Python Lab 7.1: Read Data from CSV Files and a Produce a Data Frame

> The purpose of this practice is to help you apply the concepts discussed up to **now**:
>
> - read and parse data from different file types
> - use Pandas to create a DataFrame
> - perform basic data frame manipulation

In this practice we will use a csv file which contains nutritial information in cereals. The columns in the file are the following:

- Name: Name of cereal
- mfr: Manufacturer of cereal
  - A = American Home Food Products;
  - G = General Mills
  - K = Kelloggs
  - N = Nabisco
  - P = Post
  - Q = Quaker Oats
  - R = Ralston Purina
- type:
  - cold
  - hot
- calories: calories per serving
- protein: grams of protein
- fat: grams of fat
- sodium: milligrams of sodium
- fiber: grams of dietary fiber
- carbo: grams of complex carbohydrates
- sugars: grams of sugars
- potass: milligrams of potassium
- vitamins: vitamins and minerals - 0, 25, or 100, indicating the typical percentage of FDA recommended
- shelf: display shelf (1, 2, or 3, counting from the floor)
- weight: weight in ounces of one serving
- cups: number of cups in one serving
- rating: a rating of the cereals (Possibly from Consumer Reports?

To Do:

Open `lab7_1.py` and `cereal.csv` in the text editor. Have a look at the csv file and then:

1. Load this csv into a data frame for further processing. Remember to import the required libraries on top of your program.

2. Print the first 5 rows from this data frame

▶ HINT 1: Load csv to a data frame and print

```
cereal_df = pd.read_csv("cereal.csv"))
print(cereal_df.head(5))
```

3. Check the data types of calories, fiber and fat

▶ HINT 2: Find out datatypes

```
print(cereal_df['calories'].dtypes)
print(cereal_df['fiber'].dtypes)
print(cereal_df['fat'].dtypes)
```

4. Notice that the datatype of fat is object and not int or float. This means that some missing or invalid values caused python not treat this column as numeric. In Line 5 in the fat column it says "not provided". We can specify during the data load to treat this value as NaN and thus our datatype will not be affected.

```
cereal_df = pd.read_csv("cereal.csv", na_values = "not provided")
```

Rerun you code now to see the data fix.

5. Create a calculated column called "low/high" to display "high fat" or "low fat" depending on a condition: less than 5% is low fat, 5% and more is high fat.Display again the first rows to verify your results.

▶ HINT 3: Define a calculated column

```
cereal_df['low/high']= np.where(cereal_df['fat']<5, 'low', 'high')
print(cereal_df.head(5))
```

Execute your program

Remember in order to execute your code you type in the terminal:

```
python lab7_1.py
```

{% next %}

Check Your Code

Execute the below to evaluate the correctness of your code using `check50`, but be sure to test it yourself also.

```
check50 mkotsovoulou/ods6001a/main/labs/lab7_1
```

Execute the below to evaluate the style of your code using `style50`.

```
style50 lab7_1.py
```

