

# [Bài đọc] Cấu trúc lặp while

Trong Python vòng lặp while được sử dụng để thực thi một khối lệnh lặp đi lặp lại chừng nào điều kiện của vòng lặp vẫn còn đúng. Khi điều kiện sai, các lệnh sau vòng lặp while mới được thực hiện. Vòng lặp while thường được sử dụng trong các trường hợp không xác định được trước số vòng lặp.

## Cú pháp:

**while <điều kiện>:**

các khối lệnh **Ví dụ:** Chương trình in ra 10 từ 'Hello world'

```
count = 0
while count < 10:
    count += 1
    print("Hello world")
```

Kết quả của đoạn mã:

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Ta thấy trước vòng lặp while, biến count có giá trị = 0, do đó thỏa mãn điều kiện < 10, nên các câu lệnh bên trong while được thực hiện, in ra dòng thứ nhất và count lúc này có giá trị = 1. Xong bước 1, count tiếp tục đem so sánh với điều kiện, vẫn thỏa mãn <10, tiếp tục thực thi khối lệnh. Cứ như thế đến sau bước thứ 10, count có giá trị là 10, so sánh điều kiện không thỏa mãn < 10. Chương trình không tiếp tục thực thi các khối lệnh trong while mà kết thúc. (Vì không có lệnh nào sau while)

*Lưu ý: Khối lệnh trong vòng lặp while phải đảm bảo đến một lúc nào đó, điều kiện sẽ không đúng, để tránh trường hợp lặp vô hạn, gây đứng chương trình.*

Trong ví dụ trên, thử xóa dòng code thứ nhất trong vòng lặp while, xem kết quả.

### Vòng lặp while kết hợp với else

#### Cú pháp:

```
while <điều kiện>:  
    các khối lệnh  
else:  
    các khối lệnh
```

#### Ví dụ:

```
count = 0  
while count < 10:  
    count += 1  
    print("Hello world")  
else:  
    print("Kết thúc vòng lặp while")
```

Kết quả của đoạn mã:

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Hello world

Kết thúc vòng lặp while

Nhận thấy về phương thức hoạt động tương tự như cú pháp của khối lệnh if else. Ở đây chỉ khác với if else là khi điều kiện còn đúng thì các khối lệnh ở trong while được thực thi lặp nhiều lần còn trong if else thì chỉ một lần (Tất nhiên trong điều kiện if không có chứa while hoặc for)

## [Bài đọc] Lệnh break, continue

### Lệnh break

Lệnh break trong Python được sử dụng để điều khiển thoát khỏi vòng lặp khi thỏa mãn một điều kiện bên ngoài nào đó. Lệnh break được đặt ở trong vòng lặp

### Cú pháp

```
break
```

**Ví dụ:** Dùng vòng lặp while để kiểm tra trong chuỗi có kí tự 'a' hay không?

```
text = input("Nhập vào một chuỗi: ")
index = 0
answer = False
while index < len(text):
    if (text[index] == 'a' or text[index] == 'A'):
        answer = True
        break
    index += 1
if answer:
    print('Chuỗi nhập vào có chứa kí tự a')
else:
    print('Chuỗi không chứa kí tự a')
```

Nhận xét: Ta thấy rằng, yêu cầu đề bài chỉ kiểm tra sự tồn tại của a hoặc A trong chuỗi, nên khi phát hiện thỏa mãn điều kiện là tìm thấy a hoặc A, thì break ra khỏi vòng lặp, không cần kiểm tra các kí tự khác sau đó của chuỗi nữa. Trong trường hợp này, break làm cho chương trình xử lý nhanh hơn, tiết kiệm tài nguyên, đặc biệt trong các bài toán xử lý tính toán phức tạp, số vòng lặp lớn.

### Lệnh continue

Lệnh continue cũng là một lệnh điều khiển vòng lặp giống như break. Chỉ khác ở chỗ, continue thay vì thoát ra khỏi vòng lặp như break thì continue bỏ qua vòng lặp hiện tại và tiến hành thực thi ở bước lặp kế tiếp.

### Cú pháp

```
continue
```

**Ví dụ:** Chương trình in ra các số chẵn từ 1 đến 10 sử dụng continue

```
count = 0
while count < 10:
    count += 1
    if count % 2 == 1:
        continue
```

Ta thấy rằng ở ví dụ trên, khi count là các số lẻ thì thỏa mãn điều kiện if, do đó lệnh continue được thực thi, các câu lệnh sau continue được bỏ qua, không được thực thi trong bước hiện tại (nghĩa là không in ra số lẻ). Chương trình tiếp tục với bước tiếp theo của vòng lặp while.