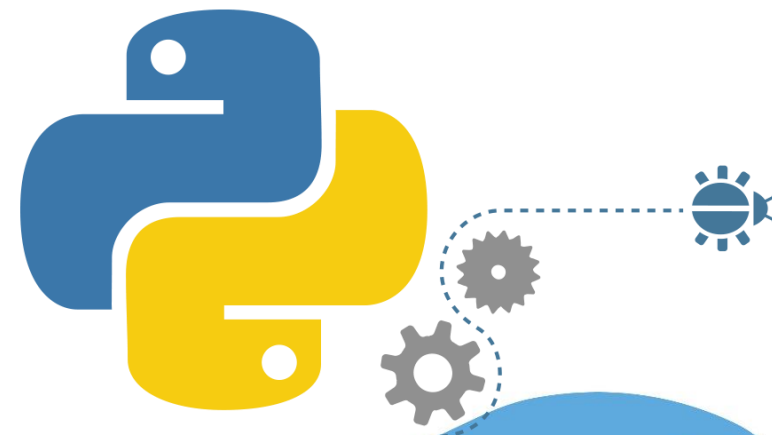


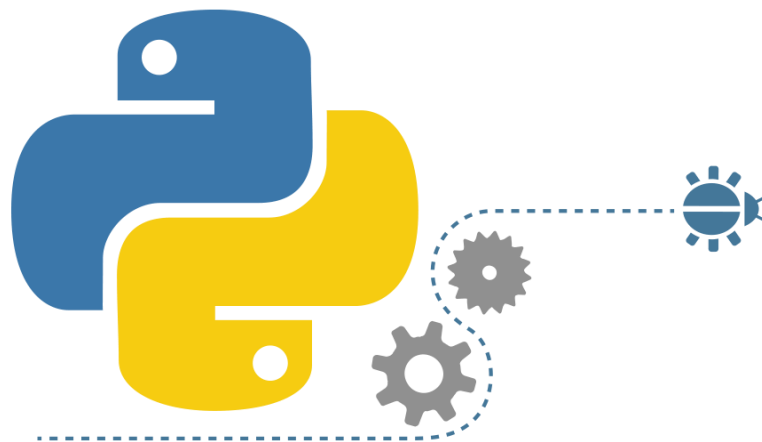


Môn Học **KỸ THUẬT LẬP TRÌNH** **VỚI PYTHON**

GV: Ths. Trần Duy Thanh
thanhtd@uel.edu.vn



XỬ LÝ DANH SÁCH





Mục tiêu bài học

- Hiểu được cơ chế vận hành của List
- Khai báo và sử dụng được List
- Thực hiện được các hàm: duyệt, gán, chèn, nối đuôi, xóa, đảo ngược, sắp xếp, slicing
- Hiểu và thực hiện được List đa chiều



Nội dung bài học

- 6.1. Cách khai báo và sử dụng List
- 6.2. Cách duyệt List
- 6.3. Gán giá trị cho các phần tử trong List
- 6.4. Phương thức insert
- 6.5. Phương thức append
- 6.6. Phương thức remove
- 6.7. Phương thức reverse
- 6.8. Phương thức sort
- 6.9. Slicing
- 6.10. List đa chiều

6.1. Cách khai báo và sử dụng List

List trong Python là một đối tượng dùng để lưu tập các đối tượng khác. List có thể chứa bất kỳ kiểu dữ liệu nào. Tuy nhiên ta nên thống nhất một kiểu để dễ dàng trong quá trình xử lý.

-Khai báo list rỗng:

```
lst = []
```

-Khai báo list có các giá trị:

```
lst = [2, -3, 0, 4, -1]
```

-Khai báo list có 10 phần tử với giá trị mặc định là 0:

```
lst=[0]*10
```

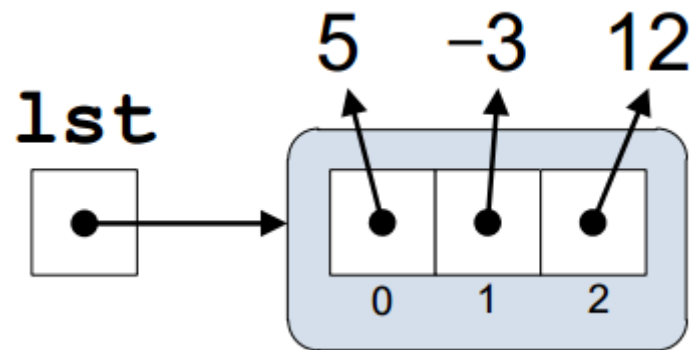
-Khai báo list có 10 phần tử với giá trị mặc định là 0.5:

```
lst=[0.5]*10
```

6.1. Cách khai báo và sử dụng List

Ví dụ:

```
lst=[5,-3,12]
```



```
print(lst[0])
```

```
print(lst)
```

```
print(len(lst))
```



6.2. Cách duyệt List

Với Python, ta có thể duyệt list theo nhiều cách. Bài học hướng dẫn 2 cách:

- Duyệt theo collection
- Duyệt theo chỉ số index

```
print("***35")
lst=[5,7,2,9,6,3,10,17,16]
for x in lst:
    print(x,end='\t')
print()
print("***35")
for i in range(len(lst)):
    x=lst[i]
    print(x,end='\t')
print()
print("***35")
for i in range(len(lst)-1,-1,-1):
    x=lst[i]
    print(x,end='\t')
```

5 7 2 9 6 3 10 17 16

5 7 2 9 6 3 10 17 16

16 17 10 3 6 9 2 7 5

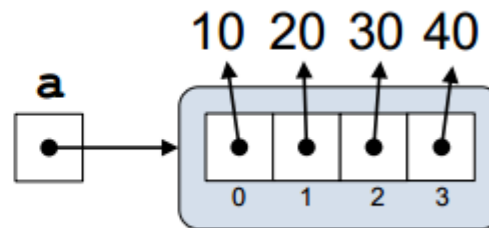


6.3. Gán giá trị cho các phần tử trong List

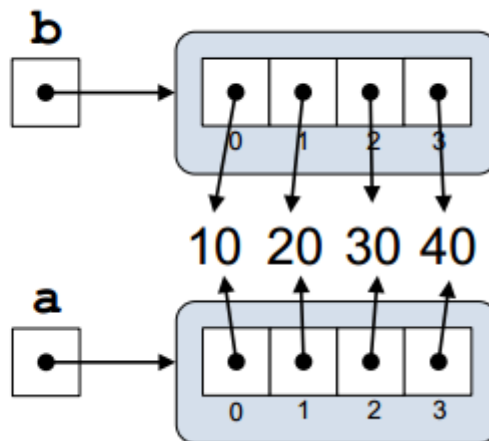
`lst = [2, 4, 6, 8]` → `lst` tham chiếu tới List

`lst[2]` → tham chiếu tới phần tử thứ 2 (giá trị = 6)

`a = [10, 20, 30, 40]`



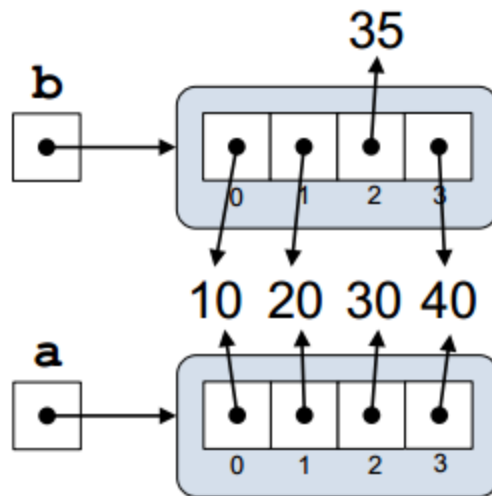
`b = [10, 20, 30, 40]`





6.3. Gán giá trị cho các phần tử trong List

`b[2] = 35`

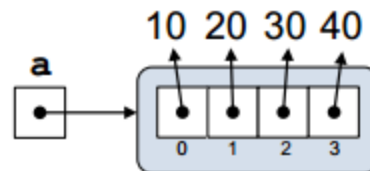




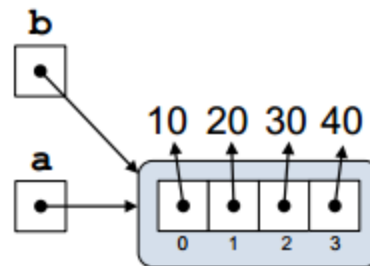
6.3. Gán giá trị cho các phần tử trong List

Ví dụ gán tham chiếu:

