

Guión | Vídeo sobre ciclos While

Hola a todos, en este tutorial vamos a hablar sobre los ciclos *While*. Este ciclo es una estructura de control de flujo que nos permite repetir una sección de código hasta que se cumpla una condición.

Para utilizar el ciclo *While*, necesitamos establecer una condición que se evalúe como Verdadera o Falsa. Mientras la condición sea Verdadera, el código dentro del ciclo se ejecutará. Veamos un ejemplo sencillo:

```
i = 0
while i < 5:
    print("El valor de i es:", i)
    i = i + 1
```

En este ejemplo, estamos inicializando una variable llamada *i* con el valor de 0. Luego, mientras *i* sea menor que 5, se imprimirá en pantalla el valor de *i* y se le sumará 1 a la variable en cada iteración.

Es importante destacar que si la condición nunca se evalúa como Falsa, el ciclo se ejecutará de forma infinita. Por eso, es importante asegurarnos de que en algún momento la condición se vuelva falsa para que el ciclo termine.

También podemos utilizar la palabra clave *break* para salir del ciclo en cualquier momento. Veamos un ejemplo:

```
i = 0
while i < 10:
    print("El valor de i es:", i)
    if i == 5:
        break
    i += 1
```

En este caso, el ciclo se ejecutará hasta que *i* sea igual a 5, momento en el que se utiliza la palabra clave *break* para salir del ciclo.

En resumen, el ciclo *While* es una estructura de control de flujo que nos permite repetir una sección de código mientras se cumpla una condición. Es importante asegurarnos de que la condición se vuelva Falsa en algún momento para evitar que el ciclo se ejecute de forma infinita. También podemos utilizar la palabra clave *break* para salir del ciclo en cualquier momento. Espero que este tutorial haya sido útil para entender los ciclos *While*.

Hola, en el siguiente tutorial veremos que es el ciclo while y cómo funciona. El ciclo while es una estructura de control que permite ejecutar una serie de instrucciones de manera repetida y tantas veces sea necesario. Su sintaxis se define de la siguiente manera: primero, tenemos la palabra reservada while, a continuación, se define una condición, y por último, se enuncia el bloque de instrucciones que se desea ejecutar en cada iteración del ciclo.

Si nos damos cuenta, esta estructura se asemeja a la de un condicional if, como el que vemos abajo en pantalla. Recordemos cómo funciona: primero, validamos que la condición que sigue a la palabra reservada se cumpla; si es así, el bloque a continuación se ejecuta. De lo contrario, el bloque se salta y la ejecución del programa continúa en la siguiente expresión.

Entonces, un condicional if se ejecuta 0 o 1 veces, dependiendo de si la condición definida es verdadera o no.

En un ciclo while ocurre algo similar: si la condición declarada es verdadera, se ejecuta el bloque de código contenido, pero aquí, en vez de finalizar la expresión y continuar con la siguiente, volvemos al inicio de esta, y repetimos el mismo proceso una y otra vez siempre y cuando la condición se siga cumpliendo, el bloque se repite un número n de veces, hasta que la condición deje de cumplirse y se convierta en falsa.

A continuación se muestran algunos ejemplos de ciclos while. En este caso en particular, lo primero que hacemos es inicializar la variable i con el valor 0. En seguida, declaramos el inicio del ciclo while y definimos una condición para que este se ejecute: la condición en este caso depende del valor de i; mientras el valor de i sea menor a 10, el ciclo se va a ejecutar. Dado que el valor actual cumple con la condición, entonces entramos al bloque de código del ciclo.

Ya dentro del bloque, se imprime en pantalla el valor actual de i, y enseguida, aumentamos 1 al valor actual de la variable y volvemos nuevo al inicio del ciclo while, validamos de nuevo la condición y ejecutamos el bloque, así una y otra vez mientras la condición de que i sea menor a 10 se siga cumpliendo. Al llegar a la décima iteración del ciclo, el valor de i ha alcanzado el valor de 10, por lo que al regresar al inicio del ciclo, la condición ya no es satisfecha y el ciclo termina ahí.

Frecuentemente, estaremos utilizando ciclos para recorrer ciertas estructuras de datos, como el caso de los arreglos. En el siguiente ejemplo, imprimiremos los elementos del siguiente arreglo, y para ello nos valdremos de un ciclo while y el valor de una variable: la variable i será utilizada como el índice del arreglo, para recorrer uno a uno sus elementos, empezando por el elemento 0, el cual imprimimos en pantalla con la función print, para luego aumentar el valor del índice y volver al inicio del ciclo, y así sucesivamente hasta que hayamos recorrido todo los elementos del arreglo

Eso sería todo, espero que lo anterior te haya ayudado a entender mejor cómo funcionan los ciclos y a visualizar maneras en que podrías implementarlos.