

# Programando en Python III



# 1. Funciones

- 1.1 Llamadas
- 1.2 Parámetros
- 1.3 Variables
- 1.4 None



Programando en Python

# Funciones



# Funciones

## ¿Qué son?

**Son fragmentos de código definidos de manera que se pueden reutilizar y realizan un determinado algoritmo.**

**Reducen el número total de líneas de tu proyecto y lo hacen más legible!**

**Las funciones pueden recibir uno o más argumentos y a su vez devolver algún resultado o nada.**

# Funciones

## Llamar a una función

Acción de usar una función.

```
def sumar():  
    print 5 + 10  
  
sumar()
```

# Funciones

## Predefinidas en librerías

```
>>> abs(-3)↵
3
>>> abs(round(2.45, 1))↵
2.5
>>> from math import sin↵
>>> sin(1)↵
0.8414709848078965
```

# Funciones

## Definir una función

```
1 def cuadrado(x):  
2     return x ** 2
```

# Funciones

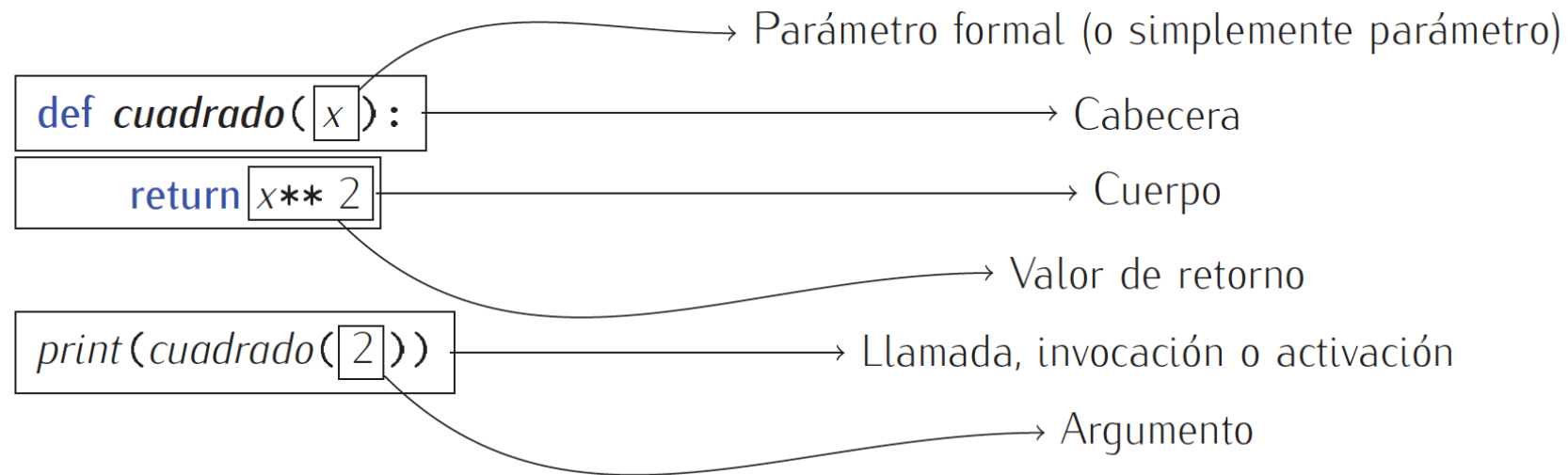
## Usar una función

```
1 def cuadrado(x):  
2     return x ** 2  
3  
4 print(cuadrado(2))  
5 a = 1 + cuadrado(3)  
6 print(cuadrado(a * 3))
```



# Funciones

## Definiciones



# Funciones

## Nombres

**Las reglas para dar nombre a las funciones y a sus parámetros son las mismas que las de las variables. Pero cuidado, no se puede usar el mismo nombre para una variable y una función al mismo tiempo.**

# Funciones

## Funciones que utilizan funciones

No hay problema si necesitas utilizar otra función dentro de una función.

```
1 from math import sin
2
3 def xsin(x):
4     return x * sin(x)
```

# Funciones

No importa el nombre del identificador

```
1 def cubo(z):  
2     return z ** 3
```

```
1 def cubo(x):  
2     return x ** 3
```

```
1 y = 1  
2 print(cubo(y))
```

# Funciones

También puedes utilizar sentencias de control

```
1 def es_mayor_de_edad(edad):  
2     if edad < 18:  
3         resultado = False  
4     else:  
5         resultado = True  
6     return resultado
```

```
1 def es_mayor_de_edad(edad):  
2     if edad < 18:  
3         return False  
4     else:  
5         return True
```

```
1 def es_mayor_de_edad(edad):  
2     if edad < 18:  
3         return False  
4     return True
```

# Funciones

## Más de un parámetro

```
1 def área_rectángulo(altura, anchura):  
2     return altura * anchura  
3  
4 print(área_rectángulo(3, 4))
```

# Funciones

## Sin parámetros

```
1 def lee_entero():  
2     return int(input())  
3  
4 a = lee_entero()
```

# Funciones

## Cómo ordenar un programa para que sea legible

```
1 from math import sqrt
2
3 def cuadrado(x):
4     return x**2
5
6 def suma_cuadrados(vector):
7     suma = 0
8     for elemento in vector:
9         suma += cuadrado(elemento)
10    return suma
11
12 # Programa principal
13 mivector = []
14 for i in range(3):
15     mivector.append(float(input('Dame un número: ')))
16 s = suma_cuadrados(mivector)
17 print('Distancia al origen:', sqrt(s))
```



# Funciones

## Variables locales y globales

```
1 from math import sqrt
2
3 def área_triángulo(a, b, c):
4     s = (a + b + c) / 2
5     return sqrt(s * (s-a) * (s-b) * (s-c))
6
7 print(área_triángulo(1, 3, 2.5))
8 print(s)
```

# None

## Tipo de datos especial

**Significa ausencia de valor.**

# Ejercicios



## Notebook 3



## Contacto

Correo: [a.cobo.aguilera@gmail.com](mailto:a.cobo.aguilera@gmail.com)

LinkedIn: [Aurora Cobo Aguilera](#)

GitHub: [AuroraCoboAguilera](#)

Google Scholar: [Aurora Cobo Aguilera](#)





red.es

Centro de  
Referencia Nacional  
en Comercio Electrónico  
y Marketing

CRN  
Digital



UNIÓN EUROPEA

*"El FSE invierte en tu futuro"*

**Fondo Social Europeo**

