sales data analysis report

In [2]:

```
import matplotlib
import pandas as pd
import numpy as np
import seaborn as sb
from matplotlib import pyplot as plt
import random as rnd
```

In [3]:

```
sd=pd.read_csv("Sales.csv") # reading csv file
sd
```

Out[3]:

	month	order_date	identifier	order_num	sub_order_num	quantity	order_status	manifesttime	sup_name	state	...	ship
0	11/1/2022	11/28/2022	kv4xc	4.334940e+11	433493748710_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	
1	11/1/2022	11/2/2022	kv4xc	4.565440e+11	456544057575_1	1	Delivered	11/3/2022	Turbo Traders	Madhya Pradesh	...	
2	11/1/2022	11/4/2022	kv4xc	8.348670e+11	834867460958_1	1	Delivered	11/5/2022	Turbo Traders	Madhya Pradesh	...	
3	11/1/2022	11/29/2022	kv4xc	2.551520e+11	255151569682_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	
4	11/1/2022	11/8/2022	kv4xc	8.056740e+11	805674293744_1	1	rto	11/8/2022	Turbo Traders	Madhya Pradesh	...	
...	...	...	...	...	...	...	...	...	...	...	...	
491	1/1/2023	1/28/2023	kv4xc	5.033090e+11	503309473282_1	1	Shipped	1/29/2023	Turbo Traders	Madhya Pradesh	...	
492	1/1/2023	1/5/2023	kv4xc	8.418950e+11	841895496191_1	1	rto	1/6/2023	Turbo Traders	Madhya Pradesh	...	
493	1/1/2023	1/13/2023	kv4xc	3.588180e+11	358817598782_1	1	Delivered	1/14/2023	Turbo Traders	Madhya Pradesh	...	
494	1/1/2023	1/28/2023	kv4xc	2.689940e+11	268994158296_1	1	Shipped	1/28/2023	Turbo Traders	Madhya Pradesh	...	
495	1/1/2023	1/21/2023	kv4xc	2.114660e+11	211465971448_1	1	Delivered	1/23/2023	Turbo Traders	Madhya Pradesh	...	

496 rows × 33 columns

In [3]:

```
sd.columns #display all the colunms of the data set
```

Out[3]:

```
Index(['month', 'order_date', 'identifier', 'order_num', 'sub_order_num',
       'quantity', 'order_status', 'manifesttime', 'sup_name', 'state', 'pin',
       'reseller_state', 'reseller_pin', 'end_customer_state',
       'end_customer_pin', 'gstin', 'hsn_code', 'gst_amount', 'gst_rate',
       'meesho_price', 'net_commission', 'commission_gst', 'adj',
       'shipping_charges_total', 'gst', 'taxable_shipping',
       'shipping_gst_18_percent', 'meesho_price_plus_shipping_charges_total',
       'tcs_taxable_amount', 'end_customer_state_new', 'financial_year',
       'month_number', 'supplier_id'],
      dtype='object')
```

In [4]:

```
len(sd.columns)
```

Out[4]:

```
33
```

Sales data column description

month: This column represents the month in which the order was placed.

order_date: This column represents the date on which the order was placed.

identifier: This column represents a unique identifier for the order.

order_num: This column represents the order number.

sub_order_num: This column represents the sub-order number.

quantity: This column represents the quantity of the product that was ordered.

order_status: This column represents the status of the order.

manifesttime: This column represents the time at which the order was shipped.

sup_name: This column represents the name of the supplier.

state: This column represents the state in which the supplier is located.

pin: This column represents the pin code of the supplier.

reseller_state: This column represents the state in which the reseller is located.

reseller_pin: This column represents the pin code of the reseller.

end_customer_state: This column represents the state in which the end customer is located.

end_customer_pin: This column represents the pin code of the end customer.

gstin: This column represents the GSTIN (Goods and Services Tax Identification Number) of the supplier.

hsn_code: This column represents the Harmonized System of Nomenclature (HSN) code for the product.

gst_amount:

This column represents the GST (Goods and Services Tax) amount.

gst_rate: This column represents the GST rate.

meesho_price: This column represents the price of the product charged by Meesho.

net_commission: This column represents the net commission earned by Meesho.

commission_gst: This column represents the GST on the commission earned by Meesho.

adj: This column represents any adjustment made to the order.

shipping charges_total: This column represents the total shipping charges.

gst: This column represents the GST on the shipping charges.

taxable_shipping: This column represents the taxable shipping amount.

shipping_gst 18 percent:

This column represents the GST on shipping at the rate of 18%.

meesho_price_plus_shipping_charges_total: This column represents the total price of the product and shipping charges charged by Meesho.

tcs_taxable_amount: This column represents the amount on which Tax Collected at Source (TCS) is applicable.

end_customer_state_new: This column represents the new state in which the end customer is located.

financial year: This column represents the financial year in which the order was placed.

month_number: This column represents the number of the month in which the order was placed.

supplier_id: This column represents the ID of the supplier.

analysis just by seeing data

The given data seems to be related to order details, with columns representing various attributes such as the month and order date, order status, quantity, shipping charges, GST rate, and supplier details. The dataset also includes information related to GST, HSN codes, commissions, and taxes, states and PIN codes of the reseller and end customer.

