**What’s the Difference?**

**Compiled Languages**

* Your code is converted into a program **before** it runs.
* It goes through a process called **compilation** to become a file the computer can understand quickly.
* If there’s an error, the compiler finds it **before running** the program.

**Examples:** C, C++, Rust **Think of it like:** Making a movie—editing everything first, then showing the final version.

**Interpreted Languages**

* Your code is **read line by line** as it runs.
* Errors only appear when that specific line is executed.
* Slower than compiled code, but easier to test and tweak.

**Examples:** Python, JavaScript **Think of it like:** Live theater—performing as you go, with no full rehearsal.

**What About C#?**

C# does something in between:

1. First, it **compiles** your code into an intermediate version called **IL (Intermediate Language)**.
2. Then, when the app runs, a part of the system called **CLR (Common Language Runtime)** reads and runs that IL code.

So C# is **semi-compiled and semi-interpreted**—you get performance **and** flexibility.