

lambda

`lambda` trong Python là một **hàm ẩn danh (anonymous function)** – tức là một hàm không cần đặt tên.

Nó thường được dùng khi bạn cần một hàm nhỏ, nhanh gọn, và chỉ dùng **một lần** hoặc **dùng ngay tại chỗ** (ví dụ: trong `sorted`, `map`, `filter`, `reduce`, v.v.)

```
lambda arguments: expression
```

- **arguments**: là danh sách các đối số (giống như trong `def`)
- **expression**: là giá trị sẽ được trả về (chỉ 1 dòng)

Ví dụ cơ bản

```
add = lambda x, y: x + y
print(add(2, 3)) # 5
```

```
#Tương đương với:
def add(x, y):
    return x + y
```

Ứng dụng phổ biến

2. Dùng trong `sorted` để sắp xếp theo key

```
items = [(1, 'apple'), (3, 'banana'), (2, 'cherry')]
sorted_items = sorted(items, key=lambda item: item[1])
print(sorted_items) # [(1, 'apple'), (3, 'banana'), (2, 'cherry')]
```

2. Dùng với `map()`

```
nums = [1, 2, 3]
squared = list(map(lambda x: x**2, nums))
print(squared) # [1, 4, 9]
```

3. Dùng với `filter()`

```
nums = [1, 2, 3, 4, 5]
even = list(filter(lambda x: x % 2 == 0, nums))
print(even) # [2, 4]
```

4. Dùng trực tiếp (one-time use)

```
print((lambda name: f"Hello, {name}!")("VHTSoft")) # Hello, VHTSoft!
```

Lưu ý

- `lambda` **chỉ dùng cho các hàm ngắn gọn, một dòng**.
- Không thể có nhiều dòng hoặc câu lệnh `if`, `for`, `while` trong thân `lambda`.
- Khi logic phức tạp hơn, bạn nên dùng `def` để code dễ đọc hơn.

Một ví dụ thực tế kết hợp `lambda` và `Callable`

```
from typing import Callable

def run_operation(func: Callable[[int], int], number: int) -> int:
    return func(number)

print(run_operation(lambda x: x * 10, 5)) # 50
```

Tác giả: Đỗ Ngọc Tú
Công Ty Phần Mềm VHTSoft

Phiên bản #1

Được tạo 18 tháng 4 2025 03:40:24 bởi Đỗ Ngọc Tú

Được cập nhật 18 tháng 4 2025 04:46:33 bởi Đỗ Ngọc Tú