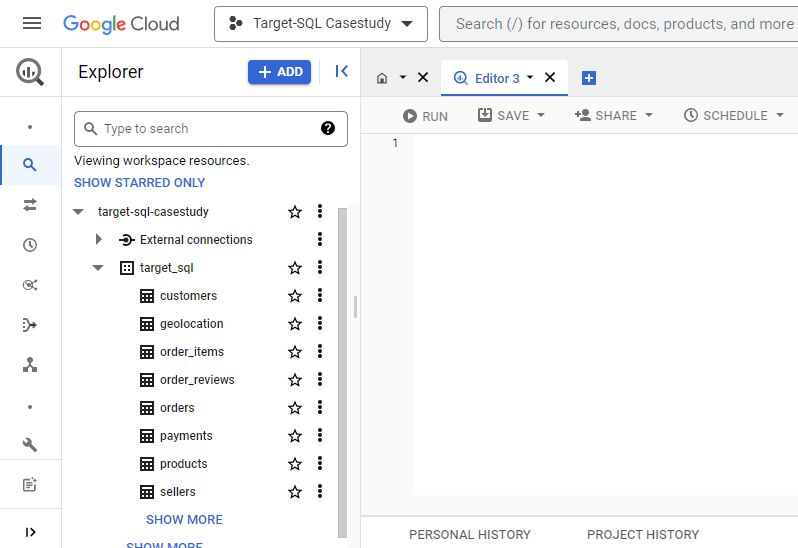
**1. Importing the data & Initial exploration of dataset like checking the characteristics of data**

* Data is imported in Bigquery:                      

**1. Data type of columns in a table**

**-**For customer Table using SQL Query

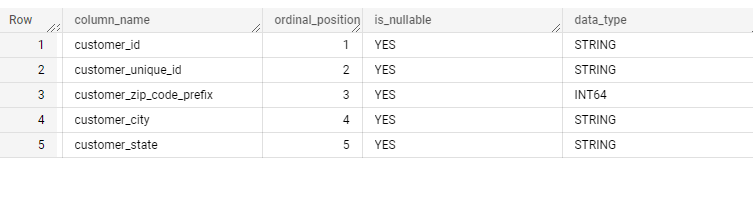
SELECT

  \*,

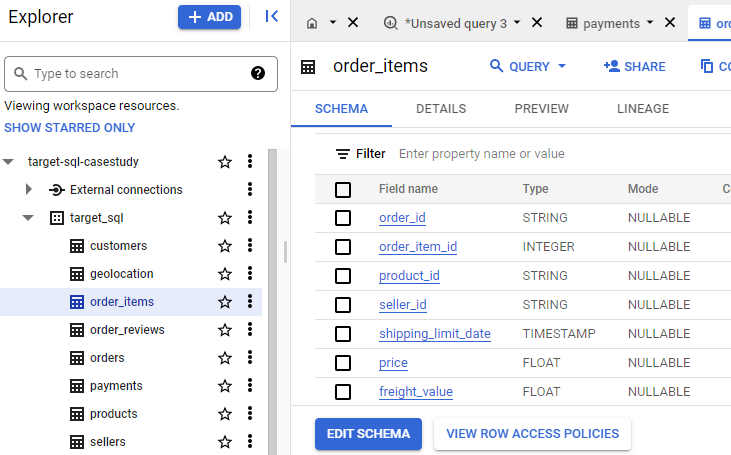
  data\_type

FROM `target-sql-casestudy.target\_sql.INFORMATION\_SCHEMA.COLUMNS`

WHERE table\_name='customers'



-For order\_items Table



**2. Time period for which the data is given**

**-**Time period for given data is considered as difference between first order purchasing date & last order purchasing date.

Time period (in Days)=772 days

First order purchased on 2016-09-04 at 21:15:19 UTC

Last order purchased on 2018-10-17 at 17:30:18 UTC

SELECT

  MIN(order\_purchase\_timestamp) AS order\_purchase\_first\_day,

  MAX(order\_purchase\_timestamp) AS order\_purchase\_last\_day,

  DATE\_DIFF(MAX(order\_purchase\_timestamp), MIN(order\_purchase\_timestamp), Day) as order\_purchase\_time\_period

FROM `target\_sql.orders`

| Row | **order\_purchase\_first\_day** | **order\_purchase\_last\_day** | **order\_purchase\_time\_period** |  |
| --- | --- | --- | --- | --- |
| 1 | 2016-09-04 21:15:19 UTC | 2018-10-17 17:30:18 UTC | 772 |  |

**3. Cities and States of customers ordered during the given period**

SELECT

  o.customer\_id,

  c.customer\_state,

  c.customer\_city,

  o.order\_purchase\_timestamp

FROM `target\_sql.orders` AS o

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

ORDER BY o.order\_purchase\_timestamp

| Row | **customer\_id** | **customer\_state** | **customer\_city** | **order\_purchase\_timestamp** |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 08c5351a6aca1c1589a38f244edeee9d | RR | boa vista | 2016-09-04 21:15:19 UTC |  |  |
| 2 | 683c54fc24d40ee9f8a6fc179fd9856c | RS | passo fundo | 2016-09-05 00:15:34 UTC |  |  |
| 3 | 622e13439d6b5a0b486c435618b2679e | SP | sao jose dos campos | 2016-09-13 15:24:19 UTC |  |  |
| 4 | 86dc2ffce2dfff336de2f386a786e574 | SP | sao joaquim da barra | 2016-09-15 12:16:38 UTC |  |  |
| 5 | b106b360fe2ef8849fbbd056f777b4d5 | SP | sao paulo | 2016-10-02 22:07:52 UTC |  |  |
| 6 | 355077684019f7f60a031656bd7262b8 | SP | sao paulo | 2016-10-03 09:44:50 UTC |  |  |
| 7 | 7ec40b22510fdbea1b08921dd39e63d8 | RS | panambi | 2016-10-03 16:56:50 UTC |  |  |
| 8 | 70fc57eeae292675927697fe03ad3ff5 | RJ | rio de janeiro | 2016-10-03 21:01:41 UTC |  |  |
| 9 | 6f989332712d3222b6571b1cf5b835ce | RS | porto alegre | 2016-10-03 21:13:36 UTC |  |  |
| 10 | b8cf418e97ae795672d326288dfab7a7 | SP | hortolandia | 2016-10-03 22:06:03 UTC |  |  |

**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?**

-Growing trend on e-commerce in Brazil is observed Year-by-Year as no of products ordered by customers increased year by year (2016 to 2018).

SELECT

  x.order\_purchase\_year,

  COUNT(\*) AS total\_items\_order\_by\_year

FROM(

SELECT

  \*,

  EXTRACT(YEAR FROM order\_purchase\_timestamp) AS order\_purchase\_year

FROM `target\_sql.orders`

) AS x

GROUP BY x.order\_purchase\_year

ORDER BY x.order\_purchase\_year

| Row | **order\_purchase\_year** | **total\_items\_order\_by\_year** |  |
| --- | --- | --- | --- |
| 1 | 2016 | 329 |  |
| 2 | 2017 | 45101 |  |
| 3 | 2018 | 54011 |  |

-More complete scenario can be obtained by further seeing the month wise trend (From 2016 to 2018). It is clearly visible that as we move from Jan'2017 to Nov'2017, no of orders purchased by customer went from 800 to 7544 and then further dropped down  to 6512 in Aug'2018, which clearly indicates that there is upward growth on e-commerce in Brazil Year-By-Year but upward growth is not strictly increasing over the whole period, and also observed few seasonal peaks .

