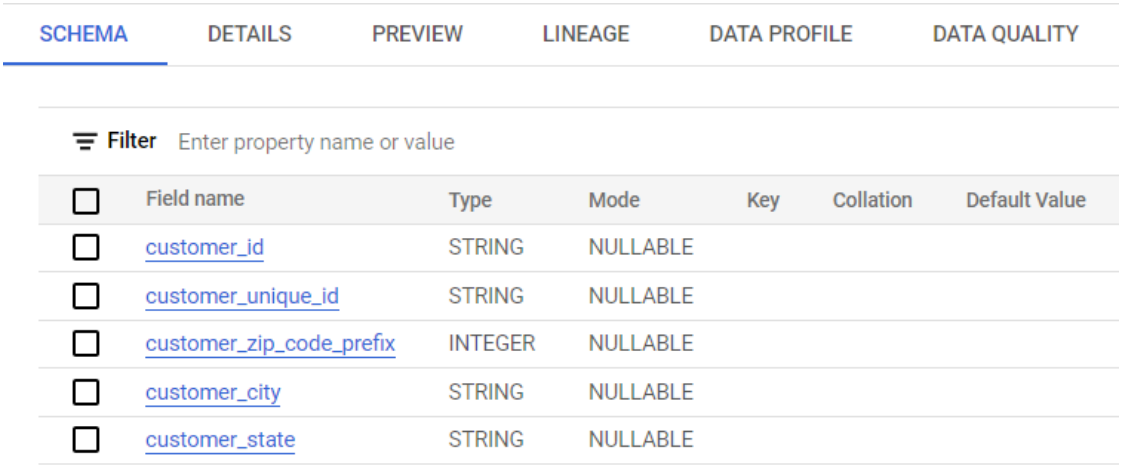
Business Case 1 : Target SQL

- **Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset:**

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



Field name	Type	Mode	Key	Collation	Default Value
customer_id	STRING	NULLABLE			
customer_unique_id	STRING	NULLABLE			
customer_zip_code_prefix	INTEGER	NULLABLE			
customer_city	STRING	NULLABLE			
customer_state	STRING	NULLABLE			

Key Points:

- The customer data includes various columns containing information about customers who have previously made purchases from the target.
- Except for the 'customer_zip_code_prefix,' all these columns contain text-based information, and it's possible for some of them to be empty or NULL for certain customers.

---

2. **Get the time range between which the orders were placed.**

Query:

```
select
min(order_purchase_timestamp) as first_purchase,
max(order_purchase_timestamp) as last_purchase,
timestamp_diff(max(order_purchase_timestamp), min(order_purchase_timestamp), day) as duration_in_days
from target.orders
```

Output:

Row	first_purchase	last_purchase	duration_in_days
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC	772

Key Points:

- The first order was made in September 2016, and the most recent one was in October 2018, within a time span of 772 days in between.

---

3. **Count the Cities & States of customers who ordered during the given period.**

Query:

```
select customer_city, customer_state, count(*) as total_count,
from target.customers as c1
join target.orders as o1
on c1.customer_id = o1.customer_id
group by customer_city, customer_state
order by total_count desc ;
```

Output:

Row	customer_city	customer_state	total_count
1	sao paulo	SP	15540
2	rio de janeiro	RJ	6882
3	belo horizonte	MG	2773
4	brasilia	DF	2131
5	curitiba	PR	1521
6	campinas	SP	1444
7	porto alegre	RS	1379
8	salvador	BA	1245
9	guarulhos	SP	1189
10	sao bernardo do campo	SP	938

Key Points:

- Sao Paulo customers led with the highest order count, followed by those in Rio de Janeiro, Belo Horizonte, and other areas.

- **In-depth Exploration:**

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

Query:

```
select extract(year from order_purchase_timestamp) as Year,
count(*) as Number_of_orders
from target.orders
group by Year
order by Year ;
```

Output:

Row	Year	Number_of_orders
1	2016	329
2	2017	45101
3	2018	54011

Key Points:

- We can notice a rise in the number of orders, starting from 329 in 2016 and reaching 54,000 by 2018.

---

5. **Can we see some kind of monthly seasonality in terms of the no. of orders being placed?**

Query:

```
select extract(month from order_purchase_timestamp) as Month,
count(*) as Number_of_orders
from target.orders
group by Month
order by Month ;
```

Output:

Row	Month	Number_of_orders
1	1	8069
2	2	8508
3	3	9893
4	4	9343
5	5	10573
6	6	9412
7	7	10318
8	8	10843
9	9	4305
10	10	4959
11	11	7544
12	12	5674

Key Points:

- We can observe an increase in the number of orders during the second and third quarters of the year, with an unexpected drop in the fourth quarter.
- The highest number of orders occurred in May, July, and August.
- The lowest number of orders was recorded in September, October, and December.

---

6. **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**

Query:

```
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'
  when extract(hour from order_purchase_timestamp) between 19 and 23 then 'Night'
  else 'Other'
end as time_of_day,
count(*) as Number_of_orders
from `target.orders`
group by time_of_day
order by Number_of_orders desc ;
```

Output:

Row	time_of_day	Number_of_orders
1	Afternoon	38135
2	Night	28331
3	Mornings	27733
4	Dawn	5242

Key Points:

- From this, we can deduce that Brazilian customers tend to place the most orders during the afternoon and the fewest during the dawn hours.

- **Evolution of E-commerce orders in the Brazil region:**

7. **Get the month on month no. of orders placed in each state.**

Query:

```
select cust.customer_state,
extract(month from ord.order_purchase_timestamp) as Month,
count(*) as Number_of_orders
from target.customers as cust
join target.orders as ord
on cust.customer_id = ord.customer_id
group by cust.customer_state, Month
order by cust.customer_state, Month ;
```

Output:

Row	customer_state	Month	Number_of_orders
1	AC	1	8
2	AC	2	6
3	AC	3	4
4	AC	4	9
5	AC	5	10
6	AC	6	7
7	AC	7	9
8	AC	8	7
9	AC	9	5
10	AC	10	6
11	AC	11	5
12	AC	12	5

8. **How are the customers distributed across all the states?**

Query:

```
select customer_state, count(*) as number_of_customers
from `target.customers`
group by customer_state
order by number_of_customers desc ;
```

Output:

Row	customer_state	number_of_customers
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

Key Points:

- The largest customer base is in the state of São Paulo, with approximately 41,750 customers. It is followed by Rio de Janeiro, Minas Gerais, Rio Grande do Sul, and so forth.

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

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

Query:

```
with cte as (
  select extract(year from t1.order_purchase_timestamp) as Year,
round(sum(t2.payment_value), 2) as total_cost_of_orders
  from `target.orders` as t1
  join `target.payments` as t2
  on t1.order_id = t2.order_id
  where (extract(year from t1.order_purchase_timestamp) = 2018 and extract(month from
t1.order_purchase_timestamp) between 1 and 8)
  or extract(year from t1.order_purchase_timestamp) = 2017
  group by Year )
select
round((sum(case when Year = 2018 then total_cost_of_orders else 0 end) - sum(case
when Year = 2017 then total_cost_of_orders else 0 end)) * 100.00 /
sum(case when Year = 2018 then total_cost_of_orders else 0 end), 2) as Percent_Increase
from cte
```

Output:

Row	Percent_Increase
1	16.62

Key Points:

- The analysis reveals a 16.62% uptick in order value between 2017 and 2018 (up to August).

---

10. Calculate the Total & Average value of order price for each state.

Query:

```
select cust.customer_state as State, round(sum(pmt.payment_value),2) as Total_Value,
round(avg(pmt.payment_value),2) as Average_Value
from target.customers as cust
join target.orders as ord on cust.customer_id = ord.customer_id
join target.payments as pmt on ord.order_id = pmt.order_id
group by cust.customer_state
order by Total_Value desc, Average_Value desc ;
```

Output:

Row	State	Total_Value	Average_Value
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

Key Points:

- The analysis provides insights into both the total order value and the average order value across various states.
- Notable states in this regard include SP, RJ, MG, and others.

11. Calculate the Total & Average value of order freight for each state.

Query:

```
select cust.customer_state as State,
round(sum(ots.freight_value),2) as Total_Freight_Value,
round(avg(ots.freight_value),2) as Average_Freight_Value
from target.customers as cust
join target.orders as ord on cust.customer_id = ord.customer_id
join target.order_items as ots on ord.order_id = ots.order_id
group by cust.customer_state
order by Total_Freight_Value desc, Average_Freight_Value desc ;
```

Output:

Row	State	Total_Freight_Value	Average_Freight_Value
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

Key Points:

- The analysis provides insights into both the total freight value and the average order freight value across various states.
- Notable states in this regard include SP, RJ, MG, and others.

- **Analysis based on sales, freight and delivery time.**

12. 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.

