

# Argumentos indeterminados

Quizá en alguna ocasión no sabemos de antemano cuantos elementos vamos a enviar a una función. En estos casos podemos utilizar los parámetros indeterminados por posición y por nombre.

## Por posición

Para recibir un número indeterminado de parámetros por posición, debemos crear una lista dinámica de argumentos (una tupla en realidad) definiendo el parámetro con un asterisco:

```
def indeterminados_posicion(*args):  
    for arg in args:  
        print(arg)  
  
indeterminados_posicion(5, "Hola", [1, 2, 3, 4, 5])
```

```
5  
Hola  
[1, 2, 3, 4, 5]
```

## Por nombre

Para recibir un número indeterminado de parámetros por nombre (clave-valor o en inglés *keyword args*), debemos crear un diccionario dinámico de argumentos definiendo el parámetro con dos asteriscos:

```
def indeterminados_nombre(**kwargs):  
    print(kwargs)  
  
indeterminados_nombre(n=5, c="Hola", l=[1, 2, 3, 4, 5])
```

```
{'n': 5, 'c': 'Hola', 'l': [1, 2, 3, 4, 5]}
```

Al recibirse como un diccionario, podemos iterarlo y mostrar la clave y valor de cada argumento:

```
def indeterminados_nombre(**kwargs):  
    for kwarg in kwargs:  
        print(kwarg, "=>", kwargs[kwarg])  
  
indeterminados_nombre(n=5, c="Hola", l=[1, 2, 3, 4, 5])
```

```
n => 5  
c => Hola  
l => [1, 2, 3, 4, 5]
```

## Por posición y nombre

Si queremos aceptar ambos tipos de parámetros simultáneamente, entonces debemos crear ambas colecciones dinámicas. Primero los argumentos indeterminados por valor y luego los que son por clave y valor:

```
def super_funcion(*args,**kwargs):
    total = 0
    for arg in args:
        total += arg
    print("sumatorio => ", total)
    for kwarg in kwargs:
        print(kwarg, "=>", kwargs[kwarg])

super_funcion(10, 50, -1, 1.56, 10, 20, 300, nombre="Hector", edad=27)
```

```
sumatorio => 390.56
nombre => Hector
edad => 27
```

Los nombres **args** y **kwargs** no son obligatorios, pero se suelen utilizar por convención.

Muchos frameworks y librerías los utilizan por lo que es una buena practica llamarlos así.