**Parte 1**

**¿Qué es programar? ¿Qué es un programa?**

Instrucciones dictadas al ordenador y que éste realiza al ser convertidas en impulsos eléctricos.

Y los programas tienen como intención ejecutar y automatizar tareas rutinarias.

Programar consiste pues en decirle a una máquina que realice determinada tarea.

**¿Qué es la Ingeniería del software?**

Ciencia que estudie el diseño y creación de productos software para que sean: eficaces, eficientes, extensibles, rentables y mantenibles en el tiempo.

**¿Qué son los lenguajes de Programación?**

Son los que nos ofrecen la posibilidad de controlar una computadora. Con éstos podemos crear secuencias de órdenes y crear soluciones a problemas humanos. Todos los lenguajes contienen en común alfabeto, reglas sintácticas, reglas semánticas, convencionalismos, reglas léxicas y estándares.

**Generaciones históricas de los lenguajes:**

* **1era** generación, las instrucciones accedían directamente al hardware, eran poco inteligibles, se realizaban en código máquina (binario) y se conocen como lenguajes de Bajo Nivel.
* **2da** generación, o programación simbólica, planteamiento de las instrucciones más sencillas y comprensibles para el humano, series equivalentes al lenguaje máquina llamado lenguaje ensamblador.
* **3era** generación, lenguajes de medio y alto nivel, los cuales son más parecidos al humano, pueden ser utilizados para crear programas (no solo manejar el ordenador).
* **4ta** generación, lenguajes de alto nivel, se asemejan aún más al lenguaje humano y son creados para reducir el tiempo de desarrollo y para minimizar el coste y esfuerzo del desarrollo. Estos lenguajes pertenecen a la familia de las bases de datos, análisis y manipulación de datos, generación de informes, desarrollo web y móvil y otros GUI.
* ¿Hay **5ta** generación? No es oficial, pero son los lenguajes relacionados con la IA y ML.

**Todo lenguaje de programación cuneta con una serie de elementos comunes:**

* Sintaxis
* Tipos de datos
* Variables
* Identificadores
* Comentarios
* Funciones
* Directivas
* Condicionales y Bucles
* Palabras Reservadas
* Operadores
* Signos de Puntuación
* Separadore
* Expresiones
* Errores de diversos tipos
* Depuración

La sintaxis en un lenguaje son las reglas que establecen los criterios para construir y secuenciar un lenguaje. Y la semántica establece el significado a cada uno de los elementos del lenguaje.

**Diferencias entre lenguajes alto y bajo nivel:**

**Bajo nivel**: Lenguaje máquina, binario (el 1 pasa la electricidad y el 0 no la pasa) que se ejecuta directamente por la CPU y el más cercano al lenguaje de una máquina. Para entender el binario basta con saber las potencias de 2.



**Medio nivel**: suficiente para que un humano comprenda sin demasiadas complicaciones.

**Alto nivel**: cualquier humano comprender el lenguaje.

**Lenguaje compilado vs interpretado**

El compilado es de alto nivel y necesita de otro programa (compilador) para traducir el código a lenguaje máquina. El lenguaje interpretado no necesita de otro programa, es traducido directamente a lenguaje máquina.

El compilado, requiere la traducción para la ejecución y está por tanto optimizado para ese momento; el interpretado, es convertido a lenguaje máquina mientras se ejecuta y por tanto está optimizado para el desarrollo.

La ventaja del compilado es que es más rápido ya que el ejecutable está en lenguaje máquina y no necesita ser traducido a medida que se va ejecutando. Sin embargo, cada vez que se cambia el código debe realizarse la compilación.

La ventaja del interpretado es que el tiempo entre escritura y probar es menor, por eso es mejor para el desarrollo. Sin embargo, la velocidad de ejecución es menor y se necesita un intérprete instalado.

**Parte 2**

**Tipos de datos:**

* **Cadena de caracteres** (‘string’)
* **Números enteros** (‘int’ o ‘integer’)
* **Números decimales** (‘float’ o ‘double’)
* **Valores Binarios** – Verdadero (1) o Falso (0) (‘True’ y ‘False’)
* **Fechas** (‘date’ y ‘timestamp’)
* **Listas** (‘arrays’ o ‘lists’)
* **Enumerados**
* **Diccionarios** (‘dic’)

De forma particular, **cada lenguaje aporta otros tipos de datos** como los números irreales o las tuplas (conjuntos de datos ordenados del mismo o diferente tipo).

* **Tipado débil o inferido** (lenguaje no exige declara el tipo de un valor o dato)
* **Tipado dinámico** (igual que tipado débil, pero si se declara seria estricto)
* **Tipado fuerte** (exige declara el tipo de un valor o dato)

**Las variables y las constantes:**

Las variables es un dato que almacena un valor, se guardan en el caso de que sea necesario su uso en varias ocasiones para hacer más sencillo y dinámico el código o fragmentos de código. Su valor a lo largo del desarrollo del programa puede cambiar. Para la elección de su nombre hay que tener en cuenta que no pueden tener espacios, no pueden empezar por números, no pueden tener caracteres extraños y se recomienda que comiencen por minúscula.

Las constantes son igual a las variables pero su valor no cambia durante el desarrollo del programa.

**Módulos, paquetes y librerías:**

A medida que el programa crezca y se haga complejo, hará falta de una buena organización y arquitectura para desarrollar un buen código, que este sea mantenible y pueda evolucionar con el paso del tiempo.

Un **módulo** es una porción de programa que tiene a utilizarse frecuentemente, con lo que se decide aislar este fragmento para reutilizarse las veces que sea necesario. Se utilizan para encapsular y modular y también para reutilizar código.

Un **paquete** es un conjunto de módulos almacenados de forma independiente que permite cargar aquellos que necesitemos de forma asilada o en bloque.

Una **librería** son implementaciones ya desarrolladas que ayudan a simplificar tareas complejas. A diferencia de los módulos, nos pueden proporcionar funcionalidades antes inexistentes.