How regular expressions can save your day!

- Roy Savio Athayde

Hello!

I am Savio

I am here because I love to ~~give presentations~~.
use shortcuts.

You can find me at roysavio@gmail.com

Is this session for geeks?

No! Because geeks already know a lot about regular expressions (or regexps).

Who is the session for?

- It is for technical writers like you and me, who can do with shortcuts for mundane tasks, because they already perform so many anyway.

- It is to tell you about the **possibilities** with regular expressions and how powerful they can be.

Why regular expressions?

- Gradually content is increasingly being stored as XML.
  (Used to be DOC files, FM files, HTML files, and so on)
- A little reg-exp knowledge can save you hours of work.
- Free text editors such as Notepad Plus support find/replace using regular expressions.

A regular expression (regex or regexp) is a special text string for describing a search pattern. You can think of regular expressions as wildcards on ***steroids****.*

- www.regular-expressions.info

“

A example of how regular exps can help you

My company just got acquired. So I need to change all the documentation download URLs.

Old: www.OldCompany.com/docs/WritingForDummies.pdf

New: www.NewCompany.com/docs/WritingForDummies_**OldCompany**.pdf

Search for:

(www\.)(OldCompany)(\.com\/docs\/)(WritingForDummies)(\.pdf)

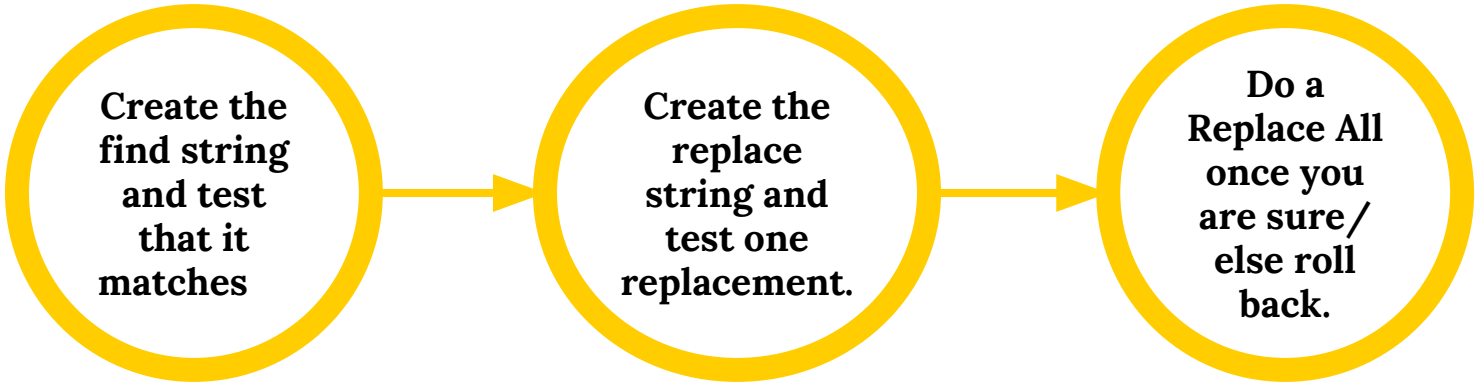
Replace with:

```
\1NewCompany\3\4_\2\5/
```

Tip

We can temporarily store parts of a string if we need to later use the parts in the replace operation. These are called **numbered backreferences**.

So here's how you do it



Basic rules for syntax

Character	Explanation
^	The beginning of a line.
$	The end of a line.
?	The ? (question mark) matches when the preceding character occurs 0 or 1 times only. For example, **colou?r** will find both color (u is found 0 times) and colour (u is found 1 time).

Character	Explanation
*	The * (asterisk) matches when the preceding character occurs 0 or more times, for example, **tre*** will find tree (e is found 2 times) and tread (e is found 1 time) and trick (e is found 0 times).
+	The + (plus) matches when the preceding character occurs 1 or more times, for example, **tre+** will find tree (e is found 2 times) and tread (e is found 1 time) but NOT trick (0 times).
|	The vertical bar or pipe is called alternation and means find the left hand OR right values, for example, **gr(a|e)y** will find 'gray' or 'grey.'