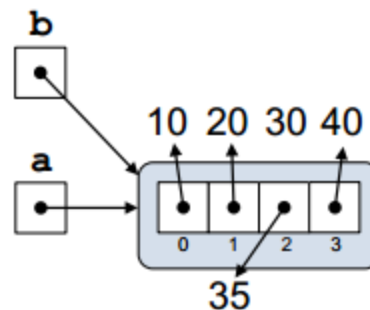
`a = [10, 20, 30, 40]`



`b = a`



`b[2] = 35`



6.4. Phương thức insert

Python hỗ trợ hàm insert trong list. Tự động chèn vào vị trí thích hợp insert(vị trí muốn chèn, giá trị muốn chèn):

```
lst=[1,2,3]
print(lst)
lst.insert(2,9)
print(lst)
lst.insert(0,17)
print(lst)
```



```
[1, 2, 3]
[1, 2, 9, 3]
[17, 1, 2, 9, 3]
```

6.5. Phương thức append

Python hỗ trợ hàm append trong list. Chèn giá trị mới vào cuối danh sách

```
lst=[1,2,3]  
lst.append(-113)  
print(lst)
```



[1, 2, 3, -113]

6.6. Phương thức remove

Python hỗ trợ hàm remove, xóa phần tử đầu tiên trong List

```
lst=[2,0,1,8,0]  
lst.remove(0)  
print(lst)
```



[2, 1, 8, 0]

```
del lst[0]
```

6.7. Phương thức reverse

Python hỗ trợ hàm reverse để đảo danh sách:

```
lst=[8,1,0,2]  
print(lst)  
lst.reverse()  
print(lst)
```



```
[8, 1, 0, 2]  
[2, 0, 1, 8]
```



6.8. Phương thức sort

Python hỗ trợ hàm sort để sắp xếp list

```
lst=[8,1,0,2]  
print(lst)  
lst.sort()  
print(lst)
```



```
[8, 1, 0, 2]  
[0, 1, 2, 8]
```

Hoặc

```
lst=sorted(lst)
```



6.9. Slicing

Slicing dùng để trích lọc list

list [*begin* : *end* : *step*]

List: là danh sách

Begin: Vị trí bắt đầu cắt

End: Vị trí cuối cùng cắt

Step: bước nhảy



6.9. Slicing

Ví dụ:

```
lst = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120]
print(lst) # [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120]
print(lst[0:3]) # [10, 20, 30]
print(lst[4:8]) # [50, 60, 70, 80]
print(lst[2:5]) # [30, 40, 50]
print(lst[-5:-3]) # [80, 90]
print(lst[:3]) # [10, 20, 30]
print(lst[4:]) # [50, 60, 70, 80, 90, 100, 110, 120]
print(lst[:]) # [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120]
print(lst[-100:3]) # [10, 20, 30]
print(lst[4:100]) # [50, 60, 70, 80, 90, 100, 110, 120]
print(lst[2:-2:2]) # [30, 50, 70, 90]
print(lst[::2]) # [10, 30, 50, 70, 90, 110]
print(lst[::-1]) # [120, 110, 100, 90, 80, 70, 60, 50, 40, 30, 20, 10]
```



6.10. List đa chiều

Python hỗ trợ List đa chiều

```
matrix = [  
    [100, 14, 8, 22, 71],  
    [ 0, 243, 68, 1, 30],  
    [ 90, 21, 7, 67, 112],  
    [115, 200, 70, 150, 8]  
]  
print(matrix)
```



6.10. List đa chiều

Duyệt list đa chiều:

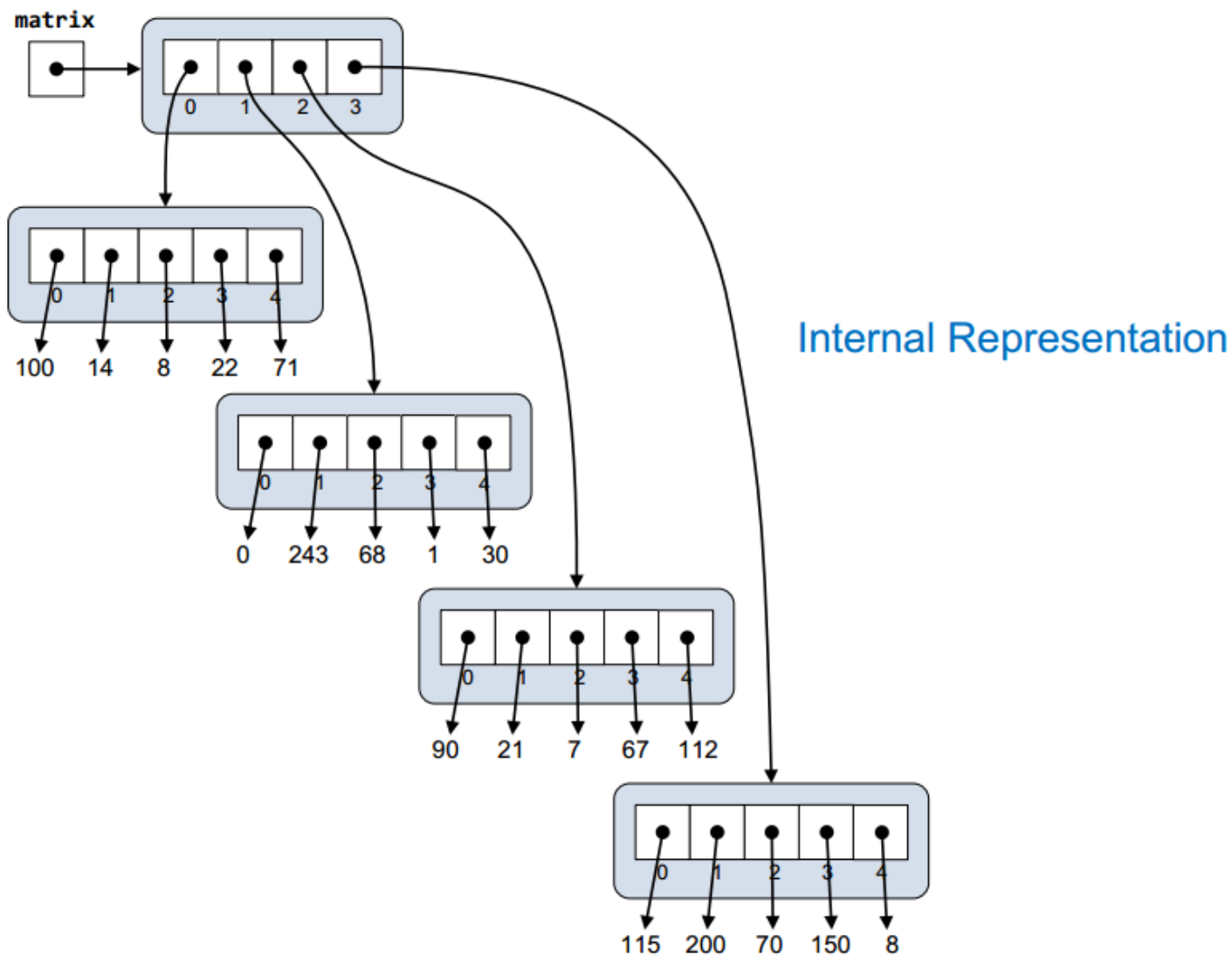
```
matrix = [  
    [100, 14, 8, 22, 71],  
    [ 0, 243, 68, 1, 30],  
    [ 90, 21, 7, 67, 112],  
    [115, 200, 70, 150, 8]  
]  
print(matrix)  
  
for row in matrix: # duyệt từng dòng  
    for elem in row: # Lấy từng phần tử trên mỗi dòng  
        print('{:>4}'.format(elem), end='')  
    print()
```

6.10. List đa chiều

Duyệt list đa chiều:

matrix

	0	1	2	3	4
0	100	14	8	22	71
1	0	243	68	1	30
2	90	21	7	67	112
3	115	200	70	150	8

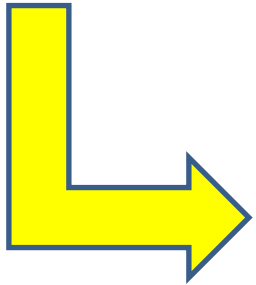




6.10. List đa chiều

Khởi tạo list đa chiều có row dòng column cột:

```
row=5  
column=3  
lst=[ [0]*column]*row  
print(lst)
```



```
[[0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0]]
```



6.10. List đa chiều

```
arr2D = []
rowsize=5
columnsize=3
for i in range(rowsize):
    onerow = []
    for j in range(columnsize):
        onerow.append(randrange(-100,100))
    arr2D.append(onerow)

for i in range(len(arr2D)):
    for j in range(len(arr2D[i])):
        print(arr2D[i][j],end='\t')
    print()
for row in arr2D:
    for column in row:
        print(column,end='\t')
    print()
```



THANK YOU

028 37244555 www.uel.edu.vn

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KINH TẾ - LUẬT

Số 669, đường Quốc lộ 1, khu phố 3, phường Linh Xuân,
quận Thủ Đức, Thành phố Hồ Chí Minh.