CS 65
Fall 2025

Lab 6: Functions

Total points: 4

In this lab, you are going to practice defining your own functions, including those with parameters and return values.

This lab is a little unique because I will be specifying the name and parameter list of thee specific functions you must implement. You can add your own code to test each function (and you are encouraged to do so), but all that matters is the function implementation. The automated tests will test if you have implemented each function correctly. A way of testing software, in which you test only the function that is being developed, is called a *unit test*.

First, create a new file in Thonny and save it. Name the file Lab6.py.

Part 1:
Create a function named **`introductions`**. The function should accept one parameter, a string, called `name`. The function should simply print out the phrase. `"Introducing..."` followed by the input parameter `name` (*with no spaces between*) followed by an exclamation point (!).

Hint: your function should contain:

```
def introductions(name):
```

Write some test code to run your function. For example:

```
introductions("Gus")
introductions("Professor Urness")
introductions("The Great Houdini")
```

Should result in

```
Introducing...Gus!
Introducing...Professor Urness!
Introducing...The Great Houdini!
```

Part 2:
Create a function named **`sleep_counter`**. The function should accept a parameter, `hours`, which is a float value.

If the value of `hours` is between the values of 7.0 and 12.0 (including both 7.0 and 12.0) the function should print the phrase, **"You are well rested!"**
However, if the value of `hours` is less than 7.0; and if the value of `hours` is greater than 0.0; the function should print the phrase, **"You need coffee."**
In all other circumstances – if the value of hours is 0.0 or less than 0.0 or if hours is greater than 12.0, the function should print the phrase, **"input error"**

Hint: your function should contain:

```
def sleep_counter(hours):
```

Write some test code to run your function. For example:

```
sleep_counter(7.0)
sleep_counter(8.5)
sleep_counter(3.3)
sleep_counter(-1.1)
sleep_counter(25)
```

Should result in

```
You are well rested!
You are well rested!
You need coffee.
input error
input error
```

Part 3:
Create a function named **`print_num_vowels`**. The function should accept a parameter, `input_string`, which is a string value.

The function should use a `for` loop to loop through each character in the `input_string`. The function should also use an ***accumulator*** variable to keep track of the number of vowels that were in the `input_string`. For the purposes of this assignment, we'll count only `a`, `e`, `i`, `o`, and `u` as vowels. For this assignment, `y` is never considered a vowel. For full credit, the count should be correct for capital and lower-case input strings.

See the examples below for how the output should be formatted.

Hint: your function should contain:

```
def print_num_vowels(input_string):
```

Write some test code to run your function. For example:

```
print_num_vowels("neato")
print_num_vowels("CANDY")
print_num_vowels("ooh")
print_num_vowels("Iowa")
print_num_vowels("THIS IS SO AWESOME")
```

Should result in

```
The string neato contains 3 vowels.
The string CANDY contains 1 vowels.
The string ooh contains 2 vowels.
The string Iowa contains 3 vowels.
The string THIS IS SO AWESOME contains 7 vowels.
```

*Yes, if there is only 1 vowel, the output should be **vowels** (plural). I considered having you make this grammatically correct, but didn't want the code to get too complicated at this point in the semester.*

Disclaimer: there are many different ways to implement this. I am specifically looking for a for loop, using an accumulator as described above and like we have done in class. Many solutions you may find online will not use this approach, and only submissions that use a for loop with an accumulator variable will be awarded credit. If you get help from an outside source, remember to cite it or it will be awarded no credit.

One last reminder: the file you submit in Xuexitong must be named exactly Lab6.py (note the capitalization) and it must contain functions named exactly `introductions` and `sleep_counter` and `print_num_vowels`