BUSINESS CASE STUDY

TARGET SQL

SHILPA S
SCALER DSML OCT'22 BEGINNER BATCH
MAY 21, 2023

1.Exploratory analysis

```
SELECT *
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.sellers`
LIMIT 15
```

Row	seller_id	seller_zip_code	seller_city	seller_state
1	4be2e7f96b4fd749d52dff41f8...	69900	rio branco	AC
2	327b89b872c14d1c0be7235ef...	69005	manaus	AM
3	4221a7df464f1fe2955934e30f...	48602	bahia	BA
4	651530bf5c607240ccdd89a30...	44600	ipira	BA
5	2b402d5dc42554061f8ea98d1...	44900	irece	BA
6	d03698c2efd04a549382afa66...	45658	ilheus	BA
7	c72de06d72748d1a0dfb2125b...	46430	guanambi	BA
8	fc59392d66ef99377e50356ee...	40243	salvador	BA
9	b00af24704019bd2e1b335e70...	40130	salvador	BA

Results per page: 50 ▾ 1 – 15 of 15

```
SELECT *
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.products`
LIMIT 15
```

Row	product_id	product_category	product_name_l	product_descrip	product_photos	product_weight	product_length_	product_height_	product_width_c
1	5eb564652d...	null	null	null	null	null	null	null	null
2	09ff539a621...	babies	60	865	3	null	null	null	null
3	2f763ba79d...	electronics	45	1198	2	595	8	6	6
4	a69f15dfb80...	Watches present	53	506	6	150	11	16	6
5	e1cfc87f543...	Garden tools	39	524	4	369	26	7	7
6	106392145fc...	bed table bath	58	309	1	2083	12	2	7
7	7e33f4a1c59...	electronics	51	381	3	1075	22	5	7
8	bc9cc914f97...	HEALTH BEAUTY	55	435	1	75	14	9	7
9	5ae533eac9...	computer acces...	58	1340	1	83	12	8	8

Results per page: 50 ▾ 1 – 15 of 15

```
SELECT *
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.payments`
LIMIT 15
```

Row	order_id	payment_sequential	payment_type	payment_installments	payment_value
1	1a57108394169c0b47d8f876a...	2	credit_card	0	129.94
2	744bade1fcf9ff3f31d860ace07...	2	credit_card	0	58.69
3	8bcbe01d44d147f901cd31926...	4	voucher	1	0.0
4	fa65dad1b0e818e3ccc5cb0e3...	14	voucher	1	0.0
5	6ccb433e00daae1283ccc9561...	4	voucher	1	0.0
6	4637ca194b6387e2d538dc89b...	1	not_defined	1	0.0
7	00b1cb0320190ca0daa2c88b3...	1	not_defined	1	0.0
8	45ed6e85398a87c253db47c2d...	3	voucher	1	0.0
9	fa65dad1b0e818e3ccc5cb0e3...	13	voucher	1	0.0

Results per page: 50 ▾ 1 – 15 of 15

```
SELECT *
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.orders`
LIMIT 15
```

Row	order_id	customer_id	order_status	order_purchase_timest	order_approved_at	order_delivered_carrier_date	order_delivered_custome	order_estimated_delivery_date
1	7a4d...	725e9c75...	created	2017-11-25 11:10:3...	null	null	null	2017-12-12 00:00:00 UTC
2	35de...	4ee64f4bf...	created	2017-12-05 01:07:5...	null	null	null	2018-01-08 00:00:00 UTC
3	b535...	438449d4...	created	2017-12-05 01:07:5...	null	null	null	2018-01-11 00:00:00 UTC
4	dba5...	964a6df3...	created	2018-02-09 17:21:0...	null	null	null	2018-03-07 00:00:00 UTC
5	90ab...	7d61b9f4f...	created	2017-11-06 13:12:3...	null	null	null	2017-12-01 00:00:00 UTC
6	fa65...	9af2372a...	shipped	2017-04-20 12:45:3...	2017-04-22 09:...	2017-04-24 11:31:17 UTC	null	2017-05-18 00:00:00 UTC
7	1df2...	1240c2e6...	shipped	2017-07-13 11:03:0...	2017-07-13 11:...	2017-07-18 18:17:30 UTC	null	2017-08-14 00:00:00 UTC
8	6190...	5fc4c97d...	shipped	2017-07-11 13:36:3...	2017-07-11 13:...	2017-07-13 17:55:46 UTC	null	2017-08-14 00:00:00 UTC
9	58ce...	530d41b4...	shipped	2017-07-29 18:05:0...	2017-07-29 18:...	2017-07-31 16:41:59 UTC	null	2017-08-14 00:00:00 UTC

Results per page: 50 1 – 15 of 15

```
SELECT *
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.order_reviews``
LIMIT 15
```

Row	review_id	order_id	review_score	review_comment_title	review_creation_date	review_answer_timestamp
1	be7e29...	777c6...	1	null	0001-04-17 00:00:00 UTC	0001-04-17 07:40:00 UTC
2	e12151...	4338a...	1	null	0001-04-17 00:00:00 UTC	0001-04-17 09:04:00 UTC
3	41d614...	b8aae...	1	null	0001-04-17 00:00:00 UTC	0002-04-17 03:48:00 UTC
4	c95032...	b159d...	1	null	0001-04-17 00:00:00 UTC	0001-04-17 10:24:00 UTC
5	76823a...	2a300...	1	null	0001-04-17 00:00:00 UTC	0002-04-17 13:58:00 UTC
6	fe270df...	a39d3...	1	null	0001-04-17 00:00:00 UTC	0003-04-17 12:49:00 UTC
7	1b71e0...	0e530f...	1	null	0001-04-17 00:00:00 UTC	0010-04-17 12:45:00 UTC
8	efe402...	264c0...	1	null	0001-04-17 00:00:00 UTC	0002-04-17 01:16:00 UTC

