like?

1. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset
 1. Data type of columns in a table
 2. Time period for which the data is given - [September 2016 to 2018](#)
 3. Cities and States covered in the dataset - [Cities of Brazil](#)



Ans 1:

1. Data Types -

- a. **Customers table** - This table contains data in String (Varchar/char) and Integer(Number)

This table contains all the information related to the customer like customer_id, customer_unique_id, customer_zip_code_prefix, customer_city and customer_state.

- b. **Orders table** - This table contains in String and Timestamp

This table contains information about the order like order id, customers id, status of the order, time of purchase, time of order approval, order delivery date by carrier, order received date of customer, estimated date of delivery.

- c. **Order items table** - This table contains data in String, Integer, Timestamp and Float data types.

This table contains information about the items that were ordered like unique order id, unique order item id, unique products id, unique sellers id, price, freight (shipping) charges, shipping last date(limit date)

- d. **Sellers table** - This table contains data in String and Integer data types.

This table contains information about the seller. The seller is given a unique ID called seller_id, their zip code, city and state is stored in the table.

- e. **Payments table** - This table contains data in String, Integer and Float data types.

This table contains payment information. Payment information is stored based on the unique order id each order has.

The table has payment type, installment, value and sequential number for the emi payments.

- f. **Products table** - This table contains information in String and Integer datatype.

This table stores information about all the products. It has unique product id, product category, name of product, description of product, product photos, product weight in gms, product length in cms, product height in cms and product width in cms.

- g. **Geolocation table** - This table contains information integer, float and string format.

This table stores information in geolocation zip code, geolocation latitude, geolocation longitude, geolocation city, geolocation state.

2. In-depth Exploration:

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?

Ans: `Select * from (`

```
SELECT count(*) as cnt, extract(month from order_purchase_timestamp) as mon,
extract(year from order_purchase_timestamp) as yr FROM
`saler-target-casestudy-368217.target_brazil.orders`
```

```
GROUP BY extract(month from order_purchase_timestamp), extract(year from
order_purchase_timestamp)
```

```
)x
```

```
ORDER BY yr, mon;
```

Analysis - Yes, there is a growing trend of ecommerce in Brazil. The peak season is around March and year end - December. It maybe because of the festivals like Good Friday, Easter, Christmas, Black friday, etc.

Google Cloud | Scaler-Target-Casestudy | Search Products, resources, docs (/)

SANDBOX Set up billing to upgrade to the full BigQuery experience. [Learn more](#) DISMISS UPGRADE

Explorer + ADD DATA | orders - x | *Unsaved query - x

Q Type to search

Viewing all resources. [Show starred resources only.](#)

scaler-target-casestudy-368217

- External connections
- target_brazil
 - customers
 - geolocation
 - order_items
 - order_reviews
 - orders
 - payments
 - products
 - sellers

Query results

1 Select * from (
2 SELECT count(*) as cnt, extract(month from order_purchase_timestamp) as mon, extract(year from order_purchase_timestamp) as yr FROM
3 scaler-target-casestudy-368217.target_brazil.orders
4)x
5 ORDER BY yr, mon;

Press Alt+F1 for Accessibility Options

| JOB INFORMATION | | RESULTS | | JSON | EXECUTION DETAILS | EXECUTION GRAPH | PREVIEW |
|-----------------|------|---------|------|------|-------------------|-----------------|---------|
| Row | cnt | mon | yr | | | | |
| 1 | 4 | 9 | 2016 | | | | |
| 2 | 324 | 10 | 2016 | | | | |
| 3 | 1 | 12 | 2016 | | | | |
| 4 | 800 | 1 | 2017 | | | | |
| 5 | 1780 | 2 | 2017 | | | | |
| 6 | 2682 | 3 | 2017 | | | | |
| 7 | 2404 | 4 | 2017 | | | | |
| 8 | 3700 | 5 | 2017 | | | | |
| 9 | 3245 | 6 | 2017 | | | | |
| 10 | 4026 | 7 | 2017 | | | | |

Results per page: 50 1 - 25 of 25 |< < > >|

PERSONAL HISTORY PROJECT HISTORY REFRESH

2.