{% next %}

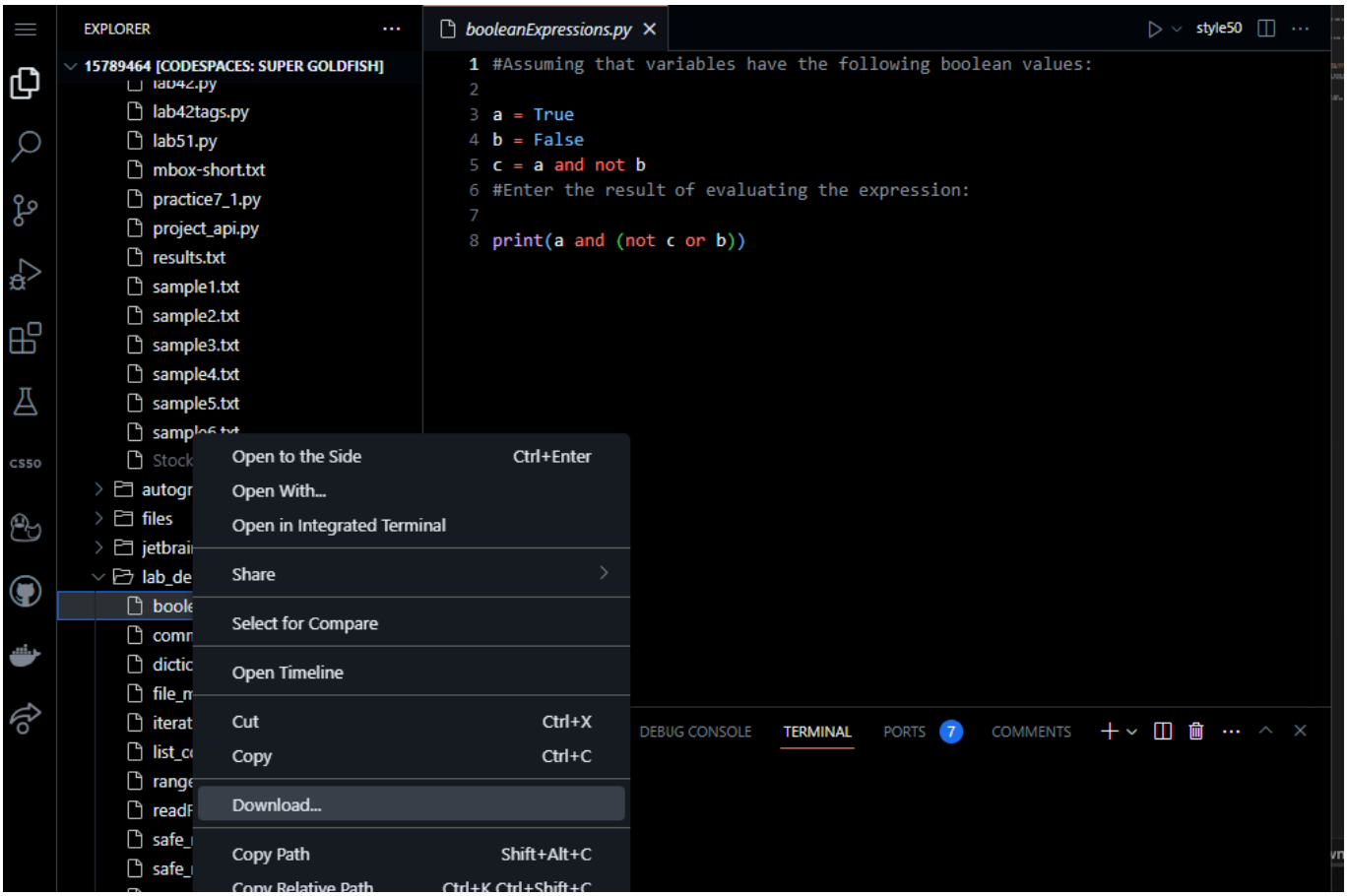
Submit your code

Execute the command below, logging in with your `GitHub username` and `Personal Access Token` when prompted. For security, you'll see asterisks (`*`) instead of the actual characters in your token.

If you do not have generated a Personal Access ToKen follow the instructions: https://docs.github.com/en/authentication/keeping-your-account-and-data-secure/creating-a-personal-access-token

```
submit50 mkotsovoulou/ods6001a/main/labs/lab7_1
```

You can re-submit your solution as many times as you want. When you are happy with your solution, download the code and upload it to Canvas.


EXPLORER
booleanExpressions.py
style50
15789464 [CODESPACES: SUPER GOLDFISH]
lab42tags.py
lab51.py
mbox-short.txt
practice7_1.py
project_api.py
results.txt
sample1.txt
sample2.txt
sample3.txt
sample4.txt
sample5.txt
files
jetbrai
lab_de
comn
dictic
file_n
iterat
list_c
range
readF
safe_
safe_
1 #Assuming that variables have the following boolean values:
2
3 a = True
4 b = False
5 c = a and not b
6 #Enter the result of evaluating the expression:
7
8 print(a and (not c or b))
Open to the Side Ctrl+Enter
Open With...
Open in Integrated Terminal
Share
Select for Compare
Open Timeline
Cut Ctrl+X
Copy Ctrl+C
Download...
Copy Path Shift+Alt+C
DEBUG CONSOLE
TERMINAL
PORTS 7
COMMENTS

Done!