SELECT

  x.order\_purchase\_year,

  x.order\_purchase\_month,

  COUNT(\*) AS total\_items\_order\_by\_month

FROM(

SELECT

  \*,

  EXTRACT(YEAR FROM order\_purchase\_timestamp) AS order\_purchase\_year,

  EXTRACT(MONTH FROM order\_purchase\_timestamp) AS order\_purchase\_month

FROM `target\_sql.orders`

) AS x

GROUP BY x.order\_purchase\_month,x.order\_purchase\_year

ORDER BY x.order\_purchase\_year,x.order\_purchase\_month

Load more

| Row | **order\_purchase\_year** | **order\_purchase\_month** | **total\_items\_order\_by\_month** |  |
| --- | --- | --- | --- | --- |
| 1 | 2016 | 9 | 4 |  |
| 2 | 2016 | 10 | 324 |  |
| 3 | 2016 | 12 | 1 |  |
| 4 | 2017 | 1 | 800 |  |
| 5 | 2017 | 2 | 1780 |  |
| 6 | 2017 | 3 | 2682 |  |
| 7 | 2017 | 4 | 2404 |  |
| 8 | 2017 | 5 | 3700 |  |
| 9 | 2017 | 6 | 3245 |  |
| 10 | 2017 | 7 | 4026 |  |
| 11 | 2017 | 8 | 4331 |  |
| 12 | 2017 | 9 | 4285 |  |
| 13 | 2017 | 10 | 4631 |  |
| 14 | 2017 | 11 | 7544 |  |
| 15 | 2017 | 12 | 5673 |  |

-We can observed some seasonality with peaks at specific month like Nov'2017 as no of orders by customer went from 4631 in Oct'2017 to 7544 in Nov'2017. Also No of orders went from 5673 in Dec'2017 to 7269 in Jan'2018.Thus, Peak is observed in month of Nov'2017 and Jan'2018. Although , data of Nov'2018 or Jan'2019 is not available to further analyse if peaks are observed every year during this time of month.

SELECT

  x.order\_purchase\_year,

  x.order\_purchase\_month,

  COUNT(\*) AS total\_items\_order\_by\_month,

  DENSE\_RANK() OVER(

    ORDER BY Count(\*) DESC

  ) AS Highest\_order\_rank

FROM(

SELECT

  \*,

  EXTRACT(YEAR FROM order\_purchase\_timestamp) AS order\_purchase\_year,

  EXTRACT(MONTH FROM order\_purchase\_timestamp) AS order\_purchase\_month

FROM `target\_sql.orders`

) AS x

GROUP BY x.order\_purchase\_month,x.order\_purchase\_year

ORDER BY x.order\_purchase\_year,x.order\_purchase\_month

Load more

| Row | **order\_purchase\_year** | **order\_purchase\_month** | **total\_items\_order\_by\_month** | **Highest\_order\_rank** |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 2016 | 9 | 4 | 23 |  |
| 2 | 2016 | 10 | 324 | 21 |  |
| 3 | 2016 | 12 | 1 | 24 |  |
| 4 | 2017 | 1 | 800 | 20 |  |
| 5 | 2017 | 2 | 1780 | 19 |  |
| 6 | 2017 | 3 | 2682 | 17 |  |
| 7 | 2017 | 4 | 2404 | 18 |  |
| 8 | 2017 | 5 | 3700 | 15 |  |
| 9 | 2017 | 6 | 3245 | 16 |  |
| 10 | 2017 | 7 | 4026 | 14 |  |
| 11 | 2017 | 8 | 4331 | 12 |  |
| 12 | 2017 | 9 | 4285 | 13 |  |
| 13 | 2017 | 10 | 4631 | 11 |  |
| 14 | 2017 | 11 | 7544 | 1 |  |
| 15 | 2017 | 12 | 5673 | 10 |  |
| 16 | 2018 | 1 | 7269 | 2 |  |
| 17 | 2018 | 2 | 6728 | 6 |  |

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

**-**To solve the given problem, I have firstly classify the day hours as below:

4AM-6.59AM=Dawn

7AM-11.59AM=Morning

12PM-4.59PM=Afternoon

5PM-7.59 PM=Evening

8PM-3.59AM(Next Day)=Night

It is observed that Brazilian customers tend to buy more during Afternoon i.e.32211 products ordered which is almost twice as that of products ordered during evening. Only 896 products ordered during Dawn, and least among all.

SELECT

  x.time\_classify,

  count(\*) AS no\_of\_products\_ordered

FROM

(SELECT

  \*,

  EXTRACT(HOUR FROM order\_purchase\_timestamp) AS hour\_,

  case

   when EXTRACT(HOUR FROM order\_purchase\_timestamp) between 4 and 6

      then 'Dawn'

   when EXTRACT(HOUR FROM order\_purchase\_timestamp) between 7 and 11

      then 'Morning'

   when EXTRACT(HOUR FROM order\_purchase\_timestamp) between 12 and 16

      then 'Afternoon'

   when EXTRACT(HOUR FROM order\_purchase\_timestamp) between 17 and 19

      then 'Evening'

   ELSE 'Night'

end AS time\_classify

FROM `target\_sql.orders`

) AS x

GROUP BY x.time\_classify

ORDER BY no\_of\_products\_ordered DESC

| Row | **time\_classify** | **no\_of\_products\_ordered** |  |
| --- | --- | --- | --- |
| 1 | Afternoon | 32211 |  |
| 2 | Night | 26695 |  |
| 3 | Morning | 21738 |  |
| 4 | Evening | 17901 |  |
| 5 | Dawn | 896 |  |

**3. Evaluation of E-commerce orders in the Brazil region**

**1. Get month  on month orders by states**

SELECT

  x.customer\_state,

  x.order\_year,

  x.order\_month,

  count(\*) AS orders\_

  FROM

(SELECT

  \*,

  EXTRACT(MONTH FROM order\_purchase\_timestamp) AS order\_month,

  EXTRACT(YEAR FROM order\_purchase\_timestamp) AS order\_year

FROM `target\_sql.orders` AS o

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id) AS x

GROUP BY x.order\_year,x.order\_month,x.customer\_state

ORDER BY x.customer\_state,x.order\_year,x.order\_month

| Row | **customer\_state** | **order\_year** | **order\_month** | **orders\_** |  |
| --- | --- | --- | --- | --- | --- |
| 1 | AC | 2017 | 1 | 2 |  |
| 2 | AC | 2017 | 2 | 3 |  |
| 3 | AC | 2017 | 3 | 2 |  |
| 4 | AC | 2017 | 4 | 5 |  |
| 5 | AC | 2017 | 5 | 8 |  |
| 6 | AC | 2017 | 6 | 4 |  |
| 7 | AC | 2017 | 7 | 5 |  |
| 8 | AC | 2017 | 8 | 4 |  |
| 9 | AC | 2017 | 9 | 5 |  |
| 10 | AC | 2017 | 10 | 6 |  |

**2. Distribution of customers across the states in Brazil**

**-**Top 5 States by No of customers

**SP > RJ > MG>RS>PR**

SELECT

  customer\_state,

  COUNT(\*) AS No\_of\_customers\_in\_state

FROM `target\_sql.customers`

GROUP BY customer\_state

ORDER BY No\_of\_customers\_in\_state DESC

Load more

| Row | **customer\_state** | **No\_of\_customers\_in\_state** |  |
| --- | --- | --- | --- |
| 1 | SP | 41746 |  |
| 2 | RJ | 12852 |  |
| 3 | MG | 11635 |  |
| 4 | RS | 5466 |  |
| 5 | PR | 5045 |  |
| 6 | SC | 3637 |  |
| 7 | BA | 3380 |  |
| 8 | DF | 2140 |  |
| 9 | ES | 2033 |  |
| 10 | GO | 2020 |  |

-Bottom 5 states by No. of Customers

**RR > AP > AC > AM > RO**

SELECT

  customer\_state,

  COUNT(\*) AS No\_of\_customers\_in\_state

FROM `target\_sql.customers`

GROUP BY customer\_state

ORDER BY No\_of\_customers\_in\_state

|  |
| --- |
|  |
| Row | **customer\_state** | **No\_of\_customers\_in\_state** |  |
| 1 | RR | 46 |  |
| 2 | AP | 68 |  |
| 3 | AC | 81 |  |
| 4 | AM | 148 |  |
| 5 | RO | 253 |  |
| 6 | TO | 280 |  |
| 7 | SE | 350 |  |
| 8 | AL | 413 |  |
| 9 | RN | 485 |  |
| 10 | PI | 495 |  |

**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)**

Total % increase in cost of orders from 2017 to 2018 (including months between Jan to Aug only)= 136.98 %

SELECT

  x.order\_year,

  ROUND(SUM(x.payment\_value),2)AS sum\_cost\_of\_orders\_,

  ROUND((SUM(x.payment\_value) - LAG (SUM(x.payment\_value)) OVER (ORDER BY x.order\_year ))/LAG (SUM(x.payment\_value)) OVER (ORDER BY x.order\_year)\*100,2) AS cost\_of\_order\_percentage\_growth

FROM(

SELECT

  \*,

  EXTRACT(YEAR FROM o.order\_purchase\_timestamp) AS order\_year,

  EXTRACT(MONTH FROM o.order\_purchase\_timestamp) AS order\_month

FROM `target\_sql.payments` AS p

INNER JOIN `target\_sql.orders` AS o

ON p.order\_id=o.order\_id) AS x

WHERE x.order\_year IN (2017,2018)

AND x.order\_month IN (1,2,3,4,5,6,7,8)

GROUP BY x.order\_year

ORDER BY x.order\_year

| Row | **order\_year** | **sum\_cost\_of\_orders\_** | **cost\_of\_order\_percentage\_growth** |  |
| --- | --- | --- | --- | --- |
| 1 | 2017 | 3669022.12 | *null* |  |
| 2 | 2018 | 8694733.84 | 136.98 |  |

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

SELECT

  c.customer\_state,

  ROUND(SUM(oi.price),2) AS sum\_price,

  ROUND(SUM(oi.freight\_value),2) AS sum\_freight,

  ROUND(AVG(oi.price),2) AS average\_price,

  ROUND(AVG(oi.freight\_value),2) AS average\_freight

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

GROUP BY c.customer\_state

ORDER BY c.customer\_state

| Row | **customer\_state** | **sum\_price** | **sum\_freight** | **average\_price** | **average\_freight** |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | AC | 15982.95 | 3686.75 | 173.73 | 40.07 |  |
| 2 | AL | 80314.81 | 15914.59 | 180.89 | 35.84 |  |
| 3 | AM | 22356.84 | 5478.89 | 135.5 | 33.21 |  |
| 4 | AP | 13474.3 | 2788.5 | 164.32 | 34.01 |  |
| 5 | BA | 511349.99 | 100156.68 | 134.6 | 26.36 |  |
| 6 | CE | 227254.71 | 48351.59 | 153.76 | 32.71 |  |
| 7 | DF | 302603.94 | 50625.5 | 125.77 | 21.04 |  |
| 8 | ES | 275037.31 | 49764.6 | 121.91 | 22.06 |  |
| 9 | GO | 294591.95 | 53114.98 | 126.27 | 22.77 |  |
| 10 | MA | 119648.22 | 31523.77 | 145.2 | 38.26 |  |

-Top 5 states in terms of average price of orders

**PB>AL>AC>RO>PA**

SELECT

  c.customer\_state,

  ROUND(AVG(oi.price),2) AS average\_price,

  ROUND(AVG(oi.freight\_value),2) AS average\_freight

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

GROUP BY c.customer\_state

ORDER BY average\_price DESC

LIMIT 5

| Row | **customer\_state** | **average\_price** | **average\_freight** |  |
| --- | --- | --- | --- | --- |
| 1 | PB | 191.48 | 42.72 |  |
| 2 | AL | 180.89 | 35.84 |  |
| 3 | AC | 173.73 | 40.07 |  |
| 4 | RO | 165.97 | 41.07 |  |
| 5 | PA | 165.69 | 35.83 |  |

-Bottom 5 states in terms of average price of orders

**SP>PR>RS>MG>ES**

SELECT

  c.customer\_state,

  ROUND(AVG(oi.price),2) AS average\_price,

  ROUND(AVG(oi.freight\_value),2) AS average\_freight

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

GROUP BY c.customer\_state

ORDER BY average\_price

LIMIT 5

| Row | **customer\_state** | **average\_price** | **average\_freight** |  |
| --- | --- | --- | --- | --- |
| 1 | SP | 109.65 | 15.15 |  |
| 2 | PR | 119.0 | 20.53 |  |
| 3 | RS | 120.34 | 21.74 |  |
| 4 | MG | 120.75 | 20.63 |  |
| 5 | ES | 121.91 | 22.06 |  |

**5. Analysis on sales, freight and delivery time**

**1. Calculate days between purchasing, delivering and estimated delivery**

**- Days b/w order purchase & order delivery**

Average days taken for order to deliver once purchase is made= 12.09 Days

SELECT

  order\_id,

  customer\_id,

  order\_status,

  order\_purchase\_timestamp,

  order\_delivered\_customer\_date,

  DATE\_DIFF(order\_delivered\_customer\_date,order\_purchase\_timestamp,Day) AS days\_order\_delivered,

  ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,order\_purchase\_timestamp,Day)) OVER(),2) AS avg\_days\_order\_delivered

FROM `target\_sql.orders`

WHERE order\_purchase\_timestamp IS NOT NULL

AND order\_delivered\_customer\_date IS NOT NULL

| Row | **order\_id** | **customer\_id** | **order\_status** | **order\_purchase\_timestamp** | **order\_delivered\_customer\_date** | **days\_order\_delivered** | **avg\_days\_order\_delivered** |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | d7918e406132d7c81f1b845276b03a3b | 53504e2e5940107ff1e2e52a02cf6f8f | delivered | 2017-07-13 17:54:53 UTC | 2017-08-17 18:35:38 UTC | 35 | 12.09 |  |
| 2 | dc32acbe8d41196887e5d7ac6c2972bd | 1fcb9f10321dfd890dd7153a7dfcc890 | delivered | 2018-03-07 15:24:48 UTC | 2018-04-12 20:52:41 UTC | 36 | 12.09 |  |
| 3 | 4802922b16a37af8db6fb3df54b07860 | f078095ff04dedf7d217df058773668b | delivered | 2017-12-01 11:25:38 UTC | 2018-01-08 18:54:58 UTC | 38 | 12.09 |  |
| 4 | ed78b29fa4c1255d275df5d4e464503b | a5dceb0a120a9b028d3d4582422ad480 | delivered | 2018-01-17 09:33:47 UTC | 2018-02-20 18:36:45 UTC | 34 | 12.09 |  |
| 5 | 3054c6a8e45b740723ec8a7145d867b7 | 78d543ef68ecb2e26895755f8dc9c86a | delivered | 2018-03-28 13:54:39 UTC | 2018-05-02 15:38:26 UTC | 35 | 12.09 |  |
| 6 | 5d6e9993ecc20a59e637ce711858d081 | 47fd93db3f6e715f6d06f90ba501b94d | delivered | 2017-01-19 11:26:57 UTC | 2017-03-23 08:21:38 UTC | 62 | 12.09 |  |
| 7 | 86c25bbc1d16736365d6e747e71695f0 | ae79842e6e74fffdd282a258b22a757b | delivered | 2018-01-13 13:50:29 UTC | 2018-02-21 20:29:00 UTC | 39 | 12.09 |  |
| 8 | 42873babe11a5362e2b59e1bf80fbd92 | 3ec4d59cd0d003ac1da03b3c5fe970f2 | delivered | 2017-11-19 08:16:33 UTC | 2017-12-28 14:29:14 UTC | 39 | 12.09 |  |
| 9 | 482b0898624ebab18e90d224126c98a8 | 6e422478a5ccc422a4a7e50df1bff41e | delivered | 2017-11-21 10:46:57 UTC | 2017-12-21 14:22:58 UTC | 30 | 12.09 |  |
| 10 | 3522f6da18c89c3dfc0ddb540afac391 | e8af75d51a3972af7ed977ad4855aad6 | delivered | 2017-06-28 11:55:21 UTC | 2017-07-27 21:51:47 UTC | 29 | 12.09 |  |

**-Days b/w delivery date and estimated delivery date**

Average days of difference b/w Order delivery date & Order estimated delivery date= 10.96 Days. It includes both cases i.e. when order is delivered before estimated delivery date & when order is delivered after estimated delivery date