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

Ans: Mostly people visit the store in the Night and at Dawn.

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

1. Get month on month orders by region, states

Ans.

```
select *, count(*)as cnt  from (  
  
select customer_city, customer_state, extract(year from  
order_purchase_timestamp) as yr,  extract(month from  
order_purchase_timestamp) as mon from  
  
`target_brazil.customers` c  
  
JOIN  
  
`target_brazil.orders` o  
  
ON  
  
c.customer_id = o.customer_id  
  
) x  
  
GROUP BY yr, mon, customer_state, customer_city  
  
ORDER BY 3,4
```

```
1 select *, count(*)as cnt  from (  
2 select customer_city, customer_state, extract(year from order_purchase_timestamp) as yr,  extract(month from order_purchase_timestamp) as mon from  
3 `target_brazil.customers` c  
4 JOIN  
5 `target_brazil.orders` o  
6 ON  
7 c.customer_id = o.customer_id  
8 ) x  
9 GROUP BY yr, mon, customer_state, customer_city  
0 ORDER BY 3,4
```

Press Alt+F1 for Accessibility Options

Query results

[SAVE RESULTS](#)

[EXPLORE DATA](#)

| OB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | EXECUTION GRAPH | PREVIEW |
|----------------|----------------------|----------------|-------------------|-----------------|---------|
| | customer_city | customer_state | yr | mon | cnt |
| 1 | boa vista | RR | 2016 | 9 | 1 |
| 2 | passo fundo | RS | 2016 | 9 | 1 |
| 3 | sao jose dos campos | SP | 2016 | 9 | 1 |
| 4 | sao joaquim da barra | SP | 2016 | 9 | 1 |
| 5 | itu | SP | 2016 | 10 | 1 |
| 6 | poa | SP | 2016 | 10 | 1 |
| 7 | bage | RS | 2016 | 10 | 1 |
| 8 | iacu | BA | 2016 | 10 | 1 |

2. How are customers distributed in Brazil

Ans.: The customers are mostly spread around in Brazil. Mostly the customers are from sao paulo, rio de janeiro, belo horizonte.

```
select *, count(*)as cnt  from (  
  
select customer_city, customer_state from  
  
`target_brazil.customers` c  
  
JOIN  
  
`target_brazil.orders` o  
  
ON  
  
c.customer_id = o.customer_id  
  
) x  
  
GROUP BY customer_state, customer_city  
  
ORDER BY 3 desc
```

[RUN](#) [SAVE](#) [SHARE](#) [SCHEDULE](#) [MORE](#)

```
1 select *, count(*)as cnt  from (  
2 select customer_city, customer_state from  
3 `target_brazil.customers` c  
4 JOIN  
5 `target_brazil.orders` o  
6 ON  
7 c.customer_id = o.customer_id  
8 ) x  
9 GROUP BY customer_state, customer_city  
10 ORDER BY 3 desc
```

Press Alt+F1 for Accessibility Option

Query results

[SAVE RESULTS](#) [EXPLORE DATA](#)

| JOB INFORMATION | | RESULTS | JSON | EXECUTION DETAILS | EXECUTION GRAPH | PREVIEW |
|-----------------|----------------|----------------|-------|-------------------|-----------------|---------|
| Row | customer_city | customer_state | cnt | | | |
| 1 | sao paulo | SP | 15540 | | | |
| 2 | rio de janeiro | RJ | 6882 | | | |
| 3 | belo horizonte | MG | 2773 | | | |
| 4 | brasilia | DF | 2131 | | | |
| 5 | curitiba | PR | 1521 | | | |
| 6 | campinas | SP | 1444 | | | |

Results per page: 50

1 – 50 of 4310

|< < > >|

4. Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.
1. Get % increase in cost of orders from 2017 to 2018 (include months between Jan to Aug only)

