

# Unit Assessment: Python

---

**Due** Wednesday by 11:59pm

**Points** 100


**Submitting** an external tool

---

You are about to complete your first Unit Assessment! This Unit Assessment allows you to check your knowledge, as well as demonstrate your competency in key concepts from Modules 3 through 6.


After submitting the assessment, you will see a summary of your performance. While you will not be able to see your performance on individual questions, you are allowed unlimited attempts to complete the assessment.

Some of the questions on this assessment require specific resources. Download the following resources before you get started.

[cereal.csv](https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/cereal.csv)  [\(https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/cereal.csv\)](https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/cereal.csv)

[donors2008.csv](https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/donors2008.csv)  [\(https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/donors2008.csv\)](https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/donors2008.csv)

[avg\\_rain\\_state.csv](https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/avg_rain_state.csv)  ([https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/avg\\_rain\\_state.csv](https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/avg_rain_state.csv))

[youtube\\_response.json](https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/youtube_response.json)  ([https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/youtube\\_response.json](https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/Unit-Assessment-Resources/M6-UA-Python/youtube_response.json))

## ← Attempt 1

### Date Started

October 01, 2022 9:11 am

### Date Completed

October 01, 2022 11:55 am

### Time Spent

2h 43m

### Score

86%

Complete the following code below so that it will do the following tasks:

1. Open and read the *cereal.csv* file.
2. Skip the header row.
3. Read through each row.
4. Print out the cereals the following cereals that contain 5 or more grams of fiber in this order:

100% Bran  
All-Bran  
All-Bran with Extra Fiber  
Bran Flakes  
Fruit & Fibre Dates; Walnuts; and Oats  
Fruitful Bran  
Post Nat. Raisin Bran  
Raisin Bran

```
import os
import csv
cereal_csv = os.path.join(".", "cereal.csv")

with open(cereal_csv, newline="") as csvfile:
```

```

with open('cereal.csv', newline='') as csvfile:

    csvreader = csv.reader(csvfile, delimiter=',')

    csv_header = next(csvfile)

    for row in csvreader:

        if float(row[7]) >= 5:

            print(row[0])

```

:: csv

:: cereal.csv

:: row[6]

:: row[1]

Attempt #1: 5/5 (Score: 5/5)

You are trying to find which classroom “Dan” is in, i.e., `name = "Dan"`. You have access to a list of students first names only from three separate classrooms, `classroom_1`, `classroom_2`, and `classroom_3`. Dan is only in one of the classrooms.

What is the order of the following code fragments that would check all three classrooms and print out which classroom “Dan” is in using an `if-elif-else` statement.

1. `if name in classroom_1:` :
2. `print(f"{name} is in the classroom_1.")`
3. `elif name in classroom_2:` :
4. `print(f"{name} is in classroom_2.")`
5. `else:` :

6. `print(f"{name} is in classroom_3.")`

❖ `if name not in classroom_1:`

❖ `if name not in classroom_2:`

❖ `else name in classroom_2:`

❖ `elif name in classroom_3:`

Attempt #1: 5/5 (Score: 5/5)

1. Download the *donors2008.csv*.
2. Open and read the *donors2008.csv* file into a pandas DataFrame.
3. Using groupby, create a Series with the average "Amount" for each state.
4. Convert the Series to a DataFrame if needed.
5. How many states have averages equal to or greater than \$750.00?

- ☒ 9
- ☐ 8
- ☐ 10
- ☐ 11

Attempt #1: 5/5 (Score: 5/5)

The following code has a syntax error. Assuming you have a folder named "output" and a file named "employees.txt". Click on the syntax error.

```
import os
import csv

file_to_save = os.path.join("output", "employees.txt")

with open(file_to_save, "w") as new_file:
    employees = (
        f"First Name: Last Name: Salary\n"
```

```
T First Name , Last Name , SSN\nf"Caleb', 'Frost', '505-80-2901\n")\n\ntxt_file. write(employees)
```

Attempt #1: 5/5 (Score: 5/5)

Assuming you have a folder named “output” and a file named “employees.txt”. What is the correct code that needs to be used to write the data to the file?

```
import os
import csv

file_to_save = os.path.join("output", "employees.txt")

with open(file_to_save, "w") as new_file:
    employees = (
        f"First Name', 'Last Name', 'SSN\n"
        f"Caleb', 'Frost', '505-80-2901\n")

    new_file.write(employees)
```

❖ file

❖ employees

❖ employees.txt

Attempt #1: 5/5 (Score: 5/5)

What Python method is used to get all the keys from a dictionary?

