

Python as Multi Paradigm Programming Language

Nimit Thaker

Bachelor of Computer Applications
Smt. Chandaben Mohanbhai Patel Institute of
Computer Applications
Charotar University of Science and Technology
Changa-338421, Ta.:Petald, Dist: Anand, Gujarat,
India

Abhilash Shukla

Assistant Professor
Smt. Chandaben Mohanbhai Patel Institute of
Computer Applications
Charotar University of Science and Technology
Changa-338421, Ta.:Petald, Dist: Anand, Gujarat,
India

ABSTRACT

This paper will revolve around Python as a Multi-Paradigm Programming Language.” and about how python has evolved to become the best suited and best opted language for developing almost all kinds of Applications. This Paper will also include various characteristics and features of python proving why it is the widely used programming language in recent times. Further, this paper explains about the growth of python in past five years, as an upfront emerging programming language with world’s Largest developer community. Advancing further with the paper, this paper consists of the role of python in various disciplines and trending areas of Information Technology. Like IoT Applications, Machine Learning, Deep Learning, Artificial Intelligence, Cyber Security, etc. On Reaching Towards the end after the complete study and analysis, this paper concludes with ultimate views about “The Multi-Paradigm Programming Language, Python.”

Keywords

Python, Multi Paradigm, IOT, Artificial Intelligence, Cyber Security

1. INTRODUCTION

Python has become the most grossing and highly popular programming language. It has been on the top charts of “Best Programming Languages”, for five years now. This huge hype of python is because of so many reasons, which we will be discussing in this paper, along with reasons why python is best ideal programming language for modern day high-end application development or support which any other Programming language is not that much flexible in providing. The ultimate goal of this paper is to let Application Developers and technology enthusiasts experience the power of Python programming language along with providing them an in-detail knowledge all of its powerful python libraries, packages and frameworks to develop and support almost any kind of application.

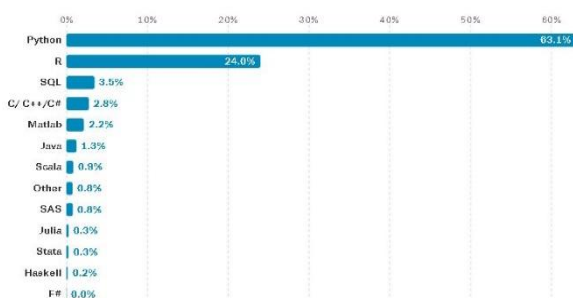


Fig 1: Popularity of Python

2. PYTHON: A MULTIPARADIGM PROGRAMMING LANGUAGE

Python is a High-Level, general-purpose programming, Interpreted language. Created by Guido van Rossum. It is an Open-Source Language and hence, the Python source code can be obtained from under the GNU General Public License (GPL). Python enables easy and clear programming for both small and large scales by providing various constructs. Its design philosophy emphasizes efficient code readability, notably using significant whitespace. Most Importantly, Python can easily incorporate object-oriented, imperative, functional and procedural programming paradigms with an enriched comprehensive standard library.

2.1 Characteristics Of Python

2.1.1 Python is Interpreted

Python is processed at runtime by the interpreter, it does not require compiling the whole program before executing it. The python interpreter directly executes the program, line by line translating each statement into a sequence of subroutines and then into machine code, this makes python more flexible than other programming language by providing smaller executable program size.

2.1.2 Python is Interactive

Python being an Interpreted language, enables users to interact with the python interpreter directly to write the programs, just by sitting at a python prompt. The Python prompt gives various detailed messages for any type of error or for any specific command being executed, it also supports interactive testing and debugging of snippets of code.

2.1.3 Python is Object-Oriented

Python is a Multi-Paradigm programming language Hence, it supports object-oriented style, rules and techniques of programming that encapsulates code within objects. Also, each and everything written in the python source code is in the form of classes and objects.

2.1.4 Python is a Beginner’s Language

Python is the best option for Initial or beginner-level programmers. It supports development for a wide range of applications ranging from simple-text processing to world Wide Web Browsers to games.

2.1.5 Python is convenient to write and is quite readable

Python is a strongly typed language, it has few keywords, simple structure and a clearly defined syntax. This allows any amateur individual to pick up the language quickly. Python’s code is clearly defined and visible to the eyes, as it uses whitespaces to indicate the beginning and end of a block, i.e. it purely works on indentation.