SELECT

  order\_id,

  customer\_id,

  order\_status,

  order\_delivered\_customer\_date,

  order\_estimated\_delivery\_date,

  DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,Day) AS days\_diff\_estimated\_delivery,

  ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,Day)) OVER(),2) AS days\_diff\_estimated\_delivery

FROM `target\_sql.orders`

WHERE order\_delivered\_customer\_date IS NOT NULL

AND order\_estimated\_delivery\_date IS NOT NULL

| Row | **order\_id** | **customer\_id** | **order\_status** | **order\_delivered\_customer\_date** | **order\_estimated\_delivery\_date** | **days\_diff\_estimated\_delivery** | **days\_diff\_estimated\_delivery\_1** |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | a9d9e48bc0905664f33c53779b5edfb6 | 7195875a1f2cdaa5e75a1f132530e156 | delivered | 2017-11-15 00:42:59 UTC | 2017-11-10 00:00:00 UTC | -5 | 10.96 |  |
| 2 | 9e7b4dc19aadc8298e2984986d3bf014 | 2ab4c04d8d6160749ea57e125c32a6a1 | delivered | 2017-12-20 14:32:58 UTC | 2017-11-10 00:00:00 UTC | -40 | 10.96 |  |
| 3 | d0240bcff59b720e55a2fc2c74344b6f | 09f758f6da558eae5292b1a7718518b2 | delivered | 2017-02-10 10:34:45 UTC | 2017-03-15 00:00:00 UTC | 32 | 10.96 |  |
| 4 | 15b1aceab346b46189a6eea3b6f4b951 | 8ec1617ce630c731866f5a041b5d408c | delivered | 2017-02-09 09:27:56 UTC | 2017-03-15 00:00:00 UTC | 33 | 10.96 |  |
| 5 | 643e340c6c562d108526c36d5a94c027 | d268eb5ab77b7f0837b538ea3856884a | delivered | 2018-01-18 22:51:54 UTC | 2018-01-05 00:00:00 UTC | -13 | 10.96 |  |
| 6 | 3f26a399c8a1c0cf2c9dc8a194b134d3 | e03322e99d1228607781f6db2ead35fb | delivered | 2018-06-06 18:51:44 UTC | 2018-05-29 00:00:00 UTC | -8 | 10.96 |  |
| 7 | 03ddab02775c66c855ad41424ae356e3 | ed87333e69d2709d374534505818c9e2 | delivered | 2018-02-19 18:11:40 UTC | 2018-03-26 00:00:00 UTC | 34 | 10.96 |  |
| 8 | 2c60bc71e6da917194c35fa7c8477e7f | c14ee009789095771e86aece808515aa | delivered | 2018-04-16 21:26:58 UTC | 2018-03-26 00:00:00 UTC | -21 | 10.96 |  |
| 9 | 7fb6586ab530ed69383ddde20ef388f9 | b0d72b227b0caa8a7156db18f5cbe0f4 | delivered | 2018-04-16 20:22:03 UTC | 2018-03-26 00:00:00 UTC | -21 | 10.96 |  |
| 10 | 5ed3ec02627cdee4cc803950b95aaf02 | 62ccb3e4d92ff9d183f032bbcc727821 | delivered | 2018-05-02 19:52:07 UTC | 2018-04-19 00:00:00 UTC | -13 | 10.96 |  |

**-where order delivered before the estimated delivery date**

Average days of difference b/w Order delivery date & Order estimated delivery date=12.71 Days when order is delivered before estimated delivery date .

SELECT

  order\_id,

  customer\_id,

  order\_status,

  order\_delivered\_customer\_date,

  order\_estimated\_delivery\_date,

  DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,Day) AS days\_diff\_estimated\_delivery,

  ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,Day)) OVER(),2) AS avg\_days\_diff\_estimated\_delivery

FROM `target\_sql.orders`

WHERE order\_delivered\_customer\_date IS NOT NULL

AND order\_estimated\_delivery\_date IS NOT NULL

AND order\_estimated\_delivery\_date>order\_delivered\_customer\_date

| Row | **order\_id** | **customer\_id** | **order\_status** | **order\_delivered\_customer\_date** | **order\_estimated\_delivery\_date** | **days\_diff\_estimated\_delivery** | **avg\_days\_diff\_estimated\_delivery** |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 445736171736c70c6187ee83c34165fe | b0cd60f17cb2a0ffd09d607e65d3874a | delivered | 2018-08-21 14:24:37 UTC | 2018-09-26 00:00:00 UTC | 35 | 12.71 |  |
| 2 | 1a2582ca80feaaf937cc94d437ba1635 | 99efadd052716ca57a64a6287caca0eb | delivered | 2018-06-09 15:42:09 UTC | 2018-07-16 00:00:00 UTC | 36 | 12.71 |  |
| 3 | ff4cdf9a24ffba79c88c81aeb858b4c7 | 671c88e6f743e984ae2f877fb66e1a1f | delivered | 2017-08-16 21:23:56 UTC | 2017-09-15 00:00:00 UTC | 29 | 12.71 |  |
| 4 | 2c0063b6f94aadf5af90291765541726 | 0541a0155b375e5d1ca9ed98e674e3b2 | delivered | 2018-05-21 23:52:14 UTC | 2018-06-25 00:00:00 UTC | 34 | 12.71 |  |
| 5 | 2c3f481e97111f73a244fdfc556e2303 | ff0feea3121816d8a9918cd346d2ab7d | delivered | 2017-02-24 10:38:56 UTC | 2017-04-03 00:00:00 UTC | 37 | 12.71 |  |
| 6 | 10492d21e254a11aa46c125a01702bfa | 8f3fc20a1f8167dde23e0da111626aa0 | delivered | 2018-04-02 16:49:39 UTC | 2018-05-08 00:00:00 UTC | 35 | 12.71 |  |
| 7 | 523dafc7e66222e30a400621133aca08 | 6ec64888059bae0f28043de36f7f4516 | delivered | 2018-06-18 19:12:12 UTC | 2018-08-07 00:00:00 UTC | 49 | 12.71 |  |
| 8 | 0516c22655fe1e1f0b3390f4ee3fc5f6 | 4830a9c02b42036ffca989c6d2f8ff67 | delivered | 2017-02-09 18:38:01 UTC | 2017-03-13 00:00:00 UTC | 31 | 12.71 |  |
| 9 | 86df4a2f1f5cd47ee3f5bf1781240d1a | b0db02848249cbe57e7e5732976ced8e | delivered | 2016-10-26 10:38:43 UTC | 2016-11-29 00:00:00 UTC | 33 | 12.71 |  |
| 10 | b4e9e179195a7cee910bf458c5b6e01e | acad76e6005b7efd172a9fed40f7d31f | delivered | 2017-02-08 13:37:45 UTC | 2017-03-16 00:00:00 UTC | 35 | 12.71 |  |

**-No of products delivered before estimated delivery date**

SELECT

  count(\*) AS no\_Products\_delivered\_before\_estimated

FROM `target\_sql.orders`

WHERE order\_delivered\_customer\_date IS NOT NULL

AND order\_estimated\_delivery\_date IS NOT NULL

AND order\_estimated\_delivery\_date>order\_delivered\_customer\_date

| Row | **no\_Products\_delivered\_before\_estimated** |  |
| --- | --- | --- |
| 1 | 88649 |  |

**-orders delivered after estimated delivery date**

Average days of difference b/w Order delivery date & Order estimated delivery date=8.87 Days when order is delivered after estimated delivery date .

