

PYTHON PROGRAMMING

**A COMPLETE STEP BY STEP
GUIDE FOR BEGINNERS**



RICH BURTNER

PYTHON

PROGRAMMING

A Complete Step by Step Guide for Beginners

RICH
BURTNER

© Copyright 2018 by All rights reserved.

It is not legal to reproduce, duplicate, or transmit any part of this document by either electronic means or printed format. Recording of this publication is strictly prohibited

CONTENTS

Introduction

Chapter One: A-Z Guide for the beginners

Chapter Two: Why learning Python is essential?

Chapter Three: Introduction to numerous data types in Python

Chapter Four: How to get started with python

Chapter Five: Concept of interpreters and interactives in Python

Chapter Six: Python or PERL; which one is best?

Chapter Seven: Python- An object-oriented programming

Chapter Eight: Python – The Subdivisions

Chapter Nine: Suitable Python Versions for Beginners

Chapter Ten: Python-Overview

Chapter Eleven: Advantages and disadvantages of python

Conclusion

Introduction

If you are viewing this book I can bet, you are a beginner to Python. If you are interested in learning Python, then you have chosen the right book. Well, Python is a very easy to understand language. In this book, I will try my best to prove myself right. No doubt, Python is a great language either you are a newbie in the world of programming or if you are already a professional programmer.

If you are a beginner in the world of programming, then you must be worrying about how to learn Python. This book will eradicate your worry for sure. Because, in

this book the basic concept of Python, and essential programming functions are explained through a straightforward step by step procedure. I have written this book in a very easy to understand manner so that everyone can understand it without any confusion.

The most interesting fact about Python Is that you need to install a specific OS in your PC, or Laptop because it works correctly and efficiently on every OS.

In the first chapters, we will learn the importance of learning Python, how to use it and the advantages and disadvantages of Python. In the fourth chapter, I will discuss step by step procedure of how to install Python using Mac, Linux or Windows. From the fifth chapter, we will start an easy to understand process

specially designed to take you from beginner level to an expert.

I am sure this book will not make you feel bored. I hope you will enjoy learning Python. Some books are available in the market on the topic “How to learn Python,” but I would like you to choose this one because once you do that, you will immediately realize the benefit in that decision.

Chapter One: A-Z Guide for the beginners



Do you want to start programming but confused about where to start? In this chapter, I will try to vanish your confusion.

Computer science is a cumulative field of study. It means that each previous topic relates to the previous one. Have you ever tried learning to program before but failed? You were not the fault. A good professor of a cumulative subject is the one

who curates the curriculum carefully.

While studying a cumulative subject, massive attention must be paid while designing the learning path.

At this instant, I would say do not choose a language for its ranking; choose a language for which you have a complete course curriculum or learning path that makes no assumptions about prior knowledge.

Now let's come to the topic.

What is Python?

Python was first designed by Guido van Rossum and appeared the first time in 1991 publicly. It is an interpreted high-level programming language and was

created on an ideology that pays particular focus on code readability. It provides a platform for explicit programming on both small and large scales. Python exhibits a dynamic type system and automatic memory management.

Python can support multiple programming paradigms like imperative, functional, object-oriented and procedural. It is also equipped with an extensive and comprehensive standard library.

Is it easy to learn Python?

Now a question may arise into your mind that is it easy to learn Python? Well, the answer is yes. Here are some reasons that will surely justify my statement.

- **Easy to understand the syntax**

As the syntax feels natural, so a code written in Python is straightforward to understand. I am taking an example case to elaborate my point clearly

`g=1`

`f=4`

`Sum=g + f`

`Print Sum`

Let's take another case

The first number = a

Second number = b

Answer = (a * b)

- **Not an overly strict language**

Defining the type of variable in Python is not necessary, and the code will run

perfectly even if you forgot to put the semicolons at the end.

