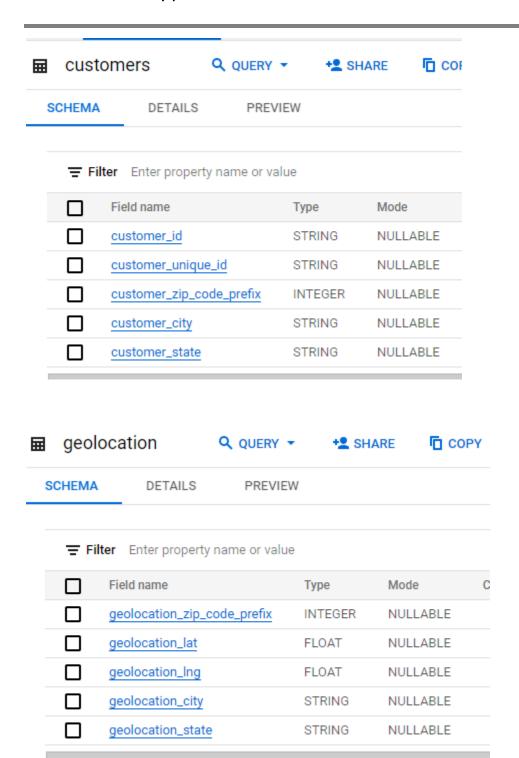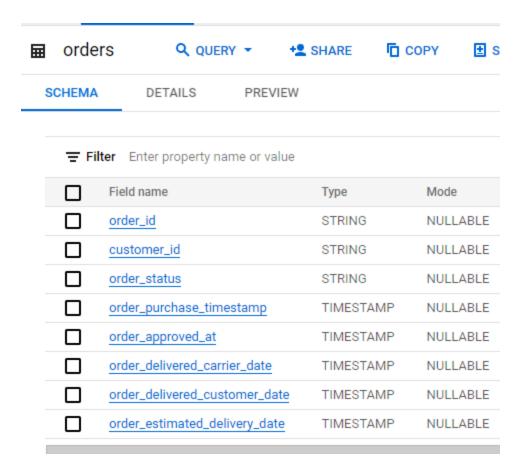# 1. Datatypes of Columns in all tables
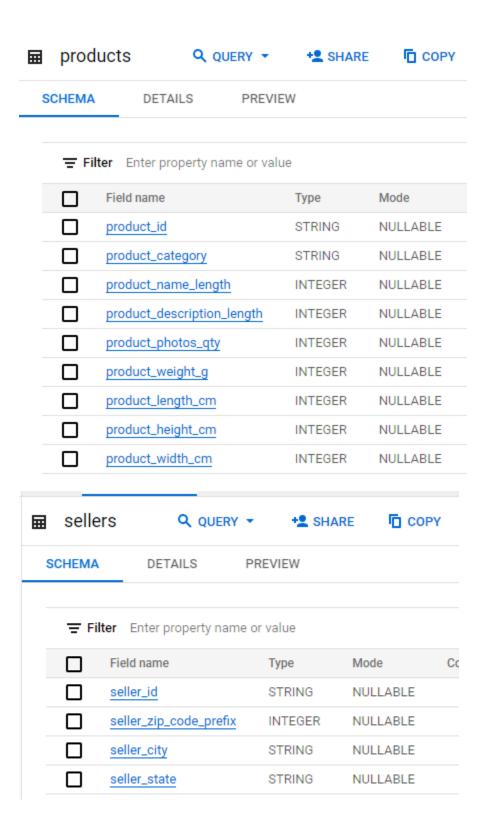
customers

Q QUERY ▾ +⚫ SHARE ⎘ CO[...]

