

# salome\_HW4\_pymoli

December 9, 2017

```
In [292]: import pandas as pd
import json as js
import os
import numpy as np
import glob
```

```
In [293]: json1_path = "../purchase_data.json"
json2_path = "../purchase_data2.json"
```

```
for json_file in (json1_path, json2_path):
    df_pymoli = pd.read_json(json_file)
```

```
df_pymoli.head()
```

```
# Print the first five rows of data to the screen
df.head()
json_pattern = os.path.join(json_path, '*.json')
file_list = glob.glob(json_pattern)
for file in file_list:
    contents.append(read(file))
```

```
Out [293]:
```

	Age	Gender	Item ID	Item Name	Price	SN
0	20	Male	93	Apocalyptic Battlescythe	4.49	Iloni35
1	21	Male	12	Dawne	3.36	Aidaira26
2	17	Male	5	Putrid Fan	2.63	Irim47
3	17	Male	123	Twilight's Carver	2.55	Irith83
4	22	Male	154	Feral Katana	4.11	Philodil43

```
In [294]: df_pymoli.columns
```

```
Out [294]: Index(['Age', 'Gender', 'Item ID', 'Item Name', 'Price', 'SN'], dtype='object')
```

```
In [295]: #Count the unique name of players by counting the number of elements in row SN
total_players = df_pymoli['SN'].unique()
print (len(total_players))
df_pymoli1 = pd.DataFrame({'Total players': [len(total_players)]})
df_pymoli1
```

74

```
Out[295]:    Total players
          0             74
```

```
In [296]: #Purchasing Analysis (Total)
          #Number of Unique Items
```

```
In [297]: df2= df_pymoli['Item ID'].unique()
          len(df2)
```

```
Out[297]: 64
```

```
In [298]: #Total Revenue, should be the sum of all price values
          Total_rev =round(df_pymoli['Price'].sum(),2)
          Total_rev
```

```
Out[298]: 228.1
```

```
In [299]: #Purchasing Analysis (Total): Average Price, Total Number of Purchases, Number of Un
```

```
In [300]: df_pymoli2 = pd.DataFrame({'Number of Unique Items': len(df2), 'Average Price':[df_pymoli['Price'].sum()/len(df2)]}
          df_pymoli2["Average Price"] = df_pymoli2["Average Price"].map("$ {:.2f}".format)
          df_pymoli2["Total Revenue"] = df_pymoli2["Total Revenue"].map("$ {:.2f}".format)
          df_pymoli2
```

```
Out[300]:    Average Price  Number of Purchases  Number of Unique Items  Total Revenue
          0          $ 2.92                78                64          $ 228.10
```

```
In [301]: #Purchasing Analysis (Gender)
          #The below each broken by gender
          #Purchase Count
```

```
In [302]: #Using groupby Gender, find the number of M, F, ND
          grouped_gender = df_pymoli.groupby('Gender')
          grouped_gender
          # The object returned is a "GroupBy" object and cannot be viewed normally...
          print(type(grouped_gender))
          # In order to be visualized, a data function must be used...
          grouped_gender.count().head()
```

```
<class 'pandas.core.groupby.DataFrameGroupBy'>
```

```
Out[302]:
```

	Age	Item ID	Item Name	Price	SN
Gender					
Female	13	13	13	13	13
Male	64	64	64	64	64
Other / Non-Disclosed	1	1	1	1	1

```
In [303]: #Number of M, F and ND that made at least one purchase
count_gender = grouped_gender["SN"].nunique()
count_gender.head()
total_players = count_gender.sum()# or see above
total_players
```

```
Out[303]: 74
```

```
In [304]: #Number of M, F and ND that made at least one purchase
count_gender = grouped_gender["SN"].nunique()
count_gender.head()
```

```
Out[304]: Gender
Female                13
Male                 60
Other / Non-Disclosed    1
Name: SN, dtype: int64
```

```
In [305]: ##Percentage of players count by Gender(absolute values)
Perc_count1 = round(count_gender / total_players, 4)*100
Perc_count1
```

```
Out[305]: Gender
Female                17.57
Male                 81.08
Other / Non-Disclosed    1.35
Name: SN, dtype: float64
```

```
In [306]: df_pymoli3= pd.DataFrame({'Total count': count_gender, "Percentage of Players": Perc_count1})
df_pymoli3["Percentage of Players"] =df_pymoli3["Percentage of Players"].map("{:.2f}")
df_pymoli3
```

```
Out[306]:
```

