

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 .



TOÁN TỬ THÀNH VIÊN TRONG PYTHON

Toán tử thành viên trong Python giúp chúng ta xác định xem một phần tử có xuất hiện trong một đối tượng kiểu container cho trước hay không, hoặc nói cách khác, xem một phần tử có là thành viên của đối tượng kiểu container cho trước không.

LOẠI TOÁN TỬ THÀNH VIÊN PYTHON

Python có hai toán tử thành viên: in và not in. Cả hai đều trả về kết quả là Boolean. Kết quả của toán tử in là ngược lại với toán tử not in.

TOÁN TỬ 'IN'

Toán tử "in" được sử dụng để kiểm tra xem một chuỗi con có hiện diện trong một chuỗi lớn hơn, bất kỳ phần tử nào có hiện diện trong một danh sách hoặc tuple, hoặc một danh sách con hoặc tuple con được bao gồm trong một danh sách hoặc tuple.

VÍ DỤ VỀ TOÁN TỬ "IN" TRONG PYTHON



< 8 Sync Dev />

```
var = "8 Sync Dev"  
a = "P"  
b = "tor"  
c = "in"  
d = "To"  
  
print (a, "in", var, ":", a in var)  
print (b, "in", var, ":", b in var)  
print (c, "in", var, ":", c in var)  
print (d, "in", var, ":", d in var)
```

Kết quả sẽ là:



< 8 Sync Dev />

```
P in 8 Sync Dev : True  
tor in 8 Sync Dev : True
```

```
in in 8 Sync Dev : True  
To in 8 Sync Dev : False
```

TOÁN TỬ 'NOT IN'

Toán tử "not in" được sử dụng để kiểm tra xem một chuỗi với giá trị cho trước không hiện diện trong đối tượng như chuỗi, danh sách, tuple, v.v.

VÍ DỤ VỀ TOÁN TỬ "NOT IN" TRONG PYTHON



< 8 Sync Dev />

```
var = "8 Sync Dev"  
a = "P"  
b = "tor"  
c = "in"  
d = "To"  
  
print (a, "not in", var, ":", a not in var)  
print (b, "not in", var, ":", b not in var)  
print (c, "not in", var, ":", c not in var)  
print (d, "not in", var, ":", d not in var)
```

Kết quả sẽ là:



< 8 Sync Dev />

```
P not in 8 Sync Dev : False  
tor not in 8 Sync Dev : False  
in not in 8 Sync Dev : False  
To not in 8 Sync Dev : True
```


TOÁN TỬ THÀNH VIÊN VỚI DANH SÁCH VÀ TUPLE

Bạn có thể sử dụng toán tử "in/not in" để kiểm tra sự hiện diện của một phần tử trong danh sách hoặc tuple.



< 8 Sync Dev />

```
var = [10,20,30,40]
a = 20
b = 10
c = a-b
d = a/2
print (a, "in", var, ":", a in var)
print (b, "not in", var, ":", b not in var)
print (c, "in", var, ":", c in var)
print (d, "not in", var, ":", d not in var)
```

Kết quả sẽ là:



< 8 Sync Dev />

```
20 in [10, 20, 30, 40] : True
10 not in [10, 20, 30, 40] : False
10 in [10, 20, 30, 40] : True
10.0 not in [10, 20, 30, 40] : False
```

Trong trường hợp cuối cùng, "d" là một số thực nhưng vẫn so sánh với True với 10 (một số nguyên) trong danh sách. Ngay cả khi một số được biểu diễn dưới các định dạng khác nhau như nhị phân, bát phân hoặc thập lục phân được cung cấp, các toán tử thành viên vẫn cho biết liệu nó có trong chuỗi hay không.

VÍ DỤ

Tuy nhiên, nếu bạn thử kiểm tra xem hai số kế tiếp có xuất hiện trong một danh sách hoặc tuple không, toán tử in sẽ trả về False. Nếu danh sách/tuple chứa các số kế tiếp như một chuỗi chính nó, thì nó sẽ trả về True.



< 8 Sync Dev />

```
var = (10,20,30,40)
a = 10
b = 20
print ((a,b), "in", var, ":", (a,b) in var)
var = ((10,20),30,40)
a = 10
b = 20
print ((a,b), "in", var, ":", (a,b) in var)
```

Kết quả sẽ là:



< 8 Sync Dev />

```
(10, 20) in (10, 20, 30, 40) : False
(10, 20) in ((10, 20), 30, 40) : True
```

TOÁN TỬ THÀNH VIÊN VỚI TẬP HỢP

Toán tử thành viên của Python cũng hoạt động tốt với các đối tượng tập hợp.



< 8 Sync Dev />

```
var = {10,20,30,40}
a = 10
b = 20
print (b, "in", var, ":", b in var)
var = {(10,20),30,40}
```

```
a = 10
b = 20
print ((a,b), "in", var, ":", (a,b) in var)
```

Kết quả sẽ là:

```
< 8 Sync Dev />

20 in {40, 10, 20, 30} : True
(10, 20) in {40, 30, (10, 20)} : True
```

TOÁN TỬ THÀNH VIÊN VỚI TỪ ĐIỂN

Sử dụng các toán tử in cũng như not in với đối tượng từ điển là được cho phép. Tuy nhiên, Python chỉ kiểm tra sự thành viên với tập hợp của các khóa và không phải giá trị.

```
< 8 Sync Dev />

var = {1:10, 2:20, 3:30}
a = 2
b = 20
print (a, "in", var, ":", a in var)
print (b, "in", var, ":", b in var)
```

Kết quả sẽ là:

```
< 8 Sync Dev />

2 in {1: 10, 2: 20, 3: 30} : True
20 in {1: 10, 2: 20, 3: 30} : False
```

Đó là một bản dịch chi tiết và rõ ràng về toán tử thành viên trong Python. Hy vọng nó giúp bạn hiểu rõ về cách sử dụng các toán tử này trong các tình huống khác nhau.

