

Python 書式

[Python]

- <http://docs.python.jp/2/library/string.html#string.Formatter>
- <http://anh.cs.luc.edu/python/hands-on/3.1/handsonHtml/float.html>

文字列フォーマット

フォーマット文字列 % (差し込みオブジェクトのタプル)

コード	対応型
%s	文字列 (数値なども変換)
%d	10 進整数
%f	10 進浮動小数点数
%x	16 進数
%X	16 進数 (大文字)
%o	8 進数
%%	% を表示

精度、桁数

%[フラグ][桁数][.精度]コード

フラグ	意味
0	0 埋め
-	値の左寄せ
+	数値の先頭に符号をつける

```
>>> t = 173
>>> w = 73
>>> print 'あなたの BMI は、%.1f です。' % (w / ((t/100.0)*2))
あなたの BMI は、21.1 です。
```

辞書のキーを指定した埋め込み

%(辞書のキー)コード

```
>>> print '%(donuts)s、%(remon)s' % {'donuts': 'ドはドーナツのド', 'remon': 'レはレモンのレ'}
ドはドーナツのド、レはレモンのレ
```

日付の書式設定

```
>>> time.strftime('%Y/%m/%d', time.localtime())
'2010/02/24'
>>> datetime.today().strftime('%Y%m%d')
'20100224'
>>> datetime.fromtimestamp(os.stat(r'/test.txt').st_ctime).strftime('%Y%m%d')
'20090824'
```

• <http://docs.python.jp/2/library/datetime.html>

指定子	意味	備考
%a	ロケールの短縮された曜日名を表示します	
%A	ロケールの曜日名を表示します	
%b	ロケールの短縮された月名を表示します	
%B	ロケールの月名を表示します	
%c	ロケールの日時を適切な形式で表示します	
%d	月中の日にちを 10 進表記した文字列 [01,31] を表示します	
%f	マイクロ秒を 10 進表記した文字列 [000000,999999] を表示します (左側から 0 埋めされます)	(1)
%H	時 (24 時間表記) を 10 進表記した文字列 [00,23] を表示します	
%I	時 (12 時間表記) を 10 進表記した文字列 [01,12] を表示します	
%j	年中の日にちを 10 進表記した文字列 [001,366] を表示します	
%m	月を 10 進表記した文字列 [01,12] を表示します	
%M	分を 10 進表記した文字列 [00,59] を表示します	
%p	ロケールの AM もしくは PM を表示します	(2)
%S	秒を 10 進表記した文字列 [00,61] を表示します	(3)
%U	年中の週番号 (週の始まりは日曜日とする) を 10 進表記した文字列 [00,53] を表示します 新年の最初の日曜日に先立つ日は 0 週に属するとします	(4)
%w	曜日を 10 進表記した文字列 [0(日曜日),6] を表示します	

%W	年中の週番号 (週の始まりは月曜日とする) を 10 進表記した文字列 [00,53] を表示します 新年の最初の月曜日に先立つ日は 0 週に属するとします	(4)
%x	ロケールの日付を適切な形式で表示します	
%X	ロケールの時間を適切な形式で表示します	
%y	世紀なしの年 (下 2 桁) を 10 進表記した文字列 [00,99] を表示します	
%Y	世紀ありの年を 10 進表記した文字列を表示します	
%z	UTC オフセットを +HHMM もしくは -HHMM の形式で表示します (オブジェクトが naive であれば空文字列)	(5)
%Z	タイムゾーンの名前を表示します (オブジェクトが naive であれば空文字列)	
%%	文字 '%' を表示します	

1. `strftime()` メソッドと共に使われた場合、`%f` 指定子は 1 桁から 6 桁の数字を受け付け、右側から 0 埋めされます。`%f` は C 標準規格の書式セットに拡張されます。
2. `strftime()` メソッドと共に使われた場合、`%p` 指定子は出力の時間フィールドのみに影響し、`%I` 指定子が使われたかのように振る舞います。
3. 範囲は 0 から 61 で正しいです ; これはうるう秒と、(極めて稀ですが) 2 秒のうるう秒を考慮してのことです。
4. `strftime()` メソッドと共に使われた場合、`%U` と `%W` 指定子は、年と曜日が指定された場合の計算でのみ使われます。
5. 例えば、`utcoffset()` が `timedelta(hours=-3, minutes=-30)` を返すとしたら、`%z` は文字列、`'-0330'` で置き換えられます。

Python3

- <https://pyformat.info/>