

# Lab exercise set 9

Due Wednesday, May 27th, 11:59pm

## Logistics

These exercises should be completed during the lab on Thursday. In order to receive full credit for the lab, you must attend the session, remain in the lab for at least 45 minutes, and submit a file that contains solutions to all these exercises.

You are encouraged to work in groups on lab exercises. If you do work with someone, you must include the name(s) of your collaborator(s) at the top of the file you submit. For more information about collaboration policies in this class, see the Academic Integrity policy.

If you complete the lab exercise early, please work on the eighth assignment. You must remain in the lab for at least 45 minutes to earn full credit. You are only allowed to work on assignments with at most two other people, either directly or indirectly, who must be formally identified as part of your assignment submission. Please see the assignment description and the course Academic Integrity pledge for the full description of the process you must follow when completing assignments. If you need additional help on the assignment, please ask the lab assistant.

## Exercises

**Please remember that you are not allowed to consult online resources when completing lab exercises.** If you have questions, please either ask the lab assistant or contact me by email.

Begin the lab by downloading the template file (**csc242lab9.py**) from the D2L site. It contains the headers for the functions you will write. Do not modify the function names or parameters in the template file. The functions written for this lab **must not** use global variables. Solutions that don't follow this rule will not earn full credit, even if they produce the correct results in all cases.

1. We will be parsing and crawling HTML documents. Let's take some time to make sure you understand the structure of an HTML document. Write a simple HTML document that has the following information:
  - a. Must be a properly formatted HTML document with `html`, `head` and `body` tags.
  - b. Filename: `lab9.html`
  - c. A title of 'Lab 9 Sample Page'
  - d. A heading (`h1`) with a value 'Important Links'

- e. A sub heading (h2) with value 'CSC 242 Fall 2019/2020'
  - f. An unordered list of links the following sites
    - a. <http://www.cdm.depaul.edu> labeled 'DePaul CDM'
    - b. <http://www.python.org> labeled 'Python Download'
    - c. <https://zoko.cdm.depaul.edu> labeled 'Mr. Zoko's CSC 242 site'
    - d. <https://d2l.depaul.edu> labeled 'D2L'
  - g. A link that will open up and email app to email you at your email address labeled 'Email Me'.
2. Implement a **recursive** function **fileStartsWithCount()** that takes as parameters the name of a folder and the characters a filename starts with so we can return a count of how many files begin with those characters were found in the folder and subfolders. The function will return an integer representing the count of files whose name starts with those characters. Please note that your function must work as described on any directory structure, not just the one provided as an example. You must use recursion to go through the subfolders. You may not use python crawlers or apis other than the ones used in the antivirus-redux example to find files in sub directories. Doing so will be an automatic 0 for this problem. You may use a for loop to loop through the contents of a folder. The following illustrates several searches using a sample set of folders and directories located in the zip file containing the exam template. You ***don't*** need to worry about punctuation, case or file error handling for this problem.

Hint: str has a startswith method:

```
>>> s='one two three'
>>> s.startswith('o')
True
>>> s.startswith('a')
False
>>> s.startswith('one')
True
>>> s.startswith('one ')
True
>>> s.startswith('one b')
False
>>> |
```

```
>>> fileStartsWithCount('Test','c')
3
>>> fileStartsWithCount('Test','f')
11
>>> fileStartsWithCount('Test','fi')
11
>>> fileStartsWithCount('Test','ca')
3
>>> fileStartsWithCount('Test','cat')
0
>>> fileStartsWithCount('Test','fa')
0
```

## Submitting the exercises

You must submit your solutions to the exercises using the lab 9 dropbox on [the D2L site](#). Submit only a single Python file (e.g. **csc242lab9.py**) with each of the completed functions and classes for the lab exercises in it. Submissions after the deadline listed above will be automatically rejected by the system. See the syllabus for the grading policy.

## Grading

The lab session is worth 10 points.