Boston University Metropolitan College

NOTE: The template below lists the minimum requirements for all syllabi at Metropolitan College. Instructors may add to this as their course and instruction styles require. Please discuss with your department chair any department-specific requirements, as well as questions you may have regarding writing your syllabus.

ALL SYLLABI MUST BE SUBMITTED EACH SEMESTER TO THE DEAN'S OFFICE IN ADDITION TO THE DEPARTMENTS.

Course Title: Information Structures with Python

Course Number: BU MET CS 521
Course Format: online

Instructor Name: Eugene Pinsky
epinsky@bu.edu

Course Time: Tue 6 pm – 8 pm
Sun 10 am - 12 pm

Office hours: by appointment/email

Course Description

This course will present an effective approach to help you learn Python. With extensive use of graphical illustrations, we will build understanding of Python and its capabilities by learning through many simple examples and analogies. The class will involve active student participation, discussions, and programming exercises. This approach will help you build a strong foundation in Python that you will be able to effectively apply in real-job situations and future courses.

Books

Required:

1. Introduction to Programming Using Python by Y. Daniel Liang (Pearson Publishing), ISBN 978-0-13274718-9

2. Course notes (from the course website) – presentation slides

Recommended:

(1) students may purchase access card to MyProgrammingLab for this book. Students are strongly encouraged to practice additional programming problems. These problems can to be submitted electronically for grading to MyProgrammingLab. We may discuss some of these exercises throughout the course.

(2) Python Programming Environment – we will be using Spyder IDE (Integrated Development Environment) and Anaconda Python Distribution. We have these installed in our virtual lab. MET Virtual Labs (VLAB) provide students with all required software. Most of the examples presented in class will be run in this environment. You can familiarize yourself with the virtual labs with the information from our website: http://www.bu.edu/metit/pc-labs/virtual-labs/

There are many on-line resources available. This is a partial list:

1. http://www.pythontutor.com/visualize.html - this website is very useful and allows to run simple Python programs and visualize the execution. Many of the illustrations in the course notes were generated using this website.
2. https://docs.python.org/2/tutorial - an official Python tutorial
3. https://www.tutorialspoint.com/python - a detailed tutorial with many simple examples
4. https://www.learnpython.org - free, interactive tutorial
5. https://www.python.org/community/sigs/current/edu-sig/ - contains links to learning resources, including two free books

Courseware

Blackboard
Course Notes

Class Policies

1) **Attendance & Absences** – clearly state your attendance policy, limit to absences, etc. List all unusual required meetings (e.g. exhibits, guest lectures, field trips, etc.)

2) **Assignment Completion & Late Work** – detail your policy regarding how students should submit completed assignments (in person, by email, on courseware site, etc.), as well as how you will address late work.

Weekly programming assignments submitted through blackboard on-line. Late homework is accepted with 50% penalty. Final projects are submitted through blackboard on-line. Students will present their projects on the last day of class. Both quizzes and final are closed-book, and are taken 'on-line'

3) **Academic Conduct Code** – Please use the following wording, or an equivalent, in your syllabus: "Cheating and plagiarism will not be tolerated in any Metropolitan College course. They will result in no credit for the assignment or examination and may lead to disciplinary actions. Please take the time to review the Student Academic Conduct Code:

Academic conduct code as specified below:

http://www.bu.edu/met/metropolitan_college_people/student/resources/conduct/code.html.

NOTE: [This should not be understood as a discouragement for discussing the material or your particular approach to a problem with other students in the class. On the contrary – you should share your thoughts, questions and solutions. Naturally, if you choose to work in a group, you will be expected to come up with more than one and highly original solutions rather than the same mistakes.]

Grading Criteria

Give a detailed list of percentage weights for assignments, papers, class participation and examinations as applicable. If you have complex grading criteria, please spell this out here as clearly as possible. Remember: the syllabus is a contract between you and your students, and will be referred to as such in the event a dispute arises.

35% homework, 15% quizzes, 30% final, 20% final project.

Class Meetings, Lectures & Assignments

Boston University Metropolitan College

List in a legible format all of the class meetings, lectures, and assignments. One example, based on a computer science course:

Lectures, Readings, and Assignments subject to change, and will be announced in class as applicable within a reasonable time frame.

Date	Topic	Readings Due	Assignments Due
Week 1	introduction to computing and problem solving, Python programming environment, Python IDEs, iPython Notebook environment, modules, input/output, running Python, core data types, simple expressions	Chapters 1, 2 Course notes: Overview, Python environment, types, mutability	
Week 2	variables, immutability, expressions, operators and Boolean expressions, operator precedence	Chapters 2, 3 Course notes: Types and variables	Homework 1 Quiz 1
Week 3	mathematical functions, strings and text manipulation, selections, control flow (if, break, continue, for, while) and iterations, files and file manipulation	Chapters 4,5,8, 13 Course notes: Collections, control flow, files, strings, string indexing and slicing, string methods	Homework 2 Quiz 2

Week 4	collections, set membership and comprehension, lists, tuples, sets, dictionaries, searching and sorting	Chapters 10, 11, 14 Course notes: Dictionaries, lists, sets, tuples, indexing and slicing, use and methods	Homework 3 Quiz 3
Week 5	advanced data structures, functions, exception handling, parameter passing, recursive functions	Chapters 6, 15 Course Notes: Exceptions, functional programming, functions, recursion	Homework 4 Quiz 4
Week 6	objects and classes, attributes, methods, data encapsulation, abstract classes, inheritance and polymorphism	Chapters 7, 12 Course notes: Classes, inheritance, polymorphism	Homework 5 Quiz 5
Week 7	course review, project presentations and final exam		Homework 6 Quiz 6