



2. Lenguajes de programación

Caso práctico Una de los aspectos del proyecto que más preocupa a Ana es la elección del lenguaje de programación a utilizar. Necesita tener muy claros los requerimientos del cliente para enfocar correctamente la elección, pues según sean éstos unos lenguajes serán más efectivos que otros para el proyecto software.

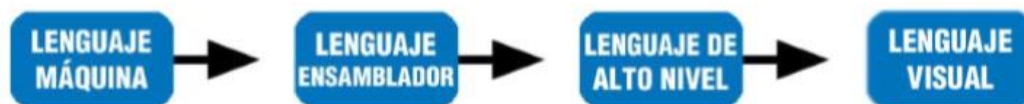
Ya dijimos anteriormente que los programas informáticos están escritos usando algún lenguaje de programación. Por tanto, podemos definir un **Lenguaje de Programación como un idioma creado de forma artificial, formado por un conjunto de símbolos y normas que se aplican sobre un alfabeto para obtener un código, que el hardware de la computadora pueda entender y ejecutar.**

Los lenguajes de programación son los que **nos permiten comunicarnos con el hardware del ordenador.**

En otras palabras, es muy importante tener muy clara la función de los lenguajes de programación. Son los instrumentos que tenemos para que el ordenador realice las tareas que necesitamos.

Hay multitud de lenguajes de programación, cada uno con unos símbolos y unas estructuras diferentes. Además, **cada lenguaje está enfocado a la programación de tareas o áreas determinadas.** Por ello, la elección del lenguaje a utilizar en un proyecto es una cuestión de extrema importancia.

Los lenguajes de programación han sufrido su propia evolución, como se puede apreciar en la figura siguiente:



Características de los Lenguajes de Programación

- ✓ Lenguaje máquina:
 - Sus instrucciones son combinaciones de unos y ceros.
 - Es el único lenguaje que entiende directamente el ordenador. (No necesita traducción).
 - Fue el primer lenguaje utilizado.
 - Es único para cada procesador (no es portable de un equipo a otro).
- ✓ Lenguaje ensamblador:
 - Sustituyó al lenguaje máquina para facilitar la labor de programación.
 - En lugar de unos y ceros se programa usando mnemotécnicos (instrucciones complejas).
 - Necesita traducción al lenguaje máquina para poder ejecutarse.
 - Sus instrucciones son sentencias que hacen referencia a la ubicación física de los archivos en el equipo.
 - Es difícil de utilizar.

- ✓ Lenguaje de alto nivel basados en código:
 - Sustituyeron al lenguaje ensamblador para facilitar más la labor de programación.
 - En lugar de mnemotécnicos, se utilizan sentencias y órdenes derivadas del idioma inglés. (Necesita traducción al lenguaje máquina).
 - Son más cercanos al razonamiento humano.
 - Son utilizados hoy día, aunque la tendencia es que cada vez menos.
- ✓ Lenguajes visuales:
 - Están sustituyendo a los lenguajes de alto nivel basados en código.
 - En lugar de sentencias escritas, se programa gráficamente usando el ratón y diseñando directamente la apariencia del software.
 - Su correspondiente código se genera automáticamente.
 - Necesitan traducción al lenguaje máquina.
 - Son completamente portables de un equipo a otro.

2.1 Concepto y características

Ya sabemos que los lenguajes de programación han evolucionado, y siguen haciéndolo, siempre hacia la mayor usabilidad de los mismos (que el mayor número posible de usuarios lo utilicen y exploten).

La elección del lenguaje de programación para codificar un programa dependerá de las características del problema a resolver.

CONCEPTO

Un lenguaje de programación es el conjunto de:

- **Alfabeto:** conjunto de símbolos permitidos.
- **Sintaxis:** normas de construcción permitidas de los símbolos del lenguaje.
- **Semántica:** significado de las construcciones para hacer acciones válidas.

