```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
Data=pd.read_csv("/content/sales_data.csv")
```

```
Data
```

:e	Day	Month	Year	Customer_Age	Age_Group	Customer_Gender	Country	State
3-26	26	November	2013	19	Youth (<25)	M	Canada	British Columbia
5-26	26	November	2015	19	Youth (<25)	M	Canada	British Columbia
4-23	23	March	2014	49	Adults (35-64)	M	Australia	New South Wales
6-23	23	March	2016	49	Adults (35-64)	M	Australia	New South Wales
4-15	15	May	2014	47	Adults (35-64)	F	Australia	New South Wales
...	...	...	...	...	...	...	...	...
6-12	12	April	2016	41	Adults (35-64)	M	United Kingdom	England
4-02	2	April	2014	18	Youth (<25)	M	Australia	Queensland
6-02	2	April	2016	18	Youth (<25)	M	Australia	Queensland
4-04	4	March	2014	37	Adults (35-64)	F	France	Seine (Paris)
6-04	4	March	2016	37	Adults (35-64)	F	France	Seine (Paris)

18 columns

```
Data.head()
```

```
Data.shape
```

```
(113036, 18)
```

```
Data.info()
```

```
Data.describe()
```

	Day	Year	Customer_Age	Order_Quantity	Unit_Cost	U
count	113036.000000	113036.000000	113036.000000	113036.000000	113036.000000	1130
mean	15.665753	2014.401739	35.919212	11.901660	267.296366	4
std	8.781567	1.272510	11.021936	9.561857	549.835483	9
min	1.000000	2011.000000	17.000000	1.000000	1.000000	
25%	8.000000	2013.000000	28.000000	2.000000	2.000000	
50%	16.000000	2014.000000	35.000000	10.000000	9.000000	
75%	23.000000	2016.000000	43.000000	20.000000	42.000000	
max	31.000000	2016.000000	87.000000	32.000000	2171.000000	35

```
Data.dtypes
```

```
Data.dropna(inplace=True)
```

```
Data.isnull().sum()
```

```
Data["Customer_Age"].mean()
```

```
Data['Customer_Age'].plot(kind="kde")
```

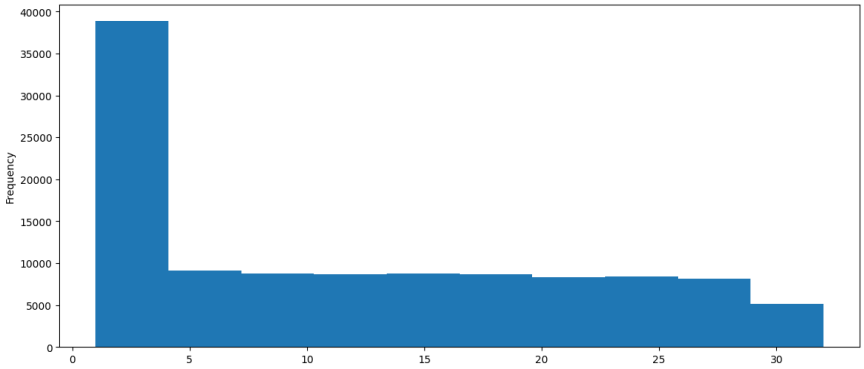
```
Data['Customer_Age'].plot(kind='box',vert=False,figsize=(14,6))
```

```
Data['Order_Quantity'].mean()
```

```
11.901659648253654
```

```
Data['Order_Quantity'].plot(kind='hist',figsize=(14,6))
```

```
<Axes: ylabel='Frequency'>
```

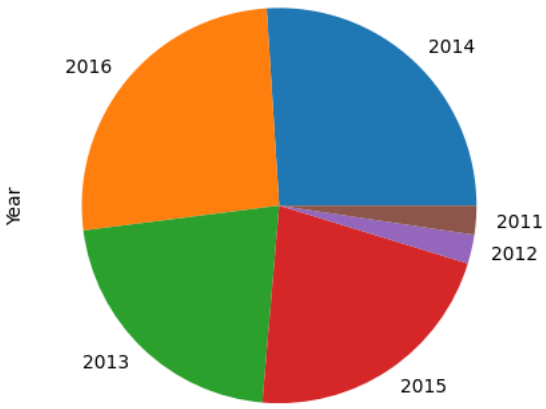



```
Data['Order_Quantity'].plot(kind='box',figsize=(14,6))
```

```
Data['Year'].value_counts()
```

```
Data['Year'].value_counts().plot(kind='pie')
```

```
<Axes: ylabel='Year'>
```




```
Data['Month'].value_counts()
```

```
Data['Month'].value_counts().plot(kind='bar',figsize=(14,6))
```

```
Data['Country'].value_counts()
```

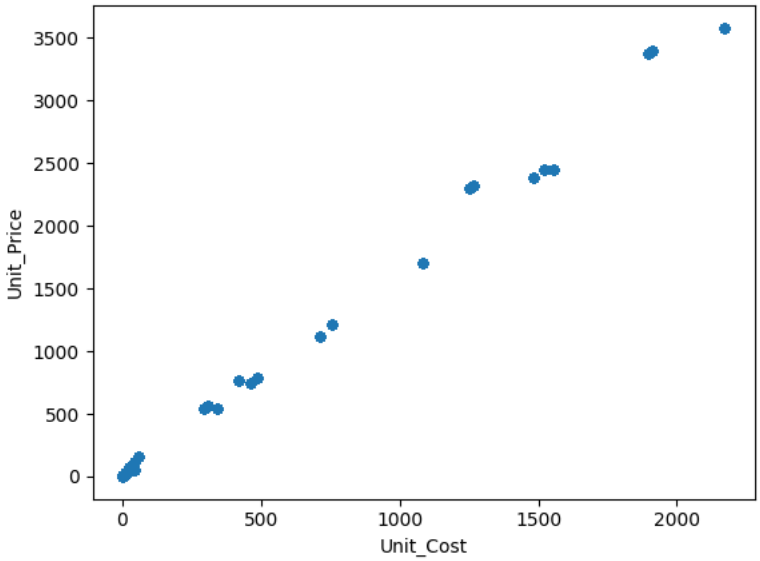
```
sales = Data.groupby('Country')['Cost'].sum().plot(kind="bar")
```

```
Data['Product'].unique()
```

```
Data["Product"].value_counts().head(10).plot(kind="bar",figsize=(10,4))
```

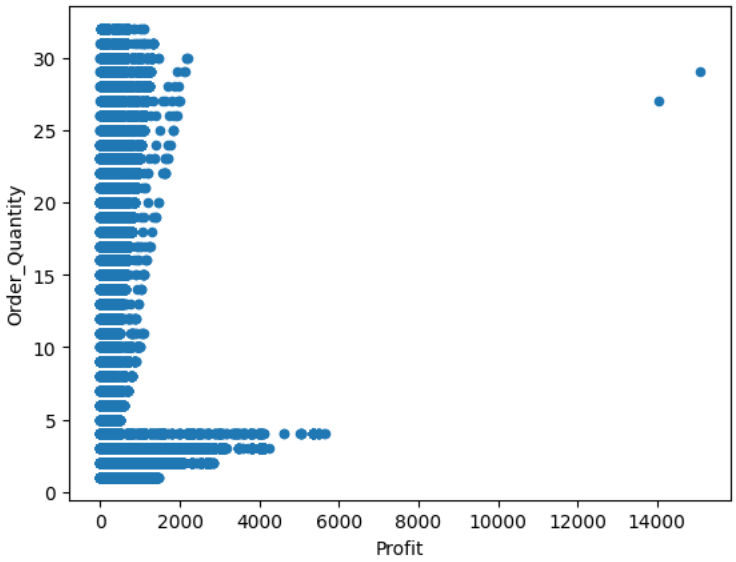
```
Data.plot(kind="scatter",x="Unit_Cost",y="Unit_Price")
```

```
<Axes: xlabel='Unit_Cost', ylabel='Unit_Price'>
```



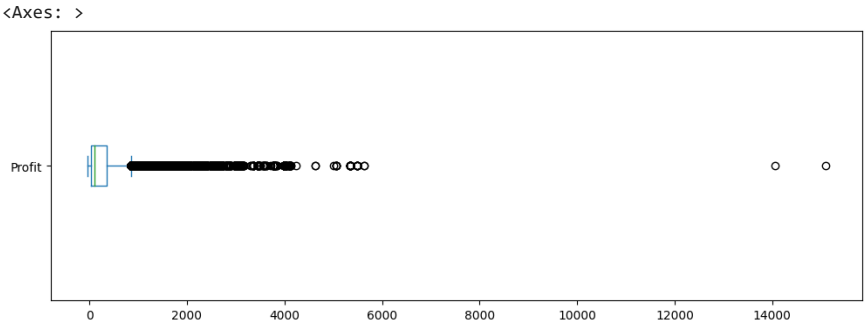

```
Data.plot(kind="scatter",x="Profit",y="Order_Quantity")
```

```
<Axes: xlabel='Profit', ylabel='Order_Quantity'>
```



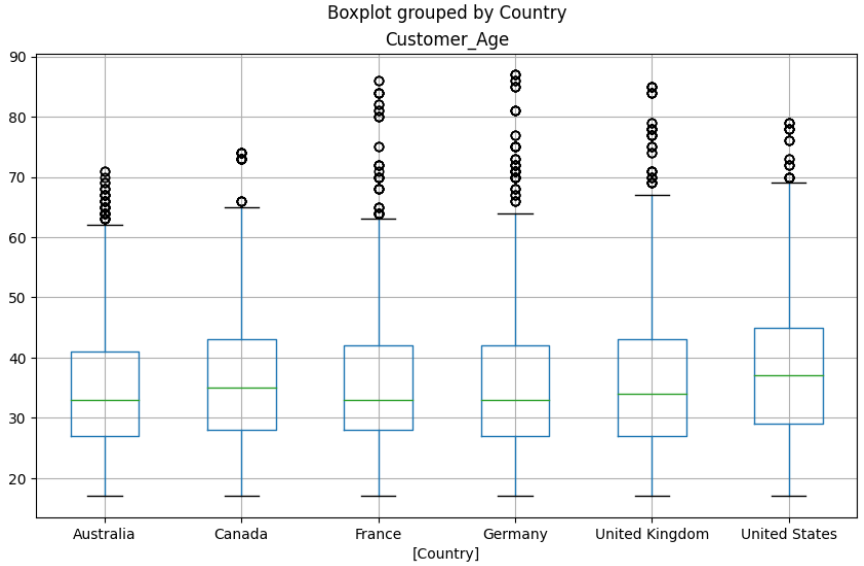

```
Data.plot(kind="box",x="Country",y="Profit",figsize=(12,4),vert=False)
```

```
<Axes: >
```




```
Data[["Customer_Age","Country"]].boxplot(by="Country",figsize=(10,6))
```

```
<Axes: title={'center': 'Customer_Age'}, xlabel='[Country]'>
```

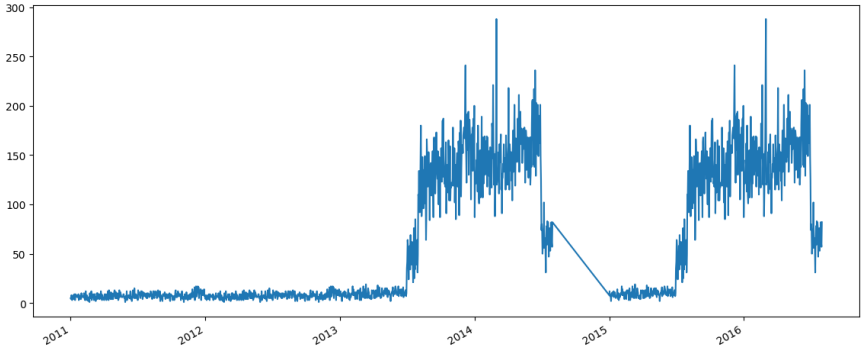



```
Data["Calculated_Date"]=Data[['Year','Month','Day']].apply(lambda x:'{}-{}-{}'.format(x[0],x[1],x[2]),axis=1)
Data["Calculated_Date"]=pd.to_datetime(Data["Calculated_Date"])
Data['Calculated_Date'].head()
```

```
0   2013-11-26
1   2015-11-26
2   2014-03-23
3   2016-03-23
4   2014-05-15
Name: Calculated_Date, dtype: datetime64[ns]
```

```
Data["Calculated_Date"].value_counts().plot(kind="line",figsize=(14,6))
```

```
<Axes: >
```




```
Data['Revenue']+50
```

```
0         1000
1         1000
2         2451
3         2138
4          468
          ...
113031     234
113032    1233
113033    1233
113034    1310
113035    1257
Name: Revenue, Length: 113036, dtype: int64
```

```
canada_orders = len(Data[Data["Country"] == "Canada"])
france_orders = len(Data[Data["Country"] == "France"])

print(f"Number of Orders in Canada: {canada_orders}")
print(f"Number of Orders in France: {france_orders}")
```

```
Number of Orders in Canada: 14178
Number of Orders in France: 10998
```

```
Data.loc[(Data['Country']=="canada") & (Data['Sub_Category']=="Bike Racks")].shape[0]
```

```
0
```

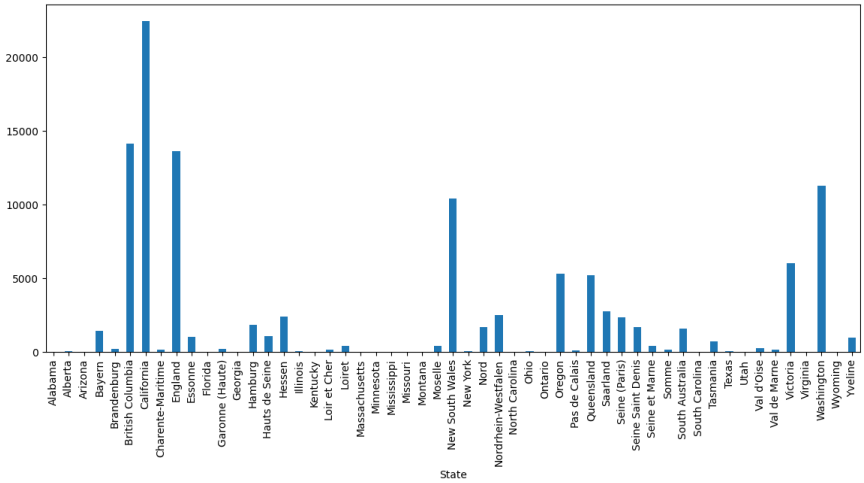
```
Data.groupby('State').size()
```

```
State
Alabama                   4
Alberta                  56
Arizona                   4
Bayern                 1426
Brandenburg             198
British Columbia      14116
California            22450
Charente-Maritime       148
England               13620
Essonne                 994
Florida                  14
Garonne (Haute)         208
Georgia                   8
Hamburg                1836
Hauts de Seine         1084
Hessen                 2384
Illinois                 28
Kentucky                 10
Loir et Cher            120
Loiret                  382
Massachusetts             2
Minnesota                 6
Mississippi               4
Missouri                  6
Montana                   6
```

```
Moselle                386
New South Wales      10412
New York                20
Nord                  1670
Nordrhein-Westfalen   2484
North Carolina           4
Ohio                    28
Ontario                  6
Oregon                5286
Pas de Calais           90
Queensland            5220
Saarland              2770
Seine (Paris)         2328
Seine Saint Denis     1684
Seine et Marne         394
Somme                  134
South Australia       1564
South Carolina          10
Tasmania               724
Texas                   30
Utah                    10
Val d'Oise             264
Val de Marne           158
Victoria              6016
Virginia                 4
Washington           11264
Wyoming                  8
Yveline                954
dtype: int64
```

```
Data.groupby('State').size().plot(kind="bar",figsize=(14,6))
```

```
<Axes: xlabel='State'>
```



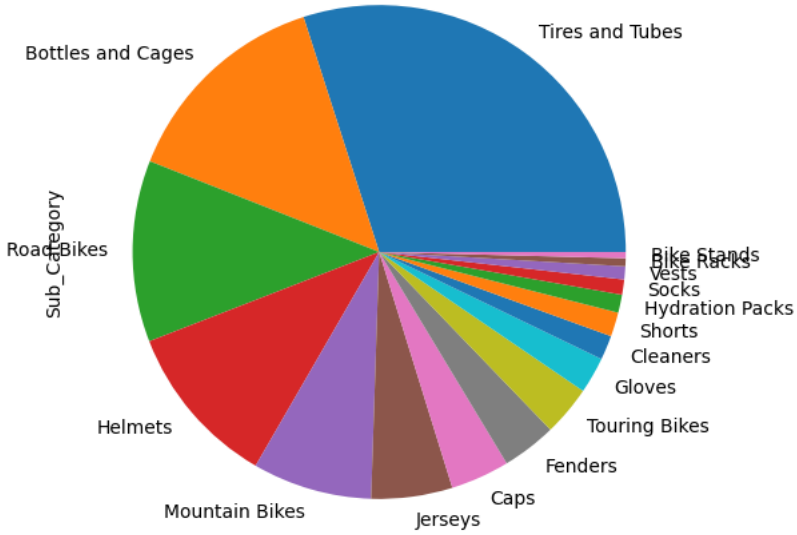

```
Data["Sub_Category"].value_counts()
```

```
Tires and Tubes      33870
Bottles and Cages    15876
Road Bikes           13430
Helmets              12158
Mountain Bikes        8854
Jerseys               6010
Caps                  4358
Fenders               4032
Touring Bikes         3698
Gloves                2686
Cleaners              1802
Shorts                1794
Hydration Packs       1334
Socks                 1122
```

```
Vests                964
Bike Racks           592
Bike Stands          456
Name: Sub_Category, dtype: int64
```

```
Data["Sub_Category"].value_counts().plot(kind="pie",figsize=(14,6))
```

```
<Axes: ylabel='Sub_Category'>
```

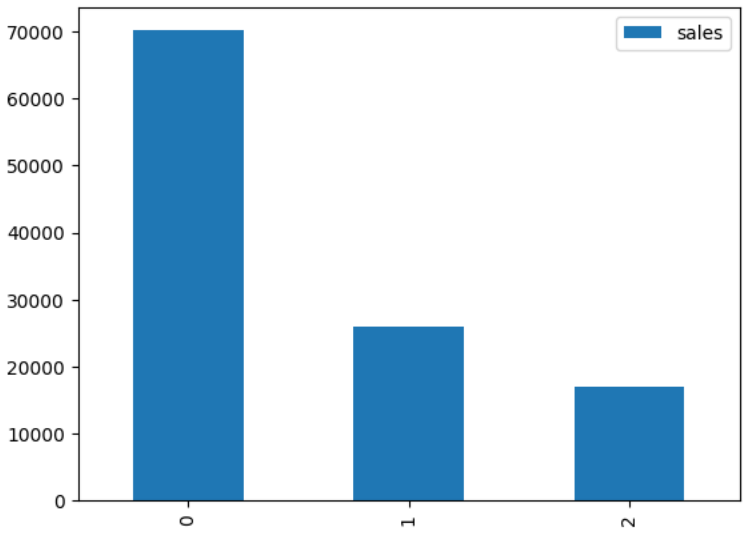



```
Data.groupby('Product_Category').size().reset_index(name='sales').loc[Data['Product_Category'].str.contains('Accessories')]
```

	Product_Category	sales
0	Accessories	70120
1	Bikes	25982
2	Clothing	16934

```
Data.groupby('Product_Category').size().reset_index(name='sales').loc[Data['Product_Category'].str.contains('Accessories')].plot(kind="l
```

```
<Axes: >
```

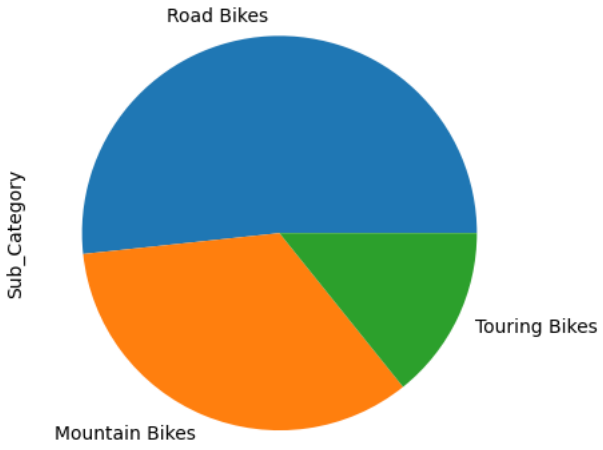



```
Data.loc[Data['Product_Category']=="Bikes","Sub_Category"].value_counts()
```

```
Road Bikes        13430
Mountain Bikes     8854
Touring Bikes      3698
Name: Sub_Category, dtype: int64
```

```
Data.loc[Data['Product_Category']=="Bikes","Sub_Category"].value_counts().plot(kind="pie")
```

```
<Axes: ylabel='Sub_Category'>
```




```
Data['Customer_Gender'].value_counts()
```

```
M    58312
F    54724
Name: Customer_Gender, dtype: int64
```

```
Data.loc[(Data['Customer_Gender']=="M") & (Data['Revenue']>=500)].shape[0]
```

```
21781
```

```
Data.sort_values(["Revenue"],ascending=False).head(5)
```

