

```
In [524]: import pandas as pd
import numpy as np
```

```
In [629]: # read first dataset
pymoli_1= 'Resources/purchase_data.json'
pymoli_1 =pd.read_json(pymoli_1)
pymoli_1.head()
```

Out[629]:

	Age	Gender	Item ID	Item Name	Price	SN
0	38	Male	165	Bone Crushing Silver Skewer	3.37	Aelalis34
1	21	Male	119	Stormbringer, Dark Blade of Ending Misery	2.32	Eolo46
2	34	Male	174	Primitive Blade	2.46	Assastnya25
3	21	Male	92	Final Critic	1.36	Pheusrical25
4	23	Male	63	Stormfury Mace	1.27	Aela59

```
In [630]: # read second dataset
pymoli_2='Resources/purchase_data2.json'
pymoli_2=pd.read_json(pymoli_2)
pymoli_2.head()
```

Out[630]:

	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 [631]: # combine two dataset
pymoli = pd.concat([pymoli_1, pymoli_2])
pymoli.head()
```

Out[631]:

	Age	Gender	Item ID	Item Name	Price	SN
0	38	Male	165	Bone Crushing Silver Skewer	3.37	Aelalis34
1	21	Male	119	Stormbringer, Dark Blade of Ending Misery	2.32	Eolo46
2	34	Male	174	Primitive Blade	2.46	Assastnya25
3	21	Male	92	Final Critic	1.36	Pheusrical25
4	23	Male	63	Stormfury Mace	1.27	Aela59

```
In [539]: #Player Count
total_player = len(pymoli['SN'].unique())
total_player_df = pd.DataFrame({'Total Players': total_player}, index=[0])
total_player_df
```

Out[539]:

	Total Players
0	612

```
In [540]: #Purchasing Analysis(Total)
unique_item = len(pymoli['Item Name'].unique())
average_price = pymoli['Price'].mean()
average_price="{:.2f}".format(average_price)
num_purchase = pymoli.shape[0]
total_revenue = pymoli['Price'].sum()
total_revenue = "{:.2f}".format(total_revenue)
purchasing_analysis_df=pd.DataFrame({'Number of Unique Items':unique_item, 'Average Price':average_price, 'Nuner of Purchase':num_purchase, 'Total Revenue':total_revenue}, index=[0])
purchasing_analysis_df
```

Out[540]:

	Average Price	Number of Unique Items	Nuner of Purchase	Total Revenue
0	\$2.93	180	858	\$2514.43

```
In [632]: #Create a cleaned dataframe with unique players
u_player = pymoli['SN'].unique()
u_gender = [list(pymoli['Gender'].loc[pymoli['SN'] == str(i)]) for i in u_player]
u_gender = [item[0] for item in u_gender]
u_pymoli_df = pd.DataFrame({'Player':u_player, 'Gender':u_gender})
u_pymoli_df.head()
```

Out[632]:

	Gender	Player
0	Male	Aelalis34
1	Male	Eolo46
2	Male	Assastnya25
3	Male	Pheusrical25
4	Male	Aela59

```
In [542]: #Gender Demographics
male = len(u_pymoli_df.loc[u_pymoli_df['Gender'] == 'Male'])
female = len(u_pymoli_df.loc[u_pymoli_df['Gender'] == 'Female'])
other = len(u_pymoli_df.loc[u_pymoli_df['Gender'] == 'Other / Non-Disclosed'])
m_percent = "{:.2f}%".format(male/total_player*100)
f_percent = "{:.2f}%".format(female/total_player*100)
other_percent = "{:.2f}%".format(other/total_player*100)
gender_df = pd.DataFrame({'Percentage of Player':[m_percent, f_percent,
other_percent], 'Total Count':[male, female, other]},index = ['Male','Female','Other / Non-Disclosed'])
gender_df
```

Out[542]:

	Percentage of Player	Total Count
Male	80.88%	495
Female	17.65%	108
Other / Non-Disclosed	1.47%	9

```
In [543]: #Purchasing Analysis(Gender)
```

```
In [544]: #purchase count
male_purchase = len(pymoli.loc[pymoli['Gender']=='Male'])
female_purchase = len(pymoli.loc[pymoli['Gender']=='Female'])
other_purchase = len(pymoli.loc[pymoli['Gender']=='Other / Non-Disclosed'])
total_purchase = male_purchase+female_purchase+other_purchase
```