CARACTERÍSTICAS

Podemos clasificar los distintos tipos de Lenguajes de Programación en base a distintas características:

- ✓ Según lo cerca que esté del lenguaje humano:
 - **Lenguajes de Programación De alto nivel:** por su esencia, están más próximos al razonamiento humano.
 - **Lenguajes de Programación De bajo nivel:** están más próximos al funcionamiento interno de la computadora:
 - Lenguaje Ensamblador.
 - Lenguaje Máquina.

- ✓ Según la técnica de programación utilizada:
 - **Lenguajes de Programación Estructurados:** Usan la técnica de programación estructurada.
 - **Lenguajes de Programación Orientados a Objetos:** Usan la técnica de programación orientada a objetos.
 - **Lenguajes de Programación Visuales:** Basados en las técnicas anteriores, permiten programar gráficamente, siendo el código correspondiente generado de forma automática.

2.2 Lenguajes de programación estructurados

Aunque los requerimientos actuales de software son bastante más complejos de lo que la técnica de programación estructurada es capaz, es necesario por lo menos conocer las bases de los Lenguajes de Programación estructurados, ya que a partir de ellos se evolucionó hasta otros lenguajes y técnicas más completas (orientada a eventos u objetos) que son las que se usan actualmente.

La programación estructurada se define como una técnica para escribir lenguajes de programación que permite sólo el uso de tres tipos de sentencias o estructuras de control:

- ✓ Sentencias secuenciales.
- ✓ Sentencias selectivas (condicionales).
- ✓ Sentencias repetitivas (iteraciones o bucles).

Los lenguajes de programación que se basan en la programación estructurada reciben el nombre **de lenguajes de programación estructurados**.

La programación estructurada fue de gran éxito por su sencillez a la hora de construir y leer programas. **Fue sustituida por la programación modular, que permitía dividir los programas grandes en trozos más pequeños** (siguiendo la conocida técnica "divide y vencerás"). A continuación, triunfaron los lenguajes orientados a objetos y de ahí a la programación visual.

VENTAJAS DE LA PROGRAMACIÓN ESTRUCTURADA

- ✓ Los programas son fáciles de leer, sencillos y rápidos.
- ✓ El mantenimiento de los programas es sencillo.
- ✓ La estructura del programa es sencilla y clara.

INCONVENIENTES

- ✓ Todo el programa se concentra en un **único bloque** (si se hace demasiado grande es difícil manejarlo).
- ✓ **No permite reutilización eficaz de código**, ya que todo va "en uno". Es por esto que a la programación estructurada le sustituyó la programación modular, donde los programas se codifican por módulos y bloques, permitiendo mayor funcionalidad.

2.3 Lenguajes de programación orientados a objetos

Después de comprender que la programación estructurada no es útil cuando los programas se hacen muy largos, es necesaria otra técnica de programación que solucione este inconveniente. Nace así la Programación Orientada a Objetos.

Los lenguajes de programación orientados a objetos tratan a los programas no como un conjunto ordenado de instrucciones (tal como sucedía en la programación estructurada) sino como **un conjunto de objetos que colaboran entre ellos para realizar acciones**.

En la P.O.O. los programas se componen de objetos independientes entre sí que colaboran para realizar acciones.

Los objetos son **reutilizables** para proyectos futuros.

Su primera desventaja es clara: **no es una programación tan intuitiva** como la estructurada.

A pesar de eso, alrededor del 55% del software que producen las empresas se hace usando esta técnica.

Razones:

- ✓ El código es reutilizable.
- ✓ Si hay algún error, es **más fácil de localizar y depurar** en un objeto que en un programa entero.

Características:

- ✓ Los objetos del programa tendrán una serie de atributos que los diferencian unos de otros.
- ✓ Se define clase como una colección de objetos con características similares.
- ✓ Mediante los llamados métodos, los objetos se comunican con otros produciéndose un cambio de estado de los mismos.
- ✓ Los objetos son, pues, como unidades individuales e indivisibles que forman la base de este tipo de programación.