

python字符串拼接总结

加号连接

第一种，通过+号的形式：

```
>>> a, b = 'hello', ' world'
>>> a + b
'hello world'
```

逗号连接

第二种，通过,逗号的形式：

```
>>> a, b = 'hello', ' world'
>>> print(a, b)
hello world
```

但是，使用,逗号形式要注意一点，就是只能用于print打印，赋值操作会生成元组：

```
>>> a, b
('hello', ' world')
```

python之禅注：实际上，这不算是字符串连接的一种方式，因为'hello', ' world'会当作一个元组存在，通过解包（unpacking）的方式赋值给变量 a 和 b.

直接连接

第三种，直接连接中间有无空格均可：

```
print('hello'      ' world')
print('hello'world')
```

Python之禅注：这算是Python里面的一种语法糖，连续的字符串会自动拼接成一个字符串。在内存中不会存在两个字符串对象。（代码可左右滑动）

```
>>> def x():
...     a = 'a' + 'b'
...
>>> dis.dis(x)
2          0 LOAD_CONST          1 ('ab')
          3 STORE_FAST          0 (a)
          6 LOAD_CONST          0 (None)
          9 RETURN_VALUE
```

百分号 %

第四种，使用%操作符。

在 Python 2.6 以前，%操作符是唯一一种格式化字符串的方法，它也可以用于连接字符串。

```
print('%s %s' % ('hello', 'world'))
```

format函数

第五种，使用format方法。

format 方法是 Python 2.6 中出现的一种代替 % 操作符的字符串格式化方法，同样可以用来连接字符串。

```
print('{}{}'.format('hello', ' world'))
```

join函数

第六种，使用join内置方法。

字符串有一个内置方法join，其参数是一个序列类型，例如数组或者元组等。

```
print('-'.join(['aa', 'bb', 'cc']))
```

f-string

第七种，使用f-string方式。

Python 3.6 中引入了 Formatted String Literals（字面量格式化字符串），简称 f-string，f-string 是 % 操作符和 format 方法的进化版，使用 f-string 连接字符串的方法和使用 % 操作符、format 方法类似。

```
>>> aa, bb = 'hello', 'world'
>>> f'{aa} {bb}'
'hello world'
```

星号 *

第八种，使用*操作符。

```
>>> aa = 'hello '
>>> aa * 3
'hello hello hello '
```

Python之禅注：*操作符其实是一种操作符重载操作，对应的魔术方法是 `__mul__`

```
>>> a = [1]
>>> a*2
[1, 1]
>>> a.__mul__(3)
[1, 1, 1]
```

小结

连接少量字符串时，推荐使用+号操作符。

如果对性能有较高要求，并且python版本在3.6以上，推荐使用f-string。例如，如下情况f-string可读性比+号要好很多：

```
a = f'姓名: {name} 年龄: {age} 性别: {gender}'
b = '姓名: ' + name + '年龄: ' + age + '性别: ' + gender
```

连接大量字符串时，推荐使用 join 和 f-string 方式，选择时依然取决于你使用的 Python 版本以及对可读性的要求。

Python之禅注：笔者曾经做过一个测试，python3.6中，数据量不大的情况下 + 操作甚至比 join 操作还快。你还有遇到过哪些神操作，欢迎留言。

作者：枫恋寒

链接：<https://segmentfault.com/a/1190000015475309>