

限定符

在上一小节中的表格中，我们知道 `.` 可以匹配除换行符以外的任意字符，使用 `.` 匹配下列文本：

```
expression
```

但是 `.` 每次只匹配一个字符，如果想一次匹配多个，则使用限定符

限定符	作用
<code>*</code>	匹配零次或多次
<code>+</code>	匹配一次或多次
<code>?</code>	匹配零次或一次
<code>{3}</code>	匹配三次
<code>{3,5}</code>	匹配三到五次
<code>{3,}</code>	匹配三次或以上

下面通过实例了解限定符的区别。 `es` 的匹配结果

```
expression
```

`es+` 的匹配结果(e, 一个或多个 s)

```
expresssion
```

`es*` 的匹配结果(e, 零或多个 s)

```
expression
```

`es?` 的匹配结果(e, 零或一个 s)

```
expression
```

贪婪与懒惰

使用限定符进行匹配时，默认匹配尽可能多的字符。无论用 `.*` 还是 `.+` 匹配下列文本，都会匹配全部

```
expression
```

这种方式称为“贪婪模式”。在限定符之后加 `?` 则匹配尽可能少的字符，称为“懒惰模式”^[49]

例如，使用贪婪模式 `a.+b` 匹配：

```
aaabab
```

使用懒惰模式 `a.+?b` 匹配：

```
aaabab
```

^[49] `.+` 匹配一个或多个任意字符，在贪婪模式中，它匹配尽可能多的字符；而懒惰模式中 `(.+?)`，则只匹配一个字符；`.{3,5}` 在贪婪模式中尽可能匹配5个字符，在懒惰模式中 `(.{3,5}?)` 只匹配3个字符；`?` 和 `*` 这样可以匹配零次的限定符，在懒惰模式下不匹配任何字符 `(.*?)`、`(.??)`

