Adrien Protzel

1. Interpreters, compilers, and assembly are various levels of human-readable interfaces for translating words into machine language binary and then executing them, or else called, language systems.
2. The JVM can run multiple languages by converting them into a common format: Java bytecode. Languages like Kotlin, Jython, Scala, and Groovy can be written and then compiled into bytecode, which the JVM then interprets. Essentially, any language that can be translated into bytecode can be executed on the JVM.
3. As we discussed last week, languages evolve and adapt constantly to meet specific needs, whether to run on a particular system or to execute in a certain order. For instance, CPython is written in C and provides extensive libraries and easy translatability for machines while maintaining user-friendly syntax. PyPy, on the other hand, aims to improve Python's performance with its JIT method for compiler speed. IronPython is designed to integrate better with C# and .NET frameworks, making it useful for developers in that ecosystem. All these iterations branch from Python and has it as a base, but they are meant to do wildly different jobs.