

# Python cho Khoa học dữ liệu

## Bài 10: Kiểu dữ liệu danh sách - Python list

Hà Minh Tuấn  
hmtuan@hcmus.edu.vn

Khoa Toán - Tin học  
Trường Đại học Khoa học Tự nhiên, ĐHQG-HCM

Ngày 3 tháng 9 năm 2025

# Nội dung bài học

1. Giới thiệu chung về Tuple

2. Các methods quan trọng trong tuple

# Giới thiệu Tuple

- Tuple là một trong các kiểu dữ liệu dựng sẵn trong Python.
- Là một dãy các phần tử, phân tách bởi dấu phẩy, đặt trong dấu ngoặc tròn () .
- Các phần tử có thể thuộc nhiều kiểu dữ liệu khác nhau.
- Tuple là **immutable** (không thể thay đổi sau khi tạo).

## Ví dụ về Tuple

```
tup1 = ("Rohan", "Physics", 21, 69.75)
tup2 = (1, 2, 3, 4, 5)
tup3 = ("a", "b", "c", "d")
tup4 = (25.50, True, -55, 1+2j)
empty = ()
single = (50,)
print(tup1)
print(empty)
print(single)
```

```
('Rohan', 'Physics', 21, 69.75)
()
(50,)
```

## Truy xuất phần tử

```
tup1 = ('physics', 'chemistry', 1997, 2000)
tup2 = (1, 2, 3, 4, 5, 6, 7)
print("tup1[0] :", tup1[0])
print("tup2[1:5] :", tup2[1:5])
```

tup1[0] : physics

tup2[1:5] : (2, 3, 4, 5)

## Cập nhật Tuple (không hợp lệ)

```
tup1 = (12, 34.56)
tup2 = ('abc', 'xyz')

# Không thể thay đổi trực tiếp phần tử
# tup1[0] = 100 # lỗi

# Tạo tuple mới bằng phép cộng
tup3 = tup1 + tup2
print(tup3)
```

(12, 34.56, 'abc', 'xyz')

## Xoá Tuple

```
tup = ('physics', 'chemistry', 1997, 2000)
print(tup)
del tup
print("After deleting tup:")
print(tup)
```

('physics', 'chemistry', 1997, 2000)

After deleting tup:

Traceback (most recent call last):

...

NameError: name 'tup' is not defined

# Các phép toán với Tuple

```
print((1, 2, 3) + (4, 5, 6))      # Nối
print('Hi!',) * 4                  # Lặp
print(3 in (1, 2, 3))             # Membership
```

```
(1, 2, 3, 4, 5, 6)
('Hi!', 'Hi!', 'Hi!', 'Hi!')
True
```

# Indexing & Slicing

```
L = ('spam', 'Spam', 'SPAM!')
print(L[2])      # offset 0
print(L[-2])     # đếm từ phải
print(L[1:])     # cắt từ index 1
```

SPAM!

Spam

('Spam', 'SPAM!')

## Không cần ngoặc

```
print('abc', -4.24e93, 18+6.6j, 'xyz')
x, y = 1, 2
print("Value of x, y:", x, y)
```

abc -4.24e+93 (18+6.6j) xyz

Value of x, y: 1 2

## Hàm built-in với Tuple

- `len(tuple)`: Độ dài của tuple.
- `max(tuple)`: Phần tử lớn nhất.
- `min(tuple)`: Phần tử nhỏ nhất.
- `tuple(seq)`: Chuyển đổi sequence (vd: list) thành tuple.

## Ví dụ hàm built-in

```
numbers = (5, 2, 9, 1)
print(len(numbers))
print(max(numbers))
print(min(numbers))
print(tuple([10, 20, 30]))
```

4  
9  
1  
(10, 20, 30)

# Các phương thức trong Tuple

- Khác với List, tuple là kiểu dữ liệu bất biến (immutable).
- Vì vậy, tuple chỉ hỗ trợ một số phương thức cơ bản:
  - ▶ `count()` : Đếm số lần xuất hiện của một phần tử.
  - ▶ `index()` : Trả về vị trí (chỉ số) đầu tiên của phần tử.

## Ví dụ: count()

```
# Tạo một tuple
numbers = (1, 2, 3, 2, 4, 2, 5)

# Đếm số lần xuất hiện của phần tử 2
count_2 = numbers.count(2)
print(count_2)
```

3

## Ví dụ: index()

```
# Tao mot tuple
fruits = ("apple", "banana", "cherry", "apple", "banana")

# Tim vi tri dau tien cua phan tu "banana"
pos_banana = fruits.index("banana")
print(pos_banana)
```

1