Ans.

```
select *, price+freight_value as c_cost, lp+lfv as o_cost,
((price+freight_value) - (lp+lfv))*100 /(lp+lfv) as inc_percentage

from (

SELECT *, lag(price) over(partition by product_id order by extract(year
from shipping_limit_date),extract(month from shipping_limit_date) ) as
lp,

lag(freight_value) over(partition by product_id order by extract(year
from shipping_limit_date), extract(month from shipping_limit_date)) as
lfv

FROM `scaler-target-casestudy-368217.target_brazil.order_items`

where

extract(year from shipping_limit_date) in (2017, 2018)

and

extract(month from shipping_limit_date) in (1, 2, 3, 4, 5, 6, 7, 8)

)
```

customers

*Unsaved query

*Unsaved query 2

RUN

SAVE

SHARE

SCHEDULE

MORE

1 select *, price+freight_value as c_cost, lp+lfv as o_cost, ((price+freight_value) - (lp+lfv))*100/(lp+lfv) as inc_percentage

2 from

Press Alt+F1 for Accessibility Options

Query results

SAVE RESULTS

EXPLORE DATA

JOB INFORMATION

RESULTS

JSON

EXECUTION DETAILS

EXECUTION GRAPH

PREVIEW

| Row | seller_id | shipping_limit_date | price | freight_value | lp | lfv | c_cost | o_cost | inc_percentage |
|-----|-------------------------------|-------------------------|--------|---------------|--------|-------|--------|--------|----------------|
| 1 | 39d54ff918774174706fb065d... | 2018-08-03 11:24:15 UTC | 39.5 | 16.47 | null | null | 55.97 | null | null |
| 2 | 1127b7f2594683f2510f1c2c83... | 2018-01-23 03:35:34 UTC | 116.89 | 13.37 | null | null | 130.26 | null | null |
| 3 | cb6c9f5888a7a090c75beaf61... | 2018-08-15 21:44:16 UTC | 114.9 | 44.72 | 116.89 | 13.37 | 159.62 | 130.26 | 22.5395363... |
| 4 | cb6c9f5888a7a090c75beaf61... | 2018-08-08 12:05:14 UTC | 114.9 | 27.78 | 114.9 | 44.72 | 142.68 | 159.62 | -10.612705... |
| 5 | 7b07b3c7487f0ea825fc6df75a... | 2017-06-05 04:22:39 UTC | 44.9 | 7.78 | null | null | 52.68 | null | null |
| 6 | 640e21a7d01df7614a3b4923e... | 2018-05-03 21:52:36 UTC | 40.35 | 13.71 | null | null | 54.06 | null | null |
| 7 | 640e21a7d01df7614a3b4923e... | 2018-05-04 19:52:08 UTC | 40.35 | 17.43 | 40.35 | 13.71 | 57.78 | 54.06 | 6.88124306... |
| 8 | 640e21a7d01df7614a3b4923e... | 2018-06-04 11:50:12 UTC | 40.35 | 19.32 | 40.35 | 17.43 | 59.67 | 57.78 | 3.27102803... |
| 9 | 640e21a7d01df7614a3b4923e... | 2018-06-07 15:15:14 UTC | 40.35 | 5.89 | 40.35 | 19.32 | 46.24 | 59.67 | -22.507122... |
| 10 | 640e21a7d01df7614a3b4923e... | 2018-06-26 13:38:35 UTC | 40.35 | 13.86 | 40.35 | 5.89 | 54.21 | 46.24 | 17.2361591... |
| 11 | 640e21a7d01df7614a3b4923e... | 2018-07-16 22:35:19 UTC | 40.35 | 19.47 | 40.35 | 13.86 | 59.82 | 54.21 | 10.3486441... |
| 12 | 640e21a7d01df7614a3b4923e... | 2018-07-05 12:10:14 UTC | 40.35 | 19.47 | 40.35 | 19.47 | 59.82 | 59.82 | 0.0 |

Results per page: 50 1 - 50 of 87183

2. Mean & Sum of price and freight value by customer state

Ans:

```
select sum(price) as p, sum(freight_value) as f, avg(price) as av_p,  
avg(freight_value) as af, customer_state  
  
from  
  
`target_brazil.order_items` oi  
  
join  
  
`target_brazil.orders` o  
  
ON  
  
oi.order_id = o.order_id  
  
join `target_brazil.customers` c  
  
ON o.customer_id = c.customer_id  
  
GROUP BY customer_state  
  
ORDER BY customer_state
```

customers ▾ × *Unsaved query ▾ × *Unsaved query 2 ▾ × *Unsaved query 3 ▾ × +

RUN **SAVE** ▾ **SHARE** ▾ **SCHEDULE** ▾ **MORE** ▾ ✓ This query will process 15.42 MB when run.

```

1 select sum(price) as p, sum(freight_value) as f, avg(price) as av_p, avg(freight_value) as af, customer_state
2 from
3 `target_brazil.order_items` oi
4 join
5 `target_brazil.orders` o
6 ON
7 oi.order_id = o.order_id
8 join `target_brazil.customers` c
9 ON o.customer_id = c.customer_id
10 GROUP BY customer_state
11 ORDER BY customer_state

```

Press Alt+F1 for Accessibility Options

Query results **SAVE RESULTS** ▾ **EXPLORE DATA** ▾ ↕

| JOB INFORMATION | | RESULTS | JSON | EXECUTION DETAILS | | EXECUTION GRAPH | PREVIEW |
|-----------------|---------------|---------------|---------------|-------------------|----------------|-----------------|---------|
| Row | p | f | av_p | af | customer_state | | |
| 1 | 15982.9499... | 3686.75000... | 173.727717... | 40.0733695... | AC | | |
| 2 | 80314.8099... | 15914.5899... | 180.889211... | 35.8436711... | AL | | |
| 3 | 22356.8400... | 5478.89000... | 135.495999... | 33.2053939... | AM | | |
| 4 | 13474.2999... | 2788.50000... | 164.320731... | 34.0060975... | AP | | |
| 5 | 511349.990... | 100156.679... | 134.601208... | 26.3639589... | BA | | |
| 6 | 227254.709... | 48351.5899... | 153.758261... | 32.7142016... | CE | | |
| 7 | 302603.939... | 50625.4999... | 125.770548... | 21.0413549... | DF | | |

Results per page: 50 ▾ 1 - 27 of 27 |< < > >|

5. Analysis on sales, freight and delivery time

1. Calculate days between purchasing, delivering and estimated delivery

Ans.: select *,

date_diff(order_purchase_timestamp,order_delivered_customer_date, DAY) as
time_to_delivery ,

date_diff(order_estimated_delivery_date, order_delivered_customer_date,
DAY) as diff_estimated_delivery

from

`target_brazil.orders`

2. Create columns:

1. time_to_delivery =

order_purchase_timestamp-order_delivered_customer_date

2. diff_estimated_delivery =

order_estimated_delivery_date-order_delivered_customer_date

Ans.

```
1 select *,
2   date_diff(order_purchase_timestamp,order_delivered_customer_date, DAY) as time_to_delivery ,
3   date_diff(order_estimated_delivery_date, order_delivered_customer_date, DAY) as diff_estimated_delivery
4 from
5   target_brazil.orders
```

Press Alt+F1 for Accessibility Options

Query results

SAVE RESULTSEXPLORE DATA

JOB INFORMATIONRESULTSJSONEXECUTION DETAILSEXECUTION GRAPHPREVIEW

| Row | Timestamp | order_approved_at | order_delivered_carrier_date | order_delivered_customer_date | order_estimated_delivery_date | time_to_delivery | diff_estimated_delivery |
|-----|-----------|-------------------------|------------------------------|-------------------------------|-------------------------------|------------------|-------------------------|
| 1 | 1:52 UTC | 2018-02-19 20:56:05 UTC | 2018-02-20 19:57:13 UTC | 2018-03-21 22:03:51 UTC | 2018-03-09 00:00:00 UTC | -30 | -12 |
| 2 | 1:56 UTC | 2016-10-10 10:40:49 UTC | 2016-10-14 10:40:50 UTC | 2016-11-09 14:53:50 UTC | 2016-12-08 00:00:00 UTC | -30 | 28 |
| 3 | 1:41 UTC | 2016-10-04 10:18:57 UTC | 2016-10-25 12:14:28 UTC | 2016-11-08 10:58:34 UTC | 2016-11-25 00:00:00 UTC | -35 | 16 |
| 4 | 1:38 UTC | 2017-04-15 15:45:14 UTC | 2017-04-27 16:06:59 UTC | 2017-05-16 14:49:55 UTC | 2017-05-18 00:00:00 UTC | -30 | 1 |
| 5 | 1:54 UTC | 2017-04-15 22:30:19 UTC | 2017-04-17 09:08:52 UTC | 2017-05-17 10:52:15 UTC | 2017-05-18 00:00:00 UTC | -32 | 0 |
| 6 | 1:13 UTC | 2017-04-16 15:05:14 UTC | 2017-04-17 10:44:19 UTC | 2017-05-16 09:07:47 UTC | 2017-05-18 00:00:00 UTC | -29 | 1 |
| 7 | 1:24 UTC | 2017-04-08 21:30:16 UTC | 2017-04-25 10:53:00 UTC | 2017-05-22 14:11:31 UTC | 2017-05-18 00:00:00 UTC | -43 | -4 |
| 8 | 1:45 UTC | 2017-04-11 20:02:27 UTC | 2017-04-12 14:47:39 UTC | 2017-05-22 16:18:42 UTC | 2017-05-18 00:00:00 UTC | -40 | -4 |

Results per page: 501 - 50 of 99441

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

Ans:

```

select customer_state, avg(freight_value) as avg_freight,
time_to_delivery, diff_estimated_delivery

from (

select customer_state, freight_value,

date_diff(order_purchase_timestamp,order_delivered_customer_date, DAY) as
time_to_delivery ,

date_diff(order_estimated_delivery_date, order_delivered_customer_date,
DAY) as diff_estimated_delivery

from

`target_brazil.order_items` oi

join

`target_brazil.orders` o

ON

oi.order_id = o.order_id

join `target_brazil.customers` c

```

ON o.customer_id = c.customer_id

) x

GROUP BY customer_state, x.time_to_delivery,x.diff_estimated_delivery

```
1 select customer_state, avg(freight_value) as avg_freight, time_to_delivery, diff_estimated_delivery
2 from (
3 select customer_state, freight_value
```

Press Alt+F1 for Accessibility Option

Query results

SAVE RESULTS EXPLORE DATA

| JOB INFORMATION | | RESULTS | JSON | EXECUTION DETAILS | | EXECUTION GRAPH | PREVIEW |
|-----------------|----------------|---------------|------------------|-------------------|--|-----------------|---------|
| row | customer_state | avg_freight | time_to_delivery | diff_estimated_c | | | |
| 27 | RJ | 18.5733333... | -7 | 45 | | | |
| 28 | MG | 19.215 | -30 | -12 | | | |
| 29 | SC | 18.51 | -30 | 28 | | | |
| 30 | SP | 12.5250000... | -7 | 44 | | | |
| 31 | RJ | 20.882 | -10 | 41 | | | |
| 32 | RJ | 14.11 | -35 | 16 | | | |
| 33 | GO | 18.72 | -23 | 9 | | | |
| 34 | SP | 10.1125 | -12 | -5 | | | |
| 35 | RS | 25.5651428... | -12 | 12 | | | |
| 36 | SP | 16.1740384... | -7 | 1 | | | |
| 37 | SP | 17.5123809... | -12 | 9 | | | |
| 38 | SP | 10.7275000... | -1 | 5 | | | |
| 39 | SP | 11.0014285... | -6 | 0 | | | |

