

파이썬 프로그래밍

문자열 메소드와 포매팅



한국기술교육대학교
온라인평생교육원

■ 문자열 포매팅

1. 튜플을 이용한 포매팅

1) 문자열 변환

- 포매팅 문자 : 문자열 내에 존재하는 %
- 포매팅을 활용한 문자열 변환 : 포매팅 문자를 포함하는 문자열 % 튜플

```
print 'name = %s, age = %s' % ('gslee', '24')
```

```
name = gslee, age = 24
```

- 문자열 구성 시 문자 내 % 사용 → 포매팅 문자를 지칭하는 방법
- 문자열 + % + 튜플
- 문자열에 %가 두 개 → 튜플이 2개
- 튜플 첫번째 요소가 앞 %s, 두번째 요소가 뒤 %s에 들어감
- 자리를 잡아주면서 %s에 들어오는 것이 문자열임을 알려줌

■ 문자열 포매팅

1. 튜플을 이용한 포매팅

```
letter = ""
안녕하세요 %s님,

오늘 밤 파티에 참석해 주실 수 있나요?

그럼..

이강성 드림""
name = '홍길동'
print letter % name
print
names = ['한학신', '정인숙', '박미경']
for name in names:
    print letter % name
    print '-' * 40
    print
```

■ 문자열 포매팅

1. 튜플을 이용한 포매팅

안녕하세요 홍길동님,
오늘 밤 파티에 참석해 주실 수 있나요?
그럼..
이강성 드림

안녕하세요 한학신님,
오늘 밤 파티에 참석해 주실 수 있나요?
그럼..
이강성 드림

안녕하세요 정인숙님,
오늘 밤 파티에 참석해 주실 수 있나요?
그럼..
이강성 드림

안녕하세요 박미경님,
오늘 밤 파티에 참석해 주실 수 있나요?
그럼..
이강성 드림

■ '-' * 40 → -라는 문자가 40번 반복하여 출력됨

■ 문자열 포매팅

1. 튜플을 이용한 포매팅

2) 숫자 변환

포매팅 문자	설명
%s	문자열을 포함한 임의의 객체를 문자열로 변환하여 출력 (str() 내장 함수 사용)
%r	문자열을 포함한 임의의 객체를 문자열로 변환하여 출력 (repr() 내장 함수 사용)
%c	1글자 문자 (ex. '%c' % 'k')
%d	10진 정수 (%5d: 5자리를 확보한 후 정수 포매팅)
%i	%d와 동일
%u	부호 없는 정수. 음수는 양수처럼 해석함 (ex. '%u' % -12 --> '4294967284')
%o	8진수 정수 (ex. '%o' % 13 --> 15)
%x	16진수 정수 (소문자 표현) (ex. '%x' % 13 --> 'd')
%X	16진수 정수 (대문자 표현) (ex. '%X' % 13 --> 'D')
%e	부동 소수점 실수를 지수 형태로 표현 (%.2e: 2자리는 소수점 이하 자리수)
%E	%e 와 동일 (대문자 E 표현)
%f	부동 소수점 실수 (%5.2f: 소수점 포함 총 5자리 확보한 후 2자리는 소수점 이하 자리수)
%g	부동 소수점을 편의에 따라 일반 실수 형식이나 지수 형식으로 변환
%G	%g와 동일 (대문자 E 표현)

- %s는 %뒤에 문자열일 경우 사용
- str() → 임의의 객체를 문자열로 반환
- str() , repr() → 모두 문자열로 반환해주는 함수
- %d → 앞에 자리를 확보하기 위해 %5d로 사용 가능 → 5자리 확보
- % 뒤 실수가 들어가는 경우 → %f
- %5.2f → 소수점 포함 총 5자리 확보한 후 소수점 이하는 2자리 사용
- %5.2f 에서 5가 전체자리수, 2가 소수점 이하 자리수
- %s, %r, %c, %d, %f

■ 문자열 포매팅

1. 튜플을 이용한 포매팅

```
print "%s -- %s -- %d -- %f -- %e" % ((1, 2), [3,4,5], 5, 5.3, 101.3)
```

```
(1, 2) -- [3, 4, 5] -- 5 -- 5.300000 -- 1.013000e+02
```

- '%s' → (1,2)를 str함수로 불러서 문자화 함
- 두 번째 %s → [3,4,5]
- %d → 5
- %f → 5.3
- %e → 101.2 → 지수표현방식으로 표현

```
print "%3d -- %5.2f -- %.2e" % (5, 5.356, 101.3)
```

```
5 -- 5.36 -- 1.01e+02
```

- %3d → 5 → 3개의 자리로 확보한 다음 5 숫자 입력
- %5.2f → 5개 자리 확보 후 소수점 이하 2자리 → 5.36
- %5.2f의 f는 자리수를 맞춰주기 위해 반올림이 됨
- %e → 지수형태의 표현 방식 (많이 활용되지 않음)
- %.2e → .2가 있으므로 소수점 뒤에 01이 2에 맵핑
- 101.3이 지수형태로 찍힌 것

```
a = 456
```

```
print '%d -- %o -- %x -- %X' % (a, a, a, a)
```

```
456 -- 710 -- 1c8 -- 1C8
```

- %o → 8진법에 해당하는 수
- %x → 16진수
- %X → 같이 사용하는 알파벳 문자를 대문자로 변경

■ 문자열 포매팅

2. 사전(Dictionary)을 이용한 포매팅

```
print '%(이름)s -- %(전화번호)s' %{'이름':'홍길동', '전화번호':5284}  
print '%(이름)s -- %(전화번호)s' %{'전화번호':5284, '이름':'홍길동'}  
print '%(이름)s -- %(전화번호)s' %{'전화번호':5284, '이름':'홍길동', '주소':'Seoul'}
```

```
홍길동 -- 5284  
홍길동 -- 5284  
홍길동 -- 5284
```

- 사전형태이기 때문에 순서가 바뀌어도 괜찮음
- 주소 항목은 % 앞에 없기 때문에 무시