

# 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 .



## HƯỚNG DẪN SỬ DỤNG CÂU LỆNH IF-ELSE TRONG PYTHON

---

## MỤC ĐÍCH

Tài liệu này cung cấp hướng dẫn chi tiết về cách sử dụng câu lệnh if-else trong Python. Nó mô tả cách sử dụng if-else để kiểm tra điều kiện và thực thi các khối mã tương ứng dựa trên kết quả của điều kiện đó.

# CÂU LỆNH IF-ELSE TRONG PYTHON

## CÚ PHÁP

Câu lệnh if-else trong Python có cú pháp như sau:



< 8 Sync Dev />

```
if điều_kiện:
    # Khối mã được thực thi
    # khi điều_kiện là True
else:
    # Khối mã được thực thi
    # khi điều_kiện là False
```

## VÍ DỤ VỀ CÂU LỆNH IF-ELSE

Để minh họa, xem ví dụ sau:



< 8 Sync Dev />

```
age = 25
print("Tuổi: ", age)
if age >= 18:
    print("Bạn đủ điều kiện để bỏ phiếu")
else:
    print("Bạn không đủ điều kiện để bỏ phiếu")
```

Đầu tiên, thiết lập biến nguyên "age" thành 25.

Sau đó, sử dụng câu lệnh if với biểu thức "age>18" theo sau là ":" bắt đầu một khối mã; điều này sẽ được thực hiện nếu "age>=18" là True.

Để cung cấp khối else, sử dụng else: khối mã được thụt vào tiếp theo chứa thông báo "Không đủ điều kiện để bỏ phiếu" sẽ được thực hiện khi "age>=18" là False.

Khi thực thi mã này, bạn sẽ nhận được kết quả sau:

```

    < 8 Sync Dev />

    Tuổi: 25
    Bạn đủ điều kiện để bỏ phiếu

```

Để kiểm tra khối else, hãy thay đổi tuổi thành 12 và chạy lại mã.

```

    < 8 Sync Dev />

    Tuổi: 12
    Bạn không đủ điều kiện để bỏ phiếu

```

## CÂU LỆNH IF ELIF ELSE TRONG PYTHON

Câu lệnh if elif else cho phép bạn kiểm tra nhiều biểu thức và thực thi một khối mã ngay khi một trong các điều kiện đánh giá là True.

Tương tự như câu lệnh else, câu lệnh elif là tùy chọn. Tuy nhiên, khác với else, chỉ có thể có tối đa một câu lệnh elif; có thể có một số câu lệnh elif sau một câu lệnh if.

## CÚ PHÁP CỦA CÂU LỆNH IF ELIF ELSE TRONG PYTHON

```

    < 8 Sync Dev />

```



```
if biểu_thức1:
    khối mã
elif biểu_thức2:
    khối mã
elif biểu_thức3:
    khối mã
else:
    khối mã
```

## CÁCH HOẠT ĐỘNG CỦA CÂU LỆNH IF ELIF ELSE

Hãy hiểu cách câu lệnh if elif else hoạt động, với ví dụ sau.

Cấu trúc giảm giá được sử dụng trong ví dụ trước đã được sửa đổi thành các mức giảm giá khác nhau:

- 20% trên số tiền vượt quá 10000,
- 10% cho số tiền giữa 5-10000,
- 5% nếu nó nằm giữa 1 đến 5000,
- không giảm giá nếu số tiền <1000.

### VÍ DỤ


Chúng ta có thể viết mã Python cho logic trên với câu lệnh if-else:

```
amount = 2500
print('Số tiền = ', amount)
if amount > 10000:
    discount = amount * 20 / 100
else:
    if amount > 5000:
```

```
discount = amount * 10 / 100
else:
    if amount > 1000:
        discount = amount * 5 / 100
    else:
        discount = 0

print('Số tiền phải trả = ', amount - discount)
```

Đặt số tiền để kiểm tra tất cả các điều kiện có thể: 800, 2500, 7500 và 15000. Kết quả sẽ thay đổi tương ứng.




< 8 Sync Dev />

```
Số tiền: 800
Số tiền phải trả = 800
Số tiền: 2500
Số tiền phải trả = 2375.0
Số tiền: 7500
Số tiền phải trả = 6750.0
Số tiền: 15000
Số tiền phải trả = 12000.0
```

Mặc dù mã sẽ hoạt động hoàn hảo, nhưng nếu bạn nhìn vào việc thụt vào cấp độ tăng dần ở mỗi câu lệnh if và else, sẽ trở nên khó quản lý nếu còn nhiều điều kiện hơn.

Câu lệnh elif giúp mã dễ đọc và hiểu hơn.

## VÍ DỤ VỀ CÂU LỆNH IF ELIF ELSE TRONG PYTHON



< 8 Sync Dev />

```
amount = 2500
print('Số tiền = ', amount)
if amount > 10000:
    discount = amount * 20 / 100
elif amount > 5000:
    discount = amount * 10 / 100
elif amount > 1000:
    discount = amount * 5 / 100
else:
    discount = 0

print('Số tiền phải trả = ', amount - discount)
```

Đặt số tiền để kiểm tra tất cả các điều kiện có thể: 800, 2500, 7500 và 15000. Kết quả sẽ thay đổi tương ứng.



< 8 Sync Dev />

```
Số tiền: 800
Số tiền phải trả = 800
Số tiền: 2500
Số tiền phải trả = 2375.0
Số tiền: 7500
Số tiền phải trả = 6750.0
Số tiền: 15000
Số tiền phải trả = 12000.0
```

Việc sử dụng câu lệnh elif giúp mã trở nên ngắn gọn và dễ đọc hơn, vì không cần thiết phải thụt vào mỗi lần có một điều kiện mới. Câu lệnh elif cũng giúp mã trở nên linh hoạt hơn khi cần thêm điều kiện kiểm tra.