PYTHON

Fundamentos de python

mayerlis cardenas quintero

2022

ACTIVIDAD 1: PYTHON

Fundamentos de Python

Python:

En términos técnicos, Python es un **lenguaje de programación de alto nivel**, orientado a objetos, con una semántica dinámica integrada, principalmente para el **desarrollo web** y de **aplicaciones informáticas**.

Es muy atractivo en el campo del **Desarrollo Rápido de Aplicaciones (RAD)** porque ofrece tipificación dinámica y opciones de encuadernación dinámicas.

Python es relativamente simple, por lo que es fácil de aprender, ya que requiere una sintaxis única que se centra en la legibilidad. Los desarrolladores pueden leer y traducir el **código Python** mucho más fácilmente que otros lenguajes.

Historia:

La historia de Python como lenguaje de programación inicia a finales de los 80s y principios de los 90s con Guido Van Rossum, una historia de 29 años de desarrollo.

## **Cronología de Python[**

Fechas de publicación:[13](https://es.wikipedia.org/wiki/Historia_de_Python#cite_note-13)​[14](https://es.wikipedia.org/wiki/Historia_de_Python#cite_note-14)​

* Comienzo de la implementación - December, 1989
* Publicación interna en CWI - 1990
* Python 0.9.0 - 20 de febrero de 1991 (publicado en alt.sources)[15](https://es.wikipedia.org/wiki/Historia_de_Python#cite_note-history-15)​
  + Python 0.9.1 - febrero de 1991
  + Python 0.9.2 - otoño de 1991
  + Python 0.9.4 - 24 de diciembre de 1991
  + Python 0.9.5 - 2 de enero de 1992 (solo para Macintosh)
  + Python 0.9.6 - 6 de abril de 1992
  + Python 0.9.7 beta - 1992
  + Python 0.9.8 - 9 de enero de 1993
  + Python 0.9.9 - 29 de julio de 1993
* Python 1.0 - enero de 1994
  + Python 1.5 - 31 de diciembre de 1997
  + Python 1.6 - 5 de septiembre de 2000
* Python 2.0 - 16 de octubre de 2000
  + Python 2.1 - 17 de abril de 2001
  + Python 2.2 - 21 de diciembre de 2001
  + Python 2.3 - 29 de julio de 2003
  + Python 2.4 - 30 de noviembre de 2004
  + Python 2.5 - 19 de septiembre de 2006
  + Python 2.6 - 1 de octubre de 2008
  + Python 2.7 - 3 de julio de 2010
* Python 3.0 - 3 de diciembre de 2008
  + Python 3.1 - 27 de junio de 2009
  + Python 3.2 - 20 de febrero de 2011
  + Python 3.3 - 29 de septiembre de 2012
  + Python 3.4 - 16 de marzo de 2014
  + Python 3.5 - 8 de febrero de 2015
  + Python 3.6 - 23 de diciembre de 2016
  + Python 3.7 - 12 de junio de 2018
  + Python 3.8 - 14 de octubre de 2019
  + Python 3.9 - 5 de octubre de 2020

La última **versión de Python** 3.10 disponible actualmente (septiembre de 2022) es **Python** 3.10.7

Características:

**Programación Orientada a objetos (POO):**

Al igual que otros lenguajes populares como Java, C++ o [Javascript](https://www.miteris.com/blog/que-es-javascript-caracteristicas-librerias/), Python es un lenguaje orientado a objetos. Un lenguaje orientado a objetos es aquel en el que el código se organiza en unidades denominadas clases y objetos. Esto permite representar conceptos cotidianos en un programa. En definitiva, a través de Python podemos expresarnos de forma similar a como lo haríamos en la vida real.

**Lenguaje interpretado:**

Existen dos tipos de lenguajes: compilados o interpretados como en el caso de Python. No es necesario compilar (transformar el lenguaje de los programas informáticos a uno equivalente) cuando se trabaja con Python, ya que los intérpretes que se utilizan con este lenguaje se encargan de ejecutar dichos programas a través de scripts propios.

**Multiplataforma:**

Python está disponible en los principales sistemas operativos como Linux, Windows, UNIX, Mac OS, etc. Es decir, este lenguaje puede ser ejecutado en casi cualquier sistema operativo siempre que se cuente con un intérprete adecuado para ello.

**Tipado dinámico:**

El tipado dinámico de Python permite que las variables no tengan que especificar su tipo. Estas adoptan un tipo automáticamente en función del valor que se les asigne mientras el lenguaje esté funcionando.

**Lenguaje open source:**

Python es un lenguaje de código abierto, por lo que no se requieren licencias de pago para comenzar a trabajar con él.

**Ampliamente respaldado:**

Sus características y funcionalidades hacen que este lenguaje sea muy interesante. Por ello, Python ha generado una comunidad de usuarios muy grande a su alrededor que puede ser de utilidad cuando queremos encontrar información o pedir ayuda para desarrollar cualquier tipo de programa o algoritmo.

**Es polivalente:**

Como ya hemos comentado, Python se utiliza para infinidad de proyectos y aplicaciones diferentes. Actualmente destacan los siguientes:

* Aprendizaje automático (Machine Learning)
* Inteligencia Artificial (IA)
* Big data y Análisis de datos
* Operaciones matemáticas
* Visualización de datos
* Programación de apps
* Desarrollo web
* Desarrollo de videojuegos
* Gestión financiera

Ventajas:

#### Una gran comunidad para ayudarte:

Python está sustentado en una gran comunidad de usuarios que explora sus posibilidades frecuentemente. Esta comunidad tan activa permite que usuarios de todos los niveles encuentren los mejores tutoriales, consejos y claves para empezar a utilizarlo.

Es decir, hablamos de un lenguaje accesible para personas con cualquier nivel de conocimientos. Además es un software gratuito y de código abierto.

#### Sintaxis sencilla:

El lenguaje de codificación de Python tiene una sintaxis muy sencilla y utiliza palabras en inglés.

#### Legibilidad:

Al ser un lenguaje de alto nivel, las líneas de código escritas en Python también son fáciles de leer e interpretar para rápidamente entender el objeto del fragmento de código analizado.

#### Escalabilidad:

Puedes empezar un programa en Python sin tener que preocuparte por la difícil tarea de reescribir o adaptar el código a otras plataformas a medida que vas creciendo. Es decir, este lenguaje te permitirá aumentar la complejidad de tus programas a lo largo del tiempo.

#### Aplicaciones en Big Data:

El último punto fuerte que destacamos de Python es la gran utilidad para el mundo del Big Data, del Data Science y de la Inteligencia Artificial. Además, este lenguaje posee librerías muy potentes de cálculo matricial y analítica de datos como las mencionadas anteriormente: Pandas, Numpy, Scikit Learn o Keras entre otras.

Desventajas:

#### Lentitud

La lentitud de Python se debe principalmente a su naturaleza dinámica y versatilidad. No obstante, hay formas de optimizar las aplicaciones de Python aprovechando la sincronización, entendiendo las herramientas de creación de perfiles y considerando el uso de múltiples intérpretes.

#### Consumo de memoria

Otra de las desventajas de Python es que, en el caso de que una tarea requiera mucha memoria, Python no es la mejor opción.**El consumo de memoria de Python es muy alto, y esto se debe a la flexibilidad de los tipos de datos.**

#### Desarrollo móvil

Python es ideal para plataformas de escritorio y servidor, pero**para el desarrollo móvil no es un lenguaje muy adecuado.** Por este motivo, apenas vemos aplicaciones móviles desarrolladas con Python.

Variables:

Las variables en Python **se crean cuando se definen por primera vez, es decir, cuando se les asigna un valor por primera vez**. Para asignar un valor a una variable se utiliza el operador de igualdad (=).

Tipos de Variable:

Python tiene **cuatro tipos primitivos:** enteros, flotantes, booleanos y cadenas o Strings.

* Los tipos de variables enteros en Python (int) se utilizan para representar datos numéricos, específicamente números enteros. Estos pueden ser tanto positivos como negativos.
* Los tipos de datos flotantes en Python (float) se utilizan para representar números de coma flotante o con decimales.
* Los tipos primitivos booleanos en python (bool) son tipos de datos binarios, es decir que pueden tomar los valores: *Verdadero* (**True**) y *Falso* (**False**). Son útiles para expresiones con condicionales y de comparaciones.
* Los tipos de datos String en Python son un arreglo de caracteres que forman cadenas para formar un mensaje o oración generalmente. Se pueden crear usando comillas simples, dobles o triples.