Results per page: 50 1 – 15 of 15

```
SELECT *
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.order_items`
LIMIT 15
```

Row	order_id	order_item_id	product_id	seller_id	shipping_limit_date	price	freight_value
1	f09e36e2...	1	44d53f1240...	b64d51f0435e884e8de603b1655155ae	2018-07-09 13:31:36 UTC	3.0	12.79
2	f9ccaff72...	1	44d53f1240...	b64d51f0435e884e8de603b1655155ae	2018-08-14 14:04:44 UTC	3.0	15.23
3	c79bdf06...	1	5304ff3fa35...	cf6f6bc4df3999b9c6440f124fb2f687	2017-05-12 19:05:20 UTC	3.5	8.72
4	37193e64...	1	98224bfc1e...	ce616e1913288884e7742faac9d981db	2018-06-28 01:30:49 UTC	3.5	7.39
5	95d6357f...	1	98224bfc1e...	ce616e1913288884e7742faac9d981db	2018-06-12 19:15:14 UTC	3.5	18.23
6	95d6357f...	2	98224bfc1e...	ce616e1913288884e7742faac9d981db	2018-06-12 19:15:14 UTC	3.5	18.23
7	95d6357f...	3	98224bfc1e...	ce616e1913288884e7742faac9d981db	2018-06-12 19:15:14 UTC	3.5	18.23

Results per page: 50 1 – 15 of 15

```
SELECT *
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.geolocation`
LIMIT 15
```

Row	geolocation_zip_code_pre	geolocation_lat	geolocation_lng	geolocation_city	geolocation_state
1	49010	-10.9105145187545...	-37.052400776992329	aracaju	SE
2	49047	-10.9268145	-37.071063000000009	aracaju	SE
3	49030	-10.9701647943045...	-37.061643830745815	aracaju	SE
4	49048	-10.9401835317389...	-37.070850242714528	aracaju	SE
5	49050	-10.9271573528005...	-37.063078689600516	aracaju	SE
6	49015	-10.9233705001607...	-37.045169150380509	aracaju	SE
7	49045	-10.9304065823184...	-37.067178493623359	aracaju	SE
8	49052	-10.9229735178891...	-37.057752502914184	aracaju	SE
9	49044	-10.9920809999999...	-37.103470999999956	aracaju	SE

Results per page: 50 1 – 15 of 15

```
SELECT *
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers`
LIMIT 15
```

Row	customer_id	customer_unique_id	customer_zip_code_prefix	customer_city	customer_state
1	0735e7e4298a2ebbb46649346570476a	fcb003b1bdc0df64b4d065d9bb94f8c4	59650	acu	RN
2	903b3d86e3990db01619a4ebe3edef4e	46824822b15da44e983b021d0e945379	59650	acu	RN
3	38c97666e962d4fea7fd6a83e69f20cd	b6108acc674ae5c99e29adc1047d1049	59650	acu	RN
4	77c2f46cf580f4874c9a5751c2d88474	402cce5c0509000eed9e77fece8056e2	63430	ico	CE
5	4d3ef4cfffb8ad4767c199c36a4cfee6	6ba00666ab7eada5ceec279b259e44b5	63430	ico	CE
6	3000841b86e1fbe9493b523245d5c68d	796a0b1a21f597704057184a16ab4d71	63430	ico	CE
7	3c325415ccc7e622c66dec4bc9120030	05d1d2d9f0161c5f397ce7fc770910d4	63430	ico	CE
8	04f3a7b250e3be964f01bf22bccdc602	c34585a0276ecc5e4fb03de755e8f7d0	63430	ico	CE
9	894202b8ef01f4719a4691e79dd24c17	01a4fe5fc00bbdb0b0a4af5a5345cca5	63430	ico	CE

Results per page: 50 1 – 15 of 15

Initial exploration of dataset like checking the characteristics of data :

1.Data type of columns in the above tables include integer, string, float, timestamp.

2.Time period for which the data is given is 2016 to 2018.

```
SELECT min(order_purchase_timestamp) as min_order_date,
max(order_purchase_timestamp) as max_order_date
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.orders`
```

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

3.Cities and States of customers ordered during the given period.

```
SELECT customer_state
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers`
GROUP BY customer_state
ORDER BY customer_state asc

SELECT customer_city
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers`
GROUP BY customer_city
ORDER BY customer_city asc
```

	customer_state
1	AC
2	AL
3	AM
4	AP
5	BA
6	CE
7	DF
8	ES
9	GO

1 – 27 of 27

	customer_city
1	abadia dos dourados
2	abadiania
3	abaete
4	abaetetuba
5	abaiara
6	abaira
7	abare
8	abatia
9	abdon batista

1 – 50 of 4119

Customers are spread across 27 states and 4119 cities as per the info. from the orders table.

```
SELECT geolocation_state
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.geolocation`
GROUP BY geolocation_state
ORDER BY geolocation_state asc
```

```
SELECT geolocation_city
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.geolocation`
GROUP BY geolocation_city
```

	geolocation_state
1	AC
2	AL
3	AM
4	AP
5	BA
6	CE
7	DF
8	ES
9	GO

50 1 – 27 of 27

	geolocation_city
1	* cidade
2	...arraial do cabo
3	4o. centenario
4	4º centenario
5	abadia de goias
6	abadia dos dourados
7	abadiania
8	abadiânia
9	abaete
10	abaetetuba

50 1 – 50 of 8011

Customers are spread across 27 states and 8011 cities as per the info. from the geolocations table.

2.In-depth Exploration

1. Is there a growing trend on e-commerce in Brazil? How can we describe a complete scenario? Can we see some seasonality with peaks at specific months?

```
SELECT EXTRACT(YEAR from order_purchase_timestamp) as year_of_purchase,
COUNT(customer_id) as no_of_unique_customers
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.orders`
GROUP BY year_of_purchase
ORDER BY year_of_purchase
```

Row	year_of_purchase	no_of_unique_customers
1	2016	329
2	2017	45101
3	2018	54011

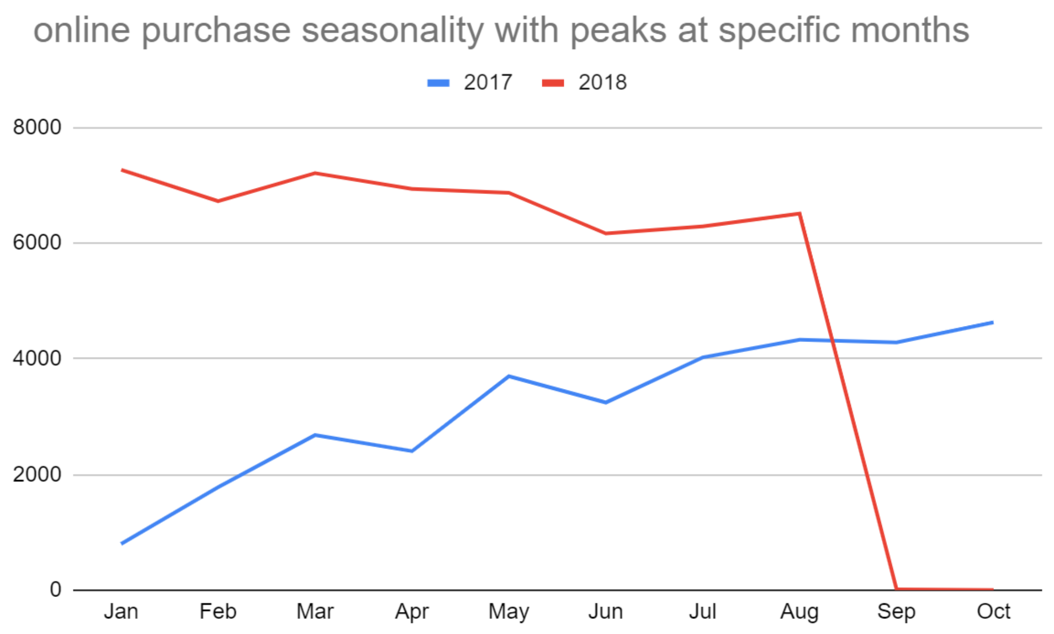
```
SELECT FORMAT_DATETIME('%b-%Y',order_purchase_timestamp) as month_yr_purchase,
COUNT(customer_id) as no_of_unique_customers
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.orders`
GROUP BY month_yr_purchase
ORDER BY month_yr_purchase asc
```

	month_yr_purchase	no_of_unique_customers
1	Apr-2017	2404
2	Apr-2018	6939
3	Aug-2017	4331
4	Aug-2018	6512
5	Dec-2016	1
6	Dec-2017	5673
7	Feb-2017	1780
8	Feb-2018	6728
9	Jan-2017	800
10	Jan-2018	7269
11	Jul-2017	4026
12	Jul-2018	6292

page: 50 1 – 25 of 25

seasonality with peaks at specific months

	2017	2018
Jan	800	7269
Feb	1780	6728
Mar	2682	7211
Apr	2404	6939
May	3700	6873
Jun	3245	6167
Jul	4026	6292
Aug	4331	6512
Sep	4285	16
Oct	4631	4



❖ A clear hike in the online purchase in June, July and Aug can be traced across 2017 and 2018.

What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

```
select time_of_purchase,
sum(no_of_unique_customers) as no_of_unique_customers
from
(SELECT EXTRACT(HOUR from order_purchase_timestamp) as hr_of_purchase,
CASE WHEN EXTRACT(HOUR from order_purchase_timestamp) BETWEEN 0 and 6
then 'Dawn'
WHEN EXTRACT(HOUR from order_purchase_timestamp)BETWEEN 6 and 12
then 'Morning'
WHEN EXTRACT(HOUR from order_purchase_timestamp) BETWEEN 12 and 18
then 'Afternoon'
else 'Night'
end as time_of_purchase,
COUNT(customer_id) as no_of_unique_customers,
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.orders`
GROUP BY 1,2
ORDER BY 1) as subquery
group by time_of_purchase
order by no_of_unique_customers desc
```

	time_of_purchase	no_of_unique_customers
1	Afternoon	38135
2	Night	28331
3	Morning	27733
4	Dawn	5242

Brazilian customers tend to buy the most in the afternoon with 38,000+ unique purchases in a year, followed by 28,000+ purchase during the night, followed by morning in the 3rd position with 27000+ unique purchases in a year and lastly during dawn with 5000+ purchases.

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

1. Get month on month orders by states.

```
SELECT FORMAT_DATETIME('%b-%Y',order_purchase_timestamp) as month_yr_purchase,
customer_state,
COUNT(o.order_id) as no_of_orders
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers` c
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON c.customer_id = o.customer_id
GROUP BY month_yr_purchase,customer_state
ORDER BY month_yr_purchase,customer_state asc
```

	month_yr_purchase	customer_state	no_of_orders
1	Apr-2017	AC	5
2	Apr-2017	AL	23
3	Apr-2017	AM	13
4	Apr-2017	BA	93
5	Apr-2017	CE	43
6	Apr-2017	DF	35
7	Apr-2017	ES	46
8	Apr-2017	GO	41
9	Apr-2017	MA	27
10	Apr-2017	MG	275
11	Apr-2017	MS	15
12	Apr-2017	MT	27

Results per page: 50 ▾ 1 – 50 of 565

.

```
SELECT FORMAT_DATETIME('%b-%Y',order_purchase_timestamp) as month_yr_purchase,
customer_state,
COUNT(o.customer_id) as no_of_unique_customers
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers` c
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON c.customer_id = o.customer_id
GROUP BY month_yr_purchase,customer_state
ORDER BY month_yr_purchase,customer_state asc
```

	month_yr_purchase	customer_state	no_of_unique_customers
1	Apr-2017	AC	5
2	Apr-2017	AL	23
3	Apr-2017	AM	13
4	Apr-2017	BA	93
5	Apr-2017	CE	43
6	Apr-2017	DF	35
7	Apr-2017	ES	46
8	Apr-2017	GO	41
9	Apr-2017	MA	27
10	Apr-2017	MG	275
11	Apr-2017	MS	15
12	Apr-2017	MT	27

Results per page: 50 ▾ 1 – 50 of 565

2. Distribution of customers across the states in Brazil

```
SELECT customer_state,
COUNT(o.customer_id) as no_of_unique_customers
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers` c
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON c.customer_id = o.customer_id
GROUP BY customer_state
ORDER BY no_of_unique_customers desc
```

	customer_state	no_of_unique_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

Results per page: 50 ▾ 1 – 27 of 27

- The state with the largest number of customers is Sao Paulo with 41,000+ unique customers, Rio De Janeiro stands second with 12,500+ customers and Minas Gerais in the 3rd position with 11,500+ customers.

4. Impact on Economy:

Analyse the money movement by e-commerce by looking at order prices, freight, and others.

4.1 Get % increase in cost of orders from 2017 to 2018 (include months between Jan to Aug only) - You can use "payment_value" column in payments table.

```
SELECT
ROUND(((payment_value_2018 - payment_value_2017)/payment_value_2017)*100,2) as percentage_increase
FROM
(select
SUM(CASE WHEN
EXTRACT(YEAR from order_purchase_timestamp)= 2017
THEN p.payment_value
ELSE 0
END) as payment_value_2017,
SUM(CASE WHEN
EXTRACT(YEAR from order_purchase_timestamp)= 2018
THEN p.payment_value
ELSE 0
END) as payment_value_2018
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.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 )
```

	percentage_increase
1	136.98

- The total payment value of orders showed a zen percentage plus increase in 2018 when compared to 2017. It shows a 137% hike in 2018.E-commerce has influenced the money movements in the country positively.

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

```
SELECT
SUM(price) as total_order_value,
SUM(freight_value) as total_freight_value,
AVG(price) as mean_order_value,
AVG(freight_value) as mean_freight_value
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers` c
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON c.customer_id = o.customer_id
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.order_items` oi
ON oi.order_id = o.order_id
```

Row	total_order_value	total_freight_value	mean_order_value	mean_freight_value
1	13591643.70001...	2251909.540000...	120.6537390146...	19.99031992898...

- The mean order value and mean freight value amounts to 121 and 20 Brazilian real respectively.

```
SELECT customer_state,
SUM(price) as total_order_value,
SUM(freight_value) as total_freight_value,
AVG(price) as mean_order_value,
```

```
AVG(freight_value) as mean_freight_value
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers` c
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON c.customer_id = o.customer_id
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.order_items` oi
ON oi.order_id = o.order_id
GROUP BY customer_state
ORDER BY mean_order_value desc,mean_freight_value desc
```

	customer_state	total_order_value	total_freight_value	mean_order_value	mean_freight_value
1	PB	115268.0800000...	25719.72999999...	191.4752159468...	42.72380398671...
2	AL	80314.81000000...	15914.58999999...	180.8892117117...	35.84367117117...
3	AC	15982.94999999...	3686.749999999...	173.7277173913...	40.07336956521...
4	RO	46140.64000000...	11417.38	165.9735251798...	41.06971223021...
5	PA	178947.8099999...	38699.30000000...	165.6924166666...	35.83268518518...
6	AP	13474.29999999...	2788.500000000...	164.3207317073...	34.00609756097...
7	PI	86914.07999999...	21218.20000000...	160.3580811808...	39.14797047970...
8	TO	49621.74000000...	11732.67999999...	157.5293333333...	37.24660317460...
9	RN	83034.97999999...	18860.09999999...	156.9659357277...	35.65236294896...
10	CE	227254.7099999...	48351.58999999...	153.7582611637...	32.71420162381...
11	SE	58920.85000000...	14111.46999999...	153.0411688311...	36.65316883116...
12	RR	7829.429999999...	2235.190000000...	150.5659615384...	42.98442307692...

Results per page: 50 1 – 27 of 27

- When it comes to the statistics of each state in Brazil, 191 and 43 Brazilian real stands as the highest mean order value and mean freight value respectively.

5. Analysis on sales, freight, and delivery time

5.1 Calculate days between purchasing, delivering and estimated delivery

```
#checking for null values if any before applying formula.
select order_purchase_timestamp, order_delivered_carrier_date,order_estimated_delivery_date
from `dsml-scaler-sql1.Business_Case_Target_SQL.orders`o
WHERE order_purchase_timestamp is null or order_delivered_carrier_date is null or order_estimated_delivery_date is null
#Found some 1783 null values are present in order_delivered_carrier_date
#No null values in order_purchase_timestamp and order_estimated_delivery_date


#diff_btwn_delivering_v/s_purchasing =order_delivered_customer_date - order_purchase_timestamp
#diff_btwn_estimated_delivery_v/s_purchase= order_estimated_delivery_date - order_purchase_timestamp
#diff_btwn_estimated_delivery_v/s_delivery = order_estimated_delivery_date - order_delivered_customer_date



#diff_btwn_delivering_vs_purchasing =order_delivered_customer_date - order_purchase_timestamp
select order_delivered_customer_date,order_purchase_timestamp,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as diff_btwn_delivering_vs_purchasing
from `dsml-scaler-sql1.Business_Case_Target_SQL.orders`o
ORDER BY diff_btwn_delivering_vs_purchasing  desc
```

	order_delivered_customer_date	order_purchase_timestamp	diff_btwn_delivering_vs_purchasing
1	2017-09-19 14:36:39 UTC	2017-02-21 23:31:27 UTC	209
2	2018-09-19 23:24:07 UTC	2018-02-23 14:57:35 UTC	208
3	2017-09-19 15:12:50 UTC	2017-03-07 23:59:51 UTC	195
4	2017-09-19 14:38:21 UTC	2017-03-09 13:26:57 UTC	194
5	2017-09-19 14:00:04 UTC	2017-03-08 22:47:40 UTC	194
6	2017-09-19 14:33:17 UTC	2017-03-08 18:09:02 UTC	194
7	2018-07-13 20:51:31 UTC	2018-01-03 09:44:01 UTC	191
8	2017-09-19 17:00:07 UTC	2017-03-13 20:17:10 UTC	189
9	2017-09-19 14:38:18 UTC	2017-03-15 11:24:27 UTC	188
10	2017-09-19 16:28:58 UTC	2017-03-16 11:36:00 UTC	187
11	2017-09-19 17:14:25 UTC	2017-03-15 23:23:17 UTC	187
12	2017-09-19 18:13:19 UTC	2017-03-17 12:32:22 UTC	186

Results per page: 50 1 – 50 of 99441

```
#diff_btwn_estimated_delivery_vs_purchase= order_estimated_delivery_date - order_purchase_timestamp
select order_estimated_delivery_date,order_purchase_timestamp,
DATE_DIFF(order_estimated_delivery_date,order_purchase_timestamp,DAY) as diff_btwn_estimated_delivery_vs_purchase
from `dsml-scaler-sql1.Business_Case_Target_SQL.orders`o
ORDER BY diff_btwn_estimated_delivery_vs_purchase desc
```

	order_estimated_delivery_date	order_purchase_timestamp	diff_btwn_estimated_delivery_vs_purchase
1	2018-07-12 00:00:00 UTC	2018-02-06 20:44:56 UTC	155
2	2018-08-03 00:00:00 UTC	2018-03-06 09:47:07 UTC	149
3	2017-07-04 00:00:00 UTC	2017-02-07 18:01:15 UTC	146
4	2017-08-08 00:00:00 UTC	2017-03-16 02:30:51 UTC	144
5	2017-08-22 00:00:00 UTC	2017-03-30 15:23:23 UTC	144
6	2017-08-04 00:00:00 UTC	2017-03-14 19:23:22 UTC	142
7	2017-10-11 00:00:00 UTC	2017-05-23 22:28:36 UTC	140
8	2018-01-30 00:00:00 UTC	2017-10-05 21:39:05 UTC	116
9	2017-10-19 00:00:00 UTC	2017-07-01 15:47:06 UTC	109
10	2018-06-13 00:00:00 UTC	2018-02-26 00:11:02 UTC	106

Results per page: 50 1 – 50 of 99441

```
#diff_estimated_delivery_vs_delivery = order_estimated_delivery_date - order_delivered_customer_date
select order_estimated_delivery_date,order_delivered_customer_date,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY) as diff_btwn_estimated_delivery_vs_delivery
from `dsml-scaler-sql1.Business_Case_Target_SQL.orders`o
ORDER BY diff_btwn_estimated_delivery_vs_delivery desc
```

	order_estimated_delivery_date	order_delivered_customer_date	diff_btwn_estimated_delivery_vs_delivery
1	2018-08-03 00:00:00 UTC	2018-03-09 23:36:47 UTC	146
2	2017-07-04 00:00:00 UTC	2017-02-14 14:27:45 UTC	139
3	2018-07-12 00:00:00 UTC	2018-02-27 16:35:43 UTC	134
4	2017-10-11 00:00:00 UTC	2017-06-09 13:35:54 UTC	123
5	2018-01-30 00:00:00 UTC	2017-10-13 13:49:07 UTC	108
6	2018-03-22 00:00:00 UTC	2017-12-28 22:18:23 UTC	83
7	2018-04-25 00:00:00 UTC	2018-02-01 01:52:34 UTC	82
8	2018-04-27 00:00:00 UTC	2018-02-08 19:21:39 UTC	77
9	2018-01-11 00:00:00 UTC	2017-10-25 20:22:10 UTC	77
10	2018-02-16 00:00:00 UTC	2017-11-30 16:12:50 UTC	77
11	2018-04-09 00:00:00 UTC	2018-01-22 22:58:10 UTC	76
12	2017-01-11 00:00:00 UTC	2016-10-27 10:58:35 UTC	75

Results per page: 50 1 – 50 of 99441

❖ As per the analysis the highest difference between purchase and estimated delivery stands at nearly 210 days, difference between estimated delivery and purchase at 155 days and difference between estimated delivery and delivery at positive 146 days meaning an early product delivery of 146 days.

**Instead of order_delivered_carrier_date have used order_delivered_customer_date in all the above 3 formulas because order_delivered_customer_date is the delivery date from customers perspective. Also order_delivered_carrier_date contains many null values. It needs to be filtered out if order_delivered_carrier_date needs to be considered.

5.2 Find time_to_delivery & diff_estimated_delivery. Formula for the same given below:

- time_to_delivery = order_delivered_customer_date-order_purchase_timestamp

```
select order_delivered_customer_date,order_purchase_timestamp,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as time_to_delivery
from `dsml-scaler-sql1.Business_Case_Target_SQL.orders`o
ORDER BY time_to_delivery desc
```

	order_delivered_customer_date	order_purchase_timestamp	time_to_delivery
1	2017-09-19 14:36:39 UTC	2017-02-21 23:31:27 UTC	209
2	2018-09-19 23:24:07 UTC	2018-02-23 14:57:35 UTC	208
3	2017-09-19 15:12:50 UTC	2017-03-07 23:59:51 UTC	195
4	2017-09-19 14:38:21 UTC	2017-03-09 13:26:57 UTC	194
5	2017-09-19 14:00:04 UTC	2017-03-08 22:47:40 UTC	194
6	2017-09-19 14:33:17 UTC	2017-03-08 18:09:02 UTC	194
7	2018-07-13 20:51:31 UTC	2018-01-03 09:44:01 UTC	191
8	2017-09-19 17:00:07 UTC	2017-03-13 20:17:10 UTC	189
9	2017-09-19 14:38:18 UTC	2017-03-15 11:24:27 UTC	188
10	2017-09-19 16:28:58 UTC	2017-03-16 11:36:00 UTC	187
11	2017-09-19 17:14:25 UTC	2017-03-15 23:23:17 UTC	187
12	[illegible]	[illegible]	[illegible]

Results per page: 50 1 – 50 of 99441

- diff_estimated_delivery = order_estimated_delivery_date - order_delivered_customer_date

```
select order_estimated_delivery_date,order_delivered_customer_date,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY) as diff_estimated_delivery
from `dsml-scaler-sql1.Business_Case_Target_SQL.orders`o
ORDER BY diff_estimated_delivery desc
```

	order_estimated_delivery_date	order_delivered_customer_date	diff_estimated_delivery
1	2018-08-03 00:00:00 UTC	2018-03-09 23:36:47 UTC	146
2	2017-07-04 00:00:00 UTC	2017-02-14 14:27:45 UTC	139
3	2018-07-12 00:00:00 UTC	2018-02-27 16:35:43 UTC	134
4	2017-10-11 00:00:00 UTC	2017-06-09 13:35:54 UTC	123
5	2018-01-30 00:00:00 UTC	2017-10-13 13:49:07 UTC	108
6	2018-03-22 00:00:00 UTC	2017-12-28 22:18:23 UTC	83
7	2018-04-25 00:00:00 UTC	2018-02-01 01:52:34 UTC	82
8	2018-04-27 00:00:00 UTC	2018-02-08 19:21:39 UTC	77
9	2018-01-11 00:00:00 UTC	2017-10-25 20:22:10 UTC	77
10	2018-02-16 00:00:00 UTC	2017-11-30 16:12:50 UTC	77
11	2018-04-09 00:00:00 UTC	2018-01-22 22:58:10 UTC	76
12	2017-01-11 00:00:00 UTC	2016-10-27 10:58:35 UTC	75

Results per page: 50 ▾ 1 – 50 of 99441

❖ The highest expected time for product delivery is 209 days and the highest difference in estimated delivery comes to a positive 146 days meaning an early product delivery of 146 days.

5.3 Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery

```
SELECT subq.customer_state,
AVG(freight_value) as mean_freight_value,
AVG(time_to_delivery) as mean_time_to_delivery,
AVG(diff_estimated_delivery) as mean_diff_estimated_delivery
from
(select customer_state,freight_value,order_delivered_customer_date,order_purchase_timestamp,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY) as diff_estimated_delivery
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers` c
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON c.customer_id = o.customer_id
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.order_items` oi
ON oi.order_id = o.order_id
ORDER BY time_to_delivery desc) as subq
GROUP BY customer_state
```

	customer_state	mean_freight_value	mean_time_to_delivery	mean_diff_estimated_delivery
1	MT	28.1662843601...	17.508196721311...	13.639344262295094
2	MA	38.2570024271...	21.203750000000...	9.1099999999999923
3	AL	35.8436711711...	23.992974238875...	7.9765807962529349
4	SP	15.1472753904...	8.25960855241909	10.26559438451439
5	MG	20.6301668063...	11.515522180072...	12.397151041263502
6	PE	32.9178626799...	17.792096219931...	12.552119129438733
7	RJ	20.9609239316...	14.689382157500...	11.14449314293797
8	DF	21.0413549459...	12.501486199575...	11.274734607218704
9	RS	21.7358043303...	14.708299364095...	13.203000163052323
10	SE	36.6531688311...	20.978666666666...	9.1653333333333276
11	PR	20.5316515679...	11.480793060718...	12.533899805275263
12	PA	35.8326851851...	23.301707779886...	13.37476280834913

Results per page: 50 1 – 27 of 27

5.4 Sort the data to get the following:

5.5 Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

```
#cte version
with cte as
(SELECT subq.customer_state,
ROUND(AVG(freight_value),0) as mean_freight_value,
ROUND(AVG(time_to_delivery),0) as mean_time_to_delivery,
ROUND(AVG(diff_estimated_delivery),0) as mean_diff_estimated_delivery
from
(select customer_state,freight_value,order_delivered_customer_date,order_purchase_timestamp,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY) as diff_estimated_delivery
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers` c
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON c.customer_id = o.customer_id
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.order_items` oi
ON oi.order_id = o.order_id
ORDER BY time_to_delivery desc) as subq
GROUP BY customer_state)

select * from cte
ORDER BY mean_freight_value desc
LIMIT 5
```

Row	customer_state	mean_freight_value	mean_time_to_delivery	mean_diff_estimated_delivery
1	PB	43.0	20.0	12.0
2	RR	43.0	28.0	17.0
3	RO	41.0	19.0	19.0
4	AC	40.0	20.0	20.0
5	PI	39.0	19.0	11.0

- ❖ Paraiba (PB) and Roraima (RR) emerges as the states with highest mean freight value of 43 Brazilian real.