	Percentage of Players	Total count
Gender		
Female	17.57	13
Male	81.08	60
Other / Non-Disclosed	1.35	1

```
In [307]: #Purchasing Analysis (Gender)
#The below each broken by gender
#Average Purchase Price
#Total Purchase Value
#Normalized Totals (normalizing for the # of people in each gender group)
```

```
In [308]: #Purchase count
purchase_count= grouped_gender["SN"].count()
purchase_count
```

```
Out [308]: Gender
          Female          13
          Male           64
          Other / Non-Disclosed    1
          Name: SN, dtype: int64
```

```
In [309]: #Average Purchase Price grouped by M, F and ND
          Average_Price = round(grouped_gender["Price"].mean(), 2)
          Average_Price.head()
```

```
Out [309]: Gender
          Female          3.18
          Male           2.88
          Other / Non-Disclosed    2.12
          Name: Price, dtype: float64
```

```
In [310]: #Total Purchase value grouped by M, F and ND
          Total_Price = grouped_gender["Price"].sum()
          Total_Price.head()
```

```
Out [310]: Gender
          Female          41.38
          Male          184.60
          Other / Non-Disclosed    2.12
          Name: Price, dtype: float64
```

```
In [311]: #Normalized Purchase value grouped by M, F and ND
          Normalized_Price = round(grouped_gender["Price"].sum()/count_gender, 2)
          Normalized_Price.head()
```

```
Out [311]: Gender
          Female          3.18
          Male           3.08
          Other / Non-Disclosed    2.12
          dtype: float64
```

```
In [312]: df_pymoli4= pd.DataFrame({'Purchase count': purchase_count, "Average Purchase price":
df_pymoli4["Average Purchase price"]=df_pymoli4["Average Purchase price"].map("$ {:.2f}",
df_pymoli4["Total Purchase value"] = df_pymoli4["Total Purchase value"].map("$ {:.2f}",
df_pymoli4["Normalized Totals"] = df_pymoli4["Normalized Totals"].map("$ {:.2f}",
df_pymoli4
```

```
Out [312]:
```

	Average Purchase price	Normalized Totals
Gender		
Female	\$ 3.18	\$ 3.18
Male	\$ 2.88	\$ 3.08
Other / Non-Disclosed	\$ 2.12	\$ 2.12

Purchase count Total Purchase value

Gender		
Female	13	\$ 41.38
Male	64	\$ 184.60
Other / Non-Disclosed	1	\$ 2.12

```
In [313]: #Age Demographics
#The below each broken into bins of 4 years (i.e. <10, 10-14, 15-19, etc.)
#Purchase Count
#Average Purchase Price
#Total Purchase Value
#Normalized Totals (normalizing for the # of people in each age group)

In [314]: # Create the bins in which Data will be held
# Bins are 0 to 25, 25 to 50, 50 to 75, 75 to 100
bins = [0, 10, 14, 19, 24, 29, 34, 39, 100]

# Create the names for the four bins
group_names = ['<10', '10-14', '15-19', '20-24', '25-29', '30-34', '35-39', '40+']

In [315]: # Cut postTestScore and place the scores into bins
#pd.cut(df_pymoli["Age"], bins, labels=group_names)

In [316]: df_pymoli["Age group"] = pd.cut(df_pymoli["Age"], bins, labels=group_names)
df_pymoli.head()
```

```
Out[316]:
```

