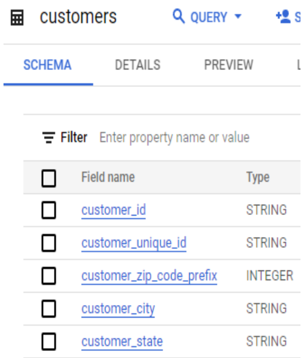
Target SQL

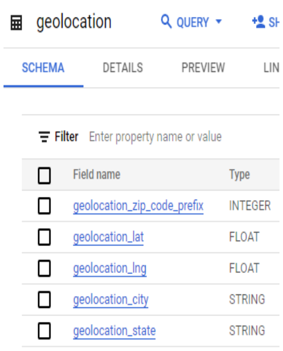
1. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset
    1. Data type of columns in a table

Ans-



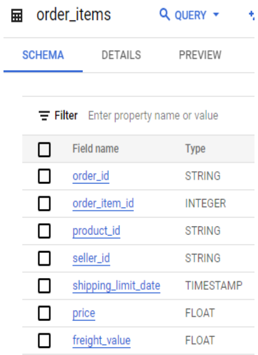
customers

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



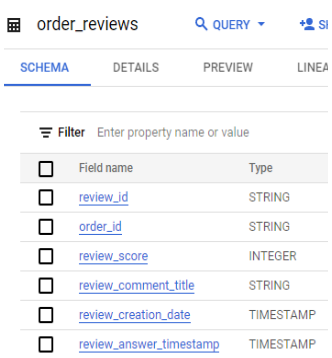
geolocation

Field name	Type
geolocation_zip_code_prefix	INTEGER
geolocation_lat	FLOAT
geolocation_lng	FLOAT
geolocation_city	STRING
geolocation_state	STRING



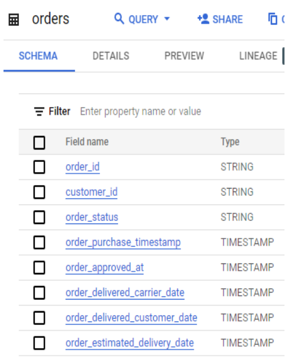
order_items

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



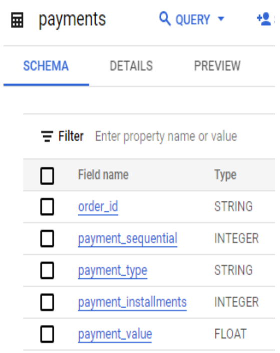
order_reviews

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



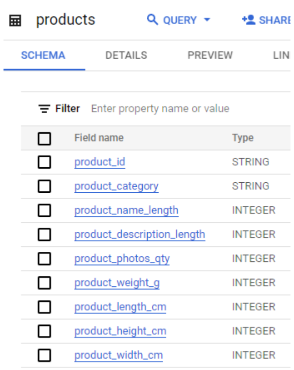
orders

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



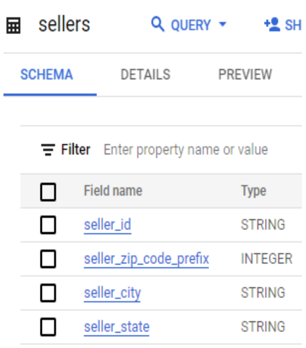
payments

Field name	Type
order_id	STRING
payment_sequential	INTEGER
payment_type	STRING
payment_installments	INTEGER
payment_value	FLOAT



products

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



sellers

Field name	Type
seller_id	STRING
seller_zip_code_prefix	INTEGER
seller_city	STRING
seller_state	STRING

2. Time period for which the data is given

Ans- 2016-2018

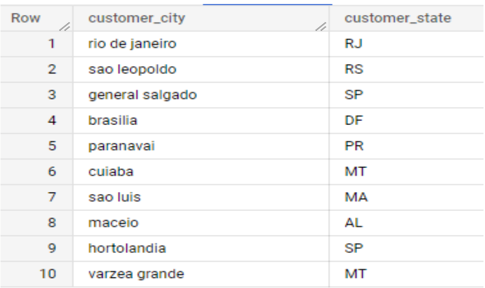
3. Cities and States of customers ordered during the given period

Ans-

```
select distinct c.customer_city,c.customer_
state

from `target.orders` as o

join `target.customers` as c
on o.customer_id=c.customer_id;
```

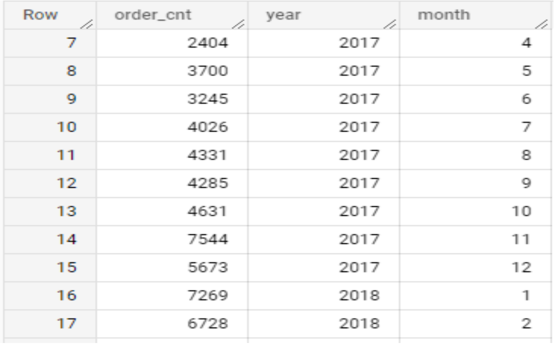


Row	customer_city	customer_state
1	rio de janeiro	RJ
2	sao leopoldo	RS
3	general salgado	SP
4	brasilia	DF
5	paranavai	PR
6	cuiaba	MT
7	sao luis	MA
8	maceio	AL
9	hortolandia	SP
10	varzea grande	MT

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 count(order_id) as order_cnt,
extract (year from order_purchase_timestamp) as year,
extract (month from order_purchase_timestamp) as month
from `target.orders`
group by year,month
order by year,month
```



Row	order_cnt	year	month
7	2404	2017	4
8	3700	2017	5
9	3245	2017	6
10	4026	2017	7
11	4331	2017	8
12	4285	2017	9
13	4631	2017	10
14	7544	2017	11
15	5673	2017	12
16	7269	2018	1
17	6728	2018	2

The numbers of orders placed is highest in month of November .

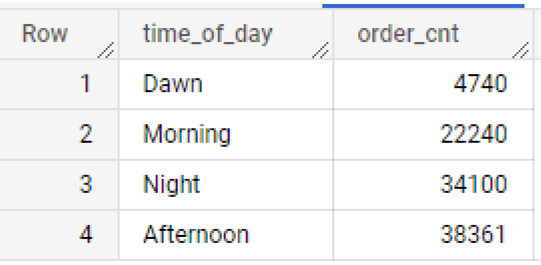
    2. What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

Ans-

```
with cte as (SELECT
  order_id,
  order_purchase_timestamp,
  EXTRACT(HOUR FROM order_purchase_timestamp) AS hour
FROM `target.orders`),

cte2 as ( select order_id,hour,
case when hour between 0 and 5 then 'Dawn'
    when hour between 6 and 11 then 'Morning'
    when hour between 12 and 17 then 'Afternoon'
    else'Night'
    end as part
from cte)

select part as time_of_day, count(order_id) as order_cnt
from cte2
group by part
order by count(order_id)
```



Row	time_of_day	order_cnt
1	Dawn	4740
2	Morning	22240
3	Night	34100
4	Afternoon	38361

```
I divided the day as follows - upto 6am = Dawn, 6am-12pm = Morning , 12pm-6pm = Afternoon,
after 6pm=Night so based on this the sales are highest in Afternoon (12pm-6pm) closely
followed by Night(after 6pm till midnight)
```

3. Evolution of E-commerce orders in the Brazil region:
    1. Get month on month orders by states

Ans –

The formula for Month-over-Month count is: Percent change = (Month 2 – Month 1) / Month 1 * 100

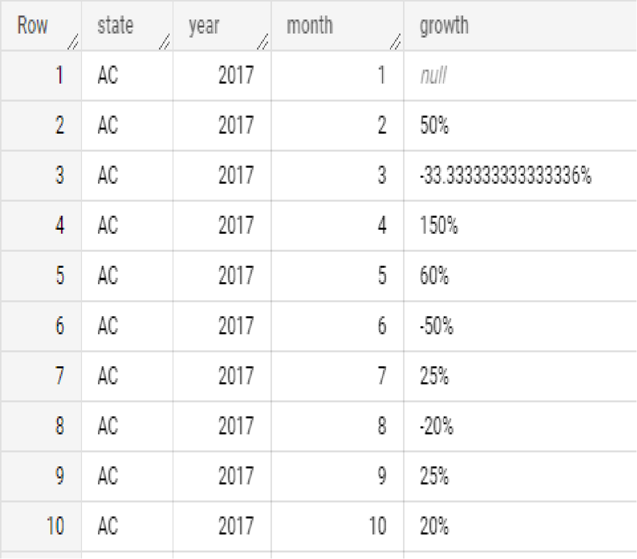
```
with x as (select o.order_id,o.customer_id, o.order_purchase_timestamp, c.customer_state

from `target.orders` o join `target.customers` c
on o.customer_id = c.customer_id),

cte as (select customer_state,
extract (year from order_purchase_timestamp) as year,
extract (month from order_purchase_timestamp) as month,
count(order_id) as order_cnt
from x
group by customer_state,year,month
order by customer_state,year,month),

