In [1]:
```
import pandas as pd
```
In [17]:
```
df=pd.read_csv('Inventory_Stock_Data.csv')
```
In [8]:
```
df.shape
```
Out[8]:
```
(118, 11)
```
In [15]:
```
#for numerical columns for invetory_stock_delevery
df[['product id','avg lead time','avg order qty','current stock','max lead time','max order qty','mod','reorder point','safety stock']]
```
Out[15]:

	product id	avg lead time	avg order qty	current stock	max lead time	max order qty	mod	reorder point	safety stock
0	19	10	1	69	30	1	1	35	25
1	24	10	3	175	30	5	0	175	144
2	35	10	1	65	30	1	1	35	25
3	37	10	3	328	30	5	1	175	146
4	44	10	3	175	30	5	0	175	145
...	...	...	...	...	...	...	...	...	...
113	1359	10	1	69	30	1	1	35	25
114	1360	10	1	35	30	1	0	35	25
115	1361	10	1	61	30	1	1	35	25
116	1362	10	1	35	30	1	0	35	25
117	1363	10	1	63	30	1	1	35	25

118 rows × 9 columns

In [16]:
```
#for categorical columns for invetory_stock_delevery
df[['order-now','product name']]
```
Out[16]:

	order-now	product name
0	green	Nike Men's Fingertrap Max Training Shoe
1	orange	Elevation Training Mask 2.0
2	green	adidas Brazuca 2014 Official Match Ball
3	green	adidas Kids' F5 Messi FG Soccer Cleat
4	orange	adidas Men's F10 Messi TRX FG Soccer Cleat
...	...	...
113	green	Adult dog supplies
114	orange	Smart watch
115	green	Toys
116	orange	Fighting video games
117	green	Summer dresses

118 rows × 2 columns

In [18]:
```
df1=pd.read_csv('Sales_Shipment_Data.csv')
```
In [19]:
```
df1.shape
```
Out[19]:
```
(180519, 46)
```
In [24]:
```
#for numerical columns for Sales_Shipment_Data
df1[['Department Id','Order Id','Product Id','shipping date (DateOrders)','Benefit per order','Days for shipment (scheduled)','Days for shipping (real)','Latit
```
Out[24]:

	Department Id	Order Id	Product Id	shipping date (DateOrders)	Benefit per order	Days for shipment (scheduled)	Days for shipping (real)	Latitude	Longitude	Order Customer Id	...	Order Item Discount	Order Item Discount Rate	Order Item Id	Order Item Product Price	Order Item Profit Ratio	Order Item Quantity	Order Item Total
0	5	53963	502	3/2/2017 17:17	9.610000	2	4	33.425201	-104.522080	5673	...	15.000000	0.15	134920	50.000000	0.11	2	85.000000
1	5	3908	502	3/3/2015 0:47	9.490000	2	4	34.034836	-118.260025	9723	...	16.000000	0.16	9722	50.000000	0.11	2	84.000000
2	5	52009	502	2/3/2017 4:43	30.709999	2	5	41.623779	-87.648697	9511	...	5.500000	0.06	129954	50.000000	0.33	2	94.500000
3	5	1179	502	1/23/2015 4:42	27.549999	2	5	41.904102	-87.937843	3326	...	5.000000	0.05	2929	50.000000	0.29	2	95.000000
4	5	56019	502	4/2/2017 17:36	47.500000	2	5	36.225159	-115.195183	907	...	5.000000	0.05	140056	50.000000	0.50	2	95.000000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
180514	4	49890	365	1/4/2017 6:20	-302.350006	4	6	33.320156	-117.156357	9615	...	47.990002	0.16	124746	59.990002	-1.20	5	251.960007
180515	4	29410	365	3/11/2016 7:16	117.010002	4	6	47.678699	-122.321762	1971	...	50.990002	0.17	73577	59.990002	0.47	5	248.960007
180516	3	24350	191	12/28/2015 10:32	-46.869999	4	6	33.823872	-118.007393	7663	...	124.989998	0.25	60910	99.989998	-0.13	5	374.959992
180517	4	45725	365	11/4/2016 11:09	125.260002	4	6	32.719875	-117.139725	415	...	38.990002	0.13	114288	59.990002	0.48	5	260.959992
180518	4	24050	365	12/24/2015 1:26	117.010002	4	6	38.495419	-121.479187	7141	...	50.990002	0.17	60174	59.990002	0.47	5	248.960007

180519 rows × 21 columns

In [25]:
```
#for categorical columns for Sales_Shipment_Data
df1[['Category Name','Class','Customer City','Customer Country','Customer Fname','Customer Lname','Customer Segment','Customer State','Customer Street','Delive
```
Out[25]:

	Category Name	Class	Customer City	Customer Country	Customer Fname	Customer Lname	Customer Segment	Customer State	Customer Street	Delivery Status	Department Name	Market	Order City	Order Country	Order Region	Order State
0	Women's Apparel	Moderate Value-Moderate Number	Roswell	EE. UU.	Jean	Dunn	Corporate	NM	8837 Blue Hills Harbour	Late	Golf	LATAM	Tlalpan	México	Central America	Distrito Federal
1	Women's Apparel	Moderate Value-Moderate Number	Los Angeles	EE. UU.	Rose	White	Corporate	CA	1396 Silver Berry Impasse	Late	Golf	LATAM	Soyapango	El Salvador	Central America	San Salvador
2	Women's Apparel	Moderate Value-Moderate Number	Harvey	EE. UU.	Joshua	Brady	Corporate	IL	8414 Wishing Horse Acres	Late	Golf	LATAM	Managua	Nicaragua	Central America	Managua
3	Women's Apparel	Moderate Value-Moderate Number	Elmhurst	EE. UU.	Mary	Smith	Corporate	IL	1269 Tawny Key	Late	Golf	LATAM	Poza Rica de Hidalgo	México	Central America	Veracruz
4	Women's Apparel	Moderate Value-Moderate Number	North Las Vegas	EE. UU.	Samantha	Kline	Corporate	NV	6396 Blue Gate Parkway	Late	Golf	LATAM	San Pedro Sula	Honduras	Central America	Cortés
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
180514	Cleats	High Value-Small Number	Escondido	EE. UU.	Charles	Smith	Corporate	CA	4088 Stony Vista	Late	Apparel	Pacific Asia	Haifa	Israel	West Asia	Haifa
180515	Cleats	High Value-Small Number	Seattle	EE. UU.	Richard	Stokes	Corporate	WA	2865 Blue Rise Way	Late	Apparel	Pacific Asia	Yuyao	China	Eastern Asia	Zhejiang
180516	Cardio Equipment	High Value-Small Number	Anaheim	EE. UU.	Mary	Smith	Corporate	CA	2792 Stony Range	Late	Footwear	Pacific Asia	Rangún	Myanmar (Birmania)	Southeast Asia	Rangún
180517	Cleats	High Value-Small Number	San Diego	EE. UU.	Mary	Allen	Corporate	CA	3983 Golden By-pass	Late	Apparel	Pacific Asia	Kahramanmara?	Turquía	West Asia	Kahramanmara?
180518	Cleats	High Value-Small Number	Sacramento	EE. UU.	Mary	Jackson	Corporate	CA	7760 Golden Limits	Late	Apparel	Pacific Asia	Jiangmen	China	Eastern Asia	Guangdong

180519 rows × 20 columns

In [ ]:
```
import pandas as pd
```
In [ ]:
```
import numpy as np
```
In [ ]:
```
df=read.csv('Inventory_Stock_Data.csv')
```
In [ ]:
```
df.shape
```
In [1]:
```
import pandas as pd
```
In [2]:
```
import numpy as np
```
In [3]:
```
df=pd.read_csv('Inventory_Stock_Data.csv')
```
In [7]:
```
sum_cols =['product id','avg lead time','avg order qty','current stock','max lead time','max order qty','mod','reorder point','safety stock']
df[sum_cols].sum()
```
Out[7]:
```
product id       81370
avg lead time     1180
avg order qty      246
current stock    18965
max lead time     3540
max order qty      374
mod                 64
reorder point    13090
safety stock     10659
dtype: int64
```
In [ ]:

