

Jorge Calvo

# Funciones en Python

Sígueme en - [LinkedIn](#) [Twitter](#) [Blog](#)



Este obra está bajo una [licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional](#).

## ▼ Funciones en Python

Las funciones en Python se puede entender como ese bloque de código, independiente del resto de mi programa, que cuando yo la llamo me devuelve un resultado, que puede ser de cualquier tipo (numérico, texto, array, ...)

### Como se programan

- Todas las funciones empiezan por la palabra reservada **def** y después el nombre que yo le quiera poner (no usar espacios, ni números ni signos), más dos paréntesis
- Posteriormente se colocan dos puntos para indicar el inicio del código que irá en su interior
- El código de la función irá **indentado** dentro de la función (como en los bucles y los condicionales)
- Finalmente toda función acaba con la palabra reservada **return** que es el valor que queremos devolver

Ejemplo de función que devuelve los datos de calcular una hipotenusa

```
def hipotenusa():  
    a=2  
    b=6  
    print("Lo valores de los catetos son 2 y 6")  
    h=((2**2)+(6**2))**0.5  
    print("El valor de la hipotenusa es " + str(h))  
    return
```

### Como se llama a una función

- Poniendo el nombre la función y sus dos paréntesis

**IMPORTANTE** Una variable que está creada en el interior de la función no afecta fuera de ella. Podemos tener variables con el mismo nombre dentro y fuera de una función y cada una podrá tener valores diferentes.

Ejemplo de llamar a una función

```
hipotenusa()  
  
Lo valores de los catetos son 2 y 6  
El valor de la hipotenusa es 6,32
```

Si yo quisiera guardar el valor devuelto por una función en una variable para usarlo posteriormente, ese valor debe estar puesto en **return**

```
def hipotenusa():  
    a=2  
    b=6  
    print("Lo valores de los catetos son 2 y 6")  
    h=((2**2)+(6**2))**0.5  
    print("El valor de la hipotenusa es " + str(h))  
    return h
```

```
valor=hipotenusa()  
print(valor)  
6,32
```

```
print(valor*2)  
12,64
```

Hay dos tipos de funciones:

- **Estáticas:** No admiten ningún valor solo ejecutan lo que hay en su interior (sin variables externas). Igual que los ejemplos anteriores de la hipotenusa
- **Dinámicas:** Admiten variables del exterior que modifican el resultado de su retorno

¿Qué pasa si yo quiero llamar a la función hipotenusa pero me gustaría cambiar los valores de sus catetos y obtener el resultado?

Se crean variables conocidas como **parametros** dentro de la función. Ahora se usarán los paréntesis que antes siempre estaban vacíos.

Función hipotenusa con parametros

```
def hipotenusa(cat1,cat2):  
    print("Lo valores de los catetos son " + str(cat1)+ " y " + str(cat2))  
    h=((cat1**2)+(cat2**2))**0.5  
    print("El valor de la hipotenusa es " + str(h))  
    return h
```

La llamo pasándole dos parametros

```
hipotenusa(3,7)  
Lo valores de los catetos son 3 y 7  
El valor de la hipotenusa es 7,61
```

## Funciones anónimas (Lambda)

Las funciones anónimas es otro tipo de función usado en Python, muy parecido y con el mismo objetivo que las estudiadas en el punto anterior, la única diferencia que este tipo de funciones se crean sin un nombre específico y usando el comando lambda, por el cual también estas funciones son conocidas como **funciones lambda**.

Se realiza un ejemplo sencillo de la suma de dos números usando una función lambda.

La función se instancia en una variable, en este caso llamada sumar y posteriormente se escribe:

**lambda parametro1, parametro2, .... : operación a realizar**

Observar el ejemplo en el terminal.

```
sumar= lambda x,y : x+y  
print(sumar(2,5))
```

```
7
```