SELECT

  order\_id,

  customer\_id,

  order\_status,

  order\_delivered\_customer\_date,

  order\_estimated\_delivery\_date,

  DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,Day) AS days\_diff\_estimated\_delivery,

  ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,Day)) OVER(),2) AS avg\_days\_diff\_estimated\_delivery

FROM `target\_sql.orders`

WHERE order\_delivered\_customer\_date IS NOT NULL

AND order\_estimated\_delivery\_date IS NOT NULL

AND order\_estimated\_delivery\_date<order\_delivered\_customer\_date

| Row | **order\_id** | **customer\_id** | **order\_status** | **order\_delivered\_customer\_date** | **order\_estimated\_delivery\_date** | **days\_diff\_estimated\_delivery** | **avg\_days\_diff\_estimated\_delivery** |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0f4519c5f1c541ddec9f21b3bddd533a | 1a8a4a30dc296976717f44e7801fdeef | delivered | 2017-09-19 14:38:21 UTC | 2017-04-11 00:00:00 UTC | -161 | -8.87 |  |
| 2 | 866314550f6d7a55c82917d9b4463e1f | 9009e9f0ac28d47bd18433d82aa028d6 | delivered | 2018-01-25 19:38:35 UTC | 2017-12-13 00:00:00 UTC | -43 | -8.87 |  |
| 3 | 465cfd7016202e455adf9b1a83736c86 | 3ae208c673c9a1bd78c2dc73fdcd1121 | delivered | 2018-02-06 15:58:54 UTC | 2017-12-27 00:00:00 UTC | -41 | -8.87 |  |
| 4 | 621f567448cbf2281b22381247c491ca | 7a4e8b61d5570efb155ab9bf4a0659c3 | delivered | 2018-04-18 21:16:43 UTC | 2018-03-13 00:00:00 UTC | -36 | -8.87 |  |
| 5 | 2e56f943f231f5fe108f43fb370b0ed6 | 66b9bc2d53ea09b027966337424fa0c7 | delivered | 2017-05-25 07:14:44 UTC | 2017-03-16 00:00:00 UTC | -70 | -8.87 |  |
| 6 | 69c7a1e070c1759e15f7a5fd24c36e5b | 6cbc69ef01df8d34bd134c6d828c1e53 | delivered | 2017-05-29 09:42:44 UTC | 2017-04-04 00:00:00 UTC | -55 | -8.87 |  |
| 7 | 32fef9bb0f0402350b640eb6f13dbe09 | 28c857caf2f37bb8751725b85dac2481 | delivered | 2018-05-03 00:39:55 UTC | 2018-03-29 00:00:00 UTC | -35 | -8.87 |  |
| 8 | b73bbac6285251dbf2dd01842e0ce0de | b97669f12cafebbe9441bf4af0a8fe2a | delivered | 2018-02-26 21:07:44 UTC | 2018-01-24 00:00:00 UTC | -33 | -8.87 |  |
| 9 | 6cbad422849c13ec8f4517ee01a6c02b | 23a74d590627edc3207d90d96ec739a6 | delivered | 2017-05-10 13:40:06 UTC | 2017-04-06 00:00:00 UTC | -34 | -8.87 |  |
| 10 | 880a2443009447fa88b518da4806fc93 | e4f06fec45100ba91af9bc48190aacf0 | delivered | 2018-03-11 21:06:06 UTC | 2018-01-30 00:00:00 UTC | -40 | -8.87 |  |

**-No of orders delivered after estimated delivery date**

SELECT

  count(\*) AS no\_Products\_delivered\_after\_estimated

FROM `target\_sql.orders`

WHERE order\_delivered\_customer\_date IS NOT NULL

AND order\_estimated\_delivery\_date IS NOT NULL

AND order\_estimated\_delivery\_date<order\_delivered\_customer\_date

| Row | **no\_Products\_delivered\_after\_estimated** |  |
| --- | --- | --- |
| 1 | 7827 |  |

**-No of orders delivered after estimated delivery date (By state)**

With order\_sum AS(

SELECT

  order\_id,

  customer\_id,

  order\_status,

  order\_delivered\_customer\_date,

  order\_estimated\_delivery\_date,

  DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,Day) AS days\_diff\_estimated\_delivery,

  ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,Day)) OVER(),2) AS avg\_days\_diff\_estimated\_delivery

FROM `target\_sql.orders`

WHERE order\_delivered\_customer\_date IS NOT NULL

AND order\_estimated\_delivery\_date IS NOT NULL

AND order\_estimated\_delivery\_date<order\_delivered\_customer\_date

)

SELECT

  c.customer\_state,

  ROUND(AVG(os.days\_diff\_estimated\_delivery),2) AS avg\_days\_state,

  COUNT(os.order\_id) AS no\_of\_orders\_state

FROM order\_sum AS os

INNER JOIN `target\_sql.customers` AS c

ON os.customer\_id=c.customer\_id

GROUP BY c.customer\_state

ORDER BY no\_of\_orders\_state DESC

| Row | **customer\_state** | **avg\_days\_state** | **no\_of\_orders\_state** |  |
| --- | --- | --- | --- | --- |
| 1 | SP | -6.35 | 2387 |  |
| 2 | RJ | -12.15 | 1664 |  |
| 3 | MG | -6.85 | 638 |  |
| 4 | BA | -10.42 | 457 |  |
| 5 | RS | -8.71 | 382 |  |
| 6 | SC | -6.99 | 346 |  |
| 7 | PR | -6.73 | 246 |  |
| 8 | ES | -9.91 | 244 |  |
| 9 | CE | -13.63 | 196 |  |
| 10 | PE | -10.65 | 172 |  |
| 11 | GO | -9.09 | 160 |  |
| 12 | DF | -5.95 | 147 |  |
| 13 | MA | -9.31 | 141 |  |
| 14 | PA | -11.62 | 117 |  |
| 15 | AL | -8.54 | 95 |  |

**2. Find time\_to\_delivery & diff\_estimated\_delivery**

**time\_to\_delivery = order\_purchase\_timestamp-order\_delivered\_customer\_date**

**diff\_estimated\_delivery = order\_estimated\_delivery\_date-order\_delivered\_customer\_date**

**-Time\_to\_delivery**

WITH difference\_in\_seconds AS (

  SELECT

    order\_id,

    customer\_id,

    order\_purchase\_timestamp,

    order\_delivered\_customer\_date,

    TIMESTAMP\_DIFF(order\_delivered\_customer\_date, order\_purchase\_timestamp, SECOND) AS seconds

  FROM `target\_sql.orders`

  WHERE order\_delivered\_customer\_date IS NOT NULL AND order\_purchase\_timestamp IS NOT NULL

),

differences AS (

  SELECT

    order\_id,

    customer\_id,

    order\_purchase\_timestamp,

    order\_delivered\_customer\_date,

    seconds,

    MOD(seconds, 60) AS seconds\_part,

    MOD(seconds, 3600) AS minutes\_part,

    MOD(seconds, 3600 \* 24) AS hours\_part

  FROM difference\_in\_seconds

)

SELECT

  order\_id,

  customer\_id,

  order\_purchase\_timestamp,

  order\_delivered\_customer\_date,

  CONCAT(

    FLOOR(seconds / 3600 / 24), ' days ',

    FLOOR(hours\_part / 3600), ' hours ',

    FLOOR(minutes\_part / 60), ' minutes ',

    seconds\_part, ' seconds'

