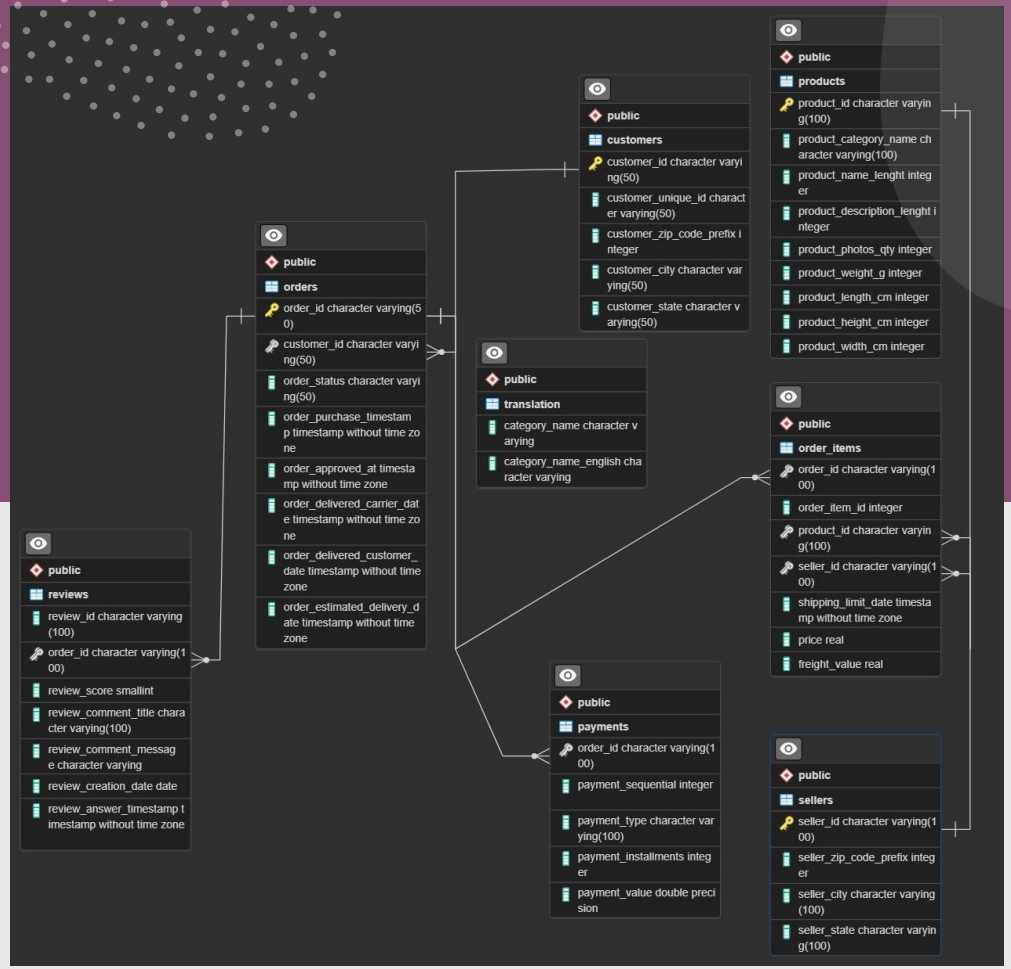
# End-to-End SQL Project:

Brazilian E-Commerce Analysis

# ERD Diagram:


public
orders
order_id character varying(50)
customer_id character varying(50)
order_status character varying(50)
order_purchase_timestamp timestamp without time zone
order_approved_at timestamp without time zone
order_delivered_carrier_date timestamp without time zone
order_delivered_customer_date timestamp without time zone
order_estimated_delivery_date timestamp without time zone
public
customers
customer_id character varying(50)
customer_unique_id character varying(50)
customer_zip_code_prefix integer
customer_city character varying(50)
customer_state character varying(50)
public
products
product_id character varying(100)
product_category_name character varying(100)
product_name_lenght integer
product_description_lenght integer
product_photos_qty integer
product_weight_g integer
product_length_cm integer
product_height_cm integer
product_width_cm integer
public
translation
category_name character varying
category_name_english character varying
public
reviews
review_id character varying(100)
order_id character varying(100)
review_score smallint
review_comment_title character varying(100)
review_comment_message character varying
review_creation_date date
review_answer_timestamp timestamp without time zone
public
order_items
order_id character varying(100)
order_item_id integer
product_id character varying(100)
seller_id character varying(100)
shipping_limit_date timestamp without time zone
price real
freight_value real
public
payments
order_id character varying(100)
payment_sequential integer
payment_type character varying(100)
payment_installments integer
payment_value double precision
public
sellers
seller_id character varying(100)
seller_zip_code_prefix integer
seller_city character varying(100)
seller_state character varying(100)