	Date	Day	Month	Year	Customer_Age	Age_Group	Customer_Gender	Country
112073	2015-07-24	24	July	2015	52	Adults (35-64)	M	Australia
112072	2013-07-24	24	July	2013	52	Adults (35-64)	M	Australia
71129	2011-07-08	8	July	2011	22	Youth (<25)	M	Canada
70307	2011-04-30	30	April	2011	44	Adults (35-64)	M	Canada
70601	2011-09-30	30	September	2011	19	Youth (<25)	F	Canada

```
Data['Revenue'].max()
```

```
58074
```

```
Data.loc[Data['Revenue']> 10_1000 ,"Order_Quantity"].mean()
```

```
nan
```

```
Data.loc[Data['Revenue']<10_1000 ,"Order_Quantity"].mean()
```

```
11.901659648253654
```

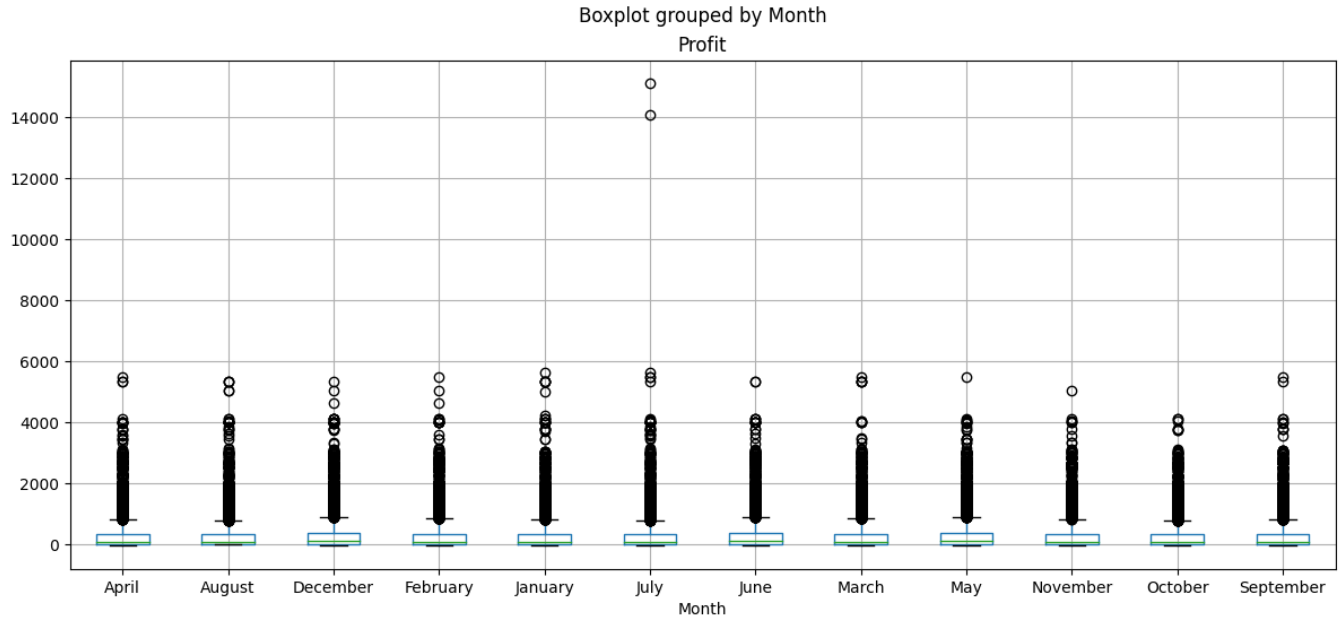
```
Data.loc[(Data['Year']==2016) & (Data['Month']=="May")].shape[0]
```

```
5015
```

```
Data.loc[(Data['Year']==2016) & (Data['Month'].isin(["May","June","July"]))].shape[0]
```

```
12164
```

```
ax = Data.boxplot(column="Profit", by="Month", figsize=(14, 6))
```




```
tax_rate = 0.072

us_sales = Data[Data["Country"] == "United States"]

us_sales["Unit_Price"] = us_sales["Unit_Price"] * (1 + tax_rate)

print(us_sales["Unit_Price"].head())
```