2.1.6 Python is Portable

Python has the capability to run on a very wide variety of hardware platforms with the same interface. It seamlessly runs on almost all operating systems such as Windows, Linux, UNIX, Amigo, Mac OS etc, without any changes.

2.1.7 Python is Extendable

Python is so versatile and flexible programming language with a broad standard Library that it allows the programmers to add existing or create new low-level/High-Level modules and packages to the python interpreter. These modules and tool-packages enables the developers with very portable and cross-platform development possibilities which helps them to create and customize their programs, applications or software tools to be more accurate and efficient. This Expendability of Python programming Language is one of the key reasons that it is becoming the most popular programming language.

2.1.8 Database Support

Python being such a full-fledged Programming language supports all the major or minor, commercial as well as exclusive databases. As Python is scalable, portable and extensible programming language hence, it is very conventional to write any type of code to make an application enabled with any feature required including any database support like, MySQL, PostgreSQL, SQLite, ElephantSQL, MongoDB, BitCan, Google Cloud SQL, Amazon Relational Database Service(RDS) and much more.

2.1.9 GUI Programming

Every application comes with a simple but effective User Interface, hence programming language with GUI Programming unctionalities is very necessary. Similarly, Python with all the other functionality also provides GUI Programming facilities. Python allows to write code in a such a way that an user interface for an application can be easily created, just by importing libraries like, Tkinter, Flexx, CEF Python, Dabo, Pyforms, PyGUI wxPython etc.

2.1.10 Python as Scripting Language

Python can also be used as a scripting language, both batch and Interactive use is possible with python. Also, it follows simple scoping rules, hence flexible dynamic typing can be done. moreover, as being a scripting language, python allows easy access to other programs. It can be compiled to byte-code for building large applications.

2.1.11 Easy Memory Management

Python is outgrowing nowadays for many reasons, and one of the reasons is Efficient Memory management. It supports automatic garbage collection. Python Memory management consists of a private heap that contains all Python objects and data structures, this is Done by Python memory manager. This Python feature allows developers to create advanced applications with low size and clean and fast execution.

2.1.12 Python for web development

web applications are on a spark now a days, and so is python. Python offers a variety of choices for developing web applications, this is possible because of python's scalability as discussed earlier in this paper. Python's Standard Library incorporates many protocols for web development such as HTML, XML, ISON, E-Mail Processing, It also provides base for FTP, IMAP, and other Internet protocols.

Python being so flexible provides a hands-on easy to use socket interface, not just this, python has many more libraries for web development like, Requests as HTTP Client Library,

BeautifulSoup as an HTML Parser that handles all sorts of oddball HTML, Feedparser for parsing RSS/Atom feeds, Paramiko for implementing the SSH2 protocol, and lastly Twisted Python, a very great framework for asynchronous network programming. Python has many development frameworks, which makes web application development very convenient and reduces the steps and efforts, like Django, Pyramid, CherryPy, Sanic, etc. There are many micro-frameworks as well, like, Flask, Bottle, Tornado, etc. Python also integrates many content management systems such as, Plone, djangoCMS, etc.

