
Introduction to Python

1. Definition of Python:

- Python is a general-purpose, **interpreted**, high-level programming language.
- Python:
 - **High-level language** – It is a **human-readable language** which easily understandable and easy to debug
 - **Interpreted as well as compiled** – Python is basically called an interpreted language, **but can also be called** a compiled interpreted language. In Python, first, all the source code is at **once converted to byte code** which is the action of the compiler, and then the byte code is executed by the interpreter.
 - Python is also called a **scripting** language.
 - Python supports **functional** programming as well as **Object-oriented** programming approach.

Note: A scripting language is a programming language that is interpreted. These types of languages are used in automation, etc. These languages use interpreters and are executed line by line.

2. History or Origin of Python

- It was invented in the **Netherlands** in the **early '90s**.
- **Guido Van Rossum** was the creator of this beautiful language.
- Guido released the first version **in 1991**.
- Python was derived from the **ABC programming language**, a general-purpose programming language.
- It is **open-source** software that can be downloaded freely and the code is customizable as well.
- **Why the name Python?**
 - There was a **TV show** by the name **Monty Python's Flying Circus** which was a very much popular fun show in the 1970s.
 - While creating Python, **Guido** also used to read this show's published scripts. Guido needed a short and a **different name** for this language, hence he named it "Python".

-

3. Features of Python

1. **Simple:** Python syntax is very easy. Developing and understanding python is very easy than others.
2. **Open Source** – We can **download freely** and customize the code as well
3. **Dynamically typed** – Dynamically type will be assigned to data.
4. **Platform independent:** Python programs are not dependent on any specific operating systems. We can run on all operating systems happily.
5. **Portable:** If a program **gives the same result** on any platform then it is a portable program. Python used to give the same result on any platform.
6. **Huge library** – Python has a big library to fulfill the requirements.
7. **Database connectivity** – Python provides interfaces to connect with all major databases like oracle, MySQL
8. **Batteries included** – Python provides **inbuilt libraries** called **batteries**. Some of them are below
 1. Boto – amazon web services library
 2. MySQL-connector-python – To connect with MySQL
 3. NumPy – To process arrays
 4. Pandas – powerful data structures for data analysis, time series, and statistics

4. Application area of Python

Python is being used to develop:

- **Standalone applications:** An application that **needs to install** on every machine to work with that application.
- **Web applications:** An application that follows a **client-server architecture**. The client is a program, which sends requests to the server. The server is a program, mainly it can do three things. Captures the request from the client, process the request, and Sends the response to the client
- Database applications.
- To process huge amounts of data. Eg Hadoop, Spark.
- Machine learning.
- Artificial Intelligence.
- Data science.
- Network servers.
- IoT
- Application scripting etc.

5. Different flavors of python

- Flavors of python refer to the **different types of python compilers**. These are useful to **integrate** various programming languages.
1. **CPython: Standard python compiler** implemented in C language. This is the python software being downloaded and used by the programmers directly.
 2. **Jython** : Initially called as **JPython**, later renamed to **Jython**. Designed to run on a Java program. Write **Python code** that **runs on the Java platform (JVM)**. Python Syntax + Java Libraries.
 3. **IronPython**: Designed for **.NET framework**.
 4. **PyPy**: The main advantage of PyPy is performance will be improved because the JIT (Just-In-Time) compiler is available inside PVM.
 5. **RubyPython**: This is a **bridge** between **Ruby and Python** interpreters. It encloses a python interpreter inside the Ruby application.
 6. **AnacondaPython**: Anaconda is a **free and open-source** Python programming language. This is mainly for **data science** and **machine learning-related** applications (large-scale data processing, predictive analytics, scientific computing). This aims to **simplify package management and deployment**.

6. Python Installation procedure For Windows:

- **Step 1: Install Python**
- **Check if Python is already installed:**
 - Open Command Prompt (press Win + R, type cmd, and press Enter).
 - Run **python --version** or python3 --version.
 - If a version number appears (e.g., Python 3.12.7), Python is installed.
 - If we get an error (e.g., “python’ is not recognized”), proceed to install Python.
- **Download the Python installer:**
 - Visit the official Python website: python.org/downloads.
 - Click the “Download Python” button to get the latest version (e.g., Python 3.12.7).
- **Run the installer:**
 - Double-click the downloaded .exe file.
 - In the installer:
 - **Check “Add Python to PATH”** at the bottom of the first screen.
 - Select **“Install Now”** for default settings, or choose “Customize Installation” to:
 - Ensure **“pip”** and **“tkinter”** are selected (tkinter is included by default for GUI development).
 - Install for **all users** (optional, requires admin privileges).
 - Click **“Next”** and complete the installation (default path is typically C:\Users\<YourUsername>\AppData\Local\Programs\Python\Python312).
 - Note the installation path if customized.

6. Python Installation procedure For Windows:

- **Verify Python installation:**
 - Open a new Command Prompt.
 - Run `python --version` (or `python3 --version`).
 - Expected output: Python 3.12.7 (or similar).
 - Run `pip --version` to verify pip (Python's package manager): e.g., pip 24.2 from

Step 2: Set Up Environment for Python GUI Development

- **Choose a code editor or IDE:**
 - **Recommended IDEs:**
 - **PyCharm:** Download from jetbrains.com/pycharm (Community Edition is free).
 - **VS Code:** Download from code.visualstudio.com, install the Python extension.
 - **IDLE:** Included with Python, launch by typing `idle` in Command Prompt.
 - **Text editor:** Notepad++ or any editor for simple scripts.
 - Install and configure your chosen IDE (e.g., set Python interpreter in PyCharm or VS Code to the installed Python path).

7. Program execution steps

1. First write the code in **notepad or IDE** (Initially, it would be **better** to write the code in **notepad** if you are new to python)
2. Save the file with **.py or .python** extension.
3. Run or execute the program (For windows execute the file from command prompt and for linux from terminal)
4. Command for execution – “**python filename.py**” or “**py filename.py**”