

Data Mining

Lab - 4

NAME: AYUSH J. MARADIA

Step 1. Import the necessary libraries

```
In [1]: import pandas as pd
import numpy as np
```

Step 2. Import the dataset from this [address](https://raw.githubusercontent.com/justmarkham/DAT8/master/data/chipotle.tsv).

Step 3. Assign it to a variable called chipo.

```
In [2]: chipo = pd.read_csv("https://raw.githubusercontent.com/justmarkham/DAT8/master/data/chipotle.tsv", sep = "\t")
chipo
```

```
Out[2]:
```

	order_id	quantity	item_name	choice_description	item_price
0	1	1	Chips and Fresh Tomato Salsa	NaN	\$2.39
1	1	1	Izze	[Clementine]	\$3.39
2	1	1	Nantucket Nectar	[Apple]	\$3.39
3	1	1	Chips and Tomatillo-Green Chili Salsa	NaN	\$2.39
4	2	2	Chicken Bowl	[Tomatillo-Red Chili Salsa (Hot), [Black Beans...	\$16.98
...
4617	1833	1	Steak Burrito	[Fresh Tomato Salsa, [Rice, Black Beans, Sour ...	\$11.75
4618	1833	1	Steak Burrito	[Fresh Tomato Salsa, [Rice, Sour Cream, Cheese...	\$11.75
4619	1834	1	Chicken Salad Bowl	[Fresh Tomato Salsa, [Fajita Vegetables, Pinto...	\$11.25
4620	1834	1	Chicken Salad Bowl	[Fresh Tomato Salsa, [Fajita Vegetables, Lettu...	\$8.75
4621	1834	1	Chicken Salad Bowl	[Fresh Tomato Salsa, [Fajita Vegetables, Pinto...	\$8.75

4622 rows × 5 columns

Step 4. See the first 10 entries

```
In [4]: chipo.head(10)
```

Out[4]:

	order_id	quantity	item_name	choice_description	item_price	revenue	
	0	1	1	Chips and Fresh Tomato Salsa	NaN	\$2.39	\$2.39
	1	1	1	Izze	[Clementine]	\$3.39	\$3.39
	2	1	1	Nantucket Nectar	[Apple]	\$3.39	\$3.39
	3	1	1	Chips and Tomatillo-Green Chili Salsa	NaN	\$2.39	\$2.39
	4	2	2	Chicken Bowl	[Tomatillo-Red Chili Salsa (Hot), [Black Beans...	\$16.98	16.9816.98
	5	3	1	Chicken Bowl	[Fresh Tomato Salsa (Mild), [Rice, Cheese, Sou...	\$10.98	\$10.98
	6	3	1	Side of Chips	NaN	\$1.69	\$1.69
	7	4	1	Steak Burrito	[Tomatillo Red Chili Salsa, [Fajita Vegetables...	\$11.75	\$11.75
	8	4	1	Steak Soft Tacos	[Tomatillo Green Chili Salsa, [Pinto Beans, Ch...	\$9.25	\$9.25
	9	5	1	Steak Burrito	[Fresh Tomato Salsa, [Rice, Black Beans, Pinto...	\$9.25	\$9.25

Step 5. What is the number of observations in the dataset?

```
In [5]: # Solution 1
chipo.shape[0]
```

Out[5]: 4622

```
In [6]: # Solution 2
chipo.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4622 entries, 0 to 4621
Data columns (total 6 columns):
#   Column                Non-Null Count  Dtype
---  -
0   order_id              4622 non-null   int64
1   quantity              4622 non-null   int64
2   item_name             4622 non-null   object
3   choice_description     3376 non-null   object
4   item_price            4622 non-null   object
5   revenue               4622 non-null   object
dtypes: int64(2), object(4)
memory usage: 216.8+ KB
```

Step 6. What is the number of columns in the dataset?

```
In [7]: chipo.shape[1]
```

Out[7]: 6

Step 7. Print the name of all the columns.

```
In [8]: chipo.columns
```

```
Out[8]: Index(['order_id', 'quantity', 'item_name', 'choice_description', 'item_price',
              'revenue'],
              dtype='object')
```

Step 8. How is the dataset indexed?

```
In [9]: chipo.index
```

Out[9]: RangeIndex(start=0, stop=4622, step=1)

