# Python Programming, ELENG X442.3

1 Semester credit in EECS

## Course Description

### This course provides an introduction to the fundamentals of Python programming. First, we’ll look at the types of data that Python supports and the different ways they can be stored and manipulated. Next we’ll examine the basics of reading and writing to files, and the ways that Python uses object-oriented programming to make it easy to access both local and remote information.

### Then we’ll examine the programming constructs that are available in Python. One of Python’s most attractive features is the simplicity of its core language, so even if you don’t have any programming experience, you won’t be overwhelmed. We’ll look at the ways that you can incorporate conditional execution (**if** statements) and iterative execution (**for** and **while** loops) into your programs to make them more useful. We’ll also discuss Python’s consistent method of dealing with problems when they arise.

### Having covered the basics of the language, we’ll learn how to create functions and use the many functions that are a part of Python’s standard library to manipulate text, access your computer’s operating system, communicate with remove computers, and write CGI scripts for Web servers.

### Finally, we’ll delve a little deeper into the workings of Python’s object-oriented programming model. We’ll learn how to create modules that introduce new types of data into Python, and how to manipulate that data.

## Learning Outcomes

After successfully completing this course, you will:

* Be able to store, access, and manipulate both numeric and character data with Python;
* Understand the difference between arrays, tuples, and dictionaries for storing objects in Python;
* Be able to read and write files and access operating system commands from Python programs;
* Be able to search for and modify text in documents by using regular expressions;
* Be able to access and process information on Web servers from within a Python program;
* Be able to create a module of object definitions and methods to extend and customize Python for your own uses.

## Course Materials and Technical Requirements

### Required Textbook

### *The Quick Python Book, 2nd ed.* (Vernon L. Ceder, Manning, 2010)

### Recommended Readings Textbook

* *Think Python* (Allen B. Downey, Green Tea Press) Available for free download under Creative Commons License: <http://www.greenteapress.com/thinkpython/>

### *Hello! Python* (Anthony Briggs, Manning, 2012)

You are free to purchase your textbooks from any vendor. Please be sure to thoroughly review the return policies before making a purchasing decision as UC Berkeley Extension does not reimburse students for course materials in the event of a textbook change or an unexpected cancellation or rescheduled course section.

### Technical Requirements

You will need access to a computer with Python installed -- preferably version 3. Python is available for Windows, UNIX, and Macintosh operating systems. If you're using a Linux system, Python is probably already installed. If you need to install Python, follow the links at Python.org.

If you're interested in CGI programming, you should have access to a computer on which you can install CGI programs.

This course is built on a Learning Management system (LMS) called Canvas. In order to use Canvas, this is a list of basic [computer system requirements](http://guides.instructure.com/s/2204/m/4214/l/82542-what-are-the-basic-computer-specifications-for-canvas).

### Optional

Canvas allows you to record audio or video files of yourself and upload them in the course. Although doing so is not required for any of the activities, using these features will enhance your engagement in the course. If you would like to use these features, you will need to have a webcam and a microphone installed on your computer.

## Learning Activities

This course consists of ten modules. Each module includes a combination of reading assignment, learning objectives, and instructor commentary on the topics covered. Among the learning activities, you will either have a quiz at the completion of the module, or you will turn in a written assignment. You are also expected to participate in online discussions for each module.

### Reading

Each module will correspond to one or more suggested readings in the textbooks and/or supplementary reading material provided in the online classroom. If the material in the commentary is confusing to you, the text is a good resource to then go back to and enhance your understanding.

### Written Assignments

### Written assignments are intended to enhance your understanding of the online course materials. In many cases, it also requires accessing the wealth of online material available for support through the web community. Your assignments will help the instructor judge progress and make corrections to your understanding.

### Discussions

Students will respond to posted discussion topics in each section of the module and will be required to read and respond to at least one other student postings for each topic. In addition, students can ask questions, answer the questions posted by their classmates, and post topics of their own in the "Student Lounge" forum. Students who post facts or figures from other sources must also provide appropriate reference citations in their posts. Discussion assignments are a way for you to interact with your peers in this online class where the format can sometimes be isolating.

Each of us brings a unique perspective to this class based upon our life experiences and previous studies. Because of the asynchronous nature of this course, at times it might be hard to sustain an actual conversation in the discussion threads with your fellow students. So instead, let's look to the discussion threads as a place to connect our multiple perspectives and construct an evolving knowledge base. The process will enrich your own studies, and the words you leave behind will help students who join the class even after you've gone on to new pursuits.

When a module has a discussion assignment, please respond to the questions in a manner that reflects critical thinking. Please feel free, as well, to positively critique and offer leads and suggestions to comments and questions that other students have made. (A rubric is included in this syllabus, below).

In Module 1 you'll introduce yourself to your instructor and course mates. For your introduction, I encourage you to use Canvas features to record your voice and/or a video or yourself. You may also click on your name at the top of the course and upload a photo so that your course mates and I can get a better sense of you.

### Quizzes

Five modules will close with a quiz to test your knowledge of the module’s material. The quizzes will be administered through the Learning Management System and may be multiple choice and/or short answer. Please take your time, review carefully, and study.

### Final Exam

After completing all the modules, you will take a supervised, closed-book final examination. It covers the entire course, and you will be given three hours to complete it. It consists of multiple-choice questions as well as essay/programming questions. You must pass the final exam with a score of at least 70% or better to pass the course. You can find more information about the proctored final exam process at the Extension website: http://extension.berkeley.edu/static/studentservices/onlineinfo/#exams

### Final Project

### The final project is due in Module 10. Choose one of the following options:

### A file with a name like picture.jpg is said to have an extension of “jpg”; i.e., the extension of a file is the part of the file after the final period in its name. Write a program that takes as an argument the name of a directory (folder) and then finds the extension of each file. Then, for each extension found, it prints the number of files with that extension and the minimum, average, and maximum size for files with that extension in the selected directory.

### Write a CGI script that presents a multiple-choice test to a user. When the user presses submit, your program should read the user’s responses to the questions and display a page showing the user’s score. You can hard code the questions (and correct answers) inside your program, or read them from an external file.

### You will submit your final project by uploading it in Module 10.

## Communication and Office Hours

You can always get in touch with me during the course. You can access course email by clicking on the Inbox link on the Corner Help toolbar (see also [Canvas tutorial](http://vimeo.com/74677643)). You can expect me to respond to email within 48 hours of receiving messages unless I have notified the class otherwise (e.g., because of vacation or other reasons).

Please note: all course communication between students/instructor must occur within the course.

## Grading and Course Policies

Final grades will be assigned according to the following percentages:

|  |  |
| --- | --- |
| Discussion Board Assignments | % 10 |
| Written Assignments | % 15 |
| Quizzes | % 15 |
| Final Exam | % 40 |
| Comprehensive Final Project | % 20 |

Table 1: Final Grade Percentages

You must pass the final project requirements with at least 70% to pass the course.

### Grading Information

Final grades follow the UC Berkeley grading system:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Letter Grade | A | A- | B+ | B | B- | C+ | C | C- | D+ | D | D- | F |
| Percentage | 100-94 | 93-90 | 89-86 | 85-83 | 82-80 | 79-76 | 75-73 | 72-70 | 69-66 | 65-63 | 62-60 | < 60 |

Table 2: UC Berkeley Grading Systems

To view your final grade and request official transcripts, log in to your student account at [extension.berkeley.edu](http://extension.berkeley.edu/) and go to "My Enrollment History."

When I grade your writing assignments, I'll be looking at content, organization, and mechanics. Please keep the following criteria in mind:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Poor | Needs Improvement | Meets Expectations | Exceptional |
| Content | Poor writing style with little or no specific details, no evidence of having studies the material, and/or off topic. | Adequately written; some points elaborated but with minimal use of concepts from the material. | Well, written, most points elaborated with clear and detailed information that supports thoughts and ideas and uses concepts from the material. | Well written, fully elaborates points. Clear and detailed information supports thoughts and ideas and shows full acquisition of concepts from the material. |
| Organization and Mechanics | Little or no structure present. Grammatical errors interfere with comprehension. | Organization present but awkward. Some grammatical errors present. | Good organization with few statements out of place. Minor grammatical errors. | Clearly organized and remains focused. Few or no grammatical errors. |

Table 3: Criteria for Writing Assignments

In the Discussion Assignments forum, I'll also be looking for evidence of participation:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Poor | Needs Improvement | Meets Expectations | Exceptional |
| Participation | Minimal posts in number or length. Posts show little or no reflection on the topics or previous posts, or are rude or abusive. | Posts address the topic but consist mostly of a rote repetition of the study materials. Little or no reflection on previous posts. | Posts address the topic with reflection. Many responses build on previous posts. | Posts show a genuine interest in contributing to the overall life of the forum. Information is accurate and trustworthy. Comments are supportive. |

Table 4: Evidence of Participation for the Discussion Assignments Forum

### DSP Accommodations

Any student requiring course accommodations due to a physical, emotional, or learning disability must contact the [Disabled Student Services](http://extension.berkeley.edu/static/studentservices/career/) (DSS) office right away. Be sure to review our detailed [DSP accommodations instructions](http://extension.berkeley.edu/static/studentservices/career/#disabled).