5.6 Top 5 states with highest/lowest average time to delivery

```
with cte as
(SELECT subq.customer_state,
ROUND(AVG(freight_value)) as mean_freight_value,
ROUND(AVG(time_to_delivery)) as mean_time_to_delivery,
ROUND(AVG(diff_estimated_delivery)) as mean_diff_estimated_delivery
from
(select customer_state,freight_value,order_delivered_customer_date,order_purchase_timestamp,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY) as diff_estimated_delivery
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers` c
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON c.customer_id = o.customer_id
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.order_items` oi
ON oi.order_id = o.order_id
ORDER BY time_to_delivery desc) as subq
GROUP BY customer_state)

select * from cte
ORDER BY mean_time_to_delivery desc
LIMIT 5
```

Row	customer_state	mean_freight_value	mean_time_to_delivery	mean_diff_estimated_delivery
1	AP	34.0	28.0	17.0
2	RR	43.0	28.0	17.0
3	AM	33.0	26.0	19.0
4	AL	36.0	24.0	8.0
5	PA	36.0	23.0	13.0

- Amapa (AP) and Roraima (RR) emerges as the states with highest mean time to delivery i.e., 28 days.

5.7 Top 5 states where delivery is really fast/ not so fast compared to estimated date

```
#cte version
with cte as
(SELECT subq.customer_state,
ROUND(AVG(freight_value)) as mean_freight_value,
ROUND(AVG(time_to_delivery)) as mean_time_to_delivery,
ROUND(AVG(diff_estimated_delivery)) as mean_diff_estimated_delivery
from
(select customer_state,freight_value,order_delivered_customer_date,order_purchase_timestamp,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) as time_to_delivery,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY) as diff_estimated_delivery
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.customers` c
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON c.customer_id = o.customer_id
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.order_items` oi
ON oi.order_id = o.order_id
ORDER BY time_to_delivery desc) as subq
GROUP BY customer_state)

select * from cte
ORDER BY mean_diff_estimated_delivery desc
LIMIT 5
```

Row	customer_state	mean_freight_value	mean_time_to_delivery	mean_diff_estimated_delivery
1	AC	40.0	20.0	20.0
2	RO	41.0	19.0	19.0
3	AM	33.0	26.0	19.0
4	RR	43.0	28.0	17.0
5	AP	34.0	28.0	17.0

- Acre (AC) and Rondonia (RO) are the top 2 states with highest difference in estimated delivery of a positive 20 and 19 days meaning an early delivery of 20 and 19 days.

6. Payment type analysis:

6.1 Month over Month count of orders for different payment types

```
select *
from
(SELECT FORMAT_DATETIME('%b-%Y',order_purchase_timestamp) as month_yr_purchase,
payment_type,
COUNT(payment_value) as count_payment_value
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.payments` p
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON p.order_id = o.order_id
GROUP BY FORMAT_DATETIME('%b-%Y',order_purchase_timestamp),payment_type) as subq
ORDER BY month_yr_purchase desc
```

	month_yr_purchase	payment_type	count_payment_value
1	Sep-2018	not_defined	1
2	Sep-2018	voucher	15
3	Sep-2017	credit_card	3283
4	Sep-2017	voucher	287
5	Sep-2017	UPI	903
6	Sep-2017	debit_card	43
7	Sep-2016	credit_card	3
8	Oct-2018	voucher	4
9	Oct-2017	voucher	291
10	Oct-2017	credit_card	3524
11	Oct-2017	UPI	993

Results per page: 50 1 – 50 of 90

```
#Year wise count of orders grouped by payment type
select *
from
(SELECT EXTRACT(YEAR from order_purchase_timestamp) as year_of_purchase,
payment_type,
COUNT(payment_value) as count_payment_value
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.payments` p
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON p.order_id = o.order_id
GROUP BY EXTRACT(YEAR from order_purchase_timestamp),payment_type) as subq
ORDER BY year_of_purchase desc
```

	year_of_purchase	payment_type	count_payment_value
1	2018	credit_card	41969
2	2018	voucher	2725
3	2018	not_defined	3
4	2018	debit_card	1105
5	2018	UPI	10213
6	2017	voucher	3027
7	2017	credit_card	34568
8	2017	UPI	9508
9	2017	debit_card	422
10	2016	credit_card	258
11	2016	voucher	23
12	2016	debit_card	2

Results per page: 50 ▾ 1 – 13 of 13 |<

❖ Credit card showed the highest hike in the number of transactions from nearly 250 in 2016 to 34,500+ in 2017 and reaching nearly 42000 in 2018. UPI holds the second highest number of transactions with 10000+ transactions in 2018, nearly 9500 transactions in 2017 f rom just 63 transactions in 2016.

