Python for Everyone: Getting Started with Python for Non-Developers

Days: 4

Prerequisites: Before attending this course, students must have:

- Ability to use computers to start programs, open and save files, navigate application menus and interfaces
- Ability to understand logical concepts such as comparisons
- Understand number theory
- Ability to create, understand, and follow structured directions or step-by-step procedures
- Ability to understand and apply abstract concepts to concrete examples

Audience: This is a basic-level introduction to scripting basics and Python for attendees new to programming. The core focus is on “thinking like a programmer” using Python.

This course provides an excellent kick start for users new to Python and scripting or programming, enabling them to use basic Python skills on the job in a variety of ways. This in a basic-level Python course geared for student who are new to Software Development and use Python in projects, or system administrators and web site administrators who want to use Python to support their server installations, as well as anyone else who wants to automate or simplify common tasks with the use of Python scripts.

Description: This hands-on course teaches students from non-development backgrounds the basics of programming and scripting using Python. This course leads through programming essentials, working through the fundamentals of writing and running Python scripts to advanced features such as file operations, regular expressions, working with binary data, and extensive functionality of Python modules.

This course provides an excellent kick start for users new to Python and scripting, enabling them to use basic Python skills on the job in a variety of ways. Students can apply the course skills to use Python in basic web development projects or automate or simplify common tasks with the use of Python scripts. The course also serves as a solid primer course / foundation for continued Python study in support for next level web development with Python, Python for data science / machine learning or Python for systems admin or networking support.

Course Objectives: Throughout the course students will learn to write essential Python scripts using the most current and efficient skills and techniques. This course is about **50% hands-on lab to 50% lecture ratio**, combining engaging, informed instructor presentations, demonstrations and discussions with extensive machine-based student labs and practical project work. Working within in an engaging, hands-on learning environment, guided by our expert instructor, students will learn to:

- Create working Python scripts following best practices
- Use python data types appropriately
- Read and write files with both text and binary data
- Search and replace text with regular expressions
- Get familiar with the standard library and its work-saving modules
- Use lesser-known but powerful Python data types
- Create "real-world", professional Python applications
- Work with dates, times, and calendars
- Know when to use collections such as lists, dictionaries, and sets
- Understand Pythonic features such as comprehensions and iterators
- Write robust code using exception handling

Python for Everyone: Getting Started with Python for Non-Developers

OUTLINE:

SESSION 1: AN OVERVIEW OF PYTHON

- What is python?
- Python Timeline
- Advantages/Disadvantages of Python
- Getting help with pydoc

LESSON 1: AN OVERVIEW OF SOFTWARE DEVELOPMENT

- Thinking as a developer
- Pseudocode as a design tool
- Analysis – Understanding the problem
- Design – creating the solution
- Let's Get Hands-On: Scripting and programming basics

LESSON 2: AN OVERVIEW OF PYTHON

- What is python?
- Python Timeline
- Advantages/Disadvantages of Python
- Getting help with pydoc

LESSON 3: THE PYTHON ENVIRONMENT

- Starting Python
- Using the interpreter
- Running a Python script
- Python scripts on Unix/Windows
- Editors and IDEs

LESSON 4: GETTING STARTED

- Using variables
- Builtin functions
- Strings
- Numbers
- Converting among types
- Writing to the screen
- Command line parameters

LESSON 5: FLOW CONTROL

- About flow control
- White space
- Conditional expressions
- Relational and Boolean operators
- While loops
- Alternate loop exits

LESSON 6: ARRAY TYPES

- About array types (AKA sequences)
- Lists and list methods
- Tuples
- Indexing and slicing
- Iterating through a sequence
- Nested sequences
- Sequence functions, keywords, and operators
- List comprehensions
- Generator Expressions

LESSON 7: WORKING WITH FILES

- File overview
- Opening a text file
- Reading a text file
- Writing to a text file

LESSON 8: DICTIONARIES AND SETS

- About dictionaries
- Creating dictionaries
- Iterating through a dictionary
- About sets
- Creating sets
- Working with sets

LESSON 9: FUNCTIONS

- Defining functions
- Returning values
- Parameters
- Global and local scope

LESSON 10: SORTING

- The sorted() function
- Alternate keys
- Lambda functions
- Sorting collections
- Using operator.itemgetter()
- Reverse sorting

Python for Everyone: Getting Started with Python for Non-Developers

LESSON 11: ERRORS AND EXCEPTION HANDLING

- Syntax errors
- Exceptions
- Using try/catch
- Handling multiple exceptions
- Ignoring exceptions

LESSON 12: MODULES AND PACKAGES

- The import statement
- Module search path
- Creating Modules
- Using packages
- Function and Module aliases

LESSON 13: INTRODUCTION TO OBJECT ORIENTED CLASSES

- About o-o programming
- Defining classes
- Constructors
- Methods
- Instance data
- Properties
- Class methods and data