**Note**

* Instructions have been included for each segment. You do not have to follow them exactly, but they are included to help you think through the steps.

In [9]:



*# Dependencies and Setup*

**import** pandas **as** pd

​

*# File to Load (Remember to Change These)*

file\_to\_load **=** "Resources/purchase\_data.csv"

​

*# Read Purchasing File and store into Pandas data frame*

purchase\_data **=** pd.read\_csv(file\_to\_load)

**Player Count**

* Display the total number of players

In [12]:



*#Use the length of list of screen names "SN", for total players*

total\_players **=** len(purchase\_data["SN"].value\_counts())

​

*#Create a data frame with total players named player count*

player\_count **=** pd.DataFrame({"Total Players":[total\_players]})

player\_count

Out[12]:

|  | **Total Players** |
| --- | --- |
| **0** | 576 |

**Purchasing Analysis (Total)**

* Run basic calculations to obtain number of unique items, average price, etc.
* Create a summary data frame to hold the results
* Optional: give the displayed data cleaner formatting
* Display the summary data frame

In [17]:



purchase\_data.head()

Out[17]:

|  | **Purchase ID** | **SN** | **Age** | **Gender** | **Item ID** | **Item Name** | **Price** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 0 | Lisim78 | 20 | Male | 108 | Extraction, Quickblade Of Trembling Hands | 3.53 |
| **1** | 1 | Lisovynya38 | 40 | Male | 143 | Frenzied Scimitar | 1.56 |
| **2** | 2 | Ithergue48 | 24 | Male | 92 | Final Critic | 4.88 |
| **3** | 3 | Chamassasya86 | 24 | Male | 100 | Blindscythe | 3.27 |
| **4** | 4 | Iskosia90 | 23 | Male | 131 | Fury | 1.44 |

In [117]:



*#Calculate for unique items, average price, purchase count, and revenue*

number\_of\_unique\_items **=** len((purchase\_data["Item ID"]).unique())

average\_price **=** (purchase\_data["Price"]).mean()

number\_of\_purchases **=** (purchase\_data["Purchase ID"]).count()

total\_revenue **=** (purchase\_data["Price"]).sum()

​

*#Create data frame with obtained values*

summary\_df **=** pd.DataFrame({"Number of Unique Items":[number\_of\_unique\_items],

"Average Price":[average\_price],

"Number of Purchases": [number\_of\_purchases],

"Total Revenue": [total\_revenue]})

​

*#Format with currency style*

summary\_df.style.format({'Average Price':"${:,.2f}",

'Total Revenue': '${:,.2f}'})

​

Out[117]:

|  | **Number of Unique Items** | **Average Price** | **Number of Purchases** | **Total Revenue** |
| --- | --- | --- | --- | --- |
| **0** | 179 | $3.05 | 780 | $2,379.77 |

**Gender Demographics**

* Percentage and Count of Male Players
* Percentage and Count of Female Players
* Percentage and Count of Other / Non-Disclosed

In [87]:



*#Group purchase\_data by Gender*

gender\_stats **=** purchase\_data.groupby("Gender")

​

*#Count the total of screen names "SN" by gender*

total\_count\_gender **=** gender\_stats.nunique()["SN"]

​

*#Take total count by gender and divide by total players*

percentage\_of\_players **=** total\_count\_gender **/** total\_players **\*** 100

​

*#Create data frame with obtained values*

gender\_demographics **=** pd.DataFrame({"Percentage of Players": percentage\_of\_players, "Total Count": total\_count\_gender})

​

*#Format data frame with no index name in the corner*

gender\_demographics.index.name **=** **None**

​

*#Format values sorted by total count in descending order, along with two decimal places for the percentage*

gender\_demographics.sort\_values(["Total Count"], ascending **=** **False**).style.format({"Percentage of Players":"{:.2f}"})

​

Out[87]:

|  | **Percentage of Players** | **Total Count** |
| --- | --- | --- |
| **Male** | 84.03 | 484 |
| **Female** | 14.06 | 81 |
| **Other / Non-Disclosed** | 1.91 | 11 |

**Purchasing Analysis (Gender)**

* Run basic calculations to obtain purchase count, avg. purchase price, avg. purchase total per person etc. by gender
* Create a summary data frame to hold the results
* Optional: give the displayed data cleaner formatting
* Display the summary data frame

