

# Funciones y métodos

Daniel González Duque

# ¿Qué son y para qué sirven?

- Secciones organizadas de código.
- Se usan para llevar a cabo acciones sencillas.
- Se usan para ahorrar espacio.
- Introducen el concepto de **Modularidad** y de **Encapsulamiento**.

# Sintaxis en Python

```
def function(<parámetros>):  
    """Descripción de la función"""  
    <Operaciones de la función>  
    return <Expresión>
```

# Ejemplo

```
def printme(string):  
    " Esta función imprime una cadena de caracteres"  
    print(string)  
    return
```

# Se debe tener en cuenta

- Las funciones deben cumplir una función específica, se recomienda no poner más de una acción en una función.
- **Cualquier variable declarada dentro de una función no aparecerá en el terminal cuando se compile el código**, a menos que se dé como resultado después del `return`.

# Keyword Arguments – Ejemplo 1

```
def printme(string):  
    " Esta función imprime una cadena de caracteres"  
    print(string)  
    return  
  
printme('Hola')  
printme(string='Hola')
```



# Keyword Arguments – Ejemplo 2

```
def printinfo(nombre,edad):  
    """ Esta función imprime el nombre y la edad de la  
    persona """  
    print('Nombre: '+nombre)  
    print('Edad: '+str(edad))  
    return  
  
printinfo('Daniel',26) # Se puede llamar por el orden  
# Se puede llamar por los parámetros así sea en desorden  
printinfo(nombre='Daniel',edad=26)  
printinfo(edad=26,nombre='Daniel')
```

# Default Arguments – Ejemplo 1

```
def printinfo(nombre,edad=26):  
    """ Esta función imprime el nombre y la edad de la  
    persona"""  
    print('Nombre: '+nombre)  
    print('Edad: '+str(edad))  
    return  
  
# Si no se incluye la edad por defecto la toma como 26  
printinfo('Daniel')
```