In [6]:
```
c
```
In [7]:
```
import numpy as np
```
In [8]:
```
df1=pd.read_csv('Sales_Shipment_Data.csv')
```
In [9]:
```
df1.shape
```
Out[9]:
```
(180519, 46)
```
In [10]:
```
sum_colss=['Department Id','Order Id','Product Id','shipping date (DateOrders)','Benefit per order','Days for shipment (scheduled)','Days for shipping (real)',
df1[sum_colss].sum()
```
Out[10]:
```
Department Id                                                          982648
Order Id                                                           6538740246
Product Id                                                          125011170
shipping date (DateOrders)       3/2/2017 17:173/3/2015 0:472/3/2017 4:431/23/2...
Benefit per order                                                3966902.97405
Days for shipment (scheduled)                                          529254
Days for shipping (real)                                               631393
Latitude                                                       5365016.495643
Longitude                                                    -15328892.678974
Order Customer Id                                                  1207921135
Order Item Cardprod Id                                              125011170
Order Item Discount                                            3730378.402593
Order Item Discount Rate                                          18353.04009
Order Item Id                                                     16293644940
Order Item Product Price                                      25495158.680579
Order Item Profit Ratio                                          21779.009992
Order Item Quantity                                                    384079
Order Item Total                                              33054402.380217
Order Profit Per Order                                          3966902.97405
Product Price                                                 25495158.680579
Sales per customer                                            33054402.380217
dtype: object
```
In [14]:
```
df1['Order status'].value_counts()['Complete']
```
Out[14]:
```
59491
```
In [3]:
```
import pandas as pd
```
import numpy as np

In [4]:
```
df1=pd.read_csv('Sales_Shipment_Data.csv')
```
In [5]:
```
df1.shape
```
Out[5]:
```
(180519, 46)
```
In [7]:
```
df1['Delivery Status'].value_counts()['Late']
```
Out[7]:
```
98977
```
In [8]:
```
df1['Delivery Status'].value_counts()['Advance']
```
Out[8]:
```
41592
```
In [9]:
```
df1['Delivery Status'].value_counts()['Canceled']
```
Out[9]:
```
7754
```
In [10]:
```
df1['Delivery Status'].value_counts()['On time']
```
Out[10]:
```
32196
```
In [13]:
```
import pandas as pd
```
In [14]:
```
import numpy as np
```
In [ ]:

In [17]:

In [18]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [2]:
```
import pandas as pd
```
In [3]:
```
import numpy as np
```
In [4]:
```
df1=pd.read_csv('Sales_Shipment_Data.csv')
```
In [5]:
```
df1[date_time]=[['order date (DateOrders)','shipping date (DateOrders)']]
```
In [9]:
```
date_time
```
Out[9]:

	order date (DateOrders)	shipping date (DateOrders)
0	2/26/2017 17:17	3/2/2017 17:17
1	2/27/2015 0:47	3/3/2015 0:47
2	1/29/2017 4:43	2/3/2017 4:43
3	1/18/2015 4:42	1/23/2015 4:42
4	3/28/2017 17:36	4/2/2017 17:36
...	...	...
180514	12/29/2016 6:20	1/4/2017 6:20
180515	3/5/2016 7:16	3/11/2016 7:16
180516	12/22/2015 10:32	12/28/2015 10:32
180517	10/29/2016 11:09	11/4/2016 11:09
180518	12/18/2015 1:26	12/24/2015 1:26

180519 rows × 2 columns

In [18]:
```
df1['shipping date (DateOrders)']=pd.to_datetime(df1['shipping date (DateOrders)'])
```
In [19]:
```
df1['shipping date (DateOrders)']
```
Out[19]:
```
0        2017-03-02 17:17:00
1        2015-03-03 00:47:00
2        2017-02-03 04:43:00
3        2015-01-23 04:42:00
4        2017-04-02 17:36:00
                 ...
180514   2017-01-04 06:20:00
180515   2016-03-11 07:16:00
180516   2015-12-28 10:32:00
180517   2016-11-04 11:09:00
180518   2015-12-24 01:26:00
Name: shipping date (DateOrders), Length: 180519, dtype: datetime64[ns]
```
In [2]:
```
import pandas as pd
```
import numpy as np

In [3]:
```
df1=pd.read_csv('Sales_Shipment_Data.csv')
```
In [4]:
```
df1
```
Out[4]:

