

제7장 리스트, 튜플, 딕셔너리

■ 학습목표

- **리스트**의 개념을 이해하고 활용할 수 있다.
- **튜플**의 개념을 이해하고 활용할 수 있다.
- **딕셔너리**의 개념을 이해하고 활용할 수 있다.

리스트(List)의 기본

2개의 숫자 입력받아 합계구하기

```
a, b = 0, 0
hap = 0
```

```
a = int(input("1번째 숫자 : "))
b = int(input("2번째 숫자 : "))
```

```
hap = a + b
```

```
print(" 합계 => %d" % hap)
```

100개의 숫자 입력받아 합계구하기

```
a1, a2, a3, ....., a100 = 0, 0, 0, ....., 0
hap = 0
```

```
a1 = int(input("1번째 숫자 : "))
a2 = int(input("2번째 숫자 : "))
a3 = int(input("3번째 숫자 : "))
...
```

```
a100 = int(input("100번째 숫자 : "))
```

```
hap = a1 + a2 + a3 + ..... + a100
```

```
print(" 합계 => %d" % hap)
```

리스트(List)의 기본

■ 리스트

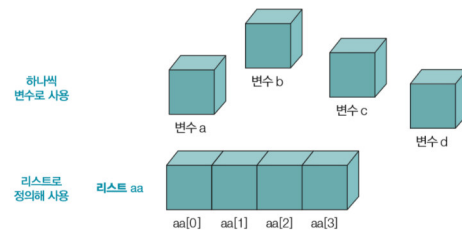


그림 7-1 리스트의 개념

- **Tip:** C/C++나 자바 같은 프로그래밍 언어에는 리스트가 없음. 리스트와 비슷한 개념인 배열(Array) 사용. 리스트는 정수, 문자열, 실수 등 서로 다른 데이터형도 하나로 묶을 수 있지만, 배열은 동일한 데이터형만 묶을 수 있음.

리스트(List)의 기본

■ 리스트 (list)

- 여러 데이터형의 자료를 저장할 수 있는 자료
- 리스트명 = [값1, 값2, 값3,]
- 첨자(인덱스)는 0부터 시작함
- 인덱스 (index), 첨자 : 리스트 대괄호 안에 들어간 숫자
- 요소(element) : 리스트 내부에 넣는 데이터

예

```
a = [5, 10, '파이썬', true]
```

```
print(a)
```



```
[5, 10, '파이썬', true]
```

[0]	[1]	[2]	[3]
5	10	파이썬	true

리스트(List)의 기본

리스트 선언하고 요소에 접근

예

```
a = [5, 10, '파이썬', true]
a[0]
a[1:2]
```

[0]	[1]	[2]	[3]
5	10	파이썬	true

⇒

```
5
10, 파이썬
```

리스트 특정 요소 변경 : 리스트명[인덱스] = 값

예

```
a = [5, 10, '파이썬', true]
a[0] = "안녕"
```

[0]	[1]	[2]	[3]
5	10	파이썬	true
안녕	10	파이썬	true

리스트(List)의 기본

예 : 4개의 숫자 입력받아 합계구하기

```
a, b, c, d = 0, 0, 0, 0
hap = 0
```

```
a = int(input("1번째 숫자 : "))
b = int(input("2번째 숫자 : "))
c = int(input("3번째 숫자 : "))
d = int(input("4번째 숫자 : "))
```

```
hap = a + b + c + d
```

```
print(" 합계 ==> %d" % hap)
```

```
a = [0, 0, 0, 0]
hap = 0
```

```
a[0] = int(input("1번째 숫자 : "))
a[1] = int(input("2번째 숫자 : "))
a[2] = int(input("3번째 숫자 : "))
a[3] = int(input("4번째 숫자 : "))
```

```
hap = a[0] + a[1] + a[2] + a[3]
```

```
print(" 합계 ==> %d" % hap)
```

1번째 숫자 : 10
2번째 숫자 : 20
3번째 숫자 : 30
4번째 숫자 : 40
합계 ==> 100

리스트(List)의 기본

리스트의 일반적인 사용

- 빈 리스트 생성 : `리스트명=[]`
- 리스트에 항목 추가 : `리스트명.append(값)`
- 리스트 개수 : `len(리스트)`
- 리스트 중간에 요소 추가 : `리스트명.insert(위치, 값)`

예:

```
a = [0, 0, 0, 0]
print(a)
print(len(a))
```

```
[0, 0, 0, 0]
4
```

```
a = []
a.append(0)
a.append(0)
a.insert(1,5)
```

```
print(a)
print(len(a))
```

```
[0, 5, 0]
3
```