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

Q QUERY ▾ +⚫ SHARE ⎘ COPY

SCHEMA   DETAILS   PREVIEW

≡ Filter   Enter property name or value

Field name	Type	Mode	C
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 COPY

SCHEMA   DETAILS   PREVIEW

≡ Filter   Enter property name or value

	Field name	Type	Mode	Colla
☐	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 COPY

SCHEMA   DETAILS   PREVIEW

≡ Filter   Enter property name or value

	Field name	Type	Mode	C
☐	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

Q QUERY ▾   +👤 SHARE   📋 COPY   ⊞ S

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

Q 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   📋 COPY

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   📋 COPY

SCHEMA   DETAILS   PREVIEW

≡ Filter   Enter property name or value

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

2.   Time Period for which Data is given

```
1  SELECT min(o.order_purchase_timestamp) as Start_Date
2  , max(o.order_purchase_timestamp) as End_Date
3  FROM `target-365505.Target.orders` o
```

Row	Start_Date	End_Date
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC

Around 2 years of Orders Data is provided from 04-Sep-2016 till 17-Oct-2018.

3.   Cities & States covered in the dataset

3.1 Cities and States covered in CUSTOMER ORDER dataset

```
SELECT distinct cus.customer_city
FROM `target-365505.Target.customers` cus

SELECT distinct cus.customer_state
FROM `target-365505.Target.customers` cus
```

JOB INFORMATION	RESULTS	JSON	EXECUTION DETAILS

Row	customer_city
1	acu
2	ico
3	ipe
4	ipu
5	ita

Results per page: 50   1 – 50 of 4119

Row	customer_state	
1	RN	
2	CE	
3	RS	
4	SC	
5	SP	

Results per page: 50 ▼   1 – 27 of 27

In these 2 years of Orders data – orders have been received from 4110 different cities of Brazil which are from 27 different states.

2. Cities and States covered in SELLER ORDER dataset

```
SELECT distinct s.seller_city
FROM `target-365505.Target.sellers` s

SELECT distinct s.seller_state
FROM `target-365505.Target.sellers` s
```

JOB INFORMATION	RESULTS	JSON	EXECUTION DETAILS

Row	seller_city	
1	rio branco	
2	manaus	
3	bahia	
4	ipira	
5	irece	

Results per page: 50 ▼   1 – 50 of 611

Row	seller_state	
1	AC	
2	AM	
3	BA	
4	CE	
5	DF	

Results per page: 50 ▼   1 – 23 of 23

In these 2 years of Orders data – orders have been received by sellers of 611 different cities of Brazil which are from 23 different states.

4. In-depth Exploration

- Average Order stats year by Year

```
SELECT X.Year_of_Purchase, round(X.ORDER_COUNT/X.Number_of_Months,0) as AVG_ORDER_COU
FROM
(
    SELECT
                Extract(Year from o.order_purchase_timestamp) as Year_of_Purchase,
                DATE_DIFF
                (
                    max(CAST(o.order_purchase_timestamp AS DATE)),
                    min(CAST(o.order_purchase_timestamp AS DATE)),
                    month
                ) as Number_of_Months,
                count(distinct o.order_id) as ORDER_COUNT
    FROM `target-365505.Target.orders` o
    group by Year_of_Purchase
    order by Year_of_Purchase
) AS X
```

Row	Year_of_Pur...	AVG_ORDER...
1	2016	110.0
2	2017	4100.0
3	2018	6001.0

- Total Order Counts month by month for each year

```
SELECT
        Extract(Year from o.order_purchase_timestamp) as Year_of_Purchase,
        Extract(Month from o.order_purchase_timestamp) as Month_of_Purchase,
        count(distinct o.order_id) as ORDER_COUNT
FROM `target-365505.Target.orders` o
group by Year_of_Purchase, Month_of_Purchase
order by Year_of_Purchase, Month_of_Purchase
```

Row	Year_of_Pur...	Month_of_P...	ORDER_CO...
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

Row	Year_of_Pur...	Month_of_P...	ORDER_CO...
12	2017	9	4285
13	2017	10	4631
14	2017	11	7544
15	2017	12	5673
16	2018	1	7269
17	2018	2	6728
18	2018	3	7211
19	2018	4	6939
20	2018	5	6873
21	2018	6	6167
22	2018	7	6292
23	2018	8	6512
24	2018	9	16
25	2018	10	4

- What time do Brazilian customers tend to buy

```
Select Time, concat(round(count(*)/(sum(count(*)) over ())*100,2),"%") as Frequency_of_Purchase_in_percentage
FROM
(
    SELECT
        Extract(Year from o.order_purchase_timestamp) as Year_of_Purchase,
        CASE
        when Extract(Time from o.order_purchase_timestamp) between "05:00:00" and "06:00:00" Then "Dawn (05:00:00 - 06:00:00)"
        when Extract(Time from o.order_purchase_timestamp) between "06:00:00" and "12:00:00" Then "Morning (06:00:00 - 12:00:00)"
        when Extract(Time from o.order_purchase_timestamp) between "12:00:00" and "16:00:00" Then "AfterNoon(12:00:00 - 16:00:00)"
        when Extract(Time from o.order_purchase_timestamp) between "16:00:00" and "19:00:00" Then "Evening(16:00:00 - 19:00:00)"
        else "Night(19:00:00 onwards till 5:00:00)"
        END as Time
    FROM `target-365505.Target.orders` o
    order by Year_of_Purchase
) as X
Group by Time
order by Frequency_of_Purchase_in_percentage
```

Row	Time	Frequency_of_Purchase_in_per...
1	Dawn (05:00:00 - 06:00:00)	0.19%
2	Evening(16:00:00 - 19:00:00)	18.7%
3	Morning (06:00:00 - 12:00:00)	22.37%
4	AfterNoon(12:00:00 - 16:00:00)	25.68%
5	Night(19:00:00 onwards till 5:0...	33.07%

5. Evolution of E-commerce orders in the Brazil region

Month on month orders by region, states

```
SELECT X.customer_state, X.customer_city,Year, Month,
ORDER_CT,
sum(ORDER_CT) over (partition by X.customer_state, X.customer_city order by X.customer_state, X.customer_city,Year, Month) as
Running_Order_Ct
FROM
    (
    SELECT
        c.customer_state, c.customer_city,Extract(Year from o.order_purchase_timestamp ) as Year,
        Extract(Month from o.order_purchase_timestamp ) as Month,
        count(o.order_purchase_timestamp) as ORDER_CT
        FROM `target-365505.Target.orders` o
    left join `target-365505.Target.customers` c
    on o.customer_id = c.customer_id
    group by c.customer_state,c.customer_city,Year, Month
    order by c.customer_state,c.customer_city,Year, Month
    ) as X
order by X.customer_state, X.customer_city,Year, Month
```

Row	customer_state	customer_city	Year	Month	ORDER_CT	Running_Or...
1	AC	brasileia	2017	2	1	1
2	AC	cruzeiro do sul	2017	12	2	2
3	AC	cruzeiro do sul	2018	5	1	3
4	AC	epitaciolandia	2017	10	1	1
5	AC	manoel urbano	2017	9	1	1
6	AC	porto acre	2017	4	1	1
7	AC	rio branco	2017	1	2	2
8	AC	rio branco	2017	2	2	4
9	AC	rio branco	2017	3	2	6
10	AC	rio branco	2017	4	4	10

6. Customers Distribution in Brazil (State Wise)

```
SELECT
c.customer_state, count(distinct c.customer_id) as Customer_Count
FROM `target-365505.Target.customers` c
group by c.customer_state
order by Customer_Count desc
```

Row	customer_state	Customer_C...
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

7. Order Distribution in Brazil (State Wise)

```
1  SELECT customer_state, concat(round(Customer_Count*100/Total,2),"%") as Order_count
2  FROM
3  (
4    SELECT
5    c.customer_state, count(distinct c.customer_id) as Customer_Count,
6    sum(count(distinct c.customer_id)) over() as Total
7    FROM `target-365505.Target.customers` c
8    group by c.customer_state
9    order by Customer_Count desc
10 )
```

Row	customer_state	Order_count
1	SP	41.98%
2	RJ	12.92%
3	MG	11.7%
4	RS	5.5%
5	PR	5.07%
6	SC	3.66%

Row	customer_state	Order_count
1	RR	0.05%
2	AP	0.07%
3	AC	0.08%
4	AM	0.15%
5	RO	0.25%
6	TO	0.28%
7	SE	0.25%

8. Impact on Economy

- % Increase in cost of orders from 2017 to 2018

```
SELECT
        Extract(Year from o.order_purchase_timestamp) as Year,
        -- sum(oi.price) as Total_Price,
        round(sum(oi.freight_value),2) as Total_Freight_Value

FROM `target-365505.Target.orders` o
left join `target-365505.Target.order_items` oi on o.order_id = oi.order_id
where Extract(Month from o.order_purchase_timestamp) <9
group by Year
order by Year
```

Row	Year	Total_Freight_Value
1	2017	497269.83
2	2018	1257625.34

- Mean & Sum of price and freight value by customer state

```
SELECT
        c.customer_state ,
        round(avg(oi.price),2) as Avg_Price_Value,
        round(sum(oi.price),2) as Total_Price_Value,
        round(avg(oi.freight_value),2) as Avg_Freight_Value,
        round(sum(oi.freight_value),2) as Total_Freight_Value
FROM `target-365505.Target.customers` c
left join `target-365505.Target.orders` o on c.customer_id = o.customer_id
left join `target-365505.Target.order_items` oi on o.order_id = oi.order_id
group by c.customer_state
order by c.customer_state
```

JOB INFORMATION	RESULTS	JSON	EXECUTION DETAILS		
Row	customer_state	Avg_Price_V…	Total_Price_…	Avg_Freight…	Total_Freigh…
1	AC	173.73	15982.95	40.07	3686.75
2	AL	180.89	80314.81	35.84	15914.59
3	AM	135.5	22356.84	33.21	5478.89
4	AP	164.32	13474.3	34.01	2788.5
5	BA	134.6	511349.99	26.36	100156.68

Results per page: 50 ▾

9. Analysis on sales, freight and delivery time

- Days between purchasing, delivering and estimated delivery

```sql
SELECT DELIVERY_HEALTH,
       concat(round(count(*)*100/(sum(count(*)) over()),2),"%") as Delivery_Percentage
FROM
(
       SELECT order_id, order_status, Delay,
              CASE
              WHEN Delay =0 THEN "ON TIME DELIVERY"
              WHEN Delay >0 THEN "EARLY DELIVERY"
              WHEN Delay <0 THEN "LATE DELIVERY"
              END AS DELIVERY_HEALTH
       From
              (
              SELECT
                     o.order_id,
                     o.order_status,
                     date_diff(o.order_estimated_delivery_date ,o.order_purchase_timestamp,day) Estimated_Delivery_Days,
                     date_diff(o.order_delivered_customer_date,o.order_purchase_timestamp,day) Actual_Delivery_Days,
                     date_diff(o.order_estimated_delivery_date ,o.order_purchase_timestamp,day) -
                     date_diff(o.order_delivered_customer_date,o.order_purchase_timestamp,day) as Delay

              FROM `target-365505.Target.orders` o
              where date_diff(o.order_delivered_customer_date,o.order_purchase_timestamp,day) is not null
              order by Delay desc
              )
)
group by Delivery_Health
```

Row	DELIVERY_HEALTH	Delivery_Percentage
1	EARLY DELIVERY	90.99%
2	LATE DELIVERY	7.57%
3	ON TIME DELIVERY	1.43%

```sql
SELECT concat(round(count(*)*100/96476,0),"%") as Avg_GT_10_days FROM
(
       SELECT distinct
              o.order_id,
              avg(date_diff(o.order_estimated_delivery_date,o.order_purchase_timestamp,day))
              Avg_Estimated_Delivery_Days,
              count(*) over() as Total
       FROM `target-365505.Target.orders` o
       where date_diff(o.order_delivered_customer_date,o.order_purchase_timestamp,day) is not null
       group by order_id
       having Avg_Estimated_Delivery_Days > 10
       order by Avg_Estimated_Delivery_Days desc
)
group by Total
```

