



## Exercise 1

Finger Exercises due Aug 6, 2020 01:30 CEST

### Exercise 1

2/6 points (graded)

**ESTIMATED TIME TO COMPLETE: 12 minutes**

**Note that you will have to answer all questions before you can click the Check button.**

#### Part 1: Function Types

For each of the following functions, specify the type of its **return**. You can assume each function is called with an appropriate argument, as specified by its docstring.

If the output can be either an int or a float, select num, which isn't a real Python type, but which we'll use to indicate that either basic numeric type is legal.

In fact, in Python, booleans True and False can be operated on as if they were the integers 1 and 0; but it is ugly and confusing to take advantage of this fact, and we will resolutely pretend that it isn't true.

What are those lines under the function definitions?

```
1. def a(x):  
    ...  
    x: int or float.  
    ...  
    return x + 1
```

Indicate the type of the output that the function **a** will yield.

Answer: num

```
2. def b(x):  
    ...  
    x: int or float.  
    ...  
    return x + 1.0
```

Indicate the type of the output that the function **b** will yield.

✔ Answer: float

```
3. def c(x, y):  
    ...  
    x: int or float.  
    y: int or float.  
    ...  
    return x + y
```



Indicate the type of the output that the function `c` will yield.

**✓ Answer: num**

```
4. def d(x, y):  
    ...  
    x: Can be int or float.  
    y: Can be int or float.  
    ...  
    return x > y
```

Indicate the type of the output that the function `d` will yield.

**Answer: boolean**

```
5. def e(x, y, z):  
    ...  
    x: Can be int or float.  
    y: Can be int or float.  
    z: Can be int or float.  
    ...  
    return x >= y and x <= z
```

Indicate the type of the output that the function `e` will yield.

**Answer: boolean**

```
6. def f(x, y):  
    ...  
    x: int or float.  
    y: int or float  
    ...  
    x + y - 2
```

Indicate the type of the output that the function `f` will yield.

**Answer: NoneType**

### Explanation:

The last function, unlike the other ones, does not have a return statement. It only does an operation (the operation is `x+y-2`). Since it does not explicitly return anything, Python by default returns the value `None` whose type is `NoneType`. So this function and any other that does not have a return statement can be rewritten as:

```
def f(x, y):  
    x + y - 2  
    return None
```

[Submit](#)

## Exercise 1

Topic: Lecture 4 / Exercise 1

[Show Discussion](#)



