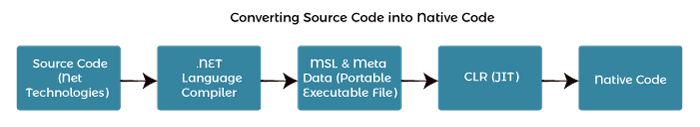
**Common Language Runtime (CLR)**

* .Net provides a runtime called Common Language Runtime Environment which runs the code and makes development easier.
* Compilers and other tools are exposed to CLR and it’s a managed execution environment.
* Different language compilers compile the code and produces MSIL code and metadata which will be executed by CLR using Just in Time (JIT) compiler which produces native code.



* **Functions of CLR:**
  + Convert IL code into Operating System native code.
  + Exception handling
  + Type safety
  + Memory management
  + Security etc.
* **Main Components of CLR:**

**A screenshot of a computer

Description automatically generated**

* + **Base Class Library Support:**
    - Class library that supports classes for .Net application.
    - Provides all types that application needs at runtime.
  + **Thread Support:**
    - It manages the parallel execution of the multi-threaded application.
  + **COM Marshaler:**
    - It provides communication between the COM(Component Object Model). objects and the application.
    - Allows interoperability to our applications.
  + **Type Checker:**
    - Will not allow unsafe casts or uninitialized variables.
    - MSIL code can be verified to guarantee type safety.
    - **Common Type System:**
      * It checks the types used in the application and verifies that they match the standards provided by the CLR.
      * Provides guidelines for declaring, using and managing data types during runtime.
      * Helps in writing language-independent code.
      * 2 categories:
      * Value Type:
        + Memory is allocated in the stack
        + Static memory allocation
        + Holds data directly
        + Primitive: numeric, bool, char etc
        + User defined: Enumeration
      * Reference Type:
        + Memory is allocated in the heap
        + Holds reference to the value of memory address
        + Dynamic memory allocation
        + String, array, class etc.
  + **Exception Manager:**
    - It handles the exception at runtime to avoid application failure.
    - Integrated with MS Windows Structured Exception Handling(SEH)
  + **Security Engine:**
    - It enforces security restrictions.
    - Provides evidence-based security that is based on the origin of the code and the user.
  + **Debug Engine:**
    - It allows you to debug different kinds of applications.
    - Debugging is useful in tracing the execution steps.
  + **MSIL to Native Compilers / Just In Time (JIT) Compiler:**
    - Converts MSIL code to native code.
    - 3 types of JIT:
      * **Pre:** compiles entire MSIL into native code before compilation
      * **Econo:** compiles only those parts of MSIL code required during execution and removes those parts that are not required anymore.
      * **Normal:** compiles only those parts of MSIL code required during execution but places them in cache for future use. It does not require recompilations of already used parts as they have been placed in cache memory.
  + **Code Manager:**
    - It manages code at execution runtime.
  + **Garbage Collector**
    - Automatic memory manager
    - Automatically allocates memory as per the requirement
    - Allocates heap memory to objects (dynamic memory allocation)
    - When objects are not in use, it destroys them and reclaims memory back.
    - Ensures safety by not allowing one object to use content of another.
    - It releases the unused memory and allocates it to a new application.
  + **Class Loader**
    - It is used to load all classes at run time.
    - Manages metadata
* **Benefits of CLR:**
  + Improves performance
  + Enhances portability
  + Increased security as it analyzes the MSIL code is safe or not
  + Automatic memory management
  + Support to components developed in other .net languages
  + Facilitates cross-language integration
  + Allows easy creation of scalable and multithreaded applications
* **References:**
  + [Common Language Runtime (CLR) in C# - GeeksforGeeks](https://www.geeksforgeeks.org/common-language-runtime-clr-in-c-sharp/)
  + [.Net Common Language Runtime - javatpoint](https://www.javatpoint.com/net-common-language-runtime)
  + [what are the components of CLR? (c-sharpcorner.com)](https://www.c-sharpcorner.com/interview-question/what-are-the-components-of-clr)
  + [Common Language Runtime (CLR) - Computer Notes (ecomputernotes.com)](https://ecomputernotes.com/csharp/dotnet/common-language-runtime)