- State mean freight_value, time_to_delivery, diff_estimated_delivery

```
SELECT
        c.customer_state,
        round(avg(oi.freight_value),2) as Avg_Freight_Value,
        round(avg(date_diff(o.order_delivered_customer_date,o.order_purchase_timestamp,day)),2)
               time_to_delivery_actual,
        round(avg(date_diff(o.order_estimated_delivery_date,o.order_purchase_timestamp,day)),2)
               time_to_delivery_estimate

FROM `target-365505.Target.customers` c
left join `target-365505.Target.orders` o on c.customer_id = o.customer_id
left join `target-365505.Target.order_items` oi on o.order_id = oi.order_id
group by c.customer_state
order by Avg_Freight_Value limit 5
```

Row	customer_state	Avg_Freight…	time_to_deli…	time_to_deli…
1	SP	15.15	8.26	18.9
2	PR	20.53	11.48	24.38
3	MG	20.63	11.52	24.31
4	RJ	20.96	14.69	26.09
5	DF	21.04	12.5	24.2

JOB INFORMATION   RESULTS   JSON   EXECUTION DETAILS

Row	customer_state	Avg_Freight…	time_to_deli…	time_to_deli…
1	RR	42.98	27.83	45.98
2	PB	42.72	20.12	32.53
3	RO	41.07	19.28	38.7
4	AC	40.07	20.33	40.7
5	PI	39.15	18.93	29.94

10. Payment type analysis

- Month over Month count of orders for different payment types

```
SELECT payment_type, Year, Month, Order_Ct,
    sum(Order_Ct) over (partition by payment_type order by  payment_type, Year, Month) as Running_Total
From
(
    SELECT  p.payment_type,
            Extract(Year from o.order_purchase_timestamp) as Year,
            Extract(Month from o.order_purchase_timestamp) as Month,
            count(*) as Order_Ct
    FROM `target-365505.Target.orders` o
        LEFT JOIN `target-365505.Target.payments` p
        ON o.order_id = p.order_id
    Group by p.payment_type, Year, Month
    having payment_type is not null
    order by payment_type, Year, Month
)
```

Row	payment_type	Year	Month	Order_Ct	Running_Tot...
16	UPI	2018	3	1352	13766
17	UPI	2018	4	1287	15053
18	UPI	2018	5	1263	16316
19	UPI	2018	6	1100	17416
20	UPI	2018	7	1229	18645
21	UPI	2018	8	1139	19784
22	credit_card	2016	9	3	3
23	credit_card	2016	10	254	257
24	credit_card	2016	12	1	258

Results per page: 50 ▼   1 – 50 of 90

- Distribution of payment installments and count of orders

```sql
SELECT payment_installments, concat(round(X.Order_Ct*100/sum(Order_ct) over(),2),"%") as Order_Count
FROM
(
    SELECT  p.payment_installments,
            count(*) as Order_Ct
    FROM `target-365505.Target.orders` o
        LEFT JOIN `target-365505.Target.payments` p
        ON o.order_id = p.order_id
    Group by payment_installments
    having payment_installments is not null
    order by Order_Ct desc
) as X
order by Order_Ct desc
```

Row	payment_in…	Order_Count
1	1	50.58%
2	2	11.95%
3	3	10.07%
4	4	6.83%
5	10	5.13%
6	5	5.04%
7	8	4.11%
8	6	3.77%
9	7	1.57%
10	0	0.60%