Query:

```
select order_id,
timestamp_diff(order_delivered_customer_date,order_purchase_timestamp, day) as
time_to_deliver,
timestamp_diff(order_delivered_customer_date, order_estimated_delivery_date, day) as
diff_estimated_delivery
from target.orders
order by time_to_deliver desc
```

Output:

Row	order_id	time_to_deliver	diff_estimated_delivery
1	ca07593549f1816d26a572e06dc1eab6	209	181
2	1b3190b2dfa9d789e1f14c05b647a14a	208	188
3	440d0d17af552815d15a9e41abe49359	195	165
4	0f4519c5f1c541ddec9f21b3bddd533a	194	161
5	285ab9426d6982034523a855f55a885e	194	166
6	2fb597c2f772eca01b1f5c561bf6cc7b	194	155
7	47b40429ed8cce3aee9199792275433f	191	175
8	2fe324febf907e3ea3f2aa9650869fa5	189	167
9	2d7561026d542c8dbd8f0daeadf67a43	188	159
10	437222e3fd1b07396f1d9ba8c15fba59	187	144

Key Points:

- The analysis reveals that the actual delivery times for certain orders significantly differ from the estimated times provided by Target.
- Surprisingly, some of them deviate by more than 150 days.

13. **Find out the top 5 states with the highest & lowest average freight value.**

Query (Top 5 highest freight value):

```
select cust.customer_state, round(avg(freight_value),2) as avg_freight_value
from target.customers as cust
join target.orders as ord on cust.customer_id = ord.customer_id
join target.order_items as ots on ord.order_id = ots.order_id
group by cust.customer_state
order by avg_freight_value desc
limit 5 ;
```

Output (Top 5 highest freight value):

Row	customer_state	avg_freight_value
1	RR	42.98
2	PB	42.72
3	RO	41.07
4	AC	40.07
5	PI	39.15

Query (Lowest 5 highest freight value):

```
select cust.customer_state, round(avg(freight_value),2) as avg_freight_value
from target.customers as cust
join target.orders as ord on cust.customer_id = ord.customer_id
join target.order_items as ots on ord.order_id = ots.order_id
group by cust.customer_state
order by avg_freight_value asc
limit 5 ;
```

Output (Top 5 highest freight value):

Row	customer_state	avg_freight_value
1	SP	15.15
2	PR	20.53
3	MG	20.63
4	RJ	20.96
5	DF	21.04

Key Points:

- The analysis demonstrates that the states RR, PB, RO, AC, and PI exhibit higher average freight values which are in close proximity to the range of 39 to 43.
- And lower in states of SP, PR, MG, RJ, DF which are in close proximity to the range of 15 to 21.

---

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

Query (Top 5 highest delivery time):

```
with delivery_time as (
  select ord.order_approved_at, cust.customer_state as state,
  timestamp_diff(ord.order_delivered_customer_date,ord.order_purchase_timestamp, day)
as time_to_deliver
  from target.orders as ord
  join target.customers as cust on cust.customer_id = ord.customer_id )
select state, round(avg(time_to_deliver),2) as avg_delivery_time
from delivery_time
group by state
order by avg_delivery_time desc
limit 5 ;
```

Output (Top 5 highest delivery time):

Row	state	avg_delivery_time
1	RR	28.98
2	AP	26.73
3	AM	25.99
4	AL	24.04
5	PA	23.32

Query (Top 5 lowest delivery time):

```
with delivery_time as (
  select ord.order_approved_at, cust.customer_state as state,
  timestamp_diff(ord.order_delivered_customer_date,ord.order_purchase_timestamp, day)
as time_to_deliver
  from target.orders as ord
  join target.customers as cust on cust.customer_id = ord.customer_id )
select state, round(avg(time_to_deliver),2) as avg_delivery_time
from delivery_time
group by state
order by avg_delivery_time
limit 5 ;
```

Output (Top 5 lowest delivery time):

Row	state	avg_delivery_time
1	SP	8.3
2	PR	11.53
3	MG	11.54
4	DF	12.51
5	SC	14.48

Key Points:

- The analysis demonstrates that the states RR, AP, AM, AL, and PA exhibit higher average delivery time which is greater than 23 days usually.
- And lower in states of SP, PR, MG, DF, SC which takes around 8 to 15 days to deliver.

---

15. **Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery.**

Query:

```
with cte as (
  select cust.customer_state as state,
  timestamp_diff(order_estimated_delivery_date, order_delivered_customer_date, day) as
diff_estimated_delivery
  from target.orders as ord
  join target.customers as cust on ord.customer_id = cust.customer_id
  where order_estimated_delivery_date > order_delivered_customer_date )
select state, round(avg(diff_estimated_delivery), 2) as avg_delivery_diff
from cte
group by state
order by avg_delivery_diff desc
limit 5 ;
```

Output:

Row	state	avg_delivery_diff
1	RR	23.75
2	AP	21.87
3	AC	21.26
4	AM	20.28
5	RO	19.86

Key Points:

- The analysis shows that the actual average delivery time is generally shorter than the expected delivery time, especially in states such as RR, AP, AC, AM, and RO.

