Review of Research papers

Comparison between java and python

Python outperformed Java in terms of coding efficiency and simplicity for AI applications, even though Java boasted better memory management capabilities. Java's speed and CPU efficiency came as a pleasant surprise, but its reliance on complex class structures stands in stark contrast to Python's streamlined code. The utilisation of an ARM machine yielded positive outcomes for both languages, with Python emerging as the top choice for AI research owing to its user-friendly nature and robust libraries such as Scikit-learn, in contrast to Java's antiquated Weka.[1]

The study delves into the influence of programming languages on the process of software development, with a specific emphasis on the utilisation of multiple inheritance in Java, Python, and C++. To ensure a fair evaluation, the analysis circumvents the diamond problem and utilises CK metrics to assess complexity, as well as combined metrics developed by Goel and Bhatia to gauge reusability. The outcomes divulge that Java exhibits lower reusability in comparison to Python and C++, as indicated by its higher CBO and LCOM scores, which in turn imply reduced reusability. These findings shed light on concerns regarding Java's implementation of multiple inheritance and serve as an illustration of the drawbacks accompanying the use of simulated multiple inheritance within large-scale industrial software systems.[2]

The fundamentals of programming languages are similar across various languages, making it easier to transition between them. It is advisable for beginners to refrain from switching languages prematurely to avoid a loss of confidence. Instead, they should make their language selection based on their goals and preferences. For application development, Java, Swift, or Flutter are recommended options. For game development, JavaScript, Java, C, or C++ are suitable choices. If their focus is on web development or Artificial Intelligence, Python, JavaScript, or Ruby are suggested languages. Beginners should carefully consider their interests and goals before choosing a language that suits their needs.[3]

Java and Python are compared in the article which is about their features, applications, and syntax for beginners. It mentions that Java is a platform independent object oriented programming language used in android development, web service and big data analysis while python being simpler, flexible where it shows up on web development, data analysis as well as game making. The latter part of this report compares things like variables, comments, operators, strings and loops which are needed to help new learners distinguish between these languages. It is therefore an

---

[1] (Acevedo, n.d.)

[2] (Albalooshi & Mahmood, 2017)

[3] (Cutting & Stephen, 2021)

invaluable resource for those who have just started learning computer science and want to choose the most suitable programming language.[4]

An analysis was carried out to compare Java and Python textbooks in terms of their coverage of 100 programming concepts, revealing that Java textbooks exhibit a higher density of these concepts compared to Python. Despite differences, both languages stress core OOP concepts like class, object, and method. Nonetheless, there is a lack of emphasis on specific software engineering concepts such as abstraction in both Java and Python. While practical coding concepts are extensively covered, higher-level thinking and code efficiency receive less attention, indicating a need for additional resources to achieve course objectives regardless of the programming language used in introductory programming courses.[5]

The paper places a strong emphasis on the choice of the first programming language introduced to beginners, as it has a profound influence on their programming style when learning subsequent languages. While acknowledging the importance of effective teaching methods, the text advocates for selecting a language that effectively conveys fundamental concepts. This has resulted in a shift from Java to Python, due to its simplicity and widespread usage, although it is acknowledged that Python may not fully meet all criteria. Additionally, the text suggests exploring alternative languages like Scheme or Haskell to provide a new beginning for all students, particularly those with varying levels of proficiency in Python.[6]

The implementation of Python in Java presents certain disadvantages compared to the existing Python in C system. These drawbacks include the lack of compatibility with C-written Python modules, limited support for the Java platform, and slower performance. However, it is expected that advancements in Java runtime systems will gradually alleviate these issues. Despite the challenges, leveraging Java offers benefits such as portability, access to Java APIs, and technical advantages like true exceptions and garbage collection. Further evaluation is required to determine whether transitioning Python 2.0 to Java is preferable over its current C implementation. Nevertheless, the widespread availability and technical capabilities of Java make it an appealing choice.[7]

The article explores the challenge of deciding between Python and Java as the initial programming language for novices. Although both languages are flexible and robust, Python is suggested as the preferred option due to its simpler learning process and wider range of job opportunities. The article emphasises the significance of becoming proficient in one language before delving into another, with Python being the recommended choice for beginners. It recognizes the value of Java as a secondary language and underscores the transition from outdated languages like Pascal to contemporary ones like Python and Java in educational environments. The article

---

[4] (Gupta, n.d.)
[5](McMaster et al., 2017)
[6] (Pellet et al., n.d.)
[7] (Hugunin, n.d.)

highlights the importance of aligning programming education with real-world application value.