

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 .



PHƯƠNG THỨC CHUỖI TRONG PYTHON

Lớp str tích hợp sẵn trong Python định nghĩa các phương thức khác nhau. Chúng hỗ trợ trong việc xử lý chuỗi. Do chuỗi là một đối tượng không thể thay đổi, các phương thức này trả về một bản sao của chuỗi gốc, thực hiện các xử lý tương ứng trên nó.

Các phương thức chuỗi có thể được phân loại trong các nhóm sau –

1. Chuyển đổi chữ hoa và chữ thường
2. Căn chỉnh
3. Tách và kết hợp
4. Kiểm tra logic
5. Tìm kiếm và thay thế
6. Định dạng
7. Dịch

CHUYỂN ĐỔI CHỮ HOA VÀ CHỮ THƯỜNG

Các phương thức trong nhóm này hỗ trợ chuyển đổi các ký tự của chuỗi thành chữ hoa hoặc chữ thường.

Ví dụ:



< 8 Sync Dev />

```
s = "hello world"
print(s.upper()) # Chuyển thành chữ hoa
print(s.capitalize()) # Chuyển ký tự đầu tiên thành chữ hoa
print(s.title()) # Chuyển thành chữ in hoa theo tiêu đề
```

CĂN CHỈNH

Các phương thức trong nhóm này hỗ trợ căn chỉnh các chuỗi.

Ví dụ:



< 8 Sync Dev />

```
s = "hello"
print(s.center(20)) # Căn chỉnh giữa
```

```
print(s.ljust(20)) # Căn chỉnh trái  
print(s.rjust(20)) # Căn chỉnh phải
```

TÁCH VÀ KẾT HỢP

Các phương thức trong nhóm này hỗ trợ tách chuỗi thành các phần và kết hợp các phần thành một chuỗi mới.

Ví dụ:



< 8 Sync Dev />

```
s = "apple,banana,orange"  
print(s.split(',')) # Tách chuỗi thành danh sách các phần tử  
print('-'.join(['apple', 'banana', 'orange'])) # Kết hợp các  
phần tử thành chuỗi mới
```

KIỂM TRA LOGIC

Các phương thức trong nhóm này hỗ trợ kiểm tra các điều kiện logic trên chuỗi.

Ví dụ:



< 8 Sync Dev />

```
s = "hello world"  
print(s.startswith("hello")) # Kiểm tra xem chuỗi có bắt đầu  
bằng "hello" không  
print(s.endswith("world")) # Kiểm tra xem chuỗi có kết thúc  
bằng "world" không  
print(s.isalnum()) # Kiểm tra xem chuỗi có chứa toàn bộ các ký  
tự hoặc số không  
print(s.isdigit()) # Kiểm tra xem chuỗi có phải là một số không  
print(s.islower()) # Kiểm tra xem tất cả các ký tự trong chuỗi
```


có phải là chữ thường không

```
print(s.isupper()) # Kiểm tra xem tất cả các ký tự trong chuỗi  
có phải là chữ hoa không
```

TÌM KIẾM VÀ THAY THẾ

Các phương thức trong nhóm này hỗ trợ tìm kiếm và thay thế các phần của chuỗi.

Ví dụ:



< 8 Sync Dev />

```
s = "hello world"  
print(s.find("world")) # Tìm vị trí đầu tiên của "world" trong  
chuỗi  
print(s.replace("world", "universe")) # Thay thế "world" bằng  
"universe" trong chuỗi
```

ĐỊNH DẠNG

Các phương thức trong nhóm này hỗ trợ định dạng chuỗi.

Ví dụ:



< 8 Sync Dev />

```
s = "My name is {name} and I am {age} years old."  
print(s.format(name="John", age=30)) # Sử dụng phương thức  
format()  
print(f"My name is {'John'} and I am {30} years old.") # Sử  
dụng f-string
```

DỊCH

Các phương thức trong nhóm này hỗ trợ dịch các ký tự trong chuỗi.

Ví dụ:



< 8 Sync Dev />

```
table = str.maketrans("aeiou", "12345")  
s = "apple"  
print(s.translate(table)) # Dịch các ký tự trong chuỗi theo  
bảng dịch đã cho
```