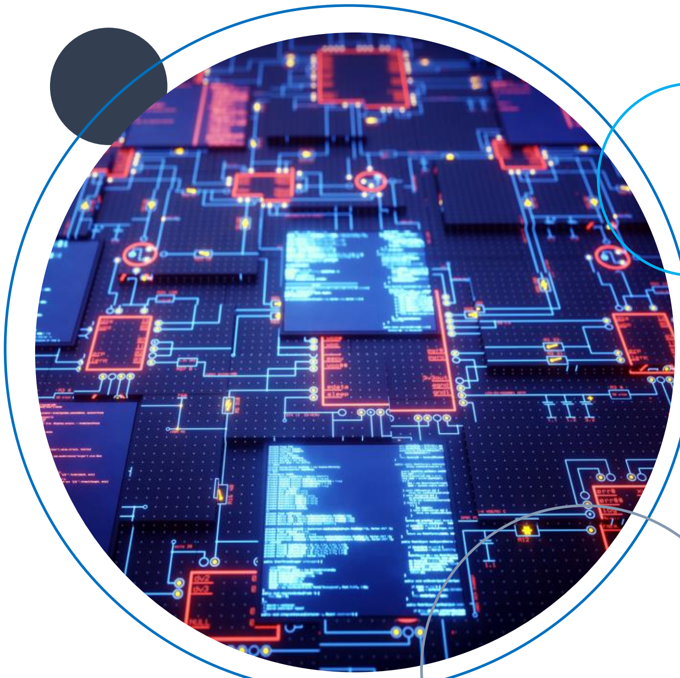
XTREMELABS



PYTHON BASICS

OVERVIEW

Module 1: Introduction to Python

This chapter introduces the Python programming language to include: the features of Python, the paradigms the development and uses and the installation of Anaconda.

Module 2: Python Objects

To be able to write a program in Python the programmer can use Anaconda, the installation of which was described in the previous chapter—or you can use IDLE, which can be downloaded from the reference given at the end of the Chapter 1. IDLE has an editor specially designed for writing a Python program.

Module 3: Conditional Statements

The chapter introduces the concept of conditional statements, compound statements, the if-elif ladder and finally the get statement. The chapter assumes importance as conditional statements are used in every aspect of programming, be it client-side development, web development, or mobile application development.

Module 4: Looping

This chapter presents the basics of the while loop and uses looping to create patterns., introduces the concept of nesting and presents the processing of lists and tuples using for loops.

Module 5: Functions

This chapter introduces the concept of functions. The chapter briefly explains the features of a function; the basic terminology and discusses the definition and use of a function. The chapter also discusses briefly the scope of a variable and presents recursion.

Module 6: Iterations, Generators, and Comprehensions

This chapter introduces the iterators and explains how to define your own iterable objects. The generators are introduced and explained in this chapter and deals with comprehensions, which makes the task of generating specific lists easy.

Module 7: File Handling

This chapter discusses the general file handling mechanism, the open() function and the various modes in which a file can be opened. This chapter also covers the functions for reading and writing to the file and introduces the functions to get and set the position of a cursor in a file. Important factions to carry out various tasks and the command line arguments are covered in this chapter as well.

Module 8: Strings

This chapter examines various aspects of strings like non-mutability, traversal, operators, and built-in functions. One of the most prominent differences between a string and a list is non-mutability. Once a value is given to a string, one cannot change the value of a character present at a position. For the users familiar with C, C++, C#, or Java, the operators discussed in the chapter - notably the * operator - will be a pleasant surprise. Moreover, Python provides many built-in functions to help the programmers to handle strings.

Module 9: Introduction to Object Oriented Paradigm

It may be stated here that the topics discussed in this chapter will be discussed in detail in the following chapters. Some of the readers not familiar with C++ (or for that matter C# or Java) may find the discussion abstract, but things will become clear as we proceed.

Module 10: Classes and Objects

This chapter presents the definition of a class, the concept of objects and discusses the instantiation of a class. Also discussed in this chapter are the scope of data members, constructor overloading, and destructors.

Module 11: Inheritance

Those students from a C++ background will have studied the importance of inheritance and composition. Inheritance was projected as a path breaking concept, which promised to solve all the problems and bring about a change in the way programming is done. But you must have understood that those who make such tall claims generally create more problems than they claim to solve. Inheritance may also create problems; many more than you can imagine. Object-oriented programming has its charms but also comes with its own problems, it is like demonetization. So, use inheritance only if required. Also remember never ever to use multiple inheritance. Remember that anything that can be done using inheritance can be done also in another way too. Composition, introduced later in the chapter, can be easily used to accomplish most of the tasks that can be done using inheritance.

Module 12: Operator Overloading

This chapter discusses operator overloading. This chapter revisits init and discusses how it can be overloaded. It presents some of the common operator overloading methods, provides an example of overloading the binary operators, discusses iadd, discusses the comparison operators, discusses bool and len and presents the concept of destructors.

Module 13: Exception Handling

This chapter concentrates on the importance and mechanisms of exception handling, presents some of the built-in exceptions in Python and summarizes the process by taking an example. This chapter also presents another example of exception handling.

Module 14: Introduction to Data Structures

We begin with the classification of a data structure. Data structures may be primary or secondary. Primary data structures are those which are provided by the language. For example, in C, “int,” “float,” and “char” are primary data structures. The primary data structures or data types of Python have been discussed in Chapter 2 .

Module 15: Stacks and Queues

This chapter presents the basic terminology and array-based implementation of stack. It discusses one of the dynamic implementations of stack and two important applications of stacks: reversal of string and infix, postfix and prefix expressions. Basics of queues and its implementation are also discussed in this chapter.

Module 16: Linked Lists

The previous chapter introduced two of the most important data structures; namely stack and queue. Stacks follow the principle of Last In First Out (LIFO) whereas queues are linear data structures which follow the principle of First In First Out (FIFO). That is, an element can be added or removed only from a specific position in these data structures. The data structure introduced in this chapter is far more flexible in terms of the insertion and removal of elements. The previous chapter also discussed the static implementation of stacks and queues. But the discussion on the dynamic implementation using linked lists was deliberately delayed. The linked list discussed in this chapter will help the user in the dynamic implementation of stack and queue. Here, it may be stated that Python provides functions for creation of linked lists and supported operations.

Module 17: Binary Search Trees

This chapter presents the definition, terminology, and representation of trees and graphs. Binary search trees (BST) are also discussed in this chapter.

Module 18: Introduction to NUMPY

This chapter introduces NumPy and discusses the creation of a basic array. The chapter also throws light on the data types provided by the package, discusses some of the standard functions for generating sequences, and discusses some important aggregate functions. The concept of broadcasting and structured arrays is also introduced and discusses in this chapter.

Module 19: Introduction to MATPLOTLIB

In the previous chapter, the methods and procedures used to deal with the data were discussed. The chapter introduced numpy which helps to accomplish numerous tasks. It is important to be able to visualize the data as well. Visualization gives an insight of the results and may help to uncover the underlying patterns. Matplotlib is a package that helps to plot various types of graphs and visualize the data. This chapter primarily discusses the pyplot package of matplotlib.

The pyplot collection of the matplotlib provides a set of functions which help programmers to perform various tasks associated with plotting. These functions provide MATLAB like capabilities to Python programmers. pyplot provides functions to plot a figure, create a plotting area, assign labels, etc. The pyplot keeps an account of the current figure and the area and hence can direct the functions to the requisite axes. The following topics are discussed in this chapter: the most important functions in pyplot and present some interesting examples, introduces basic plotting, discusses sub-plotting and presents 3D plots.

Module 20: Introduction to Image Processing

This chapter discusses image processing using standard functions in SciPy and NumPy, though image processing can also be done in other ways. The chapter introduces the basics of image manipulation in Python, discusses the contour function, introduces clipping and discusses basic transformations like translation, rotation, and scaling.