

You will learn about the Python Language and the main differences to Compiled Languages such as Java.

---

#### 1.1.1. What is Python?

- Python is a high-level, **interpreted programming language** known for its **simplicity** and **readability**.
  - Python's emphasizes **code focuses** on **minimizing** the **effort required to write and maintain code**.
  - Python offers a clean and elegant syntax that allows programmers to express concepts in fewer lines of code compared to other programming languages.
  - This simplicity and readability make Python an **excellent choice for fields such as Machine Learning**.
  - The language provides a wide range of built-in data structures and supports various programming paradigms, including procedural, object-oriented, and functional programming.
  - It also offers a vast collection of third-party libraries and frameworks due to the large community support and contribution, making it versatile and suitable for various application domains.
  - It is widely used in web development, data analysis, scientific computing, artificial intelligence, machine learning, automation, and more.
- 

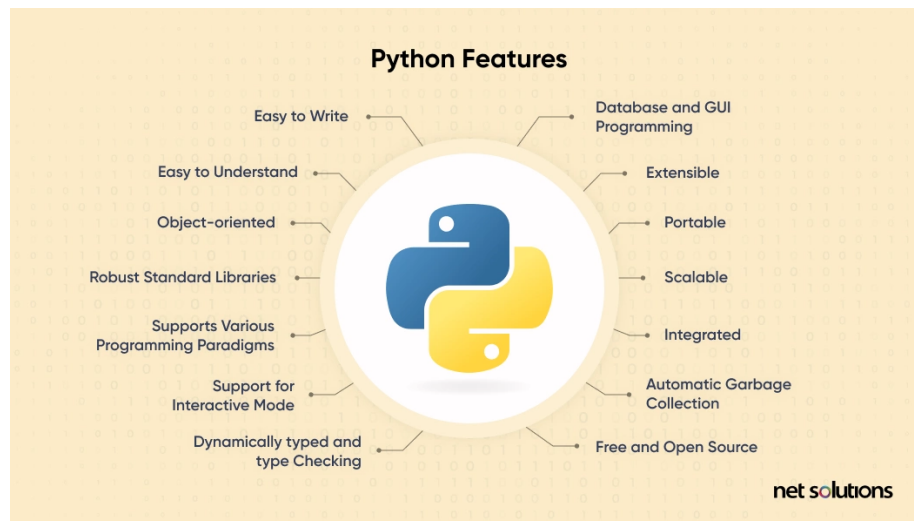
#### 1.1.2. Differences between Python and Java (syntax, typing, ecosystem)

Aspect	Python	Java (& similar languages)
Syntax	<ul style="list-style-type: none"><li>- Simple, clean, and readable syntax using indentation to define code blocks.</li><li>- Code is often much shorter and more concise.</li></ul>	<ul style="list-style-type: none"><li>- Verbose syntax using curly braces and semicolons to define code blocks.</li></ul>
Typing	<ul style="list-style-type: none"><li>- Dynamically typed: variable types determined at runtime without explicit declarations.</li></ul>	<ul style="list-style-type: none"><li>- Statically typed: requires explicit type declaration before compilation.</li></ul>
Execution-	<ul style="list-style-type: none"><li>- Interpreted language executing code line-by-line, allowing rapid development but slower runtime.</li></ul>	<ul style="list-style-type: none"><li>- Compiled into bytecode, executed on JVM, generally faster performance than Python.</li></ul>

Aspect	Python	Java (& similar languages)
Ecosystem and Use Cases	Extensive ecosystem optimized for data science, AI, machine learning with libraries like NumPy, pandas, TensorFlow, scikit-learn.	- Dominates enterprise applications, Android development, large-scale systems with strong performance, multithreading, security.
Learning Curve	- Easier and faster to learn due to simplicity, good for beginners transitioning from Java.	- Steeper learning curve due to strict syntax and static typing but aids in maintainability.

### 1.1.3. Why Python is the language of choice for AI

- Simple and readable syntax allows engineers and researchers to focus on AI problem-solving rather than complex coding.
- Extensive ecosystem of AI libraries like TensorFlow, PyTorch, scikit-learn, NumPy, and pandas accelerates development by providing pre-built tools and models. (Examples on each will be provided in this workshop)
- Rapid prototyping capabilities allow quick iteration and experimentation, essential in AI development workflows.
- Cross-platform compatibility lets Python AI models run on Windows, macOS, Linux, and more without code modification.
- Seamless integration with other languages and big data tools enhances scalability and deployment options for AI systems.
- Open-source nature encourages collaborative innovation and gives free access to cutting-edge AI technologies.
- Widely adopted by tech giants and research communities, making it a standard in AI research and industrial applications.