# QUESTION 1

## Montly Order Distribution:


```
SELECT TO_CHAR(order_approved_at, 'YYYY MONTH') AS month,
COUNT(*) AS order_count
FROM orders
WHERE order_approved_at IS NOT NULL
GROUP BY 1
ORDER BY 2 DESC;
```

| | month text | order_count bigint |
|---|---|---|
| 1 | 2017 NOVEMBER | 7395 |
| 2 | 2018 MARCH | 7288 |
| 3 | 2018 JANUARY | 7187 |
| 4 | 2018 MAY | 7066 |
| 5 | 2018 APRIL | 6778 |
| 6 | 2018 FEBRUARY | 6706 |
| 7 | 2018 AUGUST | 6620 |
| 8 | 2018 JULY | 6176 |
| 9 | 2018 JUNE | 6164 |
| 10 | 2017 DECEMBER | 5832 |
| 11 | 2017 OCTOBER | 4590 |
| 12 | 2017 AUGUST | 4348 |

# QUESTION 2:

Examine the number of orders in the order status breakdown on a monthly basis.

```
SELECT TO_CHAR(order_approved_at, 'YYYY-MM') AS month,
    order_status,
    COUNT(*) AS order_count
FROM orders
GROUP BY month, order_status
ORDER BY month
```

| | month<br>text | order_status<br>character varying (50) | order_count<br>bigint |
|---|---|---|---|
| 1 | 2016-09 | delivered | 1 |
| 2 | 2016-10 | canceled | 20 |
| 3 | 2016-10 | delivered | 265 |
| 4 | 2016-10 | invoiced | 18 |
| 5 | 2016-10 | processing | 2 |
| 6 | 2016-10 | shipped | 9 |
| 7 | 2016-10 | unavailable | 6 |
| 8 | 2016-12 | delivered | 1 |
| 9 | 2017-01 | canceled | 2 |
| 10 | 2017-01 | delivered | 715 |
| 11 | 2017-01 | invoiced | 11 |

Total rows: 132 of 132 Query complete 00:00:00.715

# QUESTION 3:
# Check the number of orders in the product category breakdown.

```
SELECT product_category_name,
COUNT(*) as order_count
FROM products
WHERE product_category_name IS NOT NULL
GROUP BY product_category_name
ORDER BY order_count desc
LIMIT 15
```

| | product_category_name character varying (100) | order_count bigint |
|---|---|---|
| 1 | cama_mesa_banho | 3029 |
| 2 | esporte_lazer | 2867 |
| 3 | moveis_decoracao | 2657 |
| 4 | beleza_saude | 2444 |
| 5 | utilidades_domesticas | 2335 |
| 6 | automotivo | 1900 |
| 7 | informatica_acessorios | 1639 |
| 8 | brinquedos | 1411 |
| 9 | relogios_presentes | 1329 |
| 10 | telefonia | 1134 |
| 11 | bebes | 919 |

Total rows: 15 of 15 | Query complete 00:00:00.7

# With the following query, we can see the sales amounts in all categories by month:

```
SELECT
    EXTRACT(MONTH FROM o.order_approved_at) AS order_month,
    p.product_category_name,
    COUNT(DISTINCT o.order_id) AS order_count
FROM
    order_items oi
INNER JOIN
    products p ON oi.product_id = p.product_id
INNER JOIN
    orders o ON oi.order_id = o.order_id
GROUP BY
    order_month, p.product_category_name
ORDER BY
    order_month, p.product_category_name;
```

| | order_month numeric | product_category_name character varying (100) | order_count bigint |
|---|---|---|---|
| 1 | 1 | agro_industria_e_comercio | 16 |
| 2 | 1 | alimentos | 21 |
| 3 | 1 | alimentos_bebidas | 10 |
| 4 | 1 | artes | 19 |
| 5 | 1 | artigos_de_festas | 2 |
| 6 | 1 | artigos_de_natal | 8 |
| 7 | 1 | audio | 24 |
| 8 | 1 | automotivo | 267 |
| 9 | 1 | bebes | 242 |
| 10 | 1 | bebidas | 40 |
| 11 | 1 | beleza_saude | 648 |

