

11. 리스트

자료의 집합

- 여러 개의 값을 일괄 적으로 관리
- . 타 프로그래밍 에서는 배열 로 표현 한다.
- . 리스트 의 형식
 - > 리스트 에 소속되는 각각의 값을 요소(Element) 또는 원소 라고 한다.

```
[리스트명] = ['값1', '값2', '값3' .... ]  
score      = [92, 97, 95, 94]  
name       = ['김범수', '나얼', '박효신', '이수']
```

- > 요소 는 보통 타입이 같지만 반드시 그렇지 는 않다.

```
name_score = ['김범수', 95, '나얼', 94, '박효신', 91, '이수', 95]
```

- . 리스트 를 사용하면 요소를 하나씩 추출하여 사용 할 수 있으므로 반복문을 사용하기 용이 하다

```
Chap01_intro > 09.List[리스트].py > ...  
1 scores = [92, 97, 95, 94]  
2 sum = 0  
3 for sco in scores :  
4     sum += sco  
5 print('총점 : ', sum)  
6 print('평균 : ', sum / len(scores))  
7  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
Microsoft Windows [Version 10.0.22621.2134]  
(c) Microsoft Corporation. All rights reserved.  
  
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311  
총점 : 378  
평균 : 94.5
```

- . List() 함수 를 이용하여 문자열을 list 로 변경하기
 - > 문자열 도 문자의 집합 이므로 리스트 이다.
 - > Python 문자열을 리스트 로 생성

```
Chap01_intro > 09.List[리스트].py  
1 print(list('Python'))  
2  
3  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
Microsoft Windows [Version 10.0.22621.2134]  
(c) Microsoft Corporation. All rights reserved.  
  
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311  
['P', 'y', 't', 'h', 'o', 'n']
```

- > 리스트 의 요소 는 문자열 의 요소 사용법과 동일하다.

```
Chap01_intro > 09.List[리스트].py > ...  
1 nums = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]  
2 print(nums[2:5])  
3 print(nums[:4])  
4 print(nums[6:])  
5 print(nums[1:7:2])  
6  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
Microsoft Windows [Version 10.0.22621.2134]  
(c) Microsoft Corporation. All rights reserved.  
  
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311  
[2, 3, 4]  
[0, 1, 2, 3]  
[6, 7, 8, 9]  
[1, 3, 5]
```

- > 2행 : 2 ~ 4까지
- > 3행 : 0 ~ 3까지
- > 4행 : 6 ~ 마지막 까지
- > 5행 : 1 ~ 6 까지 2단계씩 증가하는 index 의 값

- 리스트의 값 할당

. 리스트는 요소의 값을 대입할 수 있다.

```
Chap01_intro > 09.List[리스트].py > ...
1  nums = [0,1,2,3,4,5,6,7,8,9]
2  print(nums[2])
3  nums[2] = 55
4  print(nums[2])
5

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/P
2
55
```

> nums[2] 위치의 값이 2에서 55로 바뀌어 표현된다.

. 범위를 지정하여 값을 할당할 수 있다.

> 정확한 범위만큼 값의 요소를 할당 : 값의 변경

```
Chap01_intro > 09.List[리스트].py > ...
1  nums = [0,1,2,3,4,5,6,7,8,9]
2  nums[2:5] = [20,30,40]
3  print(nums)
4
5

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/P
[0, 1, 20, 30, 40, 5, 6, 7, 8, 9]
```

> 2행 : 2 ~ 4까지의 3개 범위의 값을 각각 20, 30, 40으로 변경

> 범위의 값과 다른 값 요소를 할당

```
Chap01_intro > 09.List[리스트].py > ...
1  nums = [0,1,2,3,4,5,6,7,8,9]
2  nums[6:8] = [60,70,80,90]
3  print(nums)
4
5

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/P
[0, 1, 2, 3, 4, 5, 60, 70, 80, 90, 8, 9]
```

> 2행 : 6~7까지 2개 범위 요소 값을 60, 70, 80, 90 4개의 값으로 등록
6 ~ 7 범위의 데이터 삭제
index 6 자리부터 60, 70, 80, 90 대입
기존의 8, 9는 뒤로 밀리면서 index 변경

- 리스트의 값 삭제

. 범위 지정하여 삭제하기

```
Chap01_intro > 09.List[리스트].py > ...
1  nums = [0,1,2,3,4,5,6,7,8,9]
2  nums[6:9] = []
3  print(nums)
4
5

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/P
[0, 1, 2, 3, 4, 5, 9]
```

> 2행 : 6, 7, 8 인덱스 요소를 삭제함

- 리스트 연산

. 리스트의 합 (+)

> 두 리스트를 합하여 표현할 수 있다.

```
Chap01_intro > 09.List[리스트].py > ...
2  nums2 = [10,11,12,13]
3
4  listadd = nums1 + nums2
5  print(listadd)
6

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Py
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
```

. 리스트의 곱 (*)

> 리스트의 요소를 곱하기 횟수만큼 반복하여 증가시킨다.

```
Chap01_intro > 09.List[리스트].py > nums2
1  nums2 = [10,11,12,13]
2
3  listadd = nums2 * 3
4  print(listadd)
5

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Py
[10, 11, 12, 13, 10, 11, 12, 13, 10, 11, 12, 13]
```

실습

1. Score 리스트에 실적 값 8개를 저장하고 총점과 평균을 구해 출력하세요

```
7:77..(debugpy\launcher 5402
총점 : 700, 평균 : 87.5
```

이중 리스트

- 리스트의 요소에 리스트를 배치 함으로서 리스트를 담은 리스트를 활용 할 수 있다.

. 리스트끼리의 중첩으로 [] 괄호 안에 [] 괄호를 통하여 표현한다.

```
Chap01_intro > 09.List[리스트].py > ...
1  lol = [[1,2,3],[4,5],[6,7,8,9]]
2  print(lol[0])
3  print(lol[2][1])
4
5

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>cmd /C "C:\Users\MasterD\AppData\Local\Programs\Python\Py
2023.16.0\pythonFiles\lib\python\debugpy\adapter/../../debugpy\laun
[[1, 2, 3]]
[7]
```

lol				
lol[0]	0	1	2	INDEX 값
	1	2	3	
lol[1]	0	1		INDEX 값
	4	5		
lol[2]	0	1	2	INDEX 값
	6	7	8	

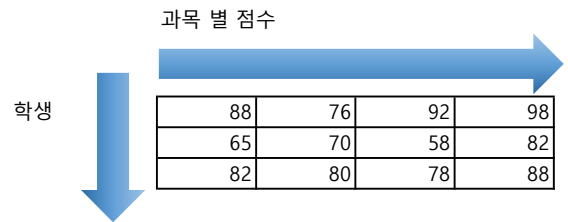
. 이차원 리스트로 성적을 처리하는 예제 만들기

```
Chap01_intro > 09.List[리스트].py > ...
1 # 학생 3 명의 4개 과목에 대한 성적
2 score = [
3     [88,76,92,98],
4     [65,70,58,82],
5     [82,80,78,88],
6 ]
7
8 total = 0
9 totalsub = 0
10 for student in score :
11     sum = 0
12     for subject in student :
13         sum += subject
14     subjects = len(student)
15     print('총점 %d, 평균 %.2f' % (sum,sum/subjects))
16     total += sum
17     totalsub += subjects
18 print('전체 평균 %.2f' % (total/totalsub))
```

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01/09.List[리스트].py

총점 354, 평균 88.50
총점 275, 평균 68.75
총점 328, 평균 82.00
전체 평균 79.75



실습

1. 위 의 예제 에서 과목 별 총점과 평균 을 구하는 로직을구현해 보세요

```
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01/09.List[리스트].py
```

1 번째 과목의 총점 : 235, 평균 : 78.33
2 번째 과목의 총점 : 226, 평균 : 75.33
3 번째 과목의 총점 : 228, 평균 : 76.00
4 번째 과목의 총점 : 268, 평균 : 89.33

리스트 컴프리 헨션

- 리스트의 요소 가 일정한 규칙을 가진 수열 일 경우

. [수식 for 변수 in 리스트 if 조건] 으로 할당 할 수 있다.

> [] 에 나열 될 값을 하나하나 적는것이 아니라 for 반복을 통한 일정 규칙의 배열 값으로 등록 한다.

```
Chap01_intro > 09.List[리스트].py > ...
1 nums = [n * 2 for n in range(1,11)]
2 for i in nums :
3     print(i,end=', ')
4
```

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01/09.List[리스트].py

2, 4, 6, 8, 10, 12, 14, 16, 18, 20,

```
nums = []
for n in range(1,11):
    nums.append(n*2)
```

< 위 로직으로 구현 할 수 있다. >

. If 를 통하여 조건을 추가 할 수 있다.

> 반복 되는 수 가 짝수 일 경우 만 2 를 곱하여 리스트 를 생성

```
Chap01_intro > 09.List[리스트_튜플].py > ...
1 nums = [n * 2 for n in range(1,11) if n % 2 == 0]
2 for i in nums :
3     print(i,end=', ')
4
```

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01/09.List[리스트_튜플].py

4, 8, 12, 16, 20,

데이터 삽입

- 리스트에 값을 삽입하는 예제
 - . Append(5) : 리스트의 마지막에 5를 등록
 - . Insert(2,56) : index 2에 56을 삽입 및 나머지 데이터 재정렬

```
Chap01_intro > 09.List[리스트.튜플].py > ...
1  nums = [1,2,3,4]
2  nums.append(5)
3  nums.insert(2,56)
4  print(nums)
5
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01_intro/09
[1, 2, 56, 3, 4, 5]
```

- . 리스트의 범위를 지정하여 데이터 삽입하기.

```
Chap01_intro > 09.List[리스트.py] > ...
1  # index 2번째 자리부터 3개의 데이터 순차적 삽입 및 index 정렬
2  nums = [1,2,3,4,5]
3  nums[2:2] = [90,91,92]
4  print(nums)
5
6  # index 2에 리스트를 삽입
7  nums = [1,2,3,4,5]
8  nums[2] = [90,91,92]
9  print(nums)
10
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01_intro/09
[1, 2, 90, 91, 92, 3, 4, 5]
[1, 2, [90, 91, 92], 4, 5]
```

>3행 : index 2의 위치부터 각각에 90, 91, 92를 삽입하고
3, 4, 5의 index는 92 다음 index로 재정렬한다.

> 8행 : index 2 위치에 [90,91,92] 리스트를 삽입한다.

- . extend()

```
Chap01_intro > 09.List[리스트.py] > ...
1  list1 = [1,2,3,4]
2  list2 = [6,7,8]
3  list1.extend(list2)
4  print(list1)
5
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01_intro/09
[1, 2, 3, 4, 6, 7, 8]
```

>3행 : extend() 기능으로 리스트 2개를 연결한다.

* extend()는 아래의

```
list1 = [1,2,3,4]
list2 = [6,7,8]
list3 = list1 + list2
print(list3)
```

과 같은 결과를 나타낸다

삭제

- 리스트의 요소를 삭제

```
Chap01_intro > 09.List[리스트.py] > ...
1  score = [88,95,70,100,99,80,78,50]
2
3  score.remove(100)
4  print(score)
5
6  del(score[2])
7  print(score)
8
9  del score[2]
10 print(score)
11
12 score[1:4] = []
13 print(score)
14
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01_intro/09
[88, 95, 70, 99, 80, 78, 50]
[88, 95, 99, 80, 78, 50]
[88, 95, 80, 78, 50]
[88, 50]
```

>3행 : 100의 값을 가진 index를 삭제

>6행 : index 2의 값 삭제

>9행 : index 2의 값 삭제

>12행 : index 1 ~ 3의 값 삭제

- 리스트의 내용 모두 삭제

```
Chap01_intro > 09.List[리스트].py > ...
1 score = [88,95,70,100,99,80,78,50]
2 score.clear()
3 print(score)
4
5 score = [88,95,70,100,99,80,78,50]
6 del score[:]
7 print(score)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311
[]
[]

>2행 : .clear() 메서드 사용

>6행 : del score[:] 사용

- 삭제 후 값을 리턴(Stack, Queue)

- . pop()
- > 기본은 가장 후위에 등록 된 데이터 부터 추출 하여 삭제 및 반환
- > 특정 index 지칭 가능

```
Chap01_intro > 09.List[리스트].py > ...
1 score = [88,95,70,100,99]
2 print(score.pop())
3 print(score.pop())
4 print(score.pop(1))
5 print(score)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Py
99
100
95
[88, 70]

>2행 : 99 삭제 및 출력

>3행 : 100 삭제 및 출력

>4행 : index 1 삭제 및 출력

> 결과 : 88 과 70 만 남게 된다.

