In [29]:
```
# Dependencies and Setup
import pandas as pd

file_imported = "Resources/purchase_data.csv"

purchase_data = pd.read_csv(file_imported)
purchase_data.head()
```

Out[29]:

	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

Player Count

In [30]:
```
# Total players is obtained from lenght of screen name 'SN'.
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[30]:

	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 [33]:
```
# Basic calculations to obtain number of unique items, average price,
# number of purchase count, and total 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()

# Creating asummary data frame to obtained the results
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]})

# Formatting using currency style
summary_df.style.format({'Average Price':"${:,.2f}",
                         'Total Revenue': '${:,.2f}'})
```

Out[33]:

	Number of Unique Items	Average Price	Number of Purchases	Total Revenue
0	183	$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 [32]:
```
# Grouping based on Gender
Gender_group = purchase_data.groupby("Gender")
# Using the screen names for total count based on Gender
Total_gender_count = Gender_group.nunique()["SN"]
Total_gender_count.head()
```

Out[32]:
```
Gender
Female                   81
Male                    484
Other / Non-Disclosed    11
Name: SN, dtype: int64
```

In [36]:
```
Percentage_of_players = Total_gender_count / Total_players * 100
Gender_demographics = pd.DataFrame({"Percentage of Players": Percentage_of_players,
                                    "Total Count": Total_gender_count})
# Formating the Gender demographic with none index name
Gender_demographics.index.name = None
# Formating the Percentage of players to 2 deimal places
# and Total count in descending order
Gender_demographics.sort_values(["Total Count"],
                                ascending = False).style.format({"Percentage of Players":"{:.2f}"})
```

Out[36]:

	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 [22]:
```
Purchase_count = Gender_group["Purchase ID"].count()

Avg_purchase_price = Gender_group["Price"].mean()

Avg_purchase_total = Gender_group["Price"].sum()

Avg_purchase_per_person = Avg_purchase_total/Total_gender_count

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})

Gender_demographics.index.name = "Gender"

Gender_demographics.style.format({"Average Purchase Value":"${:,.2f}",
                                  "Average Purchase Price":"${:,.2f}",
                                  "Avg Purchase Total per Person":"${:,.2f}"})
```

Out[22]:

Gender	Purchase Count	Average Purchase Price	Average Purchase Value	Avg Purchase Total per Person
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 [23]:
```
# 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+"]

purchase_data["Age Group"] = pd.cut(purchase_data["Age"],Age_bins, labels=Group_names)
purchase_data

Age_grouped = purchase_data.groupby("Age Group")

Total_count_by_age = Age_grouped["SN"].nunique()

# Calculating percentages by age group
Percentage_by_age = (Total_count_by_age/Total_players) * 100

# Creating a data frame to hold the result
Age_demographics = pd.DataFrame({"Percentage of Players": Percentage_by_age,
                                 "Total Count": Total_count_by_age})

Age_demographics.index.name = None
# Rounding the percentage column to two decimal points

Age_demographics.style.format({"Percentage of Players":"{:,.2f}"})
```

Out[23]:

	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 [24]:
```
Purchase_count_age = Age_grouped["Purchase ID"].count()

Avg_purchase_price_age = Age_grouped["Price"].mean()

Total_purchase_age = Age_grouped["Price"].sum()

Avg_purchase_total_per_person_age = Total_purchase_age/Total_count_by_age

# Creating a summary data frame to hold the result
Age_demographics = pd.DataFrame({"Purchase Count": Purchase_count_age,
                                 "Average Purchase Price": Avg_purchase_price_age,
                                 "Total Purchase Value":Total_purchase_age,
                                 "Average Purchase Total per Person": Avg_purchase_total_per_person_age})

Age_demographics.index.name = None

Age_demographics.style.format({"Average Purchase Price":"${:,.2f}","Total Purchase Value":"${:,.2f}",
                               "Average Purchase Total per Person":"${:,.2f}"})
```

Out[24]:

	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

In [25]:
```
Spender_grp = purchase_data.groupby("SN")
Purchase_count_by_spender = Spender_grp["Purchase ID"].count()

# Average purchase by name
Avg_purchase_price_by_spender = Spender_grp["Price"].mean()

# Purchase total
Total_purchase_by_spender = Spender_grp["Price"].sum()

# Creating a summary data frame
Spenders_data_frame = pd.DataFrame({"Purchase Count": Purchase_count_by_spender,
                                    "Average Purchase Price": Avg_purchase_price_by_spender,
                                "Total Purchase Value":Total_purchase_by_spender})

# Sorting the total purchase value column in descending order
Spenders_format = Spenders_data_frame.sort_values(["Total Purchase Value"], ascending=False).head()

Spenders_format.style.format({"Average Purchase Total":"${:,.2f}","Average Purchase Price":"${:,.2f}",
                                    "Total Purchase Value":"${:,.2f}"})
```

Out[25]:

SN	Purchase Count	Average Purchase Price	Total Purchase Value
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

In [26]:
```
# Retrieving the Item ID, Item Name, and Item Price columns
Item_data = purchase_data[["Item ID", "Item Name", "Price"]]

# Grouping by Item ID and Item Name
Item_grp = Item_data.groupby(["Item ID","Item Name"])

# Calculating the purchase count
Purchase_count_item = Item_grp["Price"].count()

# Calculating the total purchase value
Total_purchase_value = (Item_grp["Price"].sum())

# Calculating the items price
Item_price = Total_purchase_value/Purchase_count_item

# Creating a summary data frame to hold the results
Item_data_frame = pd.DataFrame({"Purchase Count": Purchase_count_item,  "Item Price": Item_price,
                                "Total Purchase Value": Total_purchase_value})

# Sorting the purchase count column in descending order
Most_popular_item_formatted = Item_data_frame.sort_values(["Purchase Count"], ascending=False).head()

# Formating with currency style
Most_popular_item_formatted.style.format({"Item Price":"${:,.2f}", "Total Purchase Value":"${:,.2f}"})
```

Out[26]:

		Purchase Count	Item Price	Total Purchase Value
Item ID	**Item Name**			
178	**Oathbreaker, Last Hope of the Breaking Storm**	12	$4.23	$50.76
145	**Fiery Glass Crusader**	9	$4.58	$41.22
108	**Extraction, Quickblade Of Trembling Hands**	9	$3.53	$31.77
82	**Nirvana**	9	$4.90	$44.10
19	**Pursuit, Cudgel of Necromancy**	8	$1.02	$8.16

Most Profitable Items

In [27]:
```
# Sorting the table above using the total purchase value in descending order
Most_profitable_item_formatted = Item_data_frame.sort_values(["Total Purchase Value"],
                                                             ascending=False).head()
# Format with currency style
Most_profitable_item_formatted.style.format({"Item Price":"${:,.2f}", "Total Purchase Value":"${:,.2f}"})
```

Out[27]:

		Purchase Count	Item Price	Total Purchase Value
Item ID	**Item Name**			
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
92	**Final Critic**	8	$4.88	$39.04
103	**Singed Scalpel**	8	$4.35	$34.80

In [ ]: