

Course Content

Course Description:

This four-day course leads the student 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. Extra emphasis is placed on features unique to Python, such as tuples, array slices, and output formatting. This is a hands-on programming class. All concepts are reinforced by informal practice during the lecture followed by graduated lab exercises. Python Programming is a practical introduction to a working programming language, not an academic overview of syntax and grammar. Students will immediately be able to use Python to complete tasks in the real world.

Target Student:

Users who want to learn Python and use it for application development, system administration, or just to automate tasks in a simple, yet powerful way.

Prerequisites:

Working/user level knowledge of an operating system such as Linux, Windows, or MacOS. Basic skill with at least one other programming language is desirable.

Topics:

About this Course

- Welcome
- Classroom etiquette
- Course outline
- Student files
- Extracting the student files
- Examples
- Lab exercises
- Appendices

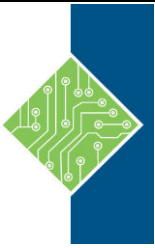
An Overview of Python

- What is Python?
- The Birth of Python
- Python Timeline
- About Interpreted Languages
- Advantages of Python
- Disadvantages of Python

- How to Get Python
- How to Get Python?
- The end of 2.x
- Getting Help
- Pydoc
- Using Pydoc

The Python Environment

- Starting Python
- If the Interpreter is Not in Your PATHs
- Using the Interpreter
- Trying Out a Few Commands
- The help() Command
- Running a Python Script\
- Python Scripts on Unix
- Python Scripts on Windows



Introduction to Python 3.x

Course ID #: 1411-900-3.x-W

Hours: 28

- Python Editors and IDEs

Getting Started

- Using Variables
- Keywords
- Built-in Functions
- Variable Typing
- Strings
- Single-delimited String Literals
- Triple-delimited String Literals
- Raw String Literals
- Unicode Characters
- String Operators and Methods
- Numeric Literals
- Math Operators and Expressions
- Converting Among Types
- Writing to the Screen
- String Formatting
- Command Line Parameters
- Reading From the Keyboard

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

Sequences

- About Sequences
- Lists
- Tuples
- Indexing and Slicing
- Iterating Through a Sequence
- Using Enumerate()
- Functions for All Sequences
- Keywords and Operators for All Sequences
- The Range() Function

- Nested Sequences
- 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
- Non-Delimited (Raw) Data

Dictionaries and Sets

- About Dictionaries
- When to Use Dictionaries
- Creating Dictionaries
- Getting Dictionary Values
- Iterating Through a Dictionary
- Reading File Data into a Dictionary
- Counting with a Dictionary
- About Sets
- Creating Sets
- Working with Sets

Functions

- Defining a Function
- Function Parameters
- Returning Values
- Variable Scope

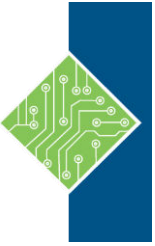
Sorting

- Sorting
- The Sorted() Function
- Alternate Keys
- Lambda Functions
- Sorting Nested Data
- \Sorting Dictionaries
- Sorting in Reverse
- Sorting Lists in Place

Introduction to Python 3.x

Course ID #: 1411-900-3.x-W

Hours: 28



Using Modules

- Regular Expressions
- RE Syntax Overview
- RE Objects
- Searching for Patterns
- Matching Without Re Objects
- Compilation Fags
- Grouping
- Special Groups
- Replacing Text
- Splitting a String

Using the Standard Library

- The Sys Module
- Interpreter Information
- STDIO
- Launching External Programs
- Paths, Directories, and Filenames
- Walking Directory Trees
- Grabbing Web Pages
- Sending E-Mail
- Math Functions
- Random Values
- Dates And Times
- Zipped Archives

An Introduction to Python Classes

- About O-O Programming
- Defining Classes
- Initializers
- Instance Methods
- Properties
- Class Methods and Data
- Static Methods
- Private Methods
- Inheritance
- Untangling the Nomenclature