Total rows: 827 of 827 | Query complete 00:00:04.351

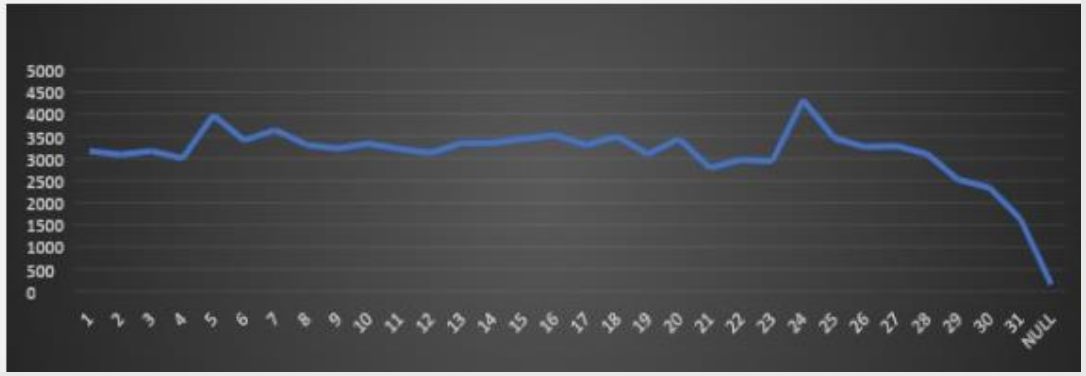
# QUESTION 4:

Examine the number of orders on the basis of days of the week (Monday, Thursday, ....) and month days (such as 1st, 2nd of the month).

```
SELECT EXTRACT(DOW FROM order_approved_at) AS
    day_of_week, COUNT(*) AS order_count
FROM orders
GROUP BY day_of_week
ORDER BY day_of_week
```

```
SELECT EXTRACT(DAY FROM  order_approved_at) AS
day_of_month,
COUNT(*) AS order_count
FROM orders
GROUP BY day_of_month
ORDER BY day_of_month;
```

| | day_of_week numeric | order_count bigint |
|---|---|---|
| 1 | 0 | 9014 |
| 2 | 1 | 13001 |
| 3 | 2 | 19154 |
| 4 | 3 | 15786 |
| 5 | 4 | 15471 |
| 6 | 5 | 14659 |
| 7 | 6 | 12196 |



# Question 5: In which cities do customers shop the most?

```
select count(distinct o.order_id) order_count,
c.customer_unique_id,
c.customer_city
from orders o
join customers c on o.customer_id = c.customer_id
join payments p on p.order_id = o.order_id
group by 2,3
order by 1 desc
```

| | order_count bigint | customer_unique_id character varying (50) | customer_city character varying (50) |
|---|---|---|---|
| 1 | 17 | 8d50f5eadf50201ccdcedfb9e2ac8455 | sao paulo |
| 2 | 9 | 3e43e6105506432c953e165fb2acf44c | praia grande |
| 3 | 7 | 1b6c7548a2a1f9037c1fd3ddfed95f33 | ituiutaba |
| 4 | 7 | 6469f99c1f9dfae7733b25662e7f1782 | santos |
| 5 | 7 | ca77025e7201e3b30c44b472ff346268 | recife |
| 6 | 6 | dc813062e0fc23409cd255f7f53c7074 | garanhuns |
| 7 | 6 | de34b16117594161a6a89c50b289d35a | santo andre |
| 8 | 6 | f0e310a6839dce9de1638e0fe5ab282a | vitoria |
| 9 | 6 | 12f5d6e1cbf93dafd9dcc19095df0b3d | curitiba |
| 10 | 6 | 47c1a3033b8b77b3ab6e109eb4d5fdf3 | jandira |
| 11 | 6 | 63cfc61cee11cbe306bff5857d00bfe4 | rio de janeiro |

Total rows: 1000 of 96218 | Query complete 00:00:06.972

# Number of Orders by City

| | customer_city character varying (50) | customer_count bigint |
|---|---|---|
| 1 | sao paulo | 14966 |
| 2 | rio de janeiro | 6612 |
| 3 | belo horizonte | 2672 |
| 4 | brasilia | 2067 |
| 5 | curitiba | 1463 |
| 6 | campinas | 1398 |
| 7 | porto alegre | 1326 |
| 8 | salvador | 1207 |
| 9 | guarulhos | 1152 |
| 10 | sao bernardo do campo | 908 |
| 11 | niteroi | 809 |

```
WITH customer_order_counts AS (
    SELECT
        c.customer_unique_id,
        c.customer_city,
        COUNT(o.order_id) AS order_count,
        ROW_NUMBER() OVER(PARTITION BY
      c.customer_unique_id
ORDER BY COUNT(o.order_id) DESC) AS rn
    FROM orders o
    JOIN customers c ON o.customer_id = c.customer_id
    JOIN payments p ON p.order_id = o.order_id
    GROUP BY
        c.customer_unique_id,
        c.customer_city)
SELECT
    customer_city,
    COUNT(*) AS customer_count
FROM customer_order_counts
WHERE rn = 1
GROUP BY customer_city
ORDER BY customer_count DESC
```

# Question 6:
# Who are the vendors who deliver orders to customers in the fastest way?

```
SELECT oi.seller_id,
ROUND(AVG(EXTRACT(EPOCH FROM
    (o.order_delivered_customer_date -
    o.order_purchase_timestamp)))/3600,2) AS delivery_time,
COUNT(o.order_id) AS order_count,
ROUND(AVG(r.review_score),2) AS average_review_score
FROM orders AS o
JOIN order_items AS oi ON o.order_id = oi.order_id
JOIN reviews AS r ON o.order_id = r.order_id
WHERE o.order_status = 'delivered'
GROUP BY oi.seller_id
ORDER BY delivery_time ASC
LIMIT 5;
```

| | seller_id<br>character varying (100) | delivery_time<br>numeric | order_count<br>bigint | average_review_score<br>numeric |
|---|---|---|---|---|
| 1 | 139157dd4daa45c25b0807ffff348363 | 29.14 | 1 | 4.00 |
| 2 | 5e063e85d44b0f5c3e6ec3131103a5... | 30.92 | 1 | 5.00 |
| 3 | 6561d6bf844e464b4019442692b40e... | 34.42 | 1 | 5.00 |
| 4 | 702835e4b785b67a084280efca3557... | 43.30 | 1 | 5.00 |
| 5 | 674207551483fec113276b67b0d871ff | 44.87 | 1 | 5.00 |

Although the average score of all 5 sellers is high and the order delivery time is short, when we look at the number of orders, we see that they are 1 in all of them. Among them, there are sellers who have a bad score despite the fact that fast shipping has been made. In other words, the seller who ships every order quickly does not have a good rating. For this reason, there is no healthy determination for the average score and order delivery time. In order to get better results, it would be better to look at sellers with both a high number of orders and a high score. I think that by adding HAVING COUNT(o.order_id) > 10 after the group by statement, the query results will be healthier by reaching the sellers who have sent at least 10 orders.

```
SELECT oi.seller_id,
ROUND(AVG(EXTRACT(EPOCH FROM (o.order_delivered_customer_date -
    o.order_purchase_timestamp)))/3600,2) AS delivery_time,
COUNT(o.order_id) AS order_count,
ROUND(AVG(r.review_score),2) AS average_review_score
FROM orders AS o
JOIN order_items AS oi ON o.order_id = oi.order_id
JOIN reviews AS r ON o.order_id = r.order_id
WHERE o.order_status = 'delivered'
GROUP BY oi.seller_id
HAVING COUNT(o.order_id) > 10
ORDER BY delivery_time ASC
LIMIT 5;
```

| | seller_id<br>character varying (100) | delivery_time<br>numeric | order_count<br>bigint | average_review_score<br>numeric |
|---|---|---|---|---|
| 1 | 6e1862e15f33d9994bc25922a85e1e... | 87.19 | 12 | 4.67 |
| 2 | 30a81d8cf85fb2ada1b1b094c9583a... | 96.60 | 20 | 4.85 |
| 3 | 3fac58ce0ad699020c7944d53c4132... | 98.54 | 16 | 4.56 |
| 4 | 5a413ade68e8f8d93071a7f52a64cb... | 98.85 | 14 | 4.64 |
| 5 | c3e1abd72a42fe690fcd89cf5720fe29 | 103.43 | 13 | 4.46 |

# Question 7:
# Which merchants sell products from more categories? Do sellers with more categories also have a high number of orders?

```
SELECT o.seller_id,
COUNT(DISTINCT o.order_id) AS order_count,
COUNT(DISTINCT p.product_category_name) AS category_count
FROM order_items o
JOIN products p ON o.product_id = p.product_id
GROUP BY o.seller_id
ORDER BY category_count DESC
```

| | seller_id character varying (100) | order_count bigint | category_count bigint |
|---|---|---|---|
| 1 | b2ba3715d723d245138f291a6fe42594 | 337 | 27 |
| 2 | 4e922959ae960d389249c378d1c939... | 420 | 23 |
| 3 | 955fee9216a65b617aa5c0531780ce60 | 1287 | 23 |
| 4 | 1da3aeb70d7989d1e6d9b0e887f97c... | 265 | 21 |
| 5 | f8db351d8c4c4c22c6835c19a46f01b0 | 667 | 19 |
| 6 | 18a349e75d307f4b4cc646a691ed42... | 121 | 17 |
| 7 | 6edacfd9f9074789dad6d62ba7950b9c | 208 | 15 |
| 8 | 70a12e78e608ac31179aea7f842204... | 315 | 15 |
| 9 | 7178f9f4dd81dcef02f62acdf8151e01 | 203 | 14 |
| 10 | fd386aa7bed2af3c7035c65506c9b4a3 | 69 | 14 |
| 11 | 8b28d096634035667e8263d57ba336... | 143 | 14 |

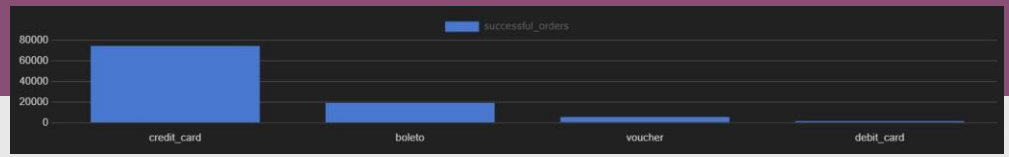
Total rows: 1000 of 3095 | Query complete 00:00:01.722

# Question 8:
# In which region do users with a high number of installments live the most?

```
SELECT
  c.customer_city,
  COUNT(DISTINCT p.order_id) AS order_count
FROM
  payments p
INNER JOIN
  orders o ON p.order_id = o.order_id
INNER JOIN
  customers c ON o.customer_id = c.customer_id
WHERE
  p.payment_installments > 6
GROUP BY
  c.customer_city
ORDER BY
  order_count DESC
```

| | customer_city character varying (50) | order_count bigint |
|---|---|---|
| 1 | sao paulo | 1436 |
| 2 | rio de janeiro | 844 |
| 3 | belo horizonte | 340 |
| 4 | brasilia | 237 |
| 5 | porto alegre | 197 |
| 6 | curitiba | 190 |
| 7 | salvador | 174 |
| 8 | campinas | 159 |
| 9 | guarulhos | 119 |
| 10 | recife | 96 |
| 11 | fortaleza | 95 |

# The most successful payment was by credit card with 74586 units, with a total amount of 12,101,094.87:




```
SELECT payment_type,
COUNT(order_id) AS successful_orders,
SUM(payment_value) AS total_successful_payments
FROM (
SELECT o.order_id,
p.payment_type,
p.payment_value
FROM orders o
INNER JOIN payments p ON o.order_id = p.order_id
WHERE o.order_status = 'delivered'
) AS temp
GROUP BY payment_type
ORDER BY total_successful_payments DESC
```

# Question 9:
# In which categories are installment payments mostly used?

```
SELECT p.product_category_name,
COUNT(DISTINCT o.order_id) AS order_count
FROM customers c
INNER JOIN orders o ON c.customer_id = o.customer_id
INNER JOIN order_items oi ON o.order_id = oi.order_id
INNER JOIN products p ON oi.product_id = p.product_id
INNER JOIN payments pm ON o.order_id = pm.order_id
WHERE pm.payment_installments > 1
GROUP BY p.product_category_name
ORDER BY 2 DESC;
```

| | product_category_name character varying (100) | order_count bigint |
|---|---|---|
| 1 | cama_mesa_banho | 5965 |
| 2 | beleza_saude | 5006 |
| 3 | relogios_presentes | 3794 |
| 4 | esporte_lazer | 3480 |
| 5 | moveis_decoracao | 3353 |
| 6 | utilidades_domesticas | 3197 |
| 7 | informatica_acessorios | 2562 |
| 8 | cool_stuff | 2217 |
| 9 | brinquedos | 2008 |
| 10 | perfumaria | 1945 |
| 11 | automotivo | 1882 |

Total rows: 73 of 73 | Query complete 00:00:01.271

# Thanks for watching!