

Parámetros arbitrarios en Python

Cuando en una función definida en Python vemos un argumento precedido del carácter “*” quiere decir que la función no sabe exactamente el número de argumentos que se le van a pasar. Los parámetros fijos(“los normales”) siempre serán esperados, sin embargo con los arbitrarios esto no es así.

Se declaran al final de la función porque tomarán el resto de argumentos no tomados ya por argumentos fijos.

```
def func(argumento_fijo, *argumentos_arbitrarios)
```

La función anterior puede ser llamada por ejemplo:

```
func(ar1,ar2,ar3)
```

```
func(ar1,ar2)
```

Como vemos “ar1” siempre dará valor a argumento fijo, sin embargo “ar2” y “ar3” formarán la tupla de “argumentos_arbitrarios”

Ejemplo

Un ejemplo de uso podría ser concatenar varias cadenas a una cadena base dada, pero no sabemos el número exacto de cadenas a priori, de forma que creamos una función flexible para este caso.

(Existen otras formas de realizar esta tarea evidentemente)

```
1#!/usr/bin/python
2def concatena(cad_base, *otras):
3    for o in otras:
4        cad_base += " " + o
5    print cad_base
6
7concatena("Base","hola","adios")
```

El ejemplo anterior da como salida “Base hola adios”