

# list methods 이해

7주차\_02\_02

한 동 대 학 교  
김경미 교수

# 학습목표

2

- ▶ 리스트에서 다양한 메소드 활용하기

# 리스트, Methods

3

<code>.append()</code>	리스트 내에 새로운 아이템 한 개를 추가하여, 마지막에 위치한다
<code>.insert()</code>	리스트 내에 index 번호와 아이템 내용을 추가한다
<code>.extend()</code>	다른 이름의 리스트, 아이템 모두를 추가한다
<code>.sort()</code>	리스트의 아이템들을 순서대로 정렬, 순서는 ascii code 순이다
<code>.pop()</code>	리스트 내에 존재하는 아이템의 index 번호를 입력 받아 삭제한다
<code>.remove()</code>	리스트 내에 존재하는 아이템을 삭제, 동일한 아이템 있으면 처음 아이템만 삭제

# .append() 예제 I

4

```
# method append
```

```
>>> t1 = ['x', 'y', 'z']
```

```
>>> t1.append('a')
```

```
>>> t1
```

```
['x', 'y', 'z', 'a']
```

```
>>> t1.append('e')
```

```
>>> t1
```

```
['x', 'y', 'z', 'a', 'e']
```

## .append() 예제 2

5

```
# method append
```

```
>>> s1 = ['h', 'e', 'l', 'l', 'o']
```

```
>>> s1.append('!')
```

```
>>> s1
```

```
['h', 'e', 'l', 'l', 'o', '!']
```

```
>>> s1.append('~')
```

```
>>> s1
```

```
['h', 'e', 'l', 'l', 'o', '!', '~']
```

# .insert() 예제 I

```
# method insert
```

```
>>> t1 = ['x', 'y', 'z']
```

```
>>> t1
```

```
['x', 'y', 'z']
```

```
>>> t1.insert(1, 'a')
```

```
>>> t1
```

```
['x', 'a', 'y', 'z']
```

```
>>> t1.insert(1, 'e')
```

```
>>> t1
```

```
['x', 'e', 'a', 'y', 'z']
```

## .insert() 예제2

7

```
# method insert
```

```
>>> s1 = ['h', 'e', 'l', 'l', 'o']
```

```
>>> s1.insert(0, '>')
```

```
>>> s1
```

```
['>', 'h', 'e', 'l', 'l', 'o', '!']
```

```
>>> s1.insert(3, '!')
```

```
>>> s1
```

```
['>', 'h', 'e', 'l', 'l', 'l', 'o', '!']
```

# .extend() 예제 I

```
# method extend
```

```
>>> t1 = ['x', 'y', 'z']
```

```
>>> t2 = ['d', 'e']
```

```
>>> t1.extend(t2)
```

```
>>> t1
```

```
['x', 'y', 'z', 'd', 'e']
```

```
>>> t2.extend(t1)
```

```
>>> t2
```

```
['d', 'e', 'x', 'y', 'z', 'd', 'e']
```



## .extend() 예제 2

9

```
# method extend
```

```
>>> s1 = ['h', 'e', 'l']
```

```
>>> s2 = ['l', 'o']
```

```
>>> s1.extend(s2)
```

```
>>> s1
```

```
['h', 'e', 'l', 'l', 'o']
```

```
>>> s2.extend(s1)
```

```
>>> s2
```

```
['l', 'o', 'h', 'e', 'l', 'l', 'o']
```

# .sort() 예제 I

10

#method sort

```
>>> t = ['d', 'c', 'e', 'b', 'a']  
>>> t.sort()  
>>> t  
['a', 'b', 'c', 'd', 'e']
```

```
>>> s = ['h', 'e', 'l', 'l', 'o']  
>>> s.sort()  
>>> s  
['e', 'h', 'l', 'l', 'o']
```

# .sort()와 sorted

11

- ▶ .sort()

- ▶ list를 그 자리에서 정렬, 인덱스를 변경하고 None을 반환

- ▶ sorted

- ▶ 새로운 정렬된 목록을 반환하며, 원래 목록은 영향을 받지 않음

# .sort()와 sorted 차이

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

```
>>> b = [5, 2, 3, 1, 4]
```

```
>>> print(a.sort())
```

```
None
```

```
>>> print(a)
```

```
[1, 2, 3, 4, 5]
```

```
>>> print(sorted(b))
```

```
[1, 2, 3, 4, 5]
```

```
>>> print(b)
```

```
[5, 2, 3, 1, 4]
```

# .sort() 오름차순, 내림차순

13

- ▶ .sort() : 기본값이 오름차순,
- ▶ reverse : reverse 값을 True로 할 경우, 내림차순 정렬
  - ▶ sort() 메소드에 옵션으로 추가 가능

# .sort() 오름차순, 내림차순

```
>>> fruits=['apple','banana','lemon','grape','melon']
```

```
>>> newFruits = sorted(fruits, reverse=True)
```

```
>>> print(newFruits)
```

```
['melon', 'lemon', 'grape', 'banana', 'apple']
```

```
>>> print(fruits)
```

```
['apple', 'banana', 'lemon', 'grape', 'melon']
```

```
>>> fruits.sort(reverse=True)
```

```
>>> print(fruits)
```

```
['melon', 'lemon', 'grape', 'banana', 'apple']
```

# ·pop() 예제 I

15

```
#method pop(index)
```

```
>>> t = ['a', 'b', 'c', 'd', 'e']
```

```
>>> x = t.pop(0) + t.pop(1)
```

```
>>> t
```

```
['b', 'd', 'e']
```

```
>>> t.append('a')
```

```
>>> t
```

```
['b', 'd', 'e', 'a']
```

## ·pop() 예제2

16

```
#method pop(index)
```

```
>>> s = ['h', 'a', 'n', 'd', 'o', 'n', 'g']
```

```
>>> x = s.pop(2) + s.pop(4)
```

```
>>> s
```

```
['h', 'a', 'd', 'o', 'g']
```

```
>>> s.insert(0, '~')
```

```
>>> s
```

```
['~', 'h', 'a', 'd', 'o', 'g']
```



# .remove() 예제 I

17

```
# method remove(value)
```

```
>>> t = ['a', 'b', 'c']
```

```
>>> t.remove('b')
```

```
>>> t
```

```
['a', 'c']
```

## .remove() 예제2

18

```
# method remove(value)
```

```
>>> s = ['h', 'a', 'n', 'd', 'o', 'n', 'g']
```

```
>>> s.remove('n')
```

```
>>> s
```

```
['h', 'a', 'd', 'o', 'n', 'g']
```

# 강의 요약

19

<code>.append()</code>	리스트 내에 새로운 아이템 한 개를 추가하여, 마지막에 위치한다
<code>.insert()</code>	리스트 내에 index 번호와 아이템 내용을 추가한다
<code>.extend()</code>	다른 이름의 리스트, 아이템 모두를 추가한다
<code>.sort()</code>	리스트의 아이템들을 순서대로 정렬, 순서는 ascii code 순이다
<code>.pop()</code>	리스트 내에 존재하는 아이템의 index 번호를 입력 받아 삭제한다
<code>.remove()</code>	리스트 내에 존재하는 아이템을 삭제, 동일한 아이템 있으면 처음 아이템만 삭제

# 목표 달성 질문

20

- ▶ 다음 리스트 메소드 기능을 설명하시오
  - ▶ `.pop()`
  - ▶ `.append()`
  - ▶ `.insert()`

# 감사합니다