

## QUESTIONS TO SOLVE:

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
  3. Cities and States covered in the dataset
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?
  2. What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?
3. Evolution of E-commerce orders in the Brazil region:
  1. Get month on month orders by region, states
  2. How are customers distributed in Brazil
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)
  2. Mean & Sum of price and freight value by customer state
5. Analysis on sales, freight and delivery time
  1. Calculate days between purchasing, delivering and estimated delivery
  2. Create columns:
    - $\text{time\_to\_delivery} = \text{order\_purchase\_timestamp} - \text{order\_delivered\_customer\_date}$
    - $\text{diff\_estimated\_delivery} = \text{order\_estimated\_delivery\_date} - \text{order\_delivered\_customer\_date}$
  3. Group data by state, take mean of freight\_value, time\_to\_delivery, diff\_estimated\_delivery
  4. Sort the data to get the following:
    - Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5
    - Top 5 states with highest/lowest average time to delivery
    - Top 5 states where delivery is really fast/ not so fast compared to estimated date
6. Payment type analysis:
  1. Month over Month count of orders for different payment types
  2. Distribution of payment installments and count of orders

## Evaluation Criteria (80 points)

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)
- 9.

INSIGHTS ARE HIGHLIGHTED IN GREEN AND QUESTIONS IN RED.

## Approach:

### QUESTION – 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

customers

QUERY

SHARE

SCHEMA

DETAILS

PREVIEW

Filter

Enter property name or value

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

geolocation

QUERY

SHARE

SCHEMA

DETAILS

PREVIEW

Filter

Enter property name or value

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

order\_items

QUERY

SHARE

SCHEMA

DETAILS

PREVIEW

Filter

Enter property name or value

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

order\_reviews

QUERY

SHARE

SCHEMA

DETAILS

PREVIEW

Filter

Enter property name or value

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

orders

QUERY

SHARE

COPY

SCHEMA

DETAILS

PREVIEW

Filter

Enter property name or value

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

payments

QUERY

SHARE

SCHEMA

DETAILS

PREVIEW

Filter

Enter property name or value

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

products

QUERY

SHARE

SCHEMA

DETAILS

PREVIEW

Filter

Enter property name or value

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

sellers

QUERY

SHARE

SCHEMA

DETAILS

PREVIEW

Filter

Enter property name or value

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

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

```
SELECT min(order_purchase_timestamp) as data_start_date, max(order_purchase_timestamp) as data_end_date
FROM `target-sql-358310.TARGET_CASE_STUDY.orders`
```

## Query results

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	
Row	//	data_start_date	//	data_end_date	//
1		2016-09-04 21:15:19 UTC		2018-10-17 17:30:18 UTC	

## 3. Cities and States covered in the dataset

```
SELECT distinct geolocation_city, geolocation_state
FROM `target-sql-358310.TARGET_CASE_STUDY.geolocation`
```

Or

```
SELECT distinct customer_city, customer_state
FROM `target-sql-358310.TARGET_CASE_STUDY.customers`
```

union

```
SELECT distinct seller_city, seller_state
FROM `target-sql-358310.TARGET_CASE_STUDY.sellers`
```

Row	//	geolocation_city	//	geolocation_state
1		aracaju		SE
2		riachuelo		SE
3		nossa senhora do socorro		SE
4		barra dos coqueiros		SE
5		itaporanga d'ajuda		SE
6		sao cristovao		SE
7		são cristóvão		SE
8		santo amaro das brotas		SE
9		pirambu		SE
10		laranjeiras		SE

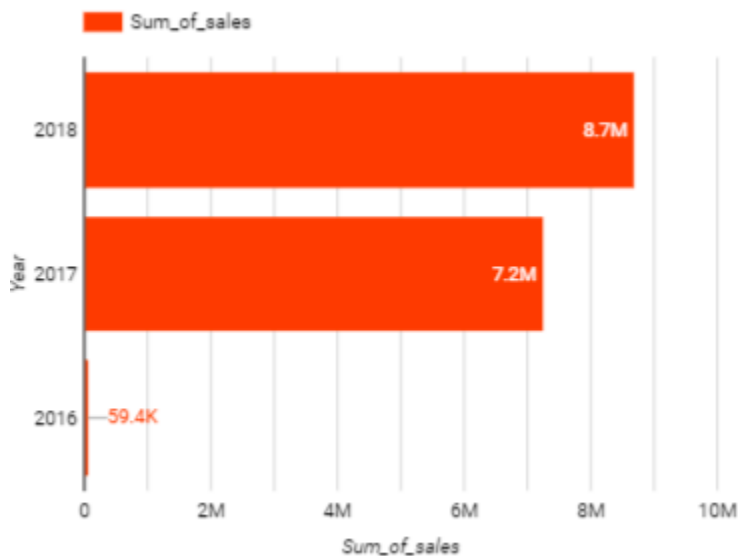
## INSIGHTS:

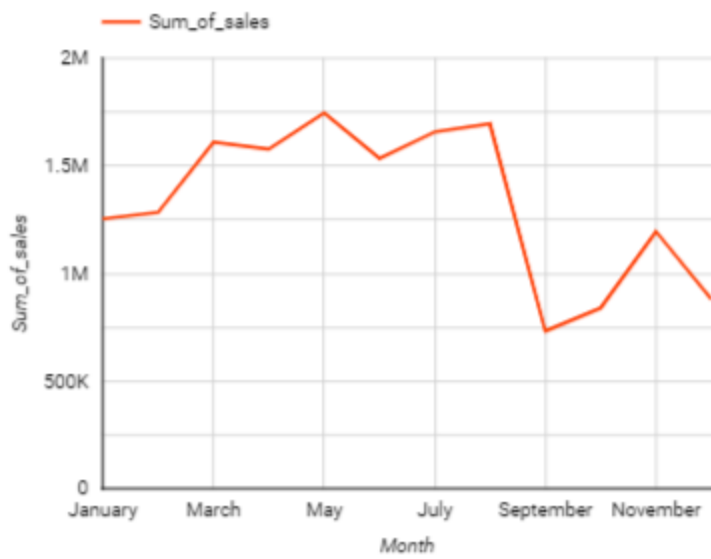
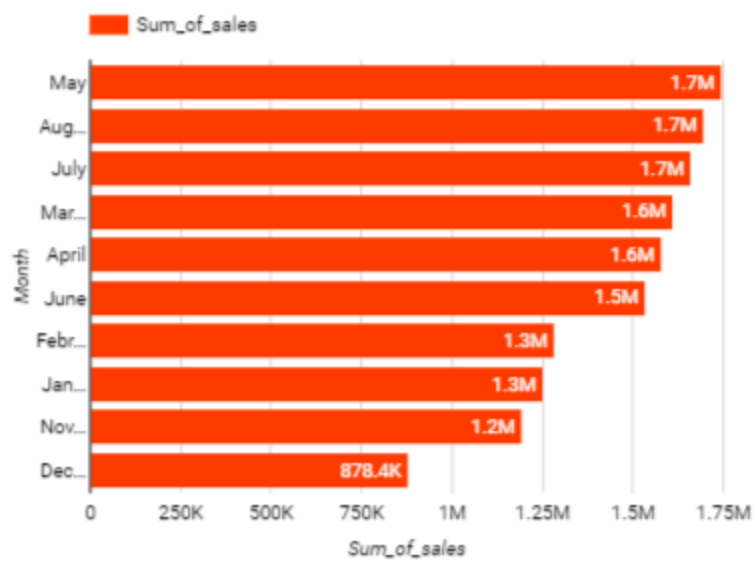
- **DATE IS FROM SEP 2016 TO OCT 2018.**
- **DATA TYPE OF EACH COLUMN ARE SHOWN IN SCREEN SHOTS ABOVE.**
- **CITIES and States covered in the dataset ARE UNION OF BOTH CUSTOMER AND SELLER DATA.**

