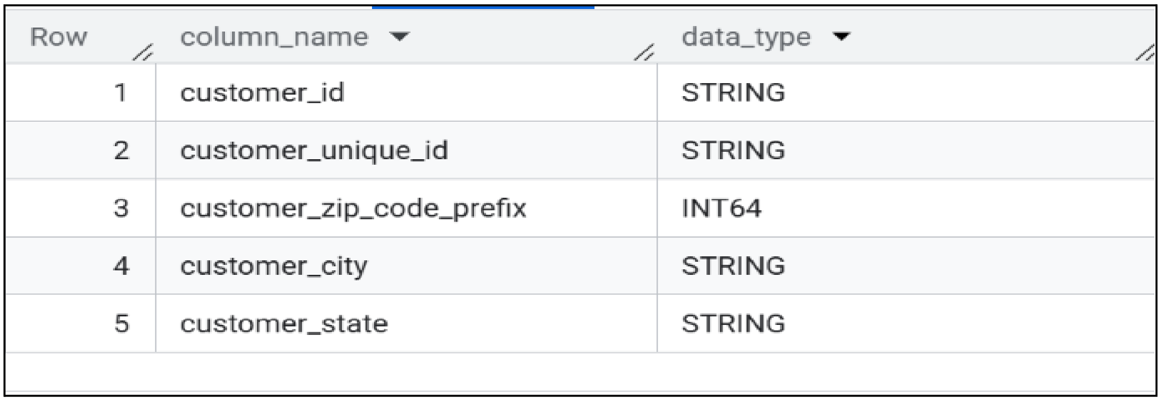
Introduction

This project analyzes Target's operations in Brazil using a dataset of 100,000 orders placed between 2016 and 2018, covering customer demographics, order details, payments, freight, and reviews. The objective is twofold: from a business perspective, to uncover actionable insights that can help Target optimize its pricing, logistics, and customer experience; and from a learner's perspective, to build practical expertise in SQL by solving real-world analytical problems. The findings are presented in a question-and-answer format, where each analysis highlights key insights and is followed by strategic recommendations to support data-driven decision-making.

1. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset.

1.a : Data type of all columns in the "customers" table.

```
select column_name ,
      data_type
from targetsql.INFORMATION_SCHEMA.COLUMNS
where table_name = "customers" ;
```



Row	column_name	data_type
1	customer_id	STRING
2	customer_unique_id	STRING
3	customer_zip_code_prefix	INT64
4	customer_city	STRING
5	customer_state	STRING

Interpretations :

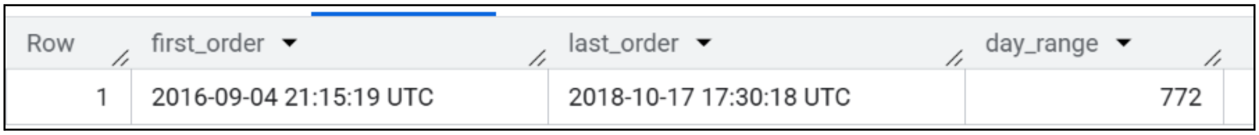
- customer_id and customer_unique_id → STRING
  These are identifiers. Storing them as strings is good idea since IDs are not used for numeric calculations.
- customer_zip_code_prefix → **INT64**
  This is numeric because it represents the first part of the ZIP code.
- `customer_city` and `customer_state` → **STRING**
  Both are categorical variables (location info), so string format is appropriate.

Business Recommendations :

- ZIP codes can be stored as strings to preserve leading zeros.
- **Regional logistics optimization**: With `customer_zip_code_prefix`, Target can optimize **last-mile delivery routes** and reduce freight costs.
- **Repeat purchase analysis**: By comparing `customer_id` vs `customer_unique_id`, the company can differentiate between **new vs returning customers** — helping in loyalty program design.

1.b : Get the time range between which the orders were placed.

```
select
    min(order_purchase_timestamp) as first_order,
    max(order_purchase_timestamp) as last_order,
    date_diff(max(order_purchase_timestamp) , min(order_purchase_timestamp) ,
day)                as day_range
from `targetsql.orders`;
```



Row	first_order	last_order	day_range
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC	772

Interpretations :

- The dataset spans **multiple festive cycles (2016, 2017, part of 2018)**, which allows us to analyze **shopping season peaks** like Black Friday and holiday sales.
- Since the data covers over 100,000 orders, it gives a **statistically reliable view** of customer demand across regions and time.

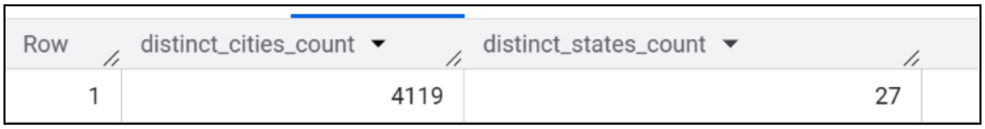
Business Recommendations :

- Since we have multiple years, Target Brazil should compare **Q4 demand (holiday season)** across years to refine **inventory stocking and marketing**.
- The time span allows tracking **year-on-year order growth** — if 2017 → 2018

- With 2+ years of customer data, Target can study **repeat purchase behavior** (who kept ordering year after year) and create **personalized loyalty programs**.

1.c : Count the Cities & States of customers who ordered during the given period.

```
select count (distinct customer_city) as distinct_cities_count ,
       count(distinct customer_state) as distinct_states_count
from `targetsql.customers`
```



Row	distinct_cities_count	distinct_states_count
1	4119	27

Interpretations :

- **Distinct cities with customers:** 4,119 **Distinct states with customers:** 27 Target already has **nationwide penetration** in Brazil through its online presence.

Business Recommendations :

- Since Target already has reach across all 27 states, focus on **regional marketing strategies** to grow demand in underrepresented areas (e.g., rural towns)
- Nationwide coverage means delivery reliability is key. Invest in **regional warehouses** or **3PL (third-party logistics) partnerships** to serve remote cities faster.
- Offer **premium same-day/next-day delivery** in urban centers.
- **economical standard shipping** in smaller towns to balance cost vs. customer satisfaction.
- Study which products sell best in each state and **customize inventory** to suit regional preferences.

