

Desc.

Uso y entendimiento de los bucles, así como cuando aplicarlos, diferenciarlos y hacer un uso correcto de estos.

For

Utilizado para iterar sobre secuencias o cualquier objeto iterable, existen muchos tipos de for aunque el más utilizado es siempre usando el índice.

For in

El **For in** captura todos los elementos de un objeto y por cada uno de ellos realiza una acción.

Sintaxis:

```
for variable in secuencia:  
    # Código que se ejecuta en cada iteración
```

Iterar sobre una lista de elementos

```
lista = [1, 2, 3, 4, 5]  
for elemento in lista:  
    # Por cada elemento haz X
```

Range()

Range nos permite ejecutar **X** veces.

```
for i in range(5):  
    # Se ejecuta 5 veces
```

Enumerate()

Obtiene índices y el valor de los elementos en una secuencia.

```
for indice, valor in enumerate(secuencia):  
    # Hacer algo con indice y valor
```

Diccionarios

```
diccionario = {  
    "Nombre": "yo",  
    "ataque": 10,  
    "vida": 100.0,  
    "defensa": 20  
}  
  
for clave in diccionario:  
    # Hacer algo con clave  
  
for clave, valor in diccionario.items():  
    # Hacer algo con clave y valor
```

While

Enbuclo un bloque de código mientras una condición sea verdadera.

Sintaxis:

```
while condicion:  
    # Código que se ejecuta mientras la condición sea verdadera
```

```
contador = 0  
while contador < 5:  
    print(contador)  
    contador += 1
```

While | For Else

For else

Se ejecuta una vez el bucle ha finalizado.

```
for elemento in secuencia:
    # Hacer algo con elemento
else:
    # Código a ejecutar si el bucle termina
```

```
lista = [1,2,3,4,5]

for num in lista:
    print(num)
else:
    # Código a ejecutar si el bucle termina
```

While else

Se ejecuta cuando la condición es **falsa**.

```
while condicion:
    # True
else:
    # False
```

```
contador = 0
while contador < 5:
    print(contador)
    contador += 1
else:
    # ...
```

Pirámide de carácter

Crearemos una pirámide por terminal, el objetivo de este ejercicio es entender el código y saber porque ocurre el output recibido.

Procedimiento

- Constantes: Alto pirámide y carácter a construir
- Mostrar el resultado en terminal
- `input()` pidiendo el carácter a construir y como de alta debe ser la pirámide

Ejercicio

```
# Solicitar al usuario altura y carácter

# Mostrado en pantalla
```

Amogus

Deberemos construir un among us por consola siguiendo las siguientes reglas:

- 10 jugadores (el usuario debe escribir sus nombres)
- Uso de un While y un For
- Asignar un impostor aleatorio a un jugador
- Tres intentos de adivinar quien es el impostor

Ejercicio

```
# Pedir nombres al usuario

# Asignación de impostor

# Empieza el juego

# Mostramos si ganamos o no
```

Resultado

Ejercicio 1

```
altura = int(input("Introduce la altura de la pirámide: "))
caracter = input("Introduce el carácter que deseas usar para construir la pirámide: ")

for i in range(altura):
    print(' ' * (altura - i - 1) + caracter * (2 * i + 1))
```

Ejercicio 2

```
import random

# Solicitar al usuario que ingrese los nombres de los jugadores
print("Introduce los nombres de los jugadores:")
jugadores = [input("Jugador {}: ".format(i+1)) for i in range(10)]

# Elegir un impostor aleatorio de la lista de jugadores
impostor = random.choice(jugadores)

print("\n¡Bienvenido al juego Encuentra al Impostor!")
print("Debes adivinar quién es el impostor entre los jugadores.")
print("Tienes 3 intentos para adivinar correctamente.")

intentos = 0

# Bucle while para continuar el juego hasta que el jugador agote los intentos o
# adivine al impostor
while intentos < 3:
    # Mostrar los nombres de los jugadores para que el jugador haga su elección
    print("\nElige un jugador:")
    for index, jugador in enumerate(jugadores, start=1):
        print(f"{index}. {jugador}")

    opcion = int(input("Introduce el número del jugador que crees que es el
impostor: "))

    # Verificar si la opción seleccionada es el impostor
    if opcion < 1 or opcion > len(jugadores):
        print("Opción inválida. Inténtalo de nuevo.")
        continue

    jugador_elegido = jugadores[opcion - 1]

    if jugador_elegido == impostor:
        print("¡Felicitades! ¡Has encontrado al impostor! El impostor era:",
impostor)
        break
    else:
        print("Ese jugador no es el impostor. Inténtalo de nuevo.")
        intentos += 1

# Si el jugador agotó los intentos sin adivinar al impostor, mostrar al impostor
if intentos == 3:
```

```
print("\n¡Lo siento! Has agotado tus intentos. El impostor era:", impostor)
```