To delve a little more...

- You will need to insert a left backslash or an escape character to escape metacharacters such as a dot (.) or another backslash (\) if these are part of the string you are searching for.
- You can match a **range** of characters (and/or perform case insensitive matches).
- You can also specify occurrence.

Any drawbacks?

- One major drawback is that regular expressions might not be very easy to read. Complex expressions can be very confusing.
- A sample:

(^[2][5][0-5].|^[2][0-4][0-9].|^[1][0-9][0-9].|^[0-9][0-9].|^[0- 9].)([2][0-5][0-5].|[2][0-4][0-9].|[1][0-9][0-9].|[0-9][0-9].|[0- 9].)([2][0-5][0-5].|[2][0-4][0-9].|[1][0-9][0-9].|[0-9][0-9].|[0- 9].)([2][0-5][0-5]|[2][0-4][0-9]|[1][0-9][0-9]|[0-9][0-9]|[0-9])$}

This guess is to match IP addresses.

Any guesses what this is for?

Back to the first scenario

My company just got acquired. So I need to change all the documentation download URLs.

Old: www.OldCompany.com/docs/WritingForDummies.pdf

New: www.NewCompany.com/docs/WritingForDummies_**OldCompany**.pdf

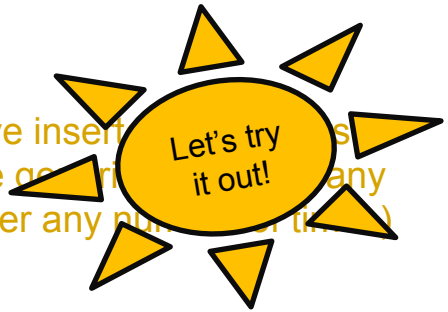
Search for:

(www\.)(OldCompany)(\.com\/docs\/)(.*)(\.pdf)

Replace with:

\1NewCompany\3\4_\2\5/

We have inserted ... indicate ... character any ...



A few more examples....

Removing blank lines

My XML has many blank lines making readability difficult. I want to remove all blank lines in my XML.

I could do a search for \n (or \r) which indicate newlines and carriage returns, and replace them with single newlines and carriage returns.

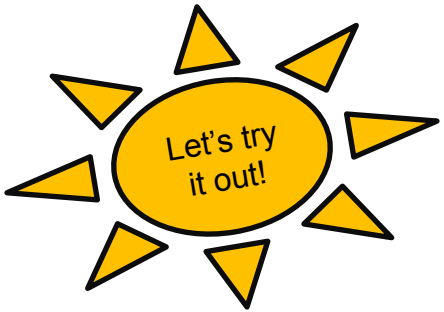
Search for:

`\r\n\r\n`

Replace with:

`\r\n`

Must be run multiple times



Swapping element order in an XML file

My XML has <firstname>Donald</firstname><lastname>Duck</lastname> and I now need to change those to lastname, firstname.

I could store the firstname and lastname as I showed earlier (backreferences) and swap them in the replace operation.

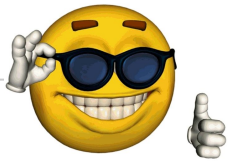
Somebody may be smiling in the audience and wondering why not use XPath. Try a regexp if you do not know XPath and XSL tricks.

Swapping the day and month in the date

My company has begun to ship the manuals to Asia. I need to change mm/dd/yyyy to dd/mm/yyyy in APAC manuals.

You now know the tricks for backreferences and how to swap them.

You can even try **extracting** email addresses so that you can verify if all the addresses are valid.



A few useful tips

- A good online resource: http://www.zytrax.com/tech/web/regex.htm
- Always test to ensure your software supports what you are trying to use.

With XML

- Try removing all newlines first.
- Use a logical break like an end tag and again insert new lines (find: <\/para> replace: <\/para>**\n**).

- You can search online for "**regular expression tester**" and try out your find and replace strings.

A picture is worth a thousand words

A regexp is worth a thousand manual operations saved.

– The lazy writer

Thanks!

Any questions ?

You can find me at

- roysavio@gmail.com
- Oops! I don’t use Twitter. ☺