2. In-depth Exploration:

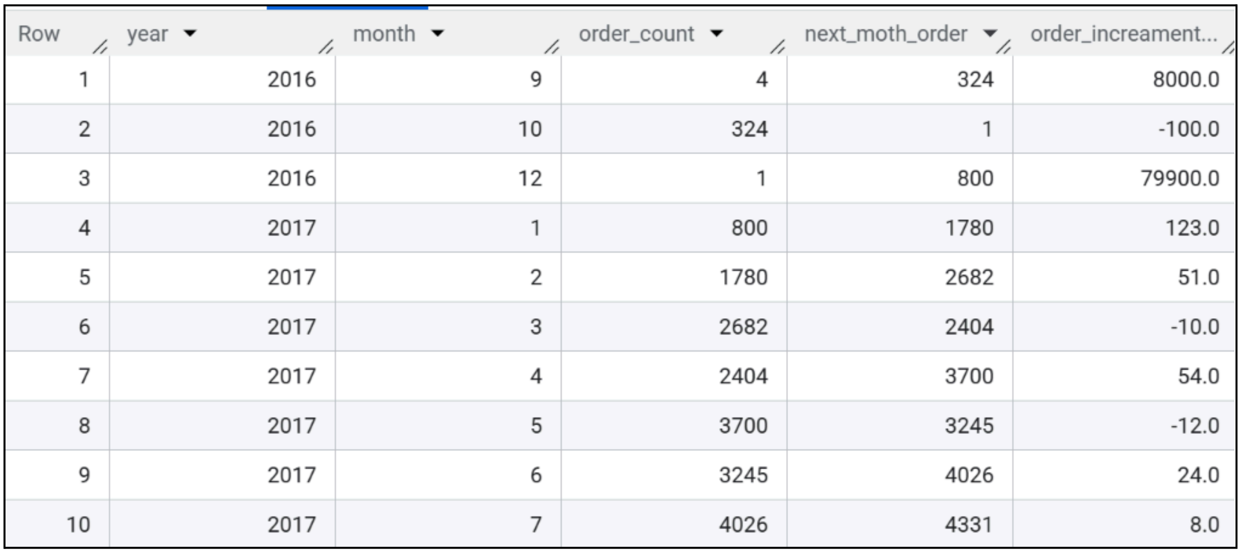
2 . a : Is there a growing trend in the no. of orders placed over the past years?

```
select *  ,
      lead ( order_count) over (order by year ,month  ) as next_moth_order ,
       round((((lead ( order_count) over (order by year ,month  )) -
     (order_count))/order_count* 100)) as order_increament_prcntg

 from
   (
   select  extract(year from order_purchase_timestamp ) as year ,
           extract(month from order_purchase_timestamp ) as month ,
           count(order_id) as order_count
   from `targetsql.orders`
   group by 1,2

   ) as temp_table

   order by year,month
```



Row	year	month	order_count	next_moth_order	order_increament...
1	2016	9	4	324	8000.0
2	2016	10	324	1	-100.0
3	2016	12	1	800	79900.0
4	2017	1	800	1780	123.0
5	2017	2	1780	2682	51.0
6	2017	3	2682	2404	-10.0
7	2017	4	2404	3700	54.0
8	2017	5	3700	3245	-12.0
9	2017	6	3245	4026	24.0
10	2017	7	4026	4331	8.0

Interpretations :

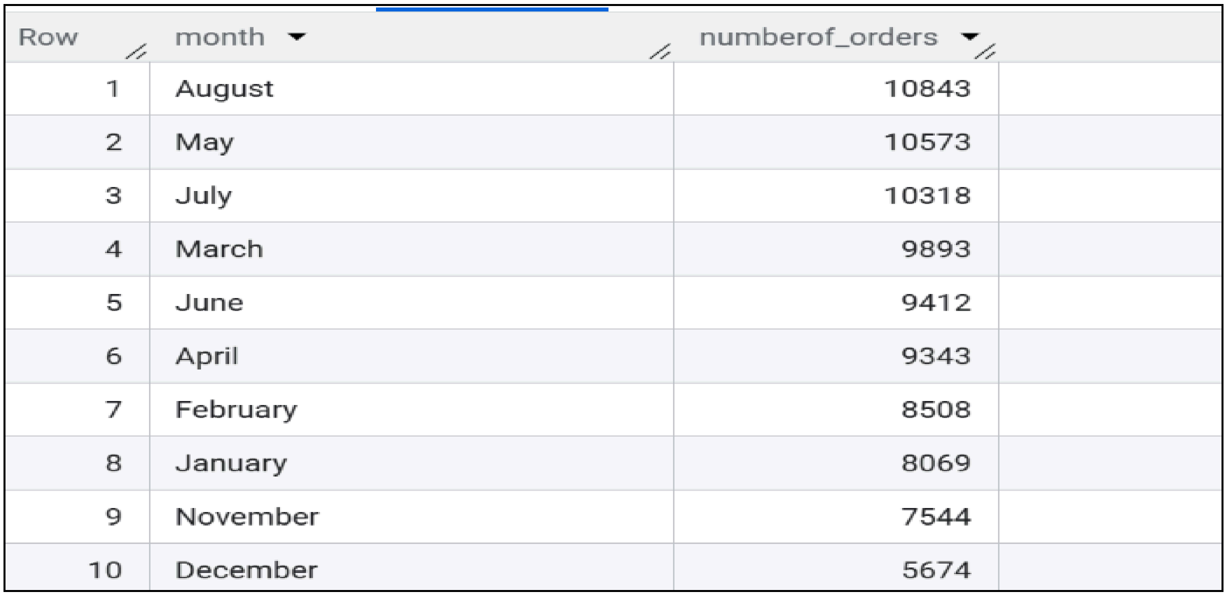
- In late **2016**, the dataset shows very few orders as data starts from September.
- In **2017**, there is a strong increase in monthly orders: e.g., Jan (800 orders) → Aug (4,331 orders). Despite some small dips (March -10%, May -12%), the overall **trend is upward**.
- By **mid-2017**, order volumes are consistently crossing **4,000 per month**, showing strong adoption of Target's e-commerce platform in Brazil.

Business Recommendations :

- Since order growth is strong, Target should **invest in additional fulfillment centers** and expand **delivery fleet capacity** to handle future demand.
- Use month-on-month order data to build **forecasting models** that predict seasonal drops and peaks → optimize staffing and inventory.
- Months with negative growth (e.g., March, May) can be boosted with **targeted discounts or campaigns** to maintain momentum.
- As order volumes grow, focus on converting **first-time buyers into repeat customers** through loyalty points, faster shipping, or personalized offers.

2.b : Can we see some kind of monthly seasonality in terms of the no. of orders being placed?

```
select FORMAT_DATE('%B', order_purchase_timestamp) as month , count(order_id) as
numberof_orders
from `targetsql.orders`
group by 1
order by numberof_orders desc
```



Row	month	numberof_orders
1	August	10843
2	May	10573
3	July	10318
4	March	9893
5	June	9412
6	April	9343
7	February	8508
8	January	8069
9	November	7544
10	December	5674

Interpretations :

- Late 2016: Very few orders (dataset just begins).
- 2017: Orders gradually increase through the year, **peaking in November** (7,544 orders). December shows a drop (5,673), likely post-holiday slowdown.
- 2018: Orders **remain consistently high** (6,100–7,200 range), confirming stabilized demand after the growth phase.

Business Recommendations :

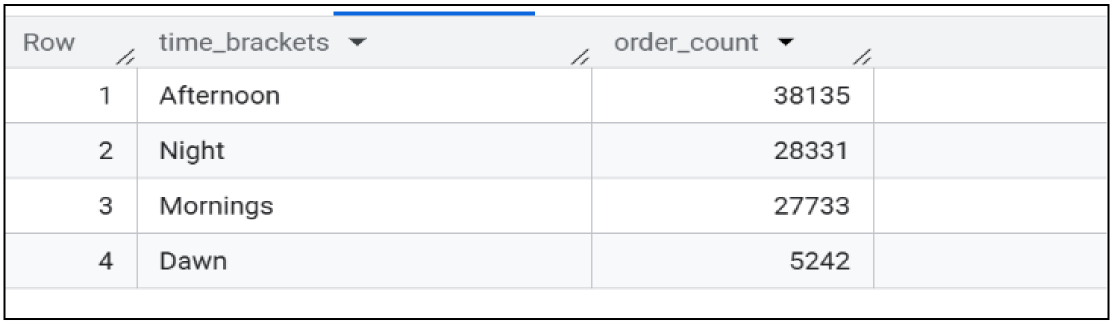
- **Black Friday Readiness**: Increase inventory, expand seller partnerships, and ramp up logistics capacity in November to handle the seasonal surge.
- **Holiday Campaign Shift:** Instead of focusing only on December holidays, front-load promotions to November when customer intent to buy is highest.
- **Post-Peak Engagement:** Since December orders drop after November, run loyalty programs and targeted promotions in January–March to maintain momentum.
- **Long-Term Planning**: Use these seasonal insights to forecast demand and allocate budgets for marketing, warehousing, and delivery staff.

2.c : During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)

- **0-6 hrs : Dawn**
- **7-12 hrs : Mornings**
- **13-18 hrs : Afternoon**
- **19-23 hrs : Night**

```
select
```

```
(case when extract(hour from order_purchase_timestamp ) between 0 and 6 then "Dawn"
     when extract(hour from order_purchase_timestamp ) between 7 and 12 then "Mornings"
     when extract(hour from order_purchase_timestamp ) between 13 and 18 then "Afternoon"
     else "Night"
     end) as  time_brackets , count(order_id) as order_count
from `targetsql.orders`
group by  1
order by order_count desc
```



Row	time_brackets	order_count
1	Afternoon	38135
2	Night	28331
3	Mornings	27733
4	Dawn	5242

Interpretations :

- The Afternoon is the most popular time for placing orders.
- Ordering activity remains strong at Night and in the Mornings.
- Dawn sees significantly less order activity.

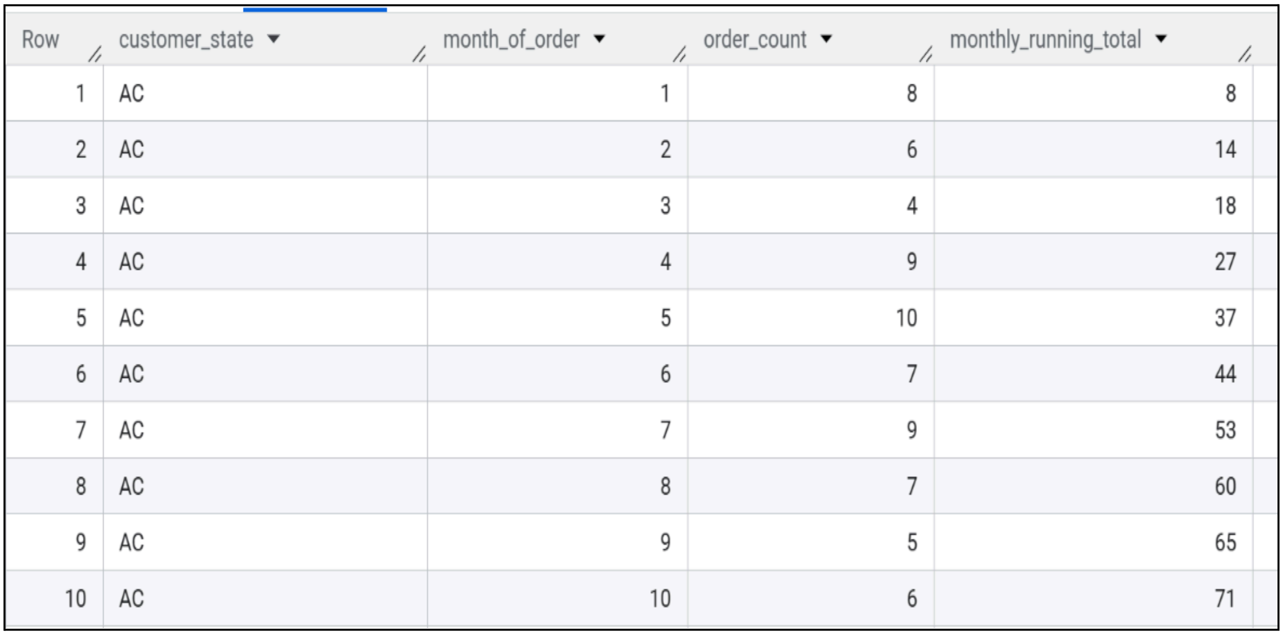
Business Recommendations :

- Marketing & Promotion Timing: Schedule marketing emails, push notifications, and special deals to go live around noon or early afternoon, aligning with peak shopping activity.
- Operational Readiness: Ensure customer service and fulfillment teams are well-staffed during afternoons to handle surges in orders and inquiries.
- Tech Infrastructure: Plan any necessary site maintenance or backend updates outside peak hours (ideally very early morning—Dawn) to minimize impacts on customers.
- Personalization: Consider running flash sales or exclusive offers in the afternoon or evening to further boost conversions during these already high-activity hours.

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

3.a : Get the month on month no. of orders placed in each state.

```
select * , sum(order_count) over(partition by customer_state order by month_of_order )
monthly_running_total
from
(
  select customer_state ,extract(month from order_purchase_timestamp) as month_of_order,
  count(order_id) as order_count
  from `targetsql.orders` o join `targetsql.customers` c
  on o.customer_id = c.customer_id
  group by customer_state ,extract(month from order_purchase_timestamp)
  order by customer_state , month_of_order
)
```



Row	customer_state	month_of_order	order_count	monthly_running_total
1	AC	1	8	8
2	AC	2	6	14
3	AC	3	4	18
4	AC	4	9	27
5	AC	5	10	37
6	AC	6	7	44
7	AC	7	9	53
8	AC	8	7	60
9	AC	9	5	65
10	AC	10	6	71

Interpretations :

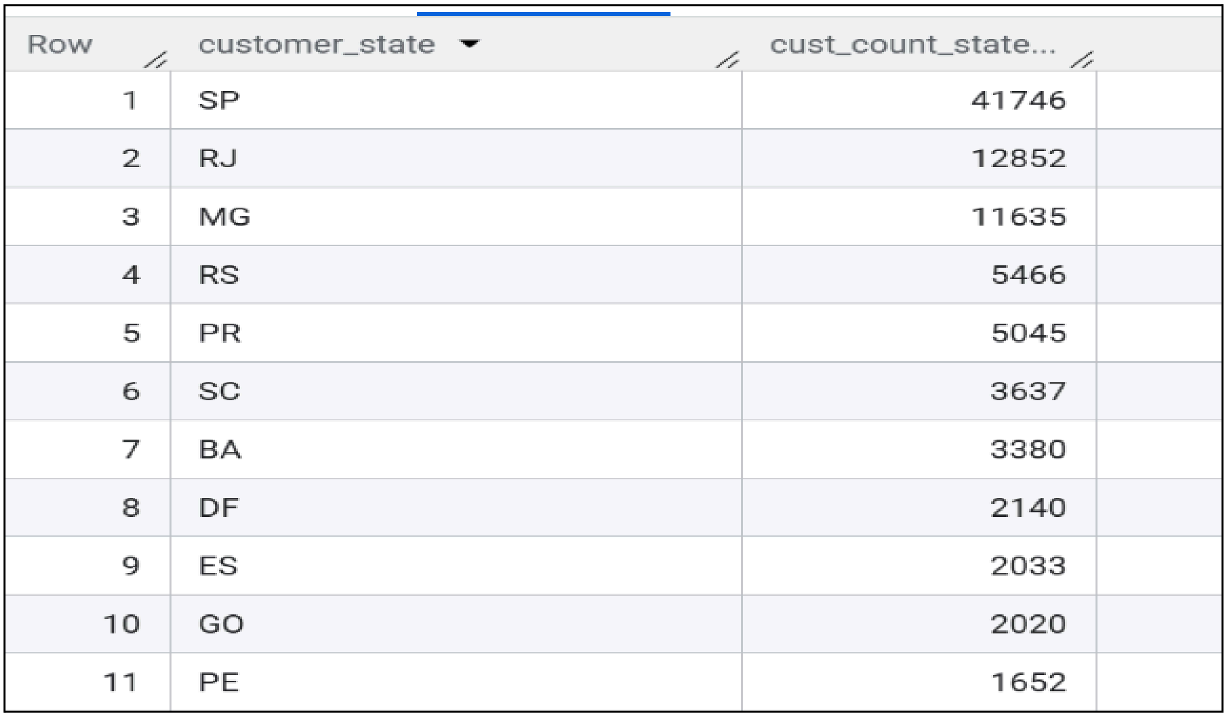
- Some states have low orders and other have high number of orders
- Sales fluctuate over time and locations hence one size fits for all wont work