- **User-friendly language**

Python is a more user-friendly programming language as compared to Java because it has a more subconscious coding style. Java and Python have their advantages and disadvantages, but if you are going to start your career as a programmer, then I would recommend you start by learning Python as it is less complicated than Java.

What is the use of Python Programming?

The examples of some scripting languages are PERL, PHP, and Python, etc. Python is a very useful language for physics sims

or statistical models. You can use Python for creating a desktop application like Blender 3d and for games pygame. You can also translate python into a binary code like java. An interesting fact is that most Ubuntu configuration utilities are Pythonic.

Some other uses of Python are listed below:

- **Software and SAAS products**

To help technology and software-driven companies with their software and SAAS products.

- **Python big data**

Help leverage cloud platforms for building and delivering solutions.

- **Mobile Web solutions**

Build customer and enterprise mobile app for customer engagement and workforce productivity

- **IoT solutions**

Build solutions to monitor, learn and capture insights from connected devices.

Chapter Two: Why learning Python is essential?

After a brief intro, I would like to inform you that based on its usage learning python is very important. Python has diversified applications, and the most exciting thing is that it has a syntax that is understandable by both beginners and experts.

In the following section, I will discuss some reasons to make you realize that how important is it to learn python.

- As discussed earlier Python is a user-friendly language. Python is one of the most-used dynamic type, programming language.

- In the past few years, two programming languages were most popular for developing websites by their framework. Those languages are Python and Ruby. These languages are beneficial for writing quick programs or scripts, so they are often known as “Scripting languages.”. In Data Analytics Python is the most widely used language. Because of the Python’s improved lib support, Python is popular among developers because it provides a compelling alternative to performing data manipulation tasks.
- NumPy, Pandas, matplotlib, IPython, and SciPy are the examples of some necessary libraries in

Python that can be very useful for Data Analysis.

Python is equipped with much more libraries specially designed for Data Analysis. Conclusively, I would say Python is an “essential to learning” programming language.

Chapter Three: Introduction to numerous data types in Python

Python consists of mainly five standard data types. In this book, I will discuss only three data types used In Python one by one.

When we program, the information provided to the computer is referred to as data.

Five data types in Python are:

- Numbers
- Strings
- Boolean

NUMBERS:

-Integers:

N=1,2,3

I=-111

-Float:

Decimal points i.e. g=6.87, f= 10*e^-10

The first of our basic data types are numbers including integers, e.g., 3, -11. Real number or decimal number which have floating points are called floats. Complex numbers, e.g., J which is equal to the square root(-1) which is an imaginary number.

STRINGS:

Besides these, we also have strings which resemble words each string is comprised of letter or elements and. They are defined using quotation marks such as name or surname or even space which is an empty character.

Name="MARK"

Surname="Quinn"

Space=' '

BOOLEAN: True or False

This data type may be less familiar to you. Boolean data type has two conditions, i.e., true or false. For example, we can test variable name equals mark by calling the bool function, which evaluates the name equal to mark as true or false and in this case, it returns true before moving on

N=bool (name=='MARK')

N=true

Chapter Four: How to get started with python

In this section, I will guide you completely how to install python in your PC, or Laptop, Python is very easy to Install. First, I will let you know how to install Python on Windows.

A few years back, a significant update to Python created a split between Python versions. This thing is a little bit confusing for the newcomers, but it is not a big issue. In this book, I will talk about the installation of both versions of Python.

When you search Python for windows on google, you will immediately see the

divisions. The repository ask whether you want to install latest releases of Python 2 or Python 3 (2.7.1 and 3.6.1 respectively).

If you're going to learn Python thoroughly, I will recommend you download both the versions side by side. This will allow you to learn the latest version of Python as well as the older version.

But if you think you only need a particular version of Python, then you can install just Python 2 or Python 3. In this book, I will discuss the installation of both, so I recommend you install both versions. Under the main entry, you will see an “x86-64” installer. Click on this installer this installer will automatically install the appropriate 32-bit or 64-bit version on your laptop or computer.

