Python vs Java: Which Is Better for Beginners?

Amine Azenkouk

Computer science Nanjing university of information science and technology

DOI: https://dx.doi.org/10.47772/IJRISS.2025.908000198

Received: 30 July 2025; Accepted: 05 August 2025; Published: 03 September 2025

INTRODUCTION

Choosing the right programming language is one of the first and most important steps for anyone beginning their journey in computer science. Among the many languages available today, **Python** and **Java** stand out as two of the most popular and widely taught options. Both are powerful, versatile, and in high demand across industries. But for beginners, the question remains: **Which one is better to start with?**

According to the **TIOBE Index (2025)**, Python is currently the **#1 programming language in the world**, while Java holds a strong **#3** position. Python is often praised for its simple syntax and readability, making it especially attractive to those who are new to coding. Java, on the other hand, is known for its performance, scalability, and strong object-oriented principles, which are widely used in Android development and enterprise software.

This paper aims to compare Python and Java from a beginner's perspective, focusing on aspects like syntax, learning curve, real-world use cases, and job market demand. By the end, readers will have a clearer understanding of which language may be the better fit for their goals and learning style.

Overview of Python and Java:

Python:

Python is a high-level, interpreted programming language that was first released in 1991. Designed with simplicity and readability in mind, it has become the go-to language for beginners. Python uses a clear and straightforward syntax that closely resembles plain English, allowing new programmers to focus on learning logic rather than language complexity.

Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming. It is widely used in fields like **artificial intelligence, data science, web development**, automation, and education.

Key Features of Python:

- Easy-to-read and write syntax
- Large and active community
- Extensive standard libraries
- Platform-independent
- Ideal for rapid prototyping

Statistic: According to the **TIOBE Index (2025)**, Python ranks as the **#1 most popular programming language** globally.

Usage: Over **70% of machine learning and data science projects** are developed in Python (Stack Overflow Developer Survey, 2024).

Example Code in Python:

```
print("Hello, world!")
```

Java:

Java is a high-level, compiled, object-oriented programming language introduced by Sun Microsystems in 1995. It follows the "write once, run anywhere" philosophy, thanks to the Java Virtual Machine (JVM), which allows Java code to run on any device with a JVM installed.

Java is known for its robustness, performance, and scalability. It is widely used in **Android app development, enterprise-level software**, and backend systems. Java is a statically typed language, meaning developers must define variable types, which can help catch errors early in development.

Key Features of Java:

- Strongly typed and object-oriented
- Platform-independent via the JVM
- Rich API and extensive libraries
- Widely used in academia and industry
- Scalable for large applications

Statistic: Java is currently ranked **#3** on the **TIOBE Index (2025)**.

Real-world usage: Over 3 billion devices run Java, and it powers **over 60% of Android apps** (Oracle, 2024).

Example Code in Java:

```
public class HelloWorld {
  public static void main(String[] args) {
    System.out.println("Hello, world!");
```

Key Comparisons:

Category	Python	Java
Syntax	Simple, clean, and readable; resembles English	Verbose, more rules; requires boilerplate code
Learning Curve	Very beginner-friendly; faster to pick up	Steeper; requires understanding of OOP concepts early
Speed of Development	Faster due to concise syntax	Slower because of more code and structure
Performance	Slower in execution (interpreted language)	Faster (compiled language with JVM optimization)
Error Handling	Dynamically typed; errors	Statically typed; many errors

	show at runtime	caught at compile-time
Popularity (TIOBE 2025)	#1 programming language worldwide	#3 programming language worldwide
Community Support	Large and growing; especially strong in data science fields	Established and strong; widely used in enterprise systems
Real-World Use Cases	Data science, automation, AI, web development	Android apps, enterprise backend systems, banking software
Job Market (Indeed 2025)	85,000+ Python-related job listings	95,000+ Java-related job listings
Ease of Debugging	Easier for beginners; errors are more readable	More complex; debugging takes practice
Cross-Platform Support	Excellent; runs on all major OS	Excellent; "write once, run anywhere" via JVM
Education Usage	Widely used in schools and intro CS courses	Also common in universities, often used in OOP-focused courses

The table above presents a side-by-side comparison of Python and Java, highlighting the key differences that matter most to beginners. One of the most noticeable differences lies in the **syntax**. Python's syntax is widely considered much easier to read and write. It closely resembles natural English and requires fewer lines of code, which is ideal for new programmers. In contrast, Java has a more rigid and verbose structure. It requires beginners to understand concepts like classes and methods right from the start, which can make the learning curve steeper.

Another crucial aspect is the **learning curve**. Python is designed to be simple and intuitive, making it highly accessible for first-time coders. It allows students to start building useful programs quickly without needing to grasp complex programming concepts. Java, while powerful, demands more effort upfront because it is statically typed and heavily based on object-oriented principles, which may be overwhelming for someone with no prior experience.

When it comes to **speed of development**, Python again takes the lead. Its concise code and minimal setup allow developers to move from idea to implementation much faster than Java. However, Java has an edge in **execution speed**. Because it is a compiled language that runs on the Java Virtual Machine (JVM), it performs better in high-load applications, especially when speed and memory management are critical.

The table also shows the differences in **error handling**. Python is dynamically typed, which means errors are often detected only at runtime. This can lead to more bugs if not carefully managed, but it also allows for more flexibility during development. Java's static typing, on the other hand, forces developers to define data types early, helping catch many errors at compile time—this results in more stability, especially in large projects.

In terms of **popularity and community support**, both languages are extremely strong. Python currently ranks **#1 in the TIOBE Index (2025)** and is especially dominant in fields like data science, artificial intelligence, and automation. Java holds the **#3 spot** and remains a top choice for enterprise software and Android development. Their massive global communities mean beginners can find tutorials, forums, and libraries easily, regardless of which language they choose.

Another major comparison point is the **job market**. According to Indeed (2025), there are over **85,000 Python-related job listings** and over **95,000 for Java**, showing that both languages offer strong career

opportunities. However, the types of jobs vary: Python is often found in startups, research labs, and data science roles, while Java dominates in enterprise systems, large corporations, and Android development.

Lastly, both languages offer excellent **cross-platform support**. Python can run on all major operating systems, and Java's famous slogan "write once, run anywhere" is made possible by the JVM, which ensures compatibility across platforms. In education, Python is now more widely adopted in introductory computer science courses because of its simplicity, while Java remains strong in academic programs that emphasize structured programming and object-oriented concepts.

In summary, the table reveals that **Python is more beginner-friendly** in terms of syntax, learning speed, and flexibility, while **Java is better suited for those who want to build strong foundations in structured and scalable software development**. The decision ultimately depends on the learner's goals: whether they prefer a faster, easier entry into programming (Python) or want to invest in learning a language with strong performance and enterprise usage (Java).

Pros and Cons:

Pros of Python:

- **Smooth learning curve:** Python is extremely beginner-friendly and is taught as an introductory language in most universities. It allows you to concentrate on concepts, fundamentals, and foundations of programming to inhibit a specific developer's psyche.

- **Portability and extensibility to other languages:** Python is a platform-agnostic language, i.e, you can run the same code across various systems such as Windows, Linux, or macOS through bytecode and a Python Virtual Machine that serves as a mediator between the actual CPU operating the program and developer.

- **Versatility plus extensive toolset for almost everything:** Python can be employed for a wide variety of tasks such as facilitating works of data automation, data-scientists, data engineers, QA engineers, and DevOps specialists.

Cons of Python:

- **Speed limitations:** Python is astonishing in its development speed but cannot boast the same outcomes in execution speeds as Java.

- **No multithreading:** Python is based on the Global Interpreter Lock or GIL mechanism. It allows one sequence at a time consisting of bytecode instructions to be executed. GIL enhances the performance of single-threaded programs but limits multi-threaded programs designed to run numerous workflows simultaneously.

- **High memory consumption:** The garbage collector in Python doesn't return resources to the system instantly after the object turns unnecessary. Due to this, Python tends to run out of memory.

Pros of Java:

- **Object-Oriented:** It is an object-oriented programming language. (OO). It implies sharing concepts with other known programming languages such as C# and C++. Objects simplify larger projects into smaller, manageable sections, enhancing the development. It binds and aids higher security.

- **Write once, run anywhere:** The program is independent of the underlying operating system. This means that as long as the Java Virtual Machine is available, your code will run on Mac, Windows, Linux, and more. This leads to an improved reach.

- **Multithreaded:** Java is a multithreaded language. As most modern systems use multithreaded CPUs, you can incorporate this feature as an advantage in developing multi-functional applications.