Business Recommendations :

- Identify Key Markets: Focus growth efforts and promotions on states with larger order volumes. For example state BA and MG (not able to show in the screenshot attached) with running total of over 3000 and 6000 orders respectively.
- Optimize Logistics: For regions with slow or small order growth, optimize delivery costs by consolidating shipments or using regional courier partners.

3.b : How are the customers distributed across all the states?

```
select customer_state , count(distinct customer_id)  as cust_count_statewise
from `targetsql.customers`
group by customer_state
order by cust_count_statewise desc
```



Row	customer_state	cust_count_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
11	PE	1652

Interpretations :

- A few states like SP,RJ,MG etc may have the disproportionately large number of customers showing stronghold
- Other states like AC, AP, RR show lower numbers of customers showing low awareness, lack of distribution etc .

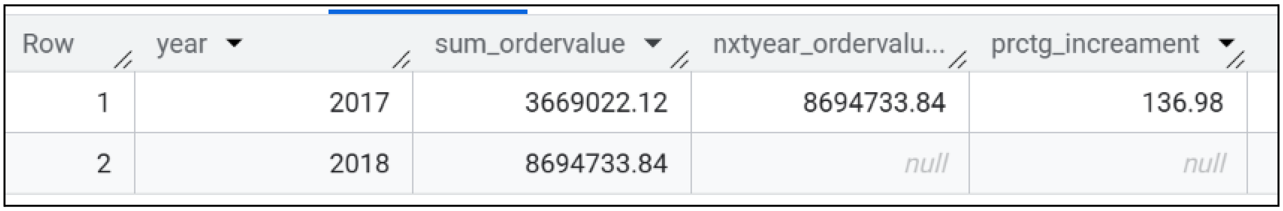
Business Recommendations :

- Resource Allocation : Allocate sales teams, warehouses, and ad budgets according to customer density in each state.
- Focus Marketing on High-Customer States :Double down with loyalty programs, upselling, or cross-selling in states where you already have strong customer bases.
- Expansion Strategy: For states with low penetration but high potential (large population, rising income), increase marketing efforts, improve distribution channels, or run targeted campaigns.
- High-Customer States customer retention, loyalty programs, personalized offers, and premium product upselling.
- Medium-Customer States - scale marketing campaigns and improve delivery coverage.
- Low-Customer States Low-Customer States (North, Northeast regions):Potential for expansion and customer acquisition.Investigate barriers: is it logistics (longer delivery times, higher freight costs), pricing power, or lack of awareness? Strategy could be targeted discounts, free/low shipping promotions, and regionalized advertising.

4 . Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.

4.a : Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only).

```
select * ,  lead(sum_ordervalue) over(order by year) as nxtyear_ordervalue_sum  ,
round(((lead(sum_ordervalue) over(order by year) - sum_ordervalue)/sum_ordervalue
*100),2) as prctg_increament
from
(
    select extract( year from o.order_purchase_timestamp) year,
    round(sum(p.payment_value),2) as sum_ordervalue
    from `targetsql.orders` o join `targetsql.payments` p
    on o.order_id=p.order_id
    where extract(year from o.order_purchase_timestamp) = 2017 and extract( month from
    o.order_purchase_timestamp) in (1,2,3,4,5,6,7,8)
    or extract( year from o.order_purchase_timestamp) = 2018  and extract( month from
    o.order_purchase_timestamp) in (1,2,3,4,5,6,7,8)
    group by 1
    order by 1
)
```



Row	year	sum_ordervalue	nxtyear_ordervalu...	prctg_increament
1	2017	3669022.12	8694733.84	136.98
2	2018	8694733.84	null	null

Interpretations :

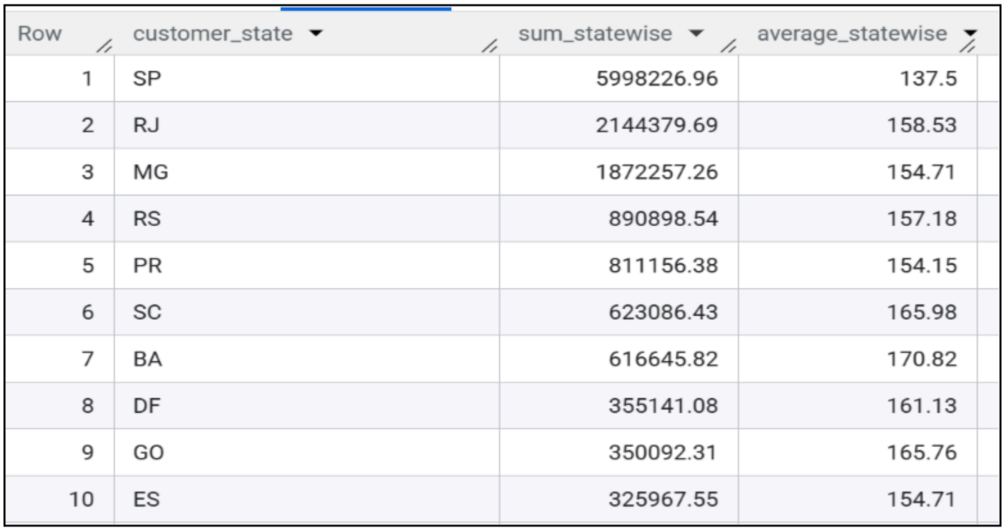
- The cost of orders (total revenue or average order value, depending on calculation) more than doubled in 2018 compared to the same months in 2017.
- A 136.98% increase is significant, suggesting a strong surge in sales activity or order value.

Business Recommendations :

- With rapid customer acquisition, the next focus should be on retaining 2018 customers and converting them into repeat buyers.
- Sustainability Check : Important to check if this growth is profitable. If growth is driven by heavy discounts or high shipping subsidies, margins may be under pressure.
- Capacity & Logistics: With more than double orders, Target must ensure supply chain, warehouses, and delivery partners can handle the increased demand.

4.b : Calculate the Total & Average value of order price for each state.

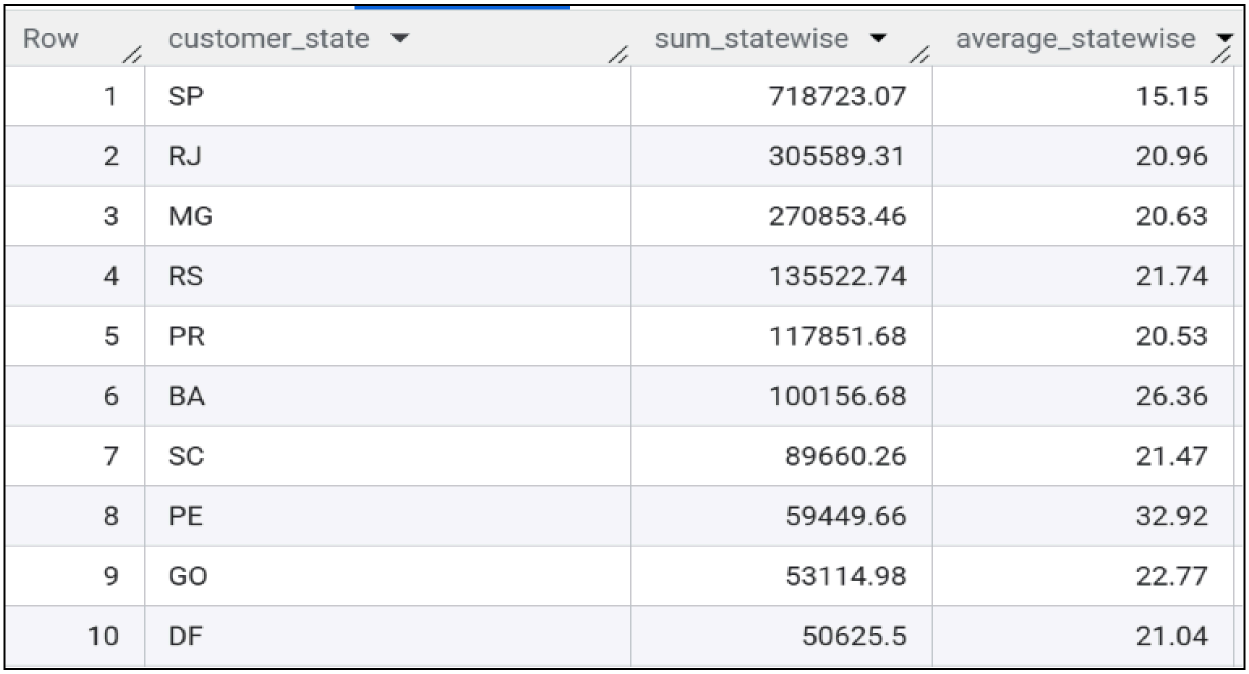
```
select c.customer_state , round(sum (p.payment_value),2) as sum_statewise ,
round(avg(p.payment_value),2) as average_statewise
from `targetsql.orders` o left join `targetsql.payments` p
on o.order_id = p.order_id
left join `targetsql.customers` c
on c.customer_id = o.customer_id
group by 1
order by sum_statewise desc
```



Row	customer_state	sum_statewise	average_statewise
1	SP	5998226.96	137.5
2	RJ	2144379.69	158.53
3	MG	1872257.26	154.71
4	RS	890898.54	157.18
5	PR	811156.38	154.15
6	SC	623086.43	165.98
7	BA	616645.82	170.82
8	DF	355141.08	161.13
9	GO	350092.31	165.76
10	ES	325967.55	154.71

4.c : Calculate the Total & Average value of order freight for each state.

```sql
select c.customer_state , round(sum (oi.freight_value),2) as sum_statewise ,
round(avg(oi.freight_value),2) as average_statewise
from `targetsql.orders` o left join `targetsql.order_items` oi
on o.order_id = oi.order_id
left join `targetsql.customers` c
on c.customer_id = o.customer_id
group by 1
order by sum_statewise desc
```



Row	customer_state	sum_statewise	average_statewise
1	SP	718723.07	15.15
2	RJ	305589.31	20.96
3	MG	270853.46	20.63
4	RS	135522.74	21.74
5	PR	117851.68	20.53
6	BA	100156.68	26.36
7	SC	89660.26	21.47
8	PE	59449.66	32.92
9	GO	53114.98	22.77
10	DF	50625.5	21.04

Interpretation

- High Total Freight States like SP,RJ,MG etc suggesting high number of orders or long-distance deliveries.
- Example: SP have high total freight due to sheer order volume.
- High Average Freight States like RR ,RO Indicates states that are geographically distant from distribution centers
- Low Average Freight States : Usually closer to logistics hubs.

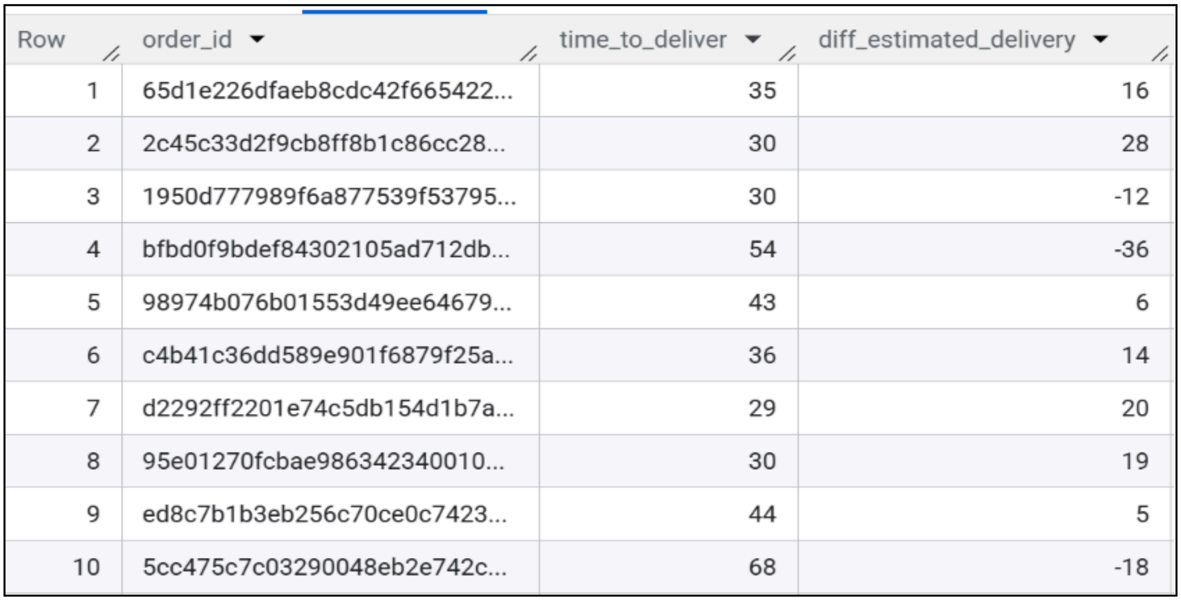
Business Recommendations :

- Pricing Strategy → Consider subsidizing shipping in high-cost states to encourage adoption.
- Growth Strategy → If a state has low total freight but high average freight, it indicates low penetration but expensive delivery → good to analyze whether expansion is worth it.
- High Average Freight could suggest logistics inefficiency or the need for regional warehouses.

