

## ✓ Ex2 - Getting and Knowing your Data

This time we are going to pull data directly from the internet. Special thanks to: <https://github.com/justmarkham> for sharing the dataset and materials.

### ✓ Step 1. Import the necessary libraries

```
import pandas as pd
```

### ✓ Step 2. Import the dataset from this [address](#).


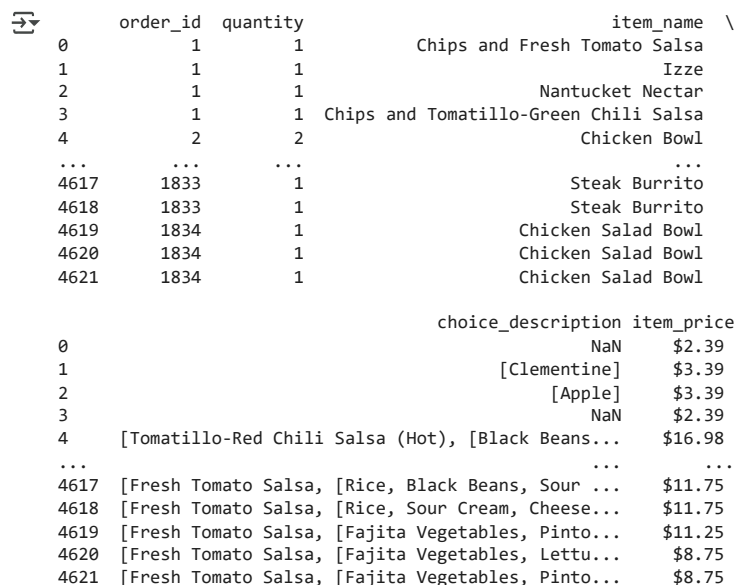
```
url = "https://raw.githubusercontent.com/thieu1995/csv-files/main/data/pandas/chipotle.tsv"
```

 [Show hidden output](#)

### ✓ Step 3. Assign it to a variable called chipo.

```
chipotle = pd.read_csv(url, sep='\t')
```

```
print(chipotle)
```

	order_id	quantity	item_name \
0	1	1	Chips and Fresh Tomato Salsa
1	1	1	Izze
2	1	1	Nantucket Nectar
3	1	1	Chips and Tomatillo-Green Chili Salsa
4	2	2	Chicken Bowl
...	...	...	...
4617	1833	1	Steak Burrito
4618	1833	1	Steak Burrito
4619	1834	1	Chicken Salad Bowl
4620	1834	1	Chicken Salad Bowl
4621	1834	1	Chicken Salad Bowl


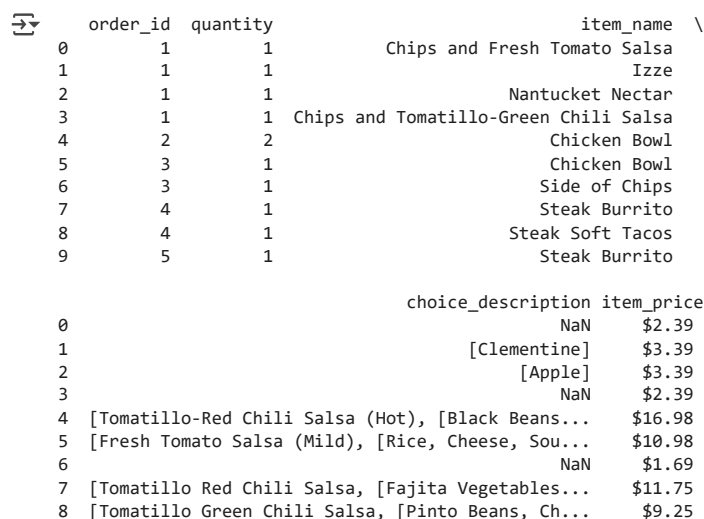
  

	choice_description	item_price
0	NaN	\$2.39
1	[Clementine]	\$3.39
2	[Apple]	\$3.39
3	NaN	\$2.39
4	[Tomatillo-Red Chili Salsa (Hot), [Black Beans...	\$16.98
...	...	...
4617	[Fresh Tomato Salsa, [Rice, Black Beans, Sour ...	\$11.75
4618	[Fresh Tomato Salsa, [Rice, Sour Cream, Cheese...	\$11.75
4619	[Fresh Tomato Salsa, [Fajita Vegetables, Pinto...	\$11.25
4620	[Fresh Tomato Salsa, [Fajita Vegetables, Lettu...	\$8.75
4621	[Fresh Tomato Salsa, [Fajita Vegetables, Pinto...	\$8.75

