

正则表达式之零宽断言

零宽断言:用于查找特定内容之前或之后的内容，但并不包括特定内容本身。

断言用来声明一个应该为真的事实。正则表达式中只有当断言为真时才会继续进行匹配。

简单来说就是匹配一个位置，这个位置满足某个正则，但不纳入结果的，所以叫“零宽断言”，而且这个位置的前面或后面需要满足某种正则。

分类	代码/语法	说明
零宽度正预测先行断言	(?=exp)	匹配exp前面的位置
零宽度正回顾后发断言	(?<=exp)	匹配exp后面的位置
零宽度负预测先行断言	(?!exp)	匹配后面跟的不是exp的位置
零宽度负回顾后发断言	(?<!exp)	匹配前面不是exp的位置

下面结合例子来理解下什么是零宽断言。

零宽断言

正则表达式: (?<=<div>).* (?=</div>)

匹配字符串: <div>www.axiaowei.cn</div>

匹配结果: www.axiaowei.cn

```
1 # coding: utf-8
2
3 import re
4
5 html = '<div>www.axiaowei.cn</div>'
6 print(re.compile(r'(?<=<div>).* (?=</div>)').findall(html))
7
8
9
```

Run: 零宽断言测试

"D:\Python Project\web2\venv\Scripts\python3.exe" "D:/Python Project/web2/零宽断言测试.py"

[' www.axiaowei.cn']

Process finished with exit code 0

正则表达式: (?<=<div>).*

匹配字符串: <div>www.axiaowei.cn</div>

匹配结果: www.axiaowei.cn</div>

The screenshot shows a Python IDE with two tabs: 'calc.py' and '零宽断言测试.py'. The active tab contains the following Python code:

```
1 # coding: utf-8
2
3 import re
4
5 html = '<div>www.axiaowei.cn</div>'
6 print(re.compile(r'(?<=div>).*').findall(html))
7
8
9
```

Below the code editor is a 'Run' button and a console window. The console output shows the command executed and the result:

```
"D:\Python Project\web2\venv\Scripts\python3.exe" "D:/Python Project/web2/零宽断言测试.py"
['www.axiaowei.cn</div>']
```

The process finished with exit code 0.

正则表达式: `(?=www).*(?<=cn)`

匹配字符串: `<div>www.axiaowei.cn</div>`

匹配结果: `www.axiaowei.cn`

```
1 # coding: utf-8
2
3 import re
4
5 html = '<div>www.axiaowei.cn</div>'
6 print(re.compile(r'(?=www).*?(?<=cn)').findall(html))
7
8
```

Run: 零宽断言测试

"D:\Python Project\web2\venv\Scripts\python3.exe" "D:/Python Project/web2" ['www.axiaowei.cn']

Process finished with exit code 0

负向零宽断言

正则表达式: `\d{3}(?! \d)` -> 匹配三位数字的后面不能是数字

匹配字符串: 235456123

匹配结果: 123

```
1 # coding: utf-8
2
3 import re
4
5 s = '235456123'
6 print(re.compile(r'\d{3}(?!\\d)').findall(s))
7
8
9
```

Run: 零宽断言测试 ×

“D:\Python Project\web2\venv\Scripts\python3.exe” “D:/Python Project/web2/

['123']

正则表达式: `(?<![a-z])\d{7}` -> 匹配前面不是小写字母的七位数字
匹配字符串: 1111111d2222222d3333333s4444444
匹配结果: 1111111, 4444444

```
1 # coding: utf-8
2
3 import re
4
5 s = '111111d222222d333333S444444'
6 print(re.compile(r'(?![a-z])\d{7}').findall(s))
7
8
9
```

Run: 零宽断言测试

"D:\Python Project\web2\venv\Scripts\python3.exe" "D:/Python Project/web2/零宽断言测试.py"

['1111111', '4444444']

Process finished with exit code 0

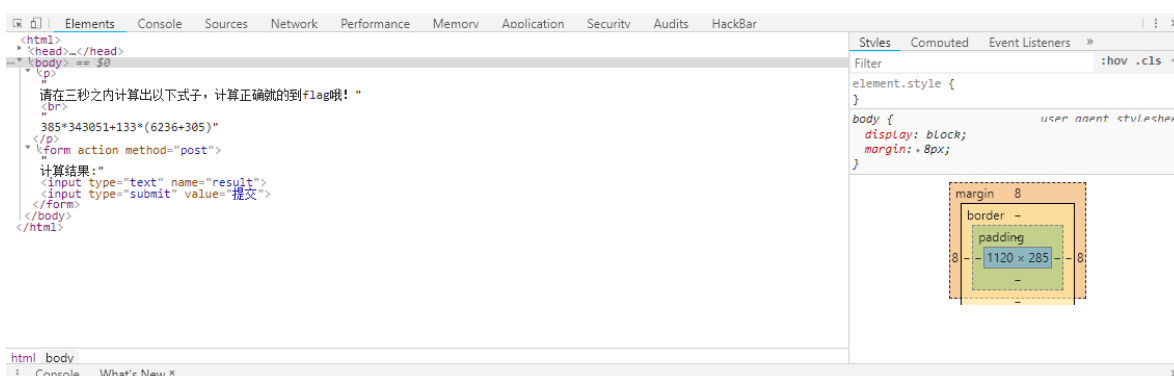
这里举一个常见的简单计算题，我们利用零宽断言进行匹配。

三秒中算出式子（每次刷新式子也会变化），想必能算出来输入提交也超过三秒了吧，直接写python

请在三秒之内计算出以下式子，计算正确就到flag哦！

$385*343051+133*(6236+305)$

计算结果:



```
# coding: utf-8

import re
import requests

url = 'http://123.206.31.85:10002/'
convert = requests.session()
html = convert.get(url).text
reg = re.compile(r'.(?=</p>)').findall(html) # 零宽断言匹配
```

```

link = ''
for i in reg:
    link += i
data = eval(link)

post = convert.post(url, {'result': data})
print('三秒之内计算出以下式子: \n', link, '=', data)
print(post.text)

```

The screenshot shows a Python IDE with a file named `calc.py`. The script performs the following steps:

- Imports `re` and `requests`.
- Defines a URL: `url = 'http://123.206.31.85:10002/'`.
- Creates a `requests` session and gets the HTML content of the URL.
- Prints the HTML content.
- Uses three regular expressions to extract data from the HTML:
 - `reg = re.compile(r'd[\\(\\+*\\)]*')` to find all digits and operators (positive lookahead).
 - `reg = re.compile(r'(?=\\d\\.*(?<=\\))')` to find decimal points (zero-width lookahead).
 - `reg = re.compile(r'.*(?= /p>')` to find the end of the paragraph (zero-width lookahead).
- Concatenates the extracted parts into a string `link`.
- Evaluates the string `link` using `eval()` to get the result `data`.
- Sends a POST request to the same URL with `data` as the 'result' parameter.
- Prints the original expression and the result.
- Prints the text received from the POST request.

The Run console shows the output of the script:

```

"D:\Python Project\web2\venv\Scripts\python3.exe" "D:/Python Project/web2/calc.py"
三秒之内计算出以下式子:
888*168186+298*(3819+2061) = 151101408
<p>flag{b37d6bdd7bb132c7c7f6072cd318697c}</p>
Process finished with exit code 0

```

参考资料:

<http://deerchao.net/tutorials/regex/regex.htm>