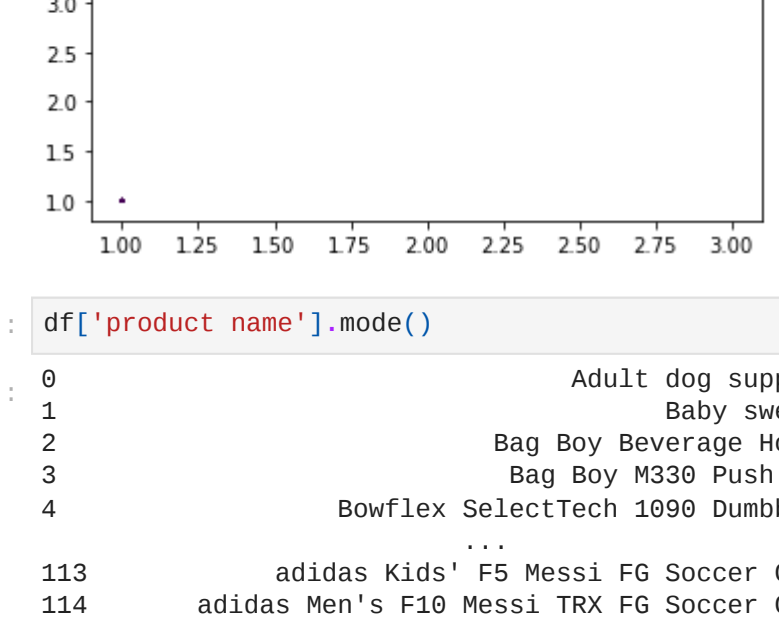
	Product Category Id	Category Name	Class	Customer City	Customer Country	Customer Fname	Customer Id	Customer Lname	Customer Segment	Customer State	...	Order Item Discount Rate	Order Item Id	Order Item Product Price	Order Item Profit Ratio	Order Item Quantity	Order Item Total	Order Profit Per Order
0	24	Women's Apparel	Moderate Value-Moderate Number	Roswell	EE. UU.	Jean	5673	Dunn	Corporate	NM	...	0.15	134920	50.000000	0.11	2	85.000000	9.610000
1	24	Women's Apparel	Moderate Value-Moderate Number	Los Angeles	EE. UU.	Rose	9723	White	Corporate	CA	...	0.16	9722	50.000000	0.11	2	84.000000	9.490000
2	24	Women's Apparel	Moderate Value-Moderate Number	Harvey	EE. UU.	Joshua	9511	Brady	Corporate	IL	...	0.06	129954	50.000000	0.33	2	94.500000	30.709999
3	24	Women's Apparel	Moderate Value-Moderate Number	Elmhurst	EE. UU.	Mary	3326	Smith	Corporate	IL	...	0.05	2929	50.000000	0.29	2	95.000000	27.549999
4	24	Women's Apparel	Moderate Value-Moderate Number	North Las Vegas	EE. UU.	Samantha	907	Kline	Corporate	NV	...	0.05	140056	50.000000	0.50	2	95.000000	47.500000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
180514	17	Cleats	High Value-Small Number	Escondido	EE. UU.	Charles	9615	Smith	Corporate	CA	...	0.16	124746	59.990002	-1.20	5	251.960007	-302.350006
180515	17	Cleats	High Value-Small Number	Seattle	EE. UU.	Richard	1971	Stokes	Corporate	WA	...	0.17	73577	59.990002	0.47	5	248.960007	117.010002
180516	9	Cardio Equipment	High Value-Small Number	Anaheim	EE. UU.	Mary	7663	Smith	Corporate	CA	...	0.25	60910	99.989998	-0.13	5	374.959992	-46.869999
180517	17	Cleats	High Value-Small Number	San Diego	EE. UU.	Mary	415	Allen	Corporate	CA	...	0.13	114288	59.990002	0.48	5	260.959992	125.260002
180518	17	Cleats	High Value-Small Number	Sacramento	EE. UU.	Mary	7141	Jackson	Corporate	CA	...	0.17	60174	59.990002	0.47	5	248.960007	117.010002