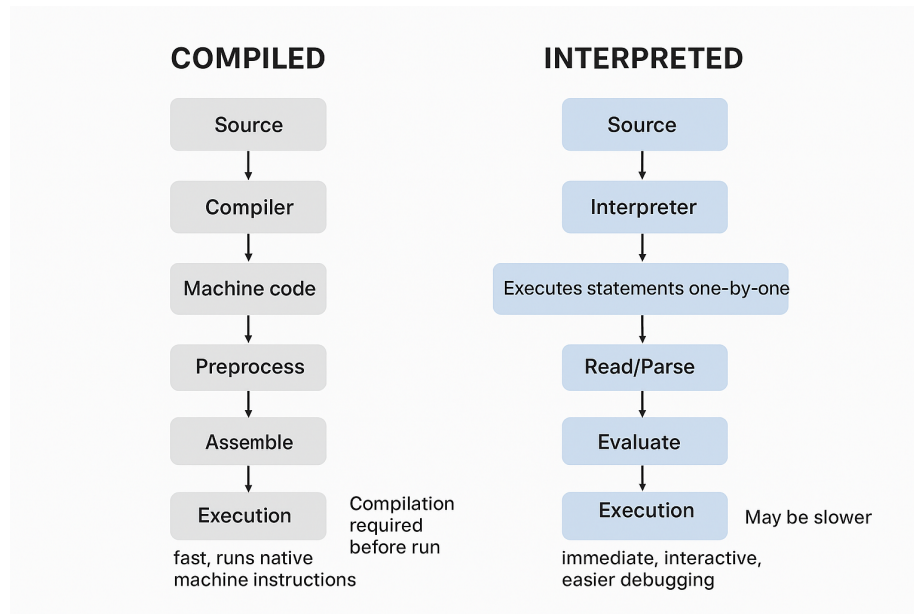
**Compiled language vs Interpreted language** Main difference is: How the source code is mapped and translated to executable instructions to the computer.

1. Compiled language:

- The entirety of the source code is translated to machine language (binary executable) before running the program.
- The binary executable can run directly on the operating system without needing the original source code.
- It is platform dependent; can run only on architecture it was compiled for.
- Faster execution speeds.

2. Interpreted language:

- Line by Line is translated into machine language at the runtime; this is done using a program called interpreter.
- Platform independent.
- Slower execution.
- More dynamic and flexible.



### Extra Read.

**Python's popularity and areas of application** Python has experienced remarkable growth in popularity and has established itself as one of the most widely used programming languages across various domains. Its versatility and simplicity have contributed to its widespread adoption by developers, data scientists, researchers, and hobbyists alike.

Python's popularity can be attributed to several key factors:

1. Readability and Ease of Use

Python's clean and intuitive syntax promotes readability, making it easier for developers to understand and write code. Its focus on simplicity allows programmers to express concepts using fewer lines of code compared to other languages, reducing the time and effort required for development.

2. Extensive Ecosystem of Libraries and Frameworks

Python boasts a rich ecosystem of third-party libraries and frameworks, offering solutions for a wide range of tasks. These libraries, such as NumPy, Pandas, Matplotlib, TensorFlow, and Django, enable developers to leverage pre-existing functionalities and accelerate development, making Python a popular choice for scientific computing, data analysis, machine learning, web development, and more.

3. Data Science and Machine Learning

Python has become the de facto language for data science and machine learning projects. Its libraries, like NumPy, Pandas, Scikit-learn, and TensorFlow, provide powerful tools for data manipulation, analysis, modeling, and deep learning. The simplicity of Python syntax combined with the availability of these libraries has attracted a large community of data scientists and machine learning practitioners.

#### 4. Web Development

Python offers robust frameworks such as Django and Flask, which streamline web development and allow developers to create scalable and secure web applications. The clean and readable syntax of Python facilitates rapid prototyping and development, making it a popular choice for building web-based solutions.

#### 5. Automation and Scripting

Python's ease of use and cross-platform compatibility make it ideal for automation and scripting tasks. Whether it's automating repetitive tasks, writing system scripts, or performing file operations, Python provides a flexible and efficient platform for automation across different operating systems.

#### 6. Scientific Computing and Research

Python's rich scientific computing libraries, along with its simplicity and integration capabilities, have made it a go-to language for researchers and scientists. Python's usage extends to diverse fields, including physics, biology, astronomy, chemistry, and more, where it serves as a powerful tool for data analysis, simulations, and research workflows.

#### 7. Education and Community

Python's beginner-friendly nature and extensive documentation have made it a popular choice for teaching programming and computer science. Its gentle learning curve and focus on readability help novices grasp fundamental programming concepts quickly. Additionally, Python has a vibrant and inclusive community that actively supports newcomers, offering resources, forums, and educational initiatives.

These are just a few examples of the areas where Python excels. Its versatility, ease of use, and vast ecosystem of libraries and frameworks continue to attract developers and enthusiasts across industries, solidifying Python's position as a top programming language.