Getting Started with Python

Python, a versatile, high-level programming language, was created by Guido van Rossum and first released in 1991. It is designed for general-purpose programming and features a dynamic type system and automatic memory management. Python supports multiple programming styles, including object-oriented, imperative, functional, and procedural, and comes with a vast, comprehensive standard library.

Python Versions

Python 3.x: This is the current version and is actively being developed.
Python 2.x: This is the legacy version. It ceased receiving new features and only receives security updates until 2020. Although migration to Python 3 is becoming simpler, many projects still utilize Python 2.

Both versions can be downloaded and installed from the official site. Third-party providers also offer pre-packaged distributions that include common libraries for areas like math, data analysis, or scientific computing, easing the initial setup process. A comparison between the versions is available in the Python 3 vs. Python 2 documentation.

Verifying Installation

To confirm a successful Python installation, open your preferred terminal and run the following command:
$ python --version

If Python 3 is the default version (Version ≥ 3.0):
Python 3.6.0

If Python 2 is the default version (Version ≤ 2.7):
Python 2.7.13

On systems like MacOS or many Linux distributions, python --version may still show Python 2 even if Python 3 is installed. In this case, use python3 to explicitly launch the Python 3 interpreter.

Running "Hello, World"

Python code can be executed using IDLE, a script file, or the interactive shell.

Using IDLE

IDLE is a basic editor bundled with the standard Python installation. Open IDLE, wait for the >>> prompt, then type:
>> print("Hello, World")
The output will be:
Hello, World

Running a Python Script File

Create a file named hello.py and add:
print('Hello, World')

Run the file using:
$ python hello.py
Output:
Hello, World

Using the Interactive Python Shell (REPL)

Run python or python3 in the terminal to start the interactive shell. This allows real-time execution of Python code.

To exit the shell use:
exit() or **quit()** or press **CTRL + D**

To cancel a command use **CTRL + C**.

Other Ways to Run Python Code

Run as a string:
$ python -c 'print("Hello, World")'

Online shells allow Python execution without local installation.

PIP is used for package management using pip install <package name>.

IDEs like PyCharm and tools like Jupyter Notebooks are used for larger projects.

Python Tutor helps visualize code execution, and PEP8 provides formatting guidelines.