

Bài 7: List trong Python

Mục lục:

1. Tổng quan
2. Cú pháp
3. Các phương thức của list
4. Nội dung tiếp theo

1. Tổng quan

Có 4 kiểu dữ liệu tập hợp trong Python, đó là

- List: tập hợp các phần tử có thứ tự và có thể thay đổi, cho phép xuất hiện lặp lại các phần tử.
- Tuple: tập hợp các phần tử có thứ tự và không thể thay đổi, cho phép xuất hiện lặp lại các phần tử.
- Set: tập hợp các phần tử không có thứ tự, không được đánh chỉ số, không cho phép xuất hiện lặp lại các phần tử.
- Dictionary: tập hợp các phần tử không có thứ tự, có thể thay đổi giá trị, được đánh chỉ số, không cho phép xuất hiện lặp lại các phần tử.

Nội dung này sẽ giới thiệu với các bạn về kiểu dữ liệu tập hợp list.

2. Cú pháp

Tên_biến = [các phần tử của list]

Trong đó:

- *Tên_biến*: người lập trình tự đặt
- *Các phần tử của list*: thường cùng kiểu, phân tách nhau bởi dấu phẩy.

Ví dụ:

```
# tập các string
names = ["Joe", "Marry", "Barry", "Tony"]
# tập các giá trị số nguyên
numbers = [1, 2, 3, 4, 5, 12, 200]
#tập các giá trị hỗn tạp
others = ["Harry", 10, 13.25, "Jenny"]
```

Ta có thể thay đổi giá trị một phần tử trong list:

```
names[0] = "Allen"
numbers[1] = 225
```

Ta cũng có thể thay đổi nhiều giá trị cùng lúc:

```
numbers[2:5] = [9,9,9]
print(numbers)
# result is [1, 225, 9, 9, 9, 12, 200]
```

Ta có cách khác để tạo ra một list: dùng hàm tạo: list()

```
fruits = list(("apple", "orange", "cherry"))
```

3. Các phương thức của list

Method	Description	Example
append()	Thêm phần tử vào cuối list	numbers.append(100)
clear()	Xóa bỏ tất cả các phần tử trong list	numbers.clear()
copy()	Trả về bản sao của list	otherFruit = fruits.copy()
count()	Trả về số lần xuất hiện của phần tử cho trước	numbers.count(9) # 3
extend()	Thêm các phần tử của một list vào cuối tập hợp hiện tại	fruits.extend(names)
Index()	Trả về chỉ số đầu tiên là vị trí xuất hiện của	numbers.index(9) # 2

	phần tử cho trong tham số	
insert(index, value)	Chèn phần tử vào vị trí index trong list	numbers.insert(1, 333) # chèn thêm phần tử 333 vào vị trí 1 của list
pop()	Xóa bỏ phần tử tại vị trí cho trước	numbers.pop(1) #xóa bỏ phần tử tại vị trí 1
remove()	Xóa bỏ phần tử đầu tiên trùng với giá trị cho trước	numbers.remove(9)
reverse()	Đảo thứ tự các phần tử trong list hiện tại	numbers.reverse()
sort(reverse=True False)	Sắp xếp list hiện tại theo thứ tự tăng hoặc giảm dần giá trị. Mặc định sắp xếp tăng. Nếu reverse=True thì sắp xếp giảm dần	numbers.sort()# mặc định. numbers.sort(reverse=True) #sắp xếp giảm dần.

4. Nội dung tiếp theo: **Câu lệnh điều khiển rẽ nhánh trong Python**