In [114]:



*#Count the total purchases by gender*

purchase\_count **=** gender\_stats["Purchase ID"].count()

​

*#Average purchase prices by gender*

avg\_purchase\_price **=** gender\_stats["Price"].mean()

​

*#Average purchase total by gender*

avg\_purchase\_total **=** gender\_stats["Price"].sum()

​

​

*#Average pruchse total by gender divided by purchase count by uniquie shoppers*

avg\_purchase\_per\_person **=** avg\_purchase\_total**/**total\_count\_gender

​

*#Create data frame with obtained values*

gender\_demographics **=** pd.DataFrame({"Purchase Count": purchase\_count,

"Average Purchase Price": avg\_purchase\_price,

"Average Purchase Value":avg\_purchase\_total,

"Avg Purchase Total per Person": avg\_purchase\_per\_person})

​

*#Place index at top left as "Gender"*

gender\_demographics.index.name **=** "Gender"

​

*#Format with currency style*

gender\_demographics.style.format({"Average Purchase Value":"${:,.2f}",

"Average Purchase Price":"${:,.2f}",

"Avg Purchase Total per Person":"${:,.2f}"})

Out[114]:

|  | **Purchase Count** | **Average Purchase Price** | **Average Purchase Value** | **Avg Purchase Total per Person** |
| --- | --- | --- | --- | --- |
| **Gender** |  |  |  |  |
| **Female** | 113 | $3.20 | $361.94 | $4.47 |
| **Male** | 652 | $3.02 | $1,967.64 | $4.07 |
| **Other / Non-Disclosed** | 15 | $3.35 | $50.19 | $4.56 |

**Age Demographics**

* Establish bins for ages
* Categorize the existing players using the age bins. Hint: use pd.cut()
* Calculate the numbers and percentages by age group
* Create a summary data frame to hold the results
* Optional: round the percentage column to two decimal points
* Display Age Demographics Table

In [92]:



*#Establish bins for ages*

age\_bins **=** [0, 9.90, 14.90, 19.90, 24.90, 29.90, 34.90, 39.90, 99999]

group\_names **=** ["<10", "10-14", "15-19", "20-24", "25-29", "30-34", "35-39", "40+"]

​

In [93]:



*#Segment and sort age values into bins established above*

purchase\_data["Age Group"] **=** pd.cut(purchase\_data["Age"],age\_bins, labels**=**group\_names)

purchase\_data

​

Out[93]:

|  | **Purchase ID** | **SN** | **Age** | **Gender** | **Item ID** | **Item Name** | **Price** | **Age Group** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 0 | Lisim78 | 20 | Male | 108 | Extraction, Quickblade Of Trembling Hands | 3.53 | 20-24 |
| **1** | 1 | Lisovynya38 | 40 | Male | 143 | Frenzied Scimitar | 1.56 | 40+ |
| **2** | 2 | Ithergue48 | 24 | Male | 92 | Final Critic | 4.88 | 20-24 |
| **3** | 3 | Chamassasya86 | 24 | Male | 100 | Blindscythe | 3.27 | 20-24 |
| **4** | 4 | Iskosia90 | 23 | Male | 131 | Fury | 1.44 | 20-24 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |
| **775** | 775 | Aethedru70 | 21 | Female | 60 | Wolf | 3.54 | 20-24 |
| **776** | 776 | Iral74 | 21 | Male | 164 | Exiled Doomblade | 1.63 | 20-24 |
| **777** | 777 | Yathecal72 | 20 | Male | 67 | Celeste, Incarnation of the Corrupted | 3.46 | 20-24 |
| **778** | 778 | Sisur91 | 7 | Male | 92 | Final Critic | 4.19 | <10 |
| **779** | 779 | Ennrian78 | 24 | Male | 50 | Dawn | 4.60 | 20-24 |

780 rows × 8 columns

In [123]:



*#Create new data frame with the added "Age Group" and group it*

age\_grouped **=** purchase\_data.groupby("Age Group")

​

*#Count total players by age category*

total\_count\_age **=** age\_grouped["SN"].nunique()