## Academic Integrity, Research, and Proper Citation

As an online student, you are encouraged to reach out to your fellow students in the online classroom to discuss materials and ask each other questions, but there are limits to this collaboration. Reviewing lecture and reading materials and studying for exams can be enjoyable and enriching things to do with fellow students. This is recommended. However, unless otherwise instructed, homework assignments are to be completed independently, and materials submitted as homework should be the result of your own independent work.

As a UC Berkeley student you are bound by the [UC Berkeley Extension Code of Student Conduct,](http://extension.berkeley.edu/upload/academic_integrity.pdf) which clearly defines what constitutes cheating, as well as plagiarism and other forms of academic misconduct.

You will be asked to take your Pledge to Academic Integrity before you can access the course content.

## Course Evaluation and Course End Date

### Course Evaluation

UC Berkeley Extension is committed to improving our online courses and instruction. Please take a few minutes to participate in our Course Evaluation to share your opinions about this course. We are interested in your online learning experience, and your candid feedback will help us plan for the future and make improvements. Please complete the evaluation before your course End Date. The evaluation does not request any personal information, and your responses will remain strictly confidential.

### Course End Date

Your access to the online classroom will expire on the course End Date, which is indicated in the initial e-mail you received when you enrolled.

As you work through the course, please keep the End Date in mind, and if you want to save any commentary or assignments for future reference, please make sure to print or copy/paste those materials before your access ends.

## Canvas Tech Support and UC Berkeley Extension Student Services

### Canvas Tech Support

The learning management system (LMS) used in this course is Canvas, which has convenient mobile apps for phones and tablets. Part of the orientation materials in your course will help you make sure that your computer is at par with Canvas specifications. Anytime you are in Canvas you can report problems, get support, and search Canvas user guides from the Help link on the top menu bar. Other options:

* Canvas Support 24/7 Hotline:  855-308-2758
* Email: support@instructure.com

### UC Berkeley Extension FAQs and Student Services

Start at the [Student Services page](http://extension.berkeley.edu/static/studentservices/) to find help with issues such as the following:

* Course registration
* Exam proctoring services
* Refunds, withdrawals, and transfers
* Grade options
* Requests for transcripts or official receipts

If you need further help, Extension's [Online Learning page](http://extension.berkeley.edu/static/online/#contact)s lists contact information for the online program coordinators for Extension's academic departments.

## Pacing Yourself through the Course

With this course, you have the freedom to design your study time to meet your schedule. However, this style of learning also requires dedication and commitment to ensure that you get the homework and assignments done in a timely fashion. Use these tips to stay on track and get the most from this class.

* **Get started as soon as possible**: Students who submit their first assignment within the first month are more likely to complete the course than those who delay. Please do not wait to submit everything toward the end of the class. Assume that I will need a week to grade each assignment and provide feedback that will help you complete subsequent assignments.
* **Create a planning calendar**: Plan your homework and submittal dates, and stick with them. Students who submit assignments regularly are more likely to complete the course than those who do not. Take the time now at the beginning of the course to plan your study time by using the course Calendar tool.
* **Ask questions**: You are always encouraged to ask questions. Use course mail to ask me questions, ask for feedback, or just to request encouragement. I'm always pleased to be of assistance. Use the Student Lounge forum for general questions about the course or assignments that other students might need answered as well.
* **Submit complete assignments**: Submit only completed assignments. If you are unsure about a question, e-mail me and ask me first before submitting the assignment. I'm always pleased to be of assistance. Any incomplete assignments will be returned to you to complete and resubmit.

## Course Outline

You'll find complete instructions for your assignments within the course modules.