☐ get()

☒ keys()

☐ `get_keys()`

☐ `key()`

Attempt #1: 5/5 (Score: 5/5)

What is the output when the following code is run?

```
students = ("Frank", "Mary", "Jasmine", "Ivana", "Ahmed")
students.append("Serena")
print(students)
```

- ☐ The code won't compile.
- ☐ `['Frank', 'Mary', 'Jasmine', 'Ivana', 'Ahmed', 'Serena']`
- ☒ `AttributeError: 'tuple' object has no attribute 'append'`
- ☐ `('Frank', 'Mary', 'Jasmine', 'Ivana', 'Ahmed', 'Serena')`
- ☐ `('Frank', 'Mary', 'Jasmine', 'Ivana', 'Ahmed')`

Attempt #1: 5/5 (Score: 5/5)

Using pandas, how would you export a DataFrame, "df" as the file, "data\_file.csv" into a folder, "Output" without the index, but with the header? (Select all that apply)

- ☒ `df.to_csv("Output/data_file.csv", index=False)`
- ☐ `df.to_csv("Output/data_file.csv")`
- ☐ `df.to_csv("Output/data_file.csv", header=True)`
- ☐ `df.to_csv("Output/data_file.csv", index=False, header=False)`
- ☐ `df.to_csv("Output/data_file.csv", header=False)`

✓ `df.to_csv("Output/data_file.csv", index=False, header=True)`

Attempt #1: 5/5 (Score: 5/5)

Download `avg_rain_state.csv` and read it into a pandas DataFrame as `rain_df`.

What is the code that will return the following output.

	State	Inches	Millimetres	Rank
8	Florida	54.5	1385	5
0	Alabama	58.3	1480	4
23	Mississippi	59.0	1499	3
17	Louisiana	60.1	1528	2
10	Hawaii	63.7	1618	1

```
rain_df.sort_values(['Rank'], ascending=False).tail()
```

⚙️ `sort=True`

⚙️ `groupby`

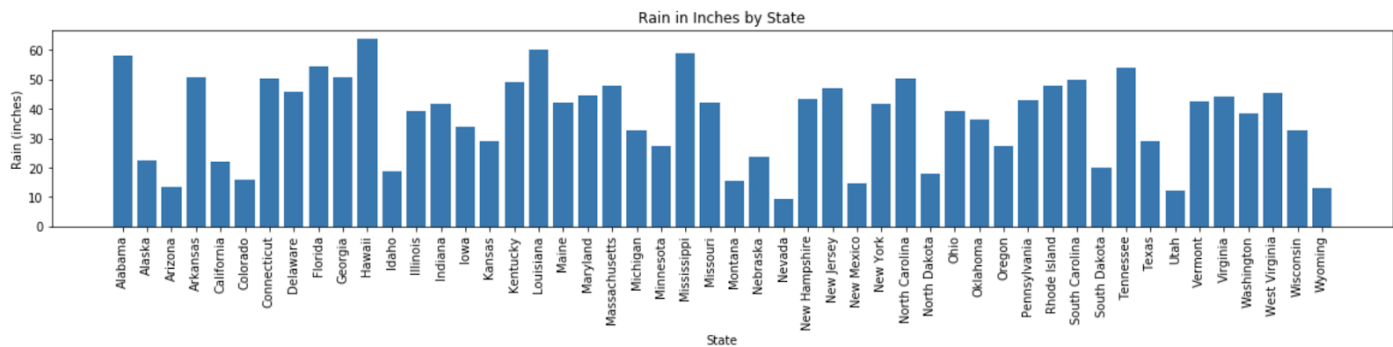
⚙️ `head()`

⚙️ `sort`

⚙️ `ascending=True`

Attempt #1: 5/5 (Score: 5/5)

7. Using the `matplotlib` module, complete the code to create the following graph.



```
# Dependencies
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Create graph

tick_locations = [value for value in df["State"] ]

plt.figure(figsize=(20,3))

plt.bar( df["State"] , df["Inches"] )

plt.xticks(tick_locations, df["State"] , rotation=

'vertical' )

plt.title("Rain in Inches by State")
plt.xlabel("State")
plt.ylabel("Rain (inches)")
plt.show()
```

⚡ 'horizontal'

⚡ df["Inches"]

⚡ x\_axis

⚡ len(df["State"])

⚡ 'vertical'

⚡ df["State"]

⚡ len(df)

Attempt #1: 5/5 (Score: 5/5)

Open and read the file, `avg_rain_state.csv` into a Pandas DataFrame and answer the following



