About Target

- Target is one of the world's most recognized brands and one of America's leading retailers. Target makes itself a preferred shopping destination by offering outstanding value, inspiration, innovation and an exceptional guest experience that no other retailer can deliver.
- This business case has information of 100k orders from 2016 to 2018 made at Target in Brazil. Its features allows viewing an order from multiple dimensions: from order status, price, payment and freight performance to customer location, product attributes and finally reviews written by customers.

Importing the dataset and doing usual exploratory analysis steps like checking the structure & characteristics of the dataset

```
SELECT
  column_name,
  data_type
FROM
  `sqlproject-391514.Target.INFORMATION_SCHEMA.COLUMNS`
WHERE
  table_name = 'customers';
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | EXECU

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

This is the time period covered by the dataset:

```
SELECT MIN(order_purchase_timestamp) AS start_date, MAX(order_estimated_delivery_date)
AS end_date
FROM Target.orders;
```

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	start_date	end_date
1	2016-09-04 21:15:19 UTC	2018-11-12 00:00:00 UTC

Let see the cities and states of customers who placed orders during the specified time period. The following SQL query helped us identify the customer distribution:

```
SELECT
  DISTINCT c.customer_city,
  c.customer_state,
  COUNT(o.customer_id) order_count
FROM
  Target.orders o
JOIN
  Target.customers c
ON
  o.customer_id = c.customer_id
GROUP BY
  1, 2
ORDER BY
  3 DESC;
```

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	customer_city	customer_state	order_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

Here, we can see that sao paulo city from SP state alone has more orders because sao paulo is the most populous and richest state in Brazil.

Growth trend:

```
SELECT
  EXTRACT(YEAR FROM o.order_purchase_timestamp) AS year,
  EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
  ROUND(SUM(p.payment_value), 2) as revenue
FROM
  `Target.orders` o
JOIN
  `Target.payments` p ON
  o.order_id = p.order_id
GROUP BY
  year, month
```

```
ORDER BY
  year, month;
```

Row	year	month	revenue
1	2016	9	252.24
2	2016	10	59090.48
3	2016	12	19.62
4	2017	1	138488.04
5	2017	2	291908.01
6	2017	3	449863.6
7	2017	4	417788.03
8	2017	5	592918.82
9	2017	6	511276.38
10	2017	7	592382.92
11	2017	8	674396.32
12	2017	9	727762.45
13	2017	10	779677.88

We can see that there is increase in revenue growth

Seasonality peaks:

```
SELECT
  EXTRACT(MONTH FROM order_purchase_timestamp) AS month,
  COUNT(DISTINCT order_id) AS order_count
FROM
  `Target.orders`
GROUP BY
  month
ORDER BY
  month;
```

w	month	order_count
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

We can observe some seasonality in e-commerce orders. The count of orders generally increases from March to August with fluctuations in between.

Understanding Buying Patterns of Brazilian Customers:

The following SQL query was executed to categorize the order purchase timestamps into four periods: dawn, morning, afternoon, and night.

```
SELECT
  CASE
    WHEN EXTRACT(HOUR FROM o.order_purchase_timestamp) BETWEEN 0 AND 5 THEN 'Dawn'
    WHEN EXTRACT(HOUR FROM o.order_purchase_timestamp) BETWEEN 6 AND 11 THEN 'Morning'
    WHEN EXTRACT(HOUR FROM o.order_purchase_timestamp) BETWEEN 12 AND 17 THEN
'Afternoon'
    WHEN EXTRACT(HOUR FROM o.order_purchase_timestamp) BETWEEN 18 AND 23 THEN 'Night'
  END AS hour,
  COUNT(o.order_id) AS order_count
FROM
  Target.orders o
JOIN
  Target.customers c
ON o.customer_id = c.customer_id
GROUP BY
```

```
  hour
ORDER BY
  order_count DESC;
```

Row	hour	order_count
1	Afternoon	38361
2	Night	34100
3	Morning	22240
4	Dawn	4740

Based on the analysis, we found that Brazilian customers tend to place most orders during the daytime, specifically in the afternoon and night.

Analyzing Month-on-Month Orders by States:

