

TÁC GIẢ: 8 SYNC

CỘNG ĐỒNG



[Kevin Nguyễn](#)



[Nhóm Chia Sẻ Công Nghệ](#)



[Nhóm BlockChain](#)



[Tiktok: 8 Sync](#)



[Youtube: 8 Sync Dev](#)



[Zalo](#)

KHÓA HỌC:



[Fullstack Python](#)



[Fullstack Nextjs](#)



[Fullstack Android-IOS](#)

Tài liệu sẽ được cập nhật định kì và thông báo trong group nên các bạn chú ý nhen .



PYTHON - VÒNG LẶP FOR

GIỚI THIỆU

Vòng lặp for trong Python có khả năng lặp qua các phần tử của bất kỳ chuỗi nào, chẳng hạn như một list, tuple hoặc một chuỗi.

CÚ PHÁP

< 8 Sync Dev />

```
for biến_lặp in chuỗi:  
    câu_lệnh(s)
```

Nếu một chuỗi chứa một danh sách biểu thức, nó sẽ được đánh giá trước. Sau đó, phần tử đầu tiên (tại chỉ mục 0) trong chuỗi được gán cho biến lặp.

Tiếp theo, khối câu lệnh được thực thi. Mỗi phần tử trong danh sách được gán cho biến lặp, và khối câu lệnh được thực thi cho đến khi toàn bộ chuỗi được tiêu thụ.

LOẠI VÒNG LẶP TRONG PYTHON

- Ngôn ngữ lập trình Python cung cấp các loại vòng lặp sau để xử lý yêu cầu lặp lại:
 1. **Vòng lặp while:** Lặp lại một câu lệnh hoặc nhóm câu lệnh trong khi một điều kiện nhất định là ĐÚNG. Nó kiểm tra điều kiện trước khi thực thi thân vòng lặp.
 2. **Vòng lặp for:** Thực thi một chuỗi các câu lệnh nhiều lần và viết tắt mã quản lý biến vòng lặp.
 3. **Vòng lặp lồng nhau:** Bạn có thể sử dụng một hoặc nhiều vòng lặp bên trong bất kỳ vòng lặp while, for hoặc do..while nào khác.

VÍ DỤ

1. VÒNG LẶP FOR VỚI CHUỖI

< 8 Sync Dev />

```
zen = '''  
Beautiful is better than ugly.  
Explicit is better than implicit.
```

```
Simple is better than complex.  
Complex is better than complicated.  
'''  
  
for char in zen:  
    if char not in 'aeiou':  
        print(char, end='')
```

2. VÒNG LẶP FOR VỚI TUPLES

```
numbers = (34, 54, 67, 21, 78, 97, 45, 44, 80, 19)  
total = 0  
for num in numbers:  
    total += num  
print("Tổng =", total)
```

3. VÒNG LẶP FOR VỚI LIST

```
numbers = [34, 54, 67, 21, 78, 97, 45, 44, 80, 19]  
for num in numbers:  
    if num % 2 == 0:  
        print(num)
```

4. VÒNG LẶP FOR VỚI ĐỐI TƯỢNG RANGE



< 8 Sync Dev />

```
for num in range(5):  
    print(num, end=' ')  
print()  
  
for num in range(10, 20):  
    print(num, end=' ')  
print()  
  
for num in range(1, 10, 2):  
    print(num, end=' ')
```

5. VÒNG LẶP FOR VỚI CÁC CHỈ SỐ CỦA CHUỖI



< 8 Sync Dev />

```
numbers = [34, 54, 67, 21, 78]  
indices = range(len(numbers))  
for index in indices:  
    print("index:", index, "number:", numbers[index])
```

6. VÒNG LẶP FOR VỚI TỪ ĐIỂN



< 8 Sync Dev />

```
numbers = {10: "Ten", 20: "Twenty", 30: "Thirty", 40: "Forty"}  
for x in numbers:  
    print(x)
```

```
numbers = {10: "Ten", 20: "Twenty", 30: "Thirty", 40: "Forty"}  
for x in numbers:  
    print(x, ":", numbers[x])
```

```
numbers = {10: "Ten", 20: "Twenty", 30: "Thirty", 40: "Forty"}  
for x in numbers.items():  
    print(x)
```

```
numbers = {10: "Ten", 20: "Twenty", 30: "Thirty", 40: "Forty"}  
for x, y in numbers.items():  
    print(x, ":", y)
```

KẾT LUẬN

Với vòng lặp for trong Python, bạn có thể dễ dàng lặp qua các phần tử của một chuỗi, danh sách, tuple hoặc từ điển và thực thi các câu lệnh cho mỗi phần tử. Điều này giúp

code của bạn trở nên gọn gàng và dễ hiểu.