180519 rows × 46 columns

In [17]:
```
df1.groupby('Delivery Status')['order date (DateOrders)'].resample('W').count().loc['Late']
```
```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Input In [17], in <cell line: 1>()
----> 1 df1.groupby('Delivery Status')['order date (DateOrders)'].resample('W').count().loc['Late']

File ~\anaconda3\lib\site-packages\pandas\core\groupby\groupby.py:2417, in GroupBy.resample(self, rule, *args, **kwargs)
   2319 """
   2320 Provide resampling when using a TimeGrouper.
   2321
   (...)
   2413 5   2000-01-01 00:03:00  5  1
   2414 """
   2415 from pandas.core.resample import get_resampler_for_grouping
-> 2417 return get_resampler_for_grouping(self, rule, *args, **kwargs)

File ~\anaconda3\lib\site-packages\pandas\core\resample.py:1437, in get_resampler_for_grouping(groupby, rule, how, fill_method, limit, kind, on, **kwargs)
   1435 # .resample uses 'on' similar to how .groupby uses 'key'
   1436 tg = TimeGrouper(freq=rule, key=on, **kwargs)
-> 1437 resampler = tg._get_resampler(groupby.obj, kind=kind)
   1438 return resampler._get_resampler_for_grouping(groupby=groupby)

File ~\anaconda3\lib\site-packages\pandas\core\resample.py:1599, in TimeGrouper._get_resampler(self, obj, kind)
   1596 elif isinstance(ax, TimedeltaIndex):
   1597     return TimedeltaIndexResampler(obj, groupby=self, axis=self.axis)
-> 1599 raise TypeError(
   1600     "Only valid with DatetimeIndex, "
   1601     "TimedeltaIndex or PeriodIndex, "
   1602     f"but got an instance of '{type(ax).__name__}'"
   1603 )

TypeError: Only valid with DatetimeIndex, TimedeltaIndex or PeriodIndex, but got an instance of 'RangeIndex'
```
In [15]:
```
df1['order date (DateOrders)']=pd.to_datetime(df1['order date (DateOrders)'])
```
In [20]:
```
df1['order date (DateOrders)']
```
Out[20]:
```
0        2017-02-26 17:17:00
1        2015-02-27 00:47:00
2        2017-01-29 04:43:00
3        2015-01-18 04:42:00
4        2017-03-28 17:36:00
                 ...
180514   2016-12-29 06:20:00
180515   2016-03-05 07:16:00
180516   2015-12-22 10:32:00
180517   2016-10-29 11:09:00
180518   2015-12-18 01:26:00
Name: order date (DateOrders), Length: 180519, dtype: datetime64[ns]
```
In [1]:
```
import pandas as pd
```
In [2]:
```
import numpy as np
```
In [3]:
```
from sklearn.cluster import KMeans
```
In [ ]:
```
import sklearn.preprocessing import MinMaxscaler
```
In [6]:
```
from matplotlib import pyplot as plt
```
In [7]:
```
%matplotlib inline
```
df=pd.read_csv('Inventory_Stock_Data.csv')

In [9]:
```
df=pd.read_csv('Inventory_Stock_Data.csv')
```
In [10]:
```
df.head()
```
Out[10]:

	order-now	product id	product name	avg lead time	avg order qty	current stock	max lead time	max order qty	mod	reorder point	safety stock
0	green	19	Nike Men's Fingertrap Max Training Shoe	10	1	69	30	1	1	35	25
1	orange	24	Elevation Training Mask 2.0	10	3	175	30	5	0	175	144
2	green	35	adidas Brazuca 2014 Official Match Ball	10	1	65	30	1	1	35	25
3	green	37	adidas Kids' F5 Messi FG Soccer Cleat	10	3	328	30	5	1	175	146
4	green	44	adidas Men's F10 Messi TRX FG Soccer Cleat	10	3	175	30	5	0	175	145

In [13]:
```
plt.scatter(df['avg lead time'],df['current stock'],df['max lead time'])
```
Out[13]:
```
<matplotlib.collections.PathCollection at 0x1f9475cbb50>
```


In [14]:
```
plt.scatter(df['avg order qty'],df['max order qty'],df['mod'],df['safety stock'])
```
Out[14]:
```
<matplotlib.collections.PathCollection at 0x1f947d19fa0>
```


In [15]:
```
df['product name'].mode()
```
Out[15]:
```
0                              Adult dog supplies
1                                    Baby sweater
2                         Bag Boy Beverage Holder
3                            Bag Boy M330 Push Cart
4               Bowflex SelectTech 1090 Dumbbells
                          ...
113           adidas Kids' F5 Messi FG Soccer Cleat
114       adidas Men's F10 Messi TRX FG Soccer Cleat
115        adidas Men's Germany Black Crest Away Tee
116    adidas Youth Germany Black/Red Away Match Soc
117                  insta-bed Neverflat Air Mattress
Name: product name, Length: 118, dtype: object
```
In [16]:
```
import pandas as pd
```
In [17]:
```
import numpy as np
```
In [18]:
```
df1=pd.read_csv('Sales_Shipment_Data.csv')
```
In [21]:
```
df1.head()
```
Out[21]:

	Product Category Id	Category Name	Class	Customer City	Customer Country	Customer Fname	Customer Id	Customer Lname	Customer Segment	Customer State	...	Order Item Discount Rate	Order Item Id	Order Item Product Price	Order Item Profit Ratio	Order Item Quantity	Order Item Total	Order Profit Per Order	Product Price	Sales
0	24	Women's Apparel	Moderate Value-Moderate Number	Roswell	EE. UU.	Jean	5673	Dunn	Corporate	NM	...	0.15	134920	50.0	0.11	2	85.0	9.610000	50.0	100.0
1	24	Women's Apparel	Moderate Value-Moderate Number	Los Angeles	EE. UU.	Rose	9723	White	Corporate	CA	...	0.16	9722	50.0	0.11	2	84.0	9.490000	50.0	100.0
2	24	Women's Apparel	Moderate Value-Moderate Number	Harvey	EE. UU.	Joshua	9511	Brady	Corporate	IL	...	0.06	129954	50.0	0.33	2	94.5	30.709999	50.0	100.0
3	24	Women's Apparel	Moderate Value-Moderate Number	Elmhurst	EE. UU.	Mary	3326	Smith	Corporate	IL	...	0.05	2929	50.0	0.29	2	95.0	27.549999	50.0	100.0
4	24	Women's Apparel	Moderate Value-Moderate Number	North Las Vegas	EE. UU.	Samantha	907	Kline	Corporate	NV	...	0.05	140056	50.0	0.50	2	95.0	47.500000	50.0	100.0

5 rows × 46 columns

In [30]:
```
df1['Product Name'].value_counts().sort_values()[::-1][:10]
```
Out[30]:
```
Perfect Fitness Perfect Rip Deck                 24515
Nike Men's CJ Elite 2 TD Football Cleat          22246
Nike Men's Dri-FIT Victory Golf Polo             21035
O'Brien Men's Neoprene Life Vest                 19298
Field & Stream Sportsman 16 Gun Fire Safe        17325
Pelican Sunstream 100 Kayak                      15500
Diamondback Women's Serene Classic Comfort Bi    13729
Nike Men's Free 5.0+ Running Shoe                12169
Under Armour Girls' Toddler Spine Surge Runni    10617
Fighting video games                               838
Name: Product Name, dtype: int64
```
In [29]:
```
df1['Category Name'].value_counts().sort_values()[::-1][:10]
```
Out[29]:
```
Cleats                  24551
Men's Footwear          22246
Women's Apparel         21035
Indoor/Outdoor Games    19298
Fishing                 17325
Water Sports            15540
Camping & Hiking        13729
Cardio Equipment        12487
Shop By Sport           10984
Electronics              3156
Name: Category Name, dtype: int64
```
In [28]:
```
df1['Customer City'].value_counts().sort_values()[::-1][:10]
```
Out[28]:
```
Caguas         66770
Chicago         3885
Los Angeles     3417
Brooklyn        3412
New York        1816
Philadelphia    1577
Bronx           1500
San Diego       1437
Miami           1314
Houston         1297
Name: Customer City, dtype: int64
```
In [2]:
```
import pandas as pd
```
import numpy as np

In [3]:
```
df1=pd.read_csv('Sales_Shipment_Data.csv')
```
In [5]:
```
df1['Shipping Mode'].value_counts().idxmax()
```
Out[5]:
```
'Standard Class'
```
In [ ]:

In [4]:
```
import pandas as pd
```
In [6]:
```
import numpy as np
```
df1=pd.read_csv('Sales_Shipment_Data.csv')

In [7]:
```
df1=pd.read_csv('Sales_Shipment_Data.csv')
```
In [8]:
```
df1['Category Name'].value_counts().sort_values()[::-1][:5]
```
Out[8]:
```
Cleats                  24551
Men's Footwear          22246
Women's Apparel         21035
Indoor/Outdoor Games    19298
Fishing                 17325
Name: Category Name, dtype: int64
```
In [ ]:
```
conda install -c conda-forge nbconvert-webpdf
```
In [ ]: