

Python Course Content

Course Description:

Python is a high-level, interpreted, general-purpose and object-oriented programming language with effective semantics. Python is designed to read and write easily to provide desktop GUI applications, websites, web applications, and system scripting. Many developers use Python to build productivity tools, desktop apps, games, etc. Python is a top-picked programming language that used in many applications to build simple scripts to complex software applications because of its limitless features like simplicity, readability, efficiency, wide resources, community, etc. Many tech giant companies like Google, Pinterest, Instagram, Dropbox, etc are using Python language to create amazing things from scratch.

Python online training at Hachion will provide certification with placement assistance. Our structured curriculum helps you to gain in-depth knowledge of python language. Students can easily learn the language as it has a simple syntax and few keywords. The course includes all the basic and advanced topics to have a solid foundation in web developing domain. It is a very easy learning language for beginners to work in the software field. Learning Python language will give you an extra edge in your career in the software domain.

Course Content:

Installing & Running Python

- Python 2.7 vs Python 3
- Local Environment Setup
- Installing Python on different platforms (Windows and Linux)
- Python Interpreter and Python Interactive Shell
- Python IDE(Pydev, Pycharm, VIM)

Introduction to Python

- Python Overview
- History Of Python
- CPython, Jython, PyPy
- Python Features
- Areas Of Application of Python
- Understanding More About Python
- Writing your First Python Program
- Interactive Mode Programming
- Script Mode Programming
- Dir and help: Getting help from the Python interpreter

Python Syntax, Keywords, and Operators

- Python Identifiers
- Various Operators and Operators Precedence

- Reserved Words, Lines and Indentation
- Multi-Line Statements, Quotation in Python
- Comments in Python, Using Blank Lines
- Command Line Arguments
- Python Input/Output: Using the Print Function
- Getting Input from User
- Python Basic Data Types And Variables

Expressions, Statements, Variables

- Working With Numbers
- Working With Booleans
- Working with Strings
- String types and formatting
- Program to find duplicate characters in a String.
- Program to reverse a string
- Program to check if String is Palindrome
- Program to remove a newline in Python
- Mini Project: Dice Rolling Simulator

Python Data Types: List, Tuples, Dictionaries

- Python Lists, Tuples, Dictionaries
- Accessing Values
- Basic Operations
- Indexing, Slicing, and Matrixes
- Built-in Functions & Methods
- Exercises on List, Tuples, and Dictionary
- Remove Duplicate from Lists
- Program to find the index of an item of a tuple
- Python program to convert a list to a tuple
- Python program to reverse a tuple
- Program to convert a tuple to a dictionary

Making Decisions – if Statements

- The Relational Operators
- The Logical Operators
- Simple if Statement,if-else Statement
- if-elif Statement
- More Advanced If, Elif & Else Processing

Loop Control

- Introduction To while Loops
- Count-Controlled while Loops

- Event-Controlled while Loops
- Using continue, Using break
- Introduction To for Loops
- For loops with files, list, tuples, and dictionaries
- Mini Project: Decode the SMS: Translate or Decode a message based on the abbreviation

Iterators

- CSS Selectors
- Creation of CSS Selectors
- Comparison between Xpath and CSS Selectors
- Relative and Absolute CSS
- Handling Dynamic CSS Selectors

Functions and Scopes

- Introduction To Functions – Why
- Defining Functions
- Calling Functions
- Functions With Multiple Arguments
- Predicate Functions, Recursive Functions
- Function Objects, Generators, Decorators
- Anonymous Functions, Higher-Order Functions
- Scope, Global Scope, Local Scope, Nested Scope

Modules

- Using Built-In Modules
- User-Defined Modules
- Module Namespaces

File I/O

- Printing to the Screen
- Reading Keyboard Input
- Opening and Closing Files
- open Function, file Object Attributes
- close() Method ,Read, write, seek
- Rename, remove
- Mkdir, chdir, rmdir
- Mini Project: Anagrams in File

Error and Exceptional Handling

- Exception Handling, Assertions: The assert Statement
- What is Exception, Handling an exception
- The except Clause with No Exceptions, the try-finally Clause

- Argument of an Exception, Raising an Exceptions
- User-Defined Exceptions

Classes and Objects

- Overview of OOP-Creating Classes
- Constructor and destructor, Static and instance variable
- Creating Instance Objects
- Built-In Class Attributes, Destroying Objects
- Class Inheritance, Overriding Methods
- Base Overloading Methods
- Overloading Operators, Data Hiding

Regular Expression

- Matching and Searching- match() and search() Functions
- Search and Replace
- Regular Expression Modifiers
- Regular Expression Patterns
- Non-greedy repetition
- Backreferences, Anchors
- Automation with regular expression
- Exercises: Validate email address, Validate web address, Processing credit card numbers

Logging and Debugging

- Logging — Logging facility for Python
- Logger Objects, Logging Levels
- Creating Log files
- Python debugger- PDB

Multithreading

- Overview of Threading model
- Properties of Threads
- Creating Thread Using Threading Module
- Using queue with threads
- Synchronizing Threads

Using Database with Python

- CRUD operations (Create, Read, Update, and Delete)
- Storing data across multiple tables in database
- Linking rows in the database
- Retrieving and processing some data and then use in an application to visualize data