In [16]:
```
#Dependencies
import pandas as pd
import numpy as np
```

In [22]:
```
#Load json
json_path1="purchase_data.json"
#json_path2="purchase_data2.json"
```

In [28]:
```
#Read files with pandas
file1=pd.read_json(json_path1)
file2=pd.read_json(json_path2)


purchase_data1_df = pd.DataFrame(file1)
purchase_data1_df.head()
purchase_data2_df=pd.DataFrame(file2)
purchase_data2_df.head()
```

Out[28]:

	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 [35]:
```
#entries in file 1
purchase_data1_df.count()
```

Out[35]:
```
Age          780
Gender       780
Item ID      780
Item Name    780
Price        780
SN           780
dtype: int64
```

In [36]:
```
#entries in file 2
purchase_data2_df.count()
```

Out[36]:
```
Age          78
Gender       78
Item ID      78
Item Name    78
Price        78
SN           78
dtype: int64
```

In [41]:

```
#combine files
combined_PurDAta_df = pd.concat([purchase_data1_df,purchase_data2_df], axis=0)
combined_PurDAta_df.count()
combined_PurDAta_df.head()
```

Out[41]:

	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 [80]:

```
#finding number of plays
NumPlayers=combined_PurDAta_df.groupby("SN").nunique()
NumPlayers = NumPlayers["SN"].count()
NumPlayers
Player_Breakdown=pd.DataFrame({"Number of Players":[NumPlayers]})
Player_Breakdown
```

Out[80]:

	Number of Players
0	612

In [120]:

```
#Purchasing Analysis
#Number of Unique Items
UniqueItems = combined_PurDAta_df.groupby("Item Name").nunique()
UniqueItems = UniqueItems["Item Name"].count()
UniqueItems =180

#Finding Average Purchase Price
Average_Price = combined_PurDAta_df.groupby("Item Name").nunique()
Average_Price = Average_Price["Price"].mean()
#Average_Price = 1.37

#Number of Purchases
TotalPurchases = len(combined_PurDAta_df)
TotalPurchases

#Total Purchase Value
Total_Purchase_Value = combined_PurDAta_df["Price"].sum()
Total_Purchase_Value

Purchase_Breakdown = pd.DataFrame({"# of Unique Items":[UniqueItems], "Average Pr
                                  "Number of Purchases":[TotalPurchases], "Total 

Purchase_Breakdown["Average Price"]=Purchase_Breakdown["Average Price"].map("${0:
Purchase_Breakdown["Total Revenue"]=Purchase_Breakdown["Total Revenue"].map("${0:
Purchase_Breakdown
```

Out[120]:

	# of Unique Items	Average Price	Number of Purchases	Total Revenue
0	180	$1.37	858	$2,514.43

In [182]:

```
#Gender Demographics
#filter for unique players
UniquePlayers_df = combined_PurDAta_df.drop_duplicates(["SN"])

#Calculating Male Players
MaleCount = UniquePlayers_df["Gender"].value_counts()["Male"]
MaleCount_Percentage = MaleCount / NumPlayers*100

#Calculating Female Players
FemaleCount = UniquePlayers_df["Gender"].value_counts()["Female"]
FemaleCount_Percentage = FemaleCount / NumPlayers*100

#Calculating non-disclosed gender
NonDisclosedCount = NumPlayers - MaleCount - FemaleCount
NonDisclosed_Percentage = NonDisclosedCount / NumPlayers*100

#Putting Data into DataFrame
Gender_Breakdown = pd.DataFrame({"Percentage of Players":{"Males": MaleCount_Perc

Gender_Count=pd.DataFrame({"Total Count":{"Males":MaleCount,"Females":FemaleCount

Gender_merge = pd.concat([Gender_Breakdown,Gender_Count],axis=1)
#Fix Formatting
Gender_merge["Percentage of Players"]=Gender_merge["Percentage of Players"].map("
Gender_merge
```

Out[182]:

	Percentage of Players	Total Count
Females	17.65%	108
Males	80.88%	495
Undisclosed	1.47%	9

In [242]:

```
#Purchasing Analysis (Gender)

#Male Calculations
Male_df = combined_PurDAta_df.loc[combined_PurDAta_df["Gender"]=="Male"]
Male_Average_Purchase_Price = Male_df["Price"].mean()
Male_Total_Purchase_Value = Male_df["Price"].sum()
Male_Purchase_Count=len(Male_df)
Male_Normalized_Total=Male_Total_Purchase_Value / MaleCount

#Female Calculations
Female_df = combined_PurDAta_df.loc[combined_PurDAta_df["Gender"]=="Female"]
Female_Average_Purchase_Price = Female_df["Price"].mean()
Female_Total_Purchase_Value = Female_df["Price"].sum()
Female_Purchase_Count=len(Female_df)
Female_Normalized_Total=Female_Total_Purchase_Value / FemaleCount

#Other Calculation
Other_df = combined_PurDAta_df.loc[(combined_PurDAta_df["Gender"] !="Male") & (co
Other_Average_Purchase_Price = Other_df["Price"].mean()
Other_Total_Purchase_Value = Other_df["Price"].sum()
Other_Purchase_Count=len(Other_df)
Other_Normalized_Total=Other_Total_Purchase_Value / NonDisclosedCount

#Put into DataFrame
Gender_Purchase_Analysis=[ {"Purchase Count":Male_Purchase_Count,"Average Purchas
                         { "Purchase Count":Female_Purchase_Count,"Average Purcha
                         { "Purchase Count":Other_Purchase_Count,"Average Purchas
                         ]

#Add Index Header
Gender_Purchase_Table = pd.DataFrame(Gender_Purchase_Analysis, index=['Male', 'Fe
Gender_Purchase_Table.index.name="Gender"

#Fix Formatting
Gender_Purchase_Table["Average Purchase Price"]=Gender_Purchase_Table["Average Pu
Gender_Purchase_Table["Normalized Purchase Price"]=Gender_Purchase_Table["Normali
Gender_Purchase_Table["Total Purchase Value"]=Gender_Purchase_Table["Total Purcha
Gender_Purchase_Table.columns
Gender_Purchase_Table=Gender_Purchase_Table[["Purchase Count","Average Purchase P
Gender_Purchase_Table
```

