Diccionario de tablas

Contamos con 9 archivos CSV diferentes y se creó una tabla a partir de ellos:

- olist_customers_dataset.csv

olist_customers_dataset	Variable	Definición	Llave
	customer_id	Identificador de órdenes	Alfanumérico
	customer_unique_id	Identificador único para cliente	Alfanumérico
	customer_zip_code_prefix	Primeros 5 dígitos de CP del cliente	Entero
	customer_city	Nombre de la ciudad del cliente	Alfanumérico
	customer_state	Código de estado geográfico del cliente	Alfanumérico

- olist_geolocation_dataset.csv

olist_geolocation_dataset	Variable	Definición	Llave
	geolocation_zip_ code_prefix	Primeros 5 dígitos de Código Postal	Entero
	geolocation_lat	Latitud	Flotante
	geolocation_lng	Longitud	Flotante
	geolocation_city	Nombre de la ciudad	Alfanumérico
	geolocation_state	Código de estado geográfico	Alfanumérico

- olist_order_items_dataset.csv

olist_order_items_dataset	Variable	Definición	Llave
	order_id	Identificador Único de orden	Alfanumérico
	order_item_id	Número secuencial que identifica el número de arículos incluídos en la misma orden	Entero
	product_id	Identificador Único de producto	Alfanumérico
	seller_id	Identificador Único de vendedor	Alfanumérico
	shipping_limit_date	Fecha máxima de entrega de producto al socio logístico por parte del vendedor	Fecha
	price	Precio del producto	Flotante
	freight_value	Precio de envío del producto	Flotante

- olist_order_payments_dataset.csv

olist_order_payments_dataset	Variable	Definición	Llave
	order_id	Identificador Único de orden	Alfanumérico
	payment_sequential	Número de métodos de pago diferenetes utilizados por el cliente	Entero
	payment_type	Método de pago (elegido por el cliente)	Alfanumérico (crdit_card, boleto, voucher, debit_card, not_defined)
	payment_installments	Número de plazos para realizar la compra (elegido por el cliente)	Entero
	payment_value	Valor de la transacción	Flotante

- olist_order_reviews_dataset.csv

olist_order_reviews_dataset	Variable	Definición	Llave
	review_id	Identificador Único de reseña	Alfanumérico
	order_id	Identificador Único de orden	Alfanumérico
	review_score	Nota de satisfacción proporcionada por el cliente	Entero (Escala 1 a 5 de menor a mayor grado de satisfacción)
	review_comment_title	Título del comentario dejado por el cliente sobre su nota de satisfacción	Alfanumérico (Portugués)
	review_comment_message	Comentario dejado por el cliente sobre su nota de satisfacción	Alfanumérico (Portugués)
	review_creation_date	Fecha de envío de encuesta de satisfacción al cliente	Fecha
	review_answer_timestamp	Fecha de llenado de encuesta	Fecha

- olist_orders_dataset.csv

olist_orders_dataset	Variable	Definición	Llave
	order_id	Identificador Único de orden	Alfanumérico
	customer_id	Identificador de cliente	Alfanumérico
	order_status	Estado de la orden	Alfanumérico (delivered, shipped, canceled, unavailable, invoiced, processing, created, approved)
	order_purchase_timestamp	Fecha de compra	Fecha
	order_approved_at	Fecha de aprovación de pago	Fecha
	order_delivered_ carrier_date	Fecha de producto entregado a socio logístico	Fecha
	order_delivered_ customer_date	Fecha real de entrega de producto a cliente	Fecha
	order_estimated_ delivery_date	Fecha estimada de entrega de producto a cliente	Fecha

- olist_products_dataset.csv

olist_products_dataset	Variable	Definición	Llave
	product_id	Identificador Único de producto	Alfanumérico
	product_category_name	Categoría raíz de producto	Alfanumérico (en Portugués)
	product_name_lenght	Número de caracteres extraídos del nombre del producto	Entero
	product_description_ lenght	Número de caracteres extraídos de la descrpición del producto	Entero
	product_photos_qty	Número de fotografías publicadas	Entero
	product_weight_g	Peso del producto	Flotante (gramos)
	product_length_cm	Largo del producto	Flotante (centímetros)
	product_height_cm	Alto del producto	Flotante (centímetros)
	product_width_cm	Ancho del producto	Flotante (centímetros)

- olist_sellers_dataset.csv

olist_sellers_dataset	Variable	Definición	Llave
	seller_id	Identificador Único de vendedor	Alfanumérico
	seller_zip_code_prefix	Primeros 5 dígitos de CP de vendedor	Entero
	seller_city	Nombre de la ciudad de vendedor	Alfanumérico
	seller_state	Estado geográfico de vendedor	Alfanumérico

- product_category_name_translation.csv

product_category_ name_translation	Variable	Definición	Llave
	product_category_name	Nombre de categoría de producto	Alfanumérico (en Portugués)
	product_category_name_ english	Nombre de categoría de producto	Alfanumérico (en Inglés)

- olist_state_location.csv

olist_state_location	Variable	Definición	Llave
		Índice de fila	Entero
	State_Code	Código Federal de estado	Alfanumérico
	State	Código de estado	Alfanumérico
	State_name	Nombre de estado	Alfanumérico (en Portugués)
	Lat	Latitud	Flotante
	Lon	Longitud	Flotante

Calidad de los datos.

