

Heroes of Pymoli

GWU Data Analytics Bootcamp Homework 4

```
import pandas as pd
import numpy as np

json = "Resources/Heroes.JSON"
df = pd.read_json(json)
```

Player Count

```
num_players = df['SN'].nunique()
pd.DataFrame({"Total Players": [num_players]})
```

	Total Players
0	573

Purchasing Analysis (Total)

```
unique_items_count = df['Item ID'].nunique()
avg_purchase_price = df['Price'].mean()
total_purchases = df['Price'].count()
total_revenue = df['Price'].sum()

tot_purchases_df = pd.DataFrame({"Number of Unique Items": [unique_items_count],
                                "Average Price": [avg_purchase_price], "Number of Purchases": [total_purchases],
                                "Total Revenue": [total_revenue]})
tot_purchases_df = tot_purchases_df.reindex(['Number of Unique Items', 'Average Price', 'Number of Purchases', 'Total Revenue'], axis=1)
tot_purchases_df['Average Price'] = tot_purchases_df['Average Price'].map("${:.2f}".format)
tot_purchases_df['Total Revenue'] = tot_purchases_df['Total Revenue'].map("${:,.2f}".format)
tot_purchases_df
```

	Number of Unique Items	Average Price	Number of Purchases	Total Revenue
0	183	\$2.93	780	\$2,286.33

Gender Demographics

```
sn_unique_df = df.drop_duplicates(subset="SN")
sn_unique_gender_df = sn_unique_df.groupby(['Gender'])
sn_unique_grouped_gender_count = sn_unique_gender_df[['Gender']].count()
sn_unique_grouped_gender_percent = (sn_unique_gender_df[['Gender']].count() /
sn_unique_gender_df[['Gender']].count().sum()) * 100
sn_unique_gender_demo_df = pd.merge(sn_unique_grouped_gender_percent,
sn_unique_grouped_gender_count, left_index=True, right_index=True, how='outer')
```

```

sn_unique_gender_demo_df =
sn_unique_gender_demo_df.rename(columns={"Gender_x":"Percent of Players",
"Gender_y":"Total Count"})
sn_unique_gender_demo_df['Percent of Players'] = sn_unique_gender_demo_df['Percent
of Players'].map("{:.2f}%".format)
sn_unique_gender_demo_df.sort_values('Percent of Players', ascending=False)

```

	Percent of Players	Total Count
Gender		
Male	81.15%	465
Female	17.45%	100
Other / Non-Disclosed	1.40%	8

Purchasing Analysis (Gender)

```

grouped_gender_df = df.groupby(['Gender'])
gender_purchase_df = grouped_gender_df.agg({'Gender':['count'], 'Price':['mean',
'sum']})
norm_gender_tots = grouped_gender_df['Price'].sum() /
(grouped_gender_df['Gender'].count() / grouped_gender_df['Gender'].count().sum())
gender_purchase_df['Normalized Totals'] = norm_gender_tots
gender_purchase_df.columns = gender_purchase_df.columns.droplevel()
gender_purchase_df = gender_purchase_df.rename(columns={"count":"Purchase Count",
"mean":"Average Purchase Price", "sum":"Total Purchase Value", "":"Normalized
Totals"})
gender_purchase_df['Average Purchase Price'] = gender_purchase_df['Average Purchase
Price'].map("${:.2f}".format)
gender_purchase_df['Total Purchase Value'] = gender_purchase_df['Total Purchase
Value'].map("${:, .2f}".format)
gender_purchase_df['Normalized Totals'] = gender_purchase_df['Normalized
Totals'].map("${:, .2f}".format)
gender_purchase_df

```

	Purchase Count	Average Purchase Price	Total Purchase Value	Normalized Totals
Gender				
Female	136	\$2.82	\$382.91	\$2,196.10
Male	633	\$2.95	\$1,867.68	\$2,301.41
Other / Non-Disclosed	11	\$3.25	\$35.74	\$2,534.29

Age Demographics

```

bins = [0, 9, 14, 19, 24, 29, 34, 39, 44, 49]
group_names = ['Under 10', '10-14', '15-19', '20-24', '25-29', '30-34', '35-39',
'40-44', '45-49']
df['Age Bracket'] = pd.cut(df["Age"], bins, labels=group_names)

sn_unique_df = df.drop_duplicates(subset="SN")
sn_unique_age_df = sn_unique_df.groupby(['Age Bracket'])
sn_unique_grouped_age_count = sn_unique_age_df[['Age']].count()

```

```

sn_unique_grouped_age_percent = (sn_unique_age_df[['Age']].count() /
sn_unique_age_df[['Age']].count().sum()) * 100
sn_unique_age_demo_df = pd.merge(sn_unique_grouped_age_percent,
sn_unique_grouped_age_count, left_index=True, right_index=True, how='outer')
sn_unique_age_demo_df = sn_unique_age_demo_df.rename(columns={"Age_x": "Percent of
Players", "Age_y": "Total Count"})
sn_unique_age_demo_df['Percent of Players'] = sn_unique_age_demo_df['Percent of
Players'].map("{:.2f}%".format)
sn_unique_age_demo_df

```

	Percent of Players	Total Count
Age Bracket		
Under 10	3.32%	19
10-14	4.01%	23
15-19	17.45%	100
20-24	45.20%	259
25-29	15.18%	87
30-34	8.20%	47
35-39	4.71%	27
40-44	1.75%	10
45-49	0.17%	1

