

Advanced Python Scripting

PTN-108

Duration: 5 days; Instructor-led

Time: 9:00 AM – 5:00 PM

ABOUT THIS COURSE

Python Fundamentals to Advanced for the duration of 4 days, covering the language references, explains object-oriented as well as functional programming techniques, error handling, modules, many of the Python libraries, and best practices. All concepts are explained through hands-on examples and exercises. The participants would be learning by doing.

OBJECTIVES

This is a hands-on course, designed to help the developers to speedup in Python, as quickly as possible. The participant's in this course, will experience the following:

- Architecture
- Input and output
- Lists, Tuples, set & Dictionaries
- Decision Making & Loops
- Error handling.
- Function
- Object-oriented features. Classes & Objects
- Creating and using libraries and packages.
- Regular Expressions
- Database access – SQLite
- CSV, JSON handling
- Network Programming
- Multi-threaded Programming

PREREQUISITES

The participants should have prior programming experience and should be familiar with basic programming constructs.

AUDIENCE

This course is designed for developers, system administrators, and test engineers, who wish to develop, automate, and test applications and systems using Python.

COURSE CONTENTS

Module 1: Python Introduction

- What's Python?
- Why do people use Python?
- Some quotable quotes
- A Python history lesson
- Advocacy News
- What's Python good for?
- What's Python not good for?
- The features list
- Python portability
- Summary

Module 2: Using the Interpreter

- Python's Interactive Prompt
- Scripting
- Program Execution Model
- Program Architecture: modules
- How to run Python programs
- Using Python IDEs

Module 3: Python Scripting

- Python Scripts in Linux/Unix & Windows
- Whitespace Significance
- Line Termination
- Comments in Python
- Basic Output Generation
- Simple User Input
- Python Modules
- Module Search Paths
- Determining the System Search Path
- input()

Module 4: Working with Variables in Python

- Python Variables
- Naming Conventions & Rules
- Types as Objects
- Variable References & Garbage Collection
- Sequence Types
- Membership Statements
- List Iteration
- Sequence Assignments
- Mutable vs Immutable Objects
- Multi Target Assignments

Module 5: Numeric Operations in Python

- More About Python's Numeric Types
- Numeric Tools
- The Decimal Module
- Operator
- Arithmetic
- Logical
- Relational
- Bitwise
- Special Operators
- Operator Precedence

Module 6: Decision making & Looping

- Comparison Operations
- The if Statement
- The if Ternary Expression
- The while Loop
- The for Loop

Module 7: Debugging

- Dealing with syntax errors
- Dealing with runtime errors
- Dealing with logical errors
- Using unit tests

Module 8: Python Strings

- Generating Strings in Python
- Immutable
- Common String Methods
- Type Conversion in Python
- Formatting String Output
- Format Specifier
- Variable Substitution
- String Indexing
- String Slicing
- String Iteration

Module 9: Python's Tuples

- Immutable
- Common Tuples Methods
- Tuples Operations
- Tuples Indexing
- Tuples Slicing
- Tuples Iteration
- Multi-Dimensional Tuples (Matrices)

Module 10: Python's Lists

- Common List Methods
- The range() Function
- List Operations
- String Indexing
- String Slicing
- String Iteration
- Multi-Dimensional Lists (Matrices)

Module 11: Python List Comprehension

- Basic List Comprehensions
- Compound List Comprehensions

Module 12: Python set data type

- Understanding & using set data type

Module 13: Python Dictionaries

- Python Dictionaries
- Assigning Values to Dictionaries
- Dictionary Methods
- Dictionaries vs Lists & Tuples
- Dictionary Indexing
- Dictionary Iteration

Module 14: Basic Input/Output with Files

- Opening Files
- Working with Files
- Controlling Output Location

Module 15: Creating Python Functions

- Function Basics
- Defining Functions
- Function Polymorphism
- Argument Defaults
- Lambdas
- Local Variables
- Understanding `__builtin__`
- Preventing Variable Modifications
- Argument Matching Methods
- Keyword Argument Methods

Module 16: Classes and Objects

- Introduction to OOP using python
- Classes and class attributes
- Instances and instance attributes
- Binding and method invocation
- Composition, Sub-classing and Derivation
- Inheritance
- Built-in functions for classes, instances and other objects
- Privacy and Delegation
- An overview of built-in python classes and modules

Module 17: Modules & Packages

- Module Basics
- Packages
- Package Creation and Importing
- Using `__all__` and `_Variables`
- Using `__name__`
- Using third party modules

Module 18: Exceptions

- About Exceptions
- Learning how exceptions work in Depth
 - Handling exceptions
 - Raising exceptions
 - Catching exceptions
- Python's Default Exception Handler
- Using Try/Except/Else/Finally Exceptions
- Generating User Defined Exceptions
- Using Asserts
- Exception Classes

Module 19: Regular Expression in Python

- Using the re module
- Searching with regular expressions
- Replacing with regular expressions
- Reusing regular expressions with `re.compile`
- The match Function
- The search Function
- Regular-expression patterns
- Backreferences
- Translation

Module 20: SQLite Database access

- Creating Database Connection
- Creating Database & Table
- INSERT Operation
- READ Operation
- Update Operation
- DELETE Operation
- COMMIT Operation
- ROLLBACK Operation

Module 21: Network Programming

- What is Socket?
- The socket module
- Server Socket Methods
- Client Socket Methods
- Developing Server & Client scripts

Module 22: Multi-threaded Programming

- Starting a New Thread
- The Threading Module
- Creating Thread Using Threading Module
- Synchronizing Threads

Module 23: CSV files Processing**Module 24: JSON Processing****Module 25: Logging & Warning modules****Module 26: Converting Python script to exe**