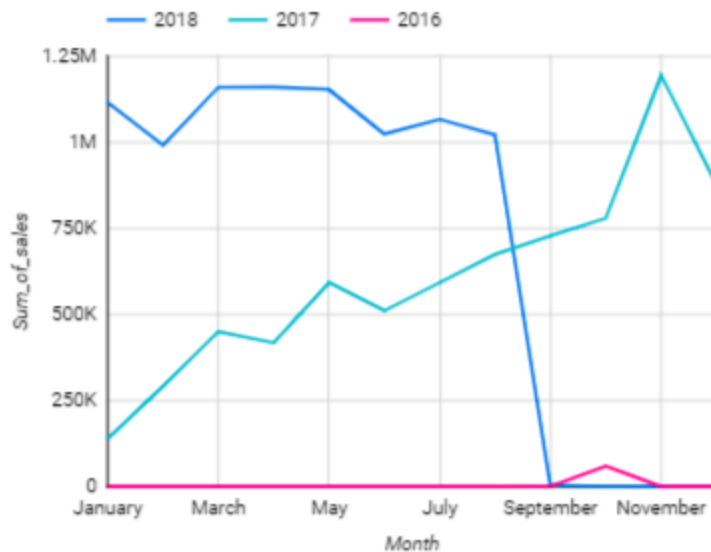
## **QUESTION – 2: In-depth Exploration:**

**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?**

```
select *  
from (  
SELECT extract (Year from order_purchase_timestamp) as Year,  
       extract (Month from order_purchase_timestamp) as Month, round(sum(payment_value),0) as Sum  
_of_sales  
FROM `target-sql-358310.TARGET_CASE_STUDY.orders` a  
left join `target-sql-358310.TARGET_CASE_STUDY.payments` b on a.order_id = b.order_id  
group by extract (Year from order_purchase_timestamp),  
         extract (Month from order_purchase_timestamp) )  
order by Year, Month, Sum_of_sales
```







## INSIGHTS:

- **YOY SALES IS INCREASING.**
- **WE COULD SEE THE TREND AROUND MAY MONTH IN BOTH 2017 AND 2018.**
- **AND NEXT TREND IS SEEN AROUND AUGUST MONTH IN BOTH 2017 AND 2018.**
- **Overall, There is pattern in the data and increase in YOY sales.**

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

```

Select Hour_of_order,
       round(sum(payment_value),0) as Sum_of_sales
       ,count(distinct(a.order_id)) as Count_of_orde
from (
SELECT case when extract(HOUR from order_purchase_timestamp) between 0 and 6 then '1. Early morning 0-6'
           when extract(HOUR from order_purchase_timestamp) between 7 and 12 then '2. Morning 7-12'
           when extract(HOUR from order_purchase_timestamp) between 13 and 18 then '3. Evening 13-18'
           else '4. Late Evening 19-24' end as Hour_of_order
       ,payment_value
       ,a.order_id
FROM `target-sql-358310.TARGET_CASE_STUDY.orders` a

```

```

left join `target-sql-358310.TARGET_CASE_STUDY.payments` b on a.order_id = b.order_id ) A
group by Hour_of_order
order by round(sum(payment_value),0) desc

```

## Query results

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS
Row	Hour_of_order	Sum_of_sales	Count_of_or...	
1	3.Evening 13-18	6252720.0	38135	
2	4. Late Evening 19-24	4505668.0	28331	
3	2. Morning 7-12	4469481.0	27733	
4	1. Early morning 0-6	781003.0	5242	

## INSIGHTS:

- More purchases are done in the evening – 13 to 18 hours of the day.
- Less purchases are done in the early morning – 0 to 6 hours of the day.

## QUESTION – 3: Evolution of E-commerce orders in the Brazil region:

### 1. Get month on month orders by region, states

```

select *
from (
SELECT c.customer_state
      ,c.customer_city
      ,extract (Year from order_purchase_timestamp) as Year
      ,extract (Month from order_purchase_timestamp) as Month
      ,round(sum(payment_value),0) as Sum_of_sales
      ,count(payment_value) as Count_of_orders
FROM `target-sql-358310.TARGET_CASE_STUDY.orders` a
left join `target-sql-358310.TARGET_CASE_STUDY.payments` b on a.order_id = b.order_id
left join `target-sql-358310.TARGET_CASE_STUDY.customers` c on a.customer_id = c.customer_id
group by c.customer_state
      ,c.customer_city
      ,extract (Year from order_purchase_timestamp)
      ,extract (Month from order_purchase_timestamp) ) A

```



order by customer\_state, customer\_city, Year, Month

Year	(All)														
Sum of Sum_of_sales	Month														
customer_state		1	2	3	4	5	6	7	8	9	10	11	12	Grand Total	
MG		153,717	161,596	194,491	177,698	181,402	181,141	180,629	186,368	85,620	100,901	157,774	110,875	1,872,212	
BA		49,425	46,933	64,531	60,431	63,873	66,337	74,639	60,360	28,328	30,669	37,550	33,589	616,665	
DF		22,862	35,734	33,108	30,768	38,448	33,546	40,398	35,693	15,594	16,826	32,233	19,932	355,142	
GO		24,859	24,571	34,253	29,429	47,086	28,677	39,951	39,352	16,461	21,479	25,371	18,603	350,092	
ES		26,780	25,901	26,611	26,636	38,084	33,203	42,801	34,945	13,314	13,734	28,881	15,070	325,960	
CE		18,182	16,670	30,070	27,948	27,825	31,800	27,100	25,662	19,705	18,391	22,535	13,587	279,475	
PA		19,683	21,957	25,622	20,600	15,733	18,871	22,621	28,429	7,730	10,251	14,432	12,383	218,312	
MT		17,140	13,431	13,195	18,273	22,524	17,956	23,997	13,616	8,268	13,988	13,432	11,222	187,042	
MA		12,196	13,740	13,933	14,150	13,094	10,322	16,301	13,315	11,340	13,525	13,061	7,542	152,519	
PB		8,486	8,889	13,005	9,348	9,407	21,624	26,447	12,375	7,048	11,869	6,034	7,016	141,548	
MS		12,200	18,056	16,349	9,262	15,815	15,208	12,602	11,021	5,668	7,687	7,451	6,211	137,530	
AL		6,939	8,649	9,605	12,577	10,122	6,129	7,854	12,590	5,192	6,472	7,445	3,390	96,964	
PE		3,142	3,558	2,619	2,528	2,039	2,999	5,304	3,833	1,577	3,669	1,621	1,984	34,873	
AM		3,433	3,610	2,400	2,069	3,635	1,416	3,550	1,996	1,518	165	3,001	1,174	27,967	
AC		1,881	897	695	1,940	3,884	1,107	2,564	1,371	2,030	959	945	1,409	19,682	
AP		2,789	1,793	1,117	1,043	2,698	618	1,905	642	499	1,697	689	771	16,261	
Grand Total		383,714	405,985	481,604	444,700	495,669	470,954	528,663	481,568	229,892	272,282	372,455	264,758	4,832,244	