Purchasing Analysis (Age)

```

grouped_age_df = df.groupby(['Age Bracket'])
age_purchase_df = df.groupby(['Age Bracket']).agg({'Age': ['count'], 'Price': ['mean',
'sum']})
age_purchase_df['Normalized Totals'] = grouped_age_df['Price'].sum() /
(grouped_age_df['Age'].count() / grouped_age_df['Age'].count().sum())
age_purchase_df.columns = age_purchase_df.columns.droplevel()
age_purchase_df = age_purchase_df.rename(columns={"count": "Purchase Count",
"mean": "Average Purchase Price", "sum": "Total Purchase Value", "" : "Normalized
Totals"})
age_purchase_df['Average Purchase Price'] = age_purchase_df['Average Purchase
Price'].map("${:.2f}".format)
age_purchase_df['Total Purchase Value'] = age_purchase_df['Total Purchase
Value'].map("${:, .2f}".format)
age_purchase_df['Normalized Totals'] = age_purchase_df['Normalized
Totals'].map("${:, .2f}".format)
age_purchase_df

```

	Purchase Count	Average Purchase Price	Total Purchase Value	Normalized Totals
Age Bracket				
Under 10	28	\$2.98	\$83.46	\$2,324.96
10-14	35	\$2.77	\$96.95	\$2,160.60
15-19	133	\$2.91	\$386.42	\$2,266.22
20-24	336	\$2.91	\$978.77	\$2,272.14
25-29	125	\$2.96	\$370.33	\$2,310.86

	Purchase Count	Average Purchase Price	Total Purchase Value	Normalized Totals
Age Bracket				
30-34	64	\$3.08	\$197.25	\$2,403.98
35-39	42	\$2.84	\$119.40	\$2,217.43
40-44	16	\$3.19	\$51.03	\$2,487.71
45-49	1	\$2.72	\$2.72	\$2,121.60

Top Spenders

```
top_spenders_df = df.groupby(['SN']).agg({'Price':['count', 'mean', 'sum']})
top_spenders_df.columns = top_spenders_df.columns.droplevel()
top_spenders_df = top_spenders_df.rename(columns={"count":"Purchase Count",
"mean":"Average Purchase Price", "sum":"Total Purchase Value"})
top_spenders_df = top_spenders_df.sort_values('Total Purchase Value',
ascending=False)
top_spenders_df = top_spenders_df.head()
top_spenders_df['Average Purchase Price'] = top_spenders_df['Average Purchase
Price'].map("${:,.2f}".format)
top_spenders_df['Total Purchase Value'] = top_spenders_df['Total Purchase
Value'].map("${:,.2f}".format)
top_spenders_df
```

	Purchase Count	Average Purchase Price	Total Purchase Value
SN			
Undirrala66	5	\$3.41	\$17.06
Saedue76	4	\$3.39	\$13.56
Mindimnya67	4	\$3.18	\$12.74
Haellysu29	3	\$4.24	\$12.73
Eoda93	3	\$3.86	\$11.58

Most Popular Items

```
grouped_items_df = df.groupby(['Item ID', 'Item Name']).agg({'Price':['count',
'mean', 'sum']})
grouped_items_df.columns = grouped_items_df.columns.droplevel()
grouped_items_df = grouped_items_df.rename(columns={"count":"Purchase Count",
"mean":"Average Purchase Price", "sum":"Total Purchase Value"})
pop_items_df = grouped_items_df.sort_values('Purchase Count', ascending=False)
pop_items_df = pop_items_df.head()
pop_items_df['Average Purchase Price'] = pop_items_df['Average Purchase
Price'].map("${:,.2f}".format)
pop_items_df['Total Purchase Value'] = pop_items_df['Total Purchase
Value'].map("${:,.2f}".format)
pop_items_df
```

		Purchase Count	Average Purchase Price	Total Purchase Value
Item ID	Item Name			

		Purchase Count	Average Purchase Price	Total Purchase Value
Item ID	Item Name			
39	Betrayal, Whisper of Grieving Widows	11	\$2.35	\$25.85
84	Arcane Gem	11	\$2.23	\$24.53
31	Trickster	9	\$2.07	\$18.63
175	Woeful Adamantite Claymore	9	\$1.24	\$11.16
13	Serenity	9	\$1.49	\$13.41

Most Profitable Items

```
grouped_items_df = df.groupby(['Item ID', 'Item Name']).agg({'Price': ['count', 'mean', 'sum']})
grouped_items_df.columns = grouped_items_df.columns.droplevel()
grouped_items_df = grouped_items_df.rename(columns={"count": "Purchase Count", "mean": "Average Purchase Price", "sum": "Total Purchase Value"})
rev_items_df = grouped_items_df.sort_values('Total Purchase Value', ascending=False)
rev_items_df = rev_items_df.head()
rev_items_df['Average Purchase Price'] = rev_items_df['Average Purchase Price'].map("${:,.2f}".format)
rev_items_df['Total Purchase Value'] = rev_items_df['Total Purchase Value'].map("${:,.2f}".format)
rev_items_df
```

		Purchase Count	Average Purchase Price	Total Purchase Value
Item ID	Item Name			
34	Retribution Axe	9	\$4.14	\$37.26
115	Spectral Diamond Doomblade	7	\$4.25	\$29.75
32	Orenmir	6	\$4.95	\$29.70
103	Singed Scalpel	6	\$4.87	\$29.22
107	Splitter, Foe Of Subtlety	8	\$3.61	\$28.88