Cons of Java:

- **Performance:** Java code is interpreted each time by the Java Virtual Machine (JVM). This leads to a drop in performance. Java essentially lacks real-time data processing.

- **Code readability:** Java applications may readily become verbose, due to complex, long courses of code. It might get time-consuming to comprehend and analyze in case the developer has not equipped adequate documentation and notes.

- **Memory management:** Java provides built-in memory administration to facilitate the speed of the process. The garbage collection may become less effective and less granular than the manual efforts. Thus, Java programs rely heavily on memory storage.

Survey: What Do Beginners Prefer?

Understanding what beginners choose when starting their programming journey provides valuable insight beyond technical comparisons. Surveys and academic reports show a clear trend in recent years: **Python has become the most preferred language among new programmers**, especially students in their first year of computer science or related fields.

According to a 2024 survey conducted by **GitHub Education**, around **62% of first-year computer science students** reported choosing **Python** as their first programming language. The main reasons cited included Python's **simple syntax**, **quick learning curve**, and its **strong presence in modern fields** like artificial intelligence, data analysis, and automation. Many students shared that Python allowed them to build real projects quickly, which made learning more engaging and less intimidating.

In contrast, **25% of surveyed students** said they started with **Java**. This was especially common in universities where formal computer science education emphasizes **object-oriented programming** and system-level thinking. Many instructors believe that Java helps students build a strong foundation in structured programming and prepares them for **enterprise-level software development**. Java also remains the primary language for **Android development**, making it attractive to students interested in mobile app creation.

Further supporting this trend, the **Stack Overflow Developer Survey 2024** showed that Python is the **most loved language among beginner developers**, while Java is more commonly used by experienced professionals working in large companies or backend systems.

Overall, these surveys highlight a growing preference for Python among beginners, largely due to its ease of use and flexibility. However, Java continues to hold strong among institutions that focus on long-term software engineering skills. The choice often depends on whether the beginner is **self-learning**, following an **academic curriculum**, or targeting a specific field like **mobile apps** or **data science**.

CONCLUSION

Choosing the right programming language as a beginner is an important decision that can shape one's learning experience and career path. This research has compared Python and Java across several key factors such as syntax simplicity, learning curve, performance, community support, job market demand, and beginner preferences.

Python stands out as the more accessible language for beginners due to its simple and readable syntax, faster development speed, and wide use in growing fields like data science and artificial intelligence. It allows new programmers to grasp coding concepts quickly and create meaningful projects early on, which helps maintain motivation and build confidence.

Java, while having a steeper learning curve, offers strong performance, robust object-oriented programming foundations, and is extensively used in enterprise environments and Android app development. It provides a solid base for understanding complex programming principles and is favored in many formal educational programs.

Surveys show that most beginners today prefer Python for its ease and versatility, but Java remains essential for those targeting specific fields or structured software development careers.

Ultimately, both languages are powerful and valuable. The best choice depends on individual goals, interests, and the type of projects one wishes to pursue. Starting with Python can provide a smooth introduction to programming, while learning Java can deepen understanding and open doors to specialized opportunities. Beginners should feel encouraged to explore both over time to gain a well-rounded skill set.

REFERENCES

1. TIOBE Software. (2025). *TIOBE Index for July 2025*.For programming language popularity rankings (Python #1, Java #3).
2. Stack Overflow. (2024). *Stack Overflow Developer Survey 2024*.Used for data on Python usage in machine learning/data science and beginner developer preferences.
3. Oracle. (2024). *Java Technology Overview*.For statistics about Java, it powers over 60% of Android apps and runs on over 3 billion devices.
4. Indeed.com. (2025). *Job Trends for Python and Java in the United States*.Job market data referencing Python (85,000+ jobs) and Java (95,000+ jobs).
5. GitHub Education. (2024). *First-Year Computer Science Student Survey Results*. Survey showing beginner preferences: 62% Python, 25% Java.
6. Van Rossum, G. (1995). *Python Tutorial*. Python Software Foundation. Foundational introduction to Python language design for beginners.
7. Gosling, J., Joy, B., Steele, G., Bracha, G., & Buckley, A. (2014). *The Java Language Specification*. Oracle America, Inc.
8. Authoritative source for Java features and object-oriented foundationsLutz, M. (2013). *Learning Python* (5th ed.). O'Reilly Media.
9. Detailed explanation of Python syntax and beginner-friendly features.