

# what is assembly

---

- In C#, an assembly is a compiled code library used for deployment, versioning, and security. It's the fundamental unit of application development and deployment in .NET. Assemblies are used to build and run applications, and they can be of two types:
  1. **Executable (EXE) Assembly**: This type of assembly is used for applications that have a main entry point, such as console applications or Windows applications.
  2. **Library (DLL) Assembly**: This type of assembly is used for code libraries that do not have a main entry point, such as class libraries.

An assembly consists of one or more files, such as:

- **Manifest**: Contains metadata about the assembly, including version information, culture, and a list of files that make up the assembly.
- **Intermediate Language (IL) Code**: The compiled code that is executed by the .NET runtime.
- **Type Metadata**: Information about the types defined in the assembly, such as classes, interfaces, and methods.
- **Resources**: Other files, such as images, strings, or other non-code data, included in the assembly.

Assemblies provide several key benefits:

- **Encapsulation**: Assemblies can encapsulate a set of functionalities, making it easier to manage and deploy.
- **Versioning**: Assemblies support versioning, allowing different versions of the same assembly to coexist.
- **Security**: Assemblies can include security features, such as code access security, to control what code can do.

Assemblies are created during the build process, where the source code is compiled into Intermediate Language (IL) and packaged into an assembly file. This file can then be distributed and executed on any machine with the appropriate .NET runtime.

---