​

*#Calculate percentages by age category*

percentage\_by\_age **=** (total\_count\_age**/**total\_players) **\*** 100

​

*#Create data frame with obtained values*

age\_demographics **=** pd.DataFrame({"Age Group": purchase\_data.groupby,

"Percentage of Players": percentage\_by\_age,

"Total Count": total\_count\_age,})

*#Format the data frame with no index name in the corner*

age\_demographics.index.name **=** **None**

​

*#Format percentage with two decimal places*

age\_demographics.style.format({"Percentage of Players":"{:,.2f}%"})

​

Out[123]:

|  | **Age Group** | **Percentage of Players** | **Total Count** |
| --- | --- | --- | --- |
| **<10** |  | 2.95% | 17 |
| **10-14** |  | 3.82% | 22 |
| **15-19** |  | 18.58% | 107 |
| **20-24** |  | 44.79% | 258 |
| **25-29** |  | 13.37% | 77 |
| **30-34** |  | 9.03% | 52 |
| **35-39** |  | 5.38% | 31 |
| **40+** |  | 2.08% | 12 |

**Purchasing Analysis (Age)**

* Bin the purchase\_data data frame by age
* Run basic calculations to obtain purchase count, avg. purchase price, avg. purchase total per person etc. in the table below
* Create a summary data frame to hold the results
* Optional: give the displayed data cleaner formatting
* Display the summary data frame

In [124]:



*#Count purchases by age group*

purchase\_count\_age **=** age\_grouped["Purchase ID"].count()

​

*#Obtain average purchase price by age group*

avg\_purchase\_price\_age **=** age\_grouped["Price"].mean()

​

*#Calculate total purchase value by age group*

total\_purchase\_value **=** age\_grouped["Price"].sum()

​

*#Calculate the average purchase per person in the age group*

avg\_purchase\_per\_person\_age **=** total\_purchase\_value**/**total\_count\_age

​

*#Create data frame with obtained values*

age\_demographics **=** pd.DataFrame({"Purchase Count": purchase\_count\_age,

"Average Purchase Price": avg\_purchase\_price\_age,

"Total Purchase Value":total\_purchase\_value,

"Average Purchase Total per Person": avg\_purchase\_per\_person\_age})

​

*#Format the data frame with no index name in the corner*

age\_demographics.index.name **=** **None**

​

*#Format with currency style*

age\_demographics.style.format({"Average Purchase Price":"${:,.2f}",

"Total Purchase Value":"${:,.2f}",

"Average Purchase Total per Person":"${:,.2f}"})

Out[124]:

|  | **Purchase Count** | **Average Purchase Price** | **Total Purchase Value** | **Average Purchase Total per Person** |
| --- | --- | --- | --- | --- |
| **<10** | 23 | $3.35 | $77.13 | $4.54 |
| **10-14** | 28 | $2.96 | $82.78 | $3.76 |
| **15-19** | 136 | $3.04 | $412.89 | $3.86 |
| **20-24** | 365 | $3.05 | $1,114.06 | $4.32 |
| **25-29** | 101 | $2.90 | $293.00 | $3.81 |
| **30-34** | 73 | $2.93 | $214.00 | $4.12 |
| **35-39** | 41 | $3.60 | $147.67 | $4.76 |
| **40+** | 13 | $2.94 | $38.24 | $3.19 |

**Top Spenders**

* Run basic calculations to obtain the results in the table below
* Create a summary data frame to hold the results
* Sort the total purchase value column in descending order
* Optional: give the displayed data cleaner formatting
* Display a preview of the summary data frame

In [109]:



*#Group purchase data by screen names*

spender\_stats **=** purchase\_data.groupby("SN")

​

*#Count the total purchases by name*

purchase\_count\_spender **=** spender\_stats["Purchase ID"].count()

​

*#Calculate the average purchase by name*

avg\_purchase\_price\_spender **=** spender\_stats["Price"].mean()

​

*#Calculate purchase total*

purchase\_total\_spender **=** spender\_stats["Price"].sum()

​

*#Create data frame with obtained values*

top\_spenders **=** pd.DataFrame({"Purchase Count": purchase\_count\_spender,

"Average Purchase Price": avg\_purchase\_price\_spender,

"Total Purchase Value":purchase\_total\_spender})

