TOUR OF SCALA

REGULAR EXPRESSION PATTERNS

Regular expressions are strings which can be used to find patterns (or lack thereof) in data. Any string can be converted to a regular expression using the `.r` method.

```
import scala.util.matching.Regex

val numberPattern: Regex = "[0-9]".r

numberPattern.findFirstMatchIn("awesomepassword") match {
  case Some(_) => println("Password OK")
  case None => println("Password must contain a number")
}
```

In the above example, the `numberPattern` is a `Regex` (regular expression) which we use to make sure a password contains a number.

You can also search for groups of regular expressions using parentheses.

```
import scala.util.matching.Regex

val keyValPattern: Regex = "([0-9a-zA-Z- ]+): ([0-9a-zA-Z-#()/. ]+)".r

val input: String =
  """background-color: #A03300;
    |background-image: url(img/header100.png);
    |background-position: top center;
    |background-repeat: repeat-x;
    |background-size: 2160px 108px;
    |margin: 0;
    |height: 108px;
    |width: 100%;""".stripMargin

for (patternMatch <- keyValPattern.findAllMatchIn(input))
  println(s"key: ${patternMatch.group(1)} value: ${patternMatch.group(2)}")
```

Here we parse out the keys and values of a String. Each match has a group of sub-matches. Here is the output:

```
key: background-color value: #A03300
key: background-image value: url(img/header100.png)
key: background-position value: top center
key: background-repeat value: repeat-x
key: background-size value: 2160px 108px
key: margin value: 0
key: height value: 108px
key: width value: 100
```

← previous next →

Contributors to this page:


ckipp01

mlachkar

ashawley

changlinli

heathermiller


Scala