**Module 1: The Basics**

**Reading Assignments**

Textbook Chapters 1-4

**Commentary**

1.1 Getting Started

1.2 History of Python

1.3 Basic Concepts

1.4 Errors

1.5 Running Python

1.6 Using Python as a Calculator

**Quiz 1**

**Discussion Assignments**

Introductions

The Basics of Python

**Module 2: Strings and Data**

**Reading Assignments**

Chapter 6

**Commentary**

2.1 Introduction to Strings

2.2 String Literals and Variables

2.3 String Indexing

2.4 String Slicing

2.5 The in Operator

2.6 String Methods versus Functions

2.7 The split and join Methods

2.8 Methods to Search for Substrings

2.9 Methods to Manipulate Text and Case

2.10 Types of Numeric Data

2.11 Octal and Hexadecimal Constants

2.12 Operators for Numeric Values

2.13 Functions for Numeric Values

2.14 Conversion of Scalar Types

**Quiz 2**

**Discussion Assignment**

Data in Python

**Module 3: Lists, Tuples, and Dictionaries**

**Reading Assignments**

Chapters 5 and 7

**Commentary**

3.1 Working with Lists

3.2 Indexing, Slicing, and Deletion in Lists

3.3 Operators and Lists

3.4 Copying Lists

3.5 Object Orientation and Methods for Lists

3.6 List Manipulation Methods

3.7 List Methods for Sorting and Counting

3.8 Tuples

3.9 Dictionaries

3.10 Functions and Methods for Dictionaries

**Quiz 3**

**Discussion Assignment**

Lists, Tuples and Dictionaries

**Module 4: Input/Output**

**Reading Assignment**

Chapter 6 (part 6.7) and Chapter 13

**Commentary**

4.1 Printing and the Modulus Operator (%)

4.2 Opening a File

4.3 Reading a File

4.4 Writing to a File

4.5 Additional File Methods

4.6 File Object Variables

4.7 Standard Streams

4.8 Pipes

**Quiz 4**

**Discussion Assignment**

Input/Output

**Module 5: Programming and Control Structures**

**Reading Assignments**

Chapter 8

**Commentary**

5.1 Assignments

5.2 Indentation

5.3 Truth, Falsehood, and Logical Operators

5.4 Introduction to Control Structures

5.5 The if, else, and elif Statements

5.6 The for Loop

5.7 The range Function

5.8 The while Loop

5.9 Control in Loops: break and continue

**Written Assignment**

Control Structures

**Discussion Assignment**

Programming and Control Structures

**Module 6: Writing Functions**

**Reading Assignments**

Chapter 9

**Commentary**

6.1 Functions

6.2 How Python Finds Your Variables

6.3 Named Arguments and Default Values

6.4 Variable Numbers of Arguments

6.5 Functional Programming and Anonymous Functions

6.6 Comprehensions

**Written Assignment**

Functions

**Discussion Assignment**

Writing Functions

**Module 7: Exceptions and Error Handling**

**Reading Assignments**

Chapter 14

**Commentary**

7.1 try/except

7.2 Multiple Exceptions

7.3 Hierarchy of Exceptions

7.4 Raising Exceptions

7.5 Other Uses for Exceptions

**Written Assignment**

Error Handling

**Discussion Assignment**

Exceptions and Error Handling

**Module 8: Using Modules: String Handling and Regular Expressions**

**Reading Assignment**

Chapters 6 and 17

**Commentary**

8.1 Introduction to Modules

8.2 The string Module

8.3 Introduction to Regular Expressions

8.4 Regular Expression Objects

8.5 Tagging of Regular Expressions

8.6 Multiple Matches

8.7 Substitutions

**Quiz 5**

**Discussion Assignment**

Using Modules: String Handling and Regul

**Module 9: Using Modules: Operating System and Internet**

**Reading Assignments**

Chapter 12

**Commentary**

9.1 The os Module

9.2 os.path

9.3 os.path.walk

9.4 Other Modules: shutil, sys, and copy

9.5 Object Persistence: pickle and shelve cgi Module

9.6 The cgi Module

9.7 The urllib Module

**Written Assignment**

Using Operating System Modules

**Discussion Assignment**

Using Modules: Operating System and Inte

**Module 10: Writing Modules**

**Reading Assignments**

Chapters 10, 11, 15 and Chapter 20.4 (optional: the rest of Chapter 20)

**Commentary**

10.1 Modules and the Search Path

10.2 Example of a Module

10.3 Classes and Object-Oriented Programming

10.4 Creating a Class for a Database, Part 1

10.5 Creating a Class for a Database, Part 2

10.6 Working with Databases

10.7 Implementing Operator Overloading

10.8 Inheritance

**Written Assignment**

UserDict and UserList Modules

**Discussion Assignment**

Writing Modules

**Final Project**

**Final Exam**