

Regular Expressions

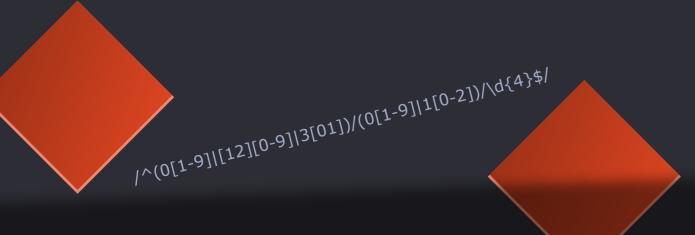
by Dylan TAILDEMAN



C'est quoi une RegEx ?

Chaine de caractères décrivant un ensemble de correspondances possibles.





```
Sh (Terminal)
echo «Becode est en Belgique!» > file.txt
grep --color -E 'Be[a-z]+' file.txt

Becode est en Belgique!
```

```
Python
import re

regex = 'Be[a-z]+'
text = "Becode est en Belgique!"
result = re.findall(regex, text)

print(result)
# ['Becode', 'Belgique']
```

```
Javascript
let regex = /Be[a-z]+/g;
let text = "Becode est en Belgique!";
let result = text.match(regex);

console.log(result);
/* ["Becode", "Belgique"] */
```

```
PHP

<?php
$regex = '/Be[a-z]+/';
$text = "Becode est en Belgique";
preg_match_all($regex, $text, $matches);

print_r($matches);
// Array ([0] => Becode [1] => Belgique)
?>
```



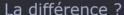
Deux types de RegEx ?



BRE (Basic Regular Expression)

VS

ERE (Extended Regular Expression)



Certains symboles littéraux et spéciaux seront inversés. Il faudra les échapper en fonction de nos besoins.

Ainsi, dans une BRE:

+ => + littéral

\+ => quantificateur

Alors que dans une ERE:

+ => quantificateur

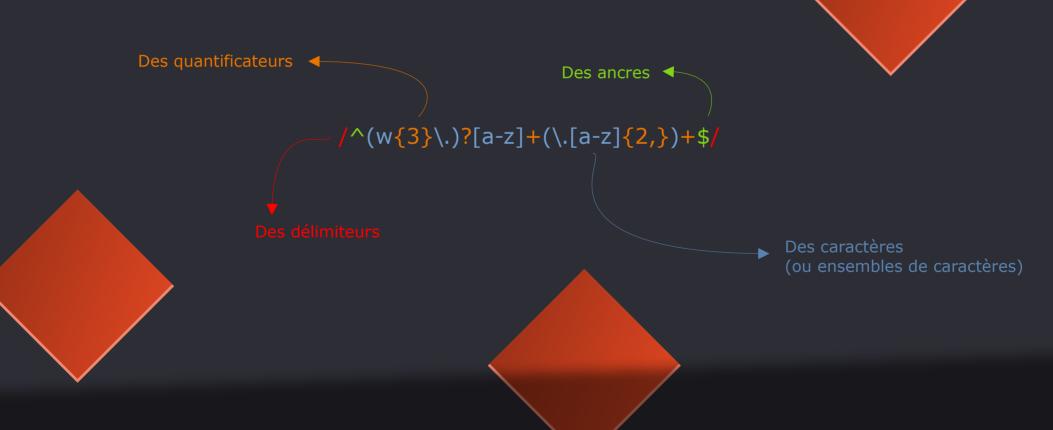
\+ => + littéral



BRE: sed, grep, vi

ERE: egrep, grep -E, awk





Les délimiteurs

- définissent le début et la fin de la RegEx,
- peuvent changer d'un langage de programmation à un autre,
- les plus courants sont l'apostrophe (`), le guillemet ('') et le slash (/).

/[A-Za-z0-9\.,;:!?()"'%\-]+/

'Be[a-z]+'

Les ancres

- indiquent à la RegEx que le motif recherché se trouvent en début (^) ou en fin (\$) de ligne,
- attention à ne pas confondre le symbole d'exclusion [^a-f] avec l'ancre de début de ligne!

/^(0[1-9]|[12][0-9]|3[01])/(0[1-9]|1[0-2])/\d{4}\$/

 $/^(w{3}\.)?[a-z]+(\.[a-z]{2,})+$/$

Les caractères (ou ensembles de caractères)

mot	ver[ts]	m[^a]ts	[a-z]	[a-zA-Z0-9]		voiture moto
mot motif marmotte	ver de terre vers l'infini feu vert	mets mats mots	a P 9	A a 0	Tous les caractères	voiture moto bus

\d	\w	\t	\smot\s
Tous les chiffres	Equivaut à [a-zA-Z0-9_]	Tabulation	mot motif marmotte

Les quantificateurs

?	*	+	{3}	{3,}	{,8}	{3,6}
0 ou 1	0 ou plus	1 ou plus	Exactement 3	Minimum 3	Maximum 8	Entre 3 et 6

Le groupement

/USA|B/ /US(A|B)/

La référence

Il est possible de faire référence à un groupe capturé précédément à l'aide de \1, \2, etc...

/(A|E)Y\1/

=> AYA, EYE x> AYE, EYA

La capture

Les parenthèses simples servent à la fois à grouper et à capturer. Ainsi, selon l'exemple précédent:

La RegEx /US(A|B)/ appliquée sur le texte «USA» renverra:

Une correspondance: 'USA' Un groupe capturé: 'A'

Si on souhaite grouper SANS capturer, on placera alors « ?: » au début de notre groupe:

/US(?:A|B)/

(Cette RegEx ne renverra alors qu'une correspondance.)

Exercices!

/#[0-9A-Fa-f]{6}/

/[a-zA-Z0-9_\.-]+@([a-zA-Z0-9_-]+(?:\.[a-zA-Z0-9_-]{2,})+)/

/<([^>\/]+)>(.*)<\/\1>/

#F841E2 #V84G14 #FFFFFF taildeman_dylan@outlook.com Laurent de la Clergerie james.bond007@agent.gov.uk charleroi@becode.org <html>coucou</html> <h1></h1> <h2>Titre raté</h3>



Merci à tous!



 $(\)((?![*+?])(?:[^\r\n\[/\])\)((?:g(?:im?|mi?)?|i(?:gm?|mg?)?|m(?:gi?|ig?)?)?)/$