regexs

February 16, 2017

```
In [14]: "Astrium limited".replaceAll("Limited", "")

Out[14]: Astrium limited

In [4]: // if we want to work it on either upper or lower case, we can to transform
        "Astrium Limited".toLowerCase.replaceAll("limited", "")

Out[4]: "astrium "

In [6]: // or, if we don't want to loose the upper case information ('A' from 'Aust
        // we can specify (?i) in the regex to match on either upper or lower case.
        "Astrium Limited".replaceAll("(?i)limited", "")

Out[6]: "Astrium "

In [7]: // this does not take into account entire words, so Limited is also removed
        "AstriumLimited".replaceAll("(?i)limited", "")

Out[7]: Astrium

In [1]: // we can tell the regex to only look at entire words using \b ('limited' a
        "AstriumLimited".replaceAll("(?i)\\blimited\\b", "")

Out[1]: AstriumLimited

In [2]: // it removes the text only if 'limited' is an entire word.
        "Astrium Limited".replaceAll("(?i)\\blimited\\b", "")

Out[2]: "Astrium "

In [3]: // remove the 'limited' word, only if it is at the end of the string.
        // it does not match here:
        "United Kingdom Limited Intellectual".replaceAll("(?i)\\blimited$", "")

Out[3]: United Kingdom Limited Intellectual

In [4]: // it matches here:
        "Astrium Limited".replaceAll("(?i)limited$", "")

Out[4]: "Astrium "
```

```
In [12]:  // remove all spaces before the word.
          "Astrium Limited".replaceAll("(?i) *\\blimited$", "")

Out[12]:  Astrium

In [8]:   // remove all of these words.
          "Nokia limited and corporation".replaceAll("(?i)\\b(Limited|Co\\.|Corporati

Out[8]:   "Nokia  and "

In [11]:  // remove all numbers (one or more digits) by '-'
          "hello 1234 world".replaceAll("[0-9]+", "-")

Out[11]:  hello - world

In [18]:  // remove all numbers, only if they are preceded by a €
          "hello €123 world".replaceAll("(?<=€)[0-9]+", "-")

Out[18]:  hello €- world
```

Regular expressions are quite powerful; it's worth reading about them: - https://docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html - http://www.regular-expressions.info/unicode.html - https://regex101.com

```
In [ ]:
```