검색

- 특정 값 찾기

- . index() : 특정 값의 index 를 반환한다.
- . count() : 특정 값의 개수 를 반환한다.

```
Chap01_intro > 09.List[리스트].py > ...
1 score = [88,95,70,100,99,80,78,50]
2 perfect = score.index(100)
3 print('만점받은 학생은' + str(perfect + 1) + ' 번 입니다.')
4 pernum = score.count(100)
5 print('만점자 수는 ' + str(pernum) + ' 명입니다.')
6
7
8
9
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Python/Chap01_intro/0
만점받은 학생은3 번 입니다.
만점자 수는 1 명입니다.

- 개수 와 최고 , 최저 의 값 찾기

```
Chap01_intro > 09.List[리스트].py > ...
1 score = [88,95,70,100,99,80,78,50]
2 print('학생 의 수는 %d 명 입니다.' % len(score))
3 print('최고 점수는 %d 입니다.' % max(score))
4 print('최저 점수는 %d 입니다.' % min(score))
5
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Py
학생 의 수는 8 명 입니다.
최고 점수는 100 입니다.
최저 점수는 50 입니다.
```

- 특정 요소 가 있는지 검사

. in , not in 을 통하여 확인 가능.

```
Chap01_intro > 09.List[리스트].py > ...
1 ans = input('결제 하시겠습니까?')
2 result = ['y','예','yes','OK']
3 if ans in result :
4     print('구입해 주셔서 감사합니다.')
5 else :
6     print('안녕히 가세요')
7
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/p
결제 하시겠습니까?y
구입해 주셔서 감사합니다.
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/p
결제 하시겠습니까?n
안녕히 가세요
```

* 모든 경우의 수를 비교 연산자 를 통해 비교 하지 않아도 된다.
if ans == 'y' or ans == 'yes' or ans == 'ok'

정렬

- 크기 순으로 재 배열

. 오름차순 sort()

. 내림차순 sort() -> reverse()

```
Chap01_intro > 09.List[리스트].py > ...
1 score = [88,95,70,100,99,80,78,50]
2 score.sort()
3 print(score)
4 score.reverse()
5 print(score)
6
7
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311
[50, 70, 78, 80, 88, 95, 99, 100]
[100, 99, 95, 88, 80, 78, 70, 50]
```

>4행 : sort 를 통해 정렬을 한 값을 역순으로 나열하여 내림차순이 된다.
* 아래 방법으로 곧바로 내림차순으로 표현 할 수 있다.

```
Chap01_intro > 09.List[리스트].py > ...
1 score = [88,95,70,100,99,80,78,50]
2 score.sort(reverse=True)
3 print(score)
4
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python3
[100, 99, 95, 88, 80, 78, 70, 50]
```

- 문자열 의 오름 차순 표시

. 문자열은 대문자가 소문자 보다 우선 순위 를 가지게 된다.

```
Chap01_intro > 09.List[리스트].py > ...
1 country = ['korea','japan','CHINA','america']
2 country.sort()
3 print(country)
4
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.
D:\Python>C:/Users/MasterD/AppData/Local/Programs/Python/Python311/python.exe d:/Py
['CHINA', 'america', 'japan', 'korea']
```

> 'C' 가 'a' 보다 우선순위

. 문자열의 대 소 문자 관계없이 순서대로 정렬

> sort.(key = str.lower)

```
Chap01_intro > 09.List[리스트].py > ...
1 country = ['korea', 'japan', 'CHINA', 'america']
2 country.sort(key=str.lower)
3 print(country)
4

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:\Users\MasterD\AppData\Local\Programs\Python\Python311\python.exe d:/Py
['america', 'CHINA', 'japan', 'korea']
```

. 원본은 그대로 두고 정렬의 결과를 반환하여 담기

> sorted()

```
Chap01_intro > 09.List[리스트].py > ...
1 country = ['korea', 'japan', 'CHINA', 'america']
2 country2 = sorted(country)
3 print(country)
4 country2.sort(key=str.lower)
5 print(country2)
6

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

D:\Python>C:\Users\MasterD\AppData\Local\Programs\Python\Python311\python.exe d:/Python/Cha
['korea', 'japan', 'CHINA', 'america']
['america', 'CHINA', 'japan', 'korea']
```

>2행 : 원본은 그대로 두고 정렬 한 값을 복사한다.

>4행 : 대 소문자 구분 없이 오름차순 정렬

실습

1. 사용자 로 부터 5개 의 성적 을 입력받아 리스트 에 저장 후 오름차순으로 정렬하여 출력해 보세요

- 정수만 입력 받을 수 있도록 함

- 5번째 성적을 정상적으로 입력 완료 후 정렬 결과 출력

```
1 번째 성적을 입력하세요 : 80
2 번째 성적을 입력하세요 : 90
3 번째 성적을 입력하세요 : 9g
정수 만 입력하세요
3 번째 성적을 입력하세요 : 95
4 번째 성적을 입력하세요 : 70
5 번째 성적을 입력하세요 :
정수 만 입력하세요
5 번째 성적을 입력하세요 : 80
['70', '80', '80', '90', '95']
```