	Age	Gender	Item ID	Item Name	Price	SN	Age group
0	20	Male	93	Apocalyptic Battlescythe	4.49	Iloni35	20-24
1	21	Male	12	Dawne	3.36	Aidaira26	20-24
2	17	Male	5	Putrid Fan	2.63	Irim47	15-19
3	17	Male	123	Twilight's Carver	2.55	Irith83	15-19
4	22	Male	154	Feral Katana	4.11	Philodil43	20-24

```
In [317]: #Using groupby age group
grouped_age = df_pymoli.groupby('Age group')
grouped_age

Out[317]: <pandas.core.groupby.DataFrameGroupBy object at 0x114b86dd8>

In [318]: count_by_age = grouped_age["SN"].nunique()
count_by_age

Out[318]: Age group
<10      5
10-14    3
15-19   11
20-24   34
25-29    8
30-34    6
35-39    6
40+      1
Name: SN, dtype: int64
```

```
In [319]: ##Percentage of players count by Gender(absolute values)
Perc_count2 = round(count_by_age / total_players, 4)*100
Perc_count2
```

```
Out[319]: Age group
<10      6.76
10-14     4.05
15-19    14.86
20-24    45.95
25-29    10.81
30-34     8.11
35-39     8.11
40+       1.35
Name: SN, dtype: float64
```

```
In [320]: df_pymoli5= pd.DataFrame({'Total count': count_by_age, "Percentage of Players": Perc_
df_pymoli5["Percentage of Players"]=df_pymoli5["Percentage of Players"].map("{:,.2f}")
df_pymoli5
```

```
Out[320]:
```

Age group	Percentage of Players	Total count
<10	6.76	5
10-14	4.05	3
15-19	14.86	11
20-24	45.95	34
25-29	10.81	8
30-34	8.11	6
35-39	8.11	6
40+	1.35	1

```
In [321]: #Age Demographics
#The below each broken into bins of 4 years (i.e. <10, 10-14, 15-19, etc.)
#Average Purchase Price
#Total Purchase Value
#Normalized Totals (normalizing for the # of people in each age group)
```

```
In [322]: #Purchase count
purchase_count2= grouped_age["SN"].count()
purchase_count2
```

```
Out[322]: Age group
<10      5
10-14     3
15-19    11
20-24    36
25-29     9
30-34     7
35-39     6
40+       1
Name: SN, dtype: int64
```

```
In [323]: #Average Purchase Price grouped by M, F and ND
Average_Price2 = round(grouped_age["Price"].mean(),2)
Average_Price2
```

```
Out[323]: Age group
<10      2.76
10-14    2.99
15-19    2.76
20-24    3.02
25-29    2.90
30-34    1.98
35-39    3.56
40+      4.65
Name: Price, dtype: float64
```

```
In [324]: #Total Purchase value grouped by M, F and ND
Total_Price2 = grouped_age["Price"].sum()
Total_Price2
```

```
Out[324]: Age group
<10      13.82
10-14     8.96
15-19    30.41
20-24   108.89
25-29    26.11
30-34    13.89
35-39    21.37
40+       4.65
Name: Price, dtype: float64
```

```
In [325]: #Normalized Purchase value grouped by M, F and ND
Normalized_Price2 = round(grouped_age["Price"].sum()/count_by_age,2)
Normalized_Price2
```

```
Out[325]: Age group
<10      2.76
10-14    2.99
15-19    2.76
20-24    3.20
25-29    3.26
30-34    2.32
35-39    3.56
40+      4.65
dtype: float64
```

```
In [326]: df_pymoli6= pd.DataFrame({'Purchase count': purchase_count2, "Average Purchase price":
df_pymoli6["Average Purchase price"]=df_pymoli6["Average Purchase price"].map("$ {:.2f}")
df_pymoli6["Total Purchase value"]=df_pymoli6["Total Purchase value"].map("$ {:.2f}")
df_pymoli6["Normalized Totals"]=df_pymoli6["Normalized Totals"].map("$ {:.2f}")
df_pymoli6
```

```
Out [326]:
```

Age group	Average Purchase price	Normalized Totals	Purchase count \
<10	\$ 2.76	\$ 2.76	5
10-14	\$ 2.99	\$ 2.99	3
15-19	\$ 2.76	\$ 2.76	11
20-24	\$ 3.02	\$ 3.20	36
25-29	\$ 2.90	\$ 3.26	9
30-34	\$ 1.98	\$ 2.32	7
35-39	\$ 3.56	\$ 3.56	6
40+	\$ 4.65	\$ 4.65	1