cte2 as(
select
row_number() over () as rw,
*
from cte)

select customer_state as state,year,month,
# order_cnt, lag(order_cnt,1) over(order by rw) as prv,
100 * (order_cnt - lag(order_cnt,1) over(order by rw)) / lag(order_cnt,1) over(order by rw) ||
 '%' as growth
from cte2
order by customer_state,year,month;
```

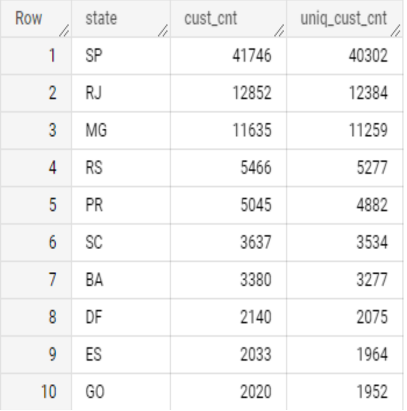


Row	state	year	month	growth
1	AC	2017	1	*null*
2	AC	2017	2	50%
3	AC	2017	3	-33.333333333333336%
4	AC	2017	4	150%
5	AC	2017	5	60%
6	AC	2017	6	-50%
7	AC	2017	7	25%
8	AC	2017	8	-20%
9	AC	2017	9	25%
10	AC	2017	10	20%

2. Distribution of customers across the states in Brazil

Ans-

```
select customer_State as state,
count(distinct customer_id) as cust_cnt,
count(distinct customer_unique_id) as uniq_cust_cnt
from `target.customers`
group by customer_state
order by cust_cnt desc;
```



Row	state	cust_cnt	uniq_cust_cnt
1	SP	41746	40302
2	RJ	12852	12384
3	MG	11635	11259
4	RS	5466	5277
5	PR	5045	4882
6	SC	3637	3534
7	BA	3380	3277
8	DF	2140	2075
9	ES	2033	1964
10	GO	2020	1952

Majority of customers are located in Sao Paulo, Rio De Janeiro and Minas Gerais while lowest number of customers in Roraima, Amapá, Acre

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) - You can use “payment_value” column in payments table

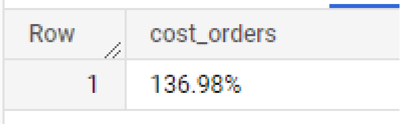
Ans-

```
with cte as (select extract (year from order_purchase_timestamp) as year,
extract (month from order_purchase_timestamp) as month,
p.payment_value
from `target.orders` o join `target.payments` p
on o.order_id = p.order_id
where  extract (year from order_purchase_timestamp) in (2017,2018) and extract (month from ord
er_purchase_timestamp)<=8),

cte2 as (select year,round(sum(payment_value),2)as payment_value from cte
group by year
order by year),
cte3 as (

select * , lag(payment_value,1) over(order by year) as prv,
payment_value- lag(payment_value,1) over(order by year) as diff,
from cte2
order by year)

select round(100*diff/prv ,2) || '%' as cost_orders
from cte3
where round(100*diff/prv ,2) is not null;
```



Row	cost_orders
1	136.98%

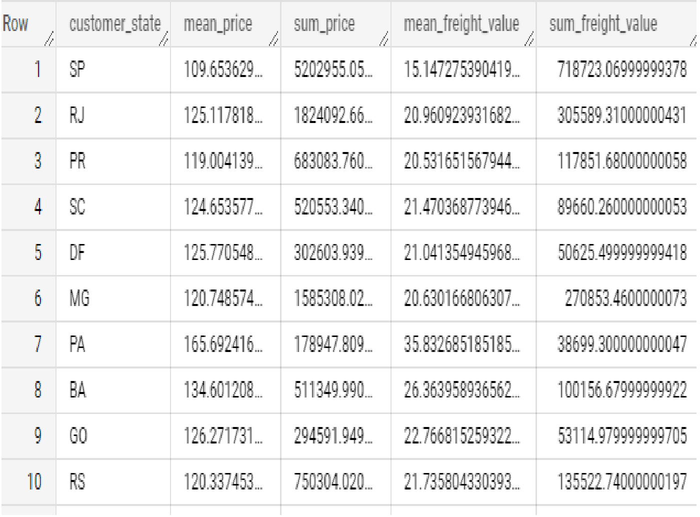
There is a 137% increase in cost of orders from 2017 to 2018 (for Jan-Aug months).

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

Ans-