```
In [545]: #groupby gender - avg price
avg_price_gender = pymoli.groupby('Gender').mean()
```

```
In [546]: #male and female avg price
m_avg_price = avg_price_gender['Price']['Male']
f_avg_price = avg_price_gender['Price']['Female']
other_avg_price = avg_price_gender['Price']['Other / Non-Disclosed']
total_avg_price = m_avg_price+f_avg_price+other_avg_price
```

```
In [547]: #total value
total_value_gender = pymoli.groupby('Gender').sum()
m_total_value = total_value_gender['Price']['Male']
f_total_value = total_value_gender['Price']['Female']
other_total_value = total_value_gender['Price']['Other / Non-Disclosed']
total_value = m_total_value+f_total_value+other_total_value
```

```
In [548]: #normalized total
m_normalized = total_value/male_purchase
f_normalized = total_value/female_purchase
other_normalized = total_value/other_purchase
```

```
In [549]: other_purchase = "${:.2f}".format(other_purchase)
m_avg_price = "${:.2f}".format(m_avg_price)
f_avg_price = "${:.2f}".format(f_avg_price)
other_avg_price = "${:.2f}".format(other_avg_price)
m_total_value = "${:.2f}".format(m_total_value)
f_total_value = "${:.2f}".format(f_total_value)
other_total_value = "${:.2f}".format(other_total_value)
m_normalized = "${:.2f}".format(m_normalized)
f_normalized = "${:.2f}".format(f_normalized)
other_normalized = "${:.2f}".format(other_normalized)
```

```
In [550]: gender_purchasing_df = pd.DataFrame({'Purchase Count':[male_purchase, fe
male_purchase, other_purchase], 'Average Purchase Price':[m_avg_price,f_
avg_price,other_avg_price ], 'Total Purchase Value':[m_total_value,f_tot
al_value,other_total_value], 'Normalized Totals':[m_normalized,f_normaliz
ed,other_normalized]}, index = ['Male','Female','Other / Non-Disclosed'
])
gender_purchasing_df
```

Out[550]:

	Average Purchase Price	Normalized Totals	Purchase Count	Total Purchase Value
Male	\$2.94	\$3.61	697	\$2052.28
Female	\$2.85	\$16.88	149	\$424.29
Other / Non-Disclosed	\$3.15	\$209.54	\$12.00	\$37.86

```
In [551]: #Age Demographics
min_age = pymoli['Age'].min()
max_age = pymoli['Age'].max()
min_age
```

Out[551]: 7

```
In [552]: max_age
```

Out[552]: 45

```
In [553]: bins = np.linspace(6,46,10, dtype = int)
bins
```

Out[553]: array([6, 10, 14, 19, 23, 28, 32, 37, 41, 46])

```
In [554]: labels = ['<=10','11-14','15-19','20-23','24-28','29-32','33-37','38-41',
,'42+']
```

```
In [633]: pymoli['Age Range'] = pd.cut(pymoli['Age'], bins=bins, labels=labels)
pymoli.head()
```

Out[633]:

	Age	Gender	Item ID	Item Name	Price	SN	Age Range
0	38	Male	165	Bone Crushing Silver Skewer	3.37	Aelalis34	38-41
1	21	Male	119	Stormbringer, Dark Blade of Ending Misery	2.32	Eolo46	20-23
2	34	Male	174	Primitive Blade	2.46	Assastnya25	33-37
3	21	Male	92	Final Critic	1.36	Pheusrical25	20-23
4	23	Male	63	Stormfury Mace	1.27	Aela59	20-23

```
In [556]: age_grouped_df = pymoli.groupby('Age Range')
age_grouped_df.max()
```

Out[556]:

	Age	Gender	Item ID	Item Name	Price	SN
Age Range						
<=10	10	Male	177	Woeful Adamantite Claymore	4.89	Yarithsurgue62
11-14	14	Other / Non-Disclosed	183	Woeful Adamantite Claymore	4.75	Undistasta86
15-19	19	Male	183	Worldbreaker	4.95	Zhisrisu83
20-23	23	Other / Non-Disclosed	183	Yearning Mageblade	4.95	Zontibe81
24-28	28	Other / Non-Disclosed	182	Yearning Mageblade	4.95	Zhisrisu83
29-32	32	Other / Non-Disclosed	182	Yearning Mageblade	4.95	Yathecal72
33-37	37	Other / Non-Disclosed	183	Wolf, Promise of the Moonwalker	4.83	Wailin72
38-41	40	Male	181	Woeful Adamantite Claymore	4.65	Yasrisu92
42+	45	Male	124	Venom Claymore	3.81	Raesurdil91

```
In [557]: age_purchasing = age_grouped_df ['Price'].count()  
age_purchasing
```

```
Out[557]: Age Range  
          <=10      37  
          11-14     34  
          15-19    144  
          20-23    295  
          24-28    190  
          29-32     70  
          33-37     54  
          38-41     31  
          42+        3  
          Name: Price, dtype: int64
```

```
In [558]: avg_price_age = age_grouped_df ['Price'].mean()  
total_value_age = age_grouped_df ['Price'].sum()  
total_value_age
```

```
Out[558]: Age Range  
          <=10    110.44  
          11-14     92.75  
          15-19    416.83  
          20-23    858.33  
          24-28    564.81  
          29-32    210.06  
          33-37    154.87  
          38-41     97.70  
          42+       8.64  
          Name: Price, dtype: float64
```

```
In [559]: normalized_totals_age = total_value_age/age_purchasing
```

```
In [560]: avg_price_age = avg_price_age.map('${:.2f}'.format)  
total_value_age = total_value_age.map('${:.2f}'.format)  
normalized_totals_age = normalized_totals_age.map('${:.2f}'.format)
```

```
In [561]: age_purchasing_df = pd.DataFrame()
age_purchasing_df['Purchase Count'] = age_purchasing
age_purchasing_df['Average Purchase Price'] = avg_price_age
age_purchasing_df['Total Purchase Value'] = total_value_age
age_purchasing_df['Normalized Totals'] = normalized_totals_age
age_purchasing_df
```

Out[561]:

	Purchase Count	Average Purchase Price	Total Purchase Value	Normalized Totals
Age Range				
<=10	37	\$2.98	\$110.44	\$2.98
11-14	34	\$2.73	\$92.75	\$2.73
15-19	144	\$2.89	\$416.83	\$2.89
20-23	295	\$2.91	\$858.33	\$2.91
24-28	190	\$2.97	\$564.81	\$2.97
29-32	70	\$3.00	\$210.06	\$3.00
33-37	54	\$2.87	\$154.87	\$2.87
38-41	31	\$3.15	\$97.70	\$3.15
42+	3	\$2.88	\$8.64	\$2.88

```
In [562]: #Top Spenders
top_total_value = pymoli['Price'].groupby(pymoli['SN']).sum().nlargest(5)
top_purchase_count = pymoli['Price'].groupby(pymoli['SN']).count()
top_avg = pymoli['Price'].groupby(pymoli['SN']).mean()
top_total_value = top_total_value.map('{:.2f}'.format)
top_total_value
```

Out[562]: SN
Undirrala66 \$17.06
Aerithllora36 \$15.10
Saedue76 \$13.56
Sondim43 \$13.02
Mindimnya67 \$12.74
Name: Price, dtype: object

```
In [563]: top_spenders_df = pd.DataFrame()
top_spenders_df['Total Purchase Value'] = top_total_value
top_spenders_df
```

Out[563]:

	Total Purchase Value
SN	
Undirrala66	\$17.06
Aerithllora36	\$15.10
Saedue76	\$13.56
Sondim43	\$13.02
Mindimnya67	\$12.74

```
In [564]: top_index = list(top_spenders_df.index)
top_index
```

Out[564]: ['Undirrala66', 'Aerithllora36', 'Saedue76', 'Sondim43', 'Mindimnya67']

```
In [565]: top_s_purchase_count = [top_purchase_count[element] for element in top_index]
top_s_avg = [top_avg[element] for element in top_index]
```

```
In [566]: top_spenders_df = pd.DataFrame()
top_spenders_df['Total Purchase Value'] = top_total_value
top_spenders_df['Purchase Count'] = top_s_purchase_count
top_spenders_df['Average Purchase Price'] = top_s_avg
top_spenders_df['Average Purchase Price'] = top_spenders_df['Average Purchase Price'].map('${:.2f}'.format)
top_spenders_df
```

Out[566]:

	Total Purchase Value	Purchase Count	Average Purchase Price
SN			
Undirrala66	\$17.06	5	\$3.41
Aerithllora36	\$15.10	4	\$3.77
Saedue76	\$13.56	4	\$3.39
Sondim43	\$13.02	4	\$3.25
Mindimnya67	\$12.74	4	\$3.18

```
In [617]: popular_grouped = pymoli.groupby(['Item ID', 'Item Name']).count().sort_values(['Price'], ascending=False)
p_purchase_value = pymoli.groupby(['Item ID', 'Item Name']).sum()
p_avg = pymoli.groupby(['Item ID', 'Item Name']).mean()
p_count = pymoli.groupby(['Item ID', 'Item Name']).count()
```



```
In [636]: p_index = list(popular_grouped.index)
```

```
In [637]: p_total = [p_purchase_value['Price'][element] for element in pop_index]
p_avg_price = [p_avg['Price'][element] for element in pop_index]
p_purchase_count = [p_count['Price'][element] for element in pop_index]
```

```
In [638]: popular_grouped['Purchase Count'] = p_purchase_count
popular_grouped['Item Price'] = p_avg_price
popular_grouped['Total Purchase Value'] = p_total
popular_grouped = popular_grouped [['Purchase Count', 'Item Price', 'Total
Purchase Value']]
popular_grouped['Item Price'] = popular_grouped['Item Price'].map('${:.2f}'.format)
popular_grouped['Total Purchase Value'] = popular_grouped['Total Purchase Value'].map('${:.2f}'.format)
popular_grouped.head()
```

Out[638]:

		Purchase Count	Item Price	Total Purchase Value
Item ID	Item Name			
84	Arcane Gem	12	\$2.45	\$29.34
39	Betrayal, Whisper of Grieving Widows	11	\$2.35	\$25.85
31	Trickster	10	\$2.32	\$23.22
44	Bonecarvin Battle Axe	9	\$2.67	\$24.04
154	Feral Katana	9	\$2.62	\$23.55

```
In [639]: #Most Profitable Item
profitable_grouped = pymoli.groupby(['Item ID', 'Item Name']).sum().sort_values(['Price'], ascending=False)
profitable_ct = pymoli.groupby(['Item ID', 'Item Name']).count()
profitable_avg = pymoli.groupby(['Item ID', 'Item Name']).mean()
profitable_total = pymoli.groupby(['Item ID', 'Item Name']).sum()
profitable_grouped.head()
```

Out[639]:

		Age	Price
Item ID	Item Name		
34	Retribution Axe	234	37.26
107	Splitter, Foe Of Subtlety	222	33.03
115	Spectral Diamond Doomblade	154	29.75
32	Orenmir	140	29.70
84	Arcane Gem	268	29.34

```
In [640]: profitable_index = list(profitable_grouped.index)
```

```
In [641]: pro_total = [profitable_total['Price'][element] for element in profitable_index]
pro_avg = [profitable_avg['Price'][element] for element in profitable_index]
pro_purchase_count = [profitable_ct['Price'][element] for element in profitable_index]
```

```
In [642]: profitable_grouped['Purchase Count'] = pro_purchase_count
profitable_grouped['Item Price'] = pro_avg
profitable_grouped['Total Purchase Value'] = pro_total
profitable_grouped = profitable_grouped[['Purchase Count', 'Item Price', 'Total Purchase Value']]
profitable_grouped['Item Price'] = profitable_grouped['Item Price'].map('${:.2f}'.format)
profitable_grouped['Total Purchase Value'] = profitable_grouped['Total Purchase Value'].map('${:.2f}'.format)
profitable_grouped.head()
```

Out[642]:

		Purchase Count	Item Price	Total Purchase Value
Item ID	Item Name			
34	Retribution Axe	9	\$4.14	\$37.26
107	Splitter, Foe Of Subtlety	9	\$3.67	\$33.03
115	Spectral Diamond Doomblade	7	\$4.25	\$29.75
32	Orenmir	6	\$4.95	\$29.70
84	Arcane Gem	12	\$2.45	\$29.34