[4622 rows x 5 columns]

### ✓ Step 4. See the first 10 entries

```
print(chipotle.head(10))
```

	order_id	quantity	item_name \
0	1	1	Chips and Fresh Tomato Salsa
1	1	1	Izze
2	1	1	Nantucket Nectar
3	1	1	Chips and Tomatillo-Green Chili Salsa
4	2	2	Chicken Bowl
5	3	1	Chicken Bowl
6	3	1	Side of Chips
7	4	1	Steak Burrito
8	4	1	Steak Soft Tacos
9	5	1	Steak Burrito

	choice_description	item_price
0	NaN	\$2.39
1	[Clementine]	\$3.39
2	[Apple]	\$3.39
3	NaN	\$2.39
4	[Tomatillo-Red Chili Salsa (Hot), [Black Beans...	\$16.98
5	[Fresh Tomato Salsa (Mild), [Rice, Cheese, Sou...	\$10.98
6	NaN	\$1.69
7	[Tomatillo Red Chili Salsa, [Fajita Vegetables...	\$11.75
8	[Tomatillo Green Chili Salsa, [Pinto Beans, Ch...	\$9.25

9 [Fresh Tomato Salsa, [Rice, Black Beans, Pinto... \$9.25

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

# Solution all

```
print(chipotle.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4622 entries, 0 to 4621
Data columns (total 5 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
dtypes: int64(2), object(3)
memory usage: 180.7+ KB
None
```

# Solution 1

```
print(chipotle.shape[0])
```

```
4622
```

# Solution 2

```
print(len(chipotle))
```

```
4622
```

# Solution 3

```
print(chipotle['order_id'].count())
```

```
4622
```

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

# Solution 1

```
print(chipotle.shape[1])
```

```
5
```

# Solution 2

```
print(len(chipotle.columns))
```

```
5
```

### ▼ Step 7. Print the name of all the columns.

```
print(chipotle.columns)
```

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

### ▼ Step 8. How is the dataset indexed?

```
chipotle.index
```

```
RangeIndex(start=0, stop=4622, step=1)
```


### ▼ Step 9. Which was the most-ordered item?

```
most_ordered = chipotle.groupby('item_name')['quantity'].sum()
```

```
most_ordered = most_ordered.sort_values()
```


```
most_ordered = most_ordered.index[-1]
```

```
print(most_ordered)
```

 Chicken Bowl

```
most_ordered_quantity = chipotle.groupby('item_name')['quantity'].sum()

print(most_ordered_quantity.idxmax())
```

 Chicken Bowl

▼ Step 10. For the most-ordered item, how many items were ordered?

```
most_ordered_quantity = chipotle.groupby('item_name')['quantity'].sum()

print(most_ordered_quantity.max())
```

 761

```
most_ordered = chipotle.groupby('item_name')['quantity'].sum()

most_ordered = most_ordered.sort_values()

most_ordered = most_ordered.iloc[-1]


print(most_ordered)
```

 761

▼ Step 11. What was the most ordered item in the choice\_description column?

```
most_ordered_choice = chipotle.groupby('choice_description')['quantity'].sum()

print(most_ordered_choice.idxmax())
```

 [Diet Coke]

▼ Step 12. How many items were orderd in total?


```
print(most_ordered_choice.max())
```

 159

▼ Step 13. Turn the item price into a float

▼ Step 13.a. Check the item price type

```
print(chipotle['item_price'].dtype)
```


 object

▼ Step 13.b. Create a lambda function and change the type of item price

```
chipotle['item_price'] = chipotle['item_price'].apply(lambda x: float(x.replace('$', '')))
```

▼ Step 13.c. Check the item price type

```
print(chipotle['item_price'].dtype)
```

 float64

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

```
sum_revenue = (chipotle['item_price'] * chipotle['quantity']).sum()

print(sum_revenue)
```

 39237.02

## ✓ Step 15. How many orders were made in the period?

```
sum_quantity = chipotle['quantity'].sum()

print(sum_quantity)

↗ 4972
```

## ✓ Step 16. What is the average revenue amount per order?

```
# Solution 1
average_revenue = sum_revenue / len(chipotle)

print(average_revenue)

↗ 8.48918649935093

# Solution 2
chipotle['total_price'] = chipotle['item_price'] * chipotle['quantity']

average_revenue = chipotle['total_price'].mean()

print(average_revenue)

↗ 8.48918649935093
```

## ✓ Step 17. How many different items are sold?

```
different_items = chipotle['item_name'].nunique()

print(different_items)

↗ 50
```