```
with cte as
(select oi.order_id,oi.price,
oi.freight_value, o.customer_id, c.customer_state
from `target.order_items` oi
join `target.orders` o
on oi.order_id=o.order_id
join `target.customers` c on
o.customer_id = c.customer_id)

select customer_state,
sum(price)/count(price) as mean_price,
sum(price) as sum_price,
sum(freight_value)/count(freight_value)
as mean_freight_value,
sum(freight_value) as sum_freight_value
from cte
group by customer_state
```



Row	customer_state	mean_price	sum_price	mean_freight_value	sum_freight_value
1	SP	109.653629...	5202955.05...	15.147275390419...	718723.06999999378
2	RJ	125.117818...	1824092.66...	20.960923931682...	305589.31000000431
3	PR	119.004139...	683083.760...	20.531651567944...	117851.68000000058
4	SC	124.653577...	520553.340...	21.470368773946...	89660.260000000053
5	DF	125.770548...	302603.939...	21.041354945968...	50625.499999999418
6	MG	120.748574...	1585308.02...	20.630166806307...	270853.4600000073
7	PA	165.692416...	178947.809...	35.832685185185...	38699.300000000047
8	BA	134.601208...	511349.990...	26.363958936562...	100156.67999999922
9	GO	126.271731...	294591.949...	22.766815259322...	53114.979999999705
10	RS	120.337453...	750304.020...	21.735804330393...	135522.7400000197

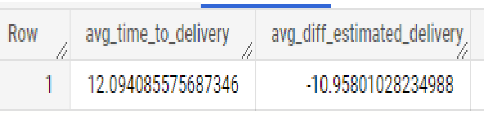
5. Analysis on sales, freight and delivery time

1. Calculate days between purchasing, delivering and estimated delivery

Ans-

```
with cte as (select order_id,order_purchase_timestamp,order_delivered_customer_date,order_esti
mated_delivery_date,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) AS time_to_delivery,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date, DAY) AS diff_estimated_
delivery,
from `target.orders`)

select avg(time_to_delivery) as avg_time_to_delivery,
avg(diff_estimated_delivery) as avg_diff_estimated_delivery
from cte
```



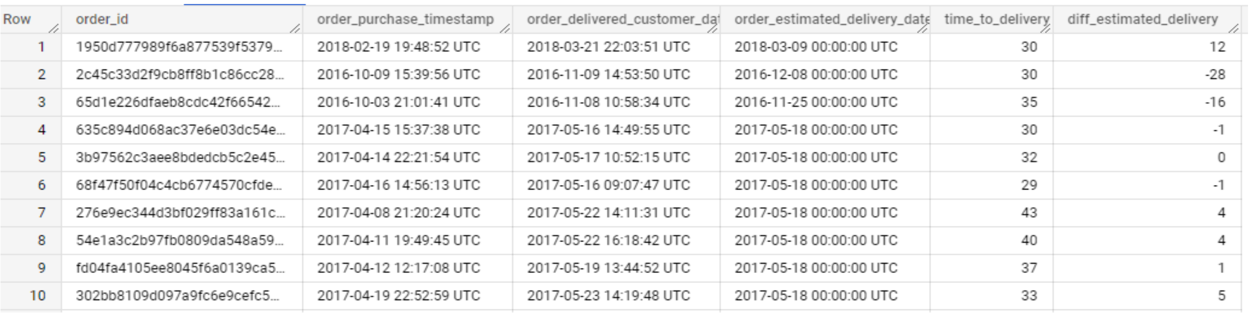
Row	avg_time_to_delivery	avg_diff_estimated_delivery
1	12.094085575687346	-10.95801028234988

The average time to delivery (between purchase and delivery) is about 12 days.
The average time between estimated and actual delivery is about 10 days i.e. products arrive 10 days earlier than expected on an average.

2. Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:
    - time_to_delivery = order_purchase_timestamp-order_delivered_customer_date
    - diff_estimated_delivery = order_estimated_delivery_date-order_delivered_customer_date

ans-

