

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 .



BÀI 5. PYTHON - ĐỔI SỐ TÙY Ý

Có thể bạn muốn định nghĩa một hàm có thể chấp nhận số lượng đổi số tùy ý hoặc biến. Hơn nữa, số lượng đổi số tùy ý có thể là đổi số theo vị trí hoặc từ khóa.

Một đổi số được tiền tố bằng một dấu sao (*) cho các đổi số theo vị trí tùy ý.

Một đổi số được tiền tố bằng hai dấu sao (**) cho các đổi số từ khóa tùy ý.

VÍ DỤ VỀ ĐỐI SỐ TÙY Ý

Dưới đây là một ví dụ về đối số tùy ý hoặc biến chiều dài theo vị trí –

```
< 8 Sync Dev />

# Tổng các số
def add(*args):
    s = 0
    for x in args:
        s += x
    return s

result = add(10, 20, 30, 40)
print(result)

result = add(1, 2, 3)
print(result)
```

Biến args được tiền tố bằng "*" lưu trữ tất cả các giá trị được truyền cho nó. Ở đây, args trở thành một tuple. Chúng ta có thể chạy một vòng lặp qua các mục của nó để cộng các số.

Kết quả sẽ là:

```
< 8 Sync Dev />

100
6
```

ĐỐI SỐ BẮT BUỘC VỚI ĐỐI SỐ TÙY Ý

Cũng có thể có một hàm với một số đối số bắt buộc trước đây số lượng biến giá trị.

VÍ DỤ



< 8 Sync Dev />

```
# Trung bình của bài kiểm tra đầu tiên và tốt nhất của các bài
kiểm tra tiếp theo
def avg(first, *rest):
    second = max(rest)
    return (first + second) / 2

result = avg(40, 30, 50, 25)
print(result)
```

Cuộc gọi sau đến hàm avg() truyền giá trị đầu tiên cho đối số bắt buộc đầu tiên, và các giá trị còn lại cho một tuple có tên là rest. Sau đó, chúng ta tìm giá trị tối đa và sử dụng nó để tính trung bình.

Kết quả sẽ là:



< 8 Sync Dev />

45.0

ĐỐI SỐ TÙY Ý TỪ KHÓA (**KWARGS)

Nếu một biến trong danh sách đối số có hai dấu sao (*) được tiền tố cho nó, hàm có thể chấp nhận số lượng đối số từ khóa tùy ý. Biến trở thành một từ điển các cặp từ khóa: giá trị.

VÍ DỤ

Dưới đây là một ví dụ về một hàm với các đối số từ khóa tùy ý. Hàm addr() có một đối số **kwargs có thể chấp nhận bất kỳ số lượng phần tử địa chỉ nào như tên,

thành phố, số điện thoại, mã pin, v.v. Bên trong hàm, từ điển kwargs của các cặp khóa từ khóa: giá trị được duyệt qua sử dụng phương thức items().



< 8 Sync Dev />

```
def addr(**kwargs):  
    for k, v in kwargs.items():  
        print("{}:{}".format(k, v))  
  
print("pass two keyword args")  
addr(Name="John", City="Mumbai")  
print("pass four keyword args")  
  
# pass four keyword args  
addr(Name="Raam", City="Mumbai", ph_no="9123134567",  
PIN="400001")
```

Kết quả sẽ là:



< 8 Sync Dev />

```
pass two keyword args  
Name:John  
City:Mumbai  
pass four keyword args  
Name:Raam  
City:Mumbai  
ph_no:9123134567  
PIN:400001
```

NHIỀU ĐỐI SỐ VỚI ĐỐI SỐ TÙY Ý TỪ
KHÓA

Nếu hàm sử dụng các loại đối số kết hợp, các đối số từ khóa tùy ý phải đứng sau các đối số theo vị trí, theo từ khóa và các đối số theo vị trí tùy ý trong danh sách đối số.

VÍ DỤ

Hãy tưởng tượng một trường hợp trong đó môn khoa học và toán học là bắt buộc, ngoài ra học sinh có thể chọn bất kỳ số lượng môn học tự chọn nào.

Dưới đây là định nghĩa của một hàm `percent()` trong đó điểm số trong khoa học và điểm số được lưu trữ trong các đối số tùy ý `**optional`.



< 8 Sync Dev />

```
def percent(math, sci, **optional):
    print("maths:", math)
    print("sci:", sci)
    s = math + sci
    for k, v in optional.items():
        print("{}:{}".format(k, v))
        s += v
    return s / (len(optional) + 2)

result = percent(math=80, sci=75, Eng=70, Hist=65, Geo=72)
print("percentage:", result)
```

Kết quả sẽ là:



< 8 Sync Dev />

```
maths: 80
sci: 75
Eng:70
Hist:65
Geo:72
percentage: 72.4
```