```
SELECT
  c.customer_state,
  EXTRACT(month FROM o.order_purchase_timestamp) AS month,
  COUNT(o.order_purchase_timestamp) AS order_count
FROM
  Target.orders o
JOIN
  Target.customers c
ON
  o.customer_id = c.customer_id
GROUP BY
  c.customer_state, month
ORDER BY
  c.customer_state, month;
```

Row	customer_state	month	order_count
1	SP	8	4982
2	SP	5	4632
3	SP	7	4381
4	SP	6	4104
5	SP	3	4047
6	SP	4	3967
7	SP	2	3357
8	SP	1	3351
9	SP	11	3012
10	SP	12	2357
11	SP	10	1908
12	SP	9	1648
13	RJ	5	1321
14	RJ	8	1307
15	RJ	3	1302
16	RJ	7	1288
17	MG	3	1237
18	MG	5	1190
19	MG	8	1177
20	RJ	2	1176
21	RJ	4	1172
22	RJ	6	1128
23	MG	7	1111
24	MG	6	1080
25	MG	2	1063

It is evident that São Paulo (SP) consistently has the highest number of orders in any given month, followed by Rio de Janeiro (RJ) and Minas Gerais (MG).

Distribution of Customers Across Brazilian States:

```
SELECT
  c.customer_state,
  COUNT(c.customer_id) AS no_of_customers
FROM
  `Target.customers` c
GROUP BY
  c.customer_state
ORDER BY
```

```
  no_of_customers DESC;
```

Row	customer_state	no_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
11	PE	1652
12	CE	1336
13	PA	975

The data reveals that the state of São Paulo (SP) has the highest number of customers, which can be attributed to its status as the most populous state in Brazil. This finding also aligns with the previous analysis, indicating a positive correlation between the population of a state and its revenue.

Percentage Increase in the Cost of Orders:

```
SELECT
  EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
  (
    (
      SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2018 AND
      EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8 THEN
      p.payment_value END)
      -
      SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017 AND
      EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8 THEN
      p.payment_value END)
    )
    /
    SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017 AND
```

```
    EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8 THEN
    p.payment_value END)
  )*100 AS percent_increase
FROM
  `Target.orders` o
JOIN
  `Target.payments` p ON o.order_id = p.order_id
WHERE
  EXTRACT(YEAR FROM o.order_purchase_timestamp) IN (2017, 2018) AND
  EXTRACT(MONTH FROM o.order_purchase_timestamp) BETWEEN 1 AND 8
GROUP BY 1
ORDER BY 1;
```

Row	month	percent_increase
1	1	705.1266954171...
2	2	239.9918145445...
3	3	157.7786066709...
4	4	177.8407701149...
5	5	94.62734375677...
6	6	100.2596912456...
7	7	80.04245463390...
8	8	51.60600520477...

January shows the highest percentage increase, followed by February and April.

Calculating Days between Purchasing, Delivery, and Estimated Delivery:

```
SELECT
  order_id,
  DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY)
  AS delivered_in_days,
  DATE_DIFF(order_estimated_delivery_date, order_purchase_timestamp, DAY)
  AS estimated_delivery_in_days,
  DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date, DAY)
  AS estimated_minus_actual_delivery_days
FROM
  `Target.orders`
WHERE
  DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY) IS NOT NULL
```

```
ORDER BY
  delivered_in_days;
```

Row	order_id	delivered_in_days	estimated_delivery_in_days	estimated_minus_actual_delivery_day
1	e65f1eeee1f52024ad1dcd034...	0	10	9
2	bb5a519e352b45b714192a02f...	0	26	25
3	434cecee7d1a65fc65358a632...	0	20	19
4	d3ca7b82c922817b06e5ca211...	0	12	11
5	1d893dd7ca5f77ebf5f59f0d20...	0	10	10
6	d5fbeedc85190ba88580d6f82...	0	8	7
7	79e324907160caea526fd8b94...	0	9	8
8	38c1e3d4ed6a13cd0cf612d4c...	0	17	16
9	8339b608be0d84fca9d8da68b...	0	28	27
10	f349cdb62f69c3fae5c4d7d3f3...	0	13	12
11	f3c6775ba3d2d9fe2826f93b71...	0	12	11
12	b70a8d75313560b4acf607739...	0	10	9
13	21a8ffca665bc7a1087d31751...	0	12	11

Finding Average Time to Delivery and Average Difference in Estimated Delivery State-wise:

```
SELECT
  c.customer_state,
  ROUND(AVG(DATE_DIFF(order_delivered_customer_date, order_purchase_timestamp, DAY)),
2)
  AS avg_time_to_delivery,
  ROUND(AVG(DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date,
DAY)), 2)
  AS avg_diff_estimated_delivery
FROM
  `Target.orders` o
JOIN
  `Target.customers` c ON o.customer_id = c.customer_id
WHERE
  DATE_DIFF(order_purchase_timestamp, order_delivered_customer_date, DAY) IS NOT NULL
  AND
  DATE_DIFF(order_estimated_delivery_date, order_delivered_customer_date, DAY) IS NOT
NULL
GROUP BY
  c.customer_state
ORDER BY
  avg_time_to_delivery;
```

Row	customer_state	avg_time_to_delivery	avg_diff_estimated_delivery
1	SP	8.3	10.14
2	PR	11.53	12.36
3	MG	11.54	12.3
4	DF	12.51	11.12
5	SC	14.48	10.61
6	RS	14.82	12.98
7	RJ	14.85	10.9
8	GO	15.15	11.27
9	MS	15.19	10.17
10	ES	15.33	9.62
11	TO	17.23	11.26
12	MT	17.59	13.43
13	PE	17.97	12.4

Month over Month Count of Orders for Different Payment Types:

```
SELECT
  p.payment_type,
  EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
  COUNT(DISTINCT o.order_id) AS order_count
FROM
  `Target.orders` o
JOIN
  `Target.payments` p
ON
  o.order_id = p.order_id
GROUP BY
  1, 2
ORDER BY
  1, 2;
```

Row	payment_type	month	order_count
1	credit_card	5	8308
2	credit_card	8	8235
3	credit_card	7	7810
4	credit_card	3	7682
5	credit_card	4	7276
6	credit_card	6	7248
7	credit_card	2	6582
8	credit_card	1	6093
9	credit_card	11	5867
10	credit_card	12	4364
11	credit_card	10	3763
12	credit_card	9	3277
13	UPI	8	2077

Under payment type we have categories like credit card, debit card, voucher and UPI methods and from this we can see that credit cards have the highest orders followed by UPI.

Count of Orders Based on the Number of Payment Installments:

```
SELECT
  p.payment_installments,
  COUNT(o.order_id) AS order_count
FROM
  `Target.orders` o
JOIN
  `Target.payments` p
ON
  o.order_id = p.order_id
WHERE
  o.order_status != 'canceled'
GROUP BY
  1
ORDER BY
  2 DESC;
```

Row	payment_installment	order_count
1	1	52184
2	2	12353
3	3	10392
4	4	7056
5	10	5292
6	5	5209
7	8	4239
8	6	3898
9	7	1620
10	9	638
11	12	133
12	15	74
13	18	27

The analysis reveals that the majority of orders (maximum count) have only one payment installment. The highest number of installments is 24, which is associated with 18 orders.

Insights:

- The data reveals that the state of SP has significantly more orders than the next five states combined. This indicates an opportunity for improvement in the other states. Focusing on these states can help increase the number of orders and expand the customer base.
- Seasonal variations in sales are observed, with increased sales during festive periods. Businesses should plan their marketing and sales strategies accordingly to capitalize on these peak periods and enhance customer satisfaction, resulting in overall sales growth.
- Improving delivery times in areas with longer delivery durations can have a positive impact on customer satisfaction and encourage repeat purchases. Streamlining logistics and implementing efficient shipping processes are key to achieving this.
- States like SP and RJ already have high order counts. To further boost sales and foster brand loyalty, it is recommended to focus on customer retention strategies, such as

personalized marketing campaigns, loyalty programs, and exceptional customer service experiences.

Recommendation:

- Make deliveries faster and make customers happier by improving how we handle shipping and logistics. This means finding better ways to organize our warehouses, plan delivery routes, and work with reliable delivery services.
- Encourage customers to keep coming back and show their loyalty by creating special programs just for them. This could include rewards for referring friends, giving them personalized deals, or joining a loyalty program.
- Check our prices and shipping fees to make sure they're fair and competitive. We want to make enough money to keep the business running, but we also want to give customers a good deal.
- Use technology to make online shopping even better. We can have chatbots to help customers with questions, make our website faster and easier to use, and suggest products that customers might like based on what they've looked at before.
- Work with different sellers to offer more cool stuff and make sure everything we sell is good quality. We want to have something for everyone and make sure our customers are happy with what they buy.
- Use social media and popular influencers to show off our products and let more people know about our brand. These influencers have a lot of influence over what people buy, so we want to use that to our advantage.
- Make sure our customer service is top-notch. We'll have people available to chat with customers online and we'll answer their questions quickly and helpfully.

- Keep an eye on what our competitors are doing and adjust our strategy if we need to. We might need to lower our prices or offer new products to stay ahead of the competition. We also want to make sure our customer service is better than theirs.