Introduction to Python

This class introduces the fundamentals of programming in Python, the world's most popular programming language. Join the millions of programmers who are using Python to build web servers, machine learning models, data pipelines, or perform data analysis.

At a glance:

- Appropriate for first time programmers or current programmers who want to learn Python.
- Class meets 2 times per week for 4 weeks; 8 class sessions, 16 hours total class time.
- Focus on foundations with a focus on writing scripts and using Web API's.

Course Objectives

By the end of this class students will be able to:

- Identify, define, and apply core programming concepts, especially:
  - Variables and data types.
  - Control flow mechanisms.
  - Operators and functions.
  - Functions and scope.
- Write and execute Python scripts.
- Use an IDE, debugger, and the terminal to read, write, debug, and execute Python code.
- Write Python code that makes web requests and uses web API's

Prerequisites

- None

Classroom Experience

Our courses have an emphasis on hands-on education. Each class session will follow a similar 3-step pattern:

First, your instructor will provide a walkthrough of a snippet of Python code. This will involve line by line analysis of the code, use of a debugger to examine the state of the code after each line executes, and “micro-exercises” to allow students to test their understanding and apply the new concepts.

Second, students will tackle a longer exercise. These exercises will challenge students to apply the concepts and — as the course progresses — combine new knowledge with previously acquired skills. During these exercises students will receive direct support and feedback from the instructor.

Third, students will see and share solutions to the exercise. One solution will be provided by the instructor. Additionally, students will be invited to share their own solutions. Those who do will receive the gift of additional feedback from their peers and the instructor. Those who do not will still have the opportunity to give feedback, and learn from their peers’ work.

Our courses are keenly focused on a class atmosphere that is:

- Interactive and challenging; wrestling with tough concepts is a cornerstone of learning.
- Welcoming and inclusive; safety and comfort allow learners to be present and engaged.
- Fun and interesting; boredom is the bane of education.

Course Outline

Class 1: Basic Syntax, Variables, Operators, Control Flow

In this class we’ll examine the very basics of Python’s syntax including critical features such as variables, data types, mathematical operators, and control flow.

Class 2: Collections and For Loops

In this class we’ll discuss the core “collection” types in Python (lists and dictionaries), including mechanisms for looping, aggregating, and transforming the contents of the collections.

Class 3: Functions

In this class we’ll introduce functions, which are a core software concept that allows us to write and reuse specific blocks of code.

Class 4: Mutability, Scope, and Error Handling

This class covers subtle and advanced features of the basics we've already introduced. For variables and data types, the concept of mutability. For functions, the concept of scope. Additionally we introduce error handling mechanics.

Class 5: File Management

This class covers a core aspect of using Python as a scripting language, specifically interacting with the file system to read, write, create, and otherwise manipulate files and folders.

Class 6: Package Management & Basic Web Requests (GET)

This class covers the use of Python with 3rd party libraries and modules. First we'll discuss the use of venv to create and manage "virtual environments." Then, we'll use the popular 3rd party "requests" library to make basic HTTP (web) requests from our Python scripts.

Class 7: Web Requests 2 (REST, POST/PUT/PATCH)

This class introduces a common style of web API: the RESTful API. We'll use RESTful API's to further elaborate on the concept of HTTP requests.

Class 8: Web Requests 3 (Web Scraping)

This class introduces a concept that has become incredibly important in web search, AI data collection, and a variety of other use cases: web scraping. We'll use another 3rd party library (BeautifulSoup) along with the requests library to fetch raw data from websites, parse those data to extract specific portions, and store those data to files.