```
select order_id,order_purchase_timestamp,order_delivered_customer_date,order_estimated_delivery_date,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) AS time_to_delivery,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date, DAY) AS diff_estimated_delivery,
from `target.orders`
```



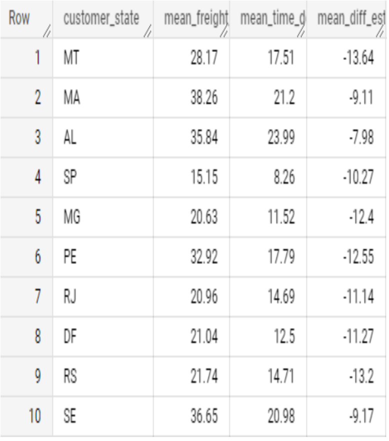
Row	order_id	order_purchase_timestamp	order_delivered_customer_dat	order_estimated_delivery_date	time_to_delivery	diff_estimated_delivery
1	1950d777989f6a877539f5379...	2018-02-19 19:48:52 UTC	2018-03-21 22:03:51 UTC	2018-03-09 00:00:00 UTC	30	12
2	2c45c33d2f9cb8ff8b1c86cc28...	2016-10-09 15:39:56 UTC	2016-11-09 14:53:50 UTC	2016-12-08 00:00:00 UTC	30	-28
3	65d1e226dfaeb8cdc42f66542...	2016-10-03 21:01:41 UTC	2016-11-08 10:58:34 UTC	2016-11-25 00:00:00 UTC	35	-16
4	635c894d068ac37e6e03dc54e...	2017-04-15 15:37:38 UTC	2017-05-16 14:49:55 UTC	2017-05-18 00:00:00 UTC	30	-1
5	3b97562c3aee8bdedcb5c2e45...	2017-04-14 22:21:54 UTC	2017-05-17 10:52:15 UTC	2017-05-18 00:00:00 UTC	32	0
6	68f47f50f04c4cb6774570cfde...	2017-04-16 14:56:13 UTC	2017-05-16 09:07:47 UTC	2017-05-18 00:00:00 UTC	29	-1
7	276e9ec344d3bf029ff83a161c...	2017-04-08 21:20:24 UTC	2017-05-22 14:11:31 UTC	2017-05-18 00:00:00 UTC	43	4
8	54e1a3c2b97fb0809da548a59...	2017-04-11 19:49:45 UTC	2017-05-22 16:18:42 UTC	2017-05-18 00:00:00 UTC	40	4
9	fd04fa4105ee8045f6a0139ca5...	2017-04-12 12:17:08 UTC	2017-05-19 13:44:52 UTC	2017-05-18 00:00:00 UTC	37	1
10	302bb8109d097a9fc6e9cefc5...	2017-04-19 22:52:59 UTC	2017-05-23 14:19:48 UTC	2017-05-18 00:00:00 UTC	33	5

3. Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery

Ans-

```
with cte as (select o.order_id,o.order_purchase_timestamp,o.order_delivered_customer_date,o.order_estimated_delivery_date,
c.customer_id,c.customer_state,oi.freight_value,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) AS time_to_delivery,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date, DAY) AS diff_estimated_delivery,
from `target.orders` o join `target.customers` c
on o.customer_id = c.customer_id
join `target.order_items` oi
on o.order_id=oi.order_id)

select distinct customer_state,
round(avg(freight_value),2)
as mean_freight_val,
round(avg(time_to_delivery),2)
as mean_time_delivery,
round(avg(diff_estimated_delivery),2)
as mean_diff_estimated_delivery
from cte
group by customer_state
```



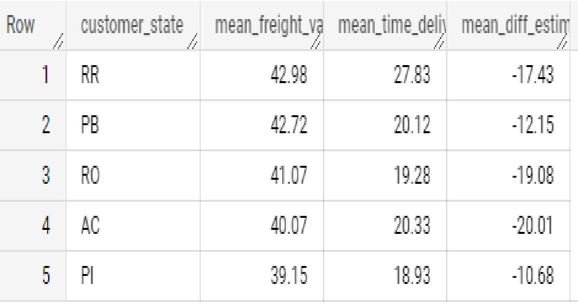
Row	customer_state	mean_freight	mean_time_d	mean_diff_est
1	MT	28.17	17.51	-13.64
2	MA	38.26	21.2	-9.11
3	AL	35.84	23.99	-7.98
4	SP	15.15	8.26	-10.27
5	MG	20.63	11.52	-12.4
6	PE	32.92	17.79	-12.55
7	RJ	20.96	14.69	-11.14
8	DF	21.04	12.5	-11.27
9	RS	21.74	14.71	-13.2
10	SE	36.65	20.98	-9.17

4. Sort the data to get the following:
5. Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

Ans-

