

## **Etude 11: Alphabetically oriented report**

### **Languages used:** Java and Python

With the languages I chose I noticed some major differences between the development of both. I set up a new environment to ensure an apples to apples comparison between java and python.

### **IDES, setup and first-time code:**

Python seemed easier to setup and initially get going in comparison to java. Just run the IDE setup and everything was set up. Java on the other hand required more initial setup with PATH env setting and having both the JRE and JDK installed before the code could be compiled and run.

### **Libraries and community:**

Both languages are fairly old and therefore have an abundance of different libraries and community support. This allows for both languages to be easy for new programmers to pick up. In my opinion, (and very biased) Java seemed easier to me due to having more experiences with it compares to Python due to knowing more libraries. But for a new programmer, going with Python is would be easier as their libraries documentation aren't as daunting as Java and with many resources online.

### **Syntax:**

Python in my opinion has an easier syntax to follow, with not having to worry about type or scope of the variable. But this does come with its downsides, it makes code 'unsafe' because of type null or other type comparisons.

### **Format and readability:**

I found Java to be more readable than that of Python. Python uses a rather 'ugly' code format proving hard to read and debug code. Python's ':' for if statements and loops I found hard to debug when a problem arises. Another point is that Python relies on indentation, things contained in if statement MUST be indented which bothers me as a mainly Java programmer who rather prefer dealing with { and }. Although my IDE gave me a good indication of what to look for.

### **Types:**

Python does not require explicit types java does. Python can use the same var for multiple different types, java can only use one type per variable declaration. I found that this has its benefits for both languages. My experience with python allowed for more concise and readable code but java eased by concerns about type safety and the comparison of the two.

### **Compilation or interpreter:**

I found that python allows for easier development by being an interpreted language, there is no need for compilation of the code. It just sequentially runs what is written. Which I found allowed me to check what my changes were done quicker and more efficiently. Java on the other hand requires compilation and then running of the code, although the difference is minute running two (javac and java) vs one (python <filename>) just makes things simpler

**Classes:**

Java requires a class within a file and main method associated in order to run the file. Python on the other hand does not require any classes, all code can be written in a global scope. This in my opinion seems like a bad thing because of privacy concerns and in large scale projects may cause a problem with data integrity and other things. But for the sake of simplicity it is much less of a hassle to set up a quick demonstration program with Python compares to Java.

**The small things:**

Python allows for multiple values from a callback. In my opinion this allows for easier and more concise code in the form of multi variable per line assignment. This also allows for generation of methods. Java on the other hand does not have this, making me have to wrap it within an array or access global scope variable. This is one of the few things I genuinely likes with Python.