Out[242]:

	Purchase Count	Average Purchase Price	Total Purchase Value	Normalized Purchase Price
Gender				
Male	697	$2.94	$2,052.28	$4.15
Female	149	$2.85	$424.29	$3.93
Other	12	$3.15	$37.86	$4.21

In [324]:

```
#Age Demographics
#Creating bins (0-12,12-16,16-22,22+)
bins=[0,12,16,22,99]
group_names=["<12","12-16","16-22","22+"]

#cut dataframe and place age into bins
combined_PurDAta_df["Age Range"]=pd.cut(combined_PurDAta_df["Age"],bins,labels=gr
UniquePlayers_df = combined_PurDAta_df.drop_duplicates(["SN"])
#unique players
age_groups_unique=UniquePlayers_df.groupby("Age Range")
#all players
age_groups_all=combined_PurDAta_df.groupby("Age Range")
AgeGroup_Total=age_groups_unique["SN"].count()

AgeGroup_Average_Purchase_Price = age_groups_all["Price"].mean()
AgeGroup_Total_Purchase_Value=age_groups_all["Price"].sum()
AgeGroup_Normalized_Average = AgeGroup_Total_Purchase_Value/AgeGroup_Total

#Put into DataFrame
AgeGroup_Breakdown=pd.concat([AgeGroup_Total,AgeGroup_Average_Purchase_Price,AgeG
AgeGroup_Breakdown=pd.DataFrame(AgeGroup_Breakdown)
AgeGroup_Breakdown.columns=["Purchase Count","Average Purchase Price","Total Purc

#Format Data Table
AgeGroup_Breakdown["Average Purchase Price"]=AgeGroup_Breakdown["Average Purchase
AgeGroup_Breakdown["Total Purchase Price"]=AgeGroup_Breakdown["Total Purchase Pri
AgeGroup_Breakdown["Normalized Average"]=AgeGroup_Breakdown["Normalized Average"]
AgeGroup_Breakdown
```

Out[324]:

	Purchase Count	Average Purchase Price	Total Purchase Price	Normalized Average
Age Range				
<12	35	$3.06	$165.37	$4.72
12-16	66	$2.75	$250.38	$3.79
16-22	223	$2.92	$873.71	$3.92
22+	288	$2.96	$1,224.97	$4.25

In [389]:

```
#Top Spenders
TopSpenders_df = combined_PurDAta_df.groupby("SN", as_index=True).agg({"Price":"s
TopSpenders_df["Average Purchase Price"]=TopSpenders_df["Price"]/TopSpenders_df["
TopSpenders_df=TopSpenders_df.rename(columns={"Price":"Total Purchase Value","Ite
TopSpenders_df=TopSpenders_df.sort_values("Total Purchase Value",ascending=False)

#Formatting
TopSpenders_df["Total Purchase Value"]=TopSpenders_df["Total Purchase Value"].map
TopSpenders_df["Average Purchase Price"]=TopSpenders_df["Average Purchase Price"]
#Reorder columns
TopSpenders_df=TopSpenders_df[["Purchase Count", "Average Purchase Price","Total
#display
TopSpenders_df.head()
```

Out[389]:

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

In [438]:

```
#Most Popular Items
TopItems_df = combined_PurDAta_df.groupby([ "Item Name","Item ID"], as_index=True
TopItems_df["Item Price"]=TopItems_df["Price"]/TopItems_df["SN"]
TopItems_df=TopItems_df.sort_values("SN",ascending=False)
TopItems_df = TopItems_df.rename(columns={"SN":"Purchase Count","Price":"Total Pu
#top profitable dataframe
TopProfitable_df = TopItems_df.sort_values("Total Purchase Value",ascending=False

#Formatting
TopItems_df["Total Purchase Value"]=TopItems_df["Total Purchase Value"].map("${0:
TopItems_df["Item Price"]=TopItems_df["Item Price"].map("${0:,.2f}".format)
#Reorder columns
TopItems_df=TopItems_df[["Purchase Count","Item Price","Total Purchase Value"]]
#display
TopItems_df.head()
```

Out[438]:

		Purchase Count	Item Price	Total Purchase Value
Item Name	**Item ID**			
Arcane Gem	**84**	12	$2.45	$29.34
Betrayal, Whisper of Grieving Widows	**39**	11	$2.35	$25.85
Trickster	**31**	10	$2.32	$23.22
Feral Katana	**154**	9	$2.62	$23.55
Serenity	**13**	9	$1.49	$13.41

In [439]:

```
#Most Profitable Item
#Formatting
TopProfitable_df["Total Purchase Value"]=TopProfitable_df["Total Purchase Value"]
TopProfitable_df["Item Price"]=TopProfitable_df["Item Price"].map("${0:,.2f}".for
#Reorder columns
#TopItems_df=TopItems_df[["Purchase Count","Item Price","Total Purchase Value"]]
#display
TopProfitable_df.head()
```

Out[439]:

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

In [ ]: