SHRI RAMSWAROOP MEMORIAL UNIVERSITY

विद्ययाऽमृतमश्नुते

UTTAR PRADESH

# *PYTHON PROGRAMMING LANGUAGE*

**Anurag Soni**

**SCIENCE INSTITUTE OF TECHNOLOGY**

**SHRI RAMSWAROOP MEMORIAL UNIVERSITY**

**2020**

# TABLE OF CONTENT

# Introduction of Python

**Python** is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This **tutorial** gives enough understanding on **Python programming** language.

## Why to Learn Python?

**Python** is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

**Python** is a MUST for students and working professionals to become a great Software Engineer  specially when they are working in Web Development Domain. I will list down some of the key  advantages of learning Python:

- **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not  need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** − You can actually sit at a Python prompt and interact with the  interpreter directly to write your programs.
- **Python is Object-Oriented** − Python supports Object-Oriented style or technique of  programming that encapsulates code within objects.
- **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple  text processing to WWW browsers to games.

# History of Python

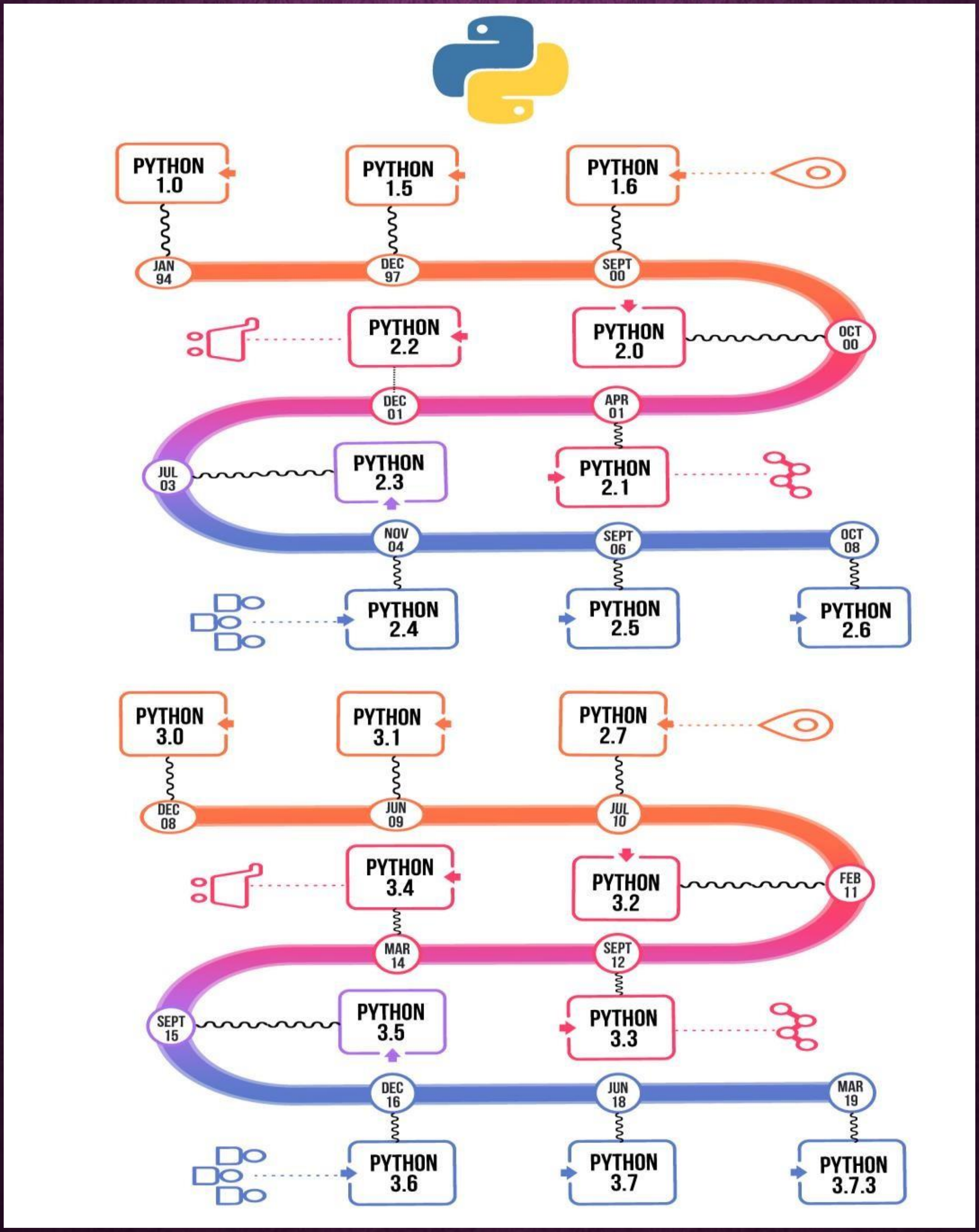
Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

# Python Releases Versions



# Python Features

Python's features include −

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax.

  This allows the student to pick up the language quickly.

- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.

- **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.

- **A broad standard library** − Python's bulk of the library is very portable and cross- platform compatible on UNIX, Windows, and Macintosh.

- **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.

- **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

- **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.

- **Databases** − Python provides interfaces to all major commercial databases.

- **GUI Programming** − Python supports GUI applications that can be created and ported to
  many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** − Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are list below −

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.