- **Analysis based on the payments:**

16. Find the month on month no. of orders placed using different payment types.

Query:

```
select extract(month from ord.order_purchase_timestamp) as month,
sum(case when pmt.payment_type = 'UPI' then 1 else 0 end) as UPI,
sum(case when pmt.payment_type = 'credit_card' then 1 else 0 end) as credit_card,
sum(case when pmt.payment_type = 'voucher' then 1 else 0 end) as voucher,
sum(case when pmt.payment_type = 'debit_card' then 1 else 0 end) as debit_card,
sum(case when pmt.payment_type = 'not_defined' then 1 else 0 end) as others
from target.orders as ord
join target.payments as pmt
on pmt.order_id = ord.order_id
group by month
order by month ;
```

Output:

Row	month	UPI	credit_card	voucher	debit_card	others
1	1	1715	6103	477	118	0
2	2	1723	6609	424	82	0
3	3	1942	7707	591	109	0
4	4	1783	7301	572	124	0
5	5	2035	8350	613	81	0
6	6	1807	7276	563	209	0
7	7	2074	7841	645	264	0
8	8	2077	8269	589	311	2
9	9	903	3286	302	43	1
10	10	1056	3778	318	54	0
11	11	1509	5897	387	70	0
12	12	1160	4378	294	64	0

Key Points:

- This illustrates that the majority of payments are made using credit cards, followed by UPI, vouchers, and debit cards.
- Additionally, there appears to be a certain pattern of seasonality in orders, with higher activity in the third quarter compared to the fourth quarter.

---

17. **Find the no. of orders placed on the basis of the payment installments that have been paid.**

Query:

```
select pmt.payment_installments, count(*) as total_orders
from target.orders as ord
join target.payments as pmt
on pmt.order_id = ord.order_id
where pmt.payment_installments >= 1
group by pmt.payment_installments
order by pmt.payment_installments ;
```

Output:

Row	payment_installment	total_orders
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

Key Points:

- This highlights the relationship between the number of orders placed and the number of payment installments that have been paid since the purchase.

Insights from the whole SQL Analysis (Target - Brazil)

- **Customer Data**: The customer data contains various columns with information about customers who have made purchases from the target. Most columns are text-based, and some may be empty or NULL for certain customers.

- **Order History:** The first order was placed in September 2016, and the most recent one occurred in October 2018, with a span of 772 days in between.

- **Top Ordering Regions:** Sao Paulo had the highest order count, followed by Rio de Janeiro, Belo Horizonte, and other areas.

- **Order Growth:** There is a noticeable increase in the number of orders, starting from 329 in 2016 and reaching 54,000 by 2018.

- **Seasonal Trends:** Order numbers tend to rise during the second and third quarters of the year but drop unexpectedly in the fourth quarter. The peak months for orders were May, July, and August, while September, October, and December saw the lowest order counts.

- **Order Timing:** Brazilian customers appear to prefer placing orders during the afternoon and fewer orders during the dawn hours.

- **Customer Distribution:** The largest customer base is in the state of São Paulo, followed by Rio de Janeiro, Minas Gerais, Rio Grande do Sul, and others.

- **Order Value Uptick**: There was a 16.62% increase in order value observed between 2017 and 2018 (up to August).

- **Order Value Analysis:** The analysis provides insights into both total order value and average order value across various states, with notable states including SP, RJ, MG, and others.

- **Freight Value Analysis:** Similarly, the analysis offers insights into total freight value and average order freight value across states, with notable states again being SP, RJ, MG, and others.

- **Delivery Time Deviations:** Some orders experienced significant deviations in actual delivery times compared to the estimated times provided by Target, with deviations of more than 150 days in some cases.

- **Freight Value by State:** States like RR, PB, RO, AC, and PI exhibit higher average freight values (in the range of 39 to 43), while states like SP, PR, MG, RJ, and DF have lower average freight values (in the range of 15 to 21).

- **Delivery Time by State:** States like RR, AP, AM, AL, and PA have longer average delivery times (greater than 23 days), whereas states like SP, PR, MG, DF, and SC have shorter delivery times (around 8 to 15 days).

- **Actual vs. Expected Delivery:** Generally, the actual average delivery time tends to be shorter than the expected delivery time, especially in states such as RR, AP, AC, AM, and RO.

- **Payment Methods:** The majority of payments are made using credit cards, followed by UPI, vouchers, and debit cards.

- **Payment Installments:** The analysis highlights the relationship between the number of orders placed and the number of payment installments that have been paid since the purchase.