

BÀI 11: KIỂU DANH SÁCH LIST

I. Kiểu dữ liệu danh sách

1) Khái niệm

- Danh sách là một dãy hữu hạn các phần tử (có thể không cùng kiểu)
- Danh sách được đặt tên và các phần tử đều được đánh chỉ số
- Các phần tử của List được phân cách nhau ra bởi dấu phẩy (,).
- Được giới hạn bởi cặp ngoặc [], tất cả những gì nằm trong đó là những phần tử của List.

2) Cách khởi tạo

Cách 1: Sử dụng cặp dấu ngoặc[] đặt giá trị bên trong

Cú pháp:

```
[<giá trị thứ nhất>, <giá trị thứ hai>, ..., <giá trị thứ n - 1>, <giá trị thứ n>]
```

Ví dụ:

- `lst = ["Toán học", "Hình học", "Tin học", 8.5, 7.9, 6.3]`

- `empty_list = []` # khởi tạo list rỗng

Cách 2: Sử dụng List Comprehension (bao quát)

Cú pháp:

```
[Comprehension]
```

Ví dụ:

- ❖ `ds = [kteam for kteam in range(3)]`

➔ `ds` sẽ có kết quả là: `[0, 1, 2]`

- ❖ `lst = [[n, n * 1, n * 2] for n in range(1, 4)]`

➔ `lst` sẽ có kết quả là: `[[1, 1, 2], [2, 2, 4], [3, 3, 6]]`

➔ `lst` gồm 3 list con, mỗi list có 3 phần tử theo thứ tự `[n, n * 1, n * 2]`

***Nhập dữ liệu cho List các phần tử cùng kiểu

Cú pháp:

```
<tênds> = [<kiểu>(<biến chạy>) for <biến chạy> in input().split()]
```

Trong đó:

- `<tênds>`, `<biến chạy>`: được đặt theo quy tắc đặt tên biến
- `<kiểu>`: là kiểu `int` hoặc `float` tùy vào danh sách nhập vào là số nguyên hay thực

Ví dụ 1: * nhập một dãy số nguyên từ bàn phím

```
lst= [int(i) for i in input('nhập vào dãy số nguyên:').split()]
```

Kết quả: `nhập vào dãy số nguyên:10 15 20 10 20 30`

* In dãy số nguyên vừa nhập: `for k in lst: print(k)`

Kết quả:

```
nhập vào dãy số nguyên:10 15 20 10 20 30
10
15
20
10
20
30
```

Ví dụ 2: nhập một dãy số thực từ bàn phím và in ra dãy số vừa nhập

```
lst= [float(i) for i in input('nhập vào dãy số thực:').split()]
for k in lst:
    print(k)
```

```
nhập vào dãy số thực: 5.6 8.7 9.5 6.5
5.6
8.7
9.5
6.5
```

Kết quả:

Ví dụ 3: list là một danh sách các số từ 1 đến 30, mỗi số cách 3 đơn vị

```
lst= [i for i in range(1,30,3)]
for k in lst:
    print(k)
```

Kết quả thực thi chương trình

```
1
4
7
10
13
16
19
22
25
28
```

Cách 3: Nhập vào dãy số nguyên có n phần tử

```
n=int(input('nhập số nguyên n:')) # nhập số nguyên n từ bàn phím
ds=[] # khởi tạo danh sách rỗng
for i in range(0,n):
    ds.append(int(input('số nguyên =')))
```

Nhập các phần tử của danh

3) Đưa danh sách ra màn hình

Cú pháp:

```
for <biến chạy> in <tên ds>:
    print(<biến chạy>)
Hoặc:
print(<tên ds>)
```

ví dụ: `lst = ["Toán học", "Hình học", "Tin học", 8.5, 7.9, 6.3]`
`for i in lst: print(i)`

II. Các thao tác trên List trong Python

1) Các phép toán

Phép toán	Ý nghĩa	Ví dụ
+	Ghép nhiều danh sách thành một	[“Toán học”, “Tin học”]+[“Hóa học”] →[“Toán học”, “Tin học”, “Hóa học”]
*	Tạo ra một danh sách lặp đi lặp lại xâu gốc với số lần nhân	[0, 1]*3⇒[0, 1, 0, 1, 0, 1]
in	Cho biết một phần tử có xuất hiện trong list hay không, nếu có nhận giá trị True, ngược lại nhận giá trị False	“Vật lý” in [“Toán học”, “Tin học”] →False
Hàm len()	Trả về số phần tử trong danh sách	ds = [“-1”, 0, 1, 2, “...”] →len(ds) = 5

2) Đánh chỉ số và các thao tác với chỉ số trong List

a) Đánh chỉ số các phần tử trong list: bắt đầu từ 0 đến độ dài list – 1

“-1”	0	1.5	2	“...”	123	23.4	“Phong”
0	1	2	3	4	5	6	7

b) Tham chiếu tới phần tử của list bằng chỉ số đặt trong cặp ngoặc [và]

Cú pháp:

<tên list>[chỉ số]

Ví dụ:

ds = [“-1”, 0, 1.5, 2, “...”, 123, 23.4, “Tâm”]

ds[2] = 1.5, ds[5]=123

c) Sao chép danh sách: để tạo ra một list mới là đoạn con của list gốc từ chỉ số bắt đầu đến chỉ số dừng – 1

Cú pháp:

<tên_list>[vị trí bắt đầu:vị trí dừng]

Ví dụ:

ds = [“-1”, 0, 1.5, 2, “...”, 123, 23.4, “Tâm”]

print(ds[1:3])

→ kết quả: [0, 1.5]

d) Thay đổi giá trị phần tử trong danh sách

Cú pháp:

<tên_list>[chỉ số]= <giá trị thay thế>

Ví dụ:

ds = [“-1”, 0, 1.5, 2, “...”, 123, 23.4, “Tâm”]

ds[0] = “---”

print(ds)

→ kết quả: ds = [“---”, 0, 1.5, 2, “...”, 123, 23.4, “Tâm”]

e) Chèn một phần tử vào trong danh sách

Cú pháp:

<tên list>.insert(i, x)

Ý nghĩa: **Thêm phần tử x vào vị trí i** trong danh sách

Ví dụ:

```
ds = ["-1", 0, 1.5, 2, "...", 123, 23.4, "Tâm"]  
vị trí  0  1  2  3  4  
ds.insert(4,52)
```

→ kết quả: ds=['-1', 0, 1.5, 2, **52**, '...', 123, 23.4, 'Tâm']

f) Sắp xếp danh sách

Cú pháp:

```
<tên list>.sort(key=None, reverse=False)
```

Ý nghĩa: Sắp xếp các phần tử bằng cách so sánh trực tiếp

Chú ý:

- Từ khóa **reverse** chỉ có thể cho 2 giá trị, một là **True**, hai là **False**. Nếu là False, các phần tử được sắp xếp từ bé đến lớn, còn ngược lại là từ lớn đến bé
- Các phần tử trong danh sách phải so sánh được với nhau

Ví dụ 1: ds = ["Tin học", "123", "Toán học", "345", "23.5"]

```
ds.sort()  
print(ds)
```

→ ds có 5 phần tử thuộc kiểu xâu kí tự → so sánh được

Kết quả: ['123', '23.5', '345', 'Tin học', 'Toán học']

Ví dụ 2: ds = [123, 345, 23.5, 5, 8, 12]

```
ds.sort()  
print(ds)
```

Kết quả: [5, 8, 12, 23.5, 123, 345]

g) Thêm một phần tử vào cuối danh sách

Cú pháp:

```
<tên list>.append(x)
```

Ý nghĩa: **Thêm phần tử x vào cuối danh sách**

Ví dụ:

```
ds = ["-1", 0, 1.5, 2, "...", 123, 23.4, "Tâm"]  
ds.append(98)
```

→ **Kết quả:** ["-1", 0, 1.5, 2, "...", 123, 23.4, "Tâm", **98**]

h) Tính tổng các phần tử thỏa điều kiện của danh sách

tong=0

khởi tạo tổng =0

```
for i in range(0,n):
```

```
    if dk:
```

```
        tong=tong+ ds[i]
```

Tính tổng các phần tử thỏa dk của danh sách

```
print(' tong la: ',tong)
```

i) Đếm các phần tử thỏa điều kiện của danh sách

dem=0

khởi tạo dem =0

for i in range(0,n):

if dk:

dem=dem+ 1



đếm các phần tử thỏa dk của danh sách

print(' Các phần tử thỏa dk là: ',dem)