- It provides very high-level dynamic data types and supports dynamic type checking.

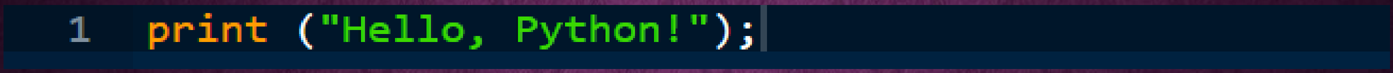
- It supports automatic garbage collection.

- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

# Hello World using Python

Just to give you a little excitement about Python, I'm going to give you a small conventional  Python Hello World program, you can try it using Demo link.

Live Demo



```
print ("Hello, Python!");
```

# Applications of Python

As mentioned before, Python is one of the most widely used language over the web. I'm going to list few of them here:

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax.

  This allows the student to pick up the language quickly.

- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.

- **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.
- **A broad standard library** − Python's bulk of the library is very portable and cross- platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

- **Extendable** − You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** − Python provides interfaces to all major commercial databases.
- **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** − Python provides a better structure and support for large programs than shell scripting.

# First Python Program

Let us execute programs in different modes of programming.

## Interactive Mode Programming:

Invoking the interpreter without passing a script file as a parameter brings up the following prompt
–

```
$ python
Python 2.4.3 (#1, Nov 11 2010, 13:34:43)
[GCC 4.1.2 20080704 (Red Hat 4.1.2-48)] on linux2
Type "help", "copyright", "credits" or "license" for more
information.
>>>
```

Type the following text at the Python prompt and press the Enter
–

```
>>> print "Hello, Python!"
```

If you are running new version of Python, then you would need to use print statement with  parenthesis as in **print ("Hello, Python!");**. However, in Python version 2.4.3, this produces the following result –

```
Hello, Python!
```

## Script Mode Programming:

Invoking the interpreter with a script parameter begins execution of the script and continues until  the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension **.py**. Type the  following source code in a test.py file –

Live Demo

```
print "Hello, Python!"
```

We assume that you have Python interpreter set in PATH variable. Now, try to run this program as follows –

```
$ python test.py
```

This produces the following result –

```
Hello, Python!
```

Let us try another way to execute a Python script. Here is the modified test.py file –

Live Demo

```
#!/usr/bin/python

print "Hello, Python!"
```

We assume that you have Python interpreter available in /usr/bin directory. Now, try to run this program as follows –

```
$ chmod +x test.py # This is to make file executable
$./test.py
```

This produces the following result –

```
Hello, Python!
```

# Python Identifiers

A Python identifier is a name used to identify a variable, function, class, module or other object. An identifier starts with a letter A to Z or a to z or an underscore ( _ ) followed by zero or more letters, underscores and digits (0 to 9).

Python does not allow punctuation characters such as @, $, and % within identifiers. Python is a case sensitive programming language. Thus, **Manpower** and **manpower** are two different identifiers in Python.

Here are naming conventions for Python identifiers −

- Class names start with an uppercase letter. All other identifiers start with a lowercase letter.
- Starting an identifier with a single leading underscore indicates that the identifier is private.
- Starting an identifier with two leading underscores indicates a strongly private identifier.
- If the identifier also ends with two trailing underscores, the identifier is a language-defined special name.

# Summary

I just loved Python as it's my favourite programming language, and got to know that where all the  programming languages are used. It's just everywhere in all the fields of science and technology.  Through this programming language I got to implement my algorithms in the real world. Its cleaner  codes helped me a lot in debugging. From Python I learned many of the new skills which ever  software engineer is destined to learn.

This programming language provided me all the kinds structures through which a software  project is managed and arranged to have high coupling and low modularity.

Through Python I learned to code Web Applications (cross-platform apps), how a page in a  website is maintained, how to handle a GET, POST, PUSH, DELETE, etc. kind of requests, and many othernecessary stuffs which are required to build a web app.

It gave me a setup to build desktop applications which are completely responsive and attractive as  well as through this programming language I learned to code Machine Learning models,  visualization, and many other skills of Data Science.

# Conclusion

While learning Python I got to know that how vast this programming language is and how difficult that gets when learning to its advanced level but it gets usual by building a habit of coding.

Learning Python brought me to a conclusion that its worth learning Python with whatever field of computer science you wanted to move ahead and its really easier when compared to other programming like JAVA, Java Script, Ruby, R, C, C++, etc. While being an interpreted language it warns you over every statement of your code when having an exception or an error.

# Future Scope

As Google, Microsoft, Amazon, and other tech giants have been shifting their code to python therefore this implies that how great the future of Python and Python Developers is going to be.

Whether you wanted to be a self-taught developer or a software developer or Data Scientist in one of those big tech giants then obviously one must go for Python for a bright future as progress stats of programming languages from Google and GitHub says that Python is having a rapid growth in its popularity with almost 800% per year.

Python developers are paid high wages for their work and is expected to increase with the passage of time.

# References

I won’t have ever been able to complete this report without referring to Guido van Rossum  creator of Python, and help from some of the below mentioned websites –

- www.wikipedia.org
- www.python.org

Special thanks to https://codewithmosh.com which acted as an instructor to me for completing  this report and provided me with all of the fundamentals of Python.

Thanks to HOD Ma’am, and Faculty Co-Ordinator of Industrial Visit Mr. Anupam Singh who  assigned us this interesting work.