

2.2 字符串开头或结尾匹配

问题

你需要通过指定的文本模式去检查字符串的开头或者结尾，比如文件名后缀，URL Scheme等等。

解决方案

检查字符串开头或结尾的一个简单方法是使用 `str.startswith()` 或者是 `str.endswith()` 方法。比如：

```
>>> filename = 'spam.txt'
>>> filename.endswith('.txt')
True
>>> filename.startswith('file:')
False
>>> url = 'http://www.python.org'
>>> url.startswith('http:')
True
>>>
```

如果你想检查多种匹配可能，只需要将所有的匹配项放入到一个元组中去，然后传给 `startswith()` 或者 `endswith()` 方法：

```
>>> import os
>>> filenames = os.listdir('.')
>>> filenames
['Makefile', 'foo.c', 'bar.py', 'spam.c', 'spam.h']
>>> [name for name in filenames if name.endswith(('.c', '.h'))]
['foo.c', 'spam.c', 'spam.h']
>>> any(name.endswith('.py') for name in filenames)
True
>>>
```

下面是另一个例子：

```
from urllib.request import urlopen
```

```
def read_data(name):
    if name.startswith(('http:', 'https:', 'ftp:')):
        return urlopen(name).read()
    else:
        with open(name) as f:
            return f.read()
```

奇怪的是，这个方法中必须要输入一个元组作为参数。如果你恰巧有一个 `list` 或者 `set` 类型的选择项，要确保传递参数前先调用 `tuple()` 将其转换为元组类型。比如：

```
>>> choices = ['http:', 'ftp:']
>>> url = 'http://www.python.org'
>>> url.startswith(choices)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: startswith first arg must be str or a tuple of str, not list
>>> url.startswith(tuple(choices))
True
```

```
>>>
```

讨论🔗

`startswith()` 和 `endswith()` 方法提供了一个非常方便的方式去做字符串开头和结尾的检查。类似的操作也可以使用切片来实现，但是代码看起来没有那么优雅。比如：

```
>>> filename = 'spam.txt'
>>> filename[-4:] == '.txt'
True
>>> url = 'http://www.python.org'
>>> url[:5] == 'http:' or url[:6] == 'https:' or url[:4] == 'ftp:'
True
>>>
```

你可以能还想使用正则表达式去实现，比如：

```
>>> import re
>>> url = 'http://www.python.org'
>>> re.match('http:|https:|ftp:', url)
<_sre.SRE_Match object at 0x101253098>
>>>
```

这种方式也行得通，但是对于简单的匹配实在是有点小材大用了，本节中的方法更加简单并且运行会更快些。

最后提一下，当和其他操作比如普通数据聚合相结合的时候 `startswith()` 和 `endswith()` 方法是很不错的。比如，下面这个语句检查某个文件夹中是否存在指定的文件类型：

```
if any(name.endswith(('.c', '.h'))) for name in listdir(dirname)):
...
```