Age group	Total Purchase value
<10	\$ 13.82
10-14	\$ 8.96
15-19	\$ 30.41
20-24	\$ 108.89
25-29	\$ 26.11
30-34	\$ 13.89
35-39	\$ 21.37
40+	\$ 4.65

```
In [327]: #Top Spenders
#Identify the the top 5 spenders in the game by total purchase value, then list (in
#Purchase Count
#Average Purchase Price
#Total Purchase Value
```

```
In [328]: grouped_SN = df_pymoli.groupby('SN')
grouped_SN
```

```
Out [328]: <pandas.core.groupby.DataFrameGroupBy object at 0x114d02e10>
```

```
In [329]: #Purchase count
purchase_count3= grouped_SN["SN"].count()
purchase_count3.head()
```

```
Out [329]: SN
Aeri79      1
Aerithllora36  1
Aesririam61  1
Aesurstilis64  1
Aidaira26    2
Name: SN, dtype: int64
```

```
In [330]: #Average Purchase Price grouped by M, F and ND
Average_Price3 = round(grouped_SN["Price"].mean(),2)
Average_Price3.head()
```



```
Out [330]: SN
Aeri79      4.15
Aerithllora36 4.65
Aesririam61  2.65
Aesurstilis64 4.25
Aidaira26    2.56
Name: Price, dtype: float64
```

```
In [331]: Total_Price3 = grouped_SN["Price"].sum()
Total_Price3.head()
```

```
Out [331]: SN
Aeri79      4.15
Aerithllora36 4.65
Aesririam61  2.65
Aesurstilis64 4.25
Aidaira26    5.13
Name: Price, dtype: float64
```

```
In [332]: #Total_Price3 = grouped_SN["Price"].sum()
#Total_Price3.sort_values(ascending=False).head(5)
```

```
In [333]: df_pymoli7= pd.DataFrame({'Purchase count': purchase_count3, "Average Purchase price": df_pymoli7["Average Purchase price"].map("$ {:.2f}"), "Total Purchase value": df_pymoli7["Total Purchase value"].map("$ {:.2f}")})
df_pymoli7.sort_values('Total Purchase value', ascending=False).head(5)
```

```
Out [333]:
```

	Average Purchase price	Purchase count	Total Purchase value
SN			
Sundaky74	\$ 3.70	2	\$ 7.41
Aidaira26	\$ 2.56	2	\$ 5.13
Eusty71	\$ 4.81	1	\$ 4.81
Chanirra64	\$ 4.78	1	\$ 4.78
Alarap40	\$ 4.71	1	\$ 4.71

```
In [334]: #Most Popular Items
#Identify the 5 most popular items by purchase count, then list (in a table):
#Item ID
#Item Name
#Purchase Count
#Item Price
#Total Purchase Value
```

```
In [335]: grouped_Item_ID = df_pymoli.groupby(['Item ID', 'Item Name'])
grouped_Item_ID.count().head()
```

```
Out [335]:
```

	Item ID	Item Name	Age	Gender	Price	SN	Age group
0		Splinter	1	1	1	1	1

1	Crucifer	1	1	1	1	1
2	Verdict	1	1	1	1	1
4	Bloodlord's Fetish	1	1	1	1	1
5	Putrid Fan	1	1	1	1	1

```
In [336]: #Average Purchase Price grouped by M, F and ND
Item_price = grouped_Item_ID["Price"].mean()
```

```
In [337]: #Item name
item_name= grouped_Item_ID["Item Name"]
item_name.count().head()
```

```
Out[337]: Item ID  Item Name
0      Splinter      1
1      Crucifer      1
2      Verdict       1
4      Bloodlord's Fetish  1
5      Putrid Fan     1
Name: Item Name, dtype: int64
```

```
In [338]: #Purchase count
purchase_count4= grouped_Item_ID["SN"].count()
purchase_count4.head()
```

```
Out[338]: Item ID  Item Name
0      Splinter      1
1      Crucifer      1
2      Verdict       1
4      Bloodlord's Fetish  1
5      Putrid Fan     1
Name: SN, dtype: int64
```

```
In [339]: Total_Price4 = grouped_Item_ID["Price"].sum()
Total_Price4.head()
```

```
Out[339]: Item ID  Item Name
0      Splinter      1.89
1      Crucifer      3.67
2      Verdict       2.65
4      Bloodlord's Fetish  1.91
5      Putrid Fan     2.63
Name: Price, dtype: float64
```

```
In [340]: df_pymoli8= pd.DataFrame({"Item Price":Item_price,'Purchase count': purchase_count4,
df_pymoli8["Item Price"]=df_pymoli8["Item Price"].map("$ {:.2f}".format)
df_pymoli8['Total purchase value']= df_pymoli8['Total purchase value'].map("{:.2f}"
df_pymoli8.head()
```

```
Out[340]: Item Price  Purchase count  Total purchase value
Item ID  Item Name
```