Year	(All)														
Sum of Sum_of_sales	Month														
customer_state	customer_city	1	2	3	4	5	6	7	8	9	10	11	12	Grand Total	
AC	rio branco	1,881	757	695	1,688	2,978	1,107	2,564	1,371	1,680	819	647	742	16,929	
AC	cruzeiro do sul					906							221	1,127	
AC	senador guimard									101			446	547	
AC	xapuri				148							298		446	
AC	manoel urbano									249				249	
AC	epitaciolandia										140			140	
AC	brasileia		140											140	
AC	porto acre				104									104	
AL	maceio	4,618	5,326	7,590	8,544	5,637	2,275	4,618	4,499	1,690	3,984	3,287	1,104	53,172	
AL	arapiraca	739	911	934	1,040	701	236	253	505	2,329	598	1,235	671	10,152	
AL	teotonio vilela	267	286		583		199		316			791	158	2,600	
AL	sao sebastiao								2,270					2,270	
AL	maragogi			71		182	127		1,651	105				2,136	
AL	penedo	160	186	99	247		574			108	219			1,593	
AL	palmeira dos indios				241	382		298	154	141			247	1,463	
AL	coruripe					162	1,063	134						1,359	
AL	delmiro gouveia		312					420		316	306			1,354	
AL	santana do mundau						389	665				214		1,268	
AL	inhapi								1,075					1,075	
AL	sao miguel dos campos					122						932		1,054	
AL	campo alegre				853					145				998	
AL	uniao dos palmares										309		640	949	
AL	rio largo				254	114		86			173	66		693	

Sum of Sum_of_sales	Month														
customer_city		1	2	3	4	5	6	7	8	9	10	11	12	Grand Total	
belo horizonte		32,508	32,414	52,415	43,403	38,754	43,869	36,421	44,587	17,482	18,702	32,620	28,589	421,764	
brasilia		22,862	35,734	32,926	30,768	38,448	33,184	40,042	35,668	15,594	16,826	32,233	19,932	354,217	
salvador		16,039	12,557	21,577	26,338	24,261	27,344	24,649	20,408	6,712	8,328	15,189	14,669	218,071	
goiania		8,703	8,569	11,948	10,980	14,061	11,037	18,632	15,705	4,158	8,774	7,268	5,660	125,495	
fortaleza		5,371	6,324	14,817	13,402	11,318	12,035	13,049	9,471	6,799	9,483	12,178	5,616	119,863	
belem		8,163	10,424	11,813	9,612	5,535	8,948	8,766	13,533	2,241	3,947	7,419	6,480	96,881	
juiz de fora		4,712	9,825	5,647	6,345	7,073	8,613	7,004	8,291	5,534	3,195	6,834	2,758	75,831	
campo grande		5,879	9,774	10,034	5,514	9,094	7,239	4,341	6,970	1,139	3,030	1,585	2,399	66,998	
sao luis		4,093	6,190	5,475	6,692	9,160	4,606	5,734	4,530	3,867	5,894	5,123	3,134	64,498	
joao pessoa		2,278	3,928	5,726	5,106	6,284	12,019	14,701	3,395	2,440	3,363	1,138	2,977	63,355	
vitoria		4,454	4,514	4,213	5,029	8,557	7,835	11,442	4,473	2,558	974	6,367	2,467	62,883	
contagem		7,404	5,163	6,447	6,235	6,602	5,960	2,949	6,726	1,841	3,923	3,054	2,645	58,949	
uberlandia		3,984	5,906	3,692	5,084	7,463	5,760	4,221	6,432	2,437	4,273	3,375	4,475	57,102	
vila velha		4,688	5,064	3,367	3,931	4,848	4,183	11,911	3,918	1,704	3,600	5,763	3,466	56,443	
maceio		4,618	5,326	7,590	8,544	5,637	2,275	4,618	4,499	1,690	3,984	3,287	1,104	53,172	

## INSIGHTS:

### ➤ Top-3 state and city combination

1. MG and belo horizonte
2. DF and brasilia
3. BA and salvador

## 2. How are customers distributed in Brazil

```
with CustNo as (  
select customer_state, customer_city, Count_of_customer_city  
      ,sum(Count_of_customer_city) over (partition by customer_state) as Count_of_customer_state  
from (  
SELECT customer_state, customer_city, count(distinct customer_unique_id) as Count_of_customer_  
city  
  
FROM `target-sql-358310.TARGET_CASE_STUDY.customers`  
group by customer_state, customer_city )A  
)
```

```
select customer_state, customer_city, Count_of_customer_city, Count_of_customer_state  
      ,round(Count_of_customer_city/Count_of_customer_state*100,1) as Perc_of_Cust  
from CustNo  
order by round(Count_of_customer_city/Count_of_customer_state*100,1) desc
```

Query results

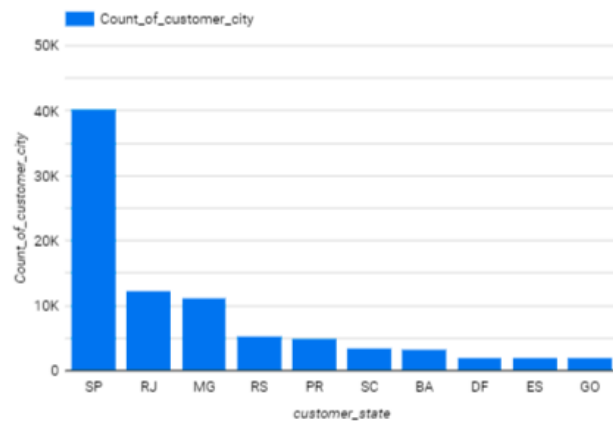
[📄 SAVE RESULTS](#)

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS		
Row	customer_state	customer_city	Count_of_customer_city	Count_of_customer_state	Perc_of_Cust	
19	PE	recife	590	1609	36.7	
20	GO	goiania	671	1953	34.4	
21	TO	palmas	90	273	33.0	
22	PR	curitiba	1465	4887	30.0	
23	MT	cuiaba	242	876	27.6	
24	RS	porto alegre	1326	5280	25.1	
25	MG	belo horizonte	2672	11269	23.7	
26	ES	vitoria	366	1965	18.6	
27	ES	vila velha	325	1965	16.5	
28	SC	florianopolis	546	3536	15.4	
29	TO	araguaina	40	273	14.7	

