

Python Lists One Worksheet

Instructions:

1. Open the `VScodeIntro` folder from the previous assignment on your desktop.
 2. Double click on the `vscode-intro.code-workspace` file.
 3. Create a new file named `lists_one.py`
 4. Read the directions for each task very carefully before you begin working on it.
 5. Every time the directions say something like "take a screenshot", do it and then copy it into the proper space in this document.
-

This worksheet is intended to show you how lists work in Python and how they compare with strings. Work through each task with the help of a partner (being sure to type/run each bit of code yourself).

Problem 0: Identify Yourself

1. What are the names of the people in your group?

>

2. Share this document with **all people** in the group and make them editors.

☐ Check the box when you're done

Problem 1: Grocery Lists

Use the code below to answer the following questions and complete the required tasks.

...

```
def do_something_0(to_add):  
    gl = ["bread", "apples"]  
    gl.append(to_add)  
  
    for elem in gl:
```

```
    print(f"Buy {elem}!")

    print(f"But the {gl[-1]} is most important!")

if __name__ == "__main__":
    do_something_0("ice cream")
...
```

BEFORE running the program, answer the following questions:

1. How many inputs to the `do_something_0()` function are there? What is the type (e.g., list, string, or int) of each input?

>

2. This program above uses a list. How do you know that there is a list?

>

3. Strings and lists are similar. What are **two ways** that the program above shows their similarity?

>

4. What do you think will be printed to the screen when you run the code?

>

NEXT, copy the code into VScode and run it. Look in the **terminal section** for the results.

1. Were your predictions correct? If not, what mistakes did you make?

>

2. Modify **the function definition** so that it takes **two string inputs** representing things to be added to the grocery list. Then, modify **the body of the function** so that it adds these items to the list.

Hint: Add a second `gl.append()` call

3. Modify **the body of the function** so that instead of saying the last item is the most important, it says the first item is the most important.
4. Copy **your code in VSCode** and paste it here.

...

...

Problem 2: Debugger

Use the code below to answer the following questions and complete the required tasks.

...

```
def do_something_1(items):
    ret = []

    for i in range(len(items)):
        item = items[i]
        if "bug" not in item:
            ret.append(item)

    return ret

if __name__ == "__main__":
    basket = ["apple", "banana", "ladybug", "corn", "stink bug",
"bread"]
    result = do_something_1(basket)
    print(result)
...
```

BEFORE running the program, answer the following questions:

1. How many inputs to the `do_something_1()` function are there? What is the type (e.g., list, string, or int) of each input?

>

2. This program above uses two separate lists. Which two lines of code **create** the lists?

>

3. Strings and lists are similar. What are **two ways** that the program above shows their similarity?

>

4. What do you think will be printed to the screen when you run the code?

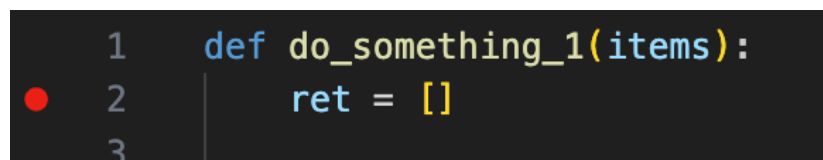
>

NEXT, copy the code into VSCode and run it. Look in the **terminal section** for the results.

1. Were your predictions correct? If not, what mistakes did you make?

>

2. Create a breakpoint on the first line of the `do_something_1()` function as seen below.

A screenshot of a code editor with a dark background. It shows a Python function definition:

```
1 def do_something_1(items):  
2     ret = []  
3
```

 A red circular breakpoint is placed on the left margin, aligned with line 2. A vertical line extends from the breakpoint to the code line.

- a. Run the debugger through the program until the **ret list has three items** in it.
- b. Take a screenshot of the debugger's **VARIABLES** section and paste it into the space below.

Note: make sure that **both lists are shown** in the screenshot

...

...

- c. Take a screenshot of the debugger's `CALL STACK` section and paste it into the space below.
- ...
- ...
- d. Based on the debugger's `VARIABLES` section, what is the value of `i` when `ret` has three items in it?
- >
3. Based on what you saw the programming doing while debugging, what would be a better name for the function than `do_something_1()`?
- >
4. Modify **the body of the function** so that it only adds items that have five letters in it to `ret`.
5. Modify **the body of the function** so that right before it returns, it appends the word `spider` onto the end of the list.
6. Copy **your code in VSCode** and paste it here.

...

...

Problem 3: Reflection

1. How is “iterating through a list” similar to “iterating through a string”?

>

2. How is “iterating through a list” different from “iterating through a string”?

>

3. In Python, a list is often referred to as a “container”. In 2-3 sentences, explain what this sentence means.

>

4. When would you choose to use a list instead of a variable in your program?

>

5. Did you notice any obvious errors with this worksheet that should be fixed for future students? If so, what were they?

>

6. Were there things in the worksheet that were overly confusing and that you think could be improved? If so, what were they?

>

Problem 4 (10pt Bonus): The Sum of All Integers

Now it's your turn. Create a function that takes a list of integers as an input. Then, using the accumulator pattern, have the function return the sum of all of the numbers in the list. Once you are done, copy **your code in VSCode** and paste it here.

...

...