Cargado de datos

```
closed_deals = pd.read_csv('..\data\olist_closed_deals_dataset.csv',
parse_dates=["won_date"])

customers = pd.read_csv('..\data\olist_customers_dataset.csv')

geolocation = pd.read_csv('..\data\olist_geolocation_dataset.csv')

marketing_qualified_leads =
pd.read_csv('..\data\olist_marketing_qualified_leads_dataset.csv',
parse_dates=["first_contact_date"])

order_items = pd.read_csv('..\data\olist_order_items_dataset.csv',
parse_dates=["shipping_limit_date"])

order_payments = pd.read_csv('..\data\olist_order_payments_dataset.csv')

order_reviews = pd.read_csv('..\data\olist_order_reviews_dataset.csv',
parse_dates=["review_creation_date", 'review_answer_timestamp'])

orders = pd.read_csv('..\data\olist_orders_dataset.csv',
parse_dates=["order_purchase_timestamp", 'order_approved_at',
'order_delivered_carrier_date', 'order_delivered_customer_date',
'order_estimated_delivery_date'])

products = pd.read_csv('..\data\olist_products_dataset.csv')

sellers = pd.read_csv('..\data\olist_sellers_dataset.csv')
```

Customers

```
customers = pd.read_csv('data\olist_customers_dataset.csv')
Data columns (total 5 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   customer_id               99441 non-null  object
 1   customer_unique_id        99441 non-null  object
 2   customer_zip_code_prefix  99441 non-null  int64
 3   customer_city             99441 non-null  object
 4   customer_state            99441 non-null  object
dtypes: int64(1), object(4)
memory usage: 3.8+ MB
```

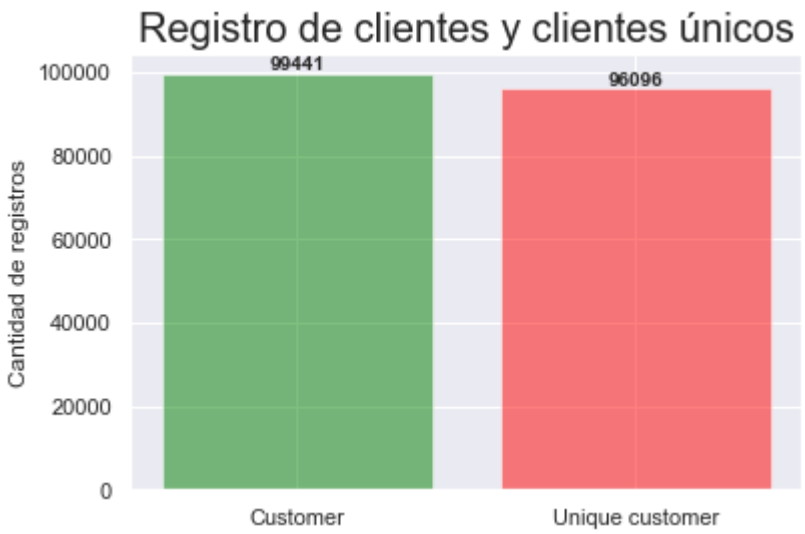
El total de registros de 'customer_id' es 99441

Se observa que existen diferentes 'customer_id' con un mismo 'customer_unique_id' (razón desconocida)

	customer_id	customer_unique_id	customer_zip_code_prefix	customer_city	customer_state
14186	1bd3585471932167ab72a84955ebefea	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
15321	a8fabc805e9a10a3c93ae5bff642b86b	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
16654	897b7f72042714efaa64ac306ba0cafc	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
36122	b2b13de0770e06de50080fea77c459e6	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
38073	42dbc1ad9d560637c9c4c1533746f86d	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
40141	dfb941d6f7b02f57a44c3b7c3fefb44b	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
48614	65f9db9dd07a4e79b625effa4c868fcb	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
52574	1c62b48fb34ee043310dcb233caabd2e	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
58707	a682769c4bc10fc6ef2101337a6c83c9	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
67996	6289b75219d757a56c0cce8d9e427900	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
72745	3414a9c813e3ca02504b8be8b2deb27f	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
74510	0e4fdc084a6b9329ed55d62dcd653ccf	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
83363	f5188d99e9281e214a4a7d1b139a8229	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
85507	89be66634d68fa73a95499b6352e085d	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
90268	0bf8bf19944a7f8b40ba86fef778ca7c	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
93591	9a1afef458843a022e431f4cb304dfe9	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP
96652	31dd055624c66f291578297a551a6cdf	8d50f5eadf50201ccdcedfb9e2ac8455	4045	sao paulo	SP

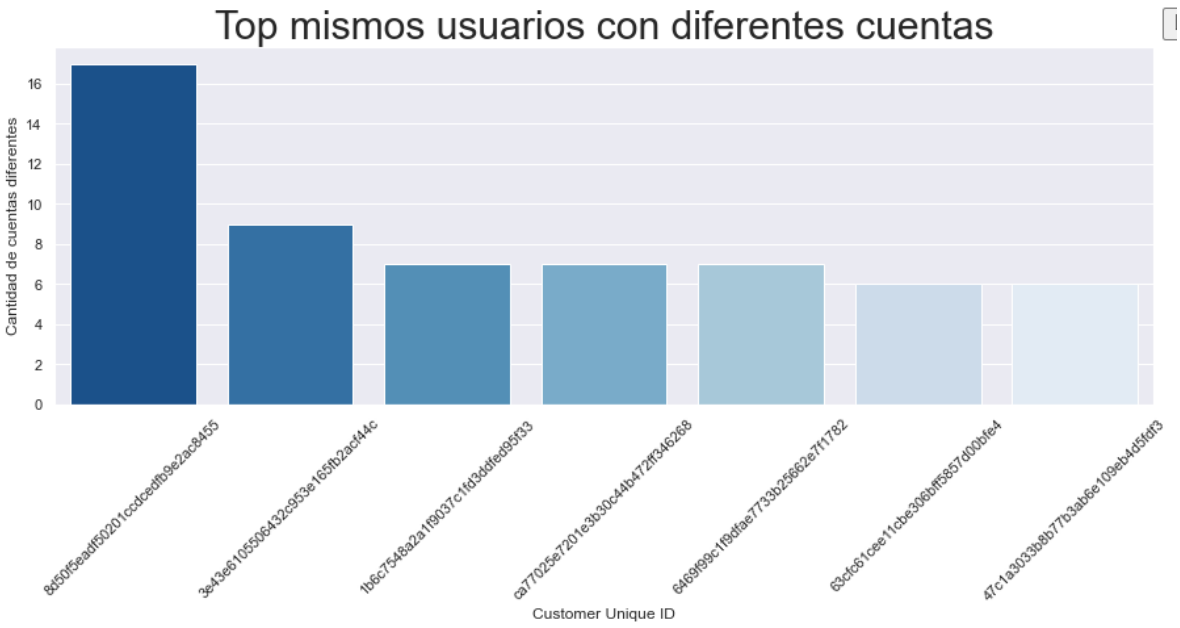
En algunos varía el 'customer_zip_code_prefix'; sin embargo, en general, se mantienen en el mismo estado.

