f–string

f–string	Sortie
`f"{'one'} {'two'}"`	one two
`f"{1} {2}"`	1 2

Chaîne de caractère

`f"{'test':>10}"`	test
`f"{'test':10}"`	test
`f"{'test':_<10}"`	test_____
`f"{'test':^10}"`	test
`f"{'test':^9}"`	test
`f"{'xylophone':.5}"`	xylop
`f"{'xylophone':10.5}"`	xylop

Entier signé ou non

`f"{42}"`	42
`f"{42:d}"`	42
`f"{42:4d}"`	42
`f"{42:04d}"`	0042
`f"{42:+d}"`	+42
`f"{-42: d}"`	-42
`f"{42: d}"`	42
`f"{-42:=5d}"`	-42
`f"{42:=+5d}"`	+42

Nombre flottant (réel)

`f"{3.1415926535:f}"`	3.141593
`f"{3.1415926535:.2f}"`	3.14
`f"{3.1415926535:06.2f}"`	003.14

Notation scientifique (e / E)

`f"{3.14e7:e}"`	3.140000e+07

Sélectionne automatiquement la notation scientifique pour les grands nombres (g / G)

`f"{42:g}, {42** 8:g}"`	42, 9.68265e+12

Binaire b, octal o, hexadécimal x/X, #b → 0b devant

`f"{42:b}"`	101010
`f"{42:x}"`	2a
`f"{42:X}"`	2A
`f"{42:#X}"`	0x2A

```
class Data(object) :
    def __str__(self) :
        return 'str'
    def __repr__(self) :
        return 'repr'
  f"{Data()!s} {Data()!r}"
```

```
str repr
```

```
personne = {'first':'Jean', 'last':'Neymar'}
  f"{personne['first']} {personne['last']}"
```

```
Jean Neymar
```

```
data = [4, 8, 15, 16, 23, 42]
  f"{data[1]} {data[5]}"
```

```
8 42
```

```
class Plante(object):
    type = 'arbre'
  f"{Plante().type}"
```

```
arbre
```

```
from datetime import datetime
dt = datetime(2001, 2, 3, 4, 5)
dfmt, tfmt ='%Y-%m-%d', '%H:%M'
  f"{dt:%Y-%m-%d %H:%M}"
  f"{dt:{dfmt} {tfmt}}"
```

```
2001-02-03 04:05
2001-02-03 04:05
```