

## 1. 리스트 만들기

## 1.1 리스트

**리스트  
(list)**

- 문자열과 달리 변경이 가능(mutable)
- 대괄호[]로 구분  
ex) color=['red','yellow','green','blue']
- 리스트 함수는 글자를 하나하나 분리  
ex) m=list('good')  
m=['g','o','o','d']

## 1.2 리스트 만들기

1.

`li=[]`

: 대괄호를 사용하여 빈 리스트를 만든다

2.

`li=list()`

: 소괄호를 사용하되, 앞에 list임을 명시한다

---

### 코드 1-1 리스트 만들기

---

```
a = ['a','b','c']  
b = list('abc')
```

```
print(a)  
print(b)
```

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

## 2. 리스트 인덱스

---

### 코드 2-1 리스트 인덱스

---

```
li = ['a','b','c']
```

```
print(li[0])  
print(li[1])  
print(li[2])
```

```
li[2] = 'd'      # 리스트 값 변경 가능  
print(li)
```

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

## 3. 리스트 활용

### 3.1 위치 찾기

#### <값으로 항목 위치 찾기>

```
li=['a','b','c','d','e']  
print(li.index('c'))
```

2

→ c는?  
2번 자리

---

#### 코드 3-1 위치 찾기

---

```
li = ['a','b','c','d','e']
```

```
print(li.index('c'))    # 리스트 li 에서 'c'의 위치 찾기
```

---

```
2
```

### 3.2 추가하기

#### <리스트 항목 추가하기>

```
li=['a','b','c','d','e']  
li.append('f')  
print(li)  
  
['a','b','c','d','e','f']
```

#### <리스트 항목 삽입하기>

```
li=['a','b','c','d','e']  
li.insert(1,'aa')  
print(li)
```

→ 1자리에 'aa' 삽입

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

---

#### 코드 3-2 리스트에 추가하기 1

---

```
li = ['a','b','c','d','e']  
li.append('f') # 리스트의 맨 끝에 추가하기  
print(li)
```

---

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

---

#### 코드 3-2-1 리스트에 추가하기 2

---

```
li = ['a','b','c','d','e']  
li.insert(0,'aa') # 원하는 위치에 추가하기  
print(li)
```

---

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

### 3.3 삭제하기

#### <리스트 항목 삭제하기>

```
li=['a','b','c','d','e']  
del li[2] → 인덱스 2번 값 삭제  
print(li)  
  
['a','b','d','e']
```

#### <값으로 항목 삭제하기>

```
li=['a','b','c','d','e']  
li.remove('e')  
print(li)  
  
['a','b','c','d']
```

#### 코드 3-3 삭제하기 1

```
li = ['aa','a','b','c','d','e']  
li.remove('aa') # 값으로 항목 삭제  
print(li)
```

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

#### 코드 3-3-1 삭제하기 2

```
li = ['a','b','c','d','e']  
del li[2] # 인덱스로 삭제하기  
  
print(li)
```

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

### 3.4 확인하기

#### <존재여부 확인하기>

```
li=['a','b','c','d','e']  
print('b' in li) → 'b'가 리스트 li 안에 있나?  
True (참)  
  
True
```

---

코드 3-4 확인하기

---

```
li = ['a','b','c','d','e']
```

```
print('b' in li)          # 'b'가 리스트 li에 있냐?
```

---

```
True
```

### 3.5 전체 개수

#### <항목의 개수 얻기>

```
li=['a','b','c','d','e']  
print(len(li))  
5
```

---

코드 3-5 리스트의 전체 항목 개수 세기

---

```
li = ['a','b','c','d','e']
```

```
print(len(li))
```

---

```
5
```

### 3.6 특정 항목의 개수

#### <특정 항목의 개수 세기>

```
li=['a','b','a','c','a','a']  
print(li.count('a'))  
4
```

→ 리스트 li 안에 'a'가 몇 개인지 출력

---

#### 코드 3-6 리스트의 특정 항목 개수 세기

---

```
li = ['a','b','c','d','e']
```

```
print(li.count('a'))    # 리스트 li 에서 'a'가 몇 개인가?
```

```
1
```

### 3.7 합 / 최대값 / 최소값

- sum() / min() / max()

---

#### 코드 3-7 리스트 항목의 합계, 최대값, 최소값 구하기

---

```
num = [1, 2, 3, 4, 5, 9, 8, 7, 6, 10]
```

```
print(sum(num))    # 합계  
print(min(num))    # 최소값  
print(max(num))    # 최대값
```

```
55
```

```
1
```

```
10
```

### 3.8 정렬 하기

#### <정렬하기>

- \* **sort()**: 리스트 자체를 내부적으로 정렬

```
li=['a','c','d','e','b']  
li.sort()                → ['a','b','c','d','e']  
print(li)
```

- \* **li.sort(reverse=True)**: 역순으로 정렬

```
li=['a','c','d','e','b']  
li.sort(reverse=True)    → ['e','d','c','b','a']  
print(li)
```

- \* **sorted**: 리스트의 정렬된 복사본을 반환

```
li=['a','c','d','e','b']  
a=sorted()               → ['a','c','d','e','b']  
print(li)                → ['a','b','c','d','e']  
print(a)
```

---

#### 코드 3-8 역순 만들기

---

```
num = [1,2,3,4,5,6,7,8,9,10]  
num.reverse()
```

```
print(num)
```

```
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
```

---

#### 코드 3-8-1 오름차순 정렬

---

```
num = [10, 9, 8, 7, 6, 5, 4, 3, 2, 1]  
num.sort()
```

```
print(num)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

---

#### 코드 3-8-1 내림차순 정렬

---

```
num = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
num.sort(reverse=True)
```

```
print(num)
```

```
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
```