



Elementos básicos de la programación en Python

PYTHON



Elementos básicos de la programación en Python

Python es un lenguaje de programación multiparadigma, es decir, cuenta con soporte para la programación orientada a objetos, la programación imperativa y la programación funcional. Python desde el 2017 está dentro de los cinco principales lenguajes de programación más populares de acuerdo con HackerRank (2019).



Una de las principales características de Python es, según Python (2022), su estructura clara y sumamente fácil de leer, esto se lo debemos a que es un lenguaje de muy alto nivel, utilizando, por ejemplo, tabuladores para observar qué elementos pertenecen a funciones, ciclos y/o condicionales. Además, cada variable se considera como un objeto.

Estructura y elementos del lenguaje

Python, al igual que otros lenguajes de programación, está formado por una serie de elementos básicos (Python, 2023). Estos son:

Ciclos

Sentencia IF

Ayuda a ejecutar un bloque de código solo si se cumple una condición determinada.

```
▶ x = 1
  if x >= 5:
    print('x es mayor o igual a 5')
  else:
    print('no se cumplio la condicion')
```



Sentencias ELIF y ELSE

Son maneras de agregar más condiciones dentro del *if*. Esto en caso de que tengamos otras opciones que considerar (no solamente “sí” o “no”).

```
▶ x = 6
  if x == 1:
    print('x es igual a uno')
  elif x == 2:
    print('x es igual a dos')
  elif x == 3:
    print('x es igual a tres')
  else:
    print('x es igual a: x')
```

Sentencia WHILE

Se utiliza para ejecutar repetidamente un bloque de código, mientras una condición dada se cumpla, es decir, resulte verdadera. Es de suma importancia cuidar que la condición que estamos evaluando en algún momento se convierta en falsa, de lo contrario tendremos un ciclo infinito, lo cual ocasionará varios conflictos en el rendimiento de tu sistema.

```
▶ [] i = 0
  while i**2 < 10:
    print(i, 'al cuadrado es menor que 10')
```

```
i = i + 1 # Esto evita que entremos en un ciclo infinito
```

```
0 al cuadrado es menor que 10  
1 al cuadrado es menor que 10  
2 al cuadrado es menor que 10  
3 al cuadrado es menor que 10
```

Sentencia FOR

El ciclo *for* se utiliza para ejecutar cierto código un número de veces, a partir de cierto elemento; puede ser utilizado de manera semejante al *while*, además puede trabajar con iterables o listas, asignando "alias" para hacer mención del elemento en turno.

```
[ ] for nombre in ['Hugo', 'Paco', 'Luis']:  
    print('Hola ' + nombre)
```

```
Hola Hugo  
Hola Paco  
Hola Luis
```

Módulos y namespaces

Un módulo es un archivo (que utiliza el sufijo `.py`), que contiene declaraciones y funciones de Python. Cada módulo incluye su propio namespace, el cual es utilizado como un espacio de nombres global por cada una de las funciones determinadas en el módulo.

Para saber sobre la **librería Math en Phyton**, te recomendamos ver el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de Diciembre). *Tema 5. Librería Math* [Archivo de video]. Recuperado de <https://bit.ly/37Q3D2G>

Paquetes

Los paquetes son una forma de organizar los espacios de nombres en Python, para ello, se utilizan el nombre de los módulos con puntos. Por ejemplo, el nombre del módulo `C.D` denomina un submódulo llamado `C` en paquete de nombre `D`.

Funciones definidas por el usuario

Para saber sobre **las funciones en Python**, te recomendamos ver los siguientes videos:

Dirección de Producción de Contenidos. (2019, 4 de Diciembre). *Tema 5. Funciones en Python* [Archivo de video]. Recuperado de <https://bit.ly/2N2UmMY>

Dirección de Producción de Contenidos. (2019, 4 de Diciembre). *Tema 5. Función range en Python* [Archivo de video]. Recuperado de <https://bit.ly/39Jh2eS>

Argumentos arbitrarios (*args)

En ocasiones, una función requiere recibir una función indefinida de argumentos. Para esos casos, Python incluye la opción de definir un argumento con el formato `*args`. Este nos permite entregarle distintos valores separados por comas.

Hoy en día, Python se ubica como uno de los lenguajes de programación favoritos por los desarrolladores gracias a su código limpio y legible, su licencia abierta, y por ser utilizado en el desarrollo de grandes aplicaciones a nivel mundial, haciéndolo ideal para las personas que desean ingresar a esta industria.



Referencias bibliográficas

- HackerRank. (2019). *HackerRank Developer Skills Report*. Recuperado de <https://www.hackerrank.com/research/developer-skills/2019>
- Python. (2022). *Beginners Guide*. Recuperado de <https://wiki.python.org/moin/BeginnersGuide/Overview>
- Python. (2023). *El tutorial de Python*. Recuperado de <https://docs.python.org/es/3/tutorial/>