

프로그래밍특론

리스트

국민대학교
이정미 교수

kook906@kookmin.ac.kr

리스트

리스트란?

리스트의 인덱싱과 슬라이싱

리스트의 연산, 수정 및 삭제

리스트의 함수들

1. 리스트(list)란?

리스트(list)

- 정의

- 파이썬 자료형 중의 하나로, 항목의 나열(다수의 자료형들도 입력 가능)
- 시퀀스(sequence) 데이터로 순서가 있음
- 대괄호([])를 사용하며 각각의 항목들은 콤마(,)로 구분

- 리스트 만들기

- 항목들은 [] 안에 써 주고, 각각의 항목은 콤마로 구분
- 리스트명 = [항목1, 항목2, 항목3,...]

2. 리스트의 인덱싱과 슬라이싱

리스트의 인덱싱

- 인덱싱

- 순서가 있는 자료형으로 인덱싱 가능 (문자열과 유사), 인덱싱은 0부터 시작

- 예제

- 리스트명이 a인 리스트: 4개의 항목으로 구성
- $a[0]$ 은 a 리스트의 첫번째 항목의 값으로, **1**을 가리킴
- $a[1] + a[2]$ 는? a 리스트의 두번째와 세번째 항목값을 가져와 더해줌으로 **5**가 됨
- 리스트의 안의 리스트 항목값 인덱싱: $a[3][1]$ 은? **'b'**
- 리스트의 마지막 항목 인덱싱: $a[-1]$ 은? **['a', 'b', 'c']**

리스트(list)

| | | |
|------|------|------|
| 10 | 20 | 30 |
| a[0] | a[1] | a[2] |

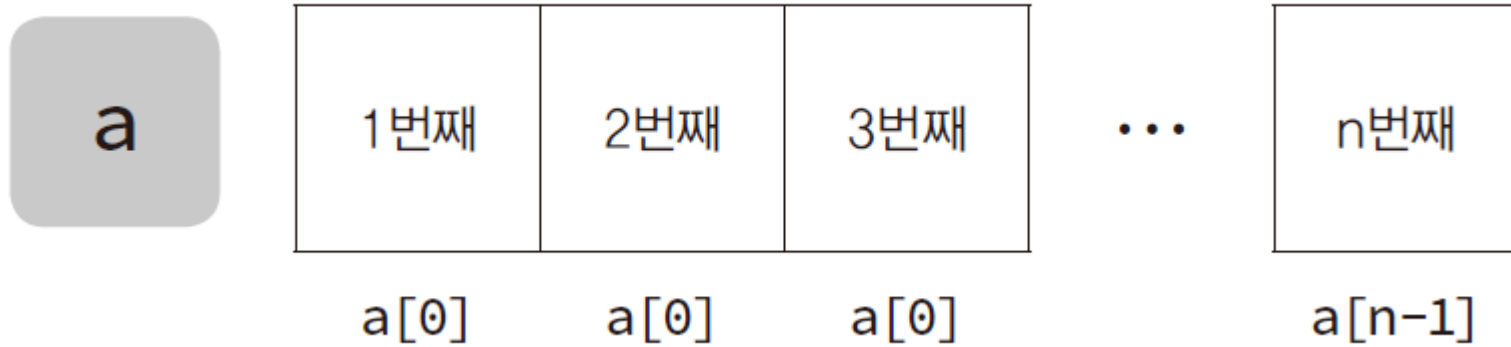
```
>>> print(a[0])
```

```
10
```

```
>>> print(a[3])
```

```
IndexError: list index out of range
```


리스트(list)



리스트(list)

```
01 a = [1, 2, 3, 4, 5]
02 print( a[0] )
03 print( a[1] )
04 print( a[2] )
05 print( a[3] )
06 print( a[4] )
```

```
01 a = [1, 2, 3, 4, 5]
02 for item in a :
03     print(item)
```