4. Sort the data to get the following:
1. Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5 — For desc, just add desc next to order by 1

```
Ans: select avg(freight_value) as avg_freight, customer_state
      from
      `target_brazil.order_items` oi
      join
      `target_brazil.orders` o
      ON
      oi.order_id = o.order_id
      join `target_brazil.customers` c
      ON o.customer_id = c.customer_id
      GROUP BY customer_state
      ORDER BY 1
      LIMIT 5
```

```
1 select avg(freight_value) as avg_freight, customer_state
2 from
3 `target_brazil.order_items` oi
4 join
5 `target_brazil.orders` o
6 ON
7 oi.order_id = o.order_id
8 join `target_brazil.customers` c
9 ON o.customer_id = c.customer_id
10 GROUP BY customer_state
11 ORDER BY 1
12
```

Press Alt+F1 for Accessibility Option

Query results

[SAVE RESULTS](#)

[EXPLORE DATA](#)



| JOB INFORMATION | | RESULTS | JSON | EXECUTION DETAILS | EXECUTION GRAPH | PREVIEW |
|-----------------|---------------|----------------|------|-------------------|-----------------|---------|
| row | avg_freight | customer_state | | | | |
| 1 | 15.1472753... | SP | | | | |
| 2 | 20.5316515... | PR | | | | |
| 3 | 20.6301668... | MG | | | | |
| 4 | 20.9609239... | RJ | | | | |
| 5 | 21.0413549... | DF | | | | |

2. Top 5 states with highest/lowest average time to delivery

Ans:

```
select customer_state, avg(time_to_delivery) as avg_actual
from (
select customer_state,
date_diff(order_purchase_timestamp,order_delivered_customer_date, DAY) as
time_to_delivery ,
date_diff(order_estimated_delivery_date, order_delivered_customer_date, DAY) as
diff_estimated_delivery
from
`target_brazil.order_items` oi
join
`target_brazil.orders` o
ON
oi.order_id = o.order_id
join `target_brazil.customers` c
ON o.customer_id = c.customer_id
) x
WHERE time_to_delivery IS NOT NULL
GROUP BY customer_state ORDER BY 2 desc LIMIT 5
```

```

1 select customer_state, avg(time_to_delivery) as avg_actual
2 from (
3 select customer_state,
4 date_diff(order_purchase_timestamp,order_delivered_customer_date, DAY) as time_to_delivery ,
5 date_diff(order_estimated_delivery_date, order_delivered_customer_date, DAY) as diff_estimated_delivery
6 from
7 `target_brazil.order_items` oi
8 join
9 `target_brazil.orders` o
10 ON
11 oi.order_id = o.order_id
12 join `target_brazil.customers` c
13 ON o.customer_id = c.customer_id
14 ) x
15 WHERE time_to_delivery IS NOT NULL
16 GROUP BY customer_state ORDER BY 2 desc LIMIT 5

```

Press Alt+F1 for Accessibility Options

Query results

[SAVE RESULTS](#)

[EXPLORE DATA](#)



JOBS INFORMATION **RESULTS** JSON EXECUTION DETAILS EXECUTION GRAPH **PREVIEW**

| # | customer_state | avg_actual |
|---|----------------|---------------|
| 1 | SP | -8.2596085... |
| 2 | PR | -11.480793... |
| 3 | MG | -11.515522... |
| 4 | DF | -12.501486... |
| 5 | SC | -14.520985... |

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

Ans:

```

select customer_state, avg(time_to_delivery) as avg_actual,
avg(diff_estimated_delivery) as avg_est

```

```

from (

```

```

select customer_state,

```

```

date_diff(order_purchase_timestamp,order_delivered_customer_date,
DAY) as time_to_delivery ,

```

```

date_diff(order_estimated_delivery_date,
order_delivered_customer_date, DAY) as diff_estimated_delivery

```

```

from

```

```

`target_brazil.order_items` oi

```

```

join

```

```

`target_brazil.orders` o

```

```

ON

```

```

oi.order_id = o.order_id

```

```

join `target_brazil.customers` c

ON o.customer_id = c.customer_id

) x

WHERE time_to_delivery IS NOT NULL

```

GROUP BY c

| | |
|----|---|
| 1 | select customer_state, avg(time_to_delivery) as avg_actual, avg(diff_estimated_delivery) as avg_est |
| 2 | from (|
| 3 | select customer_state, |
| 4 | date_diff(order_purchase_timestamp, order_delivered_customer_date, DAY) as time_to_delivery, |
| 5 | date_diff(order_estimated_delivery_date, order_delivered_customer_date, DAY) as diff_estimated_delivery |
| 6 | from |
| 7 | `target_brazil.order_items` oi |
| 8 | join |
| 9 | `target_brazil.orders` o |
| 10 | ON |
| 11 | oi.order_id = o.order_id |
| 12 | join `target_brazil.customers` c |
| 13 | ON o.customer_id = c.customer_id |
| 14 |) x |
| 15 | WHERE time_to_delivery IS NOT NULL |
| 16 | GROUP BY customer_state ORDER BY 2 desc LIMIT 5 |

Press Alt+F1 for Accessibility Options.

Query results [SAVE RESULTS](#) [EXPLORE DATA](#)

| JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | EXECUTION GRAPH | PREVIEW |
|-----------------|----------------|---------------|-------------------|-----------------|---------|
| Row | customer_state | avg_actual | avg_est | | |
| 1 | SP | -8.2596085... | 10.2655943... | | |
| 2 | PR | -11.480793... | 12.5338998... | | |
| 3 | MG | -11.515522... | 12.3971510... | | |
| 4 | DF | -12.501486... | 11.2747346... | | |
| 5 | SC | -14.520985... | 10.6688628... | | |

ustomer_state ORDER BY 2 desc LIMIT 5

6. Payment type analysis:

1. Month over Month count of orders for different payment types

Ans:

```

select count(*) as cnt, extract(month from order_purchase_timestamp) as mon, extract(year
from order_purchase_timestamp) as yr,

```

payment_type

```

from `target_brazil.payments` p join `target_brazil.orders` o

```

```

ON p.order_id = o.order_id

```

```

GROUP BY 3, 2, 4

```

```

ORDER BY 3, 2

```



```

1 select count(*) as cnt, extract(month from order_purchase_timestamp) as mon, extract(year from order_purchase_timestamp) as yr,
2 payment_type
3 from `target_brazil.payments` p join `target_brazil.orders` o
4 ON p.order_id = o.order_id
5 GROUP BY 3, 2, 4
6 ORDER BY 3, 2

```

Press Alt+F1 for Accessibility Options

Query results

SAVE RESULTS

EXPLORE DATA

| JOB INFORMATION | | RESULTS | JSON | EXECUTION DETAILS | EXECUTION GRAPH | PREVIEW |
|-----------------|-----|---------|------|-------------------|-----------------|---------|
| row | cnt | mon | yr | payment_type | | |
| 1 | 3 | 9 | 2016 | credit_card | | |
| 2 | 254 | 10 | 2016 | credit_card | | |
| 3 | 23 | 10 | 2016 | voucher | | |
| 4 | 2 | 10 | 2016 | debit_card | | |
| 5 | 63 | 10 | 2016 | UPI | | |
| 6 | 1 | 12 | 2016 | credit_card | | |
| 7 | 61 | 1 | 2017 | voucher | | |
| 8 | 197 | 1 | 2017 | UPI | | |
| 9 | 583 | 1 | 2017 | credit_card | | |
| 10 | 9 | 1 | 2017 | debit_card | | |

Results per page: 50 1 - 50 of 90

2. Distribution of payment installments and count of orders

Ans: select * from

`target_brazil.payments` p

join

`target_brazil.orders` o

ON p.order_id = o.order_id

ORDER BY p.order_id

Press Alt+F1 for Accessibility Option

[↓ SAVE RESULTS](#)
[📈 EXPLORE DATA](#)

Results per page: 50 ▼ 1 – 50 of 103886 |< < > >|

1. Initial exploration of dataset like checking the characteristics of data (10 points)
2. In-depth Exploration (10 points)
3. Evolution of E-commerce orders in the Brazil region (10 points)
4. Impact on Economy (10 points)
5. Analysis on sales, freight and delivery time (10 points)
6. Payment type analysis (10 points)
7. Actionable Insights (10 points)
8. Recommendations (10 points)