Step 9. Number of Unique Items ?

```
In [10]: chipo['item_name'].nunique()
```

Out[10]: 50

Step 10. Which was the most-ordered item?

```
In [11]: c = chipo.groupby('item_name') #target item_name column
c1 = c.sum()
c2 = c1.sort_values(['quantity']).tail(1)
c2[['order_id', 'quantity']]
```

Out[11]:

	order_id	quantity
	item_name	
	Chicken Bowl	713926 761

```
In [12]: c = chipo.groupby('item_name')
c.get_group('Chicken Bowl')
```

Out[12]:

	order_id	quantity	item_name	choice_description	item_price	revenue
	4	2	2	Chicken Bowl [Tomatillo-Red Chili Salsa (Hot), [Black Beans...	\$16.98	16.9816.98
	5	3	1	Chicken Bowl [Fresh Tomato Salsa (Mild), [Rice, Cheese, Sou...	\$10.98	\$10.98
	13	7	1	Chicken Bowl [Fresh Tomato Salsa, [Fajita Vegetables, Rice,...	\$11.25	\$11.25
	19	10	1	Chicken Bowl [Tomatillo Red Chili Salsa, [Fajita Vegetables...	\$8.75	\$8.75
	26	13	1	Chicken Bowl [Roasted Chili Corn Salsa (Medium), [Pinto Bea...	\$8.49	\$8.49

	4590	1825	1	Chicken Bowl [Roasted Chili Corn Salsa, [Rice, Black Beans,...	\$11.25	\$11.25
	4591	1825	1	Chicken Bowl [Tomatillo Red Chili Salsa, [Rice, Black Beans...	\$8.75	\$8.75
	4595	1826	1	Chicken Bowl [Tomatillo Green Chili Salsa, [Rice, Black Bea...	\$8.75	\$8.75
	4599	1827	1	Chicken Bowl [Roasted Chili Corn Salsa, [Cheese, Lettuce]]	\$8.75	\$8.75
	4604	1828	1	Chicken Bowl [Fresh Tomato Salsa, [Rice, Black Beans, Chees...	\$8.75	\$8.75

726 rows × 6 columns

Step 11. How many items were orderd in total?

```
In [13]: chipo['quantity'].sum()

Out[13]: 4972
```

Step 12. Turn the item price into a float

Step 12.a. Check the item price type

```
In [14]: chipo['item_price'].dtype # o = object

Out[14]: dtype('O')
```

Step 12.b. Create a lambda function and change the type of item price

```
In [15]: chipo['item_price'] = chipo['item_price'].apply(lambda x : float(x.replace('$', " ")))
```

Step 12.c. Check the item price type

```
In [16]: chipo['item_price'].dtype

Out[16]: dtype('float64')
```

Step 14. How much was the revenue for the period in the dataset?

```
In [17]: c = (chipo['item_price'] * chipo['quantity']).sum()
print(f"Total revenue: ${c}")

Total revenue: $39237.02
```

Step 15. How many orders were made ?

```
In [18]: chipo['order_id'].nunique()

Out[18]: 1834
```

Step 17. How many different choice descriptions are there?

```
In [19]: chipo['choice_description'].nunique()

Out[19]: 1043
```

Step 18. What items have been ordered more than 100 times?

```
In [20]: a = chipo.groupby('item_name')['quantity'].sum()  
a [a > 100]
```

```
Out[20]: item_name  
Bottled Water          211  
Canned Soda            126  
Canned Soft Drink     351  
Chicken Bowl          761  
Chicken Burrito       591  
Chicken Salad Bowl    123  
Chicken Soft Tacos    120  
Chips                  230  
Chips and Fresh Tomato Salsa 130  
Chips and Guacamole   506  
Side of Chips         110  
Steak Bowl            221  
Steak Burrito         386  
Name: quantity, dtype: int64
```

Step 19. What is the average revenue amount per order?

```
In [21]: # Solution 1  
chipo['revenue'] = chipo['quantity'] * chipo['item_price']  
order_grouped = chipo.groupby(by=['order_id']).sum()  
order_grouped['revenue'].mean()
```

```
Out[21]: 21.39423118865867
```

```
In [22]: # Solution 2  
chipo.groupby(by=['order_id']).sum()['revenue'].mean()
```

```
Out[22]: 21.39423118865867
```