리스트의 슬라이싱 - 실습

실습

```
a=[10,20,30,40,50]  
print(a[0])  
print(a[0:1])  
print(a[0:2])  
print(a[0:3])  
print(a[0:4])  
print(a[0:5])  
print(a[0:6])  
print(a[-1])
```

실행 결과

```
10  
[10]  
[10, 20]  
[10, 20, 30]  
[10, 20, 30, 40]  
[10, 20, 30, 40, 50]  
[10, 20, 30, 40, 50]  
50
```

3. 리스트의 연산, 수정 및 삭제

리스트의 연산

- 덧셈

- +기호를 사용하여 덧셈
- 항목들간의 덧셈이 아니라 리스트의 항목들 합치기

```
>>> a = [1, 2, 3]
>>> b = [4, 5, 6, 7]
>>> a+b
[1, 2, 3, 4, 5, 6, 7]
```

- 반복

- *기호를 사용하여 리스트 항목 반복하기

```
>>> a = [1, 2, 3]
>>> a*3
[1, 2, 3, 1, 2, 3, 1, 2, 3]
```

- 길이 계산

- len() 함수 사용하여 항목의 개수 구하기

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

리스트의 항목 수정과 삭제

- 수정

- 리스트의 항목을 인덱싱하여 새로운 값 할당

```
>>> a = [1, 2, 3]
>>> a[1] = 10
>>> a
[1, 10, 3]
```

- 삭제

- del 함수 사용하여 리스트 항목을 인덱싱하여 삭제

```
>>> a = [1, 2, 3]
>>> del a[-1]
>>> a
[1, 2]
```

4. 리스트의 함수들

리스트에 데이터 추가 관련 함수들

- append, insert, extend 함수 사용법
- append: 요소 하나를 추가
- insert: 특정 인덱스에 요소 추가
- extend: 리스트를 연결하여 확장

리스트에 데이터 추가 관련 함수들

실습

```
c=[10,20,30]
c.append(40)
print(c)
c.append(50)
print(c)

c.insert(1,10)  # c[1]자리에 10넣기
print(c)
d=[100,200,300]
c.extend(d)
print(c)
```

실행 결과

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

[10, 20, 30, 40, 50]

[10, 10, 20, 30, 40, 50]

[10, 10, 20, 30, 40, 50, 100, 200, 300]
```

리스트의 함수들

>>> celebList.

append
clear
copy
count
extend
index
insert
pop
remove
reverse

.까지 누르고
기다리면 목록이 짤

• 리스트의 함수들

- 리스트 이름 다음에 마침표(.)를 찍고 함수를 써줌

- `list_name.append(항목)`
 - 맨 뒤에 한 항목 추가
- `list_name.clear()`
 - 모든 항목 제거 (빈 리스트가 남음)
- `list_name.copy()`
 - 리스트를 새로운 리스트 복사, `alist = blist.copy()`
- `list_name.count(항목)`
 - 리스트 내에 해당 항목이 몇 개 있는지 알려줌
- `list_name.extend(list)`
 - List_name 리스트에 인자로 주어진 list를 연결
- `list_name.index(항목, start, stop)`
 - List에서 항목이 처음 발견되는 위치 - 없으면 에러 (start, stop은 찾을 위치 시작과 끝, 생략가능)
- `list_name.insert(index, 항목)`
 - Index 위치에 항목을 끼워 넣음
- `list_name.pop(index)`
 - Index 위치의 항목을 알려주고, 그 항목은 리스트에서 제거
- `list_name.remove(항목)`
 - 리스트에서 항목을 제거, 없으면 에러
- `list_name.reverse()`
 - 리스트 내의 항목의 순서를 뒤집음
- `list_name.sort()`
 - 리스트 항목들을 오름차순으로 정렬 (숫자 라면 크기 순, 문자열은 가나다 순)
- `list_name.sort(reverse=True)`
 - 리스트 항목들을 내림차순으로 정렬

감사합니다.