

# BUSINESS CASE STUDY PROJECT

## 1.1. Data type of all columns in the "customers" table.

```
SELECT * FROM `scaler-dsml-sql-387615.CASE_STUDY.customers`
```

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	EXECUTION GRAPH	
Row	customer_id	customer_unique_id	customer_zip_code	customer_city	customer_state	
1	0735e7e4298a2ebbb46649346...	fc0003b1bdc0df64b4d065d9b...	59650	acu	RN	
2	903b3d8e3990db01619a4ebe...	46824822b15da44e983b021d...	59650	acu	RN	
3	38c97666e962d4fea7fd6a83e...	b6108acc674ae5c99e29adc10...	59650	acu	RN	
4	77c2f46cf580f4874c9a5751c2...	402cce5c0509000eed9e77fec...	63430	ico	CE	
5	4d3ef4cffffb8ad4767c199c36a...	6ba00666ab7ead5ceec279b2...	63430	ico	CE	
6	3000841b86e1fbe9493b52324...	796a0b1a21f597704057184a1...	63430	ico	CE	
7	3c325415ccc7e622c66dec4bc...	05d1d2d9f0161c5f397ce7fc77...	63430	ico	CE	
8	04f3a7b250e3be964f01bf22bc...	c34585a0276ecc5e4fb03de75...	63430	ico	CE	
9	894202b8ef01f4719a4691e79...	01a4fe5fc00bbdb0b0a4af5a53...	63430	ico	CE	
10	9d715b9fb75a9d081c14126c0...	8f399f3b7ace8e6245422c9e1f...	63430	ico	CE	

## 1.2. Get the time range between which the orders were placed.

```
SELECT MIN(order_purchase_timestamp) AS min_timestamp,  
       MAX(order_purchase_timestamp) AS max_timestamp  
FROM `scaler-dsml-sql-387615.CASE_STUDY.orders`;
```

JOB INFORMATION		RESULTS	JSON	EXECUTION DETAILS	EXECUTION GRAPH
Row	min_timestamp	max_timestamp			
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC			

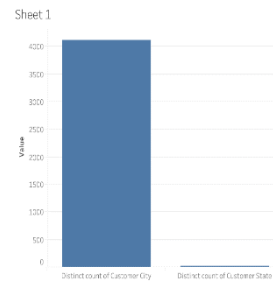
## 1.3. Count the Cities & States of customers who ordered during the given period.

```
WITH date_range AS (  
  SELECT  
    DATE(MIN(order_purchase_timestamp)) AS start_date, DATE(MAX(order_purchase_timestamp)) AS  
    end_date  
  FROM `scaler-dsml-sql-387615.CASE_STUDY.orders`  
)
```

```
SELECT COUNT(DISTINCT c.customer_city) AS city_count,  
       COUNT(DISTINCT c.customer_state) AS state_count  
FROM `scaler-dsml-sql-387615.CASE_STUDY.customers` c
```

```
JOIN `scaler-dsml-sql-387615.CASE_STUDY.orders` o ON c.customer_id = o.customer_id
CROSS JOIN date_range
WHERE DATE(o.order_purchase_timestamp) BETWEEN
date_range.start_date AND date_range.end_date;
```

Row	city_count	state_count
1	4119	27



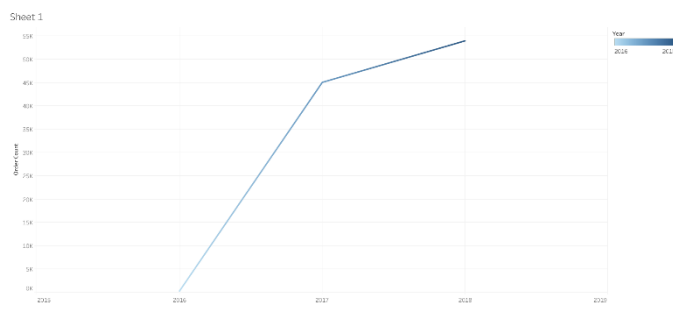
[https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/3A\\_16884481616310/Sheet1?publish=yes](https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/3A_16884481616310/Sheet1?publish=yes)

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

**Yes, there a growing trend in the no. of orders placed over the past years**

```
SELECT EXTRACT(YEAR FROM order_purchase_timestamp) AS year,
COUNT(*) AS order_count
FROM `scaler-dsml-sql-387615.CASE_STUDY.orders`
GROUP BY year
ORDER BY year;
```

Row	year	order_count
1	2016	329
2	2017	45101
3	2018	54011



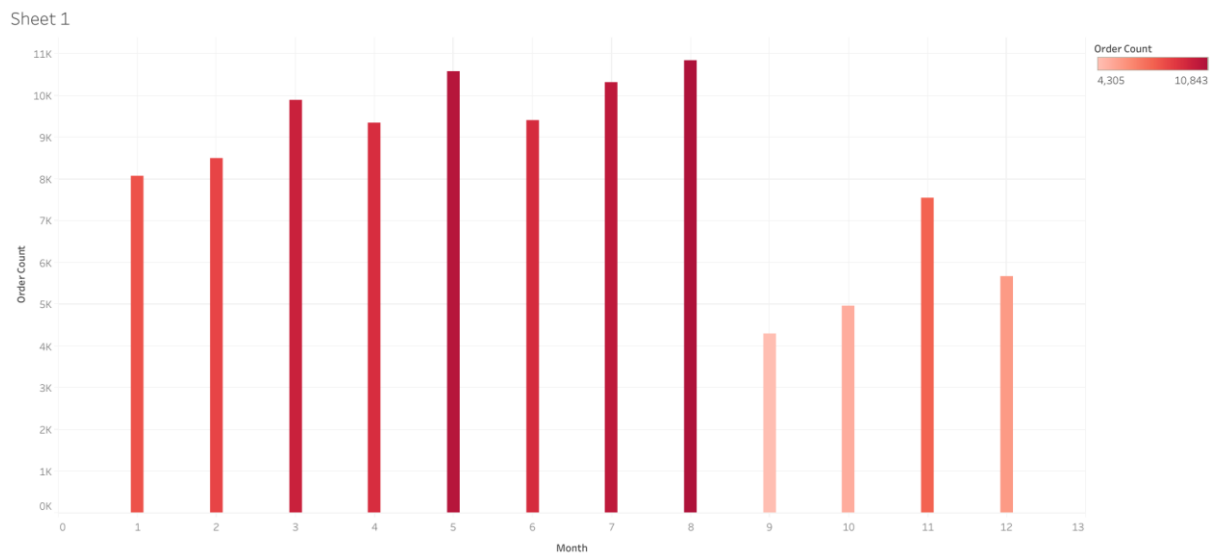
<https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Is there a growing trend in the no. of orders placed over the past years/Sheet1?publish=yes>

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

```
SELECT EXTRACT(YEAR FROM order_purchase_timestamp) AS year,  
       EXTRACT(MONTH FROM order_purchase_timestamp) AS month,  
       COUNT(*) AS order_count  
FROM `scaler-dsml-sql-387615.CASE_STUDY.orders`  
GROUP BY year, month  
ORDER BY year, month;
```

Row	year	month	order_count
1	2016	9	4
2	2016	10	324
3	2016	12	1
4	2017	1	800
5	2017	2	1780
6	2017	3	2682
7	2017	4	2404
8	2017	5	3700
9	2017	6	3245
10	2017	7	4026
11	2017	8	4331

[https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Canweesomesomethkindofmonthlyseasonalityintermsoftheno\\_ofordersbeingplaced/Sheet1?publish=yes](https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Canweesomesomethkindofmonthlyseasonalityintermsoftheno_ofordersbeingplaced/Sheet1?publish=yes)



## 2.3 During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)

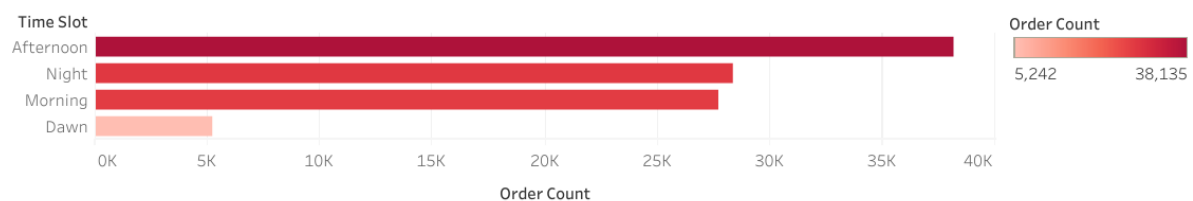
- 0-6 hrs: Dawn
- 7-12 hrs: Mornings
- 13-18 hrs: Afternoon
- 19-23 hrs: Night

```
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 'Morning'
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'
END AS time_slot,
COUNT(*) AS order_count
FROM `scaler-dsml-sql-387615.CASE_STUDY.orders`
GROUP BY time_slot;
```

Row	time_slot	order_count
1	Morning	27733
2	Dawn	5242
3	Afternoon	38135
4	Night	28331

**In Afternoon 38135 orders, the Brazilian customers mostly place their orders**

Sheet 1



<https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/DuringwhattimeofthedaydotheBraziliancustomersmostlyplacetheirordersDawnMorningAfternoonorNight/Sheet1?publish=yes>

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

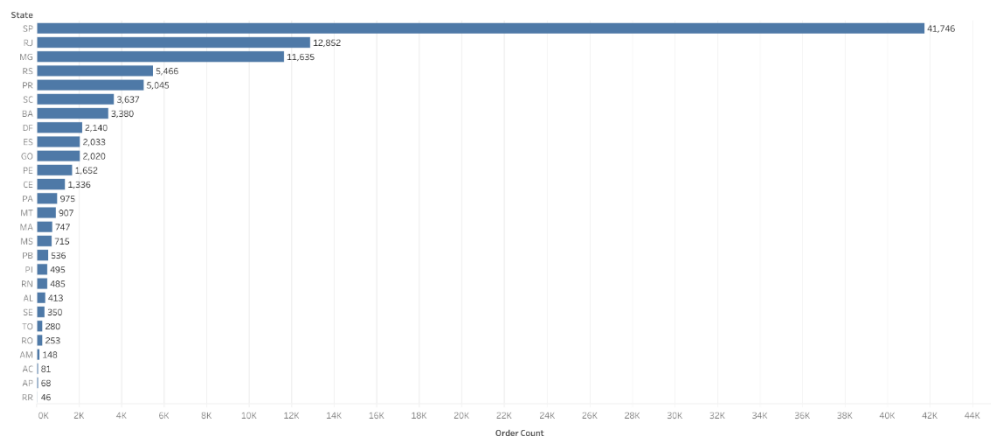
```

SELECT EXTRACT(MONTH FROM o.order_purchase_timestamp) AS month,
       c.customer_state AS state,
       COUNT(*) AS order_count
FROM `scaler-dsml-sql-387615.CASE_STUDY.orders` o
JOIN `scaler-dsml-sql-387615.CASE_STUDY.customers` c ON o.customer_id = c.customer_id
GROUP BY month, state
ORDER BY month, state;

```

Row	month	state	order_count
1	1	AC	8
2	1	AL	39
3	1	AM	12
4	1	AP	11
5	1	BA	264
6	1	CE	99
7	1	DF	151
8	1	ES	159
9	1	GO	164
10	1	MA	66
11	1	MG	971

Sheet 1



[https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Getthemonthonmonthno\\_ofordersplacedineachstate/Sheet1?publish=yes](https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Getthemonthonmonthno_ofordersplacedineachstate/Sheet1?publish=yes)

### 3.2 How are the customers distributed across all the states?

```

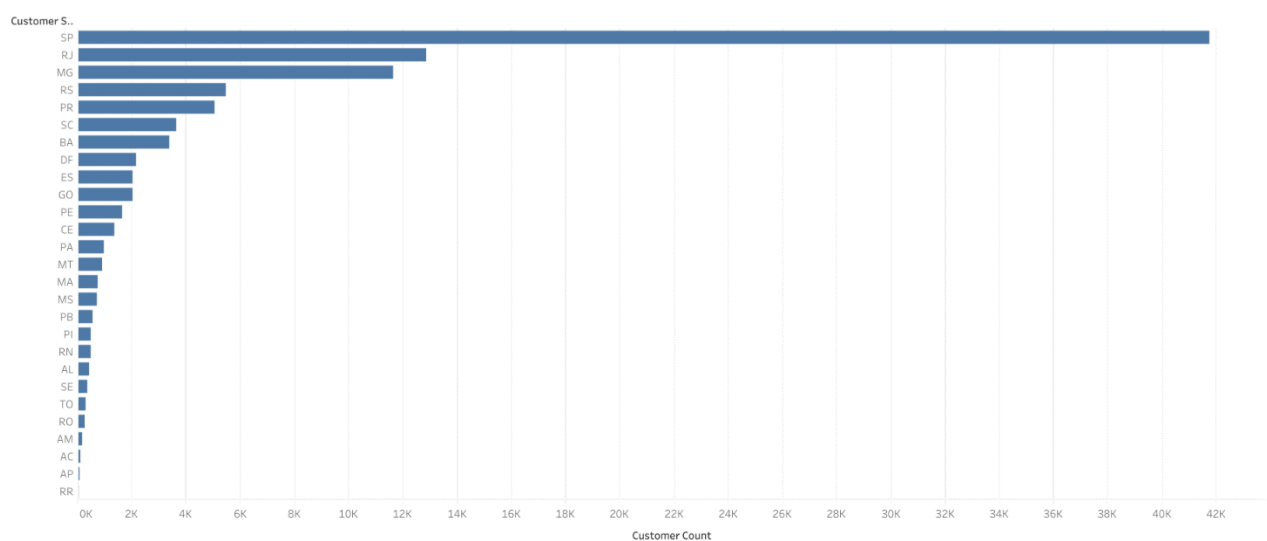
SELECT customer_state, COUNT(*) as customer_count
FROM `scaler-dsml-sql-387615.CASE_STUDY.customers`

```

GROUP BY customer\_state  
ORDER BY customer\_count DESC;

Row	customer_state	customer_count
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

Sheet 1



<https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Howarethecustomersdistributedacrossallthestates/Sheet1?publish=yes>

**4.1 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.

Row	percentage_increase
1	136.9768716466...

```

SELECT
  (SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2018 AND EXTRACT(MONTH
FROM o.order_purchase_timestamp) <= 8 THEN p.payment_value ELSE 0 END)
  - SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017 THEN p.payment_value
ELSE 0 END))
  / SUM(CASE WHEN EXTRACT(YEAR FROM o.order_purchase_timestamp) = 2017 THEN p.payment_value
ELSE 0 END) * 100
  AS percentage_increase
FROM
  `scaler-dsml-sql-387615.CASE_STUDY.orders` o
  JOIN `scaler-dsml-sql-387615.CASE_STUDY.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;

```

## 4.2 Calculate the Total & Average value of order price for each state?

```

SELECT
  customer_state,
  SUM(oi.price) AS total_order_price,
  AVG(oi.price) AS average_order_price
FROM

```

`scaler-dsml-sql-387615.CASE\_STUDY.orders` o

JOIN `refined-sum-390315.BUSINESS\_CASE.order\_items` oi ON o.order\_id = oi.order\_id

JOIN `scaler-dsml-sql-387615.CASE\_STUDY.customers` c ON o.customer\_id = c.customer\_id

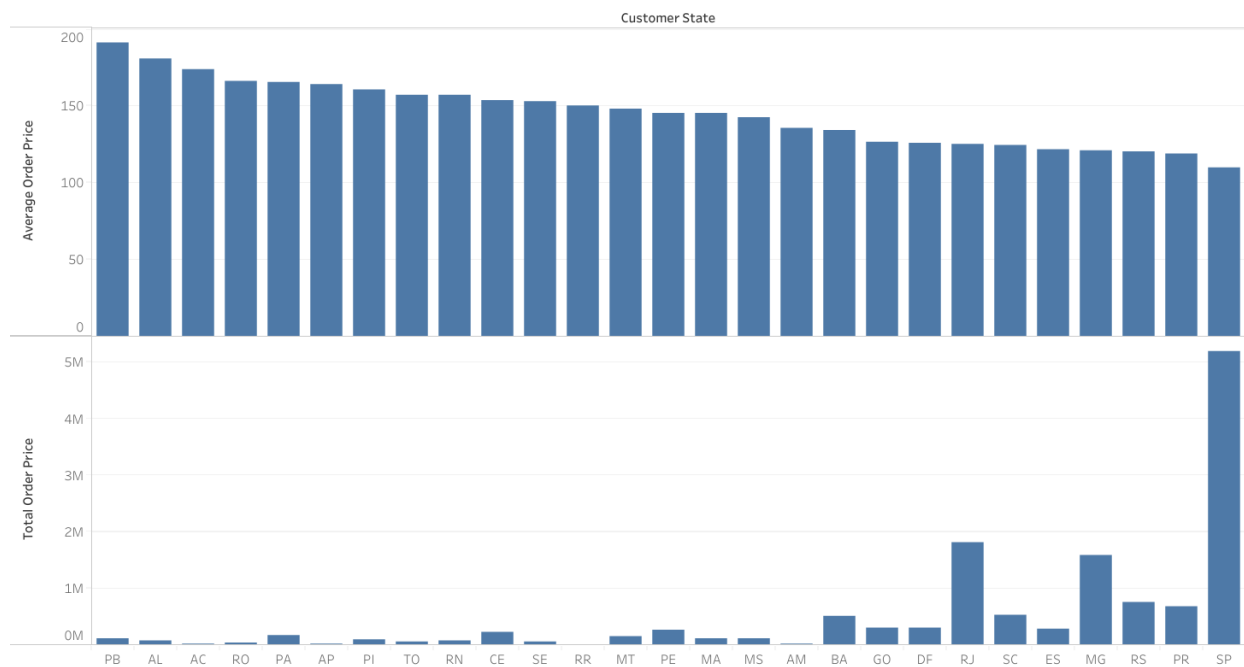
GROUP BY

customer\_state;

Row	customer_state ▼	total_order_price ▼	average_order_price
1	MT	156453.5299999...	148.2971848341...
2	MA	119648.2199999...	145.2041504854...
3	AL	80314.81	180.8892117117...
4	SP	5202955.050001...	109.6536291597...
5	MG	1585308.029999...	120.7485741488...
6	PE	262788.0299999...	145.5083222591...
7	RJ	1824092.669999...	125.1178180945...
8	DF	302603.9399999...	125.7705486284...
9	RS	750304.0200000...	120.3374530874...
10	SE	58920.85000000...	153.0411688311...
11	PR	683083.7600000...	119.0041393728...



Sheet 1



<https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/CalculateTheTotalAverageValueOfOrderPriceForEachState/Sheet1?publish=yes>

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

SELECT

customer\_state,

SUM(oi.freight\_value) AS total\_order\_freight,

AVG(oi.freight\_value) AS average\_order\_freight

FROM

`scaler-dsml-sql-387615.CASE\_STUDY.orders` o

JOIN `refined-sum-390315.BUSINESS\_CASE.order\_items` oi ON o.order\_id = oi.order\_id

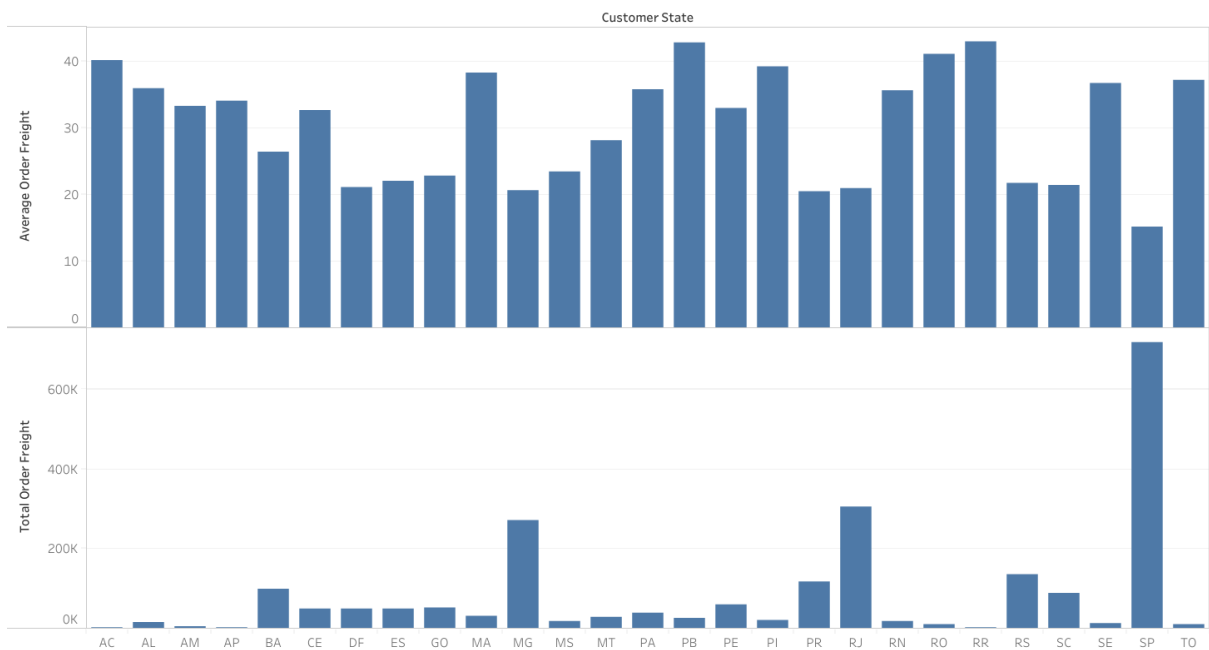
JOIN `scaler-dsml-sql-387615.CASE\_STUDY.customers` c ON o.customer\_id = c.customer\_id

GROUP BY

customer\_state;

Row	customer_state	total_order_freight	average_order_freigh
1	MT	29715.43000000...	28.16628436018...
2	MA	31523.77000000...	38.25700242718...
3	AL	15914.58999999...	35.84367117117...
4	SP	718723.0699999...	15.14727539041...
5	MG	270853.4600000...	20.63016680630...
6	PE	59449.65999999...	32.91786267995...
7	RJ	305589.3100000...	20.96092393168...
8	DF	50625.49999999...	21.04135494596...
9	RS	135522.7400000...	21.73580433039...
10	SE	14111.46999999...	36.65316883116...
11	PR	117851.6800000...	20.53165156794...

Sheet 1



<https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/CalculatetheTotalAveragevalueoforderfreightforeachstate/Sheet1?publish=yes>

**5.1** 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

**SELECT**

o.order\_id,

**DATE\_DIFF**(o.order\_delivered\_customer\_date, o.order\_purchase\_timestamp, DAY) **AS** delivery\_time,

**DATE\_DIFF**(o.order\_estimated\_delivery\_date, o.order\_delivered\_customer\_date, DAY) **AS**

diff\_estimated\_delivery

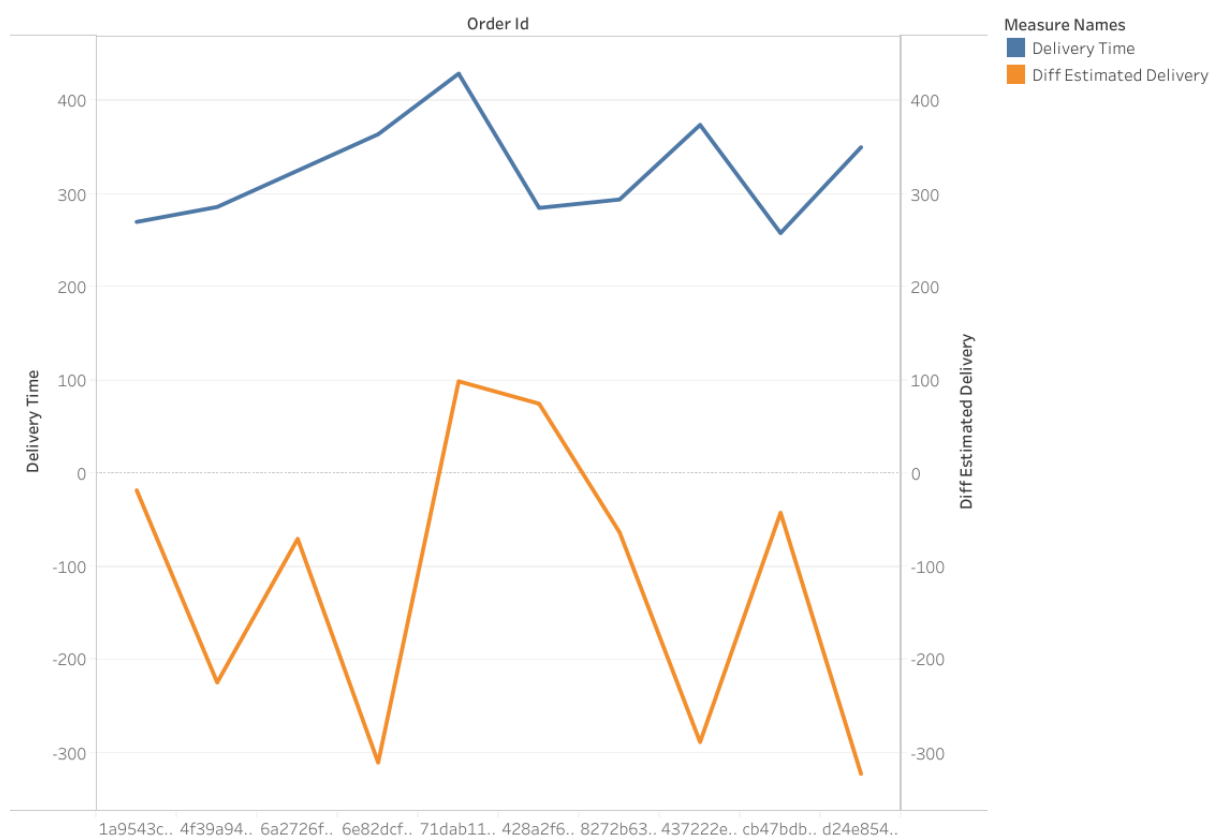
**FROM**

`scaler-dsml-sql-387615.CASE\_STUDY.orders` o

**JOIN** `refined-sum-390315.BUSINESS\_CASE.order\_items` oi **ON** o.order\_id = oi.order\_id

Row	order_id	delivery_time	diff_estimated_delivery
1	1950d777989f6a877539f5379...	30	-12
2	2c45c33d2f9cb8ff8b1c86cc28...	30	28
3	65d1e226dfaeb8cdc42f66542...	35	16
4	635c894d068ac37e6e03dc54e...	30	1
5	3b97562c3aee8bdedcb5c2e45...	32	0
6	3b97562c3aee8bdedcb5c2e45...	32	0
7	68f47f50f04c4cb6774570cfde...	29	1
8	276e9ec344d3bf029ff83a161c...	43	-4
9	54e1a3c2b97fb0809da548a59...	40	-4
10	fd04fa4105ee8045f6a0139ca5...	37	-1
11	302bb8109d097a9fc6e9cefc5...	33	-5
12	66057d37308e787052a32828...	38	-6
13	19135c945c554eebfd7576c73...	36	-2

## Sheet 1



<https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/DeliverytimevsEstimatedTimeDifference/Sheet1?publish=yes>

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

```
(
SELECT
    customer_state,
    AVG(freight_value) AS avg_freight_value
FROM
    `scaler-dsml-sql-387615.CASE_STUDY.customers` c
    JOIN `scaler-dsml-sql-387615.CASE_STUDY.orders` o ON c.customer_id = o.customer_id
    JOIN `refined-sum-390315.BUSINESS_CASE.order_items` oi ON o.order_id = oi.order_id
GROUP BY
    customer_state
ORDER BY
    avg_freight_value DESC
LIMIT 5
)
UNION ALL
(
SELECT
    customer_state,
    AVG(freight_value) AS avg_freight_value
```

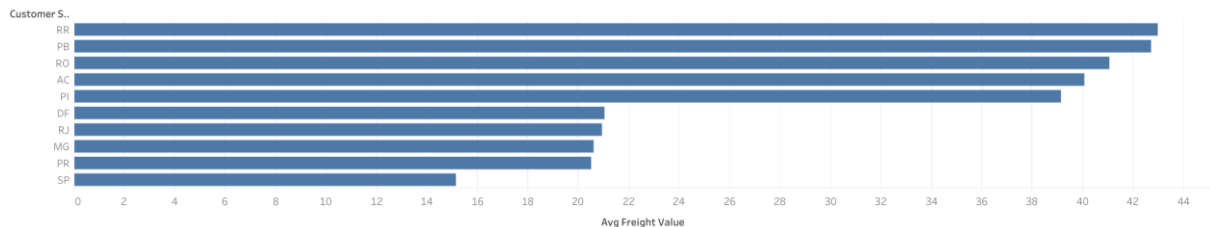
```

FROM
  `scaler-dsml-sql-387615.CASE_STUDY.customers` c
  JOIN `scaler-dsml-sql-387615.CASE_STUDY.orders` o ON c.customer_id = o.customer_id
  JOIN `refined-sum-390315.BUSINESS_CASE.order_items` oi ON o.order_id = oi.order_id
GROUP BY
  customer_state
ORDER BY
  avg_freight_value ASC
LIMIT 5
)

```

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...

Sheet 1



<https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Findoutthetop5stateswiththehighestlowestaveragefreightvalue/Sheet1?publish=yes>

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

```

(
SELECT
  c.customer_state,

```

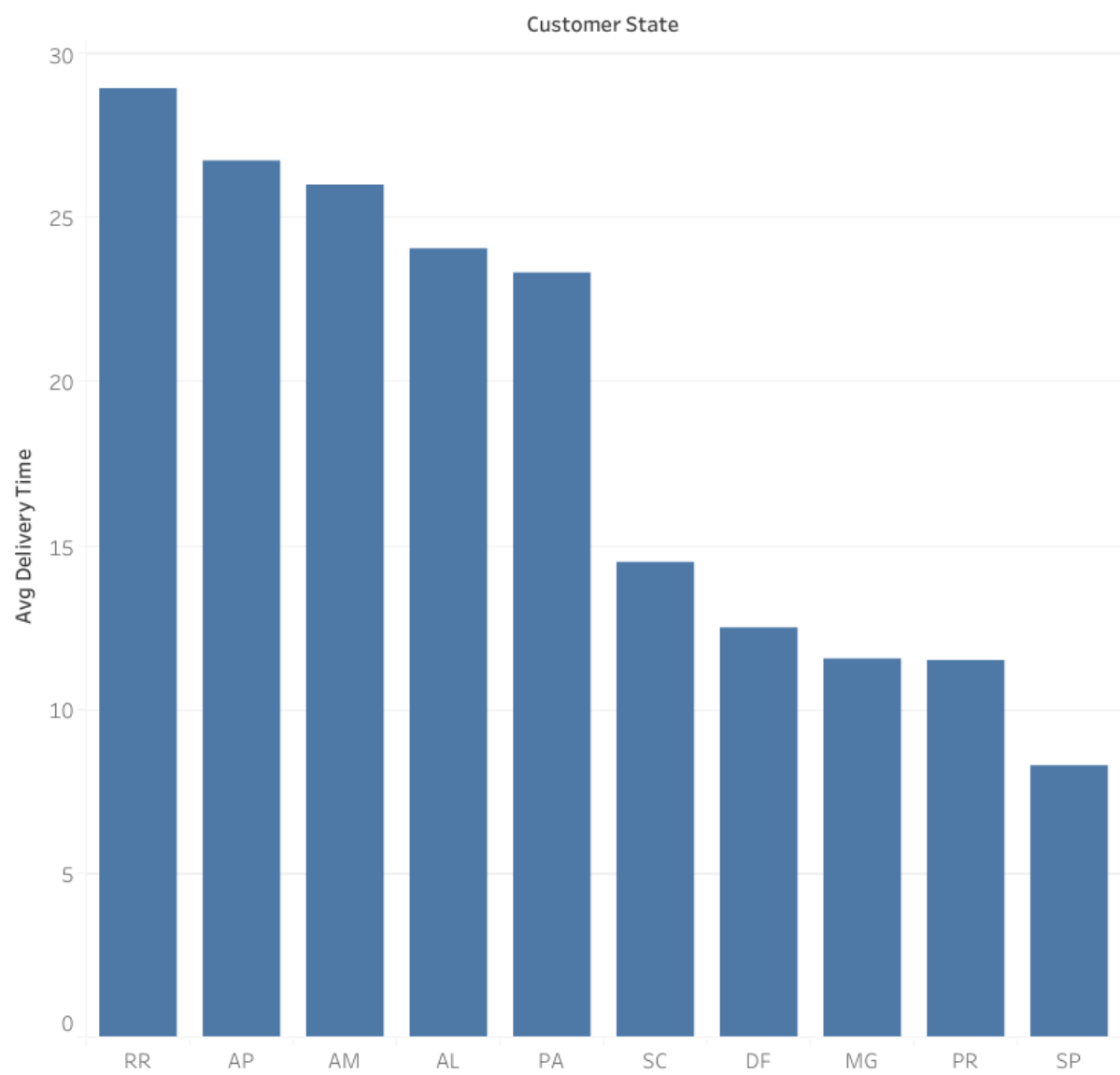
```

        AVG(date_diff(o.order_delivered_customer_date, o.order_purchase_timestamp, DAY)) AS
avg_delivery_time
FROM
    `scaler-dsml-sql-387615.CASE_STUDY.orders` o
JOIN
    `scaler-dsml-sql-387615.CASE_STUDY.customers` c ON o.customer_id = c.customer_id
GROUP BY
    c.customer_state
ORDER BY
    avg_delivery_time DESC
LIMIT 5
)
UNION ALL
(
    SELECT
        c.customer_state,
        AVG(date_diff(o.order_delivered_customer_date, o.order_purchase_timestamp, DAY)) AS
avg_delivery_time
FROM
    `scaler-dsml-sql-387615.CASE_STUDY.orders` o
JOIN
    `scaler-dsml-sql-387615.CASE_STUDY.customers` c ON o.customer_id = c.customer_id
GROUP BY
    c.customer_state
ORDER BY
    avg_delivery_time ASC
LIMIT 5
)

```

Row	customer_state	avg_delivery_time
1	RR	28.97560975609...
2	AP	26.73134328358...
3	AM	25.98620689655...
4	AL	24.04030226700...
5	PA	23.31606765327...
6	SP	8.298061489072...
7	PR	11.52671135486...
8	MG	11.54381329810...
9	DF	12.50913461538...
10	SC	14.47956019171...

Sheet 1



<https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Findoutthetop5stateswiththehighestlowestaveragedeliverytime/Sheet1?publish=yes>

**5.4. 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.**

SELECT

c.customer\_state,

AVG(date\_diff(o.order\_delivered\_customer\_date, o.order\_estimated\_delivery\_date, DAY)) AS

avg\_delivery\_difference

FROM

`scaler-dsml-sql-387615.CASE\_STUDY.orders` o

JOIN

`scaler-dsml-sql-387615.CASE\_STUDY.customers` c ON o.customer\_id = c.customer\_id

GROUP BY

c.customer\_state

HAVING

avg\_delivery\_difference < 0

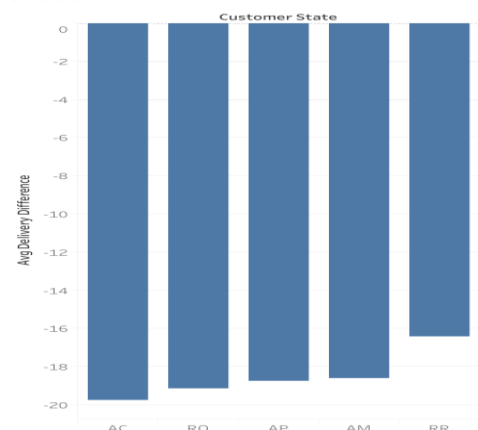
ORDER BY

avg\_delivery\_difference ASC

LIMIT 5

Row	customer_state ▼	avg_delivery_difference
1	AC	-19.762500000...
2	RO	-19.1316872427...
3	AP	-18.7313432835...
4	AM	-18.6068965517...
5	RR	-16.4146341463...

Sheet 1



[https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Findoutthetop5stateswheretheorderdeliveryisreallyfastascomparedtotheestimateddateofdelivery\\_/Sheet1?publish=yes](https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Findoutthetop5stateswheretheorderdeliveryisreallyfastascomparedtotheestimateddateofdelivery_/Sheet1?publish=yes)

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

SELECT



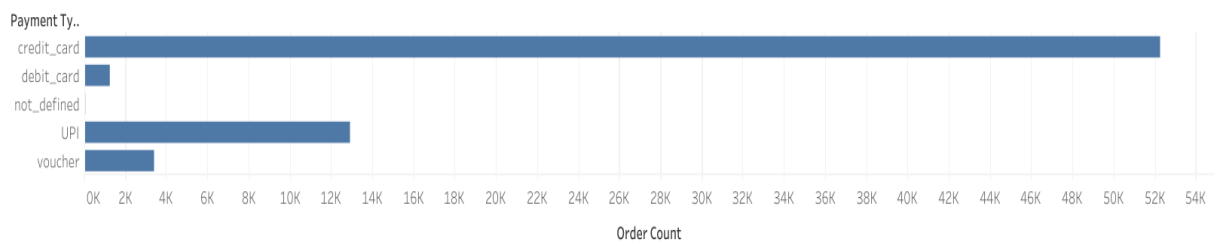
```

DATE_TRUNC(o.order_purchase_timestamp, MONTH) AS order_month,
p.payment_type,
COUNT(o.order_id) AS order_count
FROM
`scaler-dsml-sql-387615.CASE_STUDY.orders` o
JOIN
`scaler-dsml-sql-387615.CASE_STUDY.payments` p ON o.order_id = p.order_id
GROUP BY
order_month,
p.payment_type
ORDER BY
order_month

```

Row	order_month	payment_type	order_count
1	2016-09-01 00:00:00 UTC	credit_card	3
2	2016-10-01 00:00:00 UTC	credit_card	254
3	2016-10-01 00:00:00 UTC	voucher	23
4	2016-10-01 00:00:00 UTC	debit_card	2
5	2016-10-01 00:00:00 UTC	UPI	63
6	2016-12-01 00:00:00 UTC	credit_card	1
7	2017-01-01 00:00:00 UTC	voucher	61
8	2017-01-01 00:00:00 UTC	UPI	197
9	2017-01-01 00:00:00 UTC	credit_card	583
10	2017-01-01 00:00:00 UTC	debit_card	9
11	2017-02-01 00:00:00 UTC	credit_card	1356

Sheet 1



<https://public.tableau.com/app/profile/lanka.ajay.kumar/viz/Findthemonthonmonthnoofordersplacedusingdifferentpaymenttypes/Sheet1?publish=yes>

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

```

SELECT

```

```

payment_installments,
COUNT(order_id) AS order_count
FROM
`scler-dsml-sql-387615.CASE_STUDY.payments`
GROUP BY
payment_installments

```

Row	payment_installment	order_count
1	0	2
2	1	52546
3	2	12413
4	3	10461
5	4	7098
6	5	5239
7	6	3920
8	7	1626
9	8	4268
10	9	644
11	10	5328