

Python

para no programadores

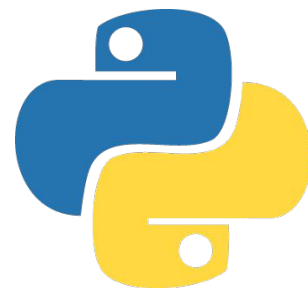
Módulo 3

¿Para qué sirven las funciones?

¿Para qué sirven las funciones?

Las *funciones* nos permiten agrupar una o más líneas de código bajo un mismo nombre. Su objetivo principal es el de evitar la repetición de código, haciendo de un archivo de código de fuente más claro, legible y fácil de mantener. Por ejemplo, consideremos el siguiente programa:

```
print("¡Hola mundo!")  
print("Desde Python.")  
  
a = 5  
b = 7  
c = a * b  
print(c)  
  
print("¡Hola mundo!")  
print("Desde Python.")
```



¿Para qué sirven las funciones?

Es algo muy común que algunas porciones de código se repitan en distintas partes de un programa, como es el caso aquí de las primeras y últimas dos líneas. Para evitar esto, lo que se hace es darle un nombre a las líneas repetidas creando así una función. La sintaxis, siguiendo el ejemplo anterior, es la siguiente:

```
def mi_funcion():  
    print("¡Hola mundo!")  
    print("Desde Python.")
```

La creación de una función requiere de la palabra reservada `def`, seguida de un nombre a elección, un par de paréntesis (que más adelante veremos para qué sirven) y dos puntos. El bloque de código dentro de la función debe tener sangría, tal como ocurría con los bucles y los condicionales.

¿Para qué sirven las funciones?

Ahora que tenemos esta función, en los lugares donde hacíamos uso de esas dos líneas simplemente pondremos `mi_funcion()`.

```
def mi_funcion():  
    print("¡Hola mundo!")  
    print("Desde Python.")
```

```
mi_funcion()
```

```
a = 5  
b = 7  
c = a * b  
print(c)
```

```
mi_funcion()
```



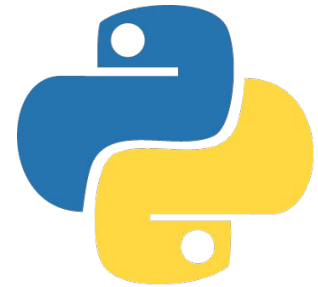
¿Para qué sirven las funciones?

En todos los lugares donde aparezca el nombre de una función junto a un par de paréntesis, Python ejecutará las líneas de código que se encuentran dentro de ella. A esto lo denominamos llamar (o invocar) a una función.

Si modificamos el código de nuestra función de esta manera:

```
def mi_funcion():  
    print("¡Hola mundo!")  
    print("Desde Python.")  
    print("Y Geany.")
```

Veremos que en ambas llamadas a `mi_funcion()`, ahora se ejecutan tres líneas en lugar de dos.



¡Muchas gracias!

¡Sigamos trabajando!