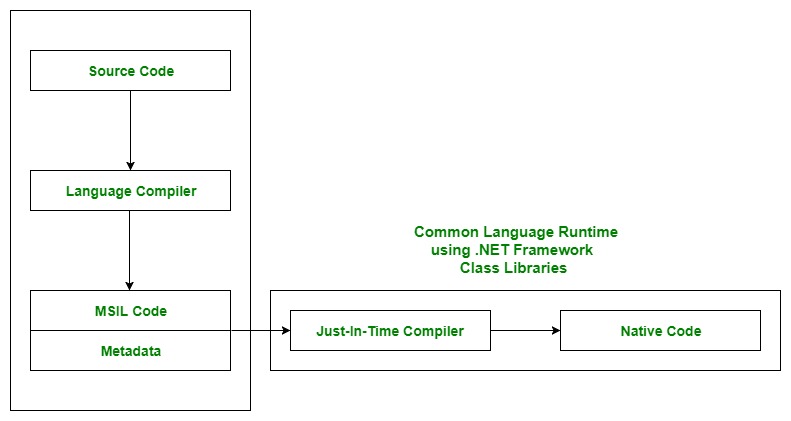
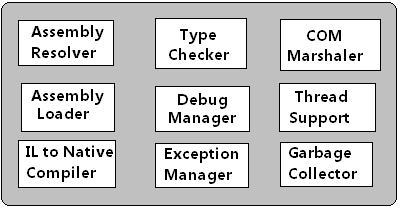
**COMMON LANGUAGE RUNTIME**

* It provides infrastructure for both compilation and execution.
* Works with only managed code (exec. using CLR)
* Source code can be written in any one of the following c#, vb.net, f#.
* All these lang. have their own dedicated compiler which does compilation part and gives MSIL code & METADATA (assembly code)
* MSIL – Microsoft intermediate lang. (intermediate code) and metadata contains all versions, dependencies, Json files, cache…
* Now, using JIT compiler(inside CLR) MSIL code gets converted to Native code
* And that native code is passed to OS for execution.
* In case of unmanaged code (No exec. using CLR), after compilation they are directly converted to native code.





* Main components of CLR: Common Language Specification (CLS), Common Type System (CTS), Garbage Collection (GC), Just In – Time Compiler (JIT)
* CLS- is responsible for converting the different .NET programming language syntactical rules and regulations into CLR understandable format
* CTS- has its own data type system, so CTS is responsible for understanding all the data type systems
* Garbage collector-used to provide the Automatic Memory Management feature
* JIT compiler- converts MSIL code 🡪 machine code