

## 4. Funciones

---

### Índice

- [1. Definición y justificación](#)
- [2. Parámetros y retorno](#)
- [3. Regla de ámbito](#)

### 1. Definición y justificación

Una función permite encapsular un fragmento de código de cara a su reutilización posterior.

Una función se compone principalmente de 4 partes como máximo:

- Nombre que la identifica
- Parámetros que recibe (opcional)
- Proceso que realiza
- Valor que devuelve (opcional)

En Python las funciones se definen de esta forma:

```
def learn():  
    print("¡Estoy aprendiendo Python!")
```

Y se la llama así:

```
learn()
```

### 2. Parámetros y retorno

Las funciones pueden recibir parámetros (también llamados argumentos) con distintos valores de entrada en cada llamada:

```
def learnSomething(something):  
    print(f"Estoy aprendiendo {something}")  
  
learnSomething("Python a saco")
```

Una función debe recibir al ser llamada el mismo número de argumentos que los parámetros con los que se la declaró.

Si fuera necesario, pueden declararse parámetros opcionales (también llamados por defecto), con un valor predefinido si no se indica otro al llamarla:

```
def learnSomethingDefault(something = "algo"):
    print(f"Estoy aprendiendo {something}")

learnSomethingDefault()
learnSomethingDefault("Python a cascoporro")
```

Las funciones también pueden devolver valores resultantes del proceso que realicen:

```
def learnSomethingAndLevelUp(something, level):
    print(f"Estoy aprendiendo {something}")

    return level + 1

level = 0
level = learnSomethingAndLevelUp("Python like a pro", level)
```

Por último, Python permite crear funciones anidadas (dentro de otras funciones) y también pasar funciones como argumentos.

### 3. Regla de ámbito

Una variable declarada fuera de una función puede ser leída dentro de la misma:

```
myText = "genial"

def beFunny():
    print(f"Una ranita iba caminando, Python es {myText}")

beFunny()
```

Una variable declarada dentro de una función no podrá ser leída fuera de la misma, y solo será accesible dentro del ámbito de la función:

```
myText = "genial"

def beMoreFunny():
    myText = "la leche,"
    myOtherText = "ni lo dudes"

    print(f"Aquella ranita seguía caminando y Python es {myText}
    {myOtherText}")

beMoreFunny()
```

```
print(myText)
print(myOtherText)
```

Para poder cambiar el valor de una variable declarada fuera de una función, debe referenciarse con la palabra reservada **global**:

```
myText = "genial"

def beSuperFunny():
    global myText
    myText = "lo mejor que me ha pasado"

    print(f"Otra ranita voló y Python es {myText}")

beSuperFunny()

print(f"Repito, ranas aparte: Python es {myText}")
```

## Referencias

[Funciones](#)

[Ámbito](#)