El total de registros 'customer_unique_id' es 96096



Diferencia de 3345 que corresponde a que un 3.4% de los clientes tienen más de 1 cuenta

Las cuentas van desde 17 por usuario único hasta 1, como se muestran en la siguiente gráfica



La distribución de los clientes por estado queda de la siguiente manera

La región más importante de mayor potencial de ventas es el estado SP (Sa~o Paolo)

Geolocation

```
geolocation = pd.read_csv('data\olist_geolocation_dataset.csv')
```

```
 #   Column                       Non-Null Count    Dtype
---  ------                       --------------    -----
 0   geolocation_zip_code_prefix  1000163 non-null  int64
 1   geolocation_lat              1000163 non-null  float64
 2   geolocation_lng              1000163 non-null  float64
 3   geolocation_city             1000163 non-null  object
 4   geolocation_state            1000163 non-null  object
dtypes: float64(2), int64(1), object(2)
memory usage: 38.2+ MB
```

1000163 datos no nulos.

Se nos recomienda en el proyecto trabajar con una finura de estado.

Tomo una tabla de Wikipedia de la distribución geográfica estatal de Brasil y hago ‘join’ para tener un nuevo Dataframe que contenga los nombres de los estados (la tabla original no los contiene), además contiene la información de la latitud y longitud de cada uno utilizando el promedio de las latitudes y longitudes para cada estado (creo también un nuevo archivo csv con dicha información para utilizarlo más adelante) llamado ‘state_location’.

```
state_location
```

✓ 0.5s

	State_Code	State	State_name	Lat	Lon
0	BR-SP	SP	São Paulo	-23.155308	-47.084074
1	BR-RN	RN	Río Grande del Norte	-5.856702	-35.990079
2	BR-AC	AC	Acre	-9.702555	-68.451852
3	BR-RJ	RJ	Río de Janeiro	-22.743477	-43.155540
4	BR-ES	ES	Espírito Santo	-20.105145	-40.503183
5	BR-MG	MG	Minas Gerais	-19.864647	-44.421615
6	BR-BA	BA	Bahía	-13.049361	-39.560649
7	BR-SE	SE	Sergipe	-10.866199	-37.181169
8	BR-PE	PE	Pernambuco	-8.179098	-35.758866
9	BR-AL	AL	Alagoas	-9.599729	-36.052017
10	BR-PB	PB	Paraíba	-7.088298	-35.821678
11	BR-CE	CE	Ceará	-4.363151	-39.004140
12	BR-PI	PI	Piauí	-5.754989	-42.509541
13	BR-MA	MA	Maranhão	-3.798997	-44.818627
14	BR-PA	PA	Pará	-2.631213	-49.485862
15	BR-AP	AP	Amapá	0.086025	-51.234304

Order_items

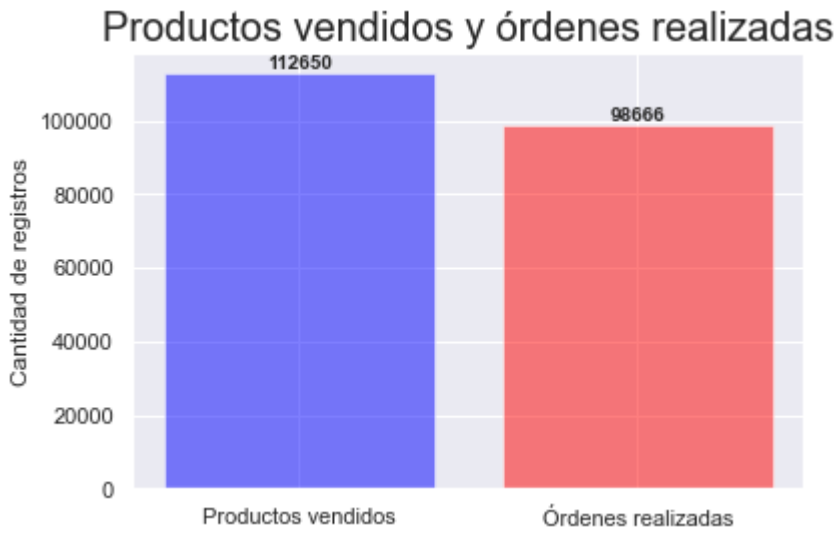
```
order_items = pd.read_csv('data\olist_order_items_dataset.csv',
parse_dates=["shipping_limit_date"])
```

```
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   order_id             112650 non-null  object
 1   order_item_id        112650 non-null  int64
 2   product_id           112650 non-null  object
 3   seller_id            112650 non-null  object
 4   shipping_limit_date  112650 non-null  datetime64[ns]
 5   price                112650 non-null  float64
 6   freight_value        112650 non-null  float64
dtypes: datetime64[ns](1), float64(2), int64(1), object(3)
memory usage: 6.0+ MB
```

112650 datos no nulos.

De los cuales únicamente 98666 corresponden a pedidos diferentes, debido a que un mismo pedido puede contener uno o más productos.

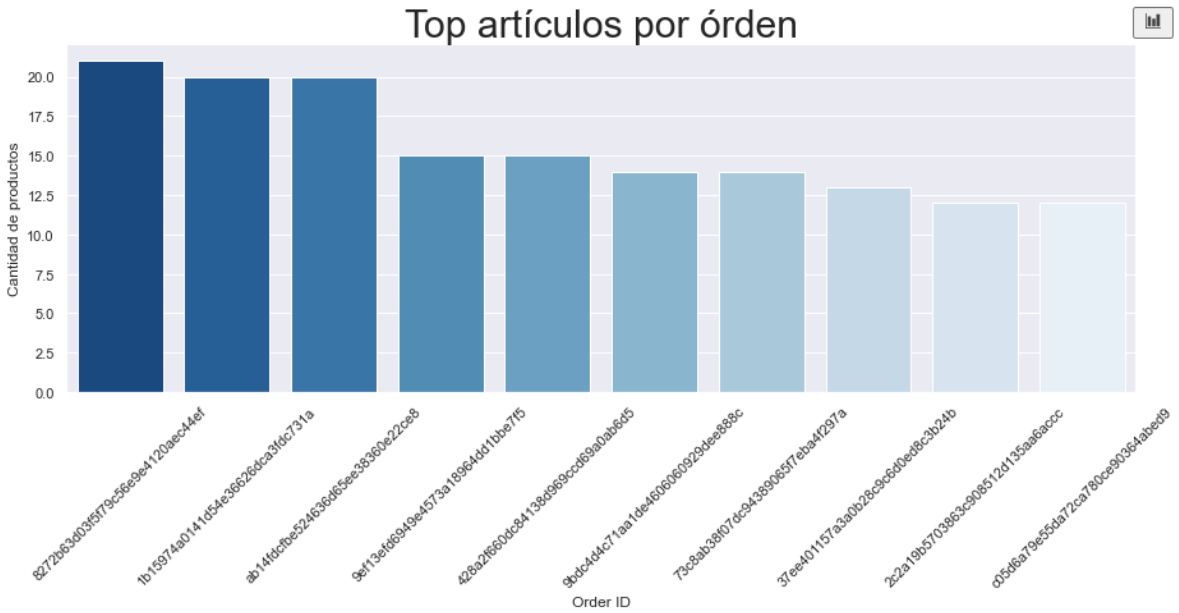
La distribución se encuentra en la siguiente gráfica:



Siendo una diferencia de 13984

Lo que nos dice que el 12.4% de todas las órdenes contienen más de 2 artículos.

La distribución va de 21 a 1 producto por orden, el siguiente gráfico es el top de ellos



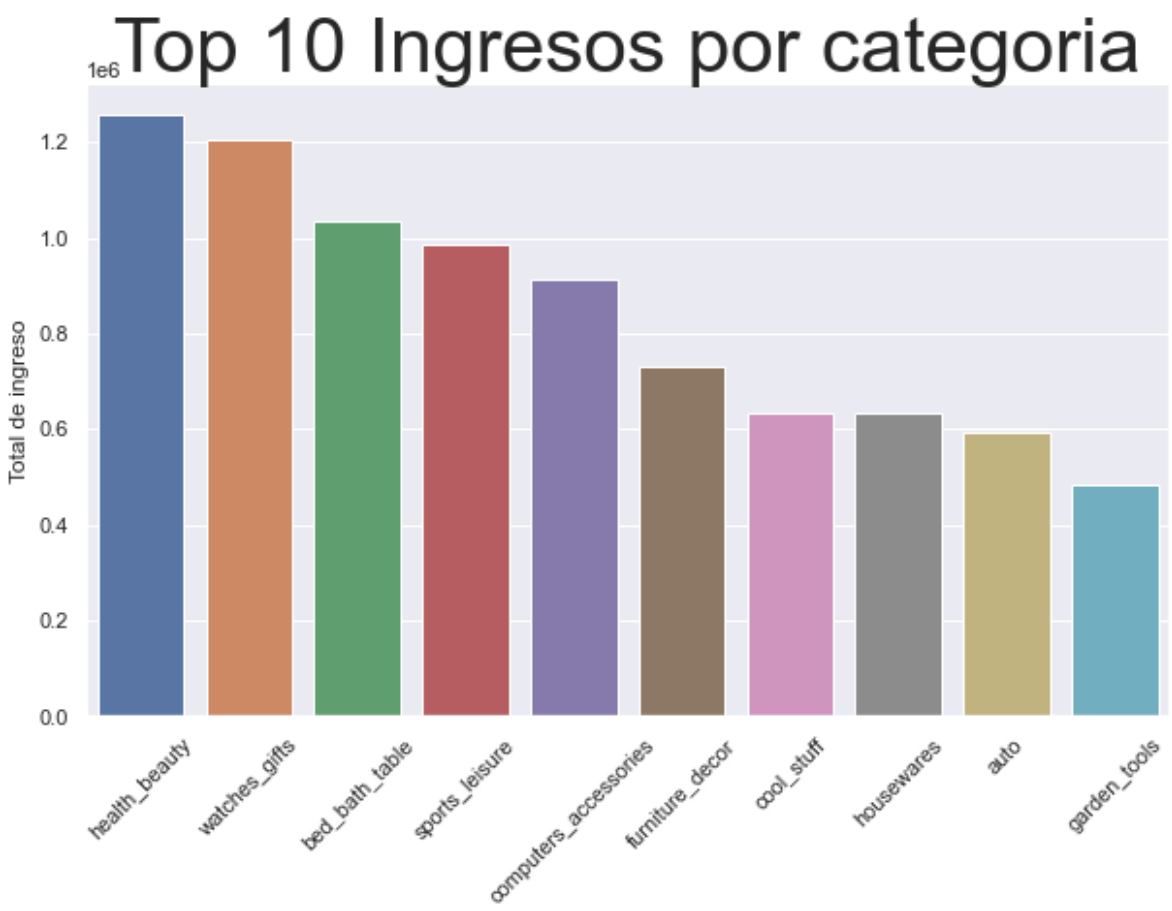
Se puede observar que, además, contamos con la información del precio del artículo y el precio del envío.

```
order_items[order_items.order_id=='8272b63d03f5f79c56e9e4120aec44ef']
```

Python

order_id	order_item_id	product_id	seller_id	shipping_limit_date	price	freight_value
120aec44ef	1	270516a3f41dc035aa87d220228f844c	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	2	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	3	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	4	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	5	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	6	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	7	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	8	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	9	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	10	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	11	05b515fdc76e888aada3c6d66c201dff	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	12	270516a3f41dc035aa87d220228f844c	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89
120aec44ef	13	270516a3f41dc035aa87d220228f844c	2709af9587499e95e803a6498a5a56e9	2017-07-21 18:25:23	1.2	7.89

Top 10 ingresos por categoría:



Order_payments

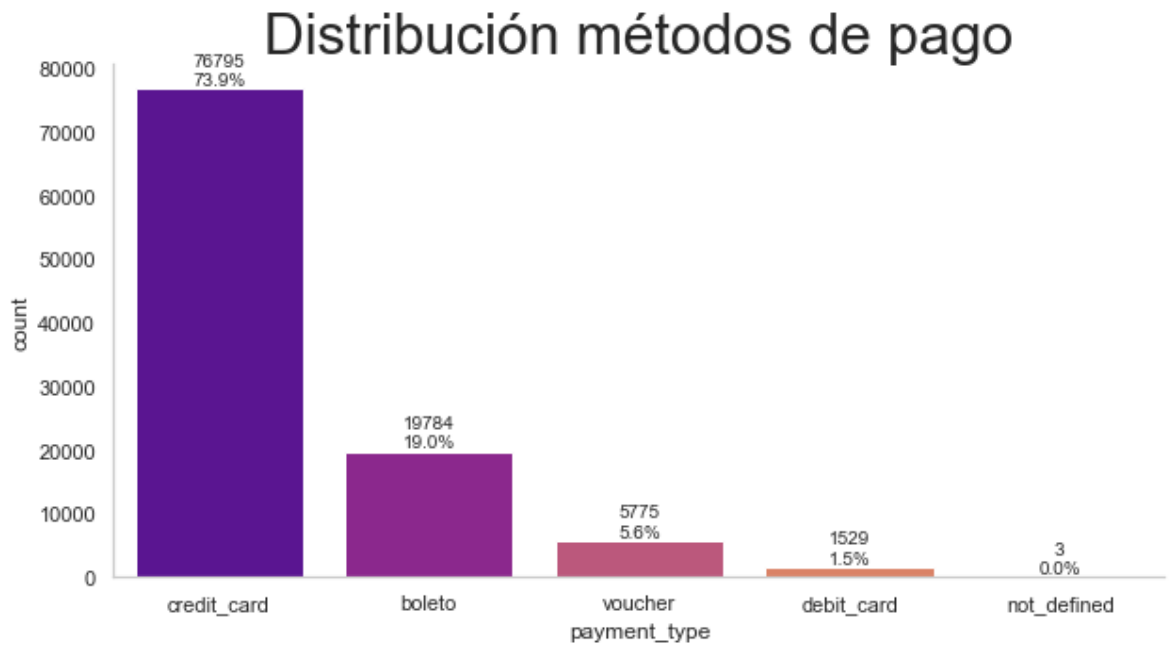
```
order_payments = pd.read_csv('data\olist_order_payments_dataset.csv')
```

```
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   order_id              103886 non-null  object
 1   payment_sequential    103886 non-null  int64
 2   payment_type          103886 non-null  object
 3   payment_installments  103886 non-null  int64
 4   payment_value         103886 non-null  float64
dtypes: float64(1), int64(2), object(2)
memory usage: 4.0+ MB
```

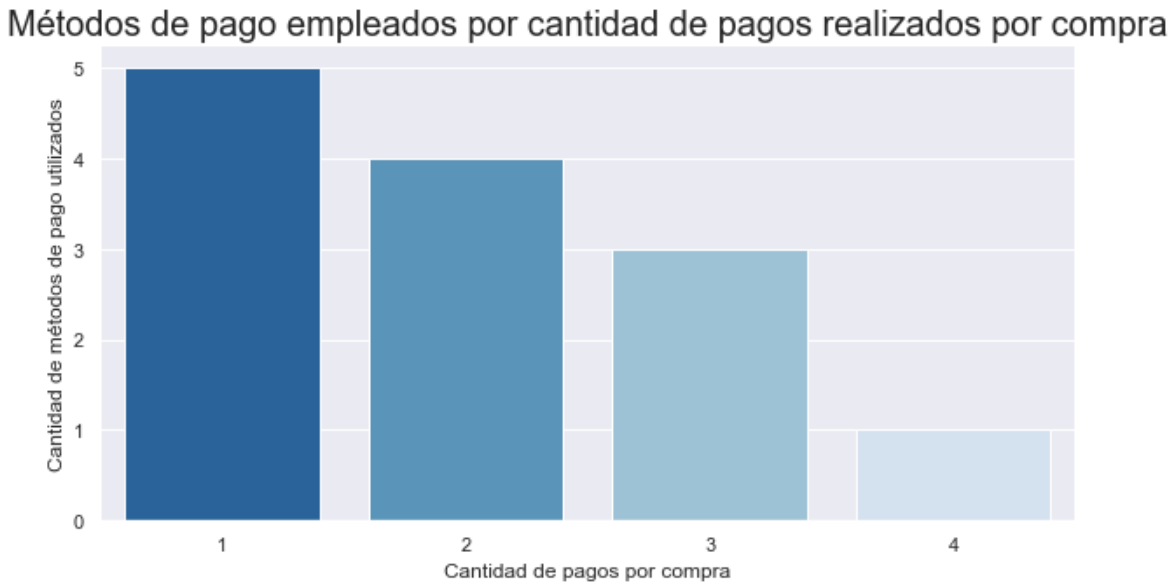
103886 datos no nulos.

