

# Python

\* what is python and why is it called an ~~interpreted~~ language?

Python is a high-level, general-purpose programming language known for its simplicity and readability. It is called an interpreted language because python programs are executed line by line by the python interpreter, rather than being fully compiled into machine code beforehand. This allows for easier debugging, faster development, and platform independence.

\* what are the key features of python that make it popular for beginners and professionals?

- easy to learn / simple syntax.
- open source.
- platform independent
- extensive libraries.
- indented programming.
- strong community support.
- interpreted language

\* what is the difference between python 2 and python 3?

Python 2

python 3

Python 2 was released in 2000. Python 3 was released 2008

- |   |   |
|---|---|
| 1) It is legacy                               | 2) It is present and future.                                      |
| 3) It is a print statement<br>(print "Hello") | 3) It uses print() as function<br>(print("Hello")).               |
| 4) It stores text as ASCII                    | 4) It uses unicode, making it better for international languages. |

5) It is reached its end life in 2020.

5) It is actively developed and supported.

6) It is outdated and no longer maintained.

6) It is the modern, improved, and supported version.

\* what are Python's applications in real-world projects?

- web development - using frameworks like Django and Flask.
- Data science & Analytics - Libraries like pandas, numpy, and matplotlib.
- Machine learning & AI - with TensorFlow, PyTorch, and scikit-learn.
- Automation & scripting - writing scripts to automate repetitive tasks.
- Game development - using libraries like pygame.
- Desktop Applications - with Tkinter, PyQt, or Kivy.
- Networking & cybersecurity - for penetration testing and network automation.
- IoT (Internet of things) - running on small devices like Raspberry Pi.
- Graphical design GUI graphical user interface -
  - Python supports GUI APP development, image editing, and visualization, making it useful in graphical design and creative projects.
- Deep learning - Deep learning is a way of computers to learn and make decisions like humans by using neural networks on big data.

\* what is PEP 8 and why is it important in python programming? [Python Enhancement proposal 8]

PEP 8 is the official style guide for python code. It provides a set of conventions and best practices for writing clean, consistent, and readable python programs. PEP stands for python enhancement proposal, and PEP8 specially focuses on code formatting standards such as [indentification] indentation, naming conventions, line length, spaces, and documentation.

Importance in python programming:-

1. Readability :- Following PEP 8 makes code easier to read and understand, both the original developer and for others working on the same project.
2. Consistency :- It ensures uniform coding style, making python code easier to read, maintain, and collaborate on.
3. Collaboration :- Teams can work together smoothly when everyone follows the same coding style.
4. Maintainability :- well-structured code is easier to debug, update, and extend.
5. Professionalism :- following PEP 8 is considered a standard practice in the python community, showing that you write clean and professional code.

\* who developed python and in which year was it released?

Python was developed by Guido van Rossum, and it was first released in 1991.

\* what do you mean by "dynamically typed" in python?

In python, being dynamically typed means that the type of a variable is determined at runtime rather than being declared explicitly. You don't need to specify the data type while creating a variable; Python automatically

assigns the type based on the value. This provides flexibility but also requires careful handling to avoid type-related errors.

\* What is the difference between a compiler and an interpreter, and which does Python use?

A compiler translates the entire source code into machine code before execution, while an interpreter translates and executes code line by line. Python primarily uses an interpreter, which allows immediate execution of code and makes it easier for debugging and rapid development.