Final Project - Analyzing Sales Data

Date: 9 January 2023

Author: Kamolchanok Thippiensak

Course: `Pandas Foundation`

```
# TODO 01 - how many columns, rows in this dataset

df.shape
```

```
(9994, 23)
```

```
# TODO 02 - is there any missing values?,
#           if there is, which colunm? how many nan values?

nan = pd.Series(df.isna().sum().sort_values(ascending=False))
nan[nan>0]
```

```
Postal Code    11
dtype: int64
```

```
# TODO 03 - your friend ask for `California` data, filter it and export csv for him

csv = df[df.State == 'California']
csv.to_csv("California_data.csv")
```

```
# TODO 04 - your friend ask for all order data in `California` and `Texas` in 2017
#           (look at Order Date), send him csv file

df[df['Order Date'].dt.year == 2017].query('State == ("California","Texas")')\
    .to_csv("California_Texas_2017.csv")
```

```python
# TODO 05 - how much total sales, average sales, and standard deviation of sales
#           your company make in 2017

y_2017 = df[df['Order Date'].dt.year == 2017]
y_2017['Sales'].agg(['sum','mean','std']).round(decimals=2)
```

```
sum     484247.50
mean       242.97
std        754.05
Name: Sales, dtype: float64
```

```python
# TODO 06 - which Segment has the highest profit in 2018

df[df['Order Date'].dt.strftime('%Y') == '2018'][['Segment','Profit']]\
    .groupby('Segment').sum().round(decimals=2).reset_index().head(1)
```

	Segment	Profit
0	Consumer	28460.17

```python
# TODO 07 - which top 5 States have the least total sales between
#           15 April 2019 - 31 December 2019

state = df[(df['Order Date'] >= '2019-04-15') & (df['Order Date'] <= '2019-12-31')]
state[['State','Sales']].groupby('State').sum().round(decimals=2)\
    .sort_values(by='Sales',ascending=True).head(5)
```

	Sales
State	
New Hampshire	49.05
New Mexico	64.08
District of Columbia	117.07
Louisiana	249.80
South Carolina	502.48

```
# TODO 08 - what is the proportion of total sales (%) in West + Central
#           in 2019 e.g. 25%

west_central_2019 = df[df['Order Date'].dt.strftime('%Y') == '2019']\
                    .query("Region == ('West','Central')")['Sales'].sum()

total = df[df['Order Date'].dt.strftime('%Y') == '2019']['Sales'].sum()

print(f"{((west_central_2019/total)*100).round(1)}%")
```

```
55.0%
```

```
# TODO 09 - find top 10 popular products(sub-category) in terms of
#           number of orders vs. total sales during 2019-2020
year = df[(df['Order Date'] >= '2019') & (df['Order Date'] <= '2020')]

order = year[['Sub-Category','Order ID']].groupby('Sub-Category')\
        .count().reset_index().head(10)
sale = year[['Sub-Category','Sales']].groupby('Sub-Category').sum()\
        .reset_index().head(10)
```

	Top 10 Product	No.of order
0	Accessories	186
1	Appliances	115
2	Art	184
3	Binders	418
4	Bookcases	54
5	Chairs	166
6	Copiers	16
7	Envelopes	62
8	Fasteners	59
9	Furnishings	260

	Top 10 Product	Total Sales
0	Accessories	41895.8540
1	Appliances	26065.5390
2	Art	5973.6440
3	Binders	49707.1430
4	Bookcases	26275.4665
5	Chairs	84229.3890
6	Copiers	49599.4100
7	Envelopes	4729.8900
8	Fasteners	960.1340
9	Furnishings	28538.8700

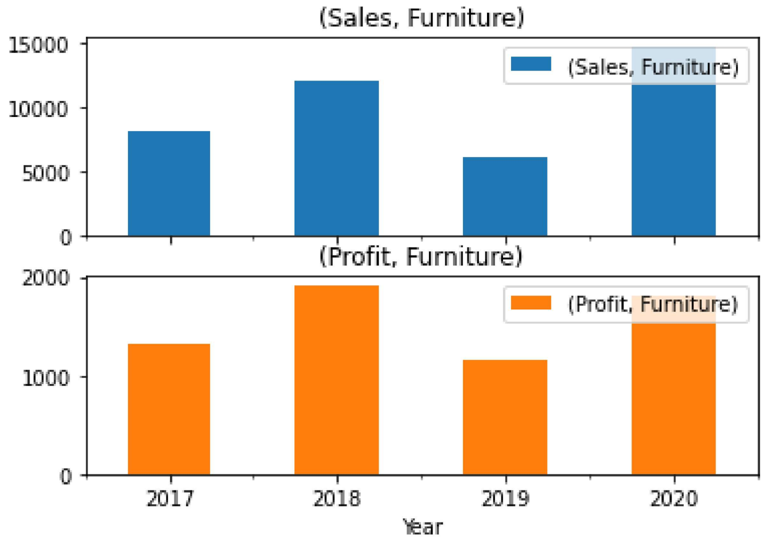
```
# TODO 10 - plot sale ans profit of Furniture sales in Seattle each year
```

```
df['Year'] = df['Order Date'].dt.strftime('%Y')
bar = df.query("City == 'Seattle'& Category == 'Furniture'")\
        .groupby(['Category','Year'])[['Sales','Profit']].sum()\
        .round(2).reset_index()

bar.pivot(columns='Category', index='Year', values=['Sales','Profit'])\
    .plot.bar(rot=0, subplots=True)
```

```
array([%lt;AxesSubplot:title={'center':'(Sales, Furniture)'}, xlabel='Year'%gt;,
       %lt;AxesSubplot:title={'center':'(Profit, Furniture)'}, xlabel='Year'%gt;],
      dtype=object)
```

Download


(Sales, Furniture)
(Sales, Furniture)
(Profit, Furniture)
(Profit, Furniture)
Year