How to install Python 2

After downloading the Python 2.

- Click on setup, a dialogue box will appear on the screen.
- Select “Install for all users” option.
- Click “Next” to proceed.

When you click “Next” a directory selection screen will appear.

- Leave the directory as “Python 27”.
- Click “Next” to proceed.

When you click next, a customization screen will appear.

- Scroll down and click “Add Python.exe to the path.”
- After clicking “Add Python.exe to Path,” select “Will be installed on local hard drive” option.

- Click “Next” to proceed.

After this point, you have to make no more decisions. Just click through the installation wizard to complete the installation process. You can also confirm the installation.

How to confirm the installation?

- Open the command prompt.
- Type ‘Python -V’.

Success! If you need only this particular version of python you can stop right here because installation is complete and the path variable is set; now you can use the Python.

How to install Python 3?

You can install the Python 3 along with python two without any issue if you want

to use and learn the latest version of Python.

On the first screen.

- Click “Add Python 3.6.exe to path” option.
- Then click “Install now” to proceed.

At this instant, you have to make a decision. If you want to remove the limitation on MAX_PATH variable, click on “Disable path length limit” option. If you don’t want to remove the limitation on MAX_PATH variable, you can finish the installation by clicking the “Close” option.

You can also confirm the installation of Python 3 by opening the command prompt and typing ‘Python -V’ command, this will

also assure you either the path variable is set or not.

Chapter Five: Concept of interpreters and interactives in Python

To run the program, the majority of languages needs you to write source code, but with Python, everything is more comfortable you don't need to write source code to run a specific program in Python that is why is defined interpreter. Python allows its user to write the instruction directly from its command prompt without editing or creating a source file (Interactive).

Interactive mode enables you to experiment with new syntax variations.

One of the benefits of working with an interpreted language is that you can test

bits of code in the interactive mode before you put them in the script. There is a little difference between interactive mode and script mode that can confuse the Python users.

For example, if you write code for a calculator in Python, you may write

```
>>>miles=26.2
```

```
>>>miles*1.62
```

```
42.182
```

The first line will assign the value to miles, but it has no visible effect. The second line is an expression so that the interpreter evaluates it and display the result. So we learn that a marathon race is about 42 kilometers.

But when you write the same code in the script, you will get no result at all because It has no visible effect. Python evaluates the result but doesn't display it unless you tell it to.

```
>>>miles=26.2
```

```
Print miles*1.61
```

This behavior is confusing. Some languages get compiled into machine code; others are read and executed by an interpreter. So like C would be an excellent example of a compiled language and Python is an excellent example of interpreted language. Python programs are just text files with py extension to their file name. These files need an interpreter to execute.

Python interpreter is an environment that we can use to execute little Python

commands so its kind of like this little sandbox where we can test out and try out different Python command and different Python functions in a very safe and very neutral environment. I can say that a dirty way to write programs in Python is Python interpreter. You can use the Python interpreter by opening up the command prompt.

Now let's go into the interactive mode.

The interpreter can also be used in interactive mode.

INTERACTIVE MODE

The common mode in which python works is the one described earlier that is script mode, while the other one is interactive mode. The basic difference in both these modes is that the interactive mode keeps the previous statements in its active memory which are fed in this. The feedback to newly given statements is immediate and interpreted by the previous fed statements. The lines that are given to the interpreter are always interpreted in two sections, in part as separate and also as a whole.

The interactive mode in comparison to the script mode is more suitable for you if you want to play with Python programming. It is also preferred for using different varieties on the syntax.

How to go in interactive mode:

If you are using the macOS or the Linux, you will have to follow the below-given steps as they are.

Firstly, you open the terminal that is there for typing.

Then type “python” in it.

If it's the windows you are using, then there are several ways to go to the interactive mode. For example, open command prompt and then type “py” in it and there you are in interactive mode. Or otherwise, you go to the taskbar or the app menu and select “Python (Command line),” “IDLE” or any other program that is similar to this one.