6.2 Count of orders based on the no. of payment instalments.

```
select *
from(
SELECT EXTRACT(YEAR from order_purchase_timestamp) as year_of_purchase,
payment_installments,
COUNT(payment_installments) as count_payment_installments
FROM `dsml-scaler-sql1.Business_Case_Target_SQL.payments` p
INNER JOIN `dsml-scaler-sql1.Business_Case_Target_SQL.orders` o
ON p.order_id = o.order_id
GROUP BY EXTRACT(YEAR from order_purchase_timestamp),payment_installments)
order by year_of_purchase desc
```

w	year_of_purchase	payment_installment	count_payment_installments
1	2018	0	2
2	2018	1	29138
3	2018	2	7020
4	2018	3	5507
5	2018	4	3697
6	2018	5	2654
7	2018	6	2020
8	2018	7	765
9	2018	8	2545
10	2018	9	262
11	2018	10	2248
12	2018	11	9

2018		2017		2016	
payment_installments	count_payment_installments	payment_installments	count_payment_installments	payment_installments	count_payment_installments
0	2				
1	29138	1	23262	1	146
2	7020	2	5362	2	31
3	5507	3	4910	3	44
4	3697	4	3375	4	26
5	2654	5	2565	5	20
6	2020	6	1882	6	18
7	765	7	848	7	13
8	2545	8	1720	8	3
9	262	9	379	9	3
10	2248	10	3038	10	42
11	9	11	14		
12	58	12	75		
13	9	13	7		
14	6	14	9		
15	48	15	26		
16	3	16	2		
17	5	17	3		
18	11	18	16		
20	5	20	12		
23	1	21	3		
24	2	22	1		
		24	16		

- ❖ The jump of count of customers who opted payment instalments from 146 in 2016 to 23,000+ in 2017 to 29,000+ in 2018 shows their movement from reluctance towards opting payment instalments to acceptance of the same.
- ❖ The enhanced number of customers opting for 10 payment instalments over 2017 and 2018 shows their readiness to opt till 10 payment instalments. The declining count beyond 10 payment instalments shows their reluctance towards higher count of payment instalments.

7. Actionable Insights

a) A clear hike in the online purchase in the months of June, July and Aug can be traced across 2017 and 2018.

a) Customers of Target in Brazil are widespread across 27 states and 4119 cities as per the info. from the orders table.
b) A clear growing trend on e-commerce in Brazil can be seen from the data from the enhancement in the number of customers from 329 in 2016 to 45,000+ in 2017 to 54,000 in 2018.
c) Brazilian customers tend to buy the most in the afternoon with 38,000+ unique purchases in a year, followed by 28,000+ purchase during the night, followed by morning in the 3rd position with 27000+ unique purchases in a year and lastly during dawn with 5000+ purchases.
d) The state with the largest number of customers is Sao Paulo (SP) with 41,000+ unique customers, Rio De Janeiro (RJ) stands second with 12,500+ customers and Minas Gerais (MG) in the 3rd position with 11,500+ customers.
e) The total payment value of orders showed a zen percentage plus increase in 2018 when compared to 2017. It shows a 137% hike in 2018.E-commerce has influenced the money movements in the country positively.
f) The mean order value and mean freight value amounts to 121 and 20 Brazilian real respectively.
g) When it comes to the statistics of each state in Brazil, 191 and 43 Brazilian real stands as the highest mean order value and mean freight value respectively.

h) As per the analysis the highest difference between purchase and estimated delivery stands at nearly 210 days, difference between estimated delivery and purchase at 155 days and difference between estimated delivery and delivery at positive 146 days meaning an early product delivery of 146 days.
i) The highest expected time for product delivery is 209 days and the highest difference in estimated delivery comes to a positive 146 days meaning an early product delivery of 146 days which is a huge achievement.
j) Paraiba (PB) and Roraima (RR) emerged as the states with highest mean freight value of 43 Brazilian real.
k) Amapa (AP) and Roraima (RR) emerged as the states with highest mean time to delivery i.e., 28 days.
l) Acre (AC) and Rondonia (RO) are the top 2 states with highest difference in estimated delivery of a positive 20 and 19 days meaning an early delivery of 20 and 19 days.
m) Credit card showed the highest hike in the number of transactions from nearly 250 in 2016 to 34,500+ in 2017 and reaching nearly 42000 in 2018. UPI holds the second highest number of transactions with 10000+ transactions in 2018, nearly 9500 transactions in 2017 from just 63 transactions in 2016.
n) The jump of count of customers who opted payment instalments from 146 in 2016 to 23,000+ in 2017 to 29,000+ in 2018 shows their movement from reluctance towards opting payment instalments to acceptance of the same.
o) The enhanced number of customers opting for 10 payment instalments over 2017 and 2018 shows their readiness to opt till 10 payment instalments. The declining count beyond 10 payment instalments shows their reluctance towards higher count of payment instalments.

8. Recommendations

a) Any roll out of offers for customers is recommended to be planned in the months of June, July, August as there is a hike in purchase during these months over the last 2 years - 2017 and 2018.
b) Since customers of Target are already widespread across the states, now priority should be on maintaining the customers along with business expansion.
c) Any roll out of offers for customers is recommended to be given during afternoon followed by night-time looking at the increased number of customers making purchase during those timings of the day.
d) The enhancing number of E-commerce transactions can be positively used to influence the money movements in the country.
e) The highest difference in estimated delivery comes to a positive 146 days meaning an early product delivery of 146 days which is a huge achievement which is a good criterion to maintain. In fact, the estimation of the same should be calculated much more accurately.
f) But when it comes to the statistics of each state, Acre (AC) and Rondonia (RO) are the top 2 states with highest difference in estimated delivery of a positive 20 and 19 days meaning an early delivery of 20 and 19 days which is still a good number.
g) Chances are there that credit card transactions will see a hike with any further customer-friendly credit-card related purchase offers.
h) Customers must be subjected for a study to learn about their thought process with respect to opting higher number of payment instalments.