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

5.a : Find the no. of days taken to deliver each order from the order's purchase date as delivery time. Also, calculate the difference (in days) between the estimated & actual delivery date of an order. Do this in a single query.

```
select order_id ,
      date_diff(order_delivered_customer_date, order_purchase_timestamp, day) as
      time_to_deliver ,
      date_diff(order_estimated_delivery_date , order_delivered_customer_date , day) as
      diff_estimated_delivery
from `targetsql.orders`
```



Row	order_id	time_to_deliver	diff_estimated_delivery
1	65d1e226dfaeb8cdc42f665422...	35	16
2	2c45c33d2f9cb8ff8b1c86cc28...	30	28
3	1950d777989f6a877539f53795...	30	-12
4	bfbd0f9bdef84302105ad712db...	54	-36
5	98974b076b01553d49ee64679...	43	6
6	c4b41c36dd589e901f6879f25a...	36	14
7	d2292ff2201e74c5db154d1b7a...	29	20
8	95e01270fcbae986342340010...	30	19
9	ed8c7b1b3eb256c70ce0c7423...	44	5
10	5cc475c7c03290048eb2e742c...	68	-18

Interpretation:

- **Delivery Time** (in days)→ Average delivery time across all orders shows **logistics efficiency**.
- **Difference between Estimated & Actual Delivery**:
  - Negative values → Delivered **before** estimate.
  - Zero → Delivered **on time**.
  - Positive values → Delivered **late**.

Business Recommendations :

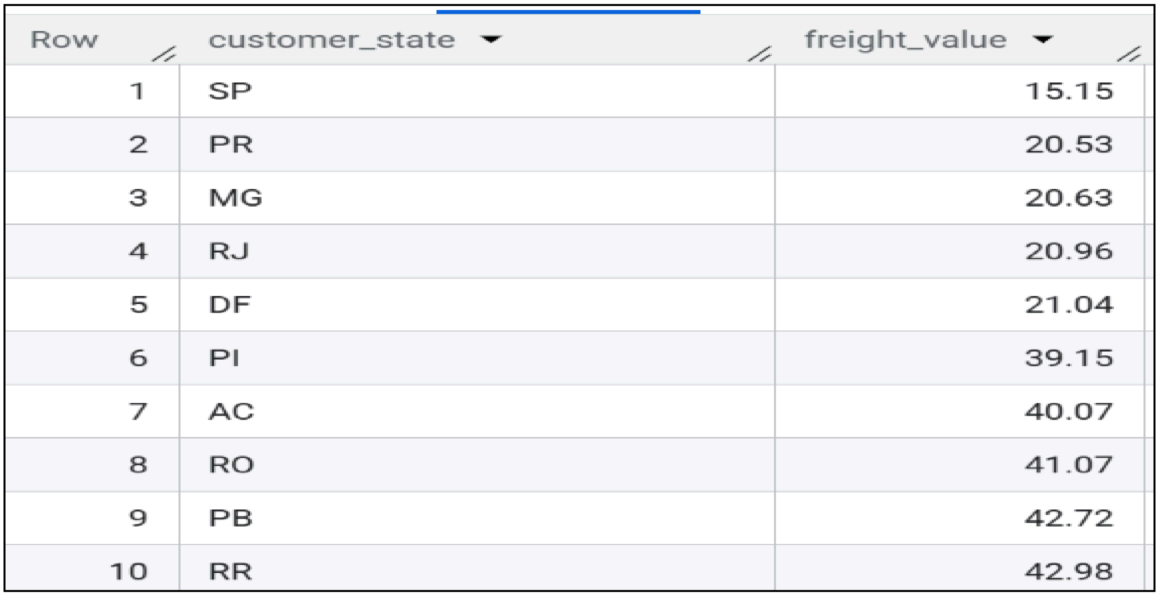
- High delivery time could be due to warehouse unavailability , remote location orders , or inefficient distribution network.Identifying the exact reason through complementary data analysis and solving the blockers.
- Huge difference in actual and estimated delivery could be due to inaccurate estimation algorithms. Integrating AI in this part can be done to increase the accuracy in estimation.

5.b : Find out the top 5 states with the highest & lowest average freight value.

```
(select  c.customer_state , round(avg(oi.freight_value),2) as freight_value
from `targetsql.customers` c inner join `targetsql.orders` o
on o.customer_id = c.customer_id
inner join `targetsql.order_items` oi
on o.order_id= oi.order_id
group by 1
order by freight_value
limit 5
)
as top_records

union all

(
select  c.customer_state , round(avg(oi.freight_value),2) as freight_value
from `targetsql.customers` c inner join `targetsql.orders` o
on o.customer_id = c.customer_id
inner join `targetsql.order_items` oi
on o.order_id= oi.order_id
group by 1
order by freight_value desc
limit 5
)
order by freight_value
```



Row	customer_state	freight_value
1	SP	15.15
2	PR	20.53
3	MG	20.63
4	RJ	20.96
5	DF	21.04
6	PI	39.15
7	AC	40.07
8	RO	41.07
9	PB	42.72
10	RR	42.98

Interpretation:

- Lowest Avg. Freight States: SP , PR , MG , RJ , DF : **densely populated, urban, and economically strong states**.They are close to major **logistics hubs and distribution centers**.Freight costs are lower because of **economies of scale** and **efficient infrastructure**.

- **Highest Avg. Freight States: PI, AC , RO, PB , RR :** These states are **geographically distant**, many located in **North/Northeast Brazil**.Lack of nearby warehouses → longer delivery routes → **expensive last-mile delivery**.Orders could be fewer, so **no economies of scale** to reduce per-order shipping costs.

Business Recommendations :

- For Lowest Freight States:
  - ☐ Push **fast shipping (same-day/next-day)** as a competitive advantage.
  - ☐ Customers here can be offered **premium paid delivery options** (same-day express) for extra revenue.
  - ☐ Since freight is cheap, bulky/heavy items can be promoted more aggressively here.

- For Highest Freight States
  - ☐ Explore **regional warehouses** or partner with **local 3PL providers** to reduce shipping distance.
  - ☐ Use **pickup points or collection lockers** in urban centers to cut last-mile delivery costs.

- ☐ Offer **free shipping thresholds** (e.g., free delivery on orders above R$X) to encourage bigger baskets and offset logistics costs.
- ☐ Focus on **digital products, lightweight items, or high-margin categories** where freight has less impact.

