

# Python Essentials for Networking & Systems Administration - TTPS4824

Master the power of Python to streamline and automate your networking and administrative tasks with this hands-on, results-driven course. Designed for real-world application, this training equips you with the skills to write efficient, professional-grade scripts, manage files and data seamlessly, and automate complex workflows. You'll learn to work with Python's powerful libraries and tools, such as SSH and RESTful services, while building a solid foundation in best practices for scripting and problem-solving. Through engaging instruction and practical labs, you'll also develop the ability to handle text processing, work with binary data, and create robust, maintainable solutions. Whether managing distributed systems or optimizing daily operations, you'll leave this course ready to deliver results immediately and confidently.

## What You'll Learn

### Overview

Python for Networking and Systems Administrators is a four-day hands-on course that equips you with essential Python scripting skills to streamline and automate administrative and networking tasks across distributed systems. Designed for hands-on learning, this course focuses on practical applications, empowering you to solve real-world challenges efficiently. You'll explore Python's versatile capabilities, from data manipulation to working with network-focused modules like SSH and RESTful services. With an emphasis on best practices and professional-grade scripting, you'll gain the skills to confidently develop robust, maintainable solutions for your day-to-day operations.

This course combines expert-led instruction, interactive demonstrations, and extensive hands-on labs to ensure a deep understanding of Python's practical applications. By the end of the training, you'll not only master scripting essentials but also learn how to harness Python's power to automate complex workflows, enhance productivity, and simplify repetitive tasks. Whether you're managing files, optimizing network operations, or handling distributed clients, you'll be equipped with the knowledge to deliver results immediately in your role.

## Objectives

Working in a hands-on learning environment led by our expert instructor you will explore:

- Develop Python Scripts Using Best Practices. Write efficient, professional-grade Python scripts to automate administrative and networking tasks.
- Leverage Core Python Data Structures. Utilize lists, dictionaries, sets, and other Python data structures to solve real-world problems effectively.
- Automate File and Data Management. Perform advanced file operations, handle binary data, and process large datasets with ease.
- Implement Regular Expressions and Parsing Techniques. Search, replace, and manipulate text data using powerful regular expressions and parsing tools.
- Automate Networking Tasks. Use Python modules to perform SSH operations, interact with RESTful services, and manage distributed clients.
- Build Resilient and Maintainable Scripts: Apply exception handling, modular programming, and other techniques to create robust, reusable Python solutions.

## Audience

This introductory-level Python course is appropriate for advanced users, 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. Students should have basic development experience in any programming language, along with a working, user-level knowledge of Unix/Linux, Mac, or Windows.

## Pre-Requisites

{{code}}            {{title}}

## Agenda

**NOTE:** This agenda is based on our standard course offering, evolved from typical industry uses and trends. We will collaborate with you to tune this course to target the skills you need most. Topics, agenda and labs are subject to change and may adjust during live delivery based on audience skill level, interests and participation.

### 1. The Python Environment

- Starting Python
- If the interpreter is not in your PATH
- Using the interpreter

- Trying out a few commands
- Running Python scripts
- Getting help
- Python Editors and IDEs
- 2. Variables and Values**
- Using variables
- Keywords and Builtins
- Variable typing
- Strings
- String operators and methods
- Numeric literals
- Math operators and expressions
- Converting among types
- 3. Basic input and output**
- Writing to the screen
- String Formatting
- Legacy String Formatting
- Command line parameters
- Reading from the keyboard
- 4. Flow Control**
- About 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
- 5. Array types**
- Lists
- Tuples and unpacking
- Indexing and slicing
- Aterating through a sequence
- Functions for all sequences
- The range() function
- List comprehensions
- Generator Expressions
- 6. Working with Files**
- Text file I/O

- Opening a text file
  - The with block
  - Reading a text file
  - Writing to a text file
- 7. Dictionaries and sets**
- When to use dictionaries?
  - Creating dictionaries
  - Getting dictionary values
  - Alterating through a dictionary
  - Reading file data into a dictionary
  - Counting with dictionaries
  - Creating Sets
  - Working with sets
- 8. Functions, modules, packages**
- Defining a function
  - Returning values
  - Function parameters
  - Variable scope
  - Creating Modules
  - The import statement
  - Where did `__pycache__` come from?
  - Module search path
  - Packages
- 9. 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
- 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
- 11. Efficient Scripting**
  - Running external programs
  - Parsing arguments
  - Creating filters to read text files
  - Logging
- 12. 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
- 13. Binary data**
  - str vs bytes
  - Binary files
  - Structured binary data
  - Bitwise operations
- 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
- 15. Sockets**
  - Sockets
  - Socket options
  - Server concepts
  - Client concepts
  - Application protocols

- Forking servers

## **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

## **17. 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

### **Bonus Topics / Time Permitting**

These topics will be included in your course materials but may or may not be presented during the live class depending on the pace of the course and attendee skill level and participation.

## **18. Sorting**

- Sorting Overview

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

## Follow On Courses

TTPS4850	Next-Level Python Programming (with Advanced Python Topics)
TTPS4876	Next-Level (Intermediate) Python for Data Science and /or Machine Learning
TTPS4878	Hands-On Data Analysis with Panda
TTPS4879	Hands-On Predictive Analytics with Python

## Related Courses

TTPS4800	Introduction to Python Programming Basics
TTPS4803	Introduction to Programming with Python (for Non-Developers)
TTPS4820	Mastering Python Programming Boot Camp
TTPS4872	Quick Start to Python for Data Science Primer: A Hands-on Technical Overview
TTPS4873	Fast Track to Python for Data Science and/or Machine Learning
TTPS4874	Applied Python for Data Science and Engineering
TTPS4894	Python Security   Introduction to Python Programming for Security Analysts & Professionals
TTPS4824	Python Essentials for Networking & Systems Administration

## Attend a Course

Please feel free to Register Online or call 844-475-4559 toll free to connect with our Registrar for assistance. If you ever need additional date options, please [contact us](#) for scheduling.