

CSCA20 Lab 3 (Week 4)

Objectives

- Practice using `help` and `dir`.
- Practice writing boolean expressions and conditional statements.
- Write functions involving type `str`

Str methods

For this section, feel free to refer to the online Python documentation:

<https://docs.python.org/3/library/stdtypes.html#str>

Additionally, you can use Python's built-in functions `dir()` and `help()` for quick reference. For example, if you'd like a list of all of Python's `str` methods, you can call `dir(str)`. If you do this, make sure to ignore everything starting with two underscores.

To see the Python docstring of a particular `str` method, for example `str.islower()`, call `help(str.islower)`. Note that we are *not* putting brackets after `islower`; we are trying *not* to call `str.islower()`. Instead, we're trying to pass the method `str.islower` directly to `help()`.

Complete the tasks below using functions from Python's `str` methods in the shell.

- Calculate the length of the string 'Hippopotomonstrosesquippedaliophobia'
- Find the number of occurrences of the letter 'a' in 'banana'
- Convert 'welcome to csc20' to all-caps
- What does 'welcome to csc20'.rstrip() do?
- What does '123'.isdigit() return?
- Exchange all occurrences of 'procrastinating' with 'starting my work early' in the str 'procrastinating is my strategy for success in CSCA20. I like procrastinating.'

Boolean expressions and conditionals

This section asks you to write the code for the functions specified below.

Put your functions in a file `lab3_p0.py`. The functions:

- `is_long(string)`: Given a string, if the length of the string is bigger than 10, return the string 'very long'. Otherwise, return the string 'kinda short'
- `shorter(string0, string1)`: Given two strings, return the length of the shorter string
- `later(string0, string1)`: Given two strings made up of lowercase letters, return the string that would appear later in the dictionary
- `where(string0, string1)`: Given a string and a single-character string, return the index of the first occurrence of the second string in the first. For example, `where("abc", "b")` should return 1. If the second string is not in the first, return -1
- `is_vowel(string)`: Given a one-character string, return True if it is a vowel, and return False otherwise

It is a good idea to write a docstring for each function. The docstring format should look like the following:

"""<one line description>

<multi line description>

Args:

<a list of arguments that the function takes, and the type of each argument>

Returns:

<a description of the return value of the function, if any>
"""

Modules

In a new file lab3_p1.py, do the following:

- Import your module lab3_p0
- Print the return values of the following function calls:
 - lab3_p0.is_long('abc')
 - lab3_p0.is_long('abcdefghijkl')
 - lab3_p0.shorter('abc', 'a')
 - lab3_p0.shorter('abc', 'abcde')
 - lab3_p0.later('abc', 'bcd')
 - lab3_p0.later('cde', 'bcd')
 - lab3_p0.where('abc', 'a')
 - lab3_p0.where('bcd', 'a')
 - lab3_p0.is_vowel('a')
 - lab3_p0.is_vowel('b')

Submit

Submit both lab3_p0.py and lab3_p1.py on MarkUs

<http://markus.utoronto.ca/csca20f18/>