



Practical Programming in Python

Inspired by 'Practical Programming' by Paul Gries, Jennifer Campbell, Jason Montojo

Lecture 7: Summary & Exercises Using Methods

Types, Classes, Methods, Object Orientation

“Any sufficiently advanced technology is indistinguishable from magic.”

– Arthur C. Clarke

Lecture 7: Summary

In this lecture you learned the following:

- Classes are like modules, except that classes contain methods and modules contain functions.
- Methods are like functions, except that the first argument must be an object of the class in which the method is defined.
- Method calls in this form: `'browning'.capitalize()` are shorthand for this: `str.capitalize('browning')`.
- Methods beginning and ending with two underscores are considered special by Python, and they are triggered by particular syntax.

Lecture 7: Exercises

When writing code, only use Python concepts that have been introduced in the lectures already.

Exercise 1:

In the Python shell, execute the following method calls:

- a. `'hello'.upper()`
- b. `'Happy Birthday!'.lower()`
- c. `'WeeeEEEEeeeeEEEEeee'.swapcase()`
- d. `'ABC123'.isupper()`
- e. `'aeiouAEIOU'.count('a')`
- f. `'hello'.endswith('o')`
- g. `'hello'.startswith('H')`
- h. `'Hello {0}'.format('Python')`
- i. `'Hello {0}! Hello {1}!'.format('Python', 'World')`

Exercise 2:

Using string method `count`, write an expression that produces the number of o's in `'tomato'`.

Exercise 3:

Using string method `find`, write an expression that produces the index of the first occurrence of o in `'tomato'`.

Exercise 4:

Using string method `find`, write a single expression that produces the index of the second occurrence of o in `'tomato'`. Hint: Call `find` twice.

Exercise 5:

Using your expression from the previous exercise, find the second o in `'avocado'`. If you don't get the result you expect, revise the expression and try again.

Exercise 6:

Using string method `replace`, write an expression that produces a string based on `'runner'` with the `n`'s replaced by `b`'s.

Exercise 7:

Variable `s` refers to `'yes'`. When a string method is called with `s` as its argument, the string `'yes'` is produced. Which string method was called?

Exercise 8:

Variable `fruit` refers to `'pineapple'`. For the following function calls, in what order are the subexpressions evaluated?

- `fruit.find('p', fruit.count('p'))`
- `fruit.count(fruit.upper().swapcase())`
- `fruit.replace(fruit.swapcase(), fruit.lower())`

Exercise 9:

Variable `season` refers to `'summer'`. Using string method `format` and variable `season`, write an expression that produces `'I love summer!'`

Exercise 10:

Variables `side1`, `side2`, and `side3` refer to 3, 4, and 5, respectively. Using string method `format` and those three variables, write an expression that produces `'The sides have lengths 3, 4, and 5.'`

Exercise 11:

Using string methods, write expressions that produce the following:

- A copy of `'boolean'` capitalized
- The first occurrence of `'2'` in `'CO2 H2O'`
- The second occurrence of `'2'` in `'CO2 H2O'`
- `True` if and only if `'Boolean'` begins lowercase
- A copy of `"MoNDaY"` converted to lowercase and then capitalized
- A copy of `" Monday"` with the leading whitespace removed

Exercise 12:

Complete the examples in the docstring and then write the body of the following function:

```
def total_occurrences(s1, s2, ch):
    """
    Return the number of times that ch occurs in s1 and s2.

    Precondition:
        len(ch) == 1

    Examples:
        >>> total_occurrences('color', 'yellow', 'l')
        3
        >>> total_occurrences('red', 'blue', 'l')

        >>> total_occurrences('green', 'purple', 'b')
    """
```

Exercise 13:

Save the function from the previous exercise in a file. Then use the `doctest` module to verify it works as intended using the technique introduced in lecture 6.