- ## Analysis of Payment Modes

```sql
SELECT payment_type, concat(round(X.Order_Ct*100/sum(Order_ct) over(),2),"%") as Order_Count
FROM
(
    SELECT  p.payment_type,
            count(*) as Order_Ct
    FROM `target-365505.Target.orders` o
        LEFT JOIN `target-365505.Target.payments` p
        ON o.order_id = p.order_id
    Group by payment_type
    having payment_type is not null
    order by Order_Ct desc
) as X
order by Order_Ct desc
```

payment_type	Order_Count
credit_card	73.92%
UPI	19.04%
voucher	5.56%
debit_card	1.47%
not_defined	0%

- Sellers in the states from where we have very low ordered placed

```
1  Select * FROM `target-365505.Target.sellers` s
2  where s.seller_state in ('RR','AP','AC')
```

Row	seller_id	seller_zip_c...	seller_city	seller_state
1	4be2e7f96b4fd749d52dff41f8...	69900	rio branco	AC

Cancelled Ordered %

```
SELECT round(Canceled_Count.cancelled_order*100/Total_Count.Total_Count,2) as
Canceled_Order_in_percentage FROM
    (SELECT count(*) as cancelled_order FROM `target-365505.Target.orders` o
    where o.order_status = 'canceled') as Canceled_Count,

    (SELECT count(*) as Total_Count FROM `target-365505.Target.orders` o
    ) as Total_Count;
```

Row	Canceled_O...
1	0.63

Insights

1. There is a growing trend in Brazil.
2. The average no. of orders has significantly increased from year by year. The jump was significantly high from 2016 to 2017 and then from 2017 to 2018 it was high but not that much as prior year.
3. There was almost no order in Nov and Dec of 2016.
4. The orders in 2017 continuously increased with exception of few months where it slightly decreases as compared to prior months. These months are Apr, Jun, Sep Dec. Looks like People tend to buy less at the end of quarter (assuming quarters begin from calendar year of Jan-Dec)
5. 2018 started with a bang but order started then onwards.
    i. Significantly decline from Aug'18 to Sep'18. Even in Oct'18 it didn't start to rise.
6. Maximum Brazilian (33%) people likes to order in Night followed by Afternoon, then Morning. People slightly tend to order more in Morning as compared to evening.
7. The top 3 states in Brazil having with maximum customer count are SP, RJ and MG which is 65% of total customers.
8. The top 3 states in Brazil from where maximum orders are placed are SP, RJ, MG
9. The bottom 3 states in Brazil from where orders are placed are very low are RR, AP, AC
10. There is around 150% increase in Freight cost from 2017 to 2018 (Jan-Aug).
11. Around 91% of the Delivery were made before time.
12. Around 7% of the Delivery were made on time.
13. Just less than 1.5% were Delayed.
14. The top 5 states with lowest freight charges are SP, PR, MG, RJ, DF
15. The top 5 states with highest freight charges are RR, PB, RO, AC, PI
16. Around 50% of the people like to have pay for their orders in 1 single payment.
17. Around 72% of the people have paid their payment with in 3 installments.
18. Almost all Brazilian use digital mode of payment like Credit Card, UPI. Around 74% orders were paid using Credit Card. UPI mode is low around

19% and Debit card is used rarely - just for 1.5% orders were paid using Debit cards.

19. Cancelled Ordered % is very low around 0.6%.
20. There is just 1 seller in the 3 states from where ordered placed are very low.

Recommendations

- The expected average delivery time for 95% of the order is > 10 days, which seems to be problematic. Customer normally doesn't wait so long. Either the estimation technique/tool is not correct, other wise the estimated time should be reduced. The actual time is comparatively vey low so most of the deliveries are on or before time.
- It should be reviewed why states like RR, PB, RO, AC, PI have freight charges. Can something be done reduce the freight charges.
- Need to give some attractive offers to people of states from where orders placed are very less like RR, AP, AC.
  Like Delivery Charges can be reduced.
- Ensure to have positive feedback from states providing good business-like SP, RJ, MG. So regular feedback review is recommended.
- Review if there can be more sellers in states from where we have very low order count. Increasing sellers will have likely hood of good earning and good purchase.