```
with cte as (select o.order_id,o.order_purchase_timestamp,o.order_delivered_customer_date,o.or
der_estimated_delivery_date,
c.customer_id,c.customer_state,oi.freight_value,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) AS time_to_delivery,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date, DAY) AS diff_estimated_
delivery,
from `target.orders` o join `target.customers` c
on o.customer_id = c.customer_id
join `target.order_items` oi
on o.order_id=oi.order_id)

select distinct customer_state,
round(avg(freight_value),2)
as mean_freight_val,
round(avg(time_to_delivery),2)
as mean_time_delivery,
round(avg(diff_estimated_delivery),2)
as mean_diff_estimated_delivery
from cte
group by customer_state
order by mean_freight_val desc
limit 5
```

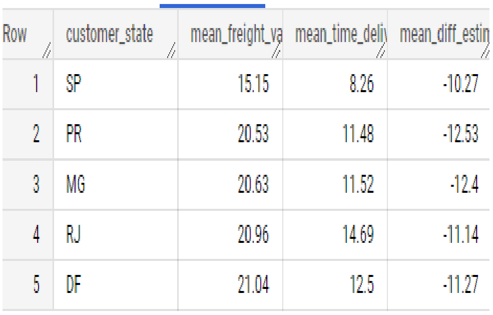


Row	customer_state	mean_freight_va	mean_time_deli	mean_diff_estim
1	RR	42.98	27.83	-17.43
2	PB	42.72	20.12	-12.15
3	RO	41.07	19.28	-19.08
4	AC	40.07	20.33	-20.01
5	PI	39.15	18.93	-10.68

Roraima has highest mean of freight value.

```
with cte as (select o.order_id,o.order_purchase_timestamp,o.order_delivered_customer_date,o.or
der_estimated_delivery_date,
c.customer_id,c.customer_state,oi.freight_value,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) AS time_to_delivery,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date, DAY) AS diff_estimated_
delivery,
from `target.orders` o join `target.customers` c
on o.customer_id = c.customer_id
join `target.order_items` oi
on o.order_id=oi.order_id)

select distinct customer_state,
round(avg(freight_value),2)
 as mean_freight_val,
round(avg (time_to_delivery),2)
 as mean_time_delivery,
round(avg (diff_estimated_delivery),2)
 as mean_diff_estimated_delivery
from cte
group by customer_state
order by mean_freight_val asc
limit 5
```



Row	customer_state	mean_freight_va	mean_time_deli	mean_diff_estin
1	SP	15.15	8.26	-10.27
2	PR	20.53	11.48	-12.53
3	MG	20.63	11.52	-12.4
4	RJ	20.96	14.69	-11.14
5	DF	21.04	12.5	-11.27

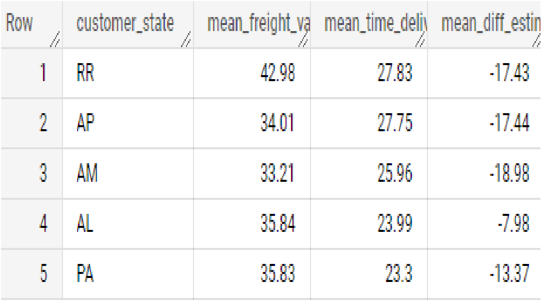
São Paulo has lowest mean of freight value.

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

Ans-

```
with cte as (select o.order_id,o.order_purchase_timestamp,o.order_delivered_customer_date,o.or
der_estimated_delivery_date,
c.customer_id,c.customer_state,oi.freight_value,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) AS time_to_delivery,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date, DAY) AS diff_estimated_
delivery,
from `target.orders` o join `target.customers` c
on o.customer_id = c.customer_id
join `target.order_items` oi
on o.order_id=oi.order_id)

select distinct customer_state,
round(avg (freight_value), 2)
as mean_freight_val,
round(avg (time_to_delivery),2)
as mean_time_delivery,
round(avg (diff_estimated_delivery), 2)
 as mean_diff_estimated_delivery
from cte
group by customer_state
order by mean_time_delivery desc
limit 5
```

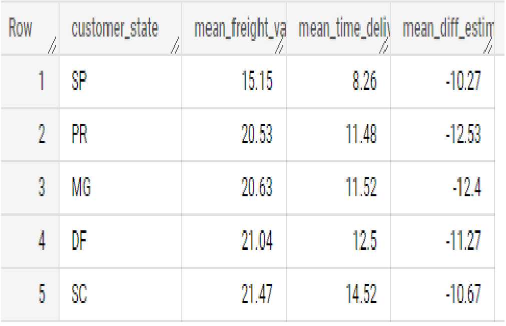


Row	customer_state	mean_freight_va	mean_time_deli	mean_diff_estin
1	RR	42.98	27.83	-17.43
2	AP	34.01	27.75	-17.44
3	AM	33.21	25.96	-18.98
4	AL	35.84	23.99	-7.98
5	PA	35.83	23.3	-13.37

Roraima has highest mean time to delivery.

