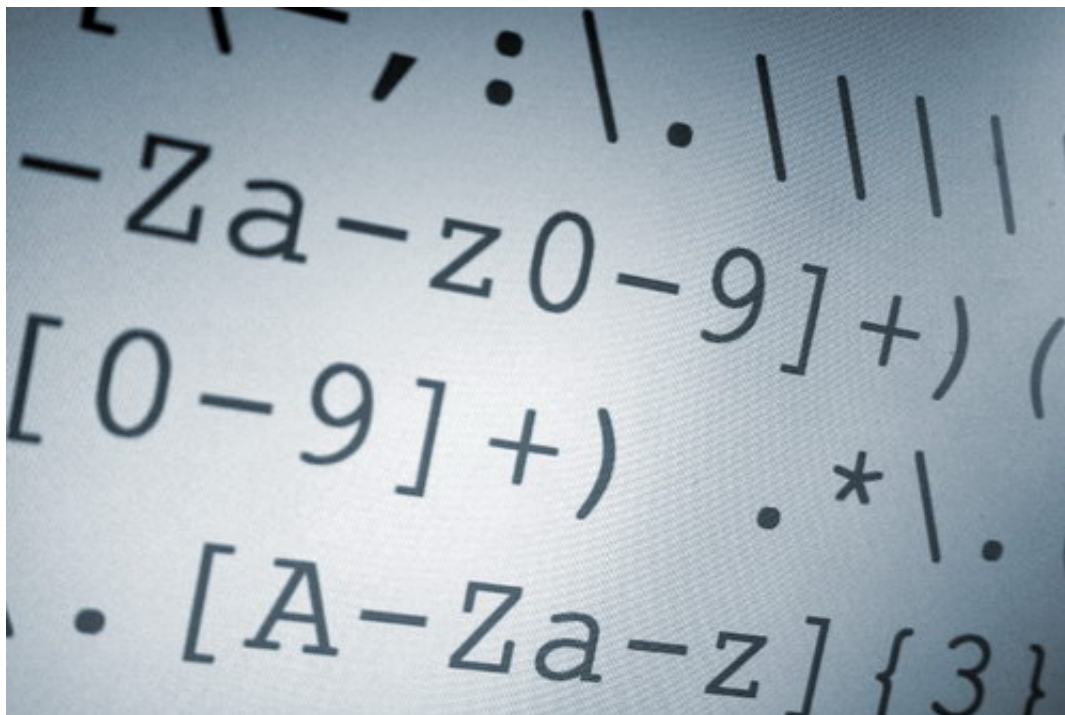


## Scala正则表达式替换

在之前的博文 [《Scala正则表达式》](#)

我简单地介绍了如何在Scala中使用正则表达式来匹配一些我们需要的内容。本篇文章将接着此文继续简单介绍如何使用Scala来匹配出我们需要的字符串，然后使用某种规则来替换匹配出来的字符串。



如果想及时了

解Spark、Hadoop或者Hbase相关的文章，欢迎关注微信公共帐号：iteblog\_hadoop

### 简单正则表达式替换

简单的正则表达式替换可以直接使用字符串的 `replaceAll` 函数，如下：

```
scala> val str = "abcdef"  
str: String = abcdef
```

```
scala> val regex = "[af]"  
regex: String = [af]
```

```
scala> val address = str.replaceAll(regex, "x")  
address: String = xbcdex
```

上面的正则表达式含义是匹配字符串中的 a 和 f 两个字符串，只要匹配到一个就算匹配成功，然后使用字符串 x 来替换匹配到的字符，所以最后的结果为 xbcdex。replaceAll 函数是Java String 类中自带的。Scala中当然也存在类似于这样的函数，这个函数就是 replaceAllIn，但是这个函数并不是 String 类中的，而是属于 scala.util.matching.Regex 类，此函数使用如下：

```
scala> val regex = "[0-9]".r
regex: scala.util.matching.Regex = [0-9]
```

```
scala> val newString = regex.replaceAllIn("123iteblog", "x")
newString: String = xxxiteblog
```

效果和上面的类似。

## 字符串分组替换

但是有时候仅仅是简单地使用相同的字符串去替换所有匹配到的字符串远远不能满足我们的需求。我们很可能想对匹配到的字符串添加一些东西，比如对 a b c d e 字符串，我们想将里面所有字符串 a 添加一个单引号，也就是替换之后的字符串为 'a' b c d e，这个该如何实现呢？很简单，使用分组实现，如下：

```
scala> """"([abc])"""".r.replaceAllIn("a b c d e", """"$1""")
res13: String = 'a' 'b' 'c' d e
```

里面的 \$1 就是分组的含义。更复杂的例子：

```
scala> """"hello { \W" world \W" } \Wn"""".replaceAll("""([\{\}]|\W\W["\n])""", ""$1""")
res12: String = hello '{ '\W'" world '\W'" }' '\W'n'
```

上面实例将字符串中所有的 {、}、\W"、以及 \Wn 都加上了单引号。

如果我们仅仅只需要替换第一个匹配到的字符，可以使用 replaceFirstIn，如下：

```
scala> """"([abc])"""".r.replaceFirstIn("a b c d e a", """"$1""")
res14: String = 'a' b c d e a
```

可以看到上面仅仅只有第一个字符 a 被替换。

## 正则表达式条件替换

我们还可能只替换符合某些条件的字符串，比如我只想使用某个字符串去替换原字符串匹配到字符，并且只对这个字符串前50个字符进行处理，超过50之后的字符即使匹配到也不替换，可以使用 `replaceSomeIn` 实现，其官方文档解释：

```
Replaces some of the matches using a replacer function that returns an
[[scala.Option]].
The replacer function takes a [[scala.util.matching.Regex.Match]] so that extra
information can be btained from the match.
```

`replaceSomeIn`函数接收一个函数，这个函数的输入为 `scala.util.matching.Regex.Match`，输出为 `scala.Option`，最后的结果是仅对返回为 `Some` 的数据进行替换，其他的不处理。下面就是替换某个字符串的前50个满足条件的字符，超过50的字符串即使匹配到也不替换，如下：

```
scala> val expr = "e".r
expr: scala.util.matching.Regex = e
```

```
scala> val quote = """"I don't like to commit myself about heaven and hell - you see, I have friends in both places. Mark Twain""""
quote: String = I don't like to commit myself about heaven and hell - you see, I have friends in both places. Mark Twain
```

```
scala> expr.replaceSomeIn(quote, m => if(m.start > 50) None else Some("-"))
res11: String = I don't lik- to commit mys-lf about h-av-n and h-ll - you see, I have friends in both places. Mark Twain
```

**本博客文章除特别声明，全部都是原创！**

**原创文章版权归过往记忆大数据（[过往记忆](#)）所有，未经许可不得转载。**

**本文链接: [【】](#) ( )**