

¿Qué es un condicional?

Como su mismo nombre lo indica una condicional, es una condición que permite tomar una opción u otra según se necesite, por ejemplo si tengo dos números y quiero sumar esos dos números si son mayores que 5 o restarlos si son menores que 5, el uso de condicionales me permite estructurar mi código de manera que pueda cumplir con estas condiciones teniendo en cuenta todas las opciones que se puedan presentar. De la siguiente forma.

```
def conditional(num_one, num_two):  
    if num_one > 5 and num_two > 5:  
        print(num_one + num_two)  
    else:  
        print(num_one - num_two)  
conditional(4,10)
```

¿Cuáles son los diferentes tipos de bucles en Python? ¿Por qué son útiles?

Los diferentes tipos de bucles en Python son for y while, los dos sirven para realizar repeticiones de una parte de nuestro código, con la diferencia que while ejecuta la repetición una y otra vez mientras la condición sea verdadera es decir que se cumpla la condición, o por ejemplo:

```
a = 1  
while a < 10:  
    print("Hola Mundo")
```

De este modo se va a imprimir infinitamente porque siempre se va a cumplir la condición.

```
a = 1  
while a < 10:  
    print("Hola Mundo")  
    a += 1
```

De este modo se va a imprimir solo 9 veces porque cada que hace una repetición o iteración va sumando uno a la variable a y cuando esta llega a 10 para el bucle porque ya no se cumple la condición de a < 10

Son útiles porque en programación muchas veces necesitamos repetir parte de nuestro código ya sea para recorrer colecciones(listas, tuplas, diccionarios) o para cumplir condiciones y cambiar comportamientos cada vez que se repite.

¿Qué es una lista por comprensión en Python?

Son una serie de bucles y condicionales que se pueden colocar todos en una sola línea de código, sirve para generar de manera dinámica una lista. el primer elemento es la acción que queremos hacer, el segundo es el bucle con la variable de iteración por ejemplo así:

```
num_list = range(1, 11)
cubed_nums = [ ]
cubed_nums = [num * 3 for num in num_list] => lista de comprension
print(cubed_nums)
```

En conclusión una lista de comprensión es una forma de crear de forma corta, clara y concisa una lista por medio de un bucle.

¿Qué es un argumento en Python?

R/ Se llama argumentos a los valores que se pasan a una función para que realice una tarea así:

```
def argumentos(num_one, num_two):
    return(num_one - num_two)

argumentos(4,10) => en esta función el 4 y el 10 serian los argumentos
```

Existen distintos tipos de argumentos

- Los argumentos posicionales: son argumentos que se pueden llamar por su posición en la definición de la función, como en el ejemplo anterior se tomaría num_one = 4 y num_two = 10.
- Los argumentos de palabra clave son aquellos que se llaman por su nombre es decir en el ejemplo anterior se llamaría de la siguiente forma: argumentos(num_one = 4, num_two=10)
- Los argumentos obligatorios: son argumentos que se deben pasar a la función, como en el ejemplo anterior si solo llamara la función así: argumentos() saltaría error.
- Los argumentos opcionales: son argumentos que no es necesario especificar. En Python, los argumentos opcionales son argumentos que tienen un valor predeterminado, por ejemplo

```
def greeting(name = 'Guest'):
    print(f'Hi {name}!')

greeting()
greeting('Aroa')
```

¿Qué es una función Lambda en Python?

Nos permite empaquetar funciones normalmente pequeñas y luego usarlas en otras funciones, lambda permite guardar la función en una variable y empaquetar el funcionamiento por ejemplo:

```
full_name = lambda first, last: f'{first} {last}' print(full_name('Pedro', 'Perez'))
```

¿Qué es un paquete pip?

Pip es una herramienta de Python que permite instalar paquetes (módulos y librerías de python) desde línea de comando del ordenador, facilitando la instalación, actualización, desinstalación de paquetes entre otros.

-Para instalar un paquete debes abrir el cmd del ordenador, ubicarte en el directorio que deseas instalar los paquetes y dar la instrucción:

- Para instalar: pip install nombre_del_paquete
- Para desinstalar: pip uninstall nombre_del_paquete
- Para actualizar: pip install --upgrade nombre_del_paquete
- Para obtener una lista de los paquetes instalados: pip list

Pip es una herramienta muy potente e imprescindible para los desarrolladores en python.