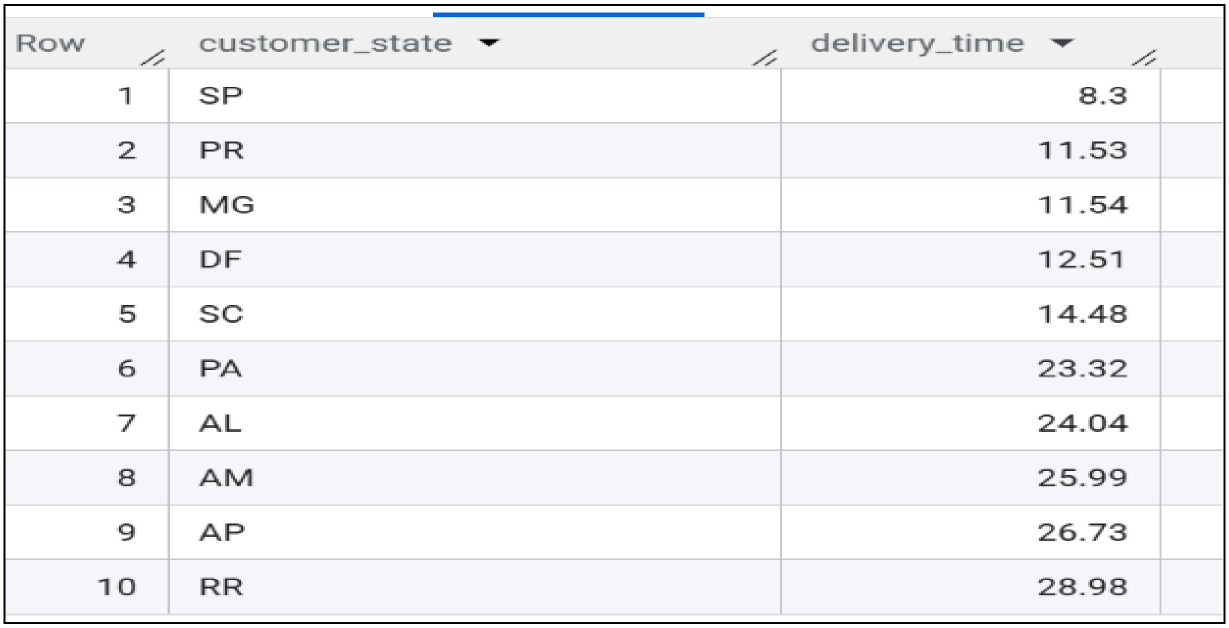
5.c : Find out the top 5 states with the highest & lowest average delivery time.

```
(select c.customer_state ,
       round(avg(date_diff(o.order_delivered_customer_date , o.order_purchase_timestamp
, day)),2) as delivery_time
from `targetsql.orders` o inner join `targetsql.customers` c
on c.customer_id = o.customer_id
group by 1
order by 2
limit 5) as top_five

union all

(select c.customer_state ,
       round(avg(date_diff(o.order_delivered_customer_date , o.order_purchase_timestamp
, day)),2) as delivery_time
from `targetsql.orders` o inner join `targetsql.customers` c
on c.customer_id = o.customer_id
group by 1
order by 2 desc
limit 5) as last_five

order by delivery_time
```



Row	customer_state	delivery_time
1	SP	8.3
2	PR	11.53
3	MG	11.54
4	DF	12.51
5	SC	14.48
6	PA	23.32
7	AL	24.04
8	AM	25.99
9	AP	26.73
10	RR	28.98

Interpretation:

Mostly same interpretations as previous question

- States with **Lowest Average Delivery Time (SP,PR,MG,DF,SC) :** These are **economically strong, urban, well-connected states** with **dense customer bases**,**proximity to distribution hubs**, **better infrastructure**, and **shorter delivery routes**.
-
- **States with Highest Average Delivery Time (PA,AL,AM,AP,RR): states** with **challenging geography** (Amazon rainforest, river transport, fewer highways).

Business Recommendations :

For Lowest Delivery Time States

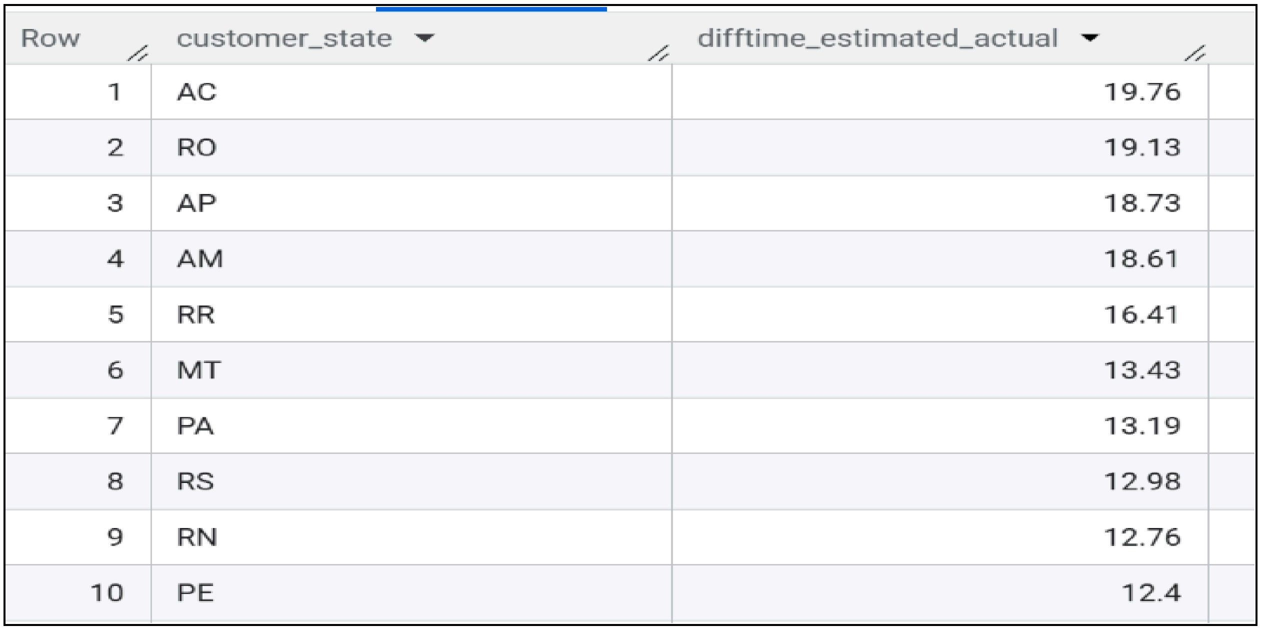
- **fast delivery guarantees** (same-day/next-day) as a selling point.
- Introduce **paid express delivery** options for customers who want even faster shipping.

For Highest Delivery Time States

- **Logistics Optimization** → Partner with **regional carriers** and invest in **micro-warehouses** closer to demand centers.
- **Long-Term Strategy** → Explore **river transport optimization**, **air freight partnerships**, and **localized inventory stocking** for the Amazon region.

5.d : Find out the top 10 states where the order delivery is really fast as compared to the estimated date of delivery.

```
select c.customer_state ,round(avg(date_diff(order_estimated_delivery_date ,
order_delivered_customer_date , day)),2) as difftime_estimated_actual
from `targetsql.orders` o inner join `targetsql.customers` c
on c.customer_id = o.customer_id
where o.order_status = "delivered"
group by 1
order by 2 desc
```



Row	customer_state	difftime_estimated_actual
1	AC	19.76
2	RO	19.13
3	AP	18.73
4	AM	18.61
5	RR	16.41
6	MT	13.43
7	PA	13.19
8	RS	12.98
9	RN	12.76
10	PE	12.4

Interpretation :

- The top 10 states like AC,RO,AP,AM,RR etc. with **fast delivery compared to estimated date** show that Target’s logistics network is **overperforming** in those regions.
- Customers here often receive products **earlier than expected**, which enhances **trust and satisfaction**.
- However, consistent early delivery may also indicate that **estimated dates are too conservative**.

Business Recommendations :

- **Customer Delight** → Highlight "early delivery" in marketing as a value proposition.
- **Adjust Estimated Dates** → Refine ETA algorithms so estimates are **closer to actuals**, reducing the gap.
- **Premium Option** → Introduce **express delivery services** in these states, since logistics already outperform.
- **Resource Reallocation** → Use learnings from these states' logistics to **replicate efficiency** in slower regions.