​

*#Sort in descending order to obtain top 5 spender names*

formatted\_spenders **=** top\_spenders.sort\_values(["Total Purchase Value"], ascending**=False**).head()

​

*#Format with currency style*

formatted\_spenders.style.format({"Average Purchase Total":"${:,.2f}",

"Average Purchase Price":"${:,.2f}",

"Total Purchase Value":"${:,.2f}"})

​

Out[109]:

|  | **Purchase Count** | **Average Purchase Price** | **Total Purchase Value** |
| --- | --- | --- | --- |
| **SN** |  |  |  |
| **Lisosia93** | 5 | $3.79 | $18.96 |
| **Idastidru52** | 4 | $3.86 | $15.45 |
| **Chamjask73** | 3 | $4.61 | $13.83 |
| **Iral74** | 4 | $3.40 | $13.62 |
| **Iskadarya95** | 3 | $4.37 | $13.10 |

**Most Popular Items**

* Retrieve the Item ID, Item Name, and Item Price columns
* Group by Item ID and Item Name. Perform calculations to obtain purchase count, average item price, and total purchase value
* Create a summary data frame to hold the results
* Sort the purchase count column in descending order
* Optional: give the displayed data cleaner formatting
* Display a preview of the summary data frame

In [110]:



*#Create new data frame with items related information*

items **=** purchase\_data[["Item ID", "Item Name", "Price"]]

​

*#Group the item data by item id and item name*

item\_stats **=** items.groupby(["Item ID","Item Name"])

​

*#Count the number of times an item has been purchased*

purchase\_count\_item **=** item\_stats["Price"].count()

​

*#Calcualte the purchase value per item*

purchase\_value **=** (item\_stats["Price"].sum())

​

*#Find individual item price*

item\_price **=** purchase\_value**/**purchase\_count\_item

​

*#Create data frame with obtained values*

most\_popular\_items **=** pd.DataFrame({"Purchase Count": purchase\_count\_item,

"Item Price": item\_price,

"Total Purchase Value":purchase\_value})

​

*#Sort in descending order to obtain top spender names and provide top 5 item names*

popular\_formatted **=** most\_popular\_items.sort\_values(["Purchase Count"], ascending**=False**).head()

​

*#Format with currency style*

popular\_formatted.style.format({"Item Price":"${:,.2f}",

"Total Purchase Value":"${:,.2f}"})

​

Out[110]:

|  |  | **Purchase Count** | **Item Price** | **Total Purchase Value** |
| --- | --- | --- | --- | --- |
| **Item ID** | **Item Name** |  |  |  |
| **92** | **Final Critic** | 13 | $4.61 | $59.99 |
| **178** | **Oathbreaker, Last Hope of the Breaking Storm** | 12 | $4.23 | $50.76 |
| **145** | **Fiery Glass Crusader** | 9 | $4.58 | $41.22 |
| **132** | **Persuasion** | 9 | $3.22 | $28.99 |
| **108** | **Extraction, Quickblade Of Trembling Hands** | 9 | $3.53 | $31.77 |

**Most Profitable Items**

* Sort the above table by total purchase value in descending order
* Optional: give the displayed data cleaner formatting
* Display a preview of the data frame

In [111]:



*#Take the most\_popular items data frame and change the sorting to find highest total purchase value*

popular\_formatted **=** most\_popular\_items.sort\_values(["Total Purchase Value"],

ascending**=False**).head()

*# Format with currency style*

popular\_formatted.style.format({"Item Price":"${:,.2f}",

"Total Purchase Value":"${:,.2f}"})

​

Out[111]:

|  |  | **Purchase Count** | **Item Price** | **Total Purchase Value** |
| --- | --- | --- | --- | --- |
| **Item ID** | **Item Name** |  |  |  |
| **92** | **Final Critic** | 13 | $4.61 | $59.99 |
| **178** | **Oathbreaker, Last Hope of the Breaking Storm** | 12 | $4.23 | $50.76 |
| **82** | **Nirvana** | 9 | $4.90 | $44.10 |
| **145** | **Fiery Glass Crusader** | 9 | $4.58 | $41.22 |
| **103** | **Singed Scalpel** | 8 | $4.35 | $34.80 |