Analysis using Pandas and other libraries

In [40]:

```
sd.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 496 entries, 0 to 495
Data columns (total 33 columns):
 #   Column                                    Non-Null Count  Dtype
---  ------                                    --------------  -----
 0   month                                     488 non-null    object
 1   order_date                                496 non-null    object
 2   identifier                                496 non-null    object
 3   order_num                                 496 non-null    float64
 4   sub_order_num                             496 non-null    object
 5   quantity                                  496 non-null    int64
 6   order_status                              496 non-null    object
 7   manifesttime                              496 non-null    object
 8   sup_name                                  496 non-null    object
 9   state                                     496 non-null    object
 10  pin                                       496 non-null    int64
 11  reseller_state                            496 non-null    object
 12  reseller_pin                              316 non-null    float64
 13  end_customer_state                        496 non-null    object
 14  end_customer_pin                          496 non-null    int64
 15  gstin                                     496 non-null    object
 16  hsn_code                                  496 non-null    int64
 17  gst_amount                                496 non-null    float64
 18  gst_rate                                  496 non-null    int64
 19  meesho_price                              496 non-null    int64
 20  net_commission                            496 non-null    int64
 21  commission_gst                            496 non-null    int64
 22  adj                                       496 non-null    int64
 23  shipping_charges_total                    496 non-null    int64
 24  gst                                       496 non-null    float64
 25  taxable_shipping                          496 non-null    float64
 26  shipping_gst_18_percent                   496 non-null    float64
 27  meesho_price_plus_shipping_charges_total  496 non-null    int64
 28  tcs_taxable_amount                        496 non-null    float64
 29  end_customer_state_new                    496 non-null    object
 30  financial_year                            496 non-null    int64
 31  month_number                              496 non-null    int64
 32  supplier_id                               496 non-null    int64
dtypes: float64(7), int64(14), object(12)
memory usage: 128.0+ KB
```

In [41]:

```
sd.describe()
```

Out[41]:

	order_num	quantity	pin	reseller_pin	end_customer_pin	hsn_code	gst_amount	gst_rate	meesho_price	net_
count	4.960000e+02	496.000000	496.0	316.000000	496.000000	496.000000	496.000000	496.0	496.000000	
mean	5.474138e+11	1.028226	470004.0	529401.142405	527325.350806	947176.209677	14.537903	5.0	305.213710	
std	2.654584e+11	0.165785	0.0	178227.521561	179472.693521	4727.185866	9.934550	0.0	208.621146	
min	1.003620e+11	1.000000	470004.0	110008.000000	110009.000000	940500.000000	5.900000	5.0	124.000000	
25%	3.156818e+11	1.000000	470004.0	411044.250000	411014.750000	940530.000000	9.940000	5.0	208.750000	
50%	5.378720e+11	1.000000	470004.0	533224.500000	534106.000000	950510.000000	12.860000	5.0	270.000000	
75%	7.683985e+11	1.000000	470004.0	674260.750000	673098.000000	950510.000000	13.060000	5.0	274.000000	
max	9.994300e+11	2.000000	470004.0	854304.000000	852122.000000	950590.000000	92.580000	5.0	1944.000000	

8 rows × 21 columns

In [42]:

```
sd.isnull().sum() # to see number of null values And columns
```

Out[42]:

```
month                                        8
order_date                                   0
identifier                                   0
order_num                                    0
sub_order_num                                0
quantity                                     0
order_status                                 0
manifesttime                                 0
sup_name                                     0
state                                        0
pin                                          0
reseller_state                               0
reseller_pin                               180
end_customer_state                           0
end_customer_pin                             0
gstin                                        0
hsn_code                                     0
gst_amount                                   0
gst_rate                                     0
meesho_price                                 0
net_commission                               0
commission_gst                               0
adj                                          0
shipping_charges_total                       0
gst                                          0
taxable_shipping                             0
shipping_gst_18_percent                      0
meesho_price_plus_shipping_charges_total     0
tcs_taxable_amount                           0
end_customer_state_new                       0
financial_year                               0
month_number                                 0
supplier_id                                  0
dtype: int64
```

In [43]:

```
sd
```

Out[43]:

	month	order_date	identifier	order_num	sub_order_num	quantity	order_status	manifesttime	sup_name	state	...	ship
0	11/1/2022	11/28/2022	kv4xc	4.334940e+11	433493748710_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	
1	11/1/2022	11/2/2022	kv4xc	4.565440e+11	456544057575_1	1	Delivered	11/3/2022	Turbo Traders	Madhya Pradesh	...	
2	11/1/2022	11/4/2022	kv4xc	8.348670e+11	834867460958_1	1	Delivered	11/5/2022	Turbo Traders	Madhya Pradesh	...	
3	11/1/2022	11/29/2022	kv4xc	2.551520e+11	255151569682_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	
4	11/1/2022	11/8/2022	kv4xc	8.056740e+11	805674293744_1	1	rto	11/8/2022	Turbo Traders	Madhya Pradesh	...	
...	...	...	...	...	...	...	...	...	...	...	...	
491	1/1/2023	1/28/2023	kv4xc	5.033090e+11	503309473282_1	1	Shipped	1/29/2023	Turbo Traders	Madhya Pradesh	...	
492	1/1/2023	1/5/2023	kv4xc	8.418950e+11	841895496191_1	1	rto	1/6/2023	Turbo Traders	Madhya Pradesh	...	
493	1/1/2023	1/13/2023	kv4xc	3.588180e+11	358817598782_1	1	Delivered	1/14/2023	Turbo Traders	Madhya Pradesh	...	
494	1/1/2023	1/28/2023	kv4xc	2.689940e+11	268994158296_1	1	Shipped	1/28/2023	Turbo Traders	Madhya Pradesh	...	
495	1/1/2023	1/21/2023	kv4xc	2.114660e+11	211465971448_1	1	Delivered	1/23/2023	Turbo Traders	Madhya Pradesh	...	

496 rows × 33 columns

In [44]:

```
sd.columns
```

Out[44]:

```
Index(['month', 'order_date', 'identifier', 'order_num', 'sub_order_num',
       'quantity', 'order_status', 'manifesttime', 'sup_name', 'state', 'pin',
       'reseller_state', 'reseller_pin', 'end_customer_state',
       'end_customer_pin', 'gstin', 'hsn_code', 'gst_amount', 'gst_rate',
       'meesho_price', 'net_commission', 'commission_gst', 'adj',
       'shipping_charges_total', 'gst', 'taxable_shipping',
       'shipping_gst_18_percent', 'meesho_price_plus_shipping_charges_total',
       'tcs_taxable_amount', 'end_customer_state_new', 'financial_year',
       'month_number', 'supplier_id'],
      dtype='object')
```

In [45]:

```
sd.quantity.value_counts() #number of quatity of products ordered by customers
```

Out[45]:

```
1    482
2     14
Name: quantity, dtype: int64
```

In [7]:

```
# 1.average quantity of ordered product

print("Average quantity of ordered product",sd.quantity.mean())
```

```
Average quantity of ordered product 1.028225806451613
```

In [123]:

```
# 2.how many orders are placed by customer in jan dec and nov

s=sd.month_number.value_counts()
s.index[0]
s
```

Out[123]:

```
12    288
1     127
11     81
Name: month_number, dtype: int64
```

In [121]:

```
#3. How many orders were shipped

l=sd.loc[sd.order_status=="Shipped"]
len(l)
```

Out[121]:

	month	order_date	identifier	order_num	sub_order_num	quantity	order_status	manifesttime	sup_name	state	...	shippi
0	11/1/2022	11/28/2022	kv4xc	4.334940e+11	433493748710_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	
3	11/1/2022	11/29/2022	kv4xc	2.551520e+11	255151569682_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	
5	11/1/2022	11/28/2022	kv4xc	4.406220e+11	440621895540_1	2	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	
12	11/1/2022	11/26/2022	kv4xc	8.159150e+11	815915111545_1	1	Shipped	11/27/2022	Turbo Traders	Madhya Pradesh	...	
20	11/1/2022	11/29/2022	kv4xc	3.791900e+11	379190321592_1	1	Shipped	11/30/2022	Turbo Traders	Madhya Pradesh	...	
...	...	...	...	...	...	...	...	...	...	...	...	
485	1/1/2023	1/20/2023	kv4xc	9.581740e+11	958174414085_1	1	Shipped	1/21/2023	Turbo Traders	Madhya Pradesh	...	
486	1/1/2023	1/29/2023	kv4xc	3.594210e+11	359421134529_1	1	Shipped	1/29/2023	Turbo Traders	Madhya Pradesh	...	
488	1/1/2023	1/29/2023	kv4xc	4.285380e+11	428537537902_1	1	Shipped	1/29/2023	Turbo Traders	Madhya Pradesh	...	
491	1/1/2023	1/28/2023	kv4xc	5.033090e+11	503309473282_1	1	Shipped	1/29/2023	Turbo Traders	Madhya Pradesh	...	
494	1/1/2023	1/28/2023	kv4xc	2.689940e+11	268994158296_1	1	Shipped	1/28/2023	Turbo Traders	Madhya Pradesh	...	

66 rows × 33 columns

In [54]:

```
# 3.How many orders were shipped in each state

l=sd.end_customer_state[sd.order_status=="Shipped"]
l.value_counts()
```

Out[54]:

```
Andhra Pradesh       8
West Bengal          7
Tamil Nadu           6
Telangana            4
Kerala               4
Karnataka            4
Maharashtra          4
Rajasthan            3
Jammu & Kashmir      3
Uttar Pradesh        3
Assam                2
Bihar                2
Odisha               2
Delhi                2
Jharkhand            2
Puducherry           1
Haryana              1
Punjab               1
Madhya Pradesh       1
telangana            1
Himachal Pradesh     1
Gujarat              1
Uttarakhand          1
Meghalaya            1
Arunachal Pradesh    1
Name: end_customer_state, dtype: int64
```

In [55]:

```
#5.which supplier has the highest number of Orders

sd.supplier_id.value_counts()
```

Out[55]:

```
733069    496
Name: supplier_id, dtype: int64
```

In [60]:

```
#6.how many order has adj greater than 70

l=sd.loc[sd.adj>70]
print(len(l))
l
```

151

Out[60]:

	month	order_date	identifier	order_num	sub_order_num	quantity	order_status	manifesttime	sup_name	state	...	ship
5	11/1/2022	11/28/2022	kv4xc	4.406220e+11	440621895540_1	2	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	
13	11/1/2022	11/7/2022	kv4xc	4.905530e+11	490552548982_4	1	Delivered	11/8/2022	Turbo Traders	Madhya Pradesh	...	
17	11/1/2022	11/1/2022	kv4xc	2.256680e+11	225667603238_1	1	Delivered	11/2/2022	Turbo Traders	Madhya Pradesh	...	
18	11/1/2022	11/2/2022	kv4xc	6.059510e+11	605950579795_1	1	Delivered	11/3/2022	Turbo Traders	Madhya Pradesh	...	
27	11/1/2022	11/16/2022	kv4xc	9.930920e+11	993092407731_1	1	Delivered	11/16/2022	Turbo Traders	Madhya Pradesh	...	
...	...	...	...	...	...	...	...	...	...	...	...	
489	1/1/2023	1/22/2023	kv4xc	2.966660e+11	296665755133_1	1	Delivered	1/23/2023	Turbo Traders	Madhya Pradesh	...	
490	1/1/2023	1/21/2023	kv4xc	5.012050e+11	501205269504_1	1	Delivered	1/21/2023	Turbo Traders	Madhya Pradesh	...	
492	1/1/2023	1/5/2023	kv4xc	8.418950e+11	841895496191_1	1	rto	1/6/2023	Turbo Traders	Madhya Pradesh	...	
493	1/1/2023	1/13/2023	kv4xc	3.588180e+11	358817598782_1	1	Delivered	1/14/2023	Turbo Traders	Madhya Pradesh	...	
495	1/1/2023	1/21/2023	kv4xc	2.114660e+11	211465971448_1	1	Delivered	1/23/2023	Turbo Traders	Madhya Pradesh	...	

151 rows x 33 columns

In [61]:

```
#7.average amount of adj on each order

sd.adj.mean()
```

Out[61]:

```
68.375
```

In [125]:

```
#8.which state has highest and lowest number of orders

l=sd.end_customer_state.value_counts()
lmax=sd.end_customer_state.value_counts().max()
l
      # gives whole list in asending order


print("highest orders : ",l.index[0])
print("lowest orders : ",l.index[-1])
```

```
highest orders :  Karnataka
lowest orders :  Up
```

In [7]:

```
#9.which state has highest number of order delivered

OD=sd.end_customer_state[sd.order_status=="Delivered"]
hOD=OD.value_counts()
print("highest orders : ",hOD.index[0])
hOD
```

```
highest orders :  Karnataka
```

Out[7]:

```
Karnataka                      45
Tamil Nadu                     39
Andhra Pradesh                 37
Kerala                         34
Maharashtra                    27
Uttar Pradesh                  21
Telangana                      21
Gujarat                        17
Goa                            10
Meghalaya                       8
Odisha                          8
Madhya Pradesh                  8
West Bengal                     8
Delhi                           5
Chhattisgarh                    5
Rajasthan                       5
Manipur                         4
Assam                           4
Uttarakhand                     4
Haryana                         4
tamilnadu                       3
Punjab                          3
Ap                              2
Jharkhand                       2
Jammu & Kashmir                 2
Others                          2
Tamilnadu                       1
Andaman and Nicobar Islands     1
Mizoram                         1
Puducherry                      1
J&k                             1
Mp                              1
Daman & Diu                     1
Andaman & Nicobar Islands       1
Up                              1
Name: end_customer_state, dtype: int64
```

In [64]:

```
#10.average gst rate for the products

sd.gst_rate.mean() #gst is same on each order
```

Out[64]:

```
5.0
```

In [67]:

```
#11.maximum ,minimum,average gst amount on products

print("maximum GST Amount : ",sd.gst_amount.max())
print("minimum GST Amount : ",sd.gst_amount.min())
print("Average GST Amount : ",sd.gst_amount.mean())
```

```
maximum GST Amount :  92.58
minimum GST Amount :  5.9
Average GST Amount :  14.537903225806456
```

In [70]:

```
#12.Average meesho price on each product

print("Average Meesho Price on each product: ", sd.meesho_price.mean())
```

```
Average Meesho Price :  305.21370967741933
```

In [82]:

```
#13.Maximum taxable charge
print("Maximum taxable charge : ",sd.taxable_shipping.max())

#what was the total taxable amount
print("Total taxable amount : ",sd.tcs_taxable_amount.sum())
```

```
Maximum taxable charge :  156.19047619999998
Total taxable amount :  142612.38094513002
```

In [8]:

```
#14.what was the most common order status

l=sd.order_status.value_counts()
print(l)
print("most common order status : ",l.index[0])
```

```
Delivered    337
Shipped       66
rto           49
Return        23
Cancelled     15
Exchange       3
Ordered        3
Name: order_status, dtype: int64
most common order status :  Delivered
```

In [12]:

```
#15.how many times the order was return by the customer

print("order was return by the customer ",len(sd[sd.order_status=="Return"]),"times.")
```

```
order was return by the customer  23 times.
```

In [13]:

```
#16.display states where order was Cancelled by the customer

l=sd.loc[sd.order_status=="Cancelled"]["end_customer_state"]
l.unique()
```

Out[13]:

```
array(['Puducherry', 'Nagaland', 'Bihar', 'Uttar Pradesh', 'Karnataka',
       'Andhra Pradesh', 'Tamil Nadu', 'Telangana', 'Chhattisgarh',
       'Rajasthan', 'Maharashtra'], dtype=object)
```

In [90]:

```
#17.display statewise  highest to lowest number of orders ordered by the customer

sd["end_customer_state"].value_counts()
```

Out[90]:

```
Karnataka                     58
Andhra Pradesh                51
Tamil Nadu                    50
Kerala                        44
Maharashtra                   35
Uttar Pradesh                 31
Telangana                     31
West Bengal                   25
Gujarat                       18
Odisha                        15
Goa                           12
Meghalaya                     11
Rajasthan                     11
Delhi                         10
Madhya Pradesh                10
Chhattisgarh                   8
Assam                          8
Bihar                          8
Jammu & Kashmir                6
Uttarakhand                    6
Jharkhand                      5
Punjab                         5
Haryana                        5
Manipur                        4
Puducherry                     4
Mizoram                        3
tamilnadu                      3
Ap                             3
Himachal Pradesh               2
Nagaland                       2
Others                         2
Andaman and Nicobar Islands    1
telangana                      1
J&k                            1
Chandigarh                     1
Daman & Diu                    1
Tamilnadu                      1
Mp                             1
Up                             1
Andaman & Nicobar Islands      1
Arunachal Pradesh              1
Name: end_customer_state, dtype: int64
```

In [40]:

```
#18.In dec 2022  total number of order Return

L=sd[(sd.order_status=="Return")&(sd.financial_year==2022)&(sd.month_number==12)]
print("Total number of order Return in december 2022 is: ",len(L))
```

```
Total number of order Return in december 2022 is:  10
```

Data Cleaning

In [8]:

```
sd.net_commission.value_counts()#zero "0"--->no net commission earned by meesho
```

Out[8]:

```
0    496
Name: net_commission, dtype: int64
```

In [21]:

```
print(sd.shipping_charges_total.value_counts())# no shipping charges are charged on products
```

```
0    496
Name: shipping_charges_total, dtype: int64
```

In [22]:

```
sd.commission_gst.value_counts()#contain same value 0--> drop column commission gst
```

Out[22]:

```
0    496
Name: commission_gst, dtype: int64
```

In [23]:

```
newsd=sd.drop(["net_commission","shipping_charges_total","commission_gst"],axis=1)
newsd
```

Out[23]:

	month	order_date	identifier	order_num	sub_order_num	quantity	order_status	manifesttime	sup_name	state	...	adj
0	11/1/2022	11/28/2022	kv4xc	4.334940e+11	433493748710_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	63
1	11/1/2022	11/2/2022	kv4xc	4.565440e+11	456544057575_1	1	Delivered	11/3/2022	Turbo Traders	Madhya Pradesh	...	62
2	11/1/2022	11/4/2022	kv4xc	8.348670e+11	834867460958_1	1	Delivered	11/5/2022	Turbo Traders	Madhya Pradesh	...	62
3	11/1/2022	11/29/2022	kv4xc	2.551520e+11	255151569682_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	63
4	11/1/2022	11/8/2022	kv4xc	8.056740e+11	805674293744_1	1	rto	11/8/2022	Turbo Traders	Madhya Pradesh	...	62
...	...	...	...	...	...	...	...	...	...	...	...	...
491	1/1/2023	1/28/2023	kv4xc	5.033090e+11	503309473282_1	1	Shipped	1/29/2023	Turbo Traders	Madhya Pradesh	...	67
492	1/1/2023	1/5/2023	kv4xc	8.418950e+11	841895496191_1	1	rto	1/6/2023	Turbo Traders	Madhya Pradesh	...	77
493	1/1/2023	1/13/2023	kv4xc	3.588180e+11	358817598782_1	1	Delivered	1/14/2023	Turbo Traders	Madhya Pradesh	...	77
494	1/1/2023	1/28/2023	kv4xc	2.689940e+11	268994158296_1	1	Shipped	1/28/2023	Turbo Traders	Madhya Pradesh	...	67
495	1/1/2023	1/21/2023	kv4xc	2.114660e+11	211465971448_1	1	Delivered	1/23/2023	Turbo Traders	Madhya Pradesh	...	72

496 rows × 30 columns

In [45]:

```
newsd.isnull().sum()# month contain 8 null values
```

Out[45]:

```
month                                    8
order_date                               0
identifier                               0
order_num                                0
sub_order_num                            0
quantity                                 0
order_status                             0
manifesttime                             0
sup_name                                 0
state                                    0
pin                                      0
reseller_state                           0
reseller_pin                           180
end_customer_state                       0
end_customer_pin                         0
gstin                                    0
hsn_code                                 0
gst_amount                               0
gst_rate                                 0
meesho_price                             0
adj                                      0
gst                                      0
taxable_shipping                         0
shipping_gst_18_percent                  0
meesho_price_plus_shipping_charges_total 0
tcs_taxable_amount                       0
end_customer_state_new                   0
financial_year                           0
month_number                             0
supplier_id                              0
dtype: int64
```

In [48]:

```
sd1 = newsd.dropna(subset=["month"]) # deleting all the 8 null value rows fron the dataset
sd1
```

Out[48]:

	month	order_date	identifier	order_num	sub_order_num	quantity	order_status	manifesttime	sup_name	state	...	adj	g
0	11/1/2022	11/28/2022	kv4xc	4.334940e+11	433493748710_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	63	1.
1	11/1/2022	11/2/2022	kv4xc	4.565440e+11	456544057575_1	1	Delivered	11/3/2022	Turbo Traders	Madhya Pradesh	...	62	1.
2	11/1/2022	11/4/2022	kv4xc	8.348670e+11	834867460958_1	1	Delivered	11/5/2022	Turbo Traders	Madhya Pradesh	...	62	1.
3	11/1/2022	11/29/2022	kv4xc	2.551520e+11	255151569682_1	1	Shipped	11/29/2022	Turbo Traders	Madhya Pradesh	...	63	1.
4	11/1/2022	11/8/2022	kv4xc	8.056740e+11	805674293744_1	1	rto	11/8/2022	Turbo Traders	Madhya Pradesh	...	62	1.
...	...	...	...	...	...	...	...	...	...	...	...	...	
491	1/1/2023	1/28/2023	kv4xc	5.033090e+11	503309473282_1	1	Shipped	1/29/2023	Turbo Traders	Madhya Pradesh	...	67	1.
492	1/1/2023	1/5/2023	kv4xc	8.418950e+11	841895496191_1	1	rto	1/6/2023	Turbo Traders	Madhya Pradesh	...	77	1.
493	1/1/2023	1/13/2023	kv4xc	3.588180e+11	358817598782_1	1	Delivered	1/14/2023	Turbo Traders	Madhya Pradesh	...	77	1.
494	1/1/2023	1/28/2023	kv4xc	2.689940e+11	268994158296_1	1	Shipped	1/28/2023	Turbo Traders	Madhya Pradesh	...	67	1.
495	1/1/2023	1/21/2023	kv4xc	2.114660e+11	211465971448_1	1	Delivered	1/23/2023	Turbo Traders	Madhya Pradesh	...	72	1.

488 rows × 30 columns

In [50]:

```
sd1.isnull().sum() # all null value rows from month is removed
```

Out[50]:

```
month                                        0
order_date                                   0
identifier                                   0
order_num                                    0
sub_order_num                                0
quantity                                     0
order_status                                 0
manifesttime                                 0
sup_name                                     0
state                                        0
pin                                          0
reseller_state                               0
reseller_pin                               177
end_customer_state                           0
end_customer_pin                             0
gstin                                        0
hsn_code                                     0
gst_amount                                   0
gst_rate                                     0
meesho_price                                 0
adj                                          0
gst                                          0
taxable_shipping                             0
shipping_gst_18_percent                      0
meesho_price_plus_shipping_charges_total     0
tcs_taxable_amount                           0
end_customer_state_new                       0
financial_year                               0
month_number                                 0
supplier_id                                  0
dtype: int64
```

In [57]:

```
#putting random values in place of null values in "reseller_pin" column

minrsl=sd1.     reseller_pin.min()
maxrsl=sd1.     reseller_pin.max()
randrsl=rnd.randrange(int(minrsl),int(maxrsl))
sd1["reseller_pin"].fillna(randrsl,inplace=True)
```

In [58]:

```
sd1.isnull().sum() #all null values are filled by random values
```

Out[58]:

```
month                                       0
order_date                                  0
identifier                                  0
order_num                                   0
sub_order_num                               0
quantity                                    0
order_status                                0
manifesttime                                0
sup_name                                    0
state                                       0
pin                                         0
reseller_state                              0
reseller_pin                                0
end_customer_state                          0
end_customer_pin                            0
gstin                                       0
hsn_code                                    0
gst_amount                                  0
gst_rate                                    0
meesho_price                                0
adj                                         0
gst                                         0
taxable_shipping                            0
shipping_gst_18_percent                     0
meesho_price_plus_shipping_charges_total    0
tcs_taxable_amount                          0
end_customer_state_new                      0
financial_year                              0
month_number                                0
supplier_id                                 0
dtype: int64
```

In [60]:

```
sd1.duplicated().unique()# no duplicate value is present in data
```

Out[60]:

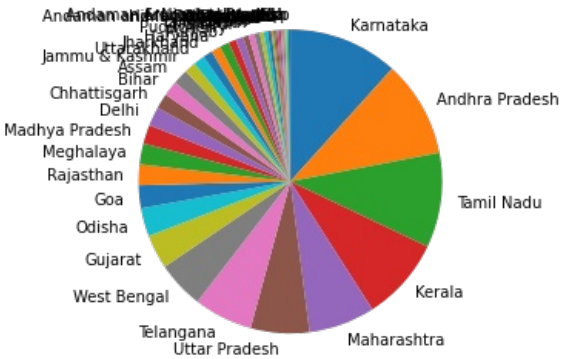
```
array([False])
```

Matplotlib

In [65]:

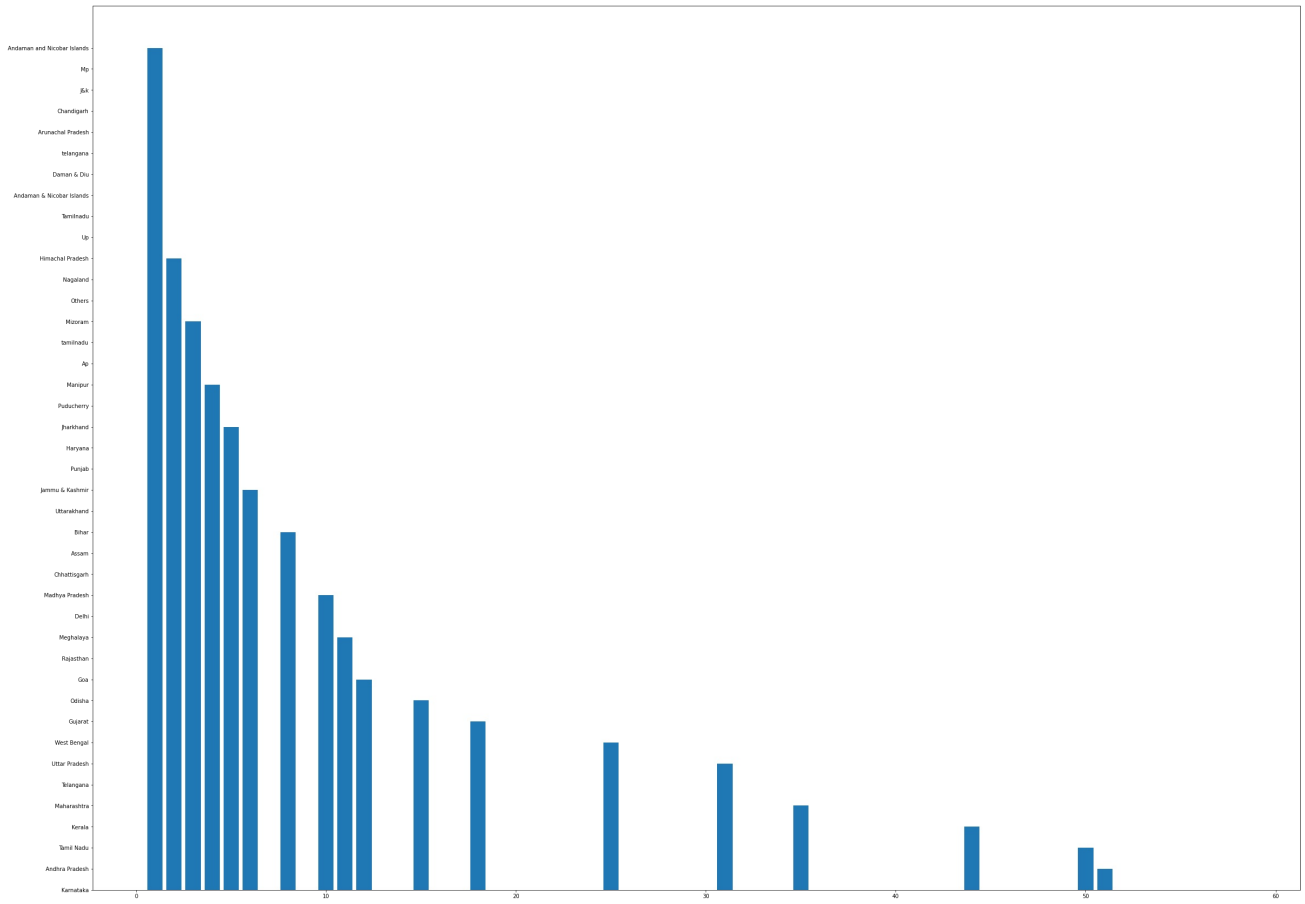
```
#1.which state had the highest and lowest number of orders,plot on a pie chart

l=sd.end_customer_state.value_counts()
plt.pie(l,labels=l.index,startangle=90,counterclock=False);
plt.axis('square');
```



In [20]:

```
#Same through Bar graph
l=sd.end_customer_state.value_counts()
labels=l.index
# plt.xlabel("count",fontsize=16,labelpad=20);
# plt.ylabel("State",fontsize=16,labelpad=20);
plt.figure(figsize=(40,30));
plt.bar(l,labels);
```



In [78]:

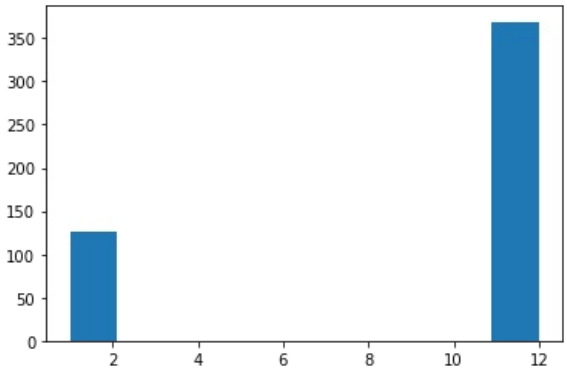
```
# 2.number of orders placed in each month

ordmnth=sd.month_number
print(sd.month_number.value_counts())
plt.hist(ordmnth)
```

```
12    288
1     127
11     81
Name: month_number, dtype: int64
```

Out[78]:

```
(array([127.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0., 369.]),
 array([ 1. ,  2.1,  3.2,  4.3,  5.4,  6.5,  7.6,  8.7,  9.8, 10.9, 12. ]),
 <a list of 10 Patch objects>)
```



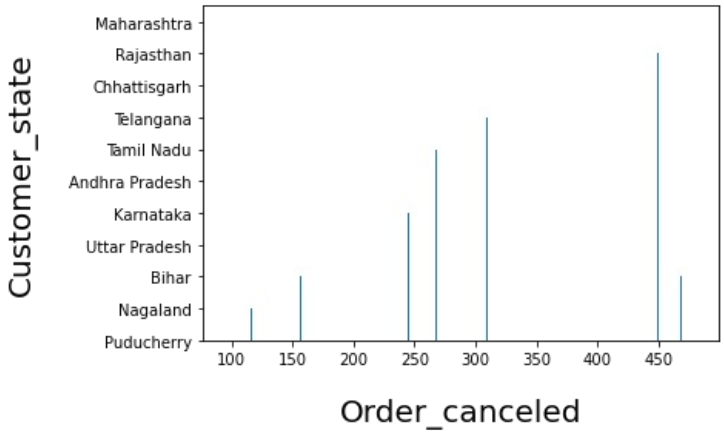
In [88]:

```
# 3. display states where order was Cancelled by the customer

nsd=sd.loc[sd.order_status=="Cancelled"]["end_customer_state"]
nsd
OC=nsd.index
plt.xlabel("Order_canceled",fontsize=20,labelpad=20)
plt.ylabel("Customer_state",fontsize=20,labelpad=20)
plt.bar(OC,nsd)
```

Out[88]:

```
<BarContainer object of 15 artists>
```

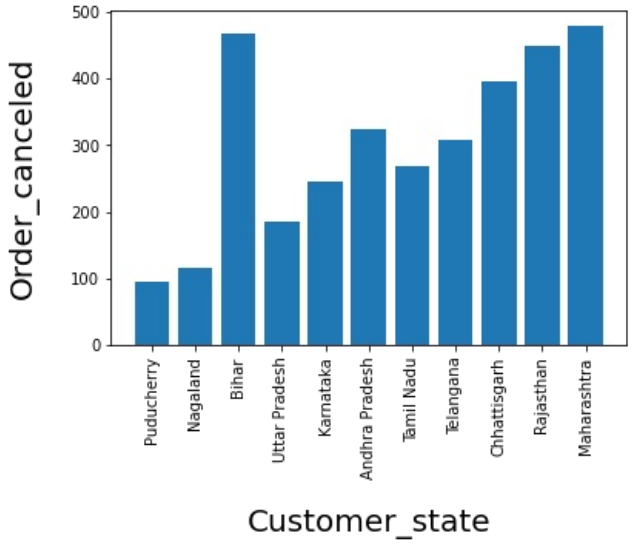


In [93]:

```
# 4. display states where order was Cancelled by the customer
nsd=sd.loc[sd.order_status=="Cancelled"]["end_customer_state"]
nsd
OC=nsd.index
plt.xlabel("Customer_state",fontsize=20,labelpad=20)
plt.ylabel("Order_canceled",fontsize=20,labelpad=20)
plt.bar(nsd,OC)
plt.xticks(rotation=90) # to remove overlaping of states 90 degree rotation
```

Out[93]:

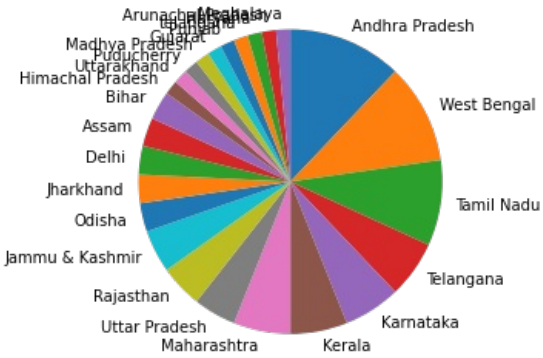
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
 <a list of 11 Text major ticklabel objects>)
```



In [127]:

```
# 5.orders were shipped in each state
l=sd.end_customer_state[sd.order_status=="Shipped"]
s=l.value_counts()

plt.pie(s,labels=s.index,startangle=90,counterclock=False);
plt.axis('square');
```



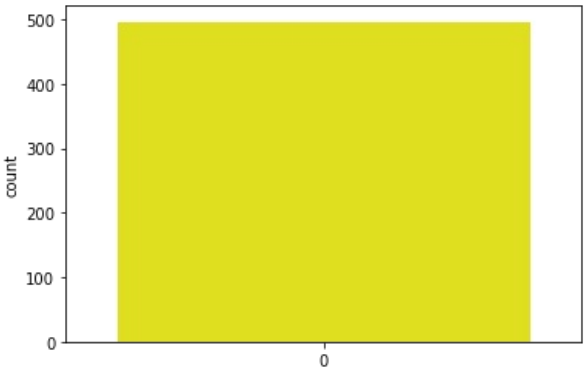
seaborn

In [101]:

```
# 6.changes in total shipping charge in different months
sb.countplot(data=sd,x="shipping_charges_total",color= "yellow")
plt.xlabel("")
   # shipping charge is constant in each month
```

Out[101]:

```
Text(0.5, 0, '')
```



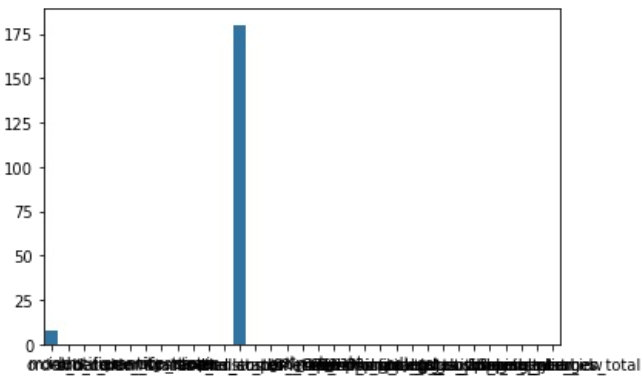
In [102]:

```
#7. total null values present in data thorugh bar graph
nullcount=sd.isnull().sum()
base_color=sb.color_palette()[0]

sb.barplot(nullcount.index.values,nullcount,color=base_color)
```

Out[102]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x21a7a8c54c0>
```



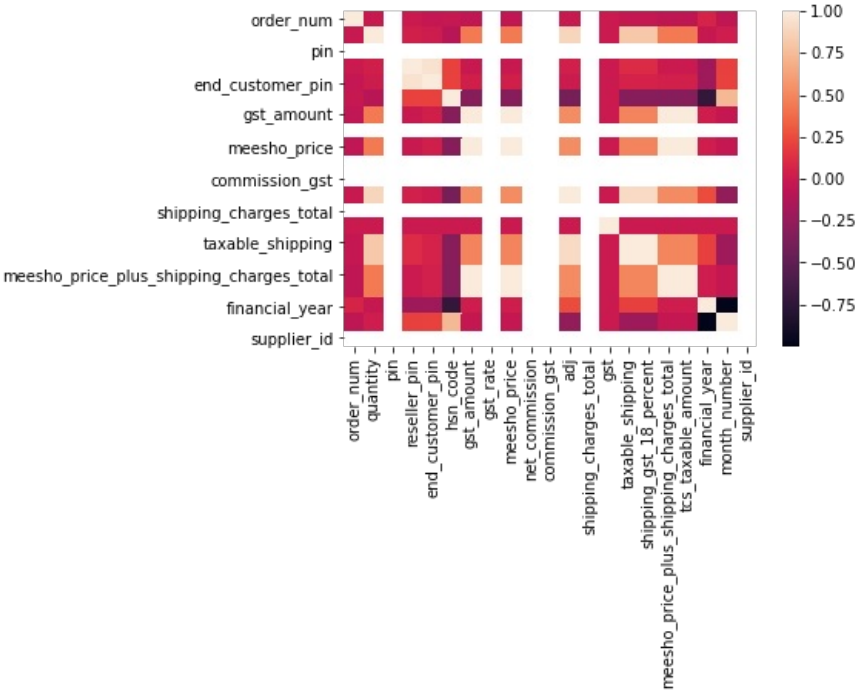
In [128]:

```
# 8.correlation btween columns

sb.heatmap(sd.corr())
```

Out[128]:

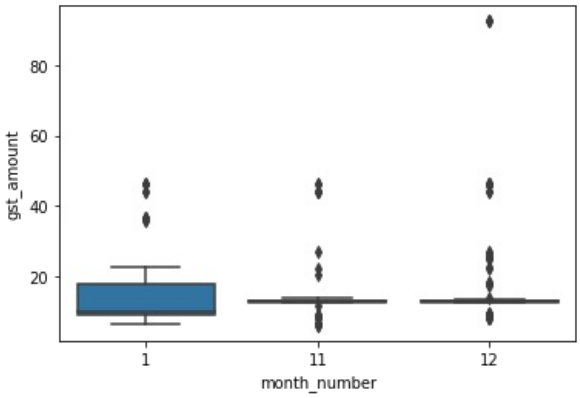
```
<matplotlib.axes._subplots.AxesSubplot at 0x21a7941c7c0>
```



In [21]:

```
#9.changes in gst amount in different month by box plot

sb.boxplot(x="month_number",y="gst_amount",data=sd)
plt.show()
```



In [ ]:

```
# 10

sb.pairplot(sd,hue="gst_amount")
plt.show()
```

In [ ]:

```

```