6. Analysis based on the payments:

6.a : Find the month on month no. of orders placed using different payment types.

```
select payment_type,
       extract( year from order_purchase_timestamp) order_year,
       extract(month from order_purchase_timestamp) as order_month,
       count(o.order_id) as order_count_payment_type
from `targetsql.orders` o  inner join `targetsql.payments` p
on o.order_id = p.order_id
group by 1,2,3
order by payment_type,order_year,order_month asc
```

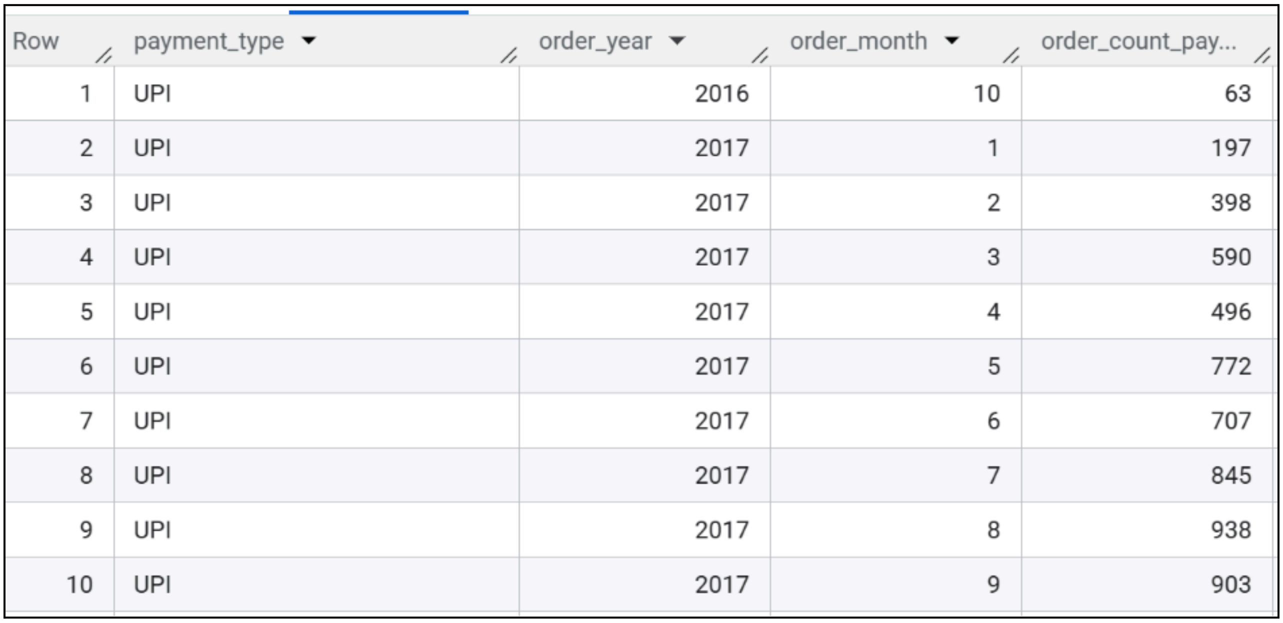
Interpretation :

- **UPI and Credit Card** are the **dominant payment types**, together making up the bulk of orders.
- **Debit Cards and Vouchers** contribute much less, but still show gradual adoption.
- **Month-on-month upward trend across all payment types** indicates overall **growing customer confidence in digital payments**.

Business Recommendations :

- **Leverage UPI Growth** → Encourage more UPI payments by offering **instant cashback, discounts, or faster refunds**.
- **Strengthen Credit Card Partnerships** → Tie up with banks for **installment offers, reward points, or co-branded promotions** to retain card dominance.

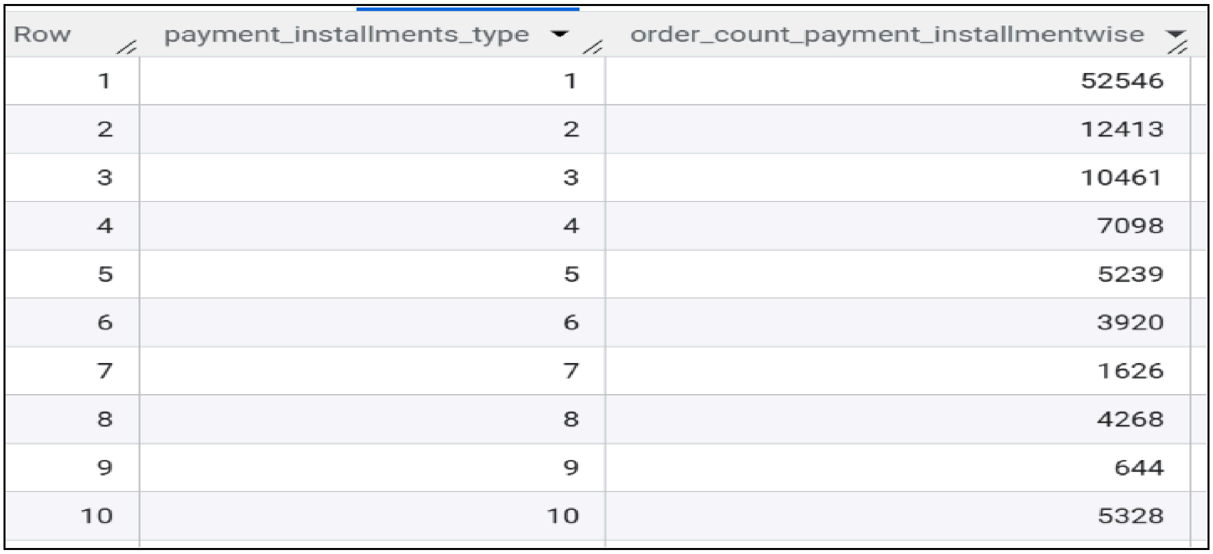
- **Revive Voucher Adoption** → Position vouchers as **gifting options** and integrate them into loyalty/referral programs.
- **Future-readiness** → Since all types are rising, ensure **scalable payment infrastructure** and **fraud protection systems** to handle growing volumes.



Row	payment_type	order_year	order_month	order_count_pay...
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

6.b : Find the no. of orders placed on the basis of the payment installments that have been paid.

```
select payment_installments as payment_installments_type , count(order_id) as
order_count_payment_installmentwise from `targetsql.payments`
where payment_installments>0  and payment_sequential >=1
group by 1
order by 1
```



Row	payment_installments_type	order_count_payment_installmentwise
1	1	52546
2	2	12413
3	3	10461
4	4	7098
5	5	5239
6	6	3920
7	7	1626
8	8	4268
9	9	644
10	10	5328

Interpretation :

- Customers prefer paying in fewer installments (or full upfront).
- Higher installment counts → fewer orders, meaning customers avoid long-term financial commitments.
- Installments are mostly used for high-value purchases, which could be less frequent.

Business Recommendations :

- Target should optimize for short-term EMIs on regular products while incentivizing long-term EMIs only for premium items to drive high-value sales without overburdening customers.
- Alongside, Use product pages to clearly show monthly breakdowns (e.g., “Only R$ 199/month for 12 months”) to make higher installment plans more appealing.