## customers State wise

	customer_state	Count_of_customer_city
1.	SP	40,345
2.	RJ	12,396
3.	MG	11,269
4.	RS	5,280
5.	PR	4,887
6.	SC	3,536
7.	BA	3,279
8.	DF	2,076
9.	ES	1,965
10.	GO	1,953
11.	PE	1,609

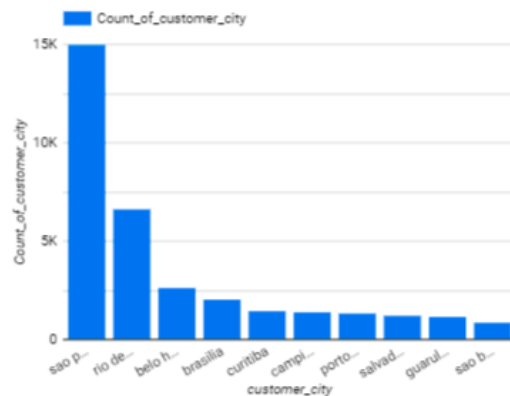
1 - 27 / 27 < >



## customers city wise

	customer_city	Count_of_customer_city
1.	sao paulo	14,984
2.	rio de janeiro	6,620
3.	belo horizonte	2,672
4.	brasilia	2,069
5.	curitiba	1,465
6.	campinas	1,398
7.	porto alegre	1,326
8.	salvador	1,209
9.	guarulhos	1,153
10.	sao bernardo do campo	908
11.	niteroi	811

1 - 50 / 4119 < >



### QUESTION – 4: Impact on Economy:

Analyze the money movemented 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)**

```

with data_base as
(SELECT --c.customer_state,
      extract (Year from order_purchase_timestamp) as Year
      ,extract (Month from order_purchase_timestamp) as Month
      ,b.freight_value
      ,d.payment_value
      ,a.order_id
FROM `target-sql-358310.TARGET_CASE_STUDY.orders` a
left join `target-sql-358310.TARGET_CASE_STUDY.order_items` b on a.order_id = b.order_id
left join `target-sql-
358310.TARGET_CASE_STUDY.customers` c on a.customer_id = c.customer_id
left join `target-sql-358310.TARGET_CASE_STUDY.payments` d on a.order_id = d.order_id )

select Year
      ,count(order_id) as sum_order_id
      ,round(sum(freight_value),1) as sum_of_fright_value
      ,round(round(sum(freight_value),1)/count(order_id),1) as avg_fright_value
      ,round(sum(payment_value),1) as sum_of_payment_value
      ,round((sum(freight_value)/sum(payment_value) ) * 100,1) as freight_value_per
from data_base
where data_base.Year in (2017,2018) and Month in (1,2,3,4,5,6,7,8)
group by Year

```

## Query results

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS		
Row	Year	sum_order_id	sum_of_frig...	avg_fright_v...	sum_of_pay...	freight_valu...
1	2018	63904	1308920.9	20.5	11122556.9	11.8
2	2017	27594	528661.3	19.2	4554984.7	11.6

- **INCREASE IN FRIGHT VALUE FROM 2017 TO 2018 BY 0.2 %**

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

```

with data_base as
(SELECT c.customer_state
      ,b.freight_value
      ,d.payment_value
      ,a.order_id
FROM `target-sql-358310.TARGET_CASE_STUDY.orders` a
left join `target-sql-358310.TARGET_CASE_STUDY.order_items` b on a.order_id = b.order_id
left join `target-sql-
358310.TARGET_CASE_STUDY.customers` c on a.customer_id = c.customer_id
left join `target-sql-358310.TARGET_CASE_STUDY.payments` d on a.order_id = d.order_id )

select customer_state
      ,round(sum(freight_value),1) as sum_of_fright_value

```

```

,round(avg(freight_value),1) as Avg_of_fright_value
,round(sum(payment_value),1) as sum_of_payment_value
,round(avg(payment_value),1) as Avg_of_payment_value
,round((sum(freight_value)/sum(payment_value) ) * 100,1) as freight_per_payment
from data_base
group by customer_state
order by sum(freight_value) desc

```

Query results								<a href="#">SAVE RESULTS</a>	
JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS					
Row	customer_state	sum_of_frig...	Avg_of_frig...	sum_of_pay...	Avg_of_pay...	freight_per_...			
1	SP	753359.7	15.2	7673188.5	153.6	9.8			
2	RJ	323413.9	21.1	2783724.3	180.5	11.6			
3	MG	281301.3	20.6	2341861.5	170.5	12.0			
4	RS	141579.7	21.8	1152019.2	176.7	12.3			
5	PR	122669.7	20.6	1074614.2	178.6	11.4			
6	BA	106538.6	26.3	802416.7	197.1	13.3			
7	SC	92216.4	21.4	799135.9	184.6	11.5			
8	PE	61923.6	32.8	378887.5	200.0	16.3			
9	GO	55237.5	22.7	516182.5	211.3	10.7			
10	DF	52118.8	21.1	434512.6	174.6	12.0			

Results per page: 50 ▼

- SP STATE HAS 15.2% OF AVG FRIGHT PRICE PER ORDER AND AVG PAYMENT VALUE IS \$ 153.6
- SP STATE HAS MINIMUM FRIGHT PER ORDER CONTRIBUTION 9.8% (SEE BELOW SCREENSHOT)

Query results								<a href="#">SAVE RESULTS</a> ▼ <a href="#">EXPLORE DATA</a> ▼	
JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS					
Row	customer_state	sum_of_fright_value	Avg_of_fright_value	sum_of_payment_value	Avg_of_payment_value	freight_per_payment			
1	SP	753359.7	15.2	7673188.5	153.6	9.8			
2	GO	55237.5	22.7	516182.5	211.3	10.7			
3	PR	122669.7	20.6	1074614.2	178.6	11.4			
4	SC	92216.4	21.4	799135.9	184.6	11.5			
5	RJ	323413.9	21.1	2783724.3	180.5	11.6			
6	MS	19739.4	23.4	165861.5	195.4	11.9			
7	DF	52118.8	21.1	434512.6	174.6	12.0			
8	MG	281301.3	20.6	2341861.5	170.5	12.0			
9	RS	141579.7	21.8	1152019.2	176.7	12.3			

- TO STATE HAS MORE FRIGHT PER PAYMENT WITH 18.6%

## Query results

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

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS		
Row	customer_state	sum_of_fright_value	Avg_of_fright_value	sum_of_payment_value	Avg_of_payment_value	freight_per_payment
1	TO	13450.6	39.7	72412.1	213.0	18.6
2	RR	2235.2	43.0	12462.2	239.7	17.9
3	RN	20074.3	35.3	117074.4	204.7	17.1
4	RO	11717.5	41.0	69184.5	236.9	16.9
5	PI	22480.6	39.2	137077.7	238.4	16.4
6	PE	61923.6	32.8	378887.5	200.0	16.3
7	MA	32290.3	38.3	199913.6	234.9	16.2
8	AM	5656.5	33.1	34873.8	202.8	16.2
9	SE	14541.3	36.6	90651.4	224.9	16.0
10	PA	30881.1	35.7	262433.1	233.0	15.7

**- FRIGHT WILL CONTRIBUTE AROUND 14.2% OF OVERALL PAY VALUE.**

with data\_base as

(SELECT c.customer\_state

,b.freight\_value

,d.payment\_value

,a.order\_id

FROM `target-sql-358310.TARGET\_CASE\_STUDY.orders` a

left join `target-sql-358310.TARGET\_CASE\_STUDY.order\_items` b on a.order\_id = b.order\_id

left join `target-sql-

358310.TARGET\_CASE\_STUDY.customers` c on a.customer\_id = c.customer\_id

left join `target-sql-358310.TARGET\_CASE\_STUDY.payments` d on a.order\_id = d.order\_id )

SELECT AVG(freight\_per\_payment) CONTRIBUSTION\_OF\_PRIGHT\_IN\_OVERALL\_PAYMENT

FROM (

select customer\_state

,round(sum(freight\_value),1) as sum\_of\_fright\_value

,round(avg(freight\_value),1) as Avg\_of\_fright\_value

,round(sum(payment\_value),1) as sum\_of\_payment\_value

,round(avg(payment\_value),1) as Avg\_of\_payment\_value

,round((sum(freight\_value)/sum(payment\_value) ) \* 100,1) as freight\_per\_payment

from data\_base

group by customer\_state)

--order by round((sum(freight\_value)/sum(payment\_value) ) \* 100,1)

## Query results

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAIL
Row	//	CONTIBUSTION_OF_PRIGHT_IN_OVERALL_PAYMENT	//	
1			14.12222222222222	

## QUESTION – 5: Analysis on sales, freight and delivery time

1. Calculate days between purchasing, delivering and estimated delivery
2. Create columns:
  - $\text{time\_to\_delivery} = \text{order\_purchase\_timestamp} - \text{order\_delivered\_customer\_date}$
  - $\text{diff\_estimated\_delivery} = \text{order\_estimated\_delivery\_date} - \text{order\_delivered\_customer\_date}$
3. Group data by state, take mean of freight\_value, time\_to\_delivery, diff\_estimated\_delivery

```
with dates_calculation as
(
  SELECT c.customer_state
    ,order_purchase_timestamp as Prurchase_date
    ,order_estimated_delivery_date as Estimated_delv_date
    ,order_delivered_customer_date as Actual_delv_date
    ,b.freight_value
    ,d.payment_value
    ,DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp, day) + 1 as diff_estim
    ated_delivery
    ,Case when order_delivered_customer_date is null then DATE_DIFF(order_estimated_delivery_d
    ate, order_purchase_timestamp, day) +1
```

```

        else DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day) +1 END as time_to_delivery

FROM `target-sql-358310.TARGET_CASE_STUDY.orders` a

left join `target-sql-358310.TARGET_CASE_STUDY.order_items` b on a.order_id = b.order_id

left join `target-sql-358310.TARGET_CASE_STUDY.customers` c on a.customer_id = c.customer_id

left join `target-sql-358310.TARGET_CASE_STUDY.payments` d on a.order_id = d.order_id )

select customer_state

,round(avg(diff_estimated_delivery),1) Mean_diff_estimated_delivery

,round(avg(time_to_delivery),1) Mean_time_to_delivery

,round(avg(freight_value),1) Avg_freight_value

,round(avg(payment_value),1) Avg_payment_value

,round(avg(freight_value)/avg(payment_value) * 100,1) Perc_Freight_of_payment

from dates_calculation

group by customer_state

```

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS			
row	customer_state	Mean_diff_estimated_delivery	Mean_time_to_delivery	Avg_freight_value	Avg_payment_value	Perc_Freight_of_payment	
1	RJ	27.2	16.2	21.1	180.5	11.7	
2	RS	29.4	16.0	21.8	176.7	12.4	
3	SP	19.9	9.6	15.2	153.6	9.9	
4	DF	25.2	13.9	21.1	174.6	12.1	
5	PR	25.4	12.8	20.6	178.6	11.5	
6	MT	32.3	19.1	29.0	228.0	12.7	
7	MA	31.5	22.6	38.3	234.9	16.3	
8	AL	33.2	25.5	35.7	243.5	14.7	
9	MG	25.3	12.8	20.6	170.5	12.1	
10	PE	31.9	19.4	32.8	200.0	16.4	
11	SE	31.2	22.1	36.6	224.9	16.3	

## TOP AND BOTTOM 5 ANALYSIS: 1. freight value

4. Sort the data to get the following:

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

Top 5 states with highest average freight value

with dates\_calculation as



```

(SELECT c.customer_state
,order_purchase_timestamp as Prurchase_date
,order_estimated_delivery_date as Estimated_delv_date
,order_delivered_customer_date as Actual_delv_date
,b.freight_value
,d.payment_value
,DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp, day) + 1 as diff_estim
ated_delivery
,Case when order_delivered_customer_date is null then DATE_DIFF(order_estimated_delivery_d
ate, order_purchase_timestamp, day) +1
else DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day) +1 END as t
ime_to_delivery
FROM `target-sql-358310.TARGET_CASE_STUDY.orders` a
left join `target-sql-358310.TARGET_CASE_STUDY.order_items` b on a.order_id = b.order_id
left join `target-sql-
358310.TARGET_CASE_STUDY.customers` c on a.customer_id = c.customer_id
left join `target-sql-358310.TARGET_CASE_STUDY.payments` d on a.order_id = d.order_id )

select *
from (
select customer_state
,round(avg(diff_estimated_delivery),1) Mean_diff_estimated_delivery
,round(avg(time_to_delivery),1) Mean_time_to_delivery
,round(avg(freight_value),1) Avg_freight_value
,round(avg(payment_value),1) Avg_payment_value
,round(avg(freight_value)/avg(payment_value) * 100,1) Perc_Freight_of_payment
from dates_calculation
group by customer_state )BASE
order by Avg_freight_value desc
Limit 5

```

## Query results

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS			
Row	customer_state	Mean_diff_e...	Mean_time...	Avg_freight...	Avg_payme...	Perc_Freigh...	
1	SP	19.9	9.6	15.2	153.6	9.9	
2	PR	25.4	12.8	20.6	178.6	11.5	
3	MG	25.3	12.8	20.6	170.5	12.1	
4	RJ	27.2	16.2	21.1	180.5	11.7	
5	DF	25.2	13.9	21.1	174.6	12.1	

## TOP AND BOTTOM 5 ANALYSIS: 2. average time to delivery

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

```
with dates_calculation as
(
SELECT c.customer_state
,order_purchase_timestamp as Prurchase_date
,order_estimated_delivery_date as Estimated_delv_date
,order_delivered_customer_date as Actual_delv_date
,b.freight_value
,d.payment_value
,DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp, day) + 1 as diff_estim
ated_delivery
,Case when order_delivered_customer_date is null then DATE_DIFF(order_estimated_delivery_d
ate, order_purchase_timestamp, day) +1
else DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day) +1 END as t
ime_to_delivery
FROM `target-sql-358310.TARGET_CASE_STUDY.orders` a
left join `target-sql-358310.TARGET_CASE_STUDY.order_items` b on a.order_id = b.order_id
left join `target-sql-
358310.TARGET_CASE_STUDY.customers` c on a.customer_id = c.customer_id
left join `target-sql-358310.TARGET_CASE_STUDY.payments` d on a.order_id = d.order_id )

select *
from (
select customer_state
,round(avg(diff_estimated_delivery),1) Mean_diff_estimated_delivery
,round(avg(time_to_delivery),1) Mean_time_to_delivery
,round(avg(freight_value),1) Avg_freight_value
,round(avg(payment_value),1) Avg_payment_value
,round(avg(freight_value)/avg(payment_value) * 100,1) Perc_Freight_of_payment
from dates_calculation
group by customer_state )BASE
order by Mean_time_to_delivery desc
Limit 5
```

