

Checkpoint 5

• ¿Qué es un condicional?

Un condicional es un sistema en el que un determinado código se ejecutará dependiendo de si una condición se cumple o no. Para eso, se usa la palabra “if” con la condición que se quiera comprobar; con la palabra “elif” se pueden añadir otros casos y “else” marcará el código que se ejecutará en caso de que ninguna de las condiciones anteriores se cumpla.

Por ejemplo, en este caso, las condiciones son las que están después de las palabras “if” y “elif”. Primero, se comprobará si la primera condición (el del bloque “if”) es cierta; si lo es, se ejecutará el código que esta indentado en la siguiente línea, si no, se comprobará la condición del siguiente bloque. En caso de que ninguna de las condiciones se cumpla/sea verdadera, se ejecutará el código del bloque “else”:

```
if x > 5:
    x += 1
elif x > 3:
    x += 2
else:
    x += 3
```

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

Hay dos tipos de bucles en python: los bucles “for” y los bucles “while”. Los primeros permiten pasar por todos los elementos de un conjunto (una lista, una tupla, un rango,...), mientras que los segundos se seguirán ejecutando mientras que una condición determinada sea cierta. Por esta razón, los bucles “while” tienen el peligro de crear bucles infinitos en el caso de que la condición no deje nunca de ser cierta.

La utilidad de cada uno depende del tipo de estructura a la que se le quiera aplicar el bucle. Los bucles “for” son útiles para trabajar con elementos como las listas, las tuplas o los rangos y los bucles “while”, en cambio, para elementos que no sean colecciones.

• ¿Qué es una comprensión de listas en python?

La comprensión de listas es una sintaxis más corta que permite crear una nueva lista a partir de una ya existente escribiendo menos código. La estructura básica es la siguiente (usar el condicional es opcional):

```
nueva_lista = [acción_a_ejecutar for item in lista_base if condición]
```

Para ver su utilidad lo podemos comparar con un bucle “for”:

```
lista_base = [1, 2, 3, 4]
nueva_lista = []

for item in lista_base:
    if item % 2 == 0:
        nueva_lista.append(item + 1)
```

Con la comprensión de listas, esto mismo se podría conseguir con una sola línea de código:

```
lista_base = [1, 2, 3, 4]
nueva_lista = [item + 1 for item in lista_base if item % 2 == 0]
```

• ¿Qué es un argumento en Python?

Un argumento es un elemento que se puede añadir a una función para poder aplicarle el código que se quiera, haciendo que sea dinámico y dependa de los argumentos añadidos cada vez que se use esa función.

Una función puede tener todos los argumentos que se quiera (o no tener ninguno), poniéndolos entre los paréntesis de la función y separados por comas. Por ejemplo, esta función tiene dos argumentos y los sumará cada vez que se use:

```
def mi_funcion(argumento1, argumento2):
    return argumento 1 + argumento 2

mi_funcion(2, 5) (=> devolverá "7")
mi_funcion(1,9) (=> devolverá "10")
```

• ¿Qué es una función de Python Lambda?

Es un tipo de función que permite envolver la acción de una función y poder ser usada como argumento en otra función. Por ejemplo, podríamos usar una función lambda con el ejemplo de la pregunta anterior y aplicárselo a otra:

```
mi_funcion = lambda argumento1, argumento2 : argumento1 + argumento2

def segunda_funcion(argumento3):
    return f"el resultado es {argumento3}"

segunda_funcion(mi_funcion(2, 5)) (=> devolverá "el resultado es 7")
segunda_funcion(mi_funcion(1, 9)) (=> devolverá "el resultado es 10")
```

• ¿Qué es un paquete pip?

Permite instalar y usar paquetes creados por otros usuarios de la misma forma que se usan los nativos de python. Hay una gran variedad de paquetes creados por la comunidad de programadores que permiten usar diferentes funciones en vez de tener que crearlas uno mismo.

Una vez instalados estos paquetes, se pueden aplicar en el código de la misma manera que se usa, por ejemplo, "decimal" (cuando se quiere trabajar con decimales se escribe "from decimal import Decimal", es decir se importa ese paquete que python ya tiene instalado de serie).