

# 리스트 개요

7주차\_02\_01

한 동 대 학 교

김경미 교수

# 학습목표

2

- ▶ 리스트를 이해하기
- ▶ 리스트에서 연산자 활용하기

# List 리스트(목록 또는 배열)

- ▶ 리스트는 값들의 나열(sequence)이다
  - ▶ 리스트 안 구성요소를
    - ▶ 원소(elements) 혹은 항목(items)이라고 부른다
    - ▶ 다양한 종류의 데이터타입으로 구성 가능하다
  - ▶ 리스트의 내용은 변경 가능하다
  - ▶ 각 항목은 ,(кома) 로 구분한다.
  - ▶ 앞 부터 순서대로 index를 가지며, index는 0부터 시작된다.

# 리스트 예제

```
>>> cheeses = ['Cheddar', 'Edam', 'Gouda']  
>>> numbers = [17, 123]  
>>> print(numbers)  
[17, 123]  
  
>>> cheeses[0]  
Cheddar
```

# 리스트에서 in operator

- ▶ in : 리스트 안에 해당 element 가 존재하는지 알려줌
- ▶ 항목 in 리스트

```
animal=['rabbit', 'dog', 'tiger', 'lion', 'snake']  
a = "lion"
```

```
if a in animal:  
    print(a,"is in animal")  
else:  
    print(a,"is not in animal")
```

```
lion is in animal
```

# 리스트에서 in operator

```
>>> cheeses = ['Cheddar', 'Edam', 'Gouda']
```

```
>>> 'Edam' in cheeses
```

```
True
```

```
>>> 'Brie' in cheeses
```

```
False
```

```
>>> cheeses
```

```
['Cheddar', 'Edam', 'Gouda']
```

```
>>> for food in cheeses :
```

```
    print(food)
```

```
Cheddar
```

```
Edam
```

```
Gouda
```

# 리스트에서 len() operator

- ▶ len(): 리스트 내 항목 개수를 반환함
- ▶ len(리스트 변수명)

```
animal=['rabbit', 'dog', 'tiger', 'lion', 'snake']
```

```
for a in range(len(animal)-1):  
    print(animal[a])
```

```
rabbit  
dog  
tiger  
lion
```

# 리스트 다루기

8

```
cheeses = ['Cheddar', 'Edam', 'Gouda']  
numbers = [1, 3, 5, 7, 9, 11]
```

```
for cheese in cheeses :  
    print(cheese)
```

```
for i in range(len(numbers)) :  
    numbers[i] = numbers[i] * 2  
    print(numbers[i])
```

```
print(numbers)
```

```
>>>  
===== RESTART: E:/1_Works/  
py =====  
Cheddar  
Edam  
Gouda  
2  
6  
10  
14  
18  
22  
[2, 6, 10, 14, 18, 22]  
>>> |
```



# 리스트, 연산자

9

- ▶ Operator
- ▶ `+`: 두개의 리스트를 합하여 새로운 리스트를 생성
- ▶ `*`: (리스트) \* (정수) 형태로 정수 수만큼 리스트의 내용이 배가된다

# The + operator concatenates lists

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

```
>>> b = [4, 5, 6]
```

```
>>> c = a + b
```

```
>>> c
```

```
[1, 2, 3, 4, 5, 6]
```

# the \* operator repeats a list a given number of times

```
>>> ['a'] * 4
```

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

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

```
>>> a * 3
```

```
[1, 2, 3, 1, 2, 3, 1, 2, 3]
```

# 리스트 slice

- ▶ Slice
- ▶ 문자열 사용과 동일
- ▶ index를 사용하여, 리스트 내의 아이템들을 일부 사용한다
- ▶ List\_fruit[:3]
- ▶ 문자열에서 사용하는 것과 동일
- ▶ 문자열이 아닌 리스트 형으로 리턴된다.

```
#list slice
>>> t = ['a', 'b', 'c', 'd', 'e', 'f']
>>> t[1:3]
['b', 'c']
>>> t[:4]
['a', 'b', 'c', 'd']
>>> t[3:]
['d', 'e', 'f']
```

# 연습문제 I

11

- ▶ 새로 생성하는 List의 item 수를 입력 받는다
- ▶ list를 생성한다
- ▶ item의 수만큼 반복해서 값을 입력 받는다
- ▶ '+' 연산자를 사용하여 list에 값을 추가한다
- ▶ 전체 리스트를 출력한다

# 연습문제 1, 코드와 결과

12

```
num = int(input("List element 개수 입력: "))
NewList = []
tempList = [0]
```

```
for i in range(num):
    print(i, "번째")
    t = input(" 추가할 element 입력: ")
    tempList = [t]
    NewList = NewList + tempList
```

```
print(NewList)
```

```
>>>
===== RESTART: E:/1_Works/
py =====
List element 개수 입력: 4
0 번째
추가할 element 입력: 12
1 번째
추가할 element 입력: 34
2 번째
추가할 element 입력: 2
3 번째
추가할 element 입력: 9
['12', '34', '2', '9']
>>> |
```

## 연습문제 2

13

- ▶ `animal=['Rabbit', 'lion', 'snake', 'cabbage', 'Apple', 'banana']`
- ▶ 리스트를 읽으면서 각 아이템에 모음이 몇 개 있는지 센 후 출력한다

## 연습문제 2, 코드와 결과

```
animal=['Rabbit', 'lion', 'snake', 'cabbage', 'Apple', 'banana']
```

```
for s in animal:
```

```
    count = 0
```

```
    for ch in s:
```

```
        if ch in 'AEIOUaeiou':
```

```
            count += 1
```

```
    print(s, "모음 수", count)
```

```
----- FILE
```

```
Rabbit 모음 수 2
```

```
lion 모음 수 2
```

```
snake 모음 수 2
```

```
cabbage 모음 수 3
```

```
Apple 모음 수 2
```

```
banana 모음 수 3
```

# 강의 요약

- ▶ 리스트 이해하기
  - ▶ 값들의 나열(sequence)
  - ▶ 다양한 종류의 데이터타입으로 구성 가능
  - ▶ 리스트 내용 변경 가능
- ▶ 리스트에서 연산자 활용하기
  - ▶  $+$  : 두개의 리스트를 합함
  - ▶  $*$  : (리스트) \* (정수) 형태, 정수 수만큼 배로 증가

# 목표 달성 질문

16

- ▶ 리스트에서 사용 가능한 산술연산자를 나열하시오
- ▶ 문자열로 구성된 리스트  $s1$  에서, 문자열을 하나씩 읽는 문장을 쓰시오



# 감사합니다