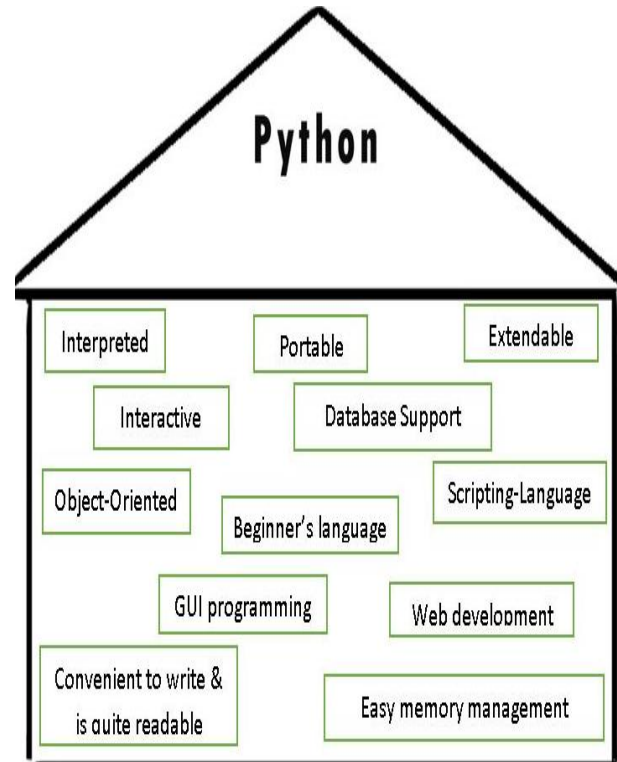


Fig 2: Features of Python

3. ROLE OF PYTHON IN VARIOUS DISCIPLINE

3.1 Python for Data Science

The data science community is vigorously switching to python programming language from R language, as python enables the data scientists with existing highly packed functionalities and also lets them to create their own to perform extra highly complex scientific and numeric calculations and computations, it also helps them generate various types of research and analysis reports, Histograms, graphs and much more.

There are many libraries or modules for applying data science in python, Mainly,

Numpy, i.e Numeric Python Package which is generally used advance Numeric Computation.

SciPy, i.e Scientific Python Library which is collection of packages for mathematics, science and engineering.

Pandas, it is one of the best libraries for data analysis and data modeling.

IPython, it is a great and functionally powerful interactive shell that facilitates data scientists with easy editing and recording of a work session, and provides a support for

visualizations and parallel computing.

Matplotlib, it is a library that facilitates data scientists with multiple ways for visualization of data as it has many functionalities for data visualization and data representations.

there are many other data science libraries which are readily available to download, import and use in a project, like, matplotlib, Scrapy, PyQt, kivy, OpenCV, PyBrain, Keras, Seaborn, etc. A developer can use anyone out of millions of libraries that suits him/her well.

3.2 Python for Networking and Digital Forensics

Python is so diversified language that not just developers but it also allows Network administrators to perform and automate various tasks or jobs. Whenever the term digital forensics is talked about, developers or penetration testers usually depend on various third-party tools such as kali linux, Metasploit, Commando VM, Parrot Security OS, etc. But with python, Security Professionals can perform similar tasks without using third party tools like, Web server fingerprinting, Simulation of attacks, Port Scanning, Website cloning, Load generation and testing of a website, creating intrusion detection and prevention systems, wireless network scanning, transmission of traffic in the network, accessing mail servers and many more implementations in the area of digital fingerprinting and security applications.

Mainly used for,

Socket Programming

Python has a socket library, that enables developers with easy methods and functions for socket programming.

Network Port scanning

The common tool used for port scanning is nmap, but using the python's Socket library for socket programming, port scanning functionality can be made available on any machine with just a few lines of code, using python-nmap library.

Geolocation extraction

Python being so extensible, it supports real-time extraction of a geolocation of an IP address. This can be done by using Python's pygeoip module.

Many other networking and information security scripts, including exploit scripts can be written in python as well, using various libraries and frameworks such as Diesel Framework, Pulsar Framework, Twisted (an event driven Network engine), NAPALM (cross-vendor API to manipulate network devices.)

3.3 Python for Artificial Intelligence

Python is being widely used in the Artificial Intelligence discipline from the last 7 years now, and the most obvious reason for that is python's least code as compared to other languages. i.e with very less coding, developers can achieve so many functionalities simply by using utilities of so many readily available libraries. These Millions of Python libraries are present readily is only made possible because of Python being an open source multiparadigm language, it encouraged so many developers to switch to python from other languages, and this indeed has made Python language's developer community so huge in number, that every other day python developers around the world provide a comprehensive support and assistance through git-hub, blogs, forums or tutorials, which makes the job of coder very easy. Some General Python Libraries for **AI** include,

AIMA: - it is the Python Library for implementation of the algorithms from **Russell and Norvig's 'Artificial Intelligence: A modern Approach'**.

pyDatalog: - It is a logic programming engine in python.

SimpleAI: - It is the python library for implementation of various AI Algorithms of various AI Books, it provides easy, tested and well documented functionalities.

EasyAI: - It is a very simple, effective and easy python engine that simulates 2 player games with AI, like Negamax, transposition tables, game solving, etc.

NLTK: - It is an open source python module that is used for Natural language and text processing. It is used for linguistic data and documentation for research and development, and text analytics with support for Windows, Mac OSX as well as Linux.

gTTS: - It is a Python library and command line interface tool to interface with Google Translate text-to-speech API.

3.4 Python for Machine Learning & Deep Learning

AI, Machine Learning and Deep Learning are the future, and are the topics that are at a boom these days, and as is Python. These days Whenever the term Machine Learning comes in the picture, Python is the ultimate solution for the developers, this justifies the popularity of python as well as how power packed language it is.

As discussed earlier in this paper python being so flexible language incorporates almost all functionalities that anyone can think of, hence Python is ideal for AI, Machine Learning and Deep Learning because of the tools, frameworks, Packages, Modules and libraries that are readily available and also for the fact that one can always create one of its own for their use.

So, here is a brief look for all the mostly used Libraries for Machine Learning and Deep Learning.

3.4.1 Python Libraries & Frameworks for Machine Learning

PyBrain: - It is a flexible, simple yet advanced algorithms for ML Tasks. PyBrain is a modular ML Library that facilitates developers with so many different environments for testing and comparing algorithms.

PyML: - It is a very powerful framework written in python itself that simulates Support Vector Machine (SVM) and other kernel methods. It works on Linux and Mac OS X.

Scikit-Learn: - It is the most common, most popular, open source general purpose machine learning library. It is a very efficient tool for data analysis in Python.

MDP-Toolkit: - It is very famous data processing framework which is easily expandable. It contains a number of supervised and unsupervised learning algorithms, data processing units for data processing sequences and complex network architecture. Developers can not only use existing algorithms, tools and modules but It is also really very easy and efficient to implement new algorithms in this framework, which provides ultimate solutions to developers to overcome various Machine Learning problems.

Torch: - It is an open-source, easy-to-use Machine Learning Framework for Computing Machine Learning Algorithms. It facilitates users with a very effective GPU support. This Tool

comes with so many pre-trained models, N-dimensional arrays, linear algebra routines, and much more features based on a scripting language called Lua. It is a very productive tool which supports all the major platforms like Linux, android, Mac OS X, iOS, and Windows.

Accord.NET: - It's a Machine Learning Framework Written in C# especially for windows. it is well documented ML Framework that provides efficient audio and image processing. It is used for numerical optimization, artificial neural networks and visualization.

Azure ML Studio: - It is a Modern cloud platform for data scientists. It is used to develop Machine Learning Models in the cloud, with a wide variety of modeling configurations and algorithms. This is served by Azure Cloud platform; it serves 10GB of Storage space per account.

Amazon Machine Learning: - It is a Machine Learning service that enables developers with tools and wizards for developing ML models. It can be connected to data stored in Amazon S3, Redshift, or RDS.

Spark MLlib: - It is one of the popular Machine Learning library by Apache. It is used in Java, Scala, Python and R for processing large-scale data. It can be easily integrated with Hadoop workflows.

3.4.2 Python Libraries & Frameworks for Deep Learning

TensorFlow: - it is open-source software library built for Deep-Learning and artificial neural networks, by google. TensorFlow enables developers to create neural networks and facilitates efficient creation of computation models using flowgraphs. There is also TensorFlow Framework for C++ and Python.

Theano: - it is a python library Optimized for GPU, used for deep learning. It includes integration with NumPy, dynamic C code generation, and symbolic differentiation. Using Theano, developers can define and evaluate mathematical expressions, including multi-dimensional arrays.

Caffe: - It is one the most popular deep learning tool designed for building apps. It provides a good interface for Matlab/ C++ / Python. It allows instant creation of neural networks for a problem using text without writing code.

Microsoft CNTK: - It refers to Microsoft Cognitive Toolkit, it is an open-source framework for deep learning with C#, C++, Python Interface support. It in-corporates many algorithms like feed-forward, CNN, RNN, LSTM Sequence-to-sequence, etc.

Keras: - It is the most ideal open-source library and tool for fast prototyping because of its user-friendliness and modularity, it makes the process of creating new DL models very easy and efficient. It is highly optimized for both CPU nad GPU.

PyTorch: - It is the widely popular, highly extensible, Open-Source, Optimized tensor package that provides framework for deep learning models and scientific computing, using GPUs and CPUs. It has many inbuilt algorithms and pre-trained models which facilitates newbie developers with their first step to deep learning concepts.

3.5 Python for IoT

This Python's Innate multi-paradigm nature has made it so flexible, effective, efficient and easy to learn, that it has encouraged many developers to switch to python and perform

various tasks using python. Hence, Python has a very strong developer community that has brought other developers with packages and frameworks for almost anything, hence Python also supports IoT.

As IoT is the current and future trend, Python walks hand in hand with the trend providing tools, frameworks, libraries and packages for IoT implementation. IoT forces billions of devices to integrate into information systems, this is seamlessly achieved using python, its tools and frameworks.

Rasberry Pi, a very popular IoT device, has chosen python as its default user-programming language, because of its flexibility and power-packed tools, libraries and frameworks.

3.5.1 Some Python Libraries and Frameworks for IoT

CoAPhon: - it is an open-source python-based CoAP Library. In order to allow applications to interact with the physical objects, CoAP – Constrained Application Protocol acts as the technology enabler, so this CoAPhon library simplifies the development of CoAP-enabled IoT Applications as it facilitates a simple and easy-to-use programming interface for manipulation of CoAP as a communication protocol for instant prototyping and deploying of IoT systems. It also implements popular extensions such as block-wise transfer and resource observing.

Zorg: - It is a python framework with a simple and modular design which enables development for robotics and other IoT projects. It requires very less coding while building better robots as it becomes very easy to configure and control projects using Zorg. it also supports wide number of hardware components including various analog and digital sensors.

PyOTA: - PyOTA is an IOTA API for python, IOTA or Internet of Things Application, is a crypto technology that enables transactions between Devices on an IoT Network. PyOTA provides support for development of various IOTA for accomplishing various purposes and goals in IoT.

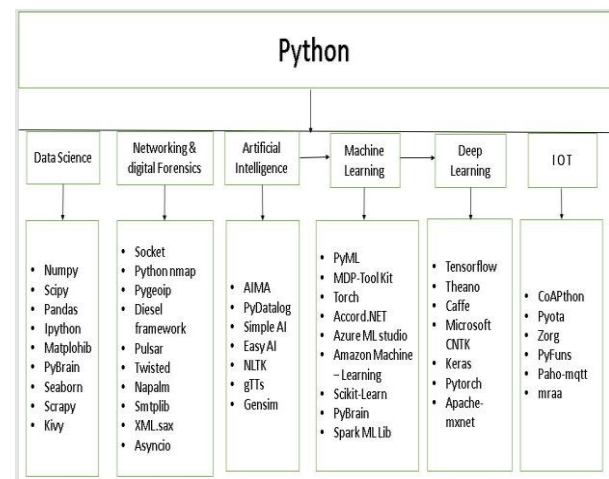


Fig 3: Python in Various Areas

4. ACKNOWLEDGMENTS

This Survey paper was supported by Smt. Chandaben Mohanbhai Patel Institute of Computer Application, CHARUSAT. I would like to show my gratitude to respected Dr. Atul M. Patel [Professor, Dean & Principal, CMPICA, CHARUSAT] for giving me this great opportunity and for sharing their pearls of wisdom and providing insights and expertise that greatly assisted the survey and research. I would

also like to extend my love and respect, and dedicate all my hardwork to my Parents, for always encouraging and guiding me towards the paths through which I can make myself better.

5. CONCLUSION

In this paper we began with introducing python as a Multiparadigm Programming Language. This paper discussed various characteristics, features and Programming language support offered by Python Programming Language which clearly stated that it is one of the most flexible language available in current time. Later this paper advanced over the investigation of how and why python is being used in various areas of the technology, and we concluded that python revolutionizes the development process of almost every kind of applications into a whole another level, the reasons for the same has also been cited in this paper.

6. REFERENCES

- [1] Python – The Fastest Growing Programming language, By K.R. Srinath [2017]
- [2] An Introduction to Python Programming for Research, by James Hetherington. [2019]
- [3] “Automate the Boring Stuff with Python”, by A L Sweigart.
- [4] “Violent Python, A cookbook for Hackers, Forensic Analysts, Penetration Testers and Security Engineers”, by TJ. O’Connor.
- [5] Why Python Rocks for Research, by Hoyt Koepke.
- [6] [https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))
- [7] <https://docs.python.org/3/>
- [8] <https://www.tensorflow.org/>
- [9] <https://pytorch.org/>
- [10] <https://github.com/>