Del total de órdenes, tenemos 99440 órdenes distintas realizadas, por lo que, una misma orden fue pagada con más de un método de pago

La distribución de los métodos de pago queda de la siguiente manera



Entre más movimientos se necesitaron para realizar el pago, menos métodos de pago fueron empleados



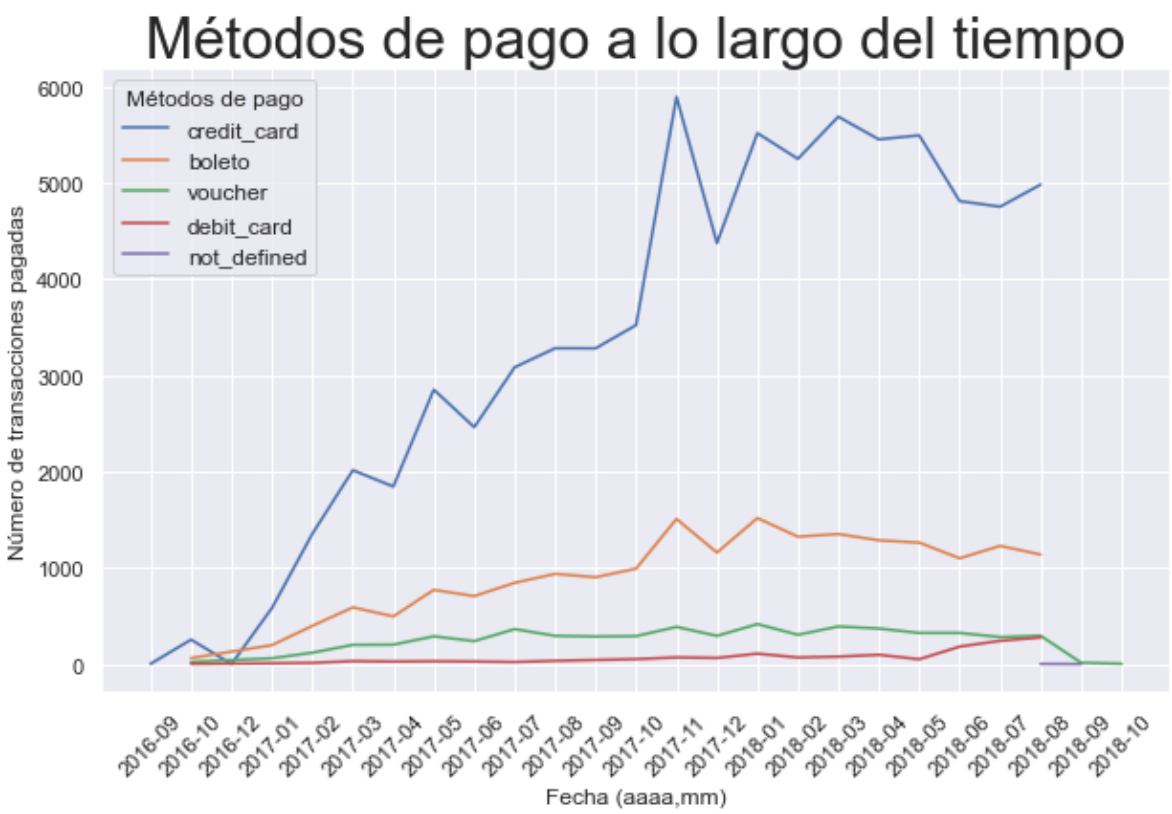
A partir de 4 operaciones para realizar el pago de la orden se observa que sólo se utiliza un método de pago, el cual es el siguiente: 'voucher'

```
order_payments[order_payments.order_id=='fa65dad1b0e818e3ccc5cb0e39231352']
```

✓ 0.1s

	order_id	payment_sequential	payment_type	payment_installments	payment_value
4885	fa65dad1b0e818e3ccc5cb0e39231352	27	voucher	1	66.02
9985	fa65dad1b0e818e3ccc5cb0e39231352	4	voucher	1	29.16
14321	fa65dad1b0e818e3ccc5cb0e39231352	1	voucher	1	3.71
17274	fa65dad1b0e818e3ccc5cb0e39231352	9	voucher	1	1.08
19565	fa65dad1b0e818e3ccc5cb0e39231352	10	voucher	1	12.86
23074	fa65dad1b0e818e3ccc5cb0e39231352	2	voucher	1	8.51
24879	fa65dad1b0e818e3ccc5cb0e39231352	25	voucher	1	3.68
28330	fa65dad1b0e818e3ccc5cb0e39231352	5	voucher	1	0.66
29648	fa65dad1b0e818e3ccc5cb0e39231352	6	voucher	1	5.02
32519	fa65dad1b0e818e3ccc5cb0e39231352	11	voucher	1	4.03
36822	fa65dad1b0e818e3ccc5cb0e39231352	14	voucher	1	0.00
39108	fa65dad1b0e818e3ccc5cb0e39231352	29	voucher	1	19.26
39111	fa65dad1b0e818e3ccc5cb0e39231352	28	voucher	1	29.05
63369	fa65dad1b0e818e3ccc5cb0e39231352	15	voucher	1	14.04

Históricamente la manera en cómo se han utilizado los métodos de pago se muestra en la siguiente gráfica:


Métodos de pago a lo largo del tiempo
Métodos de pago
credit_card
boleto
voucher
debit_card
not_defined
Número de transacciones pagadas
0
1000
2000
3000
4000
5000
6000
2016-09
2016-10
2016-12
2017-01
2017-02
2017-03
2017-04
2017-05
2017-06
2017-07
2017-08
2017-09
2017-10
2017-11
2017-12
2018-01
2018-02
2018-03
2018-04
2018-05
2018-06
2018-07
2018-08
2018-09
2018-10
Fecha (aaaa,mm)

