

Condicionales en Python:

Una condicional es una estructura de control que determina qué acción debe ejecutar el código dependiendo de si las condiciones establecidas son verdaderas (True) o falsas (False).

La sentencia condicional en Python (al igual que en muchos otros lenguajes de programación) es *if*. La sintaxis *if* seguida de la condición seguida de dos puntos:

Ejemplo 1

```
edad = 13 # variable que queremos comprobar

if edad < 16: # condición
    print("Eres menor y no puedes entrar") # acción si la condición es cierta
```

En el ejemplo anterior, si la condición se cumple (*edad < 16* es *True*) se mostrará el texto “Eres menor y no puedes trabajar” pero si la condición no se cumple (*False*) el código no indica que se realice ninguna acción. Si queremos añadir una acción para cualquier otro caso que no se corresponda con la condición debemos añadir *else*:

Ejemplo 2

```
edad = 13 # variable que queremos comprobar

if edad < 16: # condición
    print("Eres menor y no puedes entrar") # acción si la condición es cierta
else:
    print("¡Bienvenido!") # acción para el resto de los casos
```

Es posible que también se quiera añadir diferentes comprobaciones entre la primera condicional (*if*) y la última (*else*). En este caso se añade la condición (o condiciones) precedida de la palabra clave *elif* (combinación de *else if*).

Ejemplo 3

```
edad = 13 # variable que queremos comprobar

if edad < 18: # condición
    print("Eres menor y no puedes entrar.") # acción si la condición es cierta
elif edad >= 18 and edad <= 30: # segunda condición
    print("Si eres estudiante puedes beneficiarte del descuento.") # acción
elif edad >= 65: # tercera condición
    print("Si estás jubilado puedes beneficiarte del descuento.") # acción
else:
    print("¡Bienvenido!") # acción para el resto de los casos
```

El código anterior se ejecutará de manera ordenada, es decir, primero se comprueba la condición que comienza con *if*. Si es cierta (*True*) se ejecutará la acción correspondiente. Si, por el contrario, no es cierta (*False*) se comprobará la condición que sigue al primer *elif*. Si esta es cierta, se ejecutará la acción asociada. Si no lo es se pasará a comprobar la siguiente condición, y así hasta llegar al final del código.

Operadores de comparación

Cuando escribimos condiciones debemos incluir alguna expresión de comparación usando los siguientes operadores:

Operador	Símbolo	Uso con
Igualdad	==	Cadenas (<i>string</i>) y números
Desigualdad	!=	Cadenas (<i>string</i>) y números
Menor que	<	Números
Menor o igual que	<=	Números
Mayor que	>	Números
Mayor o igual que	>=	Números

Condicionales anidados

Python permite establecer una condicional en el interior de otra. A eso se le llama condiciones anidados, Se pueden anidar cuantos condicionales se necesita, aunque se recomienda no incluir más de dos o tres.

El ejemplo anterior comprueba varias condiciones por lo que también podría haber sido expresada como una condición anidada:

Ejemplo 4

```
edad = 13

if edad < 18:
    print("Eres menor y no puedes entrar.")
else:
    if edad >= 18 and edad <=30:
        print("Si eres estudiante puedes beneficiarte del descuento.")
    else:
        if edad >= 65: # cuarta condición
            print("Si estás jubilado puedes beneficiarte del descuento.")
        else:
            print("¡Bienvenido!") # acción para el resto de los casos
```

Los condicionales anidados nos permiten reaccionar a cada caso, pues mostramos un mensaje acorde a cada condición que se cumple. Sin embargo, como podemos observar, el código en este ejemplo es menos intuitivo. El uso de *elif* ayuda a simplificar la sintaxis y la compresión del código.

Condicionales compuestos

Los condicionales compuestos consisten en unir condiciones dentro de una clave *if* mediante las palabras clave *and* (y) u *or* (o). En el primer caso todas las condiciones deben ser cierta (*True*). En el segundo sólo una de las condiciones debe cumplirse.

En el ejemplo anterior, `elif edad >= 18 and edad <=30:` es un caso de condicional compuesto.

Asignación condicional de una única línea (operador ternario)

El operador ternario es una declaración *if-else* que se define en una sola línea. Una condicional simple en la que sólo se establece una condición y las acciones correspondientes al resultado de *True* o *False* pueden adoptar la sintaxis de una única línea. Por ejemplo, tomando como ejemplo un caso anterior:

```
edad = 13

if edad < 16:
    mensaje = "Eres menor y no puedes entrar"
else:
    mensaje = "¡Bienvenido!"

print(mensaje)
```

Puede adoptar la siguiente sintaxis: `[acción si True] if [condición] else [acción si False]`

Ejemplo 5

```
edad = 13

mensaje = "Eres menor y no puedes entrar" if edad < 16 else "¡Bienvenido!"

print(mensaje)
```

El resultado es un código más conciso sin reducir la comprensión.