

**grep, egrep, fgrep, rgrep** – afficher les lignes correspondant à un motif donné

### Syntaxe

**grep** [*OPTIONS*] *MOTIF* [*FICHIER...*]

**grep** [*OPTIONS*] [**-e** *MOTIF* | **-f** *FICHIER*] [*FICHIER...*]

### Options

<b>-E</b>	<code>--extended-regexp</code>	Interpréter le <i>MOTIF</i> comme une expression rationnelle étendue
<b>Contrôle de concordance</b>		
<b>-i</b>	<code>--ignore-case</code>	Ignorer la casse aussi bien dans le <i>MOTIF</i> que dans les fichiers
<b>-w</b>	<code>--word-regexp</code>	Ne sélectionner que les lignes contenant des concordances formant des mots complets.
<b>Contrôle général de l'affichage</b>		
<b>-c</b>	<code>--count</code>	Ne pas afficher les résultats normaux. À la place, afficher un décompte des lignes correspondant au motif pour chaque fichier.
<b>-l</b>	<code>--files-with-matches</code>	Ne pas afficher les résultats normaux. À la place, indiquer le nom des fichiers pour lesquels des résultats auraient été affichés.
<b>-o</b>	<code>--only-matching</code>	N'afficher que les parties qui correspondent (et ne sont pas vides) des lignes qui correspondent, chaque partie étant affichée sur une ligne séparée.
<b>-H</b>	<code>--with-filename</code>	Afficher le nom du fichier pour chaque correspondance.
<b>-h</b>	<code>--no-filename</code>	Ne pas afficher le nom des fichiers au début des lignes qui correspondent.
<b>Contrôle des lignes de contexte</b>		
<b>-A</b> <u>N</u>	<code>--after-context=<u>N</u></code>	Afficher les <u>N</u> lignes qui suivent celle contenant le motif.
<b>-B</b> <u>N</u>	<code>--before-context=<u>N</u></code>	Afficher les <u>N</u> lignes qui précèdent celle qui contient le motif.
<b>-C</b> <u>N</u> , <b>-<u>N</u></b>	<code>--context=<u>N</u></code>	Afficher <u>N</u> lignes de contexte.

## sort – trier les lignes de fichiers texte

### Syntaxe

**sort** [*OPTION*]... [*FICHIER*]...

### Options

<b>-b</b>	--ignore-leading-blanks	ignorer les blancs de tête
<b>-d</b>	--dictionary-order	ne prendre en compte que les blancs et les caractères alphanumériques
<b>-f</b>	--ignore-case	convertir les caractères minuscules en majuscules
<b>-n</b>	--numeric-sort	comparer selon la valeur numérique de la chaîne
<b>-r</b>	--reverse	inverser le résultat des comparaisons
<b>-k</b>	--key= <i>POS1</i> [, <i>POS2</i> ]	utiliser la clé de tri commençant à <i>POS1</i> (les positions sont comptées à partir de 1) et se terminant à <i>POS2</i> (la fin de la ligne par défaut)
<b>-o</b>	--output= <i>FICHIER</i>	écrire le résultat dans le <i>FICHIER</i> à la place de la sortie standard
<b>-t</b>	--field-separator= <i>SÉPARATEUR</i>	utiliser le <i>SÉPARATEUR</i> à la place d'une transition d'un caractère non blanc vers un caractère blanc
<b>-u</b>	--unique	Avec <b>-c</b> , vérifier que le classement est strict ; sans <b>-c</b> , n'afficher que la première entrée identique rencontrée

## uniq – signaler ou éliminer les lignes répétées

### Syntaxe

**uniq** [*OPTION*]... [*ENTRÉE* [*SORTIE*]]

NB. **uniq** ne détecte pas les lignes répétées, sauf si elles sont adjacentes ! C'est pourquoi il faut précéder cette commande avec **sort**

### Options

<b>-c</b>	--count	préfixer les lignes par le nombre d'occurrences
-----------	---------	---

# Expressions régulières

Classes de caractères					
.	n'importe quel caractère (sauf fin de ligne)	[ :punct: ]	caractère de ponctuation <sup>[...]</sup>		
[ ]	classe de caractères	[ :alnum: ]	caractère alphanumérique <sup>[...]</sup>		
[ ^ ]	négation de la classe de caractères	[ :digit: ]	chiffre décimal <sup>[...]</sup>		
[ a-z ]	un intervalle de caractères = [aAbBcC...zZ]	[ :xdigit: ]	chiffre hexadécimal <sup>[...]</sup>		
[ :space: ]	espace blanc ou séparateur de ligne ou de paragraphe <sup>[...]</sup>	[ :alpha: ]	caractère alphabétique <sup>[...]</sup>		
[ :blank: ]	espace blanc ou tabulation non séparateur de ligne ou de paragraphe <sup>[...]</sup>	[ :lower: ]	lettre minuscule <sup>[...]</sup>		
[ :graph: ]	caractère graphique visible <sup>[...]</sup>	[ :upper: ]	lettre capitale <sup>[...]</sup>		
Ancrage					
^	une chaîne vide au début de ligne	\$	une chaîne vide en fin de ligne		
Répétitions					
?	L'élément précédent est facultatif et peut être rencontré au plus une fois.	{ <i>n</i> }	L'élément précédent doit correspondre exactement <i>n</i> fois.		
*	L'élément précédent peut être rencontré zéro ou plusieurs fois.	{ <i>n</i> , }	L'élément précédent doit correspondre <i>n</i> fois ou plus.		
+	L'élément précédent peut être rencontré une ou plusieurs fois.	{ , <i>m</i> }	L'élément précédent doit correspondre au plus <i>m</i> fois.		
		{ <i>n</i> , <i>m</i> }	L'élément précédent doit correspondre au moins <i>n</i> fois, mais au plus <i>m</i> fois.		
Références arrières et sous-expressions					
( )	les parenthèses de groupement (delimiteur se sous-motif)				
\ <i>n</i>	la sous-chaîne déjà mise en correspondance avec la <i>n</i> -ième sous-expression entre parenthèses				
Séquences d'échappement					
\ <i>x</i>	les caractères ( , ) , [ , ] , . , * , ? , + , ^ ,   , \$ et \ doivent être référencés dans une séquence d'échappement s'ils doivent désigner littéralement le caractère correspondant				
\ t	tabulation horizontale	\ b	début / fin de mot	\ B	négation de la classe \ b
\ n	fin de ligne	\ d	nombres entiers <sup>-p</sup>	\ D	négation de la classe \ d <sup>-p</sup>
\ v	tabulation verticale	\ w	caractères alphanumériques	\ W	négation de la classe \ w
\ f	saut de page	\ s	caractères d'espacement	\ S	négation de la classe \ s
\ r	retour-chariot				