## Top 5 states with highest average time to delivery

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

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS			
Row	customer_state	Mean_diff_estimated_delivery	Mean_time_to_delivery	Avg_freight_value	Avg_payment_value	Perc_Freight_of_payment	
1	RR	47.0	31.3	43.0	239.7	17.9	
2	AP	46.7	28.8	34.1	257.7	13.2	
3	AM	46.2	27.3	33.1	202.8	16.3	
4	AL	33.2	25.5	35.7	243.5	14.7	
5	PA	37.9	24.6	35.7	233.9	15.3	

## Top 5 states with lowest average time to delivery

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS			
Row	customer_state	Mean_diff_estimated_delivery	Mean_time_to_delivery	Avg_freight_value	Avg_payment_value	Perc_Freight_of_payment	
1	SP	19.9	9.6	15.2	153.6	9.9	
2	MG	25.3	12.8	20.6	170.5	12.1	
3	PR	25.4	12.8	20.6	178.6	11.5	
4	DF	25.2	13.9	21.1	174.6	12.1	
5	SC	26.5	15.8	21.4	184.6	11.6	

## TOP AND BOTTOM 5 ANALYSIS: 3. fast/ not so fast compared to estimated date

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

```

with dates_calculation as
(
SELECT c.customer_state
,order_purchase_timestamp as Prurchase_date
,order_estimated_delivery_date as Estimated_delv_date
,order_delivered_customer_date as Actual_delv_date
,b.freight_value
,d.payment_value
,DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp, day) + 1 as diff_estim
ated_delivery
,Case when order_delivered_customer_date is null then DATE_DIFF(order_estimated_delivery_d
ate, order_purchase_timestamp, day) +1
else DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, day) +1 END as t
ime_to_delivery
FROM `target-sql-358310.TARGET_CASE_STUDY.orders` a
left join `target-sql-358310.TARGET_CASE_STUDY.order_items` b on a.order_id = b.order_id
left join `target-sql-
358310.TARGET_CASE_STUDY.customers` c on a.customer_id = c.customer_id
left join `target-sql-358310.TARGET_CASE_STUDY.payments` d on a.order_id = d.order_id )

select *, ROUND((Mean_diff_estimated_delivery - Mean_time_to_delivery),1) AS FST_OR_SLW_DLRY
from (
select customer_state
,round(avg(diff_estimated_delivery),1) Mean_diff_estimated_delivery
,round(avg(time_to_delivery),1) Mean_time_to_delivery
,round(avg(freight_value),1) Avg_freight_value
,round(avg(payment_value),1) Avg_payment_value
,round(avg(freight_value)/avg(payment_value) * 100,1) Perc_Freight_of_payment
from dates_calculation
group by customer_state )BASE
order by ROUND((Mean_diff_estimated_delivery - Mean_time_to_delivery),1) desc
Limit 5

```

Fast delivery compared to estimated date

Query results

SAVE RESULTS

EXPLORE DATA

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS				
Row	customer_state	Mean_diff_estimated_delivery	Mean_time_to_delivery	Avg_freight_value	Avg_payment_value	Perc_Freight_of_payment	FST_OR_SLW_DLRY	
1	AL	33.2	25.5	35.7	243.5	14.7	7.7	
2	MA	31.5	22.6	38.3	234.9	16.3	8.9	
3	SE	31.2	22.1	36.6	224.9	16.3	9.1	
4	ES	26.3	16.4	22.0	173.4	12.7	9.9	
5	BA	30.1	20.0	26.3	197.1	13.4	10.1	

Slow delivery compared to estimated date

Query results

SAVE RESULTS

EXPLORE DATA

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS				
Row	customer_state	Mean_diff_estimated_delivery	Mean_time_to_delivery	Avg_freight_value	Avg_payment_value	Perc_Freight_of_payment	FST_OR_SLW_DLRY	
1	AC	41.7	21.4	40.0	263.0	15.2	20.3	
2	AM	46.2	27.3	33.1	202.8	16.3	18.9	
3	RO	39.6	21.0	41.0	236.9	17.3	18.6	
4	AP	46.7	28.8	34.1	257.7	13.2	17.9	
5	RR	47.0	31.3	43.0	239.7	17.9	15.7	

OVERALL SUMMARY

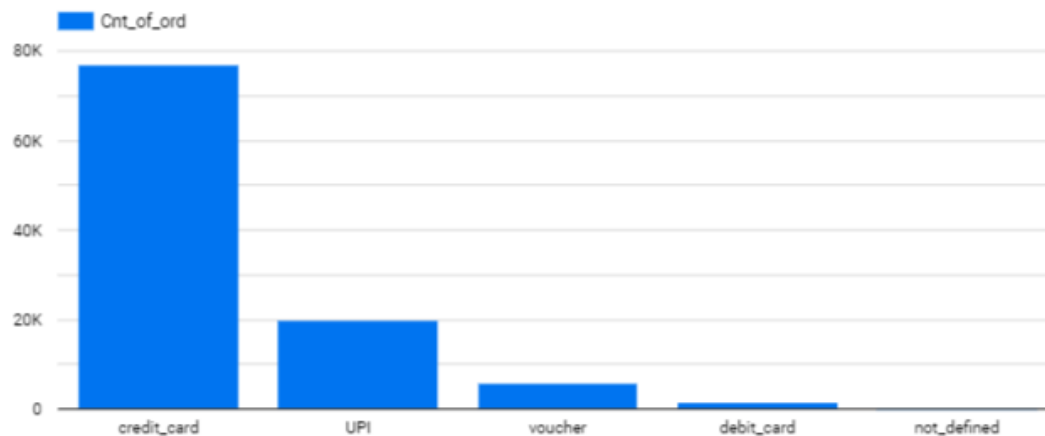
1. FRIGHT VALUE STAE			
LOWEST - SP	15.2	HIGHEST - RR	43
2.Delivery compared to estimated date			
FASTEST - AL	7.7 DAYS	SLOWEST - AC	20.3 DAYS
3. Average time to delivery			
Lowest - SP	9.6 DAYS	Highest – RR	31.2 DAYS

## QUESTION – 6: Payment type analysis

### Overall picture of payments

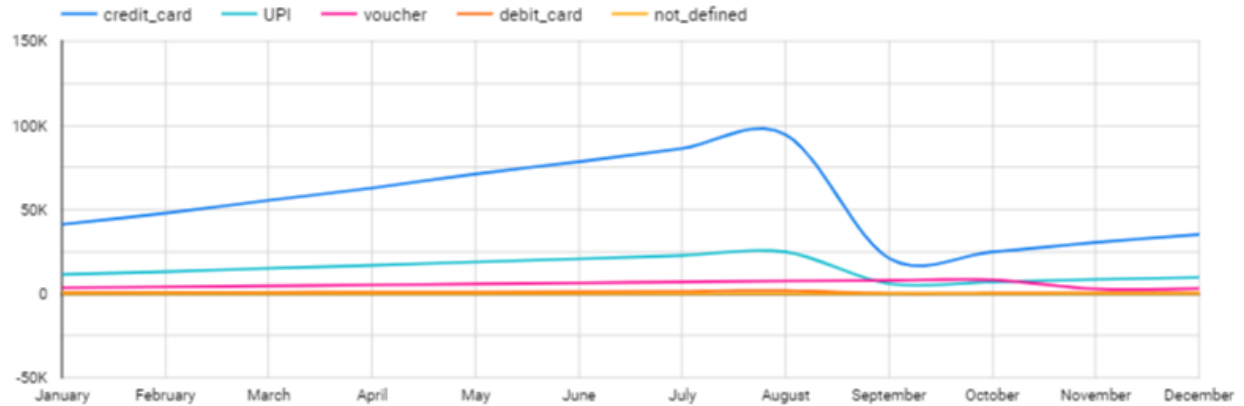
Query results			
JOB INFORMATION		RESULTS	JSON
EXECUTION DETAILS			
Row	payment_type	Count_of_or...	
1	credit_card	76795	
2	UPI	19784	
3	voucher	5775	
4	debit_card	1529	
5	not_defined	3	

### Mode of payments



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

# Mode of payments



## MOM of orders

# MoM count of orders

		payment_type / Sum_of_cnt				
Year	Month	credit_card	UPI	voucher	debit_card	not_defined
2016	September	3	-	-	-	-
	October	257	63	23	2	-
	December	258	-	-	-	-
2017	January	841	260	84	11	-
	February	2,197	658	203	24	-
	March	4,213	1,248	403	55	-
	April	6,059	1,744	605	82	-
	May	8,912	2,516	894	112	-
	June	11,375	3,223	1,133	139	-
	July	14,461	4,068	1,497	161	-
	August	17,745	5,006	1,791	195	-
	September	21,028	5,909	2,078	238	-
	October	24,552	6,902	2,369	290	-
	November	30,449	8,411	2,756	360	-
	December	34,826	9,571	3,050	424	-

Year	Month	credit_card	UPI	voucher	debit_card	not_defined
2018	December	34,828	9,071	3,000	424	-
	January	40,346	11,089	3,466	533	-
	February	45,599	12,414	3,771	602	-
	March	51,290	13,766	4,162	680	-
	April	56,745	15,053	4,532	777	-
	May	62,242	16,316	4,856	828	-
	June	67,055	17,416	5,180	1,010	-
	July	71,810	18,645	5,461	1,252	-
	August	76,795	19,784	5,756	1,529	2
	September	-	-	5,771	-	3
	October	-	-	5,775	-	-

## Month on Month % change

## MoM % change in orders

payment_type / Pr_mnt_cnt						
Year	Month	credit_card	debit_card	UPI	voucher	not_defined
2016	September	-	-	-	-	-
	October	8,366.7	-	-	-	-
	December	-99.6	-	-	-	-
2017	January	58,200	350	212.7	165.2	-
	February	132.6	44.4	102	95.1	-
	March	48.7	138.5	48.2	68.1	-
	April	-8.4	-12.9	-15.9	1	-
	May	54.6	11.1	55.6	43.1	-
	June	-13.7	-10	-8.4	-17.3	-
	July	25.3	-18.5	19.5	52.3	-
	August	6.4	54.5	11	-19.2	-
	September	0	26.5	-3.7	-2.4	-
	October	7.3	20.9	10	1.4	-
	November	67.3	34.6	52	33	-
	December	-25.8	-8.6	-23.1	-24	-

2018	January	26.1	70.3	30.9	41.5	-
	February	-4.8	-36.7	-12.7	-26.7	-
	March	8.3	13	2	28.2	-
	April	-4.1	24.4	-4.8	-5.4	-
	May	0.8	-47.4	-1.9	-12.4	-
	June	-12.4	256.9	-12.9	0	-
	July	-1.2	33	11.7	-13.3	-
	August	4.8	14.5	-7.3	5	-
	September	-	-	-	-94.9	-50
	October	-	-	-	-73.3	-

## 2. Distribution of payment installments and count of orders

```
SELECT a.payment_type, payment_installments, COUNT(a.order_id) AS cnt_of_ords
      ,sum(a.payment_value) AS payment_value
FROM `target-sql-358310.TARGET_CASE_STUDY.payments` a
group by a.payment_type, payment_installments
order by payment_installments desc, a.payment_type
```

Query results					
JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	
Row	payment_type	payment_in...	cnt_of_ords	payment_value	
20	credit_card	5	5239	961174.3	
21	credit_card	4	7098	1163907.6	
22	credit_card	3	10461	1491103.8	
23	credit_card	2	12413	1579283.0	
24	credit_card	1	25455	2440445.4	
25	credit_card	0	2	188.6	
26	debit_card	1	1529	217989.8	
27	not_defined	1	3	0.0	
28	voucher	1	5775	379436.9	



## DISTRIBUTION OF COUNT OF ORDERS OVER INSTALMENT

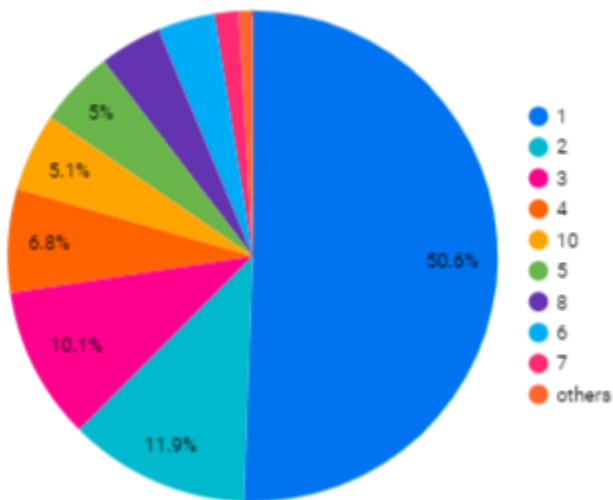
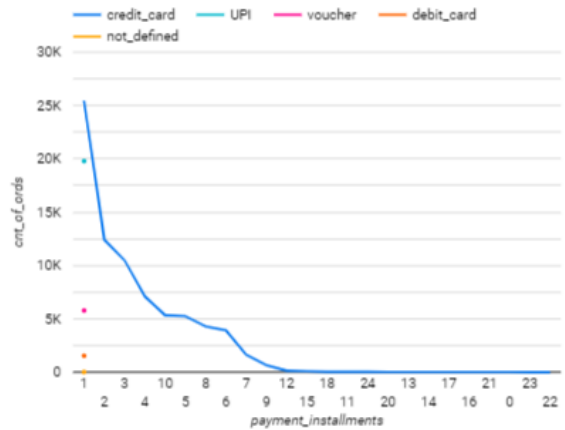
payment_inst...	payment_type / cnt_of_ords				
	credit_card	UPI	voucher	debit_card	not_defined
1	25,455	19,784	5,775	1,529	3
2	12,413	-	-	-	-
3	10,461	-	-	-	-
4	7,098	-	-	-	-
10	5,328	-	-	-	-
5	5,239	-	-	-	-
8	4,268	-	-	-	-
6	3,920	-	-	-	-
7	1,626	-	-	-	-
9	644	-	-	-	-
12	133	-	-	-	-
15	74	-	-	-	-
18	27	-	-	-	-
11	23	-	-	-	-
24	18	-	-	-	-
20	17	-	-	-	-

## DISTRIBUTION OF SUM OF PAYMENT OVER INSTALMENT

payment_inst...	payment_type / payment_value				
	credit_card	UPI	voucher	debit_card	not_defined
0	188.6	-	-	-	-
1	2,440,445.4	2,869,361.3	379,436.9	217,989.8	0
2	1,579,283	-	-	-	-
3	1,491,103.8	-	-	-	-
4	1,163,907.6	-	-	-	-
5	961,174.3	-	-	-	-
6	822,611.8	-	-	-	-
7	305,157.4	-	-	-	-
8	1,313,423.3	-	-	-	-
9	131,015.9	-	-	-	-
10	2,211,577.3	-	-	-	-
11	2,873.4	-	-	-	-
12	42,783.2	-	-	-	-
13	2,407.4	-	-	-	-
14	2,519.4	-	-	-	-
15	32,970.9	-	-	-	-

# payments

	payment_type	payment_installments
1.	credit_card	281
2.	debit_card	1
3.	voucher	1
4.	not_defined	1
5.	UPI	1



- ✓ credit card mode of payment is highest/preferred.
- ✓ Other mode of payments are UPI, voucher and debit card. All of these are just one shot payments.
- ✓ One more reason for opting credit card is EMI option available.
- ✓ Most preferred EMI options are upto 12.
- ✓ Max EMI options are 24 months.
- ✓ Out of all EMI received 1<sup>st</sup> and 10<sup>th</sup> contribute maximum amount.

## 7. Actionable Insights

- ✓ **DATE IS FROM SEP 2016 TO OCT 2018.**

- ✓ YOY SALES IS INCREASING, AND WE COULD SEE THE TREND AROUND MAY AND AUGUST MONTH IN BOTH 2017 AND 2018.
- ✓ More purchases are done in the evening – 13 to 18 hours of the day.
- ✓ Less purchases are done in the early morning – 0 to 6 hours of the day.
- ✓ TOP 3 STATES WITH SALES

1.MG and belo horizonte

2. DF and brasilia

3. BA and Salvador

- ✓ SP STATE HAS HIGHEST CUSTOMERS

- SP STATE HAS MINIMUM FRIGHT PER ORDER CONTRIBUSTION 9.8%
- TO STATE HAS MORE FRIGHT PER PAYMENT WITH 18.6%
- AVG FRIGHT WILL BE AROUND 14.2% OF OVERALL PAY VALUE

#### > OVERALL SUMMARY ON FRIGHT AND AVERAGE DELIVARY TIME

1. FRIGHT VALUE STate			
LOWEST - SP	15.2	HIGHEST - RR	43
2.Delivery compared to estimated date			
FASTEST - AL	7.7 DAYS	SLOWEST - AC	20.3 DAYS
3. Average time to delivery			
Lowest - SP	9.6 DAYS	Highest – RR	31.2 DAYS

- ✓ credit card mode of payment is highest/preferred.
- ✓ Other mode of payments are UPI, voucher, and debit card. All of these are just one-shot payments.
- ✓ One more reason for opting credit card is EMI option available.
- ✓ Most preferred EMI options are up to 12.
- ✓ Max EMI options are 24 months.
- ✓ Out of all EMI received 1st and 10th contribute maximum amount.

## 8. Recommendations

- ✓ There is increasing demand MOM based on count of order. And YOY sales is high. So aggressive pricing can help to gain more profit.
  - ✓ States and region to try and target aggressive pricing are TOP 3 STATES WITH SALES. 1.MG and belo horizonte, 2. DF and Brasilia and 3. BA and Salvador
- ✓ Average freight cost can be increased to get maximum profit in some state where contribution is less than 14%.
- ✓ EMI options more than 12 months are not can be stopped as they are in less demand. And helps company to get money back faster.
- ✓ More purchases are done in the evening – 13 to 18 hours of the day. So, make advertise and discounts available during this time so it will reach more customer.
- ✓ Less purchases are done in the early morning – 0 to 6 hours of the day. Reduce advertising cost during this time as customer interaction is less.
- ✓ Bottom 5 states with average time to delivery – target on thes state to decrease time gap.
  1. With above step customer satisfaction can be increased.
  2. New/more sales as delivery time will less.
  3. Don't increase freight value as its already higher than average.

Query results								<a href="#">SAVE RESULTS</a>	<a href="#">EXPLORE DATA</a>
JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS					
Row	customer_state	Mean_diff_estimated_delivery	Mean_time_to_delivery	Avg_freight_value	Avg_payment_value	Perc_Freight_of_payment			
1	RR	47.0	31.3	43.0	239.7	17.9			
2	AP	46.7	28.8	34.1	257.7	13.2			
3	AM	46.2	27.3	33.1	202.8	16.3			
4	AL	33.2	25.5	35.7	243.5	14.7			
5	PA	37.9	24.6	35.7	233.9	15.3			

- ✓ Delivering faster than expected in this customer states.
  1. Some resource can be diverted from this location to where we are doing very late.

Press Alt+F1 for Accessibility Options.

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

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS				
Row	customer_state	Mean_diff_estimated_delivery	Mean_time_to_delivery	Avg_freight_value	Avg_payment_value	Perc_Freight_of_payment	FST_OR_SLW_DLRY	
1	AL	33.2	25.5	35.7	243.5	14.7	7.7	
2	MA	31.5	22.6	38.3	234.9	16.3	8.9	
3	SE	31.2	22.1	36.6	224.9	16.3	9.1	
4	ES	26.3	16.4	22.0	173.4	12.7	9.9	
5	BA	30.1	20.0	26.3	197.1	13.4	10.1	

**2. These states are in bottom 5 where time to delivery is slow compared to estimated delivery. Focus should be made from above screen shot states to here.**

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

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS				
Row	customer_state	Mean_diff_estimated_delivery	Mean_time_to_delivery	Avg_freight_value	Avg_payment_value	Perc_Freight_of_payment	FST_OR_SLW_DLRY	
1	AC	41.7	21.4	40.0	263.0	15.2	20.3	
2	AM	46.2	27.3	33.1	202.8	16.3	18.9	
3	RO	39.6	21.0	41.0	236.9	17.3	18.6	
4	AP	46.7	28.8	34.1	257.7	13.2	17.9	
5	RR	47.0	31.3	43.0	239.7	17.9	15.7	