0	Splinter	\$ 1.89	1	\$1.89
1	Crucifer	\$ 3.67	1	\$3.67
2	Verdict	\$ 2.65	1	\$2.65
4	Bloodlord's Fetish	\$ 1.91	1	\$1.91
5	Putrid Fan	\$ 2.63	1	\$2.63

```
In [341]: df_pymoli8.sort_values("Purchase count",ascending=False).head(5)
```

```
Out[341]:
```

	Item ID	Item Name	Item Price	Purchase count	Total purchase value
	94	Mourning Blade	\$ 3.64	3	\$10.92
	90	Betrayer	\$ 4.12	2	\$8.24
	111	Misery's End	\$ 1.79	2	\$3.58
	64	Fusion Pummel	\$ 2.42	2	\$4.84
	154	Feral Katana	\$ 4.11	2	\$8.22

```
In [213]: #Most Profitable Items
#Identify the 5 most profitable items by total purchase value, then list (in a table)
#Item ID
#Item Name
#Purchase Count
#Item Price
#Total Purchase Value
```

```
In [346]: df_pymoli8= pd.DataFrame({"Item Price":Item_price,'Purchase count': purchase_count4,
df_pymoli8["Item Price"]=df_pymoli8["Item Price"].map("$ {:.2f}".format)
#df_pymoli8['Total purchase value']= df_pymoli8['Total purchase value'].map("{:,.2f}")
df_pymoli8.sort_values("Total purchase value", ascending= False).head(5)
```

```
Out[346]:
```

	Item ID	Item Name	Item Price	Purchase count	\
	94	Mourning Blade	\$ 3.64	3	
	117	Heartstriker, Legacy of the Light	\$ 4.71	2	
	93	Apocalyptic Battlescythe	\$ 4.49	2	
	90	Betrayer	\$ 4.12	2	
	154	Feral Katana	\$ 4.11	2	

  

	Item ID	Item Name	Total purchase value
	94	Mourning Blade	10.92
	117	Heartstriker, Legacy of the Light	9.42
	93	Apocalyptic Battlescythe	8.98
	90	Betrayer	8.24
	154	Feral Katana	8.22

```
In [347]: df_pymoli8= pd.DataFrame({"Item Price":Item_price,'Purchase count': purchase_count4,
df_pymoli8["Item Price"]=df_pymoli8["Item Price"].map("$ {:.2f}".format)
df_pymoli8['Total purchase value']= df_pymoli8['Total purchase value'].map("{:,.2f}")
df_pymoli8.sort_values("Total purchase value", ascending= False).head(5)
```

Out [347]:

Item ID	Item Name	Item Price	Purchase count	\
117	Heartstriker, Legacy of the Light	\$ 4.71	2	
93	Apocalyptic Battlescythe	\$ 4.49	2	
90	Betrayer	\$ 4.12	2	
154	Feral Katana	\$ 4.11	2	
180	Stormcaller	\$ 2.77	2	

Item ID	Item Name	Total purchase value
117	Heartstriker, Legacy of the Light	\$9.42
93	Apocalyptic Battlescythe	\$8.98
90	Betrayer	\$8.24
154	Feral Katana	\$8.22
180	Stormcaller	\$5.54

In [348]: *#Not corrected sorting of the Total purchase value using the dollar sign... Please h*