



## Glosario - Introducción a la programación con Python

---

- **Acumulador:** Variable creada usualmente dentro de una iteración, que va acumulando la suma sucesiva de un dato variable.
- **Algoritmo:** Serie finita de pasos para resolver un problema.
- **Back-end:** "Lo de lado del servidor". Tipo de código o de desarrollador cuya función es programar la lógica detrás de una aplicación.
- **Buenas prácticas de programación:** Consiste en todo aquello que tenga relación con optimizar nuestro código, desde su comprensión al leerlo, hasta el mejoramiento de su funcionamiento. Existen prácticas de programación generales, así como algunas más específicas dependiendo de cada lenguaje. Se considera parte de las buenas prácticas seguir las convenciones establecidas para cada lenguaje.
- **Bug:** En programación, corresponde a un error o falla en nuestro programa que genera un resultado o comportamiento inesperado.
- **Clase:** Modelo que define las propiedades y métodos que tendrá un determinado objeto.
- **Contador:** Variable que, dentro de una iteración, va aumentando su valor de 1 en 1.
- **Debug:** Corresponde al proceso de buscar y resolver conflictos o 'bugs' en nuestros programas. Para ello existen herramientas especializadas que conoceremos más adelante.
- **De bajo nivel:** Programa cercano a la máquina, más próximo a la arquitectura del hardware.
- **De alto nivel:** Programa más cercano a los programadores y usuarios.

- **De código abierto:** Término con el que se conoce al software distribuido y desarrollado libremente con licencias que permiten su implementación, modificación y distribución.
- **Documentación/Documentar el código:** La documentación corresponde a líneas dentro del código que no son ejecutadas, y que consisten en explicar qué está haciendo el código escrito en esa sección, notas relevantes, ejemplos, etc. Documentar el código es un ejemplo de una buena práctica de programación. Además de escribir la documentación en el código mismo del programa, usualmente es posible también acceder a ella a través de páginas web, para el caso de un lenguaje completo, o de una librería específica.
- **Editor de texto:** Software que permite crear y modificar archivos compuestos únicamente por textos sin formato, conocidos comúnmente como archivos de texto ó "texto plano". En este curso utilizaremos este tipo de software para escribir código.
- **Escapar caracteres:** Como en un lenguaje de programación existen muchos caracteres especiales (reservados para alguna función específica), a veces se hace necesario "escaparlos", o "tomar el valor literal". Para que python interprete el caracter como su valor literal, en lugar de interpretarlo como una instrucción de código, debemos escapar el caracter anteponiendo ante él un backslash `\`.
- **Framework:** Conjunto de herramientas para un propósito. Estas herramientas permiten estandarizar la forma en que se construye el programa.
- **Función:** En programación, "función" y "método" son sinónimos.
- **IDE (Integrated Desktop Environment):** No es un término que utilizaremos con frecuencia en el curso, sin embargo, es ampliamente utilizado en la jerga de programación. Corresponde a un software que contiene un set de herramientas para desarrollo de software. Un IDE normalmente contiene un editor de texto, compilador o intérprete y herramientas de debug, entre otras funcionalidades.
- **Indentación:** También conocido como sangrado. Consiste en espacios vacíos que se dan antes de una instrucción para especificar que está dentro de un contexto.
- **Imprimir en pantalla:** Definir en nuestro código, de manera explícita, que un dato debe ser mostrado en la pantalla a través de instrucciones del tipo `print`.
- **Importar:** Se refiere a traer código externo al código que se está escribiendo. Esto se logra mediante la instrucción `import`. Se pueden importar librerías, o archivos propios de extensión `.py`.
- **Interpolación:** Permite insertar el valor de una variable dentro de una cadena de texto.

- **Kernel de Python:** Corresponde al servidor levantado en el computador que permitirá ejecutar Python. Se ejecuta al escribir la instrucción `python` desde la terminal. Otro kernel muy utilizado es *iPython*, y se ejecuta desde la terminal con la instrucción `ipython`. El kernel es también el que hace posible hacer bloques de código en Jupyter Notebook.
- **Lenguaje de programación:** Un lenguaje de programación es un lenguaje diseñado para describir el conjunto de acciones que un computador debe ejecutar. Por lo tanto, un lenguaje de programación es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo.
- **Librería:** En estricto rigor, debiesen llamarse "bibliotecas", pero por el inglés de la palabra "library", es común utilizar la palabra "librería". Corresponde a un conjunto de funciones orientadas a un objetivo específico (por ejemplo, generar gráficos), estructuradas en uno o más archivos externos al programa principal que se está escribiendo. Para utilizar una librería, ésta se debe importar al código. Normalmente, son de código abierto, y se puede acceder a ellas, o descargarlas, desde un gestor de paquetes, el cual varía para cada lenguaje. Python posee muchas librerías, de las cuales varias ya quedan instaladas al instalar Anaconda. El gestor de librerías de Python es `pip`, y el de Anaconda es `conda`.
- **Llamar:** Se refiere a escribir el nombre de una variable, para obtener su valor, o el de una función, para utilizarla. En el caso de llamar a una función, se debe agregar paréntesis `()` al final del nombre, y agregar parámetros de ser necesario.
- **Método:** También se les llama `función`. Son bloques de código definidos que permiten realizar una operación específica. Para utilizarse deben ser "llamados".
- **Métodos nativos/Built-in methods:** Corresponden métodos integrados en el lenguaje mismo. Para utilizarlos, no es necesario crear un objeto; pueden ser llamados directamente.
- **Objeto:** Unidad o elemento que corresponde a la instancia de una clase. El objeto tiene propiedades y métodos asociados, y según ellos, se puede realizar distintas operaciones sobre el objeto.
- **Paradigma de programación:** Representa un enfoque particular para diseñar soluciones utilizando un lenguaje de programación. En el caso de Python, se utiliza el **Paradigma Orientado a Objetos**.
- **Programa:** Un programa informático o programa de computadora es una secuencia de instrucciones, escritas para realizar una tarea específica en una computadora.
- **Prompt:** Se llama prompt al carácter o conjunto de caracteres que se muestran en un terminal para indicar que está a la espera de órdenes.

- **Refactorización (refactoring):** Corresponde al proceso de reestructurar un código fuente, alterando su estructura interna sin modificar su comportamiento. Normalmente se aplica para optimizar el funcionamiento del código y/o facilitar su lectura por parte del programador.
- **Script:** Es un programa usualmente simple que se puede ejecutar desde la terminal.
- **Terminal:** También se le llama "Consola". Se refiere a la "Línea de comandos" de nuestro sistema operativo, por medio de la cual se pueden ejecutar distintas instrucciones, o comandos, para distintos fines. También se le conoce como "CLI", por sus siglas en inglés ("Command Line Interface").
- **Tipo de dato:** O tipo de objeto. Corresponde a la naturaleza o propiedad de un objeto. Los tipos de datos más utilizados son integer, float, string y bool.
- **Variable:** Contenedor de un valor o del resultado de una expresión. Su valor puede cambiar a lo largo de código.