Part02

**2- What's the difference between compiled and interpreted languages and in this way what about Csharp?**

* **Compiled Languages**

Compiled languages (e.g., C, C++) translate source code into machine code using a compiler before execution, offering faster runtime but requiring recompilation for changes.

* **Interpreted Languages**

Interpreted languages (e.g., Python, JavaScript) execute code line-by-line at runtime via an interpreter, allowing flexibility but slower execution.

C# is neither purely compiled nor interpreted.

C# uses a hybrid model. It is first compiled into an **Intermediate Language (IL)** by the C# compiler, then executed by the **Common Language Runtime (CLR)** using Just-In-Time (JIT) compilation. This combines the speed of compiled code with the flexibility of interpreted languages while enabling portability and runtime optimizations.

**3- Compare between implicit, explicit, Convert and parse casting**

**Implicit:** Automatic, no data loss (e.g., int to double).

int x = 10;

float y = x;

**Explicit:** Manual, possible data loss (e.g., double to int).

double x = 10.5;

int y = (int)x;

**Convert:** Uses Convert class, handles nulls safely.

string x = "123";

int y = Convert.ToInt32(x);

**Parse:** Converts strings, throws exception if invalid.

string x = "123";

int y = int.Parse(x);