Python for Networking
& Systems Administrators (SysAdmin)

Days: 4

Prerequisites: Students should have basic development experience in any programming language, along with a working, user-level knowledge of Unix/Linux, Mac, or Windows.

Audience: This introductory-level Python course is appropriate for system administrators and web site administrators, new to Python, 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: A core component of our Python Skill Journey series, Python for Networking / Systems Administrators teaches you basic Python scripting skills, and then how to leverage Python to automate your day-to-day administrative and networking tasks across a set of distributed clients. Working in a hands-on lab environment, you'll learn solid Python scripting essentials, as well as how to perform and work with more advanced features such as file operations, regular expressions, working with binary data. You'll also explore using the extensive functionality of Python modules, with an emphasis on network-focused modules such as SSH, Git, and RESTful services.

This course provides you with real hands-on experience exploring and working with Python, so you gain real basic level scripting skills, learning by doing, not just getting a quick overview of syntax and grammar. You'll leave this course able to use basic Python to complete these automated tasks in the real world, on the job, right after class.

Course Objectives: Working within an engaging, hands-on learning environment, guided by our expert Python practitioner, 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
- Automate network administrative tasks across distributed clients using SSH, REST, and More

OUTLINE:

LESSON 1: AN OVERVIEW OF PYTHON

- What is Python?
- About Interpreted Languages
- Advantages of Python
- Disadvantages of Python
- How to get Python
- Which version of Python?
- The end of Python 2
- Getting Help
- One day on Dagobah

LESSON 2: THE PYTHON ENVIRONMENT

- Starting Python
- If the interpreter is not in your PATH
- Using the interpreter
- Trying out a few commands
- Running Python scripts (explicit)
- Running Python scripts (implicit)
- Using pydoc
- Python Editors and IDEs

Python for Networking
& Systems Administrators (SysAdmin)

LESSON 3: GETTING STARTED

- Keywords and Builtins
- Variable typing
- Strings
- Single-delimited string literals
- Triple-delimited string literals
- Raw string literals
- Unicode characters
- String operators and methods
- String Methods
- Numeric literals
- Math operators and expressions
- Converting among types
- Writing to the screen
- String Formatting
- Legacy String Formatting
- Command line parameters
- Reading from the keyboard

LESSON 4: FLOW CONTROL

- What's with the white space?
- if and elif
- Conditional Expressions
- Relational Operators
- Boolean operators
- while loops
- Alternate ways to exit a loop

LESSON 5: ARRAY TYPES

- About array types
- Lists
- Tuples
- Indexing and slicing
- Iterating through a sequence
- Unpacking tuples
- Nested sequences
- Functions for all sequences
- Using enumerate()
- Operators and keywords for sequences
- The range() function
- List comprehensions
- Generator Expressions

LESSON 6: WORKING WITH FILES

- Text file I/O
- Opening a text file
- The with block
- Reading a text file
- Writing to a text file

LESSON 7: DICTIONARIES

- About dictionaries
- When to use dictionaries?
- Creating dictionaries
- Getting dictionary values
- Iterating through a dictionary
- Reading file data into a dictionary
- Counting with dictionaries
- About sets
- Creating Sets
- Working with sets

LESSON 8: FUNCTIONS

- Defining a function
- Returning values
- Function parameters
- Variable scope

LESSON 9: SORTING

- Sorting Overview
- The sorted() function
- Custom sort keys
- Lambda functions
- Sorting nested data
- Sorting dictionaries
- Sorting in reverse
- Sorting lists in place

LESSON 10: ERRORS AND EXCEPTION HANDLING

- Syntax errors
- Exceptions
- Handling exceptions with try
- Handling multiple exceptions
- Handling generic exceptions
- Ignoring exceptions
- Using else
- Cleaning up with finally

LESSON 11: USING MODULES

- What is a module?
- Creating Modules
- The import statement
- Where did __pycache__ come from?

Python for Networking
& Systems Administrators (SysAdmin)

- Module search path
- Packages
- Example
- Module Aliases
- When the batteries aren't included

LESSON 12: AN INTRODUCTION TO PYTHON CLASSES

- About o-o programming
- Defining classes
- Constructors
- Instance Methods
- Properties
- Class methods and data
- Static Methods
- Private methods
- Inheritance
- Untangling the nomenclature

LESSON 13: REGULAR EXPRESSIONS

- Regular Expressions
- RE Syntax Overview
- Finding matches
- RE Objects
- Compilation Flags
- Groups
- Special Groups
- Replacing text
- Replacing with a callback
- Splitting a string

LESSON 14: NETWORK PROGRAMMING

- Grabbing a web page
- Consuming Web services
- HTTP the easy way
- sending e-mail
- Email attachments
- Remote Access
- Copying files with Paramiko

LESSON 15: SOCKETS

- Sockets
- Socket options
- Server concepts
- Client concepts
- Application protocols
- Forking servers

LESSON 16: MULTIPROGRAMMING

- Multiprogramming
- What Are Threads?
- The Python Thread Manager
- The threading Module
- Threads for the impatient
- Creating a thread class
- Variable sharing
- Using queues
- Debugging threaded Programs
- The multiprocessing module
- Using pools
- Alternatives to multiprogramming

LESSON 17: EFFICIENT SCRIPTING

- Running external programs
- Parsing arguments
- Creating filters to read text files
- Logging

LESSON 18: SERIALIZING DATA: XML, XPATH, JSON, CSV

- About XML
- Normal Approaches to XML
- Which module to use?
- Getting Started With ElementTree
- How ElementTree Works
- Elements
- Creating a New XML Document
- Parsing An XML Document
- Navigating the XML Document
- Using XPath
- About JSON
- Reading JSON
- Writing JSON
- Customizing JSON
- Reading CSV data
- Nonstandard CSV
- Using csv.DictReader
- Writing CSV Data
- Pickle