Programming Problem 4: File Manipulation and String Functions

> The purpose of this assignment is to test your understanding and application of the concepts discussed up to **Week 3**:
>
> - read data from files
> - parse and extract information from strings

Specifications

Create a program which will read a text file containing some text and determine if the text inside a file contains a single character, a single word, multiple words, a sentence, a paragraph or a full document.

- if the document is empty contains no characters, return empty.

- if the document contains a single character, return character.

- if the document contains a single word (no spaces and periods) return word.

- if the document contains multiple words (more than one space) and a period return sentence.

- if the document contains multiple sentences (more than one period) its a paragraph.

- Every paragraphs ends with a new line character ("\n") if the text contains more than one new line character is a document.

- ask the user to provide the filename

- display 'File not found' if the user types an invalid filename

Remember to strip all spaces before and after your string before proceeding. You can replace all instances of two spaces with one space.

Hint 1 :

use a loop to keep replacing as long as 2 consequtive spaces exist in the string.

Hint 2 :

use the str.count(" ") is find out how many spaces exist. and string.count(".") to count how many periods.

The order by which you will test the conditions in your if statements plays an important role.

Test your program with different files provided in your folder.

Some Technical Details: A Basic Python main()

In some Python scripts, you may see a function definition and a conditional statement that looks like the example below:

```
def main():
    print("Hello World!")

if __name__ == "__main__":
    main()
```

In this code, there is a function called main() that prints the phrase Hello World! when the Python interpreter executes it. There is also a conditional (or if) statement that checks the value of **name** and compares it to the string "**main**". When the if statement evaluates to True, the Python interpreter executes main().

In this program write your code to accept user input in def main():

{% next %}

Execute and Test your program

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

```
python assignment4.py
```

sample1.txt should return empty

sample2.txt should return character

sample3.txt should return words

sample4.txt should return sentence

sample5.txt should return paragraph

sample6.txt should return document

{% next %}

Check Your Code

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

```
check50 mkotsovoulou/ods6001a/main/assignments/assignment4
```

Execute the below to evaluate the style of your code using `style50`.
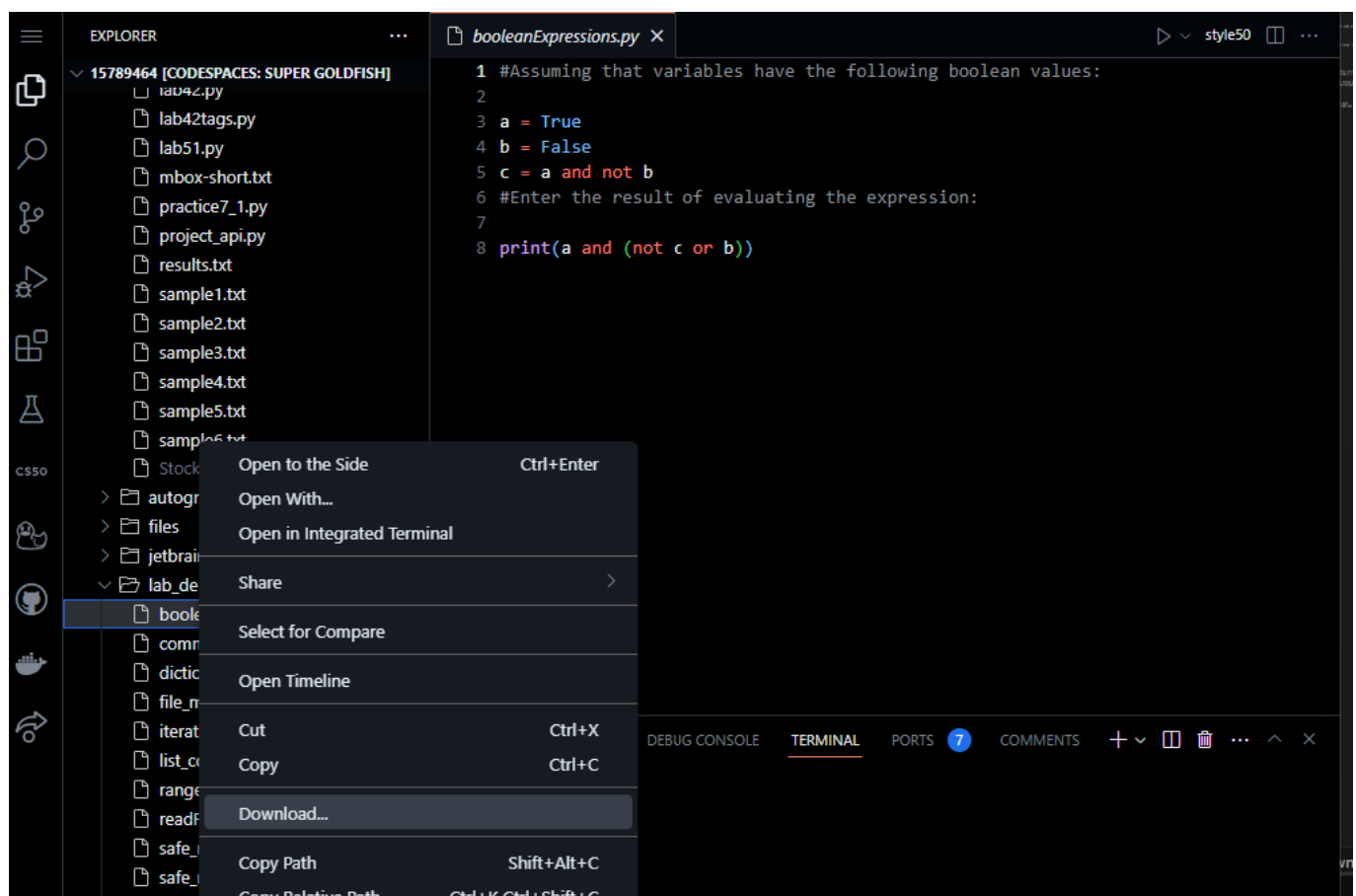
```
style50 assignment4.py
```

{% 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.

```
submit50 mkotsovoulou/ods6001a/main/assignments/assignment4
```

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.

Done!

🎉