```
-----------------------------------------------------------------------------------------------
with cte as (select o.order_id,o.order_purchase_timestamp,o.order_delivered_customer_date,o.or
der_estimated_delivery_date,
c.customer_id,c.customer_state,oi.freight_value,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) AS time_to_delivery,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date, DAY) AS diff_estimated_
delivery,
from `target.orders` o join `target.customers` c
on o.customer_id = c.customer_id
join `target.order_items` oi
on o.order_id=oi.order_id)

select distinct customer_state,
round(avg (freight_value),2)
as mean_freight_val,
round(avg (time_to_delivery),2)
 as mean_time_delivery,
round(avg (diff_estimated_delivery),2)
 as mean_diff_estimated_delivery
from cte
group by customer_state
```



Row	customer_state	mean_freight_va	mean_time_deli	mean_diff_estim
1	SP	15.15	8.26	-10.27
2	PR	20.53	11.48	-12.53
3	MG	20.63	11.52	-12.4
4	DF	21.04	12.5	-11.27
5	SC	21.47	14.52	-10.67

```
order by mean_time_delivery asc
limit 5
```

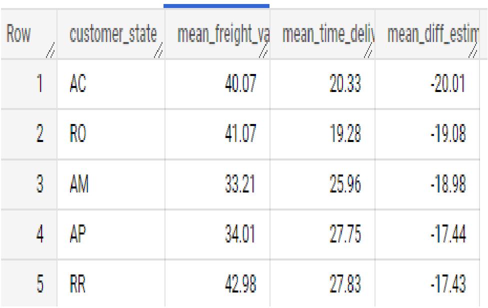
São Paulo has lowest mean time to delivery.

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

Ans-

```
with cte as (select o.order_id,o.order_purchase_timestamp,o.order_delivered_customer_date,o.or
der_estimated_delivery_date,
c.customer_id,c.customer_state,oi.freight_value,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) AS time_to_delivery,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date, DAY) AS diff_estimated_
delivery,
from `target.orders` o join `target.customers` c
on o.customer_id = c.customer_id
join `target.order_items` oi
on o.order_id=oi.order_id)

select distinct customer_state,
round(avg (freight_value),2)
 as mean_freight_val,
round(avg (time_to_delivery),2)
as mean_time_delivery,
round(avg (diff_estimated_delivery),2)
 as mean_diff_estimated_delivery
from cte
group by customer_state
order by mean_diff_estimated_delivery asc
limit 5
```

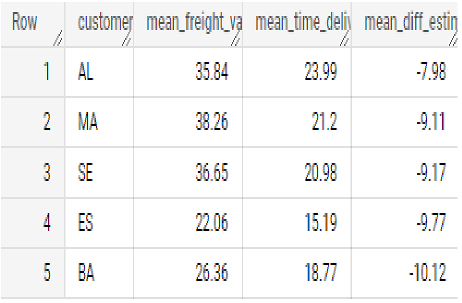


Row	customer_state	mean_freight_va	mean_time_deliv	mean_diff_estim
1	AC	40.07	20.33	-20.01
2	RO	41.07	19.28	-19.08
3	AM	33.21	25.96	-18.98
4	AP	34.01	27.75	-17.44
5	RR	42.98	27.83	-17.43

Acre has fastest delivery about 20 days earlier than estimated.

```
with cte as (select o.order_id,o.order_purchase_timestamp,o.order_delivered_customer_date,o.or
der_estimated_delivery_date,
c.customer_id,c.customer_state,oi.freight_value,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp, DAY) AS time_to_delivery,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date, DAY) AS diff_estimated_
delivery,
from `target.orders` o join `target.customers` c
on o.customer_id = c.customer_id
join `target.order_items` oi
on o.order_id=oi.order_id)

select distinct customer_state,
round(avg (freight_value), 2)
 as mean_freight_val,
round(avg (time_to_delivery),2)
as mean_time_delivery,
round(avg (diff_estimated_delivery),2)
 as mean_diff_estimated_delivery
```



Row	customer	mean_freight_va	mean_time_deliv	mean_diff_estin
1	AL	35.84	23.99	-7.98
2	MA	38.26	21.2	-9.11
3	SE	36.65	20.98	-9.17
4	ES	22.06	15.19	-9.77
5	BA	26.36	18.77	-10.12

