**Compilers and interpreters have long been used as computer programs to transform code. But they work in different ways:**

* A compiler translates code written in a high-level programming language into a lower-level language like assembly language, object code andmachine code (binary 1 and 0 bits). It converts the code ahead of time before the program runs.
* An interpreter translates the code line-by-line when the program is running. You’ve likely used interpreters unknowingly at some point in your work career.
* A compiler takes in the entire program and requires a lot of time to analyze the source code. Whereas the interpreter takes a single line of code and very little time to analyze it.
* Compiled code runs faster, while interpreted code runs slower.
* A compiler displays all errors after compilation. If your code has mistakes, it will not compile. But the interpreter displays errors of each line one by one.
* Interpretation does not replace compilation completely.
* Compilers can contain interpreters for optimization reasons like faster performance and smaller memory footprint.