

SAU BÀI NÀY EM SẼ:

- Biết cách duyệt danh sách bằng toán tử **in**.
- Biết và thực hiện được một số phương thức thường dùng với danh sách.



Trong bài trước chúng ta đã biết cách dùng lệnh append để thêm phần tử vào cuối một danh sách. Vậy Python có lệnh nào dùng để:

- Xoá nhanh một danh sách?
- Chèn thêm phần tử vào đầu hay giữa danh sách?
- Kiểm tra một phần tử có nằm trong một danh sách không?

1. DUYỆT DANH SÁCH VỚI TOÁN TỬ IN

Hoạt động 1 Sử dụng toán tử in với danh sách

Quan sát ví dụ sau để biết cách dùng toán tử **in** để duyệt một danh sách.



Ví dụ 1. Dùng toán tử **in** để kiểm tra một giá trị có nằm trong danh sách hay không.

```
>>> A = [1,2,3,4,5]
>>> 2 in A ← Số nguyên 2 nằm trong dãy A, kết quả trả lại True.
True
>>> 10 in A ← Số 10 không nằm trong dãy A, kết quả trả lại False.
False
```

Câu lệnh dùng toán tử **in** để kiểm tra **<giá trị>** có trong **<danh sách>** không, nếu có thì trả lại True nếu không thì trả về False như sau:

<giá trị> in <danh sách>

Ví dụ 2. Sử dụng toán tử **in** để duyệt từng phần tử của danh sách.

```
>>> A = [10,11,12,13,14,15]
>>> for k in A:
    print(k, end = " ")
10 11 12 13 14 15
```

Khi thực hiện lệnh này, biến **k** sẽ lần lượt nhận các giá trị từ dãy A.

- Toán tử **in** dùng để kiểm tra một phần tử có nằm trong danh sách đã cho không. Kết quả trả lại True (Đúng) hoặc False (Sai).
- <giá trị> in <danh sách>**
- Có thể duyệt nhanh từng phần tử của danh sách bằng toán tử **in** và lệnh **for** mà không cần sử dụng lệnh **range()**.



- 1.** Giả sử $A = ["0", "1", "01", "10"]$. Các biểu thức sau trả về giá trị đúng hay sai?
- $1 \in A$
 - $"01" \in A$

- 2.** Hãy giải thích ý nghĩa từ khoá **in** trong câu lệnh sau:

for i in range(10):

2. MỘT SỐ LỆNH LÀM VIỆC VỚI DANH SÁCH

Hoạt động 2 Tìm hiểu một số lệnh làm việc với danh sách

Quan sát ví dụ sau để tìm hiểu một số lệnh làm việc với dữ liệu kiểu danh sách.



Ví dụ 1. Lệnh **clear()** xoá toàn bộ một danh sách.

```
>>> A = [1,2,3,4,5]
>>> A.clear() ← Sau khi thực hiện lệnh clear(), danh sách gốc trở thành rỗng.
>>> A
[]
```

Ví dụ 2. Lệnh **remove(value)** sẽ xoá phần tử đầu tiên của danh sách có giá trị **value**. Nếu không có phần tử nào như vậy thì sẽ báo lỗi.

```
>>> A = [1,2,3,4,5]
>>> A.remove(1) ← Lệnh remove() có chức năng xoá một phần tử có giá trị cho trước.
>>> A
[2, 3, 4, 5]
>>> A.remove(10) ← Lệnh báo lỗi nếu giá trị không có trong danh sách.
Traceback (most recent call last):
  File "<pyshell#25>", line 1, in <module>
    A.remove(10)
ValueError: list.remove(x): x not in list
```

Ví dụ 3. Lệnh **insert** có chức năng chèn phần tử vào danh sách tại chỉ số cho trước.

```
>>> A = [1,2,6,10]
>>> A.insert(2,5) ← Lệnh insert() có hai tham số cần nhập: vị trí cần chèn và giá trị được chèn. Lệnh insert(2,5) sẽ chèn số 5 tại chỉ số 2.
>>> A
[1, 2, 5, 6, 10]
```

Lệnh **insert(index,value)** sẽ chèn giá trị **value** vào danh sách tại vị trí **index** và đẩy các phần tử từ vị trí này sang phải.

Chú ý nếu **k** nằm ngoài phạm vi chỉ số của danh sách thì lệnh vẫn có tác dụng: nếu **k < 0** thì chèn vào đầu danh sách, nếu **k > len()** thì chèn vào cuối danh sách.

```
>>> A = []
>>> A.insert(-10,1) ← Lệnh này chèn số 1 vào đầu danh sách A.
```

```
>>> A.insert(100,2) ← Lệnh này chèn số 2 vào cuối danh sách A.  
>>> A  
[1, 2]
```

Một số lệnh làm việc với dữ liệu danh sách:

A.append(x)	Bổ sung phần tử x vào cuối danh sách A.
A.insert(k,x)	Chèn phần tử x vào vị trí k của danh sách A.
A.clear()	Xoá toàn bộ dữ liệu của danh sách A.
A.remove(x)	Xoá phần tử x từ danh sách A.

- 
- Khi nào thì lệnh A.append(1) và A.insert(0,1) có tác dụng giống nhau?
 - Danh sách A trước và sau lệnh insert() là [1,4,10,0] và [1,4,10,5,0]. Lệnh đã dùng là gì?

THỰC HÀNH

Các lệnh làm việc với dữ liệu kiểu danh sách

Nhiệm vụ 1. Nhập số n từ bàn phím, sau đó nhập danh sách n tên học sinh trong lớp và in ra danh sách học sinh này, mỗi tên học sinh trên một dòng. Yêu cầu danh sách được in ra theo thứ tự ngược lại với thứ tự đã nhập.

Hướng dẫn. Chương trình sẽ yêu cầu nhập số tự nhiên n, sau đó sẽ lần lượt yêu cầu nhập n tên học sinh. Tuy nhiên do yêu cầu in danh sách học sinh theo thứ tự ngược lại so với thứ tự nhập nên cần dùng lệnh insert() để chèn tên học sinh được nhập vào đầu danh sách. 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.insert(0, name)  
print("Danh sách học sinh đã nhập:")  
for name in dsLop:  
    print(name)
```

Nhiệm vụ 2. Cho trước dãy số A. Viết chương trình xoá đi các phần tử có giá trị nhỏ hơn 0 từ A.

Hướng dẫn. Duyệt từng phần tử của dãy số A, kiểm tra nếu phần tử này nhỏ hơn 0 thì xoá đi.

Chương trình có thể như sau:

```
A = [0,1,-3,-10,5,9,-20,55]  
i = 0  
while i < len(A):  
    if A[i] < 0:
```

```

A.remove(A[i])
else:
    i = i + 1
print(A)

```

Nhiệm vụ 3. Cho trước dãy số A. Viết chương trình tìm và chỉ ra vị trí đầu tiên của dãy số A mà ba số hạng liên tiếp có giá trị là 1, 2, 3. Nếu tìm thấy thì thông báo vị trí tìm thấy, nếu không thì thông báo "Không tìm thấy mẫu".

Hướng dẫn. Soạn thảo chương trình sau rồi thực hiện và kiểm tra tính đúng đắn của chương trình.

```

A = [0,4,0,1,2,3,8,9,0,1,2,3,17,-16,0,1,2]
p = [1,2,3]
pkq = -1
i = 0
while i < len(A) - 3 and pkq == -1:
    if A[i] == p[0] and A[i+1] == p[1] and A[i+2] == p[2]:
        pkq = i
    else:
        i = i + 1
if pkq >= 0:
    print("Tìm thấy mẫu ",p," tại vị trí", pkq)
else:
    print("Không tìm thấy mẫu",p)

```



LUYỆN TẬP

- Cho dãy số [1,2,2,3,4,5,5]. Viết lệnh thực hiện:
 - Chèn số 1 vào ngay sau giá trị 1 của dãy.
 - Chèn số 3 và số 4 vào danh sách để dãy có số 3 và số 4 liền nhau hai lần.
- Cho trước dãy số A. Viết chương trình thực hiện công việc sau:
 - Xoá đi một phần tử ở chính giữa dãy nếu số phần tử của dãy là số lẻ.
 - Xoá đi hai phần tử ở chính giữa của dãy nếu số phần tử của dãy là số chẵn.



VẬN DỤNG

- Viết chương trình nhập n từ bàn phím, tạo và in ra màn hình dãy số A bao gồm n số tự nhiên chẵn đầu tiên.
- Dãy số Fibonacci được xác định như sau:

$$F_0 = 0$$

$$F_1 = 1$$

$$F_n = F_{n-1} + F_{n-2} \text{ (với } n \geq 2\text{).}$$

Viết chương trình nhập n từ bàn phím, tạo và in ra màn hình dãy số A bao gồm n số hạng đầu của dãy Fibonacci.