```
from cte
group by customer_state
order by mean_diff_estimated_delivery desc
limit 5
```

Alagoas has not so fast delivery just about a week earlier than estimated.

6. Payment type analysis:

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

Ans-

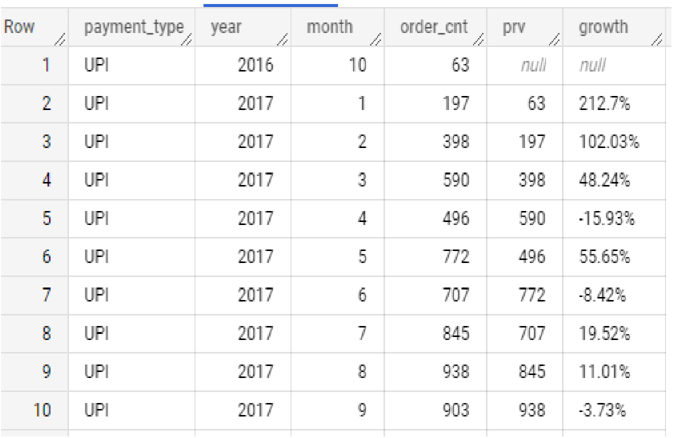
The formula for Month-over-Month count is: Percent change = (Month 2 – Month 1) / Month 1 * 100

```
with x as (select o.order_id,o.customer_id, o.order_purchase_timestamp, p.payment_type
from `target.orders` o join `target.payments` p
on o.order_id = p.order_id),

cte as (select payment_type,
extract (year from order_purchase_timestamp) as year,
extract (month from order_purchase_timestamp) as month,
count(order_id) as order_cnt
from x
group by payment_type,year,month
order by payment_type,year,month),

cte2 as(
select
row_number() over () as rw,
*
from cte)


select payment_type as payment_type,
year,
month,
order_cnt,
lag(order_cnt,1) over(order by rw) as prv,
round(100 * (order_cnt - lag(order_cnt,1) over(order by rw)) / lag(order_cnt,1) over(order by
rw),2) || '%' as growth
from cte2
order by payment_type,year,month;
```

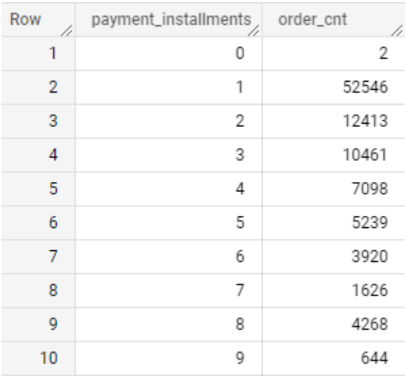


Row	payment_type	year	month	order_cnt	prv	growth
1	UPI	2016	10	63	null	null
2	UPI	2017	1	197	63	212.7%
3	UPI	2017	2	398	197	102.03%
4	UPI	2017	3	590	398	48.24%
5	UPI	2017	4	496	590	-15.93%
6	UPI	2017	5	772	496	55.65%
7	UPI	2017	6	707	772	-8.42%
8	UPI	2017	7	845	707	19.52%
9	UPI	2017	8	938	845	11.01%
10	UPI	2017	9	903	938	-3.73%

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

Ans-

```
select p.payment_installments, count(o.order_id) as order_cnt
from `target.orders` o join `target.payments` p
on o.order_id = p.order_id
group by p.payment_installments
```



Row	payment_installments	order_cnt
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

Maximum number of orders were placed with just a single payment installment.

Actionable Insights

1) Sales are highest in November
2) Sales are highest from 12-6pm closely followed by 6pm to 12 midnight
3) Majority of customers are located in Sao Paulo, Rio De Janeiro and Minas Gerais while lowest number of customers in Roraima, Amapá, Acre
4) The average time to delivery (between purchase and delivery) is about 12 days.
5) The average time between estimated and actual delivery is about 10 days i.e. products arrive 10 days earlier than expected on an average.
6) Roraima has highest mean of freight value.
7) São Paulo has lowest mean of freight value.
8) Roraima has highest mean time to delivery.
9) São Paulo has lowest mean time to delivery.
10) Maximum number of orders were placed with just a single payment installment.
11) There is a 137% increase in cost of orders from 2017 to 2018 (for Jan-Aug months).

Recommendations

1) Stock up inventory in November
2) Clearance sale in November
3) Offer more deals during 12 noon to 12 midnight
4) More deals, varieties and offers in states having majority customers like Sao Paulo, Rio De Janeiro and Minas Gerais
5) More deals and promotions in states like Sao Paulo where time to delivery is lowest.