Python for Networking & Systems Administration (TTPS4824)

Category: Software Development & Programming | **Vendor:** Trivera Tech

Duration: 32.00 hours (4 days) **26.0 CPD Hours** **Rating:** ⭐ 4.6 (5,878 reviews)

Course Information

Delivery Format: Instructor Led - Online

Course Overview

Geared for network administrators looking to automate administrative tasks across a set of distributed clients, this course is an introductory and beyond-level practical, hands-on Python course that leads you from the basics of writing and running Python scripts to more advanced features such as file operations, regular expressions, working with binary data, and using the extensive functionality of Python modules with a focus on network-focused modules such as SSH, Git, and RESTful services.

About This Course

Geared for network administrators looking to automate administrative tasks across a set of distributed clients, this course is an introductory and beyond-level practical, hands-on Python course that leads you from the basics of writing and running Python scripts to more advanced features such as file operations, regular expressions, working with binary data, and using the extensive functionality of Python modules with a focus on network-focused modules such as SSH, Git, and RESTful services.

Who Should Attend

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.

Prerequisites & Entry Requirements

General Prerequisites:

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.

Learning Outcomes

Upon successful completion of this course, participants will be able to:

This course is approximately 50% hands-on, combining expert lecture, real-world demonstrations and group discussions with machine-based practical labs and exercises. Our engaging instructors and mentors are highly experienced practitioners who bring years of current 'on-the-job' experience into every classroom.

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

Automate network administrative tasks across distributed clients using SSH, REST, and More

Detailed Course Outline

An Overview of Python

- What is Python
- The Birth of 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

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

Getting Started

- Using variables
- 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

Flow Control

- About flow control
- Whats with the white space
- if and elif
- Conditional Expressions
- Relational Operators
- Boolean operators
- while loops
- Alternate ways to exit a loop

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

Working with Files

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

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

Functions

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

Sorting

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

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

Using Modules

- What is a module
- Creating Modules
- The import statement
- Where did __pycache__ come from
- Module search path
- Packages
- Example
- Module Aliases
- When the batteries arent included

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

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

Network Programming

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

Sockets

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

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

Efficient Scripting

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

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

, Chapter 1: An Overview of Python

What is Python

The Birth of 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

Chapter 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

Chapter 3: Getting Started

Using variables

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

Chapter 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

Chapter 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

Chapter 6: Working with Files

Text file I/O

Opening a text file

The with block

Reading a text file

Writing to a text file

Chapter 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

Chapter 8: Functions

Defining a function

Returning values

Function parameters

Variable scope

Chapter 9: Sorting

Sorting Overview

The sorted() function

Custom sort keys

Lambda functions

Sorting nested data

Sorting dictionaries

Sorting in reverse

Sorting lists in place

Chapter 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

Chapter 11: Using Modules

What is a module

Creating Modules

The import statement

Where did __pycache__ come from

Module search path

Packages

Example

Module Aliases

When the batteries aren't included

Chapter 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

Chapter 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

Chapter 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

Chapter 15: Sockets

Sockets

Socket options

Server concepts

Client concepts

Application protocols

Forking servers

Chapter 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

Chapter 17: Closures

What is a closure

Why do I need one

How can I make one

Factory Function

Using functools.partial()

Chapter 18: Serializing Data: XML, 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

Additional Course Details

Nexus Humans Python for Networking & Systems Administration (TTPS4824) training program is a workshop that presents an invigorating mix of sessions, lessons, and masterclasses meticulously crafted to propel your learning expedition forward.

This immersive bootcamp-style experience boasts interactive lectures, hands-on labs, and collaborative hackathons, all strategically designed to fortify fundamental concepts.

Guided by seasoned coaches, each session offers priceless insights and practical skills crucial for honing your expertise. Whether you're stepping into the realm of professional skills or a seasoned professional, this comprehensive course ensures you're equipped with the knowledge and prowess necessary for success.

While we feel this is the best course for the Python for Networking & Systems Administration (TTPS4824) course and one of our Top 10 we encourage you to read the course outline to make sure it is the right content for you.

Additionally, private sessions, closed classes or dedicated events are available both live online and at our training centres in Dublin and London, as well as at your offices anywhere in the UK, Ireland or across EMEA.

Frequently Asked Questions

Q: What delivery options are available for Python for Networking & Systems Administration (TTPS4824)?

We offer multiple delivery formats:

- Live Instructor-Led Classroom Online (Virtual/Live Online)
- Traditional Instructor-Led Classroom Training (ILT)
- On-site delivery at your offices anywhere in United Kingdom
- Private dedicated courses customized for your team

Q: How many CPD hours does this course provide?

The 4-day Python for Networking & Systems Administration (TTPS4824) course provides up to 26.0 CPD hours of structured learning. CPD certificates can be provided upon request.

Q: What is the duration of the Python for Networking & Systems Administration (TTPS4824) training?

The training takes place over 4 day(s), with each day lasting approximately 32.00 hours including breaks for lunch and refreshments.

Q: Do you provide corporate training for Python for Networking & Systems Administration (TTPS4824)?

Yes, we provide corporate training, dedicated training, and closed classes for Python for Networking & Systems Administration (TTPS4824). Training can take place anywhere in United Kingdom including London, Manchester, Birmingham, Edinburgh, or live online allowing teams from across United Kingdom or internationally to attend.

Q: What related terms do people search for?

Popular related searches include: Python Networking

Q: Why choose Nexus Human for Python for Networking & Systems Administration (TTPS4824)?

Nexus Human is recognized as one of the leading training providers. Our trainers have won multiple awards including:

- Small Firms Best Trainer Award
- National Training Partner of the Year (Ireland) - Multiple Years
- Global Top 30 Instructor Awards (2012, 2019, 2021)
- Tech Excellence Award Nominations
- Learning Performance Institute (LPI) External Training Provider Sponsor 2024

Q: Are there any discount codes available?

Yes! Use discount code **PENPAL5** when booking your Python for Networking & Systems Administration (TTPS4824) training. Please note that only one discount code can be used per booking and cannot be combined with other special offers.

Nexus Human

Professional Training & Development

Email: info@nexushuman.com

Website: www.nexushuman.com

Phone: +353 1 889 8700 (Ireland) | +44 20 8142 8961 (UK)

© 2026 Nexus Human. All rights reserved. This brochure was generated on 13/11/2026.