

# grep

linux commande



**grep** est un outil de ligne de commande pour la recherche de motifs dans un fichier ou une entrée standard ou des entrées de pipelines. Il est utilisé pour afficher les lignes d'un fichier qui correspondent à un modèle spécifié. Le résultat peut être filtré ou utilisé comme entrée pour d'autres commandes.

Techniquement, **grep** analyse le contenu d'un fichier ou d'une entrée standard et affiche les lignes qui correspondent à un modèle spécifié. Le modèle peut être une expression régulière, qui est une forme d'expression mathématique utilisée pour décrire des ensembles de chaînes de caractères. **Grep** utilise un algorithme de correspondance rapide pour trouver rapidement les lignes qui correspondent au modèle, ce qui le rend utile pour la recherche de fichiers, la filtration de sorties et la manipulation de données complexes.

## Rechercher une chaîne de caractères dans un fichier

```
grep "motif recherché" nom_du_fichier
```

## Rechercher une chaîne de caractères dans un fichier sans respect de la casse

L'option **-i** indique de rechercher de manière insensible à la casse.

Par défaut, la commande **grep** effectue une recherche respectant la casse, ce qui signifie qu'elle ne fera correspondre que les occurrences exactes de la chaîne de caractères spécifiée. Si vous recherchez "bonjour" avec **grep**, il ne trouvera que "bonjour" et pas "Bonjour", "BONJOUR", etc.

Si vous souhaitez effectuer une recherche insensible à la casse avec **grep**, vous devez utiliser l'option **-i**, comme suit :

```
grep -i bonjour *
```

## Rechercher une chaîne de caractères dans plusieurs fichiers

```
grep "motif recherché" *
```

## Afficher les lignes correspondantes avec le nom de fichier

```
grep -H "motif recherché" *
```

L'option **-H** indique à **grep** de inclure le nom de fichier pour chaque correspondance trouvée. La sortie sera la liste des lignes de chaque fichier qui correspondent à "motif recherché".

## Rechercher une chaîne de caractères en utilisant une expression régulière

```
grep -E "^[0-9]+$" nom_du_fichier
```

L'option **-E** (extended regexp) pour rechercher les lignes du fichier `nom_du_fichier` qui correspondent à l'expression régulière `^[0-9]+$`.

L'expression régulière `^[0-9]+$` signifie "une ou plusieurs chiffres consécutifs, débutant à la première colonne de la ligne et se terminant à la dernière colonne de la ligne". En d'autres termes, la commande `grep` affichera toutes les lignes du fichier `nom_du_fichier` qui ne contiennent que des chiffres.

## Afficher le nombre de lignes correspondantes

```
grep -c "motif recherché" nom_du_fichier
```

L'option **-c** de la commande `grep` indique de ne retourner que le nombre de lignes qui correspondent au motif spécifié, sans afficher les lignes elles-mêmes.

## Afficher les lignes précédentes et suivantes correspondantes

```
grep -A 2 -B 1 "motif recherché" nom_du_fichier
```

L'option **-A 2** indique à `grep` d'afficher 2 lignes supplémentaires après chaque ligne contenant le "motif recherché".

L'option **-B 1** indique à `grep` d'afficher 1 ligne supplémentaire avant chaque ligne contenant le "motif recherché".

Ainsi, pour chaque correspondance trouvée, `grep` affichera la ligne correspondante, suivie de 1 ligne avant et 2 lignes après.

## Afficher des fichiers sans commentaires



Les fichiers de configurations peuvent contenir beaucoup de ligne. Mais la plus part de ces lignes sont des commentaires. Il est utile de les supprimer à l'affichage quand on veut uniquement conserver les paramètres pris en comptes.

—

Afficher toutes les lignes du fichier `/etc/php/7.4/fpm/php.ini` qui ne commencent pas par un point-virgule (;), qui est utilisé pour commenter les lignes dans ce type de fichier de configuration. Cela signifie que seules les lignes actives (non commentées) seront affichées à la sortie.

```
grep ^[^\;] /etc/php/7.4/fpm/php.ini
```

—

Affichons les lignes qui ne commencent pas avec un `#` dans le fichier `/etc/postfix/main.cf`. Les crochets `[]` définissent une liste de caractères à inclure ou à exclure dans la recherche, et le `^` signifie le début d'une ligne. Le `\` est utilisé pour échapper certains caractères spéciaux. Ainsi, `^[^#]` signifie que nous cherchons des lignes qui commencent avