

Python Lab 6.1: OOP

The purpose of this practice is to help you apply the concepts discussed up to **now**:

- implement object classes and develop program to use them
- test, debug and predict program behaviour

In `lab6_1.py` in the text editor at top-right, The exercise template contains a pre-defined `ShoppingList` class, which can be used to model a shopping list. Your task is to add a method to the class definition. You do not need to change any methods already defined.

Specifications

Assuming we have a `ShoppingList` object referenced in a variable named `my_list`, the object can be handled with method `add_item` to add in the shopping list items and their quantities. You can use the following code to test the `ShoppingList` object functionality:

```
if __name__ == "__main__":
    my_list = ShoppingList()
    my_list.add_item("bananas", 10)
    my_list.add_item("apples", 5)
    my_list.add_item("pineapple", 1)
```

To Do - 1

In the main program (not in the class) complete the code of a function named `total_units(my_list: ShoppingList)`, which takes a `ShoppingList` type object as its argument.

The function should calculate the total number of units in the `ShoppingList`, and return this value.

The final print statement can be used to test the functionality of your method.

```
print(total_units(my_list))
```

► HINT 1: how to solve the function to calculate the total units

```
def total_units(my_list):
    quantity = 0
    for i in range(1, my_list.number_of_items()+1):
        quantity = quantity + my_list.qty(i)
    return quantity
```

To Do - 2

Move the code which prints all the items and their quantities of the shopping list in the `print_shopping_list` method. Remove the `pass` statement! Call the method

```
def print_shopping_list(my_list):  
    pass
```

{% next %}

Execute and Test your program

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

```
python lab6_1.py
```

Check that your code produces correct results (30 total units).

```
bananas: 22 units  
apples: 7 units  
pineapple: 1 units  
30
```

{% next %}

To Do - 3

Before submitting, comment out the printing of the shopping list items and just output the total number of items

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/lab6_1
```

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

```
style50 lab6_1.py
```

{% next %}

Submit your code

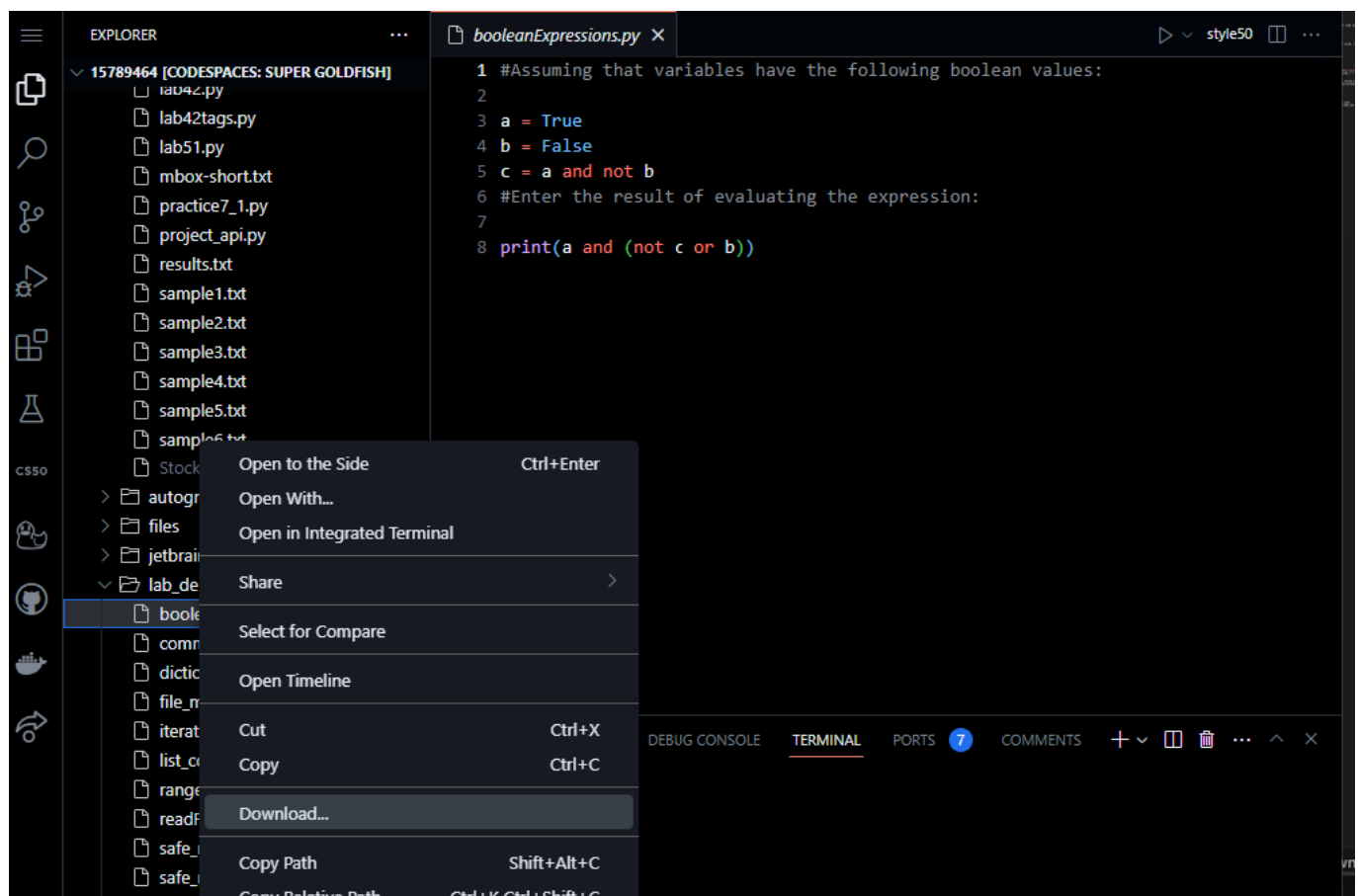
Execute the command below, logging in with your **GitHub username** and **Personal Access Token** when prompted. For security, you'll see asterisks (*) instead of the actual characters in your token.

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/lab6_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.



Done!

