
Python Programming 1

Duration : 4 Days

Overview

Python is an object oriented rapid development language deployed in many scenarios in the modern world.

This Python Programming 1 course is designed to give delegates the knowledge to develop and maintain Python scripts using the current version (V3) of Python.

There are many similarities between Python V2 and Python V3. The skills gained on this course will allow the delegate to develop their own skills further using Python V2 or V3 to support the development and maintenance of scripts.

The Python Programming 1 course comprises sessions dealing with syntax, variables and data types, operators and expressions, conditions and loops, functions, objects, collections, modules and packages, strings, pattern matching, dates, exception handling, files, and databases.

Exercises and examples are used throughout the course to give practical hands-on experience with the techniques covered.

Target Audience

The Python Programming 1 course is aimed at anyone who wants to learn Python as a first language, and developers/engineers who want to migrate to Python from another language, particularly those with little or no object-oriented knowledge.

Skills Gained

- Writing and testing simple scripts
- Representing data using built-in and custom data types
- Building expressions
- Building conditional and iterative statements
- Declaring and calling functions
- Using objects
- Creating and manipulating collections including lists, tuples, and dictionaries
- Creating and manipulating strings
- Creating modules and packages, and using third-party libraries
- Pattern matching
- Working with date and time objects
- Handling exceptions
- Reading from and writing to files and databases
- Coding in an OOP manner

Prerequisites

Familiarity with programming concepts is useful, this knowledge can be obtained by attendance on the Introduction to Programming course.

Objectives

This course aims to provide the delegate with the knowledge to be able to produce Python scripts and applications that exploit all core elements of the language including variables, expressions, selection and iteration, functions, objects, collections, strings, modules, pattern matching, exception handling, I/O, and classes.



Course Content

DAY 1

Session 1: GETTING STARTED

- About Python
- Python versions
- Python documentation
- Python runtimes
- Installing Python
- The REPL shell
- Python editors

Session 2: PYTHON SCRIPTS & SYNTAX

- Script naming
- Comments
- Docstring
- Statements
- The backslash
- Code blocks
- Whitespace
- Console IO (to enable the writing of simple programs)
- A first Python program
- Script execution

Session 3: VARIABLES & DATA TYPES

- Literals
- Identifiers
- Assignment
- Numbers (bool, int, float, complex)
- Binary, octal, and hexadecimal numbers
- Collections (str, list, tuple, set, dict)
- None
- Implicit and explicit type conversion (casting)
- The type function

Session 4: OPERATORS & EXPRESSIONS

- Arithmetic Operators
- Assignment Operators
- Comparison Operators
- Logical Operators
- Membership Operators
- Bitwise Operators
- Identity Operators

Session 5: CONDITIONS & LOOPS

- Conditional statements (if, elif, else)
- Short hand if/elif else
- Python's alternative to the ternary operator
- Iterative statements (while, for, else)
- The range function
- Iterating over a list
- Break
- Continue
- Nested conditional/iterative statements

DAY 2

Session 6: FUNCTIONS

- Declaration
- Invocation
- Default values for parameters
- Named arguments
- args and kwargs
- Returning multiple values
- Nested functions
- Functions as data
- Introduction to lambda expressions
- Variable scope
- The pass keyword

Session 7: OBJECTS

- About objects
- Attributes and the dot notation
- Data attributes
- Method attributes
- The dir function
- Magic methods and operator overloading
- Mutability
- Variables: references to objects
- Variable assignment (again)
- Pass by reference
- The id function and identity operator

Session 8: LISTS

- About lists
- List syntax including slicing
- Getting and setting list elements
- Iterating over a list
- Checking for the presence of a value
- The len function
- List methods incl. append, insert, remove, pop, clear, copy, sort, reverse etc.
- The del keyword
- Appending to and combining lists
- List comprehension

Session 9: TUPLES

- About tuples
- Tuple syntax
- Getting tuple elements including unpacking
- Iterating over a tuple
- Checking for the presence of a value
- The len function
- Appending to and combining tuples

Session 10: SETS

- About Sets
- Dictionary syntax
- Creating, adding and removing set elements
- Iterating over a set
- Membership Testing
- Sorting
- Copying
- Set methods incl. union, intersection, difference, symmetric_difference etc.

DAY 3

Session 12: STRINGS

- About strings
- String syntax including slicing
- Escape characters
- Triple-quoted strings
- Concatenation
- Placeholders
- The format method
- Other methods e.g. endswith, find, join, lower, replace, split, startswith, strip, upper etc.
- A string as a list of bytes

Session 13: MODULES & PACKAGES

- About modules
- The module search path
- Importing modules
- Namespaces
- Importing module objects
- The import wildcard
- Aliases
- Importing within a function
- Executable modules
- Reloading a module
- About packages
- Importing packaged modules
- Importing packaged module objects
- Package initialisation
- Subpackages
- Referencing objects in sibling packages
- The Standard Library
- Installing modules and packages using pip

Session 14: PATTERN MATCHING

- About regular expressions
- Regular expression special characters
- Raw strings
- About the re module
- re module functions incl. match, search, findall, full match, split, sub

DAY 4

Session 15: DATES

About the datetime module
datetime object attributes
Creating a datetime object
Date arithmetic
Formatting dates

Session 16: EXCEPTION HANDLING

About exceptions and exception handling
Handling exceptions (try, except, else, finally)
Exception types
The exception object
Raising exceptions
Custom exception types

Session 17: FILES & THE FILESYSTEM

The open function
Methods for seeking (seekable, seek)
Methods for reading from a file (readable, read, readline, readlines)
Iterating over a file
Methods for writing to a file (writable, write, writelines)
Introduction to context managers
File parsing for files of type CSV, XML, JSON, YAML
About the os module
os module functions incl. getcwd, listdir, mkdir, chdir, remove, rmdir etc.

Session 18: DATABASES

The DB-API
DP-API implementations
Establishing a connection
Creating a cursor
Executing a query
Fetching results
Transactions
Inserting, updating, and deleting records