P VAISHNAVI

DATA SCIENCE AND BUSINESS ANALYTICS
INTERN @ THE SPARKS FOUNDATION
DATASET : SAMPLESUPERSTORE.CSV (https://bit.ly/3i4rbWl)

EXPLORATORY DATA ANALYSIS - RETAIL

Task-3:

1. Perform ‘Exploratory Data Analysis’ on dataset ‘SampleSuperstore’.
2. As a business manager, try to find out the weak areas where you can work to make more profit.

```
#importing the libraries

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

```
from google.colab import files
uploaded = files.upload()
```

Choose Files SampleSuperstore.csv

- **SampleSuperstore.csv**(application/vnd.ms-excel) - 1113007 bytes, last modified: 6/3/2021 - 100% done

```
Saving SampleSuperstore.csv to SampleSuperstore.csv
```

```
#loading the dataset

data = pd.read_csv("SampleSuperstore.csv")
data
```

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Ca
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Boo
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	S
...	...	...	...	...	...	...	...	...	
9989	Second Class	Consumer	United States	Miami	Florida	33180	South	Furniture	Furn
9990	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Furniture	Furn

```
data.head(5)
```

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Sub-Category
0	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Bookcases
1	Second Class	Consumer	United States	Henderson	Kentucky	42420	South	Furniture	Chairs
2	Second Class	Corporate	United States	Los Angeles	California	90036	West	Office Supplies	Labels
3	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Furniture	Tables
4	Standard Class	Consumer	United States	Fort Lauderdale	Florida	33311	South	Office Supplies	Storage

```
data.tail(5)
```

	Ship Mode	Segment	Country	City	State	Postal Code	Region	Category	Ca
9989	Second Class	Consumer	United States	Miami	Florida	33180	South	Furniture	Furn
9990	Standard Class	Consumer	United States	Costa Mesa	California	92627	West	Furniture	Furn
9991	Standard	Consumer	United	Costa Mesa	California	92627	West	Technology	

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 13 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   Ship Mode     9994 non-null   object 
 1   Segment       9994 non-null   object 
 2   Country       9994 non-null   object 
 3   City          9994 non-null   object 
 4   State         9994 non-null   object 
 5   Postal Code   9994 non-null   int64  
 6   Region        9994 non-null   object 
 7   Category      9994 non-null   object 
 8   Sub-Category  9994 non-null   object 
 9   Sales         9994 non-null   float64
 10  Quantity      9994 non-null   int64  
 11  Discount      9994 non-null   float64
 12  Profit        9994 non-null   float64
dtypes: float64(3), int64(2), object(8)
memory usage: 1015.1+ KB
```

```
data.describe()
```

	Postal Code	Sales	Quantity	Discount	Profit
count	9994.000000	9994.000000	9994.000000	9994.000000	9994.000000
mean	55190.379428	229.858001	3.789574	0.156203	28.656896
std	32063.693350	623.245101	2.225110	0.206452	234.260108
min	1040.000000	0.444000	1.000000	0.000000	-6599.978000
25%	23223.000000	17.280000	2.000000	0.000000	1.728750
50%	56430.500000	54.490000	3.000000	0.200000	8.666500
75%	90008.000000	209.940000	5.000000	0.200000	29.364000
max	99301.000000	22638.480000	14.000000	0.800000	8399.976000

```
#detecting missing values in the dataset
data.isnull().sum()
```

```
Ship Mode       0
Segment         0
Country         0
City            0
State           0
Postal Code     0
Region          0
Category        0
Sub-Category    0
Sales           0
Quantity        0
Discount        0
Profit          0
dtype: int64
```

```
data.isna().sum()
```

```
Ship Mode       0
Segment         0
Country         0
City            0
State           0
Postal Code     0
Region          0
Category        0
Sub-Category    0
Sales           0
Quantity        0
Discount        0
Profit          0
dtype: int64
```

```
sales_data = data.groupby('Category', as_index=False)['Sales'].sum()
subcat_data = data.groupby(['Category','Sub-Category'])['Sales'].sum()
subcat_data['Sales']=map(int,subcat_data)
sales_data
```

	Category	Sales
0	Furniture	741999.7953
1	Office Supplies	719047.0320
2	Technology	836154.0330

```
data.columns
```

```
Index(['Ship Mode', 'Segment', 'Country', 'City', 'State', 'Region',
       'Category', 'Sub-Category', 'Sales', 'Quantity', 'Discount', 'Profit'],
      dtype='object')
```

```
data.index
data.index
```

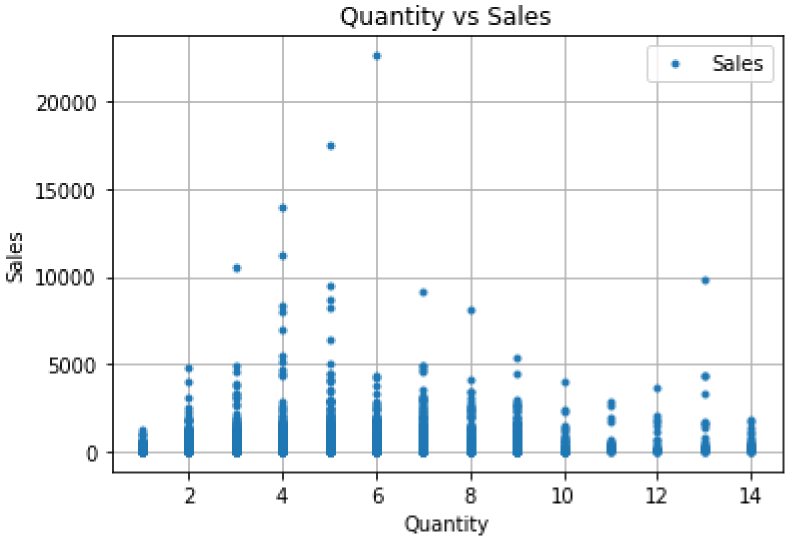
```
    RangeIndex(start=0, stop=9994, step=1)
```

```
data.nunique()
```

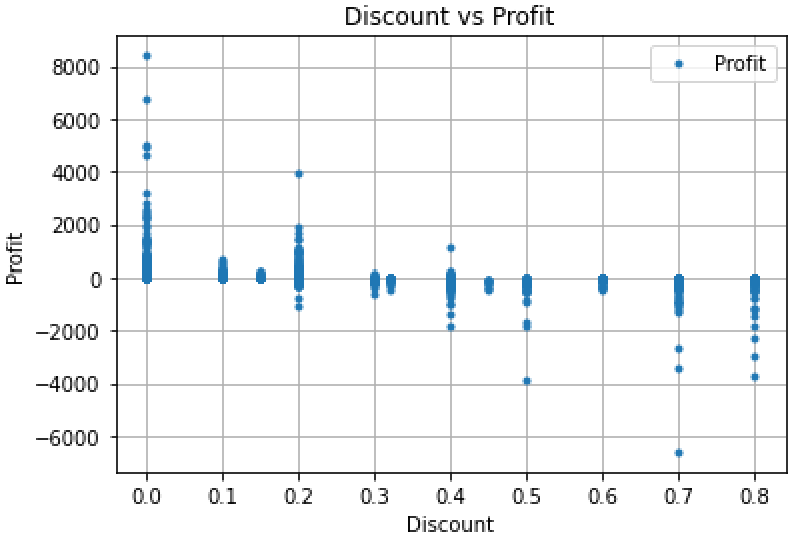
```
    Ship Mode          4
    Segment            3
    Country            1
    City             531
    State             49
    Region             4
    Category           3
    Sub-Category      17
    Sales           5825
    Quantity          14
    Discount          12
    Profit          7287
    dtype: int64
```

Exploratory Data Analysis

```
data.plot(x='Quantity',y='Sales',style='.')
plt.title('Quantity vs Sales')
plt.xlabel('Quantity')
plt.ylabel('Sales')
plt.grid()
plt.show()
```

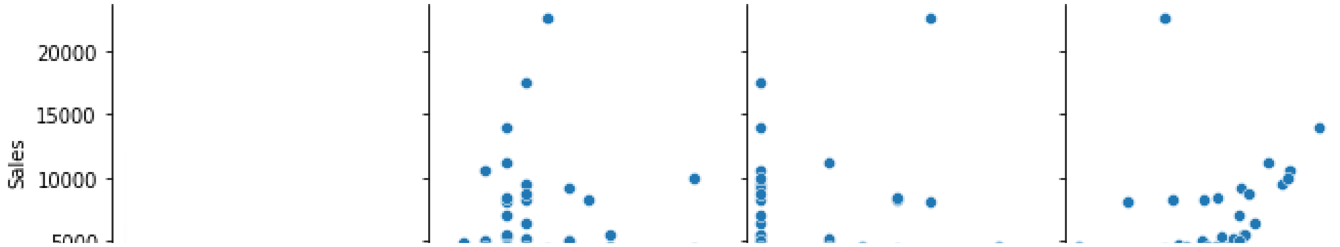


```
data.plot(x='Discount',y='Profit',style='.')
plt.title('Discount vs Profit')
plt.xlabel('Discount')
plt.ylabel('Profit')
plt.grid()
plt.show()
plt.show()
```



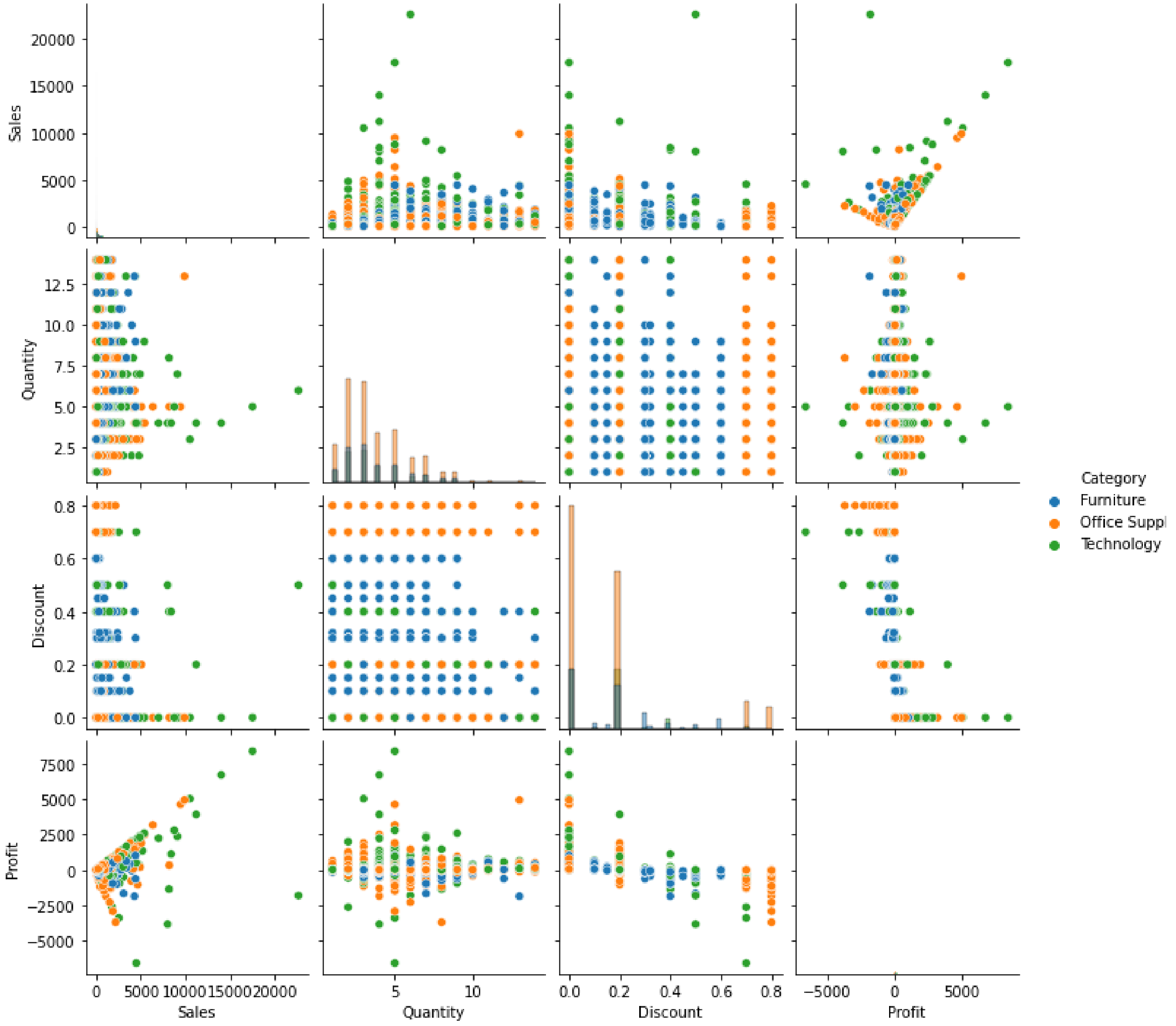
```
sb.pairplot(data)
```

```
<seaborn.axisgrid.PairGrid at 0x7fde9e233050>
```



```
sb.pairplot(data,hue='Category',diag_kind='hist')
```

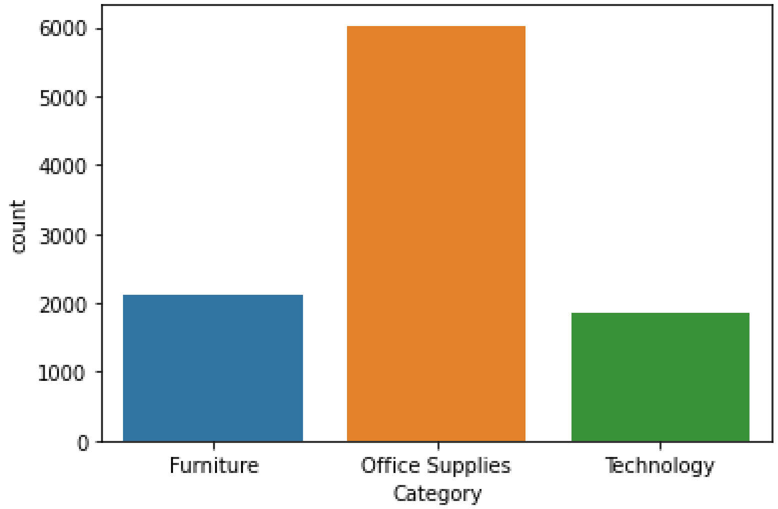
```
<seaborn.axisgrid.PairGrid at 0x7fde90a797d0>
```



```
data['Category'].value_counts()
```

```
sb.countplot(x=data['Category'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fde90382450>
```

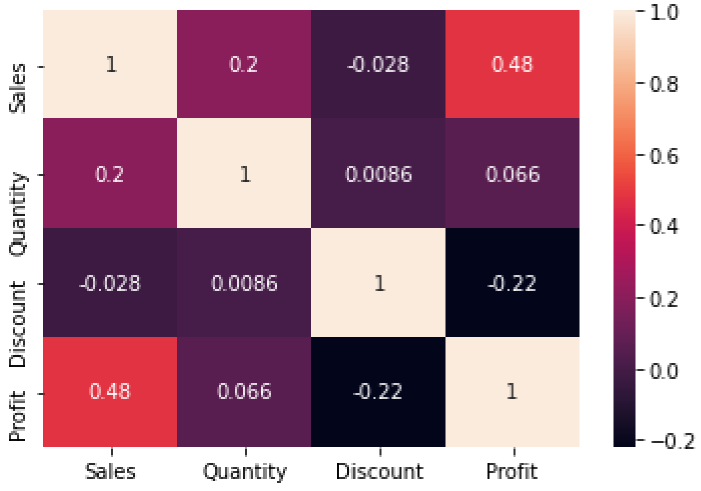


```
data.corr()
```

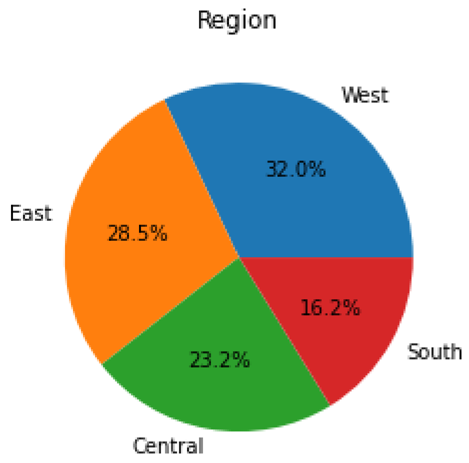
	Sales	Quantity	Discount	Profit
Sales	1.000000	0.200795	-0.028190	0.479064
Quantity	0.200795	1.000000	0.008623	0.066253
Discount	-0.028190	0.008623	1.000000	-0.219487
Profit	0.479064	0.066253	-0.219487	1.000000

```
sb.heatmap(data.corr(), annot=True)
```

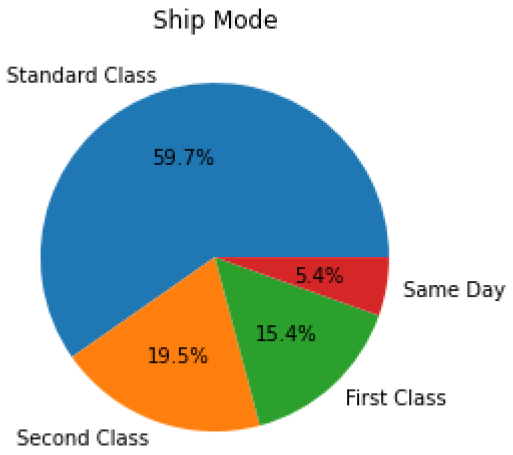
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fde8d752810>
```



```
plt.title('Region')
plt.pie(data['Region'].value_counts(),labels=data['Region'].value_counts().index,autopct='%1.
plt.show()
```

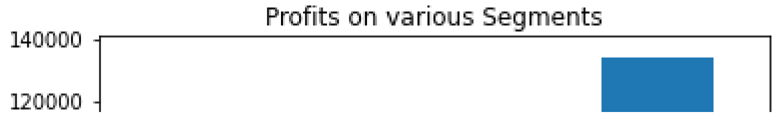


