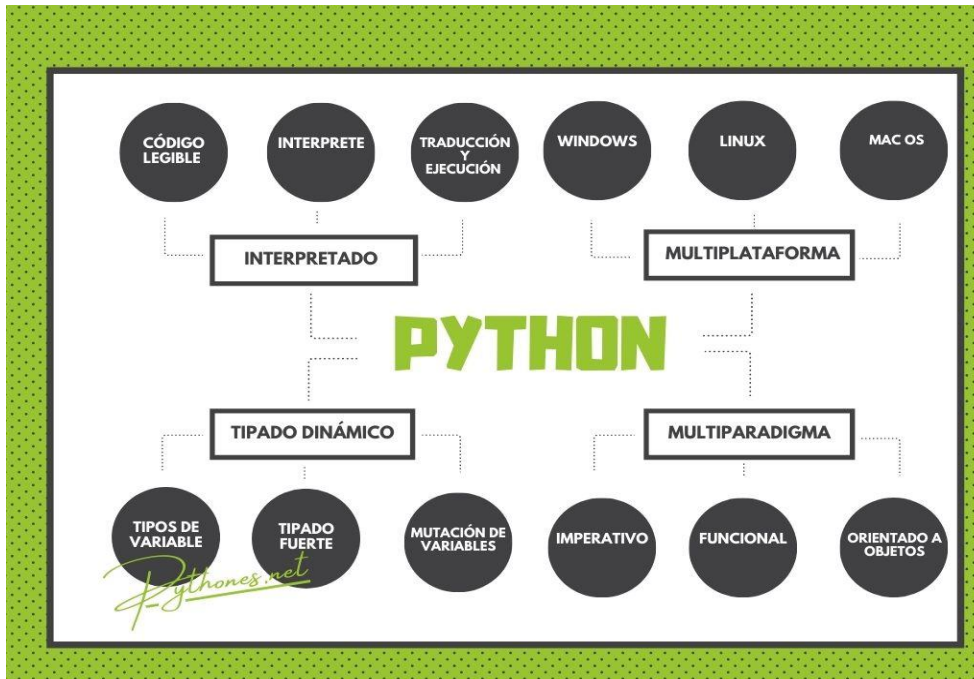


Definición de Python

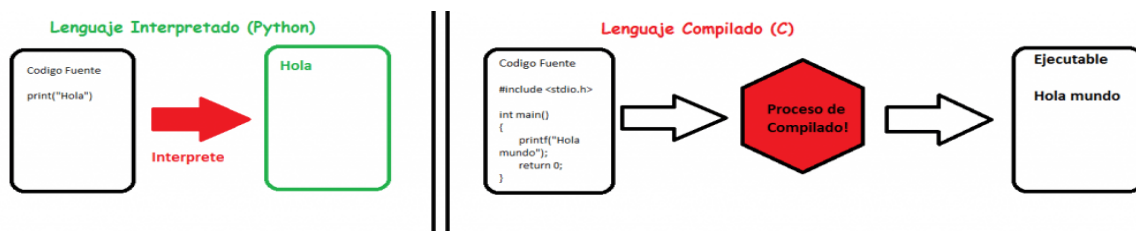
Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Y define este como un lenguaje multiparadigma, debido a que soporta orientación a objetos, programación imperativa y en menor medida programación funcional. Es interpretado de tipado dinámico y multiplataforma.



Características de Python

Es Interpretado

Normalmente los **lenguajes de programación** se pueden agrupar en **Interpretados y Compilados** según la forma en la que son traducidos. Los lenguajes se inventaron para facilitar al programador el desarrollo aplicaciones, imagina que nuestro computador trabaja en código binario y sería muy difícil programar todo en unos y ceros, además de que nos tardaríamos siglos y nadie quiere eso!!.



Los lenguajes Compilados son aquellos en los que el código del programador es traducido por completo de una sola vez mediante un proceso llamado **“Compilado”** para ser ejecutado por un sistema predeterminado. Entre los más comunes encontramos “C”, “C++”, Java, etc.

Y los lenguajes **Interpretados** son aquellos en los que el código del programador es traducido mediante un **intérprete** a medida que es necesario. Entre los más comunes encontramos **“Python”**, “Ruby”, “Javascript”, etc.

El que Python sea **interpretado** nos presenta ventajas:

- Al ser interpretado no necesitamos compilar ahorrándonos mucho tiempo en el desarrollo y prueba de una aplicación.
- Nuestro código fuente puede ser ejecutado en cualquier software siempre y cuando este disponga del intérprete (Windows, Linux, Mac, Android, Web).

Multiparadigma :

Esto nos dice que Python es un lenguaje que soporta más de un paradigma, suponiendo paradigma como modelo de desarrollo (y cada lenguaje tiene el suyo).

Los paradigmas de la programación que admite Python son:

- Imperativo
- Funcional
- Orientado a objetos

Imperativo:

Los lenguajes de programación también se pueden agrupar en imperativos y declarativos, los del primer grupo a los que pertenece Python son aquellos que describen el estado del programa y permiten su modificación mediante condiciones o instrucciones de código que le indican al computador cómo realizar una tarea.

Los declarativos son aquellos que solo “declaran” condiciones, ecuaciones, etc que describen un problema y detallan su solución.

La principal diferencia es que:

En la programación IMPERATIVA se describe paso a paso un conjunto de instrucciones que deben ejecutarse para variar el estado del programa y solucionar el problema. Mientras que en la declarativa sólo se procede a describir el problema que se quiere solucionar..

Funcional:

La programación funcional es un paradigma de la programación declarativa basada en el uso de funciones matemáticas que permite la **variación del programa** mediante la **mutación de variables**. Esto nos va a permitir operar con datos de entrada y salida. Brindándole así la posibilidad al usuario de ingresar datos que serán procesados para darnos otros datos de salida.

Orientado a Objetos (POO):

La programación orientada a objetos ofrece la particularidad en la forma de obtener los resultados. **Los objetos manipulan los objetos de entrada para la obtención de resultados (salida) específicos** donde cada objeto nos ofrece una función específica y también nos permite la agrupación de **bibliotecas o librerías**. Así como también podremos crear las nuestras propias!!

Los objetos son entidades que tienen un determinado estado, las entidades son propiedades que los diferencian.

Un ejemplo clásico en programación podría ser:

El Objeto Coche cuyas propiedades serán por ejemplo, el color, el modelo. Este objeto contiene toda la información que nos permitirá diferenciarlo de otros Objetos (otros coches).

De Tipado Dinámico:

Es de tipado dinámico cuando una variable puede tomar **diferentes valores de distintos tipos en diferentes momentos**. En python las variables son declaradas por su contenido y no por su contenedor, lo que nos va a permitir cambiar el valor y tipo de una variable durante la ejecución sin necesidad de volver a declarar..

Por ejemplo supongamos que X es una variable, y en este caso $X = 1$.

Si X está almacenando un número, pero podemos almacenar una letra durante la ejecución del programa en la misma variable X mediante una instrucción, por ejemplo $X = \text{"a"}$.

Veamos una variable como un bloque de memoria donde podemos “almacenar” un dato. O como una cajita donde guardar cosas..

En python:

- Si dentro de la variable colocamos números, la variable será de tipo numérica.
- Ahora, si colocamos letras, la variable será de tipo texto.
- Y si en ella colocamos un booleano (Verdadero o Falso), será de tipo booleano.

Es decir, vamos a guardar en ella lo que queremos y python automáticamente detectara su tipo, no hace falta que se lo indiquemos.

Y esto ha sido todo, si algún concepto no te quedo claro puedes comentar!. Yo trataré de explicarlo lo mas sencillamente posible, estos conceptos los vamos a profundizar a medida que avancemos.

Esta primera lección solo es una introducción a las **características del lenguaje** que iremos resaltando más adelante. ¿¿Has visto cuántas son!??