## questions

1. What is the mean rainfall for the entire DataFrame? 37.078 ,
2. What is the median rainfall for the entire DataFrame? 41.75
3. How many modes are there for rainfall? 3

⚙ 47.15

⚙ 5

Attempt #1: 5/5 (Score: 5/5)

From the following information that creates a pie chart. Drag the best answer for each of the following questions.

```
pies = ["Apple", "Pumpkin", "Chocolate Creme", "Cherry", "Apple Crumb", "Pecan", "Lemon Meringue", "Blueberry", "Key Lime", "Peach"]
pie_votes = [47,37,32,27,25,24,24,21,18,16]
colors = ["yellow","green","lightblue","orange","red","purple","pink","yellowgreen","lightskyblue","lightcoral"]
offset = (0,0,0,0.2,0,0,0,0,0,0)
```

**What type of pie will be offset in the pie chart?**

**What is the color of the pie wedge that is offset?**

**What is the percentage of votes for the pie wedge that is offset?**

Cherry

orange

10.0

### Pies

⚙ Apple

⚙ Chocolate Creme

⚙ Pumpkin

⚡ Peach

**Colors**

⚡ lightblue

⚡ yellow

⚡ green

⚡ lightcoral

**Percentage**

⚡ 13.7

⚡ 17.3

⚡ 5.9

⚡ 11.8

Attempt #1: 5/5 (Score: 5/5)

For the following data.

```
test_grades = {
    'Class': ['Oct', 'Oct', 'Jan', 'Jan', 'Oct', 'Jan'],
    'Name': ["Cyndy", "Logan", "Laci", "Elmer", "Crystle", "Emmie"],
    'Test Score': [90, 59, 72, 88, 98, 60]}
```

1. Create a DataFrame from the “test\_grades” dictionary.
2. Create bins, 0, 59, 69, 79, 89, 100, to hold the data.
3. Assign a letter grade, as “letter\_grades”, for the four bins as the labels.

Complete the code that will create a new column “Letter Grade” based on the four bins and labels

```
df ['Letter Grade'] = pd.cut(df ['Test Score'], bins, labels= letter_grades )
```

⚡ 'Test Score'

⚡ [letter\_grades]

⚡ sort

Attempt #1: 5/5 (Score: 5/5)

Using the following code.

```
import requests  
url = "https://api.spacexdata.com/v2/launchpads"
```

What is the length of the JSON file after you make the request?

☐ 3

☐ 6

☒ TypeError: object of type 'response' has no len()

☐ 1

Attempt #1: 0/5 (Score: 0/5)

Using the following code.

```
import request  
url = "https://api.spacexdata.com/v2/launchpads"
```

Using a for loop, what is the order of the "full name" of each launch site in the JSON file when printed to the output cell?

≡ Vandenberg Air Force Base Space Launch Complex 3W

≡ Cape Canaveral Air Force Station Space Launch Complex 40

≡ Kwajalein Atoll Omelek Island

≡ Vandenberg Air Force Base Space Launch Complex 4E

≡ Kennedy Space Center Historic Launch Complex 39A

— Kennedy Space Center Historic Education Complex 337

≡ SpaceX South Texas Launch Site

Attempt #1: 1/5 (Score: 1/5)

Add the correct exception error so that when the code is run you get the last line to print to the terminal.

```
try:
    print("Infinity looks like + " + str(10 / 0) + ".")
except ZeroDivisionError :

    try:
        print("I think her name was + " + name + "?")
    except NameError :

        try:
            print("Your name is a nonsense number. Look: " + int("Gabriel"))
        except ValueError :

            print("I made it through the gauntlet. The message survived!")
```

⚡ SyntaxError

⚡ TypeError

Attempt #1: 5/5 (Score: 5/5)

Download youtube\_response.json and use the following code to open and read the JSON file. Then answer the following question.

```
import json
import os

filepath = os.path.join("youtube_response.json")
```

```
with open(filepath) as jsonfile:  
    video_json = json.load(jsonfile)
```

Complete the code that returns the link to the video's default thumbnail?

```
video_json[    'items'    ][    0    ][    'thumbnail'    ][  
    1    ][    'hqDefault'    ]
```

❏ 'apiVersion'

❏ 'data'

❏ 'default'

❏ 'player'

❏ 'id'

Attempt #1: 0/5 (Score: 0/5)

You have converted a CSV file to a DataFrame, "df". How would you get the names of the columns from the DataFrame?

- ☐ df.headers()
- ☐ df.head()
- ☒ df.columns
- ☐ df.columns()

Attempt #1: 5/5 (Score: 5/5)

What piece of code defines the beginning of a Python function?