```
plt.title('Ship Mode')
plt.pie(data['Ship Mode'].value_counts(),labels=data['Ship Mode'].value_counts().index,autopc
plt.show()
```



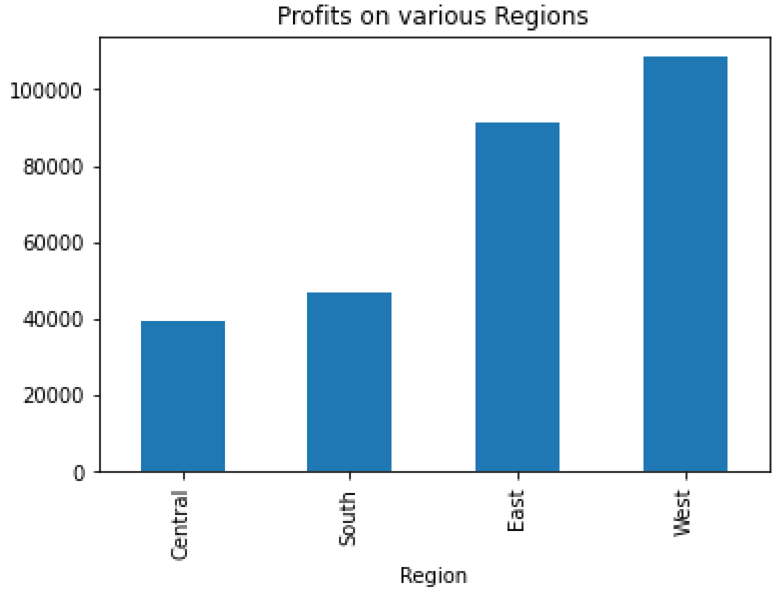
```
data.groupby('Segment')['Profit'].sum().sort_values().plot.bar()
plt.title("Profits on various Segments")
```

```
Text(0.5, 1.0, 'Profits on various Segments')
```



```
data.groupby('Region')['Profit'].sum().sort_values().plot.bar()
plt.title("Profits on various Regions")
```

```
Text(0.5, 1.0, 'Profits on various Regions')
```



```
plt.figure(figsize=(14,6))
data.groupby('State')['Profit'].sum().sort_values().plot.bar()
plt.title("Profits on various States")
```

```
Text(0.5, 1.0, 'Profits on various States')
```

Statewise Deal Analysis

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

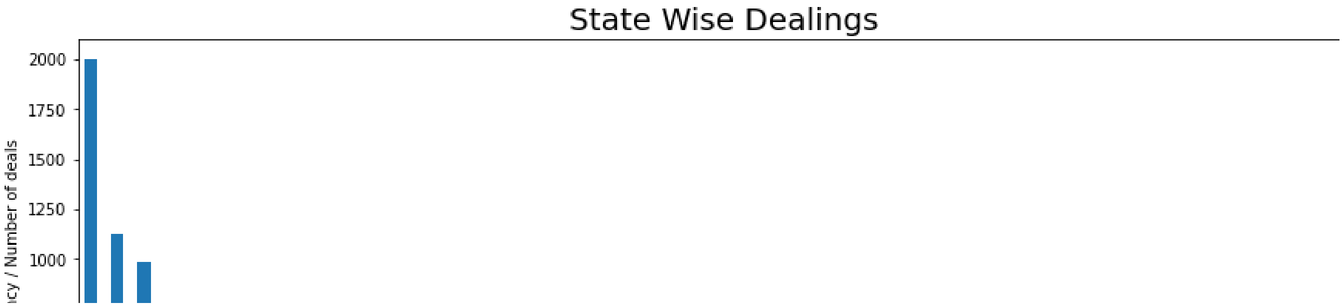
```
United States    9994
Name: Country, dtype: int64
```

```
data1 = data['State'].value_counts()
data1.head(10)
```

```
California        2001
New York          1128
Texas              985
Pennsylvania       587
Washington         506
Illinois           492
Ohio               469
Florida            383
Michigan           255
North Carolina     249
Name: State, dtype: int64
```

```
data1.plot(kind='bar',figsize=(15,5))
plt.ylabel('Frequency / Number of deals')
plt.xlabel('States')

plt.title('State Wise Dealings', fontsize = 20)
plt.show()
```



Here is top 3 state where deals are Highest.

Califonia

New York

Texas

Wyoming: Lowest Number of deal

```
data['State'].value_counts().mean()
```

```
203.9591836734694
```

✓ 0s completed at 1:44 PM