

# Unidad III

## Introducción a Python

13 de abril del 2020

# When you switch from C++ to Python



# Agenda

Funciones

Parámetros

Parámetros por omisión

Keywords como parámetros

Retorno de valores

Ejercicios

# Agenda

Funciones

Parámetros

Parámetros por omisión

Keywords como parámetros

Retorno de valores

Ejercicios

# Funciones

- ▶ Vamos a necesitar muchas veces realizar actividades repetitivamente.
- ▶ Para ello, tenemos dos opciones, repetir código u organizarlo para que esa “repetición” sea “elegante” y más sencilla de leer.
- ▶ Una función, es la forma de agrupar expresiones y sentencias (algoritmos) que realicen determinadas acciones, pero que estas se ejecutan solo cuando son llamadas.
- ▶ “Envolvemos” una serie de pasos y le ponemos un nombre. Para que esa “serie de pasos” se ejecuten, la tenemos que llamar por su nombre.

# Funciones

- ▶ En Python se definen las funciones usando la palabra *def* + el nombre que le vamos a poner a la función + paréntesis (dónde van los parámetros) + dos puntos (:)

```
def hola_mundo():  
    print("Hola Mundo")
```

```
hola_mundo()
```

## Funciones - Ejemplo

```
# Funcion que imprime los numeros impares  
# desde el 1 al 1000  
def hola_get_odd_numbers():  
    for i in range(1, 1000, 2):  
        print(i, "es impar")
```

Para ver el funcionamiento debo hacer:

```
hola_get_odd_numbers()
```

# Agenda

Funciones

Parámetros

Parámetros por omisión

Keywords como parámetros

Retorno de valores

Ejercicios

# Parámetros

- ▶ Un parámetro es un valor que recibe esa función, y que puede usarla libremente, dentro de la función.
- ▶ Ese valor solo queda dentro de la función y nadie más puede verlo.
- ▶ los parámetros se los define dentro de los paréntesis y se utilizan iguales que las variables.

```
# Funcion que imprime Hola y el nombre y apellido  
def hola_mundo(nombre, apellido):  
    print("Hola", nombre, apellido)
```

Para ver el funcionamiento debo hacer:

```
hola_mundo("Emmanuel", "Arias")  
hola_mundo("Guido", "van Rossum")
```

## Parámetros - Ejemplo

```
# Funcion que imprime los numeros impares  
# Tener Cuidado de empezar siempre con un numero  
def hola_get_odd_numbers(desde, hasta):  
    for i in range(desde, hasta, 2):  
        print(i, "es impar")
```

Para ver el funcionamiento debo hacer:

```
hola_get_odd_numbers(11, 1000)  
hola_get_odd_numbers(1, 100)  
hola_get_odd_numbers(25, 1000)
```

## Parámetros - Precaución

- ▶ Cuando llamamos la función y le pasamos los valores para que funcionen, deben ser **pasados en el mismo orden en el que fue definido**
- ▶ Esto se facilita con el uso de keywords como parámetros (lo vemos después)

# Agenda

Funciones

Parámetros

Parámetros por omisión

Keywords como parámetros

Retorno de valores

Ejercicios

## Parámetros por omisión

- ▶ En python es posible asignar valores por defecto a los parámetros de las funciones.
- ▶ Esto nos da la posibilidad de utilizar los valores por “default” de las funciones y escribir menos parámetros (ver el caso del print)

```
# Funcion que imprime los numeros impares  
# Tener Cuidado de empezar siempre con un numero  
def hola_get_odd_numbers(desde, hasta=1000):  
    for i in range(desde, hasta, 2):  
        print(i, "es impar")
```

Para ver el funcionamiento debo hacer:

```
hola_get_odd_numbers(11)  
hola_get_odd_numbers(1)  
# Puedo cambiar el ''hasta'' por default.  
hola_get_odd_numbers(11, 200)  
hola_get_odd_numbers(1, 10)
```

# Agenda

Funciones

Parámetros

Parámetros por omisión

Keywords como parámetros

Retorno de valores

Ejercicios

## Keywords como parámetros

- ▶ Es posible llamar a una función, pasándole los argumentos como “pares” clave=valor

```
hola_get_odd_numbers(desde=11, hasta=1000)
hola_get_odd_numbers(hasta=1000, desde=11)
hola_mundo(nombre="Emmanuel", apellido="Arias")
hola_mundo(apellido="Arias", nombre="Emmanuel")
```

# Agenda

Funciones

Parámetros

Parámetros por omisión

Keywords como parámetros

Retorno de valores

Ejercicios

# Retornar valores

- ▶ Es posible retornar valores de las funciones
- ▶ Para ello se utiliza la palabra reservada *return*

```
# Funcion que devuelve el promedio de  
# tres numeros:  
def get_promedio(numero1, numero2, numero3):  
    resultado = (numero1 + numero2 + numero3)/3  
    return resultado
```

# Agenda

Funciones

Parámetros

Parámetros por omisión

Keywords como parámetros

Retorno de valores

Ejercicios

# Ejercicios

- ▶ Convertir a función los 6 ejercicios vistos en la tarea 2
- ▶ Crear una función que me devuelva el precio con algún descuento realizado. Es decir la función debe recibir, el precio de un producto y el descuento, en porcentaje. (Regla de 3 simple a full.)
- ▶ Crear una función que reciba un numero entero, y que devuelva el mes correspondiente a ese número. Escribir un Error en caso de que el número del mes salga del rango entre 1 y 12.