

## TARGET-SQL BUSINESS CASE

---

### Context:

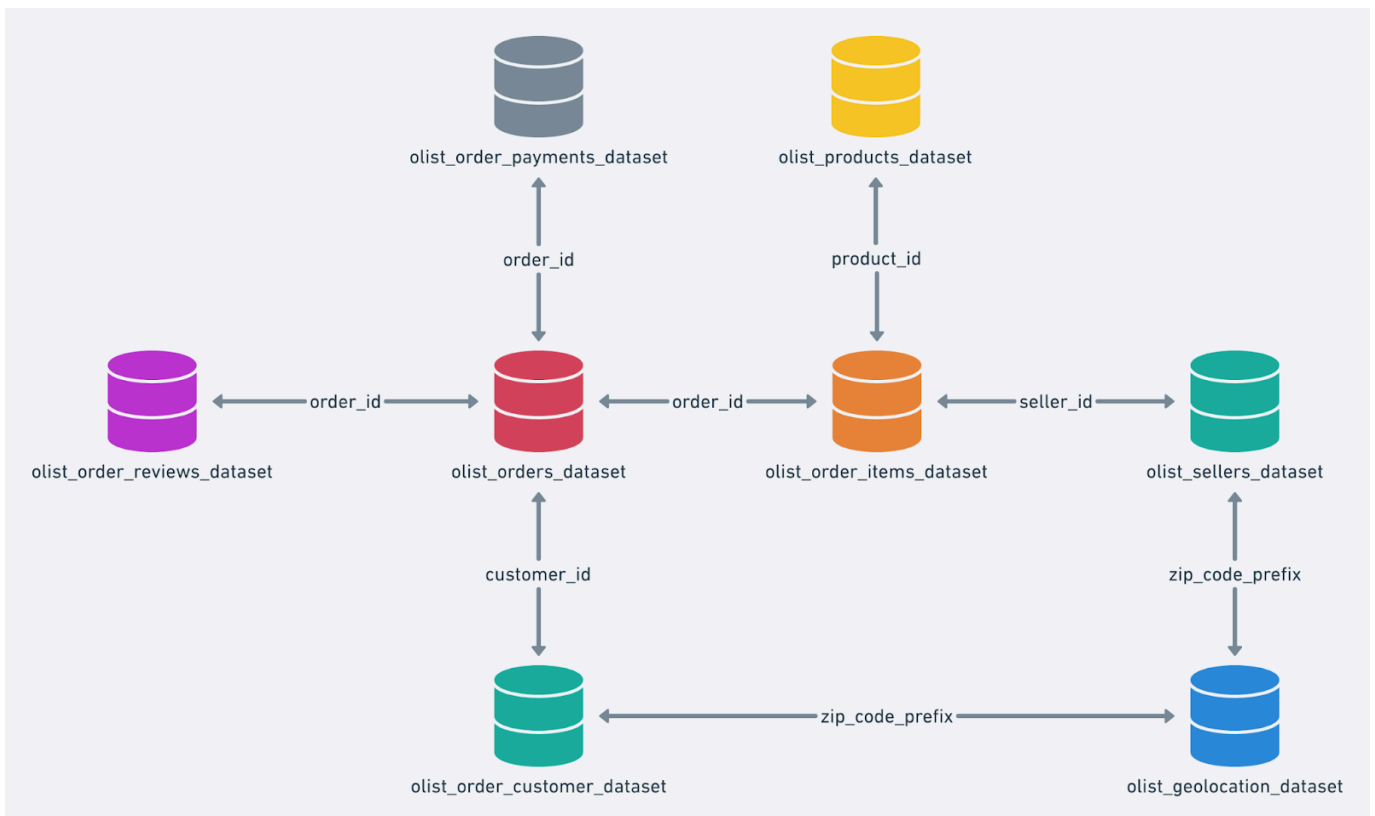
Target is a distinguished global brand and a leading retail entity in the United States. It establishes itself as the favored shopping choice by providing unparalleled value, inspiration, innovation, and an exceptional guest experience unmatched by any other retailer.

The specific focus of this business case centers on Target's operations in Brazil, offering valuable insights into 100,000 orders executed between 2016 and 2018.

### Problem Statement:

Assuming you are a data analyst/ scientist at Target, you have been assigned the task of analyzing the given dataset to extract valuable insights and provide actionable recommendations.

### Data Schema:





Observation:

All the orders were placed between 4th September 2016 and 17th October 2018.

**3. Count the Cities and states of customers who ordered during the given period.**

**Query:**

```
SELECT count(distinct customer_city) as no_of_cities,  
count(distinct customer_state) as no_of_states FROM  
`target_business_case.customers`
```

Query results				
JOB INFORMATION		RESULTS		CHART <span>PREVIEW</span> JSON
Row	no_of_cities	no_of_states		
1	4119	27		

Observation:

Target serves customers across 27 states and 4119 cities.

## Q2] In-depth Exploration:

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

Query:

```
SELECT
extract(month from order_purchase_timestamp) as month,
extract(year from order_purchase_timestamp) as year,
count(order_id) as no_of_customers
FROM `target_business_case.orders`
group by 1,2
order by 2,1
```

### Query results

JOB INFORMATION		RESULTS		CHART	PREVIEW
Row	month ▼	year ▼	no_of_orders ▼		
1	9	2016	4		
2	10	2016	324		
3	12	2016	1		
4	1	2017	800		
5	2	2017	1780		
6	3	2017	2682		
7	4	2017	2404		
8	5	2017	3700		
9	6	2017	3245		
10	7	2017	4026		

Observation:

The results show that operations started at the end of the third quarter of 2016 and since the company had to yet build trust, credibility, and brand awareness, the number of orders during the initial days was low and it started to pick up from January 2017 and continued to increase till October 2017. There was a sharp increase in orders during November 2017 which might be due to Black Friday and

Thanksgiving. Surprisingly the number of orders dropped in the month of Christmas i.e. December 2017 which may be due to people shopping for Christmas in November itself as many employees receive their 13th month salary in November itself. The orders then rose in January 2018, dipped a little in Feb 2018, and rose again in Mar 2018 which may be due to Carnaval in Brazil. Then orders dipped a little in April and then stayed linear till August 2018 after which a drastic drop was seen in September & October 2018.

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

From the results of the table above, it is difficult to analyze the monthly seasonality as the data for late 2016 and late 2018 is not enough but looking at data for 2017 and Jan 2018 - October 2018 it can be said that number of orders stays almost the same with a little peaks & dips but it peaks significantly during festive seasons like Thanksgiving, Black Friday, Carnaval, etc i.e. in the month of November and March.

## **3. 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: Morning, 13-18 hrs: Afternoon, 19-23 hrs: Night**

### **Query:**

```
SELECT
count(order_id) as no_of_orders,
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 'Morning'
when extract(HOUR from order_purchase_timestamp) between 13
and 18 then 'Evening'
when extract(HOUR from order_purchase_timestamp) between 19
and 23 then 'Night'
end as time_of_order
FROM `target_business_case.orders`
group by 2
```

order by no\_of\_orders

## Query results

			<	JOB INFORMATION	RESULTS	CHART	PDF
Row	no_of_orders	time_of_order					
1	5242	Dawn					
2	27733	Morning					
3	28331	Night					
4	38135	Evening					

Observation:

The least number of orders are placed during Dawn, Morning and Night time have almost the same number of orders and the highest number of orders are placed during the Evening time which is 38135.

### Q3] Evolution of E-commerce orders in the Brazil region:

1. Get the month-on-month no. of orders placed in each state.

Query:

```
select distinct customer_state, date_, count(order_id) over
(partition by customer_state order by date_) as
no_of_orders_per_month from
(select c.customer_state, FORMAT_DATE('%Y-%m',
DATE(EXTRACT(YEAR FROM o.order_purchase_timestamp),
EXTRACT(MONTH FROM o.order_purchase_timestamp),1)
) as date_,order_id
from `target_business_case.customers` as c inner join
`target_business_case.orders` as o
on c.customer_id = o.customer_id)t
```

JOB INFORMATION		RESULTS	CHART	PREVIEW	JS
Row	customer_state	date_	no_of_orders_per_month		
1	AC	2017-01	2		
2	AC	2017-02	5		
3	AC	2017-03	7		
4	AC	2017-04	12		
5	AC	2017-05	20		
6	AC	2017-06	24		
7	AC	2017-07	29		
8	AC	2017-08	33		
9	AC	2017-09	38		
10	AC	2017-10	44		
11	AC	2017-11	49		
12	AC	2017-12	54		
13	AC	2018-01	60		
14	AC	2018-02	63		

JOB INFORMATION		RESULTS	CHART	PREVIEW	JS
Row	customer_state	date_	no_of_orders_per_month		
15	AC	2018-03	65		
16	AC	2018-04	69		
17	AC	2018-05	71		
18	AC	2018-06	74		
19	AC	2018-07	78		
20	AC	2018-08	81		
21	AL	2016-10	2		
22	AL	2017-01	4		
23	AL	2017-02	16		
24	AL	2017-03	26		
25	AL	2017-04	49		
26	AL	2017-05	76		
27	AL	2017-06	86		
28	AL	2017-07	103		

Observation:

The above table shows the number of orders per month for each state arranged in alphabetical order.

2. How are the customers distributed across all the states?

Query:

```
select customer_state, count(customer_unique_id) as
customer_distribution
from `target_business_case.customers`
group by customer_state
order by customer_distribution desc
```

## Query results

JOB INFORMATION

RESULTS

CHART

PRE

Row	customer_state	customer_distribution
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

Observation:

On querying the dataset it is seen that the maximum number of the company's customers come from Sao Paulo followed by Rio de Janeiro. Romaria has the least number of customers.



**Q4] Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.**

**1. Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only). You can use the "payment\_value" column in the payments table to get the cost of orders.**

**Query:**

```
select *, concat(ifnull(round((safe_divide((revenue -
prev_yr_revenue),prev_yr_revenue))*100),0),'%') as
percentage_increase_in_costs
from (select *,lag(revenue,1,0) over (order by revenue) as
prev_yr_revenue
from
(select distinct extract (year from
date(o.order_purchase_timestamp)) as year,
round(sum(payment_value)) as revenue from
`target_business_case.orders` as o
inner join `target_business_case.payments` as p
on o.order_id = p.order_id
where extract (month from date(o.order_purchase_timestamp))
between 1 and 8
group by year
order by year)t)t2
order by year
```

## Query results

JOB INFORMATION		RESULTS		CHART	PREVIEW	JSON
Row	year ▼	revenue ▼	prev_yr_revenue	percentage_increase_in_costs		
1	2017	3669022.0	0.0	0%		
2	2018	8694734.0	3669022.0	137%		

**Observation:**

Compared to January 2017 - August 2017 there was an increase of 137% in the cost of orders in January 2018 - August 2018.

## 2. Calculate the Total and average value of the order price for each state.

### Query:

```
select distinct c.customer_state, round(sum(p.payment_value),2) as
total_value, round (avg(p.payment_value),2)as avg_value from
`target_business_case.customers` as c
inner join `target_business_case.orders` as o on c.customer_id =
o.customer_id
inner join `target_business_case.payments` as p on p.order_id =
o.order_id
group by 1
order by 1
```

### Query results

<	JOB INFORMATION	RESULTS	CHART	PREVIEW	JSON
Row	customer_state ▼	total_value ▼	avg_value ▼		
1	AC	19680.62	234.29		
2	AL	96962.06	227.08		
3	AM	27966.93	181.6		
4	AP	16262.8	232.33		
5	BA	616645.82	170.82		
6	CE	279464.03	199.9		
7	DF	355141.08	161.13		
8	ES	325967.55	154.71		
9	GO	350092.31	165.76		
10	MA	152523.02	198.86		
11	MG	1872257.26	154.71		

### Observation:

From the above analysis it can be said that Sao Paulo has the highest total value 5998226.96 and Pariaba has highest average value 248.33.

## 3. Calculate the Total and average value of order freight for each state.

### Query:

```

select distinct c.customer_state, round(sum(oi.freight_value),2)
as total_freight_value, round (avg(oi.freight_value),2)as
avg_freight_value from `target_business_case.customers` as c
inner join `target_business_case.orders` as o on c.customer_id =
o.customer_id
inner join `target_business_case.order_items` as oi on oi.order_id
= o.order_id
group by 1
order by 1

```

Query results				
JOB INFORMATION		RESULTS	CHART	PREVIEW
Row	customer_state	total_freight_value	avg_freight_value	
1	AC	3686.75	40.07	
2	AL	15914.59	35.84	
3	AM	5478.89	33.21	
4	AP	2788.5	34.01	
5	BA	100156.68	26.36	
6	CE	48351.59	32.71	
7	DF	50625.5	21.04	
8	ES	49764.6	22.06	
9	GO	53114.98	22.77	
10	MA	31523.77	38.26	
11	MG	270853.46	20.63	

Observation:

Sao Paulo has the highest total freight 718723.07. Roraima and Paraiba have the highest and second-highest average Freight value respectively. Higher freights can be reduced by using off times which may help the business to grow.

Q5] Analysis based on sales, freight, and delivery time.

1. 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. You can calculate the delivery time and the difference between the estimated & actual delivery date using the given formula:  
time\_to\_deliver = order\_delivered\_customer\_date - order\_purchase\_timestamp  
diff\_estimated\_delivery = order\_estimated\_delivery\_date - order\_delivered\_customer\_date

Query:

```
select order_id, extract(day from
order_delivered_customer_date - order_purchase_timestamp)
as time_to_deliver,
abs(extract (day from order_estimated_delivery_date -
order_delivered_customer_date)) as diff_estimated_delivery
from `target_business_case.orders`
where order_status = 'delivered'
```

Query results

JOB INFORMATION		RESULTS	CHART	PREVIEW	JSON
Row	order_id ▾	time_to_deliver ▾	diff_estimated_delive		
1	fd9bd6fdda953d8dd1092cb53...	9	32		
2	e21f9ce5061dd5af2d7ba4933...	13	31		
3	fb52c1aeadce7c26d4c57e429...	27	48		
4	b388b6a35d4687ab1991c81b...	5	31		
5	cbe66915eba00a69bf2b32930...	5	33		
6	5c455e9a6d56d30c1652aea99...	6	34		
7	8a3086454aae1719e85d336d...	9	34		
8	6ad06dae8c55ae30d1fcfa460...	14	32		
9	da44f554b7c4550b0d3cd650f...	15	69		
10	6589f771e13f9a378a625ea0c...	2	33		

## 2. Find out the top 5 states with the highest & lowest average freight value.

### Query:

```
(select c.customer_state, avg(oi.freight_value) as
avg_freight_value from `target_business_case.customers` as c
inner join `target_business_case.orders` as o on c.customer_id =
o.customer_id
inner join `target_business_case.order_items` as oi on o.order_id
= oi.order_id
group by 1
order by avg_freight_value desc
limit 5)
union all
(select c.customer_state, avg(oi.freight_value) as
avg_freight_value from `target_business_case.customers` as c
inner join `target_business_case.orders` as o on c.customer_id =
o.customer_id
inner join `target_business_case.order_items` as oi on o.order_id
= oi.order_id
group by 1
order by avg_freight_value asc
limit 5)
```

### Query results

< JOB INFORMATION RESULTS CHART PF		
Row	customer_state ▼	avg_freight_value ▼
1	RR	42.98442307692...
2	PB	42.72380398671...
3	RO	41.06971223021...
4	AC	40.07336956521...
5	PI	39.14797047970...
6	SP	15.14727539041...
7	PR	20.53165156794...
8	MG	20.63016680630...
9	RJ	20.96092393168...
10	DF	21.04135494596...

Observation: Romairo has the highest average freight value while Sao Paolo has the lowest average freight value

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

#### Query:

```
(select c.customer_state, round(avg(extract (day from
o.order_delivered_customer_date - o.order_purchase_timestamp)),2)
as avg_delivery_days from `target_business_case.customers` as c
join `target_business_case.orders` as o
on c.customer_id = o.customer_id
group by 1
order by avg_delivery_days desc
limit 5)
union all
(select c.customer_state, round(avg(extract (day from
o.order_delivered_customer_date - o.order_purchase_timestamp)),2)
as avg_delivery_days from `target_business_case.customers` as c
join `target_business_case.orders` as o
on c.customer_id = o.customer_id
group by 1
order by avg_delivery_days asc
limit 5)
```

#### Query results

			<	JOB INFORMATION	RESULTS	CHART	PF
Row	customer_state	avg_delivery_days					
1	RR	29.34					
2	AP	27.18					
3	AM	26.36					
4	AL	24.5					
5	PA	23.73					
6	SP	8.7					
7	PR	11.94					
8	MG	11.95					
9	DF	12.9					
10	SC	14.91					

4. Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery. You can use the difference between the averages of actual & estimated delivery date to figure out how fast the delivery was for each state.

**Query:**

```
SELECT c.customer_state,
round(AVG(extract(day from order_estimated_delivery_date -
order_delivered_customer_date)),2) AS order_delivery from
`target_business_case.orders` as o
inner join `target_business_case.customers` as c on o.customer_id
= c.customer_id
where order_status = 'delivered'
group by 1
order by order_delivery
limit 5
```

## Query results

<div><div>&lt;</div><div>JOB INFORMATION</div><div>RESULTS</div><div>CHART</div><div>PR</div></div>		
Row	customer_state ▼	order_delivery ▼
1	AL	8.71
2	MA	9.57
3	SE	10.02
4	ES	10.5
5	BA	10.79

**Observation:**

Alagoas, Maranhao, Sergipe, Espirito Santo, and Bahia are the top 5 states where the order delivery is speedy as compared to the estimated date of delivery.

## Q6] Analysis based on the payments:

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

Query:

```
select distinct p.payment_type,  
concat(format_date('%Y',o.order_purchase_timestamp),'-',format_date('%m',o.order_purchase_timestamp)) as date_,  
count(p.order_id) over (partition by payment_type,  
format_date('%m-%Y',o.order_purchase_timestamp)) as no_of_orders  
from `target_business_case.payments` as p  
inner join `target_business_case.orders` as o  
on o.order_id = p.order_id  
order by date_
```

### Query results

<div><div>&lt;</div><div>JOB INFORMATION</div><div>RESULTS</div><div>CHART</div><div>PREVI</div></div>			
Row	payment_type ▾	date_ ▾	no_of_orders ▾
1	credit_card	2016-09	3
2	credit_card	2016-10	254
3	voucher	2016-10	23
4	debit_card	2016-10	2
5	UPI	2016-10	63
6	credit_card	2016-12	1
7	debit_card	2017-01	9
8	credit_card	2017-01	583
9	UPI	2017-01	197
10	voucher	2017-01	61

Observation:

Credit cards are the most common mode of payment. UPI, Vouchers, & Debit cards are 2nd, 3rd, and 4th respectively.



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

**Query:**

```
with base as(select distinct p.payment_installments,
p.order_id
from `target_business_case.payments` as p
inner join `target_business_case.orders` as o
on o.order_id = p.order_id
where payment_installments<>0)
select distinct payment_installments, count(order_id) over
(partition by payment_installments) as no_of_orders
from base
order by payment_installments
```

### Query results

JOB INFORMATION		RESULTS	CHART
Row	payment_installments	no_of_orders	
1	1	49060	
2	2	12389	
3	3	10443	
4	4	7088	
5	5	5234	
6	6	3916	
7	7	1623	
8	8	4253	
9	9	644	
10	10	5315	

**Observation:**

The maximum number of orders come from people who have paid one installment and the number of orders goes on decreasing as the number of installments increases. This might be due to customers trying to save and pay for their installments.

**Recommendations:**

I think Target needs to work on increasing the number of customers from states that aren't contributing much to the business.

Providing special discounts in these states may help to increase the average cart value in these states.

Target should also concentrate on trying to reduce the high freight charges in certain states as this cost-cutting will help the business in the long term. They should also work on reducing delivery time in places where the delivery reaches close to the estimated date or after the estimated date. They can also craft lucrative offers/ rewards on the most common form of payment since people already buying will tend to shop extra and will come back to avail of the discounts.