**IS, ELIF, ELSE**: dependiendo del resultado de una evalución lógica ejecuta un bloque de código o no.

**Circuito lógico**: método para evitar evaluar todos los términos de una expresión lógica.

Ejemplo: if False and edad < 99 and len(nombre) > 50

**IF ternario**: forma simplificada de la sentencia if, solo cuando se desea usar una sola sentencia, no un bloque de código.

**Variable contador**: estructura de incrementar o decrementar el contenido de una variable de forma constante:

cont = cont + 1

cont += 1

x = x – 3

x -= 3

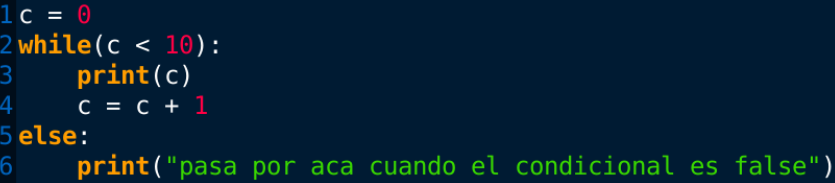
**Variable acumulador**: estructura de incrementar o decrementar el contenido de una variable de forma variable:

acu = acu + x #acu += x

acu = acu – x #acu -= x

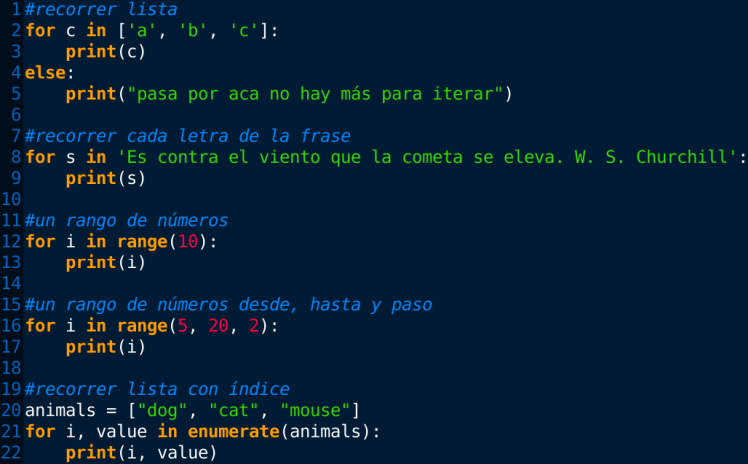
**WHILE, ELSE:**

* Mecanismo para generar una iteración o un bucle. Repite un bloque de código mientras el condicional sea verdadero.
* Tener cuidado de formar un bucle infinito



**FOR, ELSE:**

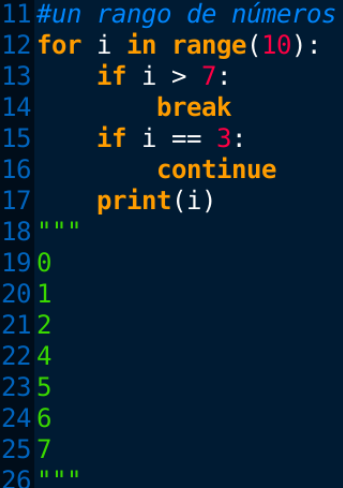
* Produce iteraciones recorriendo listas o rangos.



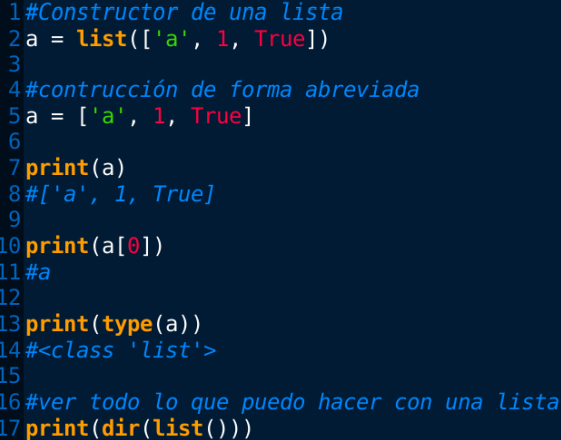
* No cambiar la cantidad de elementos de un array cuando se usa con un for. De necesitar agregar o quitar trabajar con una copia.

**BREAK, CONTINUE**

* Forma de cortar un bucle (break) o forzar al siguiente ciclo normalmente saltando un bloque de código.



**SECUENCIAS**: Listas, Tuplas, Rangos



**CONJUNTOS (set)**: colección no ordenada de eleméntos únicos

**MAPAS (diccionarios):** estructura contenedera de elementos en pares clave => valor. Las claves deben ser únicas.

**RECURSIVIDAD:**

* Forma de iterar en donde una función se llama a sí misma.
* También hay que tener cuidado de no llegar a un bucle infinito.
* Una función recursiva siempre tendrá:
* Al menos una llamada a sí misma, sino no es recursiva.
* Al menos un if con el caso base que corta la recursividad.
* Python tiene un límite de 1000 llamados recursivos de profundidad

