

6

Getting Started with Python

In This Chapter

- 6.1 Introduction
- 6.2 Python – Pluses
- 6.3 Python – Some Minuses
- 6.4 Working in Python
- 6.5 Understanding First Program/Script

6.1 INTRODUCTION

The word *Python* – isn't it scary ? Does it bring the image of big reptile that we prefer to see either in jungles or zoo ? Well, it's time to change the image. Now on, you'll remember word *Python* for its playfulness and pleasant productivity. Confused ? Well, Don't be – because, now on you'll get introduced to a new programming language namely 'Python', which promises to make you a big programming fan :-).

Python programming language was developed by **Guido Van Rossum** in February 1991. Python is based on or influenced with *two* programming languages :

- ⇒ *ABC language*, a teaching language created as a replacement of BASIC, and
- ⇒ *Modula-3*

Python is an easy-to-learn yet powerful object oriented programming language. It is a very high level programming language yet as powerful as many other middle-level not so high-level languages like C, C++, Java etc.

In this chapter, we shall introduce you to playful world of *Piquant Python* [Word 'Piquant' means pleasantly stimulating or exciting to the mind]. So, are we ready ? And... here we go.

NOTE

Do you know Python, the programming language, was named after famous BBC comedy show namely *Monty Python's Flying Circus*.

6.2 PYTHON – PLUSES

Though Python language came into being in early 1990's, yet it is competing with ever-popular languages such as C, C++, Java etc. in popularity index.

Let's see what are these places of Python

1. Easy to Use

Python is compact and very easy to use object oriented language with very simple syntax rules. It is a very high level language and thus very-very programmer-friendly.

2. Expressive Language

Python is an expressive language - fewer lines of code and simpler syntax. For example, consider following two sets of codes:

```
// C++ - Swap values
int a = 2, b = 3, tmp;
tmp = a;
a = b;
b = tmp;

# Python: Swap values
a, b = 2, 3
a, b = b, a
```

which one is compact and easier to understand? Need I say more? :).

3. Interpreted Language

Python is an interpreted language, not a compiled language. It makes Python an easy-to-debug language and thus suitable for beginners to advanced users.

4. Its Completeness

For most types of required functionality is available through various modules of Python standard library¹. For example, for diverse

functionality such as emails, web-pages, databases, GUI development, network connections and many more, everything is available in Python standard library. This is also called - Python follows "Batteries Included" philosophy.

5. Cross-platform Language

Python can run equally well on variety of platforms - Windows, Linux, Macintosh, supercomputers, smart phones etc? Isn't that amazing? And that makes Python a true cross-platform language. In other words, Python is a portable language.

6. Free and Open Source

Python language is freely available along with its source-code.

7. Variety of Usage/Applications

Python has evolved into a powerful, complex and useful language over these years. These days Python is being used in many diverse fields/applications, some of which are:

- ◆ Scripting
- ◆ Web Applications
- ◆ Game development
- ◆ Database Applications
- ◆ System Administrations
- ◆ Rapid Prototyping
- ◆ GUI Programs

6.3 PYTHON - SOME MINUSES (SO HUMAN LIKE)

Although Python is very powerful yet simple language with so many advantages, it is not the Perfect Programming language. Let's see what these are:

1. Not the Fastest Language

Python is an interpreted language not a fully compiled one. Fully compiled languages are faster than their interpreted counterparts. So, Python offers faster development times but execution-times are not that fast compared to some compiled languages.

2. Lesser Libraries than C, Java, Perl

Python offers library support for almost all computing programs, but its library is still not competent with languages like C, Java, Perl as they have larger collections available. Sometimes in some cases, these languages offer better and multiple solutions than Python.

1. If you install Python through Anaconda Python Distribution it loads most libraries and packages with Python.
2. Python even has versions that run on different languages such as Java (Jython), .NET (IronPython) etc.

3. Not Strong on Type-binding

Python interpreter is not very strong on catching 'Type mismatch' issues. For example, if you declare a variable as integer but later store a string value in it, Python won't complain or pin point it.

4. Not Easily Convertible

Because of its lack of syntax, Python is an easy language to program in. But this advantage has a flip side too. It becomes a disadvantage when it comes to translating a program into another programming language. This is because most other languages have structured defined syntax.

So now you are familiar with what all Python offers. As a free and open source language, its users are growing by leaps and bounds. As per Jan 2020 popularity index, Python was 2nd most popular programming language¹ after JavaScript. That is the reason, it's part of your syllabus. Together we'll make it playful Python :)

NOTE

Python is an interpreted language, that is, all the commands you write are interpreted and executed one by one.

6.4 WORKING IN PYTHON

Before you start working in Python, you need to install Python on your computers. There are multiple Python distributions available today.

- ❖ Default installation available from www.python.org is called **CPython installation** and comes with *Python interpreter*, *Python IDLE (Python GUI)* and *Pip (package installer)*.
- ❖ There are many other Python distributions available these days. **Anaconda Python distribution** is one such highly recommended distribution that comes preloaded with many packages and libraries (e.g., NumPy, SciPy, Panda libraries etc.).
- ❖ Many popular IDEs are also available e.g., Spyder IDE, PyCharm IDE etc. Of these, Spyder IDE is already available as a part of Anaconda Python distribution.

To install any of these distributions, **PLEASE REFER TO APPENDIX A**. We shall learn to work with both these distribution types.

Once you have Python installed on your computers, you are ready to work on it. You can work in Python in following different ways :

- (i) in Interactive mode (also called Immediate Mode)
- (ii) in Script mode

6.4.1 Working in Default CPython Distribution

The default distribution, CPython, comes with **Python interpreter**, **Python IDLE** (GUI based) and **pip** (package installer). To work in *interactive* as well as *script* mode, you need to open **Python IDLE**.

6.4.1A Working in Interactive Mode (Python IDLE)

Interactive mode of working means you type the command – one command at a time, and the Python executes the given command there and then and gives you output. In interactive mode, you type the command in front of Python command prompt `>>>`. For example, if you type `2 + 5` in front of Python prompt, it will give you result as 7 :

Result returned by Python $\xrightarrow{\hspace{10em}}$ `>>> 2 + 5` $\xleftarrow{\hspace{10em}}$ command/expression given here
 $\hspace{10em}$ \rightarrow 7

3. Source : <https://sites.google.com/site/pydatalog/pypl/PyPL-Popularity-of-Programming-Language>.