Python 3

Python is a general-purpose high-level programming language. It is an open source language, released under a GPL-compatible license.

Python Overview

Python is a general-purpose high-level programming language. It is an open source language, released under a GPL-compatible license. Python Software Foundation (PSF), a non-profit organization, holds the copyright of Python.

Guido Van Rossum conceived Python in the late 1980s. It was released in 1991 at Centrum Wiskunde & Informatica (CWI) in the Netherlands as a successor to the ABC language. He named this language after a popular comedy show called 'Monty Python's Flying Circus' (and not after Python-the snake).

In the last few years, its popularity has increased immensely. According to stackoverflow.com's recent survey, Python is in the top ten Most Popular Technologies in 2018.

Official Web Site: https://www.python.org

Python Version History:

Currently, PSF supports two versions, Python 2.x & Python 3.x. Python 2.0 was released in October 2000 and includes a large number of features. PSF continues to support version Python 2 because a large body of existing code could not be forward ported to Python 3. So, they will support Python 2 until 2020.

Python 3.0 was released on December 3rd, 2008. It was designed to rectify certain flaws in earlier version. This version is not completely backward-compatible with previous versions. However, many of its major features have since been back-ported to the Python 2.6.x and 2.7.x version series. Releases of Python 3 include 2 to 3 utilities to facilitate the automation of translation of Python 2 code to Python 3.

The following table lists all the important versions history of Python:

Version	Release Date	Important Features
Python 0.9.0	February 1991	• Classes with inheritance exception handling • Functions • Modules
Python 1.0	January 1994	• Functional programming tools (lambda, map, filter and reduce). • Support for complex numbers. • Functions with keyword arguments
Python 2.0	October 2000	• List comprehension. • Cycle-detecting garbage collector. • Support for Unicode. Unification of data types and classes
Python 2.7.0 - **Current version**	July 2010	
Python 2.7.15 - **Current sub-version**	May 2018	
Python 3	December 2008	• Backward incompatible. • print keyword changed to print() function • raw_input() function depreciated • Unified str/Unicode types. • Utilities for automatic conversion of Pytthon 2.x code
Python 3.6	December 2016	
Python 3.6.5	March 2018	
Python 3.7.0 -	May 2018	• New C API for thread-local storage • Built-in breakpoint() • Data classes • Context variables
Python 3.8.0 **New Version**	Oct. 14, 2019	• Assignment expressions • PEP 570, Positional-only arguments • PEP 587, Python Initialization Configuration (improved embedding) • PEP 590, Vectorcall: a fast calling protocol for CPython • PEP 578, Runtime audit hooks • PEP 574, Pickle protocol 5 with out-of-band data • Typing-related: PEP 591 (Final qualifier), PEP 586 (Literal types), and PEP 589 (TypedDict) • Parallel filesystem cache for compiled bytecode • Debug builds share ABI as release builds • f-strings support a handy = specifier for debugging • continue is now legal in finally: blocks

		• on Windows, the default asyncio event loop is now ProactorEventLoop • on macOS, the spawn start method is now used by default in multiprocessing • multiprocessing can now use shared memory segments to avoid pickling costs between processes • typed_ast is merged back to CPython • LOAD_GLOBAL is now 40% faster • pickle now uses Protocol 4 by default, improving performance

Python Features:

- Python is an interpreter-based language, which allows execution of one instruction at a time.
- Extensive basic data types are supported e.g. numbers (floating point, complex, and unlimited-length long integers), strings (both ASCII and Unicode), lists, and dictionaries.
- Variables can be strongly typed as well as dynamic typed.
- Supports object-oriented programming concepts such as class, inheritance, objects, module, namespace etc.
- Cleaner exception handling support.
- Supports automatic memory management.

Python Advantages

- Python provides enhanced readability. For that purpose, uniform indents are used to delimit blocks of statements instead of curly brackets, like in many languages such as C, C++ and Java.
- Python is free and distributed as open-source software. A large programming community is actively involved in the development and support of Python libraries for various applications such as web frameworks, mathematical computing and data science.
- Python is a cross-platform language. It works equally on different OS platforms like Windows, Linux, Mac OSX etc. Hence Python applications can be easily ported across OS platforms.
- Python supports multiple programming paradigms including imperative, procedural, object-oriented and functional programming styles.
- Python is an extensible language. Additional functionality (other than what is provided in the core language) can be made available through modules and packages written in other languages (C, C++, Java etc)
- A standard DB-API for database connectivity has been defined in Python. It can be enabled using any data source (Oracle, MySQL, SQLite etc.) as a backend to the Python program for storage, retrieval and processing of data.

- Standard distribution of Python contains the Tkinter GUI toolkit, which is the implementation of popular GUI library called Tcl/Tk. An attractive GUI can be constructed using Tkinter. Many other GUI libraries like Qt, GTK, WxWidgets etc. are also ported to Python.
- Python can be integrated with other popular programming technologies like C, C++, Java, ActiveX and CORBA.

Python Application Types

Even though Python started as a general-purpose programming language with no particular application as its focus, over last few years it has emerged as the language of choice for developers in some application areas. Some important applications of Python are summarized below:

Data Science

Python experienced a recent emergence in popularity charts mainly because of its Data science libraries. Huge amount of data is being generated today by web applications, mobile applications and other devices. Companies need business insights from this data.

Today Python has become the language of choice for data scientists. Python libraries like NumPy, Pandas and Matplotlib are extensively used in the process of data analysis, including the collection, processing and cleansing of data sets, applying mathematical algorithms and generating visualizations for the benefit of users. Commercial and community Python distributions by third-parties such as Anaconda and ActiveState provide all the essential libraries required for data science.

Machine Learning

This is another key application area of Python. Python libraries such as Scikit-learn, Tensorflow and NLTK are widely used for the prediction of trends like customer satisfaction, projected values of stocks etc. Some of the real-world applications of machine learning include medical diagnosis, statistical arbitrage, basket analysis, sales prediction etc.

Web Development

This is another application area in which Python is becoming popular. Web application framework libraries like django, Pyramid, Flask etc. make it very easy to develop and deploy simple as well as complex web applications. These frameworks are used extensively by various IT companies. Dropbox for example uses django as a backend to store and, synchronize local folders.

Most of the web servers today are compatible with WSGI (Web Server Gateway Interface) - a specification for the universal interface between Python web frameworks and web servers. All leading web servers such as Apache, IIS, Nginxetc can now host Python web applications. Google's App Engine hosts web applications built with almost all Python web frameworks.

Image Processing

The OpenCV library is commonly used for face detection and gesture recognition. OpenCV is a C++ library, but has been ported to Python. Because of the rapid development of this feature, Python is a very popular choice from image processing.

Game Development

Python is a popular choice for game developers. The PyGame library is extensively used for building games for desktop as well as for mobile platforms. PyGame applications can be installed on Android too.

Embedded Systems and IoT

Another important area of Python application is in embedded systems. Raspberry Pi is a very popular yet a low-cost single-board computer. It is being extensively used in automation products, robotics, IoT, and kiosk applications. Popular microcontrollers like Arduino are used in many IoT products and are being programmed with Python. A lightweight version of Python called Micropython has been developed especially for microcontrollers. A special Micropython-compatible controller called PyBoard has also been developed.

Android Apps

Although Android apps are predominantly developed using Android SDK, which is similar to Java, Python can also be used to develop Android apps. Python's Kivy library has all the functionalities required for a mobile application.

Automated Jobs

Python is extremely useful and widely used for automating CRON (Command Run ON) jobs. Certain tasks like backups, defined in Python scripts can be scheduled to be invoked automatically by the operating system scheduler to be executed at predefined times.

Python is embedded as a scripting language in many popular software products. This is similar to VBA used for writing macros in Excel, PowerPoint, etc. Python API is integrated with Maya, PaintShop Pro, etc.

Rapid Development Tool

Standard distribution of Python as developed by Rossum and maintained by Python Software Foundation is called CPython which is a reference implementation. Its alternative implementations - Jython the JRE implementation of Python and IronPython - the .NET implementation, interact seamlessly with Java and C#, respectively. For example Jython can use all Java libraries such as Swing etc. So the development time can be minimized by using simpler Python syntaxes and Java libraries for prototyping the software product.