리스트(List)의 기본

리스트 선언하고 요소에 접근하기

```
list_a = [1,2,3]
list_b = [4,5]
```

```
print("list_a = ", list_a)
print("list_b = ", list_b)
```

```
print("list_a + list_b = ", list_a + list_b)
```

```
print("list_a 길이 = ", len(list_a))
```

```
list_a = [1, 2, 3]
list_b = [4, 5]
```

```
list_a + list_b = [1, 2, 3, 4, 5]
```

```
list_a 길이 = 3
```

리스트(List)의 기본

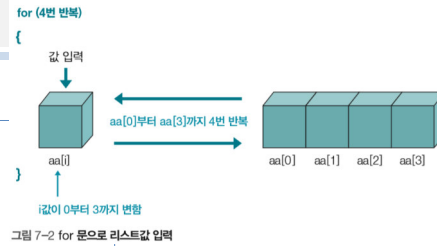
- 예 : for문으로 리스트값 입력

```
a = []
for i in range(0, 4) :
    a.append(0)
print("리스트 a: ", a)

hap = 0

for i in range(0, 4) :
    a[i] = int(input(str(i+1) + "번째 숫자 : "))
    hap += a[i]

print("합계 => %d" % hap)
```



리스트 a: [0, 0, 0, 0]

1번째 숫자 : 10
2번째 숫자 : 20
3번째 숫자 : 30
4번째 숫자 : 40
합계 => 100

리스트(List)의 기본

- 예 : for문으로 리스트값 출력

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

for element in a :
    print(element)

print("\n")

for i in range(0,3) :
    print(a[i], end = ' ')
```

10
20
30

10 20 30

리스트(List)의 기본

- 리스트 생성과 초기화

- 예

① a = []

② b = [10, 20, 30]

③ c = ['파이썬', '공부는', '꿀잼']

④ d = [10, 20, '파이썬']

① 빈 리스트 생성

② 정수로만 구성된 리스트 생성

③ 문자열로만 구성된 리스트 생성

④ 다양한 데이터형을 섞어 리스트 생성

리스트(List)의 기본

- 예 : 리스트 100개인 a를 0, 2, 4, ... (2의 배수)로 초기화한 후 리스트 b에 역순으로 넣기

a = []; b = []; value = 0

```
for i in range(0, 100) :
    a.append(value)
    value += 2
```

```
for i in range(0, 100) :
    b.append(a[99 - i])
```

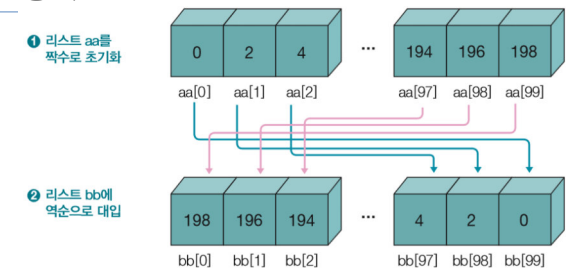


그림 7-3 리스트의 초기화 및 역순 대입

print("b[0]에는 %d이, b[99]에는 %d이 입력됩니다." % (b[0], b[99]))

b[0]에는 198이, b[99]에는 0이 입력됩니다.

리스트(List)의 기본

리스트 값 접근 방법

- (양수/음수)첨자, 콜론(:), 리스트명[시작값:끝값+1], 콜론(:)의 앞/뒤 숫자 생략, 리스트끼리 덧셈/곱셈, 리스트항목 건너뛰며 추출

예

