**Information Structures with Python**

CS 521

Online

Eric Braude

[ebraude@bu.edu](mailto:ebraude@bu.edu)

Office hours: by appointment

**Course Description**

This course covers the concepts of the object-oriented approach to software design and development using the Python programming language. It includes a detailed discussion of programming concepts starting with the fundamentals of data types, control structures methods, classes, arrays and strings, and proceeding to advanced topics such as inheritance and polymorphism, creating user interfaces, exceptions and streams. Upon completion of this course students will be capable of applying software engineering principles to design and implement Python applications that can be used in conjunction with analytics and big data.

**Books and Principal References**

1. Python Programming: An Introduction to Computer Science by John Zelle (Franklin Beedle & Associates), 2010, to be available at Barnes & Noble
2. Introduction to Computation and Programming Using Python by John V. Guttag, (MIT Press) revised and expanded edition, 2013, to be available at Barnes & Noble
3. Official online Python documentation [www.python.org](http://www.python.org)

**Courseware**

Blackboard will be necessary for the class

**Class Policies**

1. **Attendance & Absences** – students are expected to log in regularly
2. **Assignment Completion & Late Work** – assignments are to be submitted on time via blackboard. Late homework will be accepted if a reason is given in advance why its completion on time was not possible. Otherwise, late homework may not count.
3. **Academic Conduct Code** – 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:

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

**Grading Criteria**

Assignments are weighted 60%, quizzes 10%, and the final 30%.

**Class Meetings, Lectures & Assignments**

*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** |
| May 25 | Module 1: Introduction  -- Relationship between computers and programs  -- Data types, variables, expressions, and statements  -- Using the Python interpreter  -- Control Structures: loops and decision  -- Numerical computing | Guttag Chapters 1, 2 | n/a |
| June 1 | Module 2: Modularization and Files  -- Defining functions  -- Modules  -- Packages  -- Standard modules  -- Reading and writing files | Guttag Chapter 3, 4 | A first set of Python programs |
| June 8 | Module 3: Classes  -- Functions and their arguments  -- Defining Classes  -- Object Oriented Programming  -- Inheritance and Polymorphism | Zelle Chapter 4  Guttag Chapter 7 | Organizing an educational game implementation |
| June 15 | Module 4: Exceptions and User Interfaces  -- Error processing  -- Exception Raising and Handling  -- Graphics  -- Coordinates and color  -- Data Structures (Arrays, Lists, Maps, and Iterators) | Zelle Chapters 1, 2, 3  Guttag Chapter 5 | User Interfaces for an educational game |
| June 22 | Module 5: Data Structures and Databases  -- Sets, dictionaries, and hashing  -- Linked Lists  -- Stacks  -- Queues  -- Database access | Guttag Chapter 10 | Practice exercises for data structures and algorithms |
| June 29 | Module 6: Multithreading and Web Access  -- Multithreading and synchronization  -- Networking  -- Internet computing  -- Web Access | https://docs.python.org/2/tutorial/stdlib2.html#multi-threading | Sprite implementation |
| July 6 |  |  | Final |