

# C1 - Introducción a la Programación y al Diseño de Software

---



Microsoft  
.NET

## 2 – Intro. al Desarrollo de Software

### 2.2 - Lenguajes de programación

## 2.2.1 Lenguajes de programación

## Lenguaje de programación

- Conjunto de instrucciones y reglas que se utilizan para crear programas de ordenador.
- Permiten a los programadores comunicarse con los ordenadores, para realizar tareas específicas mediante la escritura de programas.
- Un programa consiste en un código fuente: una serie de instrucciones que describen qué debe hacer el programa.
- Los programas sirven para controlar el hardware y el software del ordenador, automatizar tareas, realizar cálculos complejos o manipular datos.
- Tipos de lenguajes: lenguajes de alto nivel, lenguajes de scripting, lenguajes de bajo nivel, etc. Cada lenguaje de programación tiene su propia sintaxis y conjunto de reglas que los programadores deben seguir para crear programas que funcionen correctamente.

## Lenguaje de alto y bajo nivel

El **lenguaje de alto nivel** es más cercano al lenguaje humano:

- Permite al programador usar conceptos elaborados
- >> Requiere una 'traducción' en código maquina para poder ser ejecuta por el ordenador.
- Se usa para desarrollar aplicaciones de gestión de datos e interacción con el usuario:
- Ejemplo: Java, C#, PHP .

El **lenguaje de bajo nivel** sería el ensamblador:

- Una representación 'legible' del código maquina (un conjunto de valores numéricos que el procesador lee y ejecuta).
- Sólo requiere una traducción muy directa al código maquina.
- Por su eficacia y velocidad, se utiliza para programar sistemas operativos, control de todo tipo de máquinas o aplicaciones que usan sistemas en tiempo real.

## 2.2.2 Lenguaje compilado vs interpretado

## Lenguaje compilado

- Requiere un compilador para convertir el código fuente a código máquina
- El proceso de compilación se realiza una sola vez, generando uno o varios archivos (EXE, DLL..) con el código máquina
- Cualquier modificación en el código nos obliga a compilar de nuevo
- Son más rápidos en su ejecución que los interpretados
- Algunos ejemplos de lenguajes compilados son C#, C++ o Java.

## Lenguaje interpretado

- Al ejecutarlo se lee el código fuente línea por línea.
- Una vez leída una línea, el sistema ejecuta la acción correspondiente.
- El código fuente NO se compila en un archivo ejecutable
- Es interpretado en tiempo real por el interpreter de ese lenguaje.
- Su ejecución es más lenta que los lenguajes compilados
- Ejemplos de lenguajes interpretados: Python, JavaScript, PHP.
- *Nota: El interpreter habitual del lenguaje Javascript es nuestro navegador web*

## Lenguaje mixto

- En realidad, algunos lenguajes usan un modo mixto, como C#, Java
- En ambos casos se compila el código fuente en un código intermedio: CLI para C#, Bytecode para Java
- Dicho código intermedio se ejecuta en la máquina de destino
- En .NET:
  - Se compila a código máquina durante su primera ejecución
  - Las posteriores ejecuciones no requieren compilación adicional
- En Java:
  - La maquina virtual Java (VM) interpreta el bytecode y lo ejecuta



## Tipado estático o dinámico

### En los lenguajes de tipado estático:

- Las variables deben declararse con un tipo de dato específico (string, int...)
- El tipo no puede cambiarse durante la ejecución del programa
- La verificación de los tipos se realiza durante la compilación, no durante la ejecución
- Ofrecen más garantía de que los datos manipulados sean en todo momento los correctos, y por lo tanto, son menos propensos a errores

### En los lenguajes de tipado dinámico:

- El tipo de datos de una variable o expresión se define en tiempo de ejecución
- A menudo resulta incluso opcional definir una variable antes de usarla (JavaScript, PHP)
- Esto exige menos rigor por parte del programador, pero es muchísimo más difícil detectar problemas o fallos

## Lenguajes de programación más populares

### Java:

Compilado, orientado a objetos, se utiliza ampliamente en el desarrollo de aplicaciones empresariales y para la creación de aplicaciones móviles en Android

### Python:

Interpretado, se utiliza en áreas como la inteligencia artificial, big data, la automatización de tareas o el web scrapping

### JavaScript:

Interpretado, utilizado sobre todo en el lado cliente de las aplicaciones web

### C#:

Compilado, orientado a objetos, se utiliza en la creación de aplicaciones de escritorio, web o móviles

### PHP:

Interpretado, utilizado en el lado servidor para el desarrollo de aplicaciones web

### C++:

Compilado, utilizado en el desarrollo de software de sistemas, sistemas operativos, juegos y aplicaciones de alto rendimiento

