

## 2.3 用Shell通配符匹配字符串

### 问题

你想使用 **Unix Shell** 中常用的通配符(比如 `*.py` , `Dat[0-9]*.csv` 等)去匹配文本字符串

### 解决方案

`fnmatch` 模块提供了两个函数—— `fnmatch()` 和 `fnmatchcase()` , 可以用来实现这样的匹配。用法如下:

```
>>> from fnmatch import fnmatch, fnmatchcase
>>> fnmatch('foo.txt', '*.txt')
True
>>> fnmatch('foo.txt', '?oo.txt')
True
>>> fnmatch('Dat45.csv', 'Dat[0-9]*')
True
>>> names = ['Dat1.csv', 'Dat2.csv', 'config.ini', 'foo.py']
>>> [name for name in names if fnmatch(name, 'Dat*.csv')]
['Dat1.csv', 'Dat2.csv']
>>>
```

`fnmatch()` 函数使用底层操作系统的大小写敏感规则(不同的系统是不一样的)来匹配模式。比如:

```
>>> # On OS X (Mac)
>>> fnmatch('foo.txt', '*.TXT')
False
>>> # On Windows
>>> fnmatch('foo.txt', '*.TXT')
True
>>>
```

如果你对这个区别很在意, 可以使用 `fnmatchcase()` 来代替。它完全使用你的模式大小写匹配。比如:

```
>>> fnmatchcase('foo.txt', '*.TXT')
False
>>>
```

这两个函数通常会被忽略的一个特性是在处理非文件名的字符串时候它们也是很有用的。比如, 假设你有一个街道地址的列表数据:

```
addresses = [
    '5412 N CLARK ST',
    '1060 W ADDISON ST',
    '1039 W GRANVILLE AVE',
    '2122 N CLARK ST',
    '4802 N BROADWAY',
]
```

你可以像这样写列表推导:

```
>>> from fnmatch import fnmatchcase
>>> [addr for addr in addresses if fnmatchcase(addr, '* ST')]
['5412 N CLARK ST', '1060 W ADDISON ST', '2122 N CLARK ST']
```

```
[....., .....]\n>>> [addr for addr in addresses if fnmatchcase(addr, '54[0-9][0-9]*CLARK*')]\n['5412 N CLARK ST']\n>>>
```

## 讨论🗣️

`fnmatch()` 函数匹配能力介于简单的字符串方法和强大的正则表达式之间。如果在数据处理操作中只需要简单的通配符就能完成的时候，这通常是一个比较合理的方案。

如果你的代码需要做文件名的匹配，最好使用 `glob` 模块。参考5.13小节。