

**Name:** \_\_\_\_\_

Use this quiz to help make sure you understand the videos/reading. **Answer all questions.** Make additional notes as desired. **Not sure of an answer?** Ask your instructor to explain in class.

Video: **Calling Functions** [3:22 minutes]

1. Consider the function definition shown to the right:

```
def f_to_c(fahrenheit):  
    celsius = (fahrenheit - 32) * (5 / 9)  
    return celsius
```

- What is the **name** of the function?
  - How many **parameters** does the function have?
2. List 2 reasons why functions are powerful.
    - Reason 1:
    - Reason 2:
  3. Write 2 statements that together call the above function twice, each time doing a different Fahrenheit-to-Celsius conversion:

4. When the statement

```
c = f_to_c(10.5)
```

is executed (where the **f\_to\_c** function is as defined above), what are the 4 steps that occur (be concrete in your answer, referring explicitly to this example wherever you can):

- Step 1:
- Step 2:
- Step 3:
- Step 4:

Handout (and associated exercise in a Python console): **Objects, Types, Values and Variables**

5. Consider the following statements:

```
x = 54
y = 3.713
z = 'my best friend'
```

a. What is the **type** of the object to which the variable **x** refers?

What is the **value** of that object?

b. What is the **type** of the object to which the variable **y** refers?

What is the **value** of that object?

c. What is the **type** of the object to which the variable **z** refers?

What is the **value** of that object?

6. Write a statement that assigns the variable `my_friend` the value `'Betty Bop'`

7. Suppose that you run the following **one-line** program. What happens?

```
print('hello')
```

8. Suppose that you run the following **one-line** program. What happens?

```
print(hello)
```

9. What do the following expressions evaluate to?

```
3 * (4 + 1)
```

```
3 * ('hi' + 'bye')
```

10. What is the value of **y** after the following set of statements executes?

```
y = 5
y = y * 3
y = y + 1
```

11. Assume that you have a variable **x** that has already been given a numeric value. Assume that you have put `import math` at the top of your program. Write a statement that sets the variable **y** to the sum of the sine of **x** and the cosine of **x**.

Handout: **Input-Compute-Output Programs**

The following questions refer to the annotated program shown in the handout.

12. Where does **execution** traditionally begin?

13. What is the:

- first **statement** that executes?
- second **statement** that executes?
- Third **statement** that executes?

14. Where is the **convert\_to\_celsius** function **called** (i.e., **invoked**, i.e., made to run)?

Where is it **defined** (i.e., where are the statements that execute when the function is called)?

15. Where is a **doc-comment**?

What is the purpose of doc-comments?

16. Where is an **internal comment**?

What is the purpose of internal comments?

17. What keyword marks the beginning of a **function definition**?

18. What notation marks the **body** of the function (that is, how can we tell when one function ends and another starts)?

19. How can you tell the **name** of the function being defined?

20. What is the name of the function that causes this program to pause and wait for the user to type some **input**?

21. What is the name of the function that converts that input from a **string** (i.e., a sequence of characters) to a **floating-point number** (i.e., a number that has a decimal point)?

22. List three **variables** in this program.

23. What symbol **assigns** a value to a variable?
24. What side of the assignment is the assigned variable – the **left side** or the **right side**?  
(Circle your choice.)
25. What is that name of the function that **prints** things (i.e., displays them on the console)?

From your **Session01** project: **Examples**

26. In your example file `m2e_hello_and_gooby.py`:
  - a. How many times does the word **Ciao** appear in the program's code?
  - b. How many times does **Ciao** get printed when you run (execute) the program?
  - c. Explain your answer to (b) briefly:
  
27. Using your example file `m3e_input_compute_output.py` as an example, write statement(s) that together: prompts for an integer from the user, gets a string from the user, converts the string into an integer, and stores the integer in the variable `n`.  
Hint: Instead of using **float** (which converts a string to a floating point number), use **int**.

**Reminder:** Don't forget to do the TODO's in the `m6` and `m7` files of your **Session01** project.