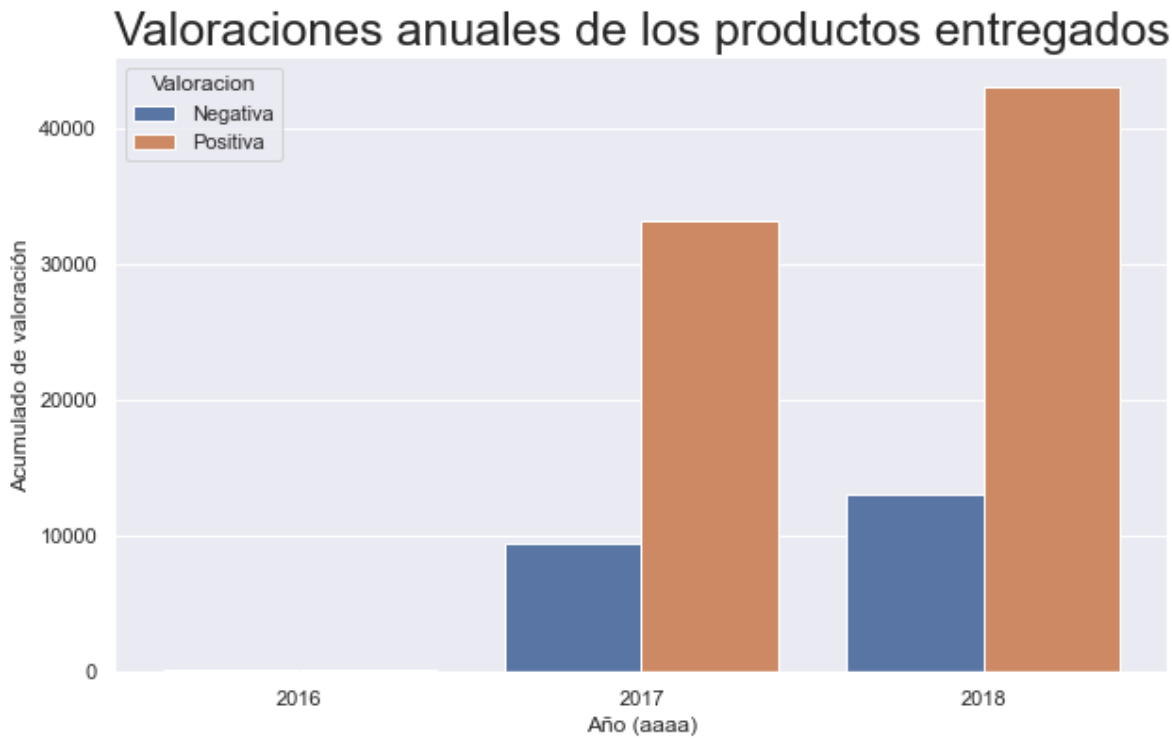
Order_reviews

```
order_reviews = pd.read_csv('data\olist_order_reviews_dataset.csv',
parse_dates=["review_creation_date", 'review_answer_timestamp'])
```

```
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   review_id               99224 non-null  object
 1   order_id                99224 non-null  object
 2   review_score            99224 non-null  int64
 3   review_comment_title    11568 non-null  object
 4   review_comment_message  40977 non-null  object
 5   review_creation_date    99224 non-null  datetime64[ns]
 6   review_answer_timestamp  99224 non-null  datetime64[ns]
dtypes: datetime64[ns](2), int64(1), object(4)
memory usage: 5.3+ MB
```

Se desconoce el motive de los datos faltantes

La comparativa entre calificaciones negativas y positivas se muestra a continuación

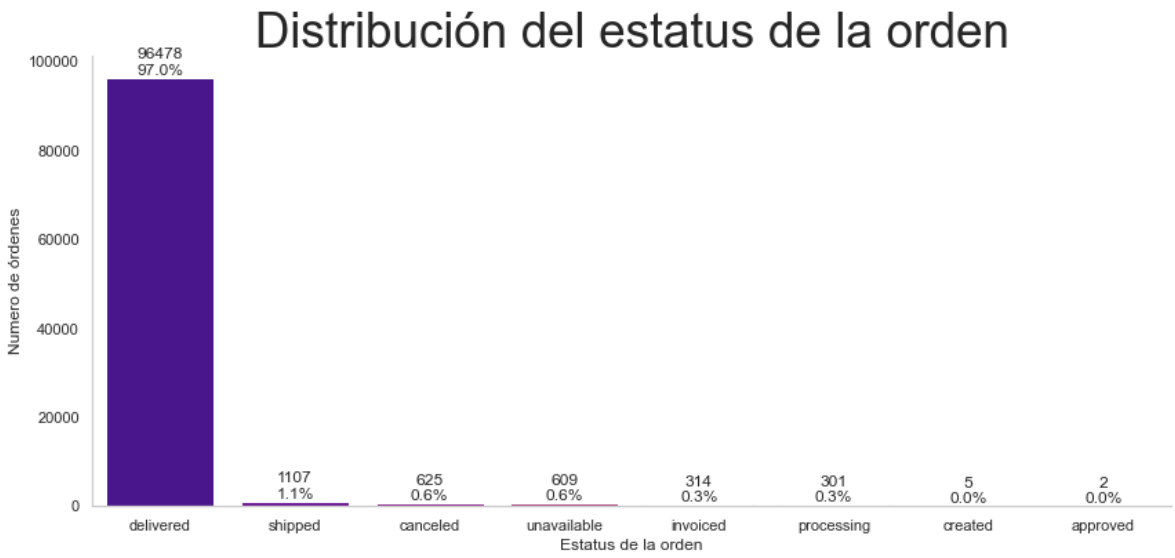


Orders

```
orders = pd.read_csv('data\olist_orders_dataset.csv',
parse_dates=["order_purchase_timestamp", 'order_approved_at',
'order_delivered_carrier_date', 'order_delivered_customer_date',
'order_estimated_delivery_date'])
```

```
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   order_id                       99441 non-null  object
 1   customer_id                    99441 non-null  object
 2   order_status                   99441 non-null  object
 3   order_purchase_timestamp       99441 non-null  datetime64[ns]
 4   order_approved_at              99281 non-null  datetime64[ns]
 5   order_delivered_carrier_date   97658 non-null  datetime64[ns]
 6   order_delivered_customer_date  96476 non-null  datetime64[ns]
 7   order_estimated_delivery_date  99441 non-null  datetime64[ns]
dtypes: datetime64[ns](5), object(3)
memory usage: 6.1+ MB
```

La distribución del estatus de las órdenes queda de la siguiente manera:



Nos queda que sólo el 3% de las ventas realizadas no fueron entregadas

En el proceso de envío, las fechas aumentan desde que se realizo la compra hasta el estimado de entrega del producto; sin embargo, existen fechas de procesos anteriores superiores al proceso siguiente, por ejemplo:

```
np.min(orders.order_delivered_carrier_date - orders.order_approved_at)
```

```
Timedelta('-172 days +18:44:38')
```

```
orders[(orders.order_delivered_carrier_date - orders.order_approved_at)
```

us	order_purchase_timestamp	order_approved_at	order_delivered_carrier_date	orc
ed	2018-07-16 18:40:53	2018-07-16 18:50:22	2018-01-26 13:35:00	

Products

```
products = pd.read_csv('data\olist_products_dataset.csv')
```

```
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   product_id                 32951 non-null  object
 1   product_category_name      32341 non-null  object
 2   product_name_lenght        32341 non-null  float64
 3   product_description_lenght 32341 non-null  float64
 4   product_photos_qty         32341 non-null  float64
 5   product_weight_g           32949 non-null  float64
 6   product_length_cm          32949 non-null  float64
 7   product_height_cm          32949 non-null  float64
 8   product_width_cm           32949 non-null  float64
dtypes: float64(7), object(2)
memory usage: 2.3+ MB
```

Existen productos que no tienen ningún tipo de descripción más que los valores de sus dimensiones

```
products[products.isnull().any(1)]
```

✓ 0.8s

product_id	product_category_name	product_name_lenght	product_description_lenght	product_photos_qty	product_weight_g
334f36de622ecbd3a	NaN	NaN	NaN	NaN	650.0
075997acef1870e9b	NaN	NaN	NaN	NaN	300.0
f19eb9f7dae4d1617	NaN	NaN	NaN	NaN	200.0
ced748f324108c733	NaN	NaN	NaN	NaN	18500.0
72f5e586bb132eae9	NaN	NaN	NaN	NaN	300.0
...	...	...	...	...	...
73b199380612c350a	NaN	NaN	NaN	NaN	1800.0
5123c17fdc34a63c56	NaN	NaN	NaN	NaN	800.0

Sin embargo, sí han sido comprados

```
order_items[order_items.product_id=='a41e356c76fab66334f36de622ecbd3a']
```

✓ 0.1s Python

	order_id	order_item_id	product_id	seller_id	shipping_
81289	b8bfa12431142333a0c84802f9529d87	2	a41e356c76fab66334f36de622ecbd3a	d9cb0052a666de5308b32f32ad5f1b1c	2018-01-2

```
order_items[order_items.order_id=='b8bfa12431142333a0c84802f9529d87']
```

✓ 0.1s Python

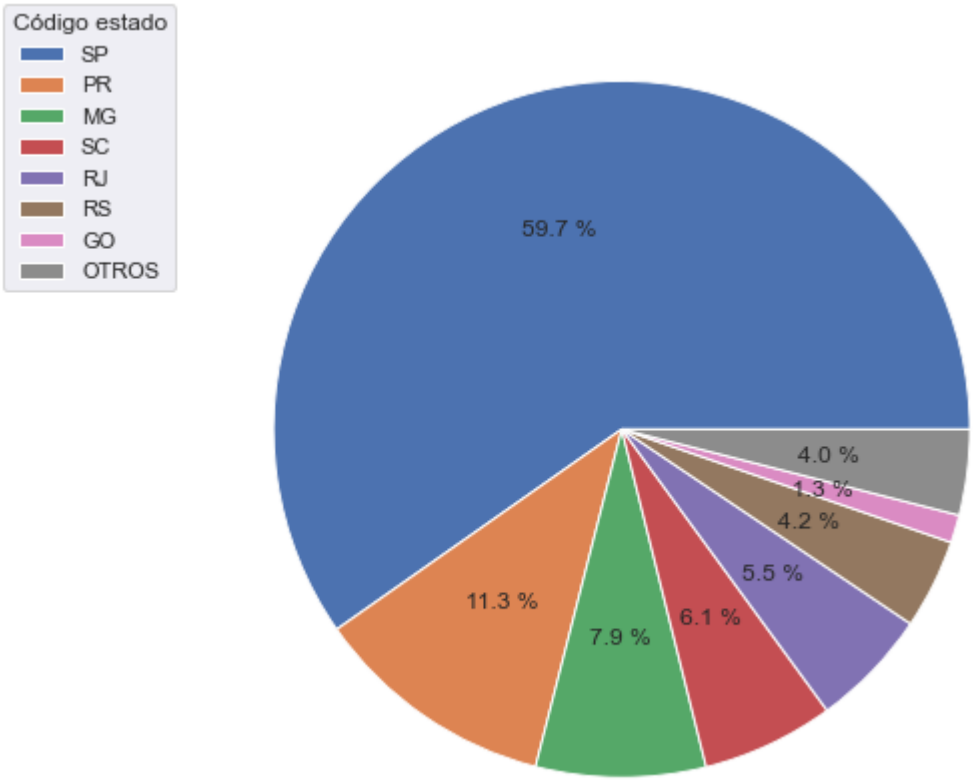
order_id	order_item_id	product_id	seller_id	shipping_limit_date	price	freight_value
02f9529d87	1	765a8070ece0f1383d0f5faf913dfb9b	218d46b86c1881d022bce9c68a7d4b15	2018-01-25 09:08:37	81.0	15.54
02f9529d87	2	a41e356c76fab66334f36de622ecbd3a	d9cb0052a666de5308b32f32ad5f1b1c	2018-01-25 09:08:37	99.3	7.77
02f9529d87	3	765a8070ece0f1383d0f5faf913dfb9b	218d46b86c1881d022bce9c68a7d4b15	2018-01-25 09:08:37	81.0	15.54

Sellers

```
sellers = pd.read_csv('data\olist_sellers_dataset.csv')
```

```
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   seller_id               3095 non-null   object
 1   seller_zip_code_prefix  3095 non-null   int64
 2   seller_city             3095 non-null   object
 3   seller_state            3095 non-null   object
dtypes: int64(1), object(3)
memory usage: 96.8+ KB
```

Distribución de los vendedores por estado



La mayor cantidad de vendedores se encuentra en SP (Sao Paolo)

Los Top 10 vendedores por categoría son

categoria: artes