

SAU BÀI NÀY EM SẼ:

- Biết được kiểu dữ liệu danh sách (list), cách khởi tạo và truy cập từng phần tử của danh sách.
- Biết và thực hiện được cách duyệt các phần tử của danh sách bằng lệnh for.
- Thực hành được một số phương thức đơn giản trên dữ liệu danh sách.



Em đã được học những kiểu dữ liệu cơ bản của Python như số nguyên, số thực, xâu kí tự, kiểu dữ liệu logic. Tuy nhiên, khi em cần lưu một dãy các số hay một danh sách học sinh thì cần kiểu dữ liệu dạng *danh sách* (còn gọi là *dãy* hay *mảng*). Kiểu dữ liệu danh sách được dùng nhiều nhất trong Python là kiểu list.

Em hãy tìm một số dữ liệu kiểu danh sách thường gặp trên thực tế.

1. KIỂU DỮ LIỆU DANH SÁCH

Hoạt động 1 Khởi tạo và tìm hiểu dữ liệu kiểu danh sách

Khởi tạo dữ liệu danh sách như thế nào? Cách truy cập, thay đổi giá trị và xoá một phần tử trong danh sách như thế nào?



Ví dụ 1. Quan sát các lệnh sau để tìm hiểu kiểu dữ liệu danh sách.

```
>>> A = [1, 2, 3, 4, 5]
>>> B = [1.5, 2, "Python", "List", 0]
>>> A[0]
1
>>> B[2]
'Python'
```

Có thể truy cập từng phần tử của danh sách thông qua chỉ số. Chỉ số của list đánh số từ 0.

Kiểu dữ liệu danh sách trong Python được khởi tạo như sau:

```
<tên list> = [<v1>, <v2>, ..., <vn>]
```

Trong đó các giá trị $\langle v_k \rangle$ có thể có kiểu dữ liệu khác nhau (số nguyên, số thực, xâu kí tự,...). Ta có thể truy cập từng phần tử của danh sách thông qua chỉ số. Chỉ số bắt đầu từ 0.

Việc chỉ số hoá từng phần tử của danh sách cho thấy, có thể dùng danh sách để biểu diễn dữ liệu tương tự như kiểu mảng trong nhiều ngôn ngữ lập trình bậc cao khác. Tuy nhiên, *danh sách của Python có một khác biệt quan trọng, đó là nó có thể gồm các phần tử có kiểu dữ liệu khác nhau.*

Ví dụ 2. Quan sát các lệnh sau để biết cách thay đổi hoặc xoá phần tử của danh sách.

```
>>> A = [1, 2, 3, 4, 5]
>>> len(A)
5
```

Lệnh `len()` tính độ dài của danh sách.

```
>>> A[1] = "One"  ← Thay đổi giá trị từng phần tử bằng lệnh gán.
>>> A
[1, 'One', 3, 4, 5]
>>> del A[4]  ← Lệnh del để xoá một phần tử của danh sách.
>>> A
[1, 'One', 3, 4]
```

Sau khi đã khởi tạo danh sách, chúng ta có thể thay đổi các giá trị của từng phần tử bằng lệnh gán hoặc xoá phần tử bằng lệnh `del`.

Ví dụ 3. Quan sát các lệnh sau để biết cách tạo danh sách rỗng (có độ dài 0) và các phép toán ghép danh sách (phép +).

```
>>> a = []  ← Lệnh tạo một danh sách rỗng có độ dài bằng 0.
>>> len(a)
0
>>> [1,2] + [3,4,5,6]  ← Các phép ghép hai danh sách.
[1, 2, 3, 4, 5, 6]
```

- List là kiểu dữ liệu danh sách (dãy, mảng) trong Python. Tạo list bằng lệnh gán với các phần tử trong cặp dấu ngoặc `[]`. Các phần tử của danh sách có thể có các kiểu dữ liệu khác nhau. Truy cập hoặc thay đổi giá trị của từng phần tử thông qua chỉ số: `<danh sách>[<chỉ số>]`
- Chỉ số của danh sách bắt đầu từ 0 đến `len() - 1`, trong đó `len()` là lệnh tính độ dài danh sách.



1. Cho danh sách `A = [1, 0, "One", 9, 15, "Two", True, False]`. Hãy cho biết giá trị các phần tử:

a) `A[0]` b) `A[2]` c) `A[7]` d) `A[len(A)]`

2. Giả sử `A` là một danh sách các số, mỗi lệnh sau thực hiện gì?

a) `A = A + [10]` b) `del A[0]` c) `A = [100] + A` d) `A = A[1]*25`

2. DUYỆT CÁC PHẦN TỬ CỦA DANH SÁCH

Hoạt động 2 Dùng lệnh for để duyệt danh sách

Quan sát các lệnh sau để biết cách dùng lệnh `for` duyệt lần lượt các phần tử của một danh sách.



Ví dụ 1. Duyệt và in ra từng phần tử của danh sách.

```
>>> A = [1,2,3,4,5]
>>> for i in range(len(A)):
>>>     print(A[i], end = " ")
1 2 3 4 5
```

Biến `i` chạy trên vùng chỉ số từ 0 đến `len(A) - 1`.

Ví dụ 2. Duyệt và in một phần của danh sách.

```
>>> A = [3, 2, 1, 5, 6, 10, 7, 12, 18]
>>> for i in range(2,5):
>>>     print(A[i], end = " ")
1 5 6
```

$0, n \rightarrow C + n - 1$

Hai ví dụ trên cho thấy dùng lệnh **for** kết hợp với lệnh **range()** để duyệt từng phần tử của danh sách.

Có thể duyệt lần lượt các phần tử của danh sách bằng lệnh **for** kết hợp với vùng giá trị của lệnh **range()**.



1. Giải thích các lệnh ở mỗi câu sau thực hiện công việc gì?

a)

```
>>> S = 0
>>> for i in range(len(A)):
>>>     if A[i] > 0:
>>>         S = S + A[i]
>>> print(S)
```

b)

```
>>> C = 0
>>> for i in range(len(A)):
>>>     if A[i] > 0:
>>>         C = C + 1
>>> print(C)
```

2. Cho dãy các số nguyên A, viết chương trình in ra các số chẵn của A.

3. THÊM PHẦN TỬ VÀO DANH SÁCH



Python có những lệnh đặc biệt để thêm phần tử vào một danh sách. Các lệnh này được thiết kế riêng cho kiểu dữ liệu danh sách và còn được gọi là phương thức (method) của danh sách.

Hoạt động 3 Tìm hiểu lệnh thêm phần tử cho danh sách

Quan sát các lệnh sau đây để biết cách thêm phần tử vào một danh sách bằng phương thức **append()**.



Ví dụ. Thêm phần tử vào cuối danh sách.

```
>>> A = [1,2]
>>> A.append(10)
>>> A
[1, 2, 10]
```

Chú ý cách dùng phương thức **append()**: gõ tên biến danh sách, dấu ".", sau đó gõ **append**.

- Python có một số lệnh dành riêng (phương thức) cho dữ liệu kiểu danh sách. Cú pháp các lệnh đó như sau:
<danh sách>.<phương thức>
- Lệnh thêm phần tử vào cuối danh sách là <danh sách>.append().



1. Sau khi thêm một phần tử vào danh sách A bằng lệnh **append()** thì độ dài danh sách A thay đổi như thế nào?

2. Danh sách A sẽ như thế nào sau các lệnh sau?

```
>>> A = [2,4,10,1,0]
>>> A.append(100)
>>> del A[1]
```



THỰC HÀNH

Khởi tạo, nhập dữ liệu, thêm phần tử cho danh sách

Nhiệm vụ 1. Nhập số n từ bàn phím, sau đó nhập danh sách n tên các bạn lớp em và in ra danh sách các tên đó, mỗi tên trên một dòng.

Hướng dẫn. Chương trình yêu cầu nhập số tự nhiên n , sau đó nhập từng tên trong danh sách, dùng phương thức `append()` để đưa dần vào danh sách.

Chú ý: Vì vùng giá trị của lệnh `range(n)` bắt đầu từ 0 nên trong thông báo nhập cần viết là `str(i+1)` để bắt đầu từ 1.

Chương trình có thể như sau:

```
dsLop = []
n = int(input("Nhập số học sinh trong lớp: "))
for i in range(n):
    name = input("Nhập họ tên học sinh thứ " + str(i+1) + ": ")
    dsLop.append(name)
print("Danh sách học sinh đã nhập:")
for i in range(len(dsLop)):
    print(dsLop[i])
```

Nhiệm vụ 2. Nhập một dãy số từ bàn phím. Tính tổng, trung bình của dãy và in dãy số trên một hàng ngang.

Hướng dẫn. Tương tự nhiệm vụ 1, chỉ khác là nhập số nguyên nên dùng lệnh `int()` để chuyển đổi dữ liệu.

```
A = []
T = 0
n = int(input("Nhập số tự nhiên n: "))
for i in range(n):
    num = int(input("Nhập số thứ " + str(i+1) + ": "))
    A.append(num)
    T = T + num
print("Dãy số đã nhập:")
for i in range(n):
    print(A[i], end = " ")
print()
print("Tổng:", T)
print("Trung bình:", T/n)
```



LUYỆN TẬP

- Viết lệnh xoá phần tử cuối cùng của danh sách A bằng lệnh `del`.
- Có thể thêm một phần tử vào đầu danh sách được không? Nếu có thì nêu cách thực hiện.



VẬN DỤNG

Cho dãy số A . Viết chương trình tính giá trị và chỉ số của phần tử lớn nhất của A . Tương tự với bài toán tìm phần tử nhỏ nhất.