

# PYTHON

## 编程基础

# re模块使用-匹配对象

## re模块使用-匹配对象

使用前面介绍的match函数和search函数，匹配成功时都会返回一个Match对象，匹配失败时则返回None。这里我们看一下如何操作返回的Match对象。

**例**

▶ 根据Match对象判断是否匹配成功。

```
1  import re
2  result1=re.search(r'python', '我喜欢学习Python ! ', re.I)
3  if result1: #判断是否匹配成功
4      print('result1:',result1) #匹配成功则输出返回的Match对象
5  result2=re.match(r'python', '我喜欢学习Python ! ', re.I)
6  if result2: #判断是否匹配成功
7      print('result2:',result2) #匹配成功则输出返回的Match对象
```

## re模块使用-匹配对象

使用前面介绍的match函数和search函数，匹配成功时都会返回一个Match对象，匹配失败时则返回None。这里我们看一下如何操作返回的Match对象。

**例**

▶ 根据Match对象判断是否匹配成功。

```
result1: <re.Match object; span=(5, 11), match='Python'>
```

### 提示

将Match对象作为判断条件时，其永远返回True；而None则返回False。因此，通过“if result1:”和“if result2:”即可判断前面的匹配是否成功。

# re模块使用-匹配对象

Match对象提供了多种方法，这里我们仅学习group、groups、start和end这几种方法的使用。

Match对象方法	描述
group([group1,...])	根据传入的组号返回对应分组的匹配结果。如果传入一个组号，则返回一个字符串形式的匹配结果；如果传入多个组号，则返回一个由多个匹配结果字符串组成的元组。如果传入0或不传入参数，则返回的是与正则表达式匹配的整个字符串
groups()	返回一个由所有分组的匹配结果字符串组成的元组
start(group=0)	返回指定分组的匹配结果字符串在原字符串中的起始位置；如果group值为0（默认值），则返回与正则表达式匹配的整个字符串在原字符串中的起始位置
end(group=0)	返回指定分组的匹配结果字符串在原字符串中的结束位置；如果group值为0（默认值），则返回与正则表达式匹配的整个字符串在原字符串中的结束位置

# re模块使用-匹配对象

**例**

► Match对象中的方法使用示例。

```
1 import re
2 str="sno:#1810101#,name:#李晓明#,age:#19#,major:#计算机#
3 sno:#1810102#,name:#马红#,age:#20#,major:#数学#"
4 rlt=re.search(r'name:#[\s\S]*?#[\s\S]*?major:#[\s\S]*?#',str, re.I)
5 if rlt: #判断是否有匹配结果
6     print('匹配到的整个字符串:', rlt.group())
```

# re模块使用-匹配对象

**例**

► Match对象中的方法使用示例。

```
7      print('name:%s, startpos:%d, endpos:%d'%(r1t.group(1),r1t.start(1), r1t.end(1)))
8      print('major:%s, startpos:%d, endpos:%d'%(r1t.group(2),r1t.start(2), r1t.end(2)))
9      print('所有分组匹配结果:', r1t.groups())
10     else:
11         print('未找到匹配信息')
```

匹配到的整个字符串: name:#李晓明#,age:#19#,major:#计算机#

name:李晓明, startpos:20, endpos:23

major:计算机, startpos:41, endpos:44

所有分组匹配结果: ('李晓明', '计算机')