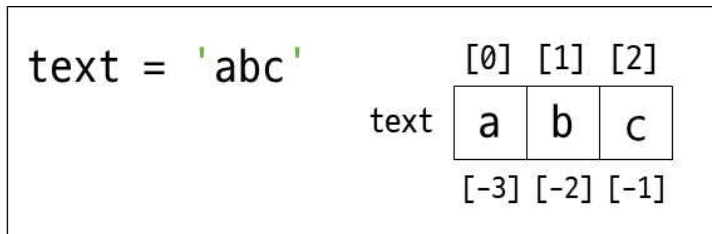


1. 문자열 인덱스



1.1 양수 인덱스

- 0부터 시작합니다.

코드 1-1 문자열 인덱스

```
text = 'abc'
```

```
print(text[0])
```

```
print(text[1])
```

```
print(text[2])
```

```
a
b
c
```

1.2 음수 인덱스

- 마지막이 -1부터 시작합니다.

코드 1-2 문자열 인덱스

```
text = 'abc'
```

```
print(text[-3])
```

```
print(text[-2])
```

```
print(text[-1])
```

```
text = 'abc'
```

```
print(text[-3]) # text[-3] = text[0]
```

```
print(text[-2]) # text[-2] = text[1]
```

```
print(text[-1]) # text[-1] = text[2]
```

2. 문자열 슬라이스

```
text = 'abcde fgh ijk'
```

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]
text	a	b	c	d	e		f	g	h		i	j	k
	[-13]	[-12]	[-11]	[-10]	[-9]	[-8]	[-7]	[-6]	[-5]	[-4]	[-3]	[-2]	[-1]

코드 2-1 문자열 슬라이스

```
text = 'abcde fgh ijk'
```

```
print(text[2:5])    # cde
print(text[1:8])    # bcde fg
print(text[-5:-1])  # h ij
```

```
cde
bcde fg
h ij
```

코드 2-2 문자열 슬라이스

```
text = 'abcde fgh ijk'
```

```
print(text[5:])     # fgh ijk
print(text[:5])     # abcde
print(text[:])      # abcde fgh ijk
```

```
fgh ijk
abcde
abcde fgh ijk
```

코드 2-3 문자열 슬라이스

```
text = 'abcde fgh ijk'
```

```
print(text[0:8:2])    # acef  
print(text[1:8:2])    # bd g
```

```
acef  
bd g
```

코드 2-4 문자열 슬라이스

```
text = 'abcde fgh ijk'
```

```
# 시작하는 인덱스가 뒤에 있어야 함  
print(text[8:0:-1])    # hgf edcb
```

```
hgf edcb
```

코드 2-5 문자열 슬라이스

```
text = 'abcde fgh ijk'
```

```
print(text[::-1])    # kji hgf edcba
```

```
kji hgf edcba
```

3. 문자열 메서드

3.1 출력 지정

- `format(a,b,c, ...)`

코드 3-1 `format()` 메서드

```
text = 'abcde {} {}'  
print(text.format('ABC',123))
```

```
abcde ABC 123
```

3.2 대체하기

- `replace(a,b)`

코드 3-2 `replace()` 메서드

```
text = 'abcde ABC ABC'  
print(text.replace('A','K'))
```

```
abcde KBC KBC
```

코드 3-2-1 `replace()` 메서드

```
text = 'abcde ABC ABC'  
print(text.replace('ABC','KKK'))
```

```
abcde KKK KKK
```

3.3 자르기

- split(a)

코드 3-3 split() 메서드

```
text = 'abcde A/B/C A.B.C'
a,b,c = text.split()          # 공백을 기준으로 자른다.
print(a)
print(b)
print(c)
```

```
abcde
A/B/C
A.B.C
```

코드 3-3-1 split() 메서드

```
text = 'abcde A/B/C A.B.C'
a,b,c = text.split('.')      # .을 기준으로 자른다.
print(a)
print(b)
print(c)
```

```
abcde A/B/C A
B
C
```

코드 3-3-2 split() 메서드

```
text = 'abcde A/B/C A.B.C'
a,b,c = text.split('/')      # / 를 기준으로 자른다.
print(a)
print(b)
print(c)
```

```
abcde A
B
C A.B.C
```

3.4 합치기

- a.join()

코드 3-4 join() 메서드

```
text = 'abcde'          # abcde문자열 사이에 /를 추가
print('/'.join(text))
```

```
a/b/c/d/e
```

3.5 개수 확인하기

- count(a)

코드 3-5 count() 메서드

```
text = 'abcde ABC ABC'
print(text.count('a'))    # 소문자 a가 몇 개 있는가?
print(text.count('A'))    # 대문자 A가 몇 개 있는가?
print(text.count('1'))    # 숫자 문자 1이 몇 개 있는가?
```

```
1
2
0
```

3.6 제거하기

- strip(a) / lstrip(a) / rstrip(a)

코드 3-6 strip() 메서드

```
text = ' abcde '
print(text.strip())      # 양쪽의 공백제거
print(text.lstrip())     # 왼쪽의 공백제거
print(text.rstrip())     # 오른쪽의 공백제거
```

```
abcde
abcde
  abcde
```

3.7 인덱스 찾기

- find(a) / rfind(a) / index(a) / rindex(a)
- find() 메서드는 찾은 문자열이 없으면 -1을 반환하고, index() 메서드는 ValueError를 발생합니다.

코드 3-7 find() / index() 메서드

```
text = 'ABC ABC'
print(text.find('A')) # 왼쪽에서 부터 해당 문자열의 인덱스를 가져온다
print(text.rfind('A')) # 오른쪽에서 부터 해당 문자열의 인덱스를 가져온다
print(text.index('A')) # 왼쪽에서 부터 해당 문자열의 인덱스를 가져온다
print(text.rindex('A')) # 오른쪽에서 부터 해당 문자열의 인덱스를 가져온다
```

```
-1
4
0
4
```

3.8 확인하기

- isalpha() / isdigit() / isalnum() / isupper() / islower()
- isalpha() : 알파벳으로 이루어졌는가?
- isdigit() : 숫자 문자열로만 이루어졌는가?
- isalnum() : 알파벳과 숫자 문자열로만 이루어졌는가?
- isupper() : 대문자로 이루어졌는가?
- islower() : 소문자로 이루어졌는가?

코드 3-8 isalpha() / isdigit() / isalnum() / isupper() / islower() 메서드

```
text1 = 'ABCabc123'
text2 = '123'
text3 = 'ABC'
text4 = 'abc'

print(text1.isalpha()) # 알파벳으로만 이루어졌는가
print(text1.isdigit()) # 숫자문자열로만 이루어졌는가
print(text1.isalnum()) # 알파벳과 숫자문자열로 이루어졌는가
print(text1.isupper()) # 대문자인가
print(text1.islower()) # 소문자인가
```

```
False
False
True
False
False
```

3.9 대/소문자 만들기

- upper() / lower()

코드 3-9 upper() / lower() 메서드

```
text = 'ABCabc'
print(text.upper())  # 문자열을 모두 대문자로 변경
print(text.lower())  # 문자열을 모두 소문자로 변경
```

```
ABCABC
abcabc
```

3.10 0 채우기

- zfill()

코드 3-10 zfill() 메서드

```
y='2020'
m='3'
d='1'

print(y.zfill(4))
print(m.zfill(2))
print(d.zfill(2))
```

```
2020
03
01
```