

# 문자열 인덱싱

- 문자열은 시퀀스 자료형(sequence data type)으로 구성
- 문자에 해당하는 바이트의 배열로 저장

```
a = "abcde"
```

a	0100 1001
b	0100 1010
c	0100 1011
d	0100 1100
e	0100 1101

- 문자열을 구성하는 내부 문자 접근 시 배열의 인덱스 사용
  - ‘HELLO’문자열의 인덱스

Hello

0 1 2 3 4

-5 -4 -3 -2 -1

# 문자열 인덱싱

- 문자열 인덱싱으로 문자추출 예

```
word = 'Python'

print(word[0],word[2] ) # word 문자열에서 0, 2 위치의 문자 추출
for i in word:          # word 문자열의 문자를 하나씩 추출하여 반복출력
    print (i, end=" ")
print()

for i in range(6):      # 0~5 범위의 인덱스를 반복하여 word 문자열의 문자를 인덱스로 접근하여 출력
    print (word[i], end=" ")
print()

#slicing index, 인덱스로 범위 지정, 시작~(끝-1)
print(word[1:3] )       # slice index (1에서 (3-1) 인덱스 범위의 문자 추출
print(word[:2])         # 앞에서 (2-1) 인덱스 범위 문자 추출
print(word[4:] )        # 4에서 끝까지 인덱스 범위 문자 추출
print(word[::-2] )      # 처음부터 끝까지 2씩 증가하면서 문자 추출

#reverse index, 마지막 인덱스부터 -를 붙여 접근
print(word[-1] )        # 마지막 인덱스에 해당하는 문자 추출
print(word[-3:] )       # -3에서 끝까지
print(word[:-2] )       # 처음부터 -2 이전까지
print(word[::-1] )      # 끝에서 처음까지 -1(1씩 감소)하면서 문자 추출

#성명과 년도를 분리, 3을 기준으로 분리
s = '홍길동2002'
name = s[:3]
y = s[3:]
print("성명 : ", name, " 년도 : ", y)
```

```
P t
P y t h o n
P y t h o n
y t
P y
o n
P t o
n
h o n
P y t h
n o h t y P
```

성명 : 홍길동    년도 : 2002

# 문자열 추출

예제 1. 주민등록번호 앞 6자리를 입력하여 생년,월, 일을 추출하는 프로그램

생년(year), 월(month), 일(day)을 추출하는 함수 정의

```
def year(num):
    return (num[:2])

def month(num):
    return (num[2:4])
def day(num):
    return (num[4:])

while True:
    name = input("성명 입력 ? ")
    if (name == 'q'):
        q = input("정말 종료하시겠습니까 (y/n) ? ")
        if (q == 'y'):
            break
        else:
            continue

    num = input("주민등록번호 앞자리 입력 ? ")
    y = year(num)
    m = month(num)
    d = day(num)
    r = name + "님은 " + y + "년" + m + '월' + d +'일에 태어났습니다.'
    print (r)
```

성명 입력 ? 홍길동  
주민등록번호 앞자리 입력 ? 901212  
홍길동님은 90년12월12일에 태어났습니다.  
성명 입력 ? 황진이  
주민등록번호 앞자리 입력 ? 800210  
황진이님은 80년02월10일에 태어났습니다.  
성명 입력 ? q  
정말 종료하시겠습니까 (y/n) ? n  
성명 입력 ? 이몽룡  
주민등록번호 앞자리 입력 ? 991011  
이몽룡님은 99년10월11일에 태어났습니다.  
성명 입력 ? q  
정말 종료하시겠습니까 (y/n) ? y

# 문자열 연산

## • 문자열 결합

- ()안에 문자열을 나열하여 결합
- +를 이용하여 문자열 결합

```
s7=('i '
    'have '
    'a '
    'dream')
s8 = ('i ' 'have ' 'a '
      'dream\n')
s9 = 'i ' + 'have ' + 'a ' + 'dream'
print(s7)
print(s8)
print(s9)
```

i have a dream

i have a dream

i have adream

# 문자열 연산

- 문자열 반복
  - 숫자를 \*하여 문자열 반복

```
#문자열 반복
p = 'python'
s10 = p * 3    #p에 저장된 문자열을 곱하는 횟수만큼 반복
print(s10)

s = 2*"happy" + 3 * "!"
print(s)

bar = 10 * '-'
print(bar)
print("* title *")
print(bar)
```

```
pythonpythonpython
happyhappy!!!
```

```
-----
* title *
-----
```

# 문자열 formatting

- 문자열 안에 특정 값을 삽입하는 처리
  - (1) %서식을 이용하여 특정 값을 형식에 맞추어 포함
  - (2) format() 함수를 이용하여 특정 값을 형식에 맞추어 포함

서식	설명
%s	문자열(string)
%c	문자 1개(character)
%d	정수(integer)
%f	실수(floating-point)
%o	8진수
%x	16진수
%%	문자 % 자체

```

h = 170.567
w = 56.91
out1 = 'height : %.2f cm' %h           #하나의 값만 삽입
out2 = 'height : %.2f cm weight : %.2f kg' %(h, w) #두 개 이상값 삽입
out3 = 'height : {0} cm weight : {1} kg'
print (out1)
print (out2)
print (out3.format(h, w))             #위치값으로 삽입
#h변수는 {0}에 소수점 2자리로, w변수는 {1}에 소수점 1자리로 출력
out4 = 'height : {0:.2f} cm weight : {1:.1f} kg'
print (out4.format(h, w))
    
```

```

height : 170.57 cm
height : 170.57 cm weight : 56.91 kg
height : 170.567 cm weight : 56.91 kg
height : 170.57 cm weight : 56.9 kg
    
```