author: Ted Krovetz
category topic: CSC 135
full title: Python in CSC 135
related notes: 2022-01-24T1852-CSC 135-Course-Information-Syllabus
source: https://krovetz.net/135/module_python/readme.html

Python in CSC 135

WW04.2 | Tuesday, January 25, 2022 | 10:26 AM

1. Python is popular
    1. Easy to learn
        1. The core Python language is small and clean
    2. It's a dynamically typed and interpreted language
        1. Don't have to declare variable types
        2. Don't compile code before running it
            1. Small programs are faster to write and execute
    3. Libraries → making it possible to get work done in many domains
2. Learning Python
    1. Python is not a prerequisite for the course
        1. As a result we will be using the first two weeks learning it
        2. Will not go deep, but will learn enough to write small programs and classes
    2. Should work through the first 5 lessons of Udacity's free Python class
        1. Will take 10 hours (plus/minus a few hours).
    3. Practice your Python skills at https://www.codestepbystep.com/
        1. CSC 20 difficulty
        2. Some problems will be turned in for credit
3. Use of Python in this class
    1. Class topic functional programming
        1. Traditionally used with languages like Scheme, Racket, or Haskell
        2. But will be going to take a different approach
            1. Want students to learn Python
            2. Use Python to explore what it means to program in a functional style
                1. Data structures are designed to be mutation-free
                    1. Once an object is created it cannot be changed (eg, Strings in Java)
                2. Functions and methods have no side effects
                    1. Functions and methods should not change any variables

        outside the function or method's scope.
        1. Function should simply be it's return value
    3. Use of higher-order functions and lambda functions.
4. **Python Resources**
    1. Follow the formatting conventions of the Python community: https://pep8.org/
    2. Here's a simple webpage that finds some of your formatting errors: https://www.pythonchecker.com/
    3. There are many Python tutorials out there. Two I've used are https://www.programiz.com/python-programming and https://www.w3schools.com/python
    4. If you'd prefer a book, here's a good free one: https://www.py4e.com/
    5. The official language documentation: https://docs.python.org/3/