  ) AS time\_to\_delivery

FROM differences

| Row | **order\_id** | **customer\_id** | **order\_purchase\_timestamp** | **order\_delivered\_customer\_date** | **time\_to\_delivery** |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 770d331c84e5b214bd9dc70a10b829d0 | 6c57e6119369185e575b36712766b0ef | 2016-10-07 14:52:30 UTC | 2016-10-14 15:07:11 UTC | 7 days 0 hours 14 minutes 41 seconds |  |
| 2 | 1950d777989f6a877539f53795b4c3c3 | 1bccb206de9f0f25adc6871a1bcf77b2 | 2018-02-19 19:48:52 UTC | 2018-03-21 22:03:51 UTC | 30 days 2 hours 14 minutes 59 seconds |  |
| 3 | 2c45c33d2f9cb8ff8b1c86cc28c11c30 | de4caa97afa80c8eeac2ff4c8da5b72e | 2016-10-09 15:39:56 UTC | 2016-11-09 14:53:50 UTC | 30 days 23 hours 13 minutes 54 seconds |  |
| 4 | dabf2b0e35b423f94618bf965fcb7514 | 5cdec0bb8cbdf53ffc8fdc212cd247c6 | 2016-10-09 00:56:52 UTC | 2016-10-16 14:36:59 UTC | 7 days 13 hours 40 minutes 7 seconds |  |
| 5 | 8beb59392e21af5eb9547ae1a9938d06 | bf609b5741f71697f65ce3852c5d2623 | 2016-10-08 20:17:50 UTC | 2016-10-19 18:47:43 UTC | 10 days 22 hours 29 minutes 53 seconds |  |
| 6 | 65d1e226dfaeb8cdc42f665422522d14 | 70fc57eeae292675927697fe03ad3ff5 | 2016-10-03 21:01:41 UTC | 2016-11-08 10:58:34 UTC | 35 days 13 hours 56 minutes 53 seconds |  |
| 7 | c158e9806f85a33877bdfd4f607b72e7 | 25456ee3b0cf84658015e46686025224 | 2017-04-14 22:06:32 UTC | 2017-05-08 11:10:26 UTC | 23 days 13 hours 3 minutes 54 seconds |  |
| 8 | b60b53ad0bb7dacacf2989fe27ad567a | 2f9902d85fcd930227f711cf47012bdf | 2017-05-10 14:03:27 UTC | 2017-05-23 13:12:27 UTC | 12 days 23 hours 9 minutes 0 seconds |  |
| 9 | c830f223aae08493ebecb52f29aa48ca | af626bcc9c27c08077b02e6d3aa00c36 | 2017-04-22 15:50:30 UTC | 2017-05-05 13:27:50 UTC | 12 days 21 hours 37 minutes 20 seconds |  |
| 10 | a8aa2cd070eeac7e4368cae3d8222e2b | 2c5519c36277c3f69df911c68cc97e50 | 2017-05-09 17:42:45 UTC | 2017-05-16 23:22:20 UTC | 7 days 5 hours 39 minutes 35 sec |  |

**-diff\_estimated\_delivery**

WITH difference\_in\_seconds AS (

  SELECT

    order\_id,

    customer\_id,

    order\_delivered\_customer\_date,

    order\_estimated\_delivery\_date,

    TIMESTAMP\_DIFF(order\_estimated\_delivery\_date, order\_delivered\_customer\_date, SECOND) AS seconds

  FROM `target\_sql.orders`

  WHERE order\_delivered\_customer\_date IS NOT NULL AND order\_estimated\_delivery\_date IS NOT NULL

),

differences AS (

  SELECT

    order\_id,

    customer\_id,

    order\_delivered\_customer\_date,

    order\_estimated\_delivery\_date,

    seconds,

    MOD(seconds, 60) AS seconds\_part,

    MOD(seconds, 3600) AS minutes\_part,

    MOD(seconds, 3600 \* 24) AS hours\_part

  FROM difference\_in\_seconds

)

SELECT

  order\_id,

    customer\_id,

    order\_delivered\_customer\_date,

    order\_estimated\_delivery\_date,

  CONCAT(

    FLOOR(seconds / 3600 / 24), ' days ',

    FLOOR(hours\_part / 3600), ' hours ',

    FLOOR(minutes\_part / 60), ' minutes ',

    seconds\_part, ' seconds'

  ) AS diff\_estimated\_delivery

FROM differences

| Row | **order\_id** | **customer\_id** | **order\_delivered\_customer\_date** | **order\_estimated\_delivery\_date** | **diff\_estimated\_delivery** |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 770d331c84e5b214bd9dc70a10b829d0 | 6c57e6119369185e575b36712766b0ef | 2016-10-14 15:07:11 UTC | 2016-11-29 00:00:00 UTC | 45 days 8 hours 52 minutes 49 seconds |  |
| 2 | 1950d777989f6a877539f53795b4c3c3 | 1bccb206de9f0f25adc6871a1bcf77b2 | 2018-03-21 22:03:51 UTC | 2018-03-09 00:00:00 UTC | -13 days -23 hours -4 minutes -51 seconds |  |
| 3 | 2c45c33d2f9cb8ff8b1c86cc28c11c30 | de4caa97afa80c8eeac2ff4c8da5b72e | 2016-11-09 14:53:50 UTC | 2016-12-08 00:00:00 UTC | 28 days 9 hours 6 minutes 10 seconds |  |
| 4 | dabf2b0e35b423f94618bf965fcb7514 | 5cdec0bb8cbdf53ffc8fdc212cd247c6 | 2016-10-16 14:36:59 UTC | 2016-11-30 00:00:00 UTC | 44 days 9 hours 23 minutes 1 seconds |  |
| 5 | 8beb59392e21af5eb9547ae1a9938d06 | bf609b5741f71697f65ce3852c5d2623 | 2016-10-19 18:47:43 UTC | 2016-11-30 00:00:00 UTC | 41 days 5 hours 12 minutes 17 seconds |  |
| 6 | 65d1e226dfaeb8cdc42f665422522d14 | 70fc57eeae292675927697fe03ad3ff5 | 2016-11-08 10:58:34 UTC | 2016-11-25 00:00:00 UTC | 16 days 13 hours 1 minutes 26 seconds |  |
| 7 | c158e9806f85a33877bdfd4f607b72e7 | 25456ee3b0cf84658015e46686025224 | 2017-05-08 11:10:26 UTC | 2017-05-18 00:00:00 UTC | 9 days 12 hours 49 minutes 34 seconds |  |
| 8 | b60b53ad0bb7dacacf2989fe27ad567a | 2f9902d85fcd930227f711cf47012bdf | 2017-05-23 13:12:27 UTC | 2017-05-18 00:00:00 UTC | -6 days -14 hours -13 minutes -27 seconds |  |
| 9 | c830f223aae08493ebecb52f29aa48ca | af626bcc9c27c08077b02e6d3aa00c36 | 2017-05-05 13:27:50 UTC | 2017-05-18 00:00:00 UTC | 12 days 10 hours 32 minutes 10 seconds |  |
| 10 | a8aa2cd070eeac7e4368cae3d8222e2b | 2c5519c36277c3f69df911c68cc97e50 | 2017-05-16 23:22:20 UTC | 2017-05-18 00:00:00 UTC | 1 days 0 hours 37 minutes 40 s |  |

**3. Group data by state, take mean of freight\_value, time\_to\_delivery, diff\_estimated\_delivery**

SELECT

  DISTINCT

  c.customer\_state,

  ROUND(AVG(oi.freight\_value) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_freight\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_delivered\_customer\_date,o.order\_purchase\_timestamp,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_order\_delivery\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_estimated\_delivery\_date,o.order\_delivered\_customer\_date,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_order\_estimated\_state

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

WHERE o.order\_purchase\_timestamp IS NOT NULL

AND o.order\_delivered\_customer\_date IS NOT NULL

AND o.order\_estimated\_delivery\_date IS NOT NULL

ORDER BY c.customer\_state

| Row | **customer\_state** | **average\_freight\_state** | **average\_order\_delivery\_state** | **average\_order\_estimated\_state** |  |
| --- | --- | --- | --- | --- | --- |
| 1 | AC | 40.05 | 20.33 | 20.01 |  |
| 2 | AL | 35.87 | 23.99 | 7.98 |  |
| 3 | AM | 33.31 | 25.96 | 18.98 |  |
| 4 | AP | 34.16 | 27.75 | 17.44 |  |
| 5 | BA | 26.49 | 18.77 | 10.12 |  |
| 6 | CE | 32.73 | 20.54 | 10.26 |  |
| 7 | DF | 21.07 | 12.5 | 11.27 |  |
| 8 | ES | 22.03 | 15.19 | 9.77 |  |
| 9 | GO | 22.56 | 14.95 | 11.37 |  |
| 10 | MA | 38.49 | 21.2 | 9.11 |  |
| 11 | MG | 20.63 | 11.52 | 12.4 |  |

**-Top 5 states with highest average freight value**

**PB>RR>RO>AC>PI**

SELECT

  DISTINCT

  c.customer\_state,

  ROUND(AVG(oi.freight\_value) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_freight\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_delivered\_customer\_date,o.order\_purchase\_timestamp,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_Day\_order\_delivery\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_estimated\_delivery\_date,o.order\_delivered\_customer\_date,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_Day\_order\_estimated\_state

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

WHERE o.order\_purchase\_timestamp IS NOT NULL

AND o.order\_delivered\_customer\_date IS NOT NULL

AND o.order\_estimated\_delivery\_date IS NOT NULL

ORDER BY average\_freight\_state DESC

LIMIT 5

|  |
| --- |
|  |
| Row | **customer\_state** | **average\_freight\_state** | **average\_Day\_order\_delivery\_state** | **average\_Day\_order\_estimated\_state** |  |
| 1 | PB | 43.09 | 20.12 | 12.15 |  |
| 2 | RR | 43.09 | 27.83 | 17.43 |  |
| 3 | RO | 41.33 | 19.28 | 19.08 |  |
| 4 | AC | 40.05 | 20.33 | 20.01 |  |
| 5 | PI | 39.12 | 18.93 | 10.68 |  |

**-Top 5 states with lowest average freight value**

**SP>PR>MG>RJ>DF**

SELECT

  DISTINCT

  c.customer\_state,

  ROUND(AVG(oi.freight\_value) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_freight\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_delivered\_customer\_date,o.order\_purchase\_timestamp,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_delivery\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_estimated\_delivery\_date,o.order\_delivered\_customer\_date,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_estimated\_state

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

WHERE o.order\_purchase\_timestamp IS NOT NULL

AND o.order\_delivered\_customer\_date IS NOT NULL

AND o.order\_estimated\_delivery\_date IS NOT NULL

ORDER BY average\_freight\_state

LIMIT 5

|  |
| --- |
|  |
| Row | **customer\_state** | **average\_freight\_state** | **average\_day\_order\_delivery\_state** | **average\_day\_order\_estimated\_state** |  |
| 1 | SP | 15.11 | 8.26 | 10.27 |  |
| 2 | PR | 20.47 | 11.48 | 12.53 |  |
| 3 | MG | 20.63 | 11.52 | 12.4 |  |
| 4 | RJ | 20.91 | 14.69 | 11.14 |  |
| 5 | DF | 21.07 | 12.5 | 11.27 |  |

**-Top 5 states with highest average time to delivery**

**RR>AP>AM>AL>PA**

SELECT

  DISTINCT

  c.customer\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_delivered\_customer\_date,o.order\_purchase\_timestamp,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_delivery\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_estimated\_delivery\_date,o.order\_delivered\_customer\_date,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_estimated\_state,

  ROUND(AVG(oi.freight\_value) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_freight\_state

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

WHERE o.order\_purchase\_timestamp IS NOT NULL

AND o.order\_delivered\_customer\_date IS NOT NULL

AND o.order\_estimated\_delivery\_date IS NOT NULL

ORDER BY average\_day\_order\_delivery\_state DESC

LIMIT 5

| Row | **customer\_state** | **average\_day\_order\_delivery\_state** | **average\_day\_order\_estimated\_state** | **average\_freight\_state** |  |
| --- | --- | --- | --- | --- | --- |
| 1 | RR | 27.83 | 17.43 | 43.09 |  |
| 2 | AP | 27.75 | 17.44 | 34.16 |  |
| 3 | AM | 25.96 | 18.98 | 33.31 |  |
| 4 | AL | 23.99 | 7.98 | 35.87 |  |
| 5 | PA | 23.3 | 13.37 | 35.63 |  |

**-Top 5 states with lowest average time to delivery**

**SP>PR>MG>DF>SC**

SELECT

  DISTINCT

  c.customer\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_delivered\_customer\_date,o.order\_purchase\_timestamp,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_delivery\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_estimated\_delivery\_date,o.order\_delivered\_customer\_date,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_estimated\_state,

  ROUND(AVG(oi.freight\_value) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_freight\_state

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

WHERE o.order\_purchase\_timestamp IS NOT NULL

AND o.order\_delivered\_customer\_date IS NOT NULL

AND o.order\_estimated\_delivery\_date IS NOT NULL

ORDER BY average\_day\_order\_delivery\_state

LIMIT 5

|  |
| --- |
|  |
| Row | **customer\_state** | **average\_day\_order\_delivery\_state** | **average\_day\_order\_estimated\_state** | **average\_freight\_state** |  |
| 1 | SP | 8.26 | 10.27 | 15.11 |  |
| 2 | PR | 11.48 | 12.53 | 20.47 |  |
| 3 | MG | 11.52 | 12.4 | 20.63 |  |
| 4 | DF | 12.5 | 11.27 | 21.07 |  |
| 5 | SC | 14.52 | 10.67 | 21.51 |  |

**-Top 5 states where delivery is really fast compared to estimated date**

**AC>RO>AM>AP>RR**

SELECT

  DISTINCT

  c.customer\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_estimated\_delivery\_date,o.order\_delivered\_customer\_date,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_estimated\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_delivered\_customer\_date,o.order\_purchase\_timestamp,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_delivery\_state,

  ROUND(AVG(oi.freight\_value) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_freight\_state

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

WHERE o.order\_purchase\_timestamp IS NOT NULL

AND o.order\_delivered\_customer\_date IS NOT NULL

AND o.order\_estimated\_delivery\_date IS NOT NULL

ORDER BY average\_day\_order\_estimated\_state DESC

LIMIT 5

| Row | **customer\_state** | **average\_day\_order\_estimated\_state** | **average\_day\_order\_delivery\_state** | **average\_freight\_state** |  |
| --- | --- | --- | --- | --- | --- |
| 1 | AC | 20.01 | 20.33 | 40.05 |  |
| 2 | RO | 19.08 | 19.28 | 41.33 |  |
| 3 | AM | 18.98 | 25.96 | 33.31 |  |
| 4 | AP | 17.44 | 27.75 | 34.16 |  |
| 5 | RR | 17.43 | 27.83 | 43.09 |  |

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

**AL>MA>SE>ES>BA**

SELECT

  DISTINCT

  c.customer\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_estimated\_delivery\_date,o.order\_delivered\_customer\_date,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_estimated\_state,

  ROUND(AVG(DATE\_DIFF(o.order\_delivered\_customer\_date,o.order\_purchase\_timestamp,Day)) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_day\_order\_delivery\_state,

  ROUND(AVG(oi.freight\_value) OVER(

    PARTITION BY c.customer\_state

  ),2) AS average\_freight\_state

FROM `target\_sql.order\_items` AS oi

INNER JOIN `target\_sql.orders` AS o

ON oi.order\_id=o.order\_id

INNER JOIN `target\_sql.customers` AS c

ON o.customer\_id=c.customer\_id

WHERE o.order\_purchase\_timestamp IS NOT NULL

AND o.order\_delivered\_customer\_date IS NOT NULL

AND o.order\_estimated\_delivery\_date IS NOT NULL

ORDER BY average\_day\_order\_estimated\_state

LIMIT 5

| Row | **customer\_state** | **average\_day\_order\_estimated\_state** | **average\_day\_order\_delivery\_state** | **average\_freight\_state** |  |
| --- | --- | --- | --- | --- | --- |
| 1 | AL | 7.98 | 23.99 | 35.87 |  |
| 2 | MA | 9.11 | 21.2 | 38.49 |  |
| 3 | SE | 9.17 | 20.98 | 36.57 |  |
| 4 | ES | 9.77 | 15.19 | 22.03 |  |
| 5 | BA | 10.12 | 18.77 | 26.49 |  |

**6. Payment type analysis**

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

SELECT

  x.payment\_type,

  x.order\_year,

  x.order\_month,

  count(\*) AS count\_of\_orders\_type\_year\_month

FROM(

SELECT

  \*,

  EXTRACT(YEAR FROM o.order\_purchase\_timestamp) AS order\_year,

  EXTRACT(MONTH FROM o.order\_purchase\_timestamp) AS order\_month

FROM `target\_sql.payments` AS p

INNER JOIN `target\_sql.orders` AS o

ON p.order\_id=o.order\_id) AS x

GROUP BY x.payment\_type,x.order\_year,x.order\_month

ORDER BY x.payment\_type,x.order\_year,x.order\_month

| Row | **payment\_type** | **order\_year** | **order\_month** | **count\_of\_orders\_type\_year\_month** |  |
| --- | --- | --- | --- | --- | --- |
| 1 | UPI | 2016 | 10 | 63 |  |
| 2 | UPI | 2017 | 1 | 197 |  |
| 3 | UPI | 2017 | 2 | 398 |  |
| 4 | UPI | 2017 | 3 | 590 |  |
| 5 | UPI | 2017 | 4 | 496 |  |
| 6 | UPI | 2017 | 5 | 772 |  |
| 7 | UPI | 2017 | 6 | 707 |  |
| 8 | UPI | 2017 | 7 | 845 |  |
| 9 | UPI | 2017 | 8 | 938 |  |
| 10 | UPI | 2017 | 9 | 903 |  |
| 11 | UPI | 2017 | 10 | 993 |  |
| 12 | UPI | 2017 | 11 | 1509 |  |

**2. Count of orders based on the no. of payment installments**

SELECT

  payment\_installments,

  count(\*) AS count\_of\_order\_based\_on\_pInstallments

FROM `target\_sql.payments`

GROUP BY payment\_installments

ORDER BY payment\_installments

| Row | **payment\_installments** | **count\_of\_order\_based\_on\_pInstallments** |  |
| --- | --- | --- | --- |
| 1 | 0 | 2 |  |
| 2 | 1 | 52546 |  |
| 3 | 2 | 12413 |  |
| 4 | 3 | 10461 |  |
| 5 | 4 | 7098 |  |
| 6 | 5 | 5239 |  |
| 7 | 6 | 3920 |  |
| 8 | 7 | 1626 |  |
| 9 | 8 | 4268 |  |
| 10 | 9 | 644 |  |
| 11 | 10 | 5328 |  |
| 12 | 11 | 23 |  |
| 13 | 12 | 133 |  |

**7. Actionable Insights**

1. Growing **trend on e-commerce in Brazil** is observed Year-by-Year as no of products ordered by customers **increased year by year** (2016 to 2018). More complete scenario can be obtained by further seeing the month wise trend (From 2016 to 2018). It is clearly visible that as we move from Jan'2017 to Nov'2017, no of orders purchased by customer went from 800 to 7544 and then further dropped down  to 6512 in Aug'2018, which clearly indicates that **there is upward growth on e-commerce in Brazil Year-By-Year** but upward growth is **not strictly increasing** over the whole period, and also observed**few seasonal peaks** .

2. We can observed some**seasonality with peaks** at specific month like **Nov'2017** as no of orders by customer went from 4631 in Oct'2017 to 7544 in Nov'2017. Also No of orders went from 5673 in Dec'2017 to 7269 in **Jan'2018**.Thus, Peak is observed in month of Nov'2017 and Jan'2018. Although , data of Nov'2018 or Jan'2019 is not available to further analyze if peaks are observed every year during this time of month.

3. It is observed that **Brazilian customers tend to buy more during Afternoon** i.e.32211 products ordered which is almost twice as that of products ordered during evening. Only 896 products ordered during **Dawn**which is**least** among all.

4.  **No of customers** :

     Top 5 States: **SP > RJ > MG>RS>PR**

     Bottom 5 states: **RR > AP > AC > AM > RO**

5. **Total % increase** in cost of orders from **2017 to 2018** (including months between Jan to Aug only)= **136.98 %**

6. **Average price of orders :**

    Top 5 states: **PB>AL>AC>RO>PA**

    Bottom 5 states: **SP>PR>RS>MG>ES**

7. **Average days** taken for **order to deliver** **once purchase is made**= **12.09 Days**

8. **Average days of difference** b/w **Order delivery date** & **Order estimated delivery date**= **10.96 Days**. It includes both cases i.e. when order is delivered before estimated delivery date & when order is delivered after estimated delivery date

9. **Average days of difference** b/w Order delivery date & Order estimated delivery date=**12.71 Days** when **order**is **delivered before estimated delivery date** .

**No of orders** delivered before estimated delivery date=**88649**

10. **Average days of difference** b/w Order delivery date & Order estimated delivery date=**8.87 Days** when **order** is **delivered after estimated delivery date** .

**No of orders** delivered after estimated delivery date=**7827**

11.**Total orders** which is **delivered after estimated delivery date** Vs **Average days of difference** b/w Order delivery date & Order estimated delivery date when **order is delivered after estimated delivery date**(Top 5 State sorted in decreasing no of orders)

**SP:  2387, 6.35 Days**

**RJ:  1664, 12.15 Days**

**MG:  638,  6.85 Days**

**BA:  457, 10.42 Days**

**RS:  382, 8.71 Days**

12. **Average freight value**

      Top 5 states :**PB>RR>RO>AC>PI**

      Bottom 5 states: **SP>PR>MG>RJ>DF**

13. **Average time to delivery**

       Top 5 states : **RR>AP>AM>AL>PA**

       Bottom 5 states: **SP>PR>MG>DF>SC**

14. **Delivery is really fast compared to estimated date**

       Top 5 states: **AC>RO>AM>AP>RR**

15. **Delivery is not so fast compared to estimated date**

      Top 5 states :**AL>MA>SE>ES>BA**

16. **Total no of order review score equal to 1**:

       Top 5 states: **SP>RJ>MG>RS>BA**

As there are 2387 orders which are delivered after estimated delivery time in SP state of Brazil, and thus received the maximum no of review score of only 1(On scale of 1-5). Thus customer experience is not very good, and thus average price of order is lowest in state SP among all the states.

**8. Recommendation**

1. As Trend on e-commerce in Brazil is not strictly increasing, and thus efforts can be made on:

-**Improving customer service** by including prompt responses to customer inquiries, fast and reliable shipping, and easy returns and exchanges

-**Optimizing the website** which is easy to navigate, has clear product descriptions and high-quality images, and is optimized for search engines

-To offer **promotions and discount** which includes free shipping, a percentage off their order, or a free gift with purchase

2. As Seasonal peaks are observed during the month of November & January, and thus following can be done to boost the sales:

-**Planning of inventory** to meet high demand, **hiring extra staff**or **outsourcing** to ensure smooth execution

-Promotions and discounts

3. As it is observed that Brazilian customers tend to buy more during Afternoon and almost twice as compared to buying during evening, and following can be recommended:

-**Scheduling e-mail campaign, social media posts** during peak buying hours

-As during afternoon, most customers prefer browsing on mobile devices, and thus making sure to **optimize the website for mobile devices**

-Offering **"Happy Hours" promotions** during evening time/less buying hours to encourage customers to shop during this time

4. Following efforts can be made to increase the customer in a state:

-Target **digital marketing** or**localized print ads** to increase the visibility in those regions

-Localizing the website i.e. using **localized language**as well as **customer assistance chatbot in local language**

-Developing **partnerships** with **local businesses**

5.To increase the % cost of orders or Average price of orders, following recommendation can be made:

-Offering **quantity discounts** when customer purchase multiple quantities of a product

-**Incentives to customers**who make repeat purchases

-Offering **bundle products** i.e. offering a discount when customers buy multiple products together

-Offering **free shipping above a threshold value**

6. To reduce the average days taken for order to deliver once purchase is made, following recommendation can be made:

-**Warehouse located closer** to customer to reduce shipping distance and delivery time

-By improving the **inventory management**

-By **automating the shipping process**

-**Partnership** with**local** vendors and couriers

-**Priority shipping** that guarantees faster delivery option at increased price

7. To decreases the cases when order is delivered after estimated delivery date, following recommendation can be made:

-Improving the inventory management system

-To provide the **accurate shipping estimate** to customers based on current inventory levels, shipping distance or seasonality

-Opting for **reliable shipping carriers**

-To **monitor delivery pipeline**and take corrective actions as required.

-**Incentives to shipping carriers** for on-time delivery, and **penalty** for late deliveries

8. To decrease the average freight price, following recommendation can be made:

-**Optimizing the packaging** to reduce the size and weight of shipments

-To find the **best shipping rates** and shipping carrier options

-Free shipping above a threshold value, which will increase in volume of orders and which in turn offset the freight cost

9. Taking the case of **state SP**, where SP state is having the highest no of customers among all state but the lowest average price of orders and maximum number of review score equal to 1. If worked upon **improving the customer experiences** by implementing the above recommendation, it can certainly boost the quantity and price of orders .