

파이썬 보충 Take 1

문자열 포매팅

1) 형식문자 활용 포매팅

```
# "%"연산자 뒤의 값이 "%d"를 치환한다."  
str1 = "나는 %d원을 갖고 있다." % 10000  
print(str1)
```

나는 10000원을 갖고 있다.

```
# 실수값 표현하기  
money = 123.456789  
print("[%f]원 입니다." % money)
```

[123.456789]원 입니다.

```
lang = "Python"  
str2 = "Hello %s" % lang  
print(str2)
```

Hello Python

```
# 순수한 % 출력은 %%로 표현  
print("성취도는 %d%% 입니다." % 72)
```

성취도는 72% 입니다.

```
# 형식문자를 포함한 문장만 별도의 변수로 지정  
str3 = "%s 날짜는 %d년 %d월 %d일 입니다."  
  
# 여러 형식문자를 치환할 경우 값들을 괄호로 묶는다.  
msg1 = str3 % ("정모", 2023, 1, 1)  
print(msg1)  
  
msg2 = str3 % ("약속", 2023, 3, 5)  
print(msg2)
```

정모 날짜는 2023년 1월 1일 입니다.
약속 날짜는 2023년 3월 5일 입니다.

```
# 자리수 지정하기 -> 오른쪽 정렬, 빈자리는 공백으로 지정
print("[%10d]원을 입금했습니다." % 120000)
```

[120000]원을 입금했습니다.

```
# 자리수 지정하기 -> 오른쪽 정렬, 빈자리는 0으로 지정
print("[%010d]원을 입금했습니다." % 120000)
```

[0000120000]원을 입금했습니다.

```
# 자리수 지정하기 -> 왼쪽 정렬, 빈자리는 공백으로 지정 (0 불가)
print("[%10d]원을 입금했습니다." % 120000)
```

[120000]원을 입금했습니다.

```
# 문자열의 자리수 지정하기 -> 빈자리는 공백지정만 가능함.
print("[%10s]에 살고 있습니다." % "대한민국")
print("[%10s]에 살고 있습니다." % "대한민국")
```

[대한민국]에 살고 있습니다.
[대한민국]에 살고 있습니다.

```
# 실습을 위한 변수
myMoney = 12345.6789
# 소수점 3자리까지 표현하기 -> 소수점 4째자리에서 반올림.
print("[%0.3f]원 입니다." % myMoney)
```

[12345.679]원 입니다.

```
# 소수점 3자리까지 표현하되 전체 글자수는 10글자 (빈자리 공백)
# 소수점이 자리수에 포함됨
# -> 오른쪽 정렬
print("[%10.3f]원 입니다." % myMoney)
# -> 왼쪽 정렬
print("[%10.3f]원 입니다." % myMoney)
```

[12345.679]원 입니다.
[12345.679]원 입니다.

```
# 소수점 3자리까지 표현하되 전체 글자수는 10글자 (빈자리 0)
# -> 빈 자리를 0으로 채우는 경우는 오른쪽 정렬만 가능함.
print("[%010.3f]원 입니다." % myMoney)
```

[012345.679]원 입니다.

2) format() 메서드 활용 포매팅

```
# 0번째 위치에 format()함수에 전달된 0번째 값을 대입
msg1 = "I eat {0} apples"
print( msg1.format(3) )
```

I eat 3 apples

```
# 여러 개의 값을 혼용하기
msg2 = "{0}은 {1}년 {2}월 {3}일 입니다."
print( msg2.format("개강일", 2023, 1, 26) )
```

개강일은 2023년 1월 26일 입니다.

```
# 이름으로 넣기 --> format() 메서드에 전달되는 파라미터 순서는 영향이 없다.
msg3 = "{name}은 {yy}년 {mm}월 {dd}일 입니다.{{}}"
print( msg3.format(mm=10, name="종강일", dd=20, yy=2023) )
```

종강일은 2023년 10월 20일 입니다.{{}}

```
# 역슬래시에 대한 주의!!!
path = "F:\nice\[04] 통계 기반 데이터 분석"
print(path)
```

F:
ice\[04] 통계 기반 데이터 분석

#02. 구조분해

```
# 리스트 혹은 튜플에 대해...
mylist = [10, 20, 30]
a = mylist[0]
b = mylist[1]
c = mylist[2]
print(a, b, c)
```

```
10 20 30
```

```
x, y, z = mylist  
print(x, y, z)
```

```
10 20 30
```

```
def foo(x, y):  
    a = x + y  
    b = x - y  
    return [a, b]
```

```
bar = foo(100, 50)  
bar
```

```
[150, 50]
```

```
i, j = foo(100, 50)  
print(i, j)
```

```
150 50
```

```
mylist = [10, 20, 30]  
  
for i in range(0, len(mylist)):  
    print(mylist[i])
```

```
10  
20  
30
```

```
mylist = [10, 20, 30]  
  
for i in mylist:  
    print(i)
```

```
10  
20  
30
```

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

```
for i, v in enumerate(mylist):  
    print(i, v)
```

```
0 10  
1 20  
2 30
```