IDLE is the GUI type of program that is dual in function. It means that is run the

files in interactive mode but does have the option to edit them and run them within the same program.

When you print the python entered line, they will appear much like this

```
Python 3.0b3 (r30b3:66303, Sep 8 2008,  
14:01:02) [MSC v.1500 32 >>bit (Intel)]  
on win32
```

```
Type "help", "copyright", "credits" or  
"license" for more information.
```

In case if you are facing trouble in running the Python make sure that it was correctly installed and also check whether the Path variable has been set or not.

How can you know that you are in interactive mode?

The sign `>>>` shows that you are in interactive mode

Why use the interactive mode?

The best reason why people are attracted to this mode is that it responds quickly. As soon as you type, you get the results. If you type `1+1` python will give you two immediately. Moreover, it is useful because the interactive mode helps you to explore what python can do. It assists in testing new and creative ways to use the python program.

The sample of the interactive session is as follows:

```
>>> 7
```

```
7
```

```
>>> print (7*6)
```

```
34
```

```
>>> "how are you" *4
```

```
'How are cookware you how are you.'
```

```
>>> "how are you." Class_
```

```
<type 'str'>
```

This might seem quite easy to follow and use in the program, but you have to be very careful in this regard. The interactive environment can be a little confusing. There is a sample Python script that is usually followed, which is given as follows.

If 1:

```
Print ("True")
```

```
Print ("Done")
```

This is the format specific \to Python only whereas if you type this in the interactive environment, you might be surprised to see what appears as a result.

```
>>> if 1:
```

```
...     Print ("True")
```

```
...     Print ("Done")
```

```
File "<stdin>," line 3
```

```
Print ("Done")
```

```
^
```

```
Syntax Error: invalid syntax
```

Now the interpreter is trying to convey is that indentation you entered for the second print is unexpected. The possible error is that there must have been a blank line entered at the end of the first statement (i.e., “if”) before the second print statement started.

For example, the right way to enter the statement should be this

If 1:

```
    Print (“True”)
```

```
Print (“Done”)
```

This would then result in the following statement which would be correct technically

```
>>> if 1:
```

```
... Print ("True")
```

```
...
```

```
True
```

```
>>> print ("Done")
```

```
Done
```

```
>>>
```

So this way of the interactive session would be useful and functional.

Chapter Six: Python or PERL; which one is best?

Python or PERL? Which one is better, this question may arise into your mind. In this section, I will compare Python with PERL and also discuss key features of both languages so that you can choose between Python and PERL easily.

Why is PERL better than Python?

PERL is a much older language; it was the first language used for backend web development. PERL is 25 years old. Python is much newer and is now only in its third revision PERL, and Python are both on Earl Raymond's list of languages to learn. He also suggests learning Lisp, and that its only major implementation today is

chatbots. The list of the top six languages to learn created by Michael O' Church doesn't include PERL.

I personally like Python because it is straightforward to learn. But Python can't be used for larger system applications. You can use Python for solving maths problems, creating games, building simple apps and also for a lot of other purposes. Relative to the lines of newly created code Pearl and Python both have faced their usage drop off in the past recent years. PERL has successfully maintained that level for the last ten years, never getting as big as Python. The job openings for PERL developers are more than that of Python developers because the number of developers of Python is decreasing day by day, but the Python developers are

increasing relatively. Initially, developers used PERL for system administration. PERL can't do the interfacing well because it was designed to support a large number of applications on the same server. PERL and Python both have their uses, but PHP replaces both these languages.

Why did I start using Python?

I started using Python since around 1996, what attracted me to Python? Well, the thing that attracted me to Python is that I was spending all my time doing annoying problems like moving files around not cleaning data more like administrative tasks onto it that are sort of tangential to high-performance computing and I was like wasting all of my time with that. I say Python is maybe a productivity tool to simplify a lot and with the right scripts that

would do data analysis. The other thing I did is I used Python as a control language for C. Python can be used as a click language for building large applications like NIC like a mix of Python and a mix of C. It was just a huge productivity win. Python is like a computational corner to matrix multiply fast Fourier transform or something like that. Variables and initial conditions can be easily changed in Python without rewriting the whole code again.

Chapter Seven: Python- An object-oriented programming

Another unique feature of Python is being an Object-oriented (OOP) program. Object-oriented programming is the one which is based on data instead of logic or objects instead of actions. This concept is applied to the majority of programming languages; C language is the only exception. The idea revolves around purposes like inheritance, polymorphism, abstraction, encapsulation, etc.

Understanding the basic concepts of OOP is very important, especially for a python or C++ as well as Java user. And that is the main reason for the inclusion of this chapter in this book. Beginning from the

basics, let us try to explain to you the meaning of the terminologies frequently applied in this language. To illustrate the concept of object-oriented programming clearly, I am going to relate it to human life. Taking the example of humans which are majorly classified into two categories/classes; Male and Female, both have similar body structure, but some of the body parts specific and are only confined to a particular category. However, actions like sleeping, eating, walking are performed by both.

Class

With the help of the above example, here we will consider Human beings as the Class. A class works as the layout for any functional creature, which determines its characteristics and actions. Just like a

living body of a human being which performs different functions.

Inheritance

While human beings are assumed as Class, their body parts become properties, and their action becomes functions. Similarly, male and female can also be considered as classes which can inherit the properties by the concept of Inheritance.

Object

Explaining this, a person named Kelvin is an object for the class Male. The physical existence of a person which is objected to change makes it an Object.

Abstraction

Abstraction means portraying the exterior stuff while masking the interior

specifications. Just like a human body can show off their skin but beneath it remains a secret to others.

Encapsulation

The idea of this terminology is the attachment of properties for the performance of the function. We can take the example of our legs which assist us in moving or performing an action.

Polymorphism

Polymorphism is the idea of elaborating the same thing differently. This is achieved either by changing the properties of work or the way it is performed.

Chapter Eight: Python – The Subdivisions

Being a programmer means having a backup of libraries required to interpret the codes of language. But python gives you the facility of having already verified and working libraries which are known as ‘Modules.’ These modules contain already written codes which can perform essential functions for you in Python programming. Also, these tasks are completed efficiently and without any errors.

While using this program, you might come in contact with phrases or words like DRY, which has the abbreviation of ‘Don’t Repeat Yourself’ or ‘Don’t Repeat

Someone Else.’ Another such word which is used in this language is Wrapper; this has a similar meaning to the wrapper used in our daily language. Python wrapper over C++ code means that the python code someone has used interacts with C++ language. This feature enables you to work with languages you are not aware of by applying the Python wrapper.

Python facilitates you in managing data analysis, governing and controlling robots and hardware; create games as well as websites.

You can also create GUI’s or Graphical User Interface with the help of Python. GUI enables to make the program more user-friendly by adding graphics to it.

To gain access to a module in Python, you have to use the `Import` statement, and the code will be executed. But before accessing any module, you must look for the built-in module in it (Python contains a range of integrated module).

Chapter Nine: Suitable Python Versions for Beginners

For the people who have decided to work with Python for the very first time, the best option would be to go for the 3rd version. In the case where your needed options are not available in the 3rd version, then go for the 2nd one. By having the giving imports at the top of the program of Python 2, you can go like Python 3.

If you are new to Python and the packages you need are already available on Python 3, start with Python 3. If some packages you need are not yet available in Python 3, then chose Python 2.

```
from __future__ import division,  
print_function
```

The best version according to me is the Python 2.7 version, which is the current update of the 2.x series. It is more feasible to use Python 2.x as most of the modules are applicable on that version, not all the modules of Python 3 work on both versions. If you become a power user of the 3rd version you might face some trouble as not all the computers have Python 3 installed. The new updates are going to be applicable on the 3rd version only, which makes this version perfectly usable for absolute beginners. Many of the modules have been shifted to Python 3, remaining are in the process of transmission.

Chapter Ten: Python-Overview

A list of frequently used keywords in Python and their description is given below.

IDE–Integrated Development Environment

You can find this option in the main menu, and it is used to edit, test and operate the tools that are created. IDLE can be coded in Python by the Tkinter GUI toolkit cross-platform, in Windows, Mac OS X, Linux or Unix, the process and features remain same. It contains the Python Shell Window which is an interactive interpreter, which adds color to your codes. It enables search within a window and also multiple windows.

IDLE contains two types of windows, namely Shell Window and Editor Window. Editor window has a subtype which is called Output window; the only difference is of the default title and context menu. You can open more than one Editor Windows at a time and use them. There is a wide range of cuisines for IDLE, depending on the type of windows that you are using.

Let us see how to run a command on IDLE now

```
print("Let's start")
```

As we save the file by “file/save” and name it start.py. Now we can run this by the “run” command.

Below are the names of few editors that write the Python program amongst many others.

- Eclipse with PyDev: a Large community with considerable variation of customization.
- Eric: This one utilizes the Qt framework, Scintilla that is code editing component.
- PyCharm: A free community, beneficial for the learners.
- Other choices include PyScriptor, LeoEditor, BlueFish, Spyder, Pt, and Geany.

Chapter Eleven: Advantages and disadvantages of python

Python has a wide range of applications in software development companies like gaming, web frameworks and applications, language development, prototyping, graphic design applications, etc.

Like every other language Python also has its advantages and disadvantages. In the succeeding section, I will let you know about the advantages and disadvantages of Python.

Firstly, I will list the advantages then we will talk about the disadvantage.

Advantages of Python

- Extensive Support Libraries
- A powerful and fast programming language.
- Clear, readable, and easy to understand syntax.
- Dynamic type programming language.
- Python Runs immediately.
- The python community always provides support to Python users.
- Modular and portable programming language.
- Less coding is required.
- Open source language.
- Easy to maintain.

Disadvantages of Python

- Design restrictions.
- Underdeveloped database access layers.
- Weak in mobile computing and browsers.
- Speed limitations.

As can be seen from the above bullet points pros exceed the cons so in conclusion Python is a useful programming language.

Conclusion

I hope this book is useful for you as a Python guide. This is quite easy to learn as it includes several images to follow up on the program. Hopefully, by now, you have understood what Python is and how it works. Python has been extensively spread in the world as the best programmer. This book is written to guide the beginners as the programming is considered a difficult task for nascent users. This book is preferred by the readers as this include images to follow and is written in easy language. Python is easy to install and has no complex language like other primitive programs. This is even

suitable for only a single individual to use and require no other links.

It includes different implementations as well including CPython. It is in fact, the reference implementation. Mostly, the implementation of the Python is ruled by the read-eval-print loop. It means that the statements are entered sequentially, and the results will be given immediately. The Python is also useful as its memory consumption is much better than any other language programmer, e.g., Java or C. many comparisons have been conducted of python and other conventional programs, and every comparison is ruled out by Python.

Python has also been used in advance purposes like artificial intelligence.

Natural languages have also been

processed using this software. This is the scripting language that has a proper modular architecture. Moreover, its syntax is simple and error free. Its tools for the processing has many variants, and thus it is used for the natural languages processing and scripting. Different operating systems and laptops have Python as the standard component these days. That is the primary reason for developing this book so that people who have yet no experience with this program should know how to use it or at least have an already developed guide. This book will let you explore and use this program when required. I hope my sole purpose to write this book is completed in the future and you may enjoy reading and then use the Python program which is expanding its

roots in the world of programming and languages. So what are you waiting for? Install Python in your PC or laptop and start learning.

Good luck and happy learning.