```
a=[10, 20, 30, 40]
b=[50, 60]
```

① a[0] # 10	⑦ a[:2] # 10, 20
② a[-1] # 40	⑧ a+b # 10, 20, 30, 40, 50, 60
③ a[-2] # 30	⑨ b*2 # 50, 60, 50, 60
④ a[0:3] # 10, 20, 30	⑩ a[:2] # 10, 30
⑤ a[2:4] # 30, 40	⑪ a[::2] # 40, 20
⑥ a[2:] # 30, 40	⑫ a[::1] # 40, 30, 20, 10

리스트(List)의 기본

리스트 값 변경

- 리스트명[첨자] = 값
- 리스트명[시작값:끝값] = [값 리스트]

리스트 항목 삭제

- del(리스트명[첨자])
- 리스트명[시작값:끝값+1] = [] → 항목 여러 개 삭제

리스트 자체 삭제

- 리스트명 = [] → 리스트 내용 모두 삭제하고 빈 리스트 만들
- 리스트명 = None → 리스트에 None 값 넣어 빈 변수로 만들
- del(리스트명) → 리스트 변수 삭제

리스트(List)의 기본

예: 리스트 값 변경, 항목 삭제, 리스트 자체 삭제

① a=[10, 20, 30]; a[1] = 200	① 10, 200, 30
② a=[10, 20, 30]; a[1:2] = [200, 201]	② 10, 200, 201, 30
③ a=[10, 20, 30]; del(a[1])	③ 10, 30
④ a=[10, 20, 30, 40, 50]; a[1:4] = []	④ 10, 50
⑤ a=[10, 20, 30]; a[]	⑤ []
⑥ a=[10, 20, 30]; a = None	⑥ 아무것도 안나옴
⑦ a=[10, 20, 30]; del(a)	⑦ 오류 발생

리스트(List)의 기본

표 7-1 리스트 조작 함수

함수	설명	사용법
append()	리스트 맨 뒤에 항목을 추가한다.	리스트명.append(값)
pop()	리스트 맨 뒤의 항목을 빼낸다(리스트에서 해당 항목이 삭제된다).	리스트명.pop()
sort()	리스트의 항목을 정렬한다.	리스트명.sort()
reverse()	리스트 항목의 순서를 역순으로 만든다.	리스트명.reverse()
index()	지정한 값을 찾아 해당 위치를 반환한다.	리스트명.index(찾을값)
insert()	지정된 위치에 값을 삽입한다.	리스트명.insert(위치, 값)
remove()	리스트에서 지정한 값을 삭제한다. 단 지정한 값이 여러 개면 첫 번째 값만 지운다.	리스트명.remove(지울값)
extend()	리스트 뒤에 리스트를 추가한다. 리스트의 더하기(+) 연산과 기능이 동일하다.	리스트명.extend(추가할리스트)
count()	리스트에서 해당 값의 개수를 센다.	리스트명.count(찾을값)
clear()	리스트의 내용을 모두 지운다.	리스트명.clear()
del()	리스트에서 해당 위치의 항목을 삭제한다.	del(리스트명[위치])
len()	리스트에 포함된 전체 항목의 개수를 센다.	len(리스트명)
copy()	리스트의 내용을 새로운 리스트에 복사한다.	새리스트=리스트명.copy()
sorted()	리스트의 항목을 정렬해서 새로운 리스트에 대입한다.	새리스트=sorted(리스트)

```
myList = [30, 10, 20];
print("현재 리스트 : %s" % myList)
myList.append(40);
print("append(40) 후 : %s" % myList)
print("pop()으로 추출한 값 : %s" % myList.pop())
print("pop() 후 : %s" % myList)
```

현재 리스트 : [30, 10, 20]
 append(40) 후 : [30, 10, 20, 40]
 pop()으로 추출한 값 : 40
 pop() 후 : [30, 10, 20]

```
myList.sort();
print("sort() 후 : %s" % myList)
myList.reverse();
print("reverse() 후 : %s" % myList)
print("20값의 위치 : %d" % myList.index(20))
```

sort() 후 : [10, 20, 30]
 reverse() 후 : [30, 20, 10]
 20값의 위치 : 1

```
myList.insert(2, 222);
print("insert(2, 222) 후 : %s" % myList)
myList.remove(222);
print("remove(222) 후 : %s" % myList)
myList.extend([77, 88, 77]);
print("extend([77, 88, 77]) 후 : %s" % myList)
```

insert(2, 222) 후 : [30, 20, 222, 10]
 remove(222) 후 : [30, 20, 10]
 extend([77, 88, 77]) 후 : [30, 20, 10, 77, 88, 77]

```
print("77값의 개수 : %d" % myList.count(77))
```

77값의 개수 : 2

```
myList = [30, 10, 20]
```

```
newList = sorted(myList)
```

```
print("sorted() 후의 myList : %s" % myList)
print("sorted() 후의 newList : %s" % newList)
```

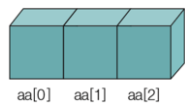
sorted() 후의 myList : [30, 10, 20]
 sorted() 후의 newList : [10, 20, 30]

2차원 리스트

- 1차원 리스트를 여러 개 연결한 것, 첨자를 2개 사용

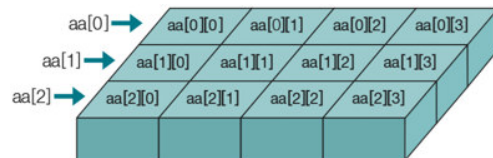
예 aa = [10, 20, 30]

10	20	30
----	----	----



aa = [[1,2,3,4],
 [5,6,7,8],
 [9,10,11,12]]

1	2	3	4
5	6	7	8
9	10	11	12



전체 리스트명 : aa

2차원 리스트

- 예: 2차원 리스트 출력

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

```
for items in a:
    print(items)
    print("\n")
```

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

```
for i in range(0,2):
    for j in range(0,3):
        print(a[i][j], end = ' ')
    print("\n")
```

1 2 3
 4 5 6