

## Les expressions régulières :

Les expressions régulières, ou plus communément regex (contraction de *regular expression*) permettent de représenter des modèles de chaînes de caractères.

Symbole	Description
.	n'importe quel caractère
[...]	tout caractère parmi ceux énumérés
[^...]	tout caractère excepté ceux énumérés
^	la position en début de ligne
\$	la position en fin de ligne
\<	la position en début de mot
\>	la position en fin de mot
Exp1   exp2	reconnaît expression 1 ou expression 2
?	0 ou 1 occurrence du caractère ou de l'expression régulière précédée
*	0 ou <i>n</i> occurrences du caractère ou de l'expression régulière précédée
[liste]	Indiquer une liste de caractères
[^liste]	Un caractère qui ne figure pas dans la liste
expr{n}	Exactement <i>n</i> occurrences de l'expression précédant les accolades
Expr{n,m}	Entre <i>n</i> et <i>m</i> occurrences de l'expression précédant les accolades
[[ :alnum:]]	Un caractère alphanumérique
[[ :alpha:]]	Un caractère alphabétique
[[ :digit:]]	Un digit entre 0 et 9
[[ :blank:]]	Un espace ou une tabulation

## La commande sed

**Sed** permet de modifier ou de supprimer une partie d'une chaîne de caractères dans un flux passé en entrée.

### Syntaxe :

```
sed [-n] [-e commandes] [-f fichier de commandes] [fichier]
```

### Principe de fonctionnement

- **sed** fonctionne en mode «flux», c'est-à-dire que le flux en entrée (fichier ou autre) est traité ligne par ligne.

- Pour chaque ligne, on applique la/les commande(s) (si cela est possible) puis on affiche sur la sortie standard la ligne modifiée ou non.

### Options & commandes :

Option	Signification
<b>-e</b> ' motif1 commande1 ; motif2 commande2 ;..'	permet de spécifier les commandes à appliquer sur le fichier. Cette option est utile lorsque vous appliquez plusieurs commandes.
<b>-f</b> fichier_commandes	les commandes sont lues à partir d'un fichier.
<b>-i</b>	applique la commande directement sur le fichier passé en entrée. Autrement dit, les modifications seront effectuées directement sur le fichier d'entrée.
<b>-n</b>	écrit seulement les lignes spécifiées (par l'option /p) sur la sortie standard

### Comment écrire une commande sed :

*Sed 's/<motif>/<substitut>/flags' [nom de fichier]*

#### flags :

- **g** : global, c'est-à-dire on remplace partout.
- **p** : imprime la ligne (utile avec l'option -n).
- **< n >** : on substitue seulement la n<sup>ème</sup> occurrence du motif/regex dans le fichier.

```
you@you-VirtualBox: ~
you@you-VirtualBox:~$ echo "salut tous le monde"|sed "s/ /_/g"
salut_tous_le_monde
you@you-VirtualBox:~$
```

Pour indiquer un champ bien particulier dans un motif en l'entoure avec **\(expression\)**

Ensuite on peut faire référence au champ indiqué dans la partie substitution comme suit : **\ordre du champ**

#### Exemples :

```
you@you-VirtualBox: ~
you@you-VirtualBox:~$ echo "11-12-2018"|sed "s/\(.*\)-\(.*\)-\(.*\)/jour=\1 mois=\2 annee=\3/g"
jour=11 mois=12 annee=2018
you@you-VirtualBox:~$

you@you-VirtualBox:~$ echo "salut mr Marc Dark"|sed "s/\([A-Z].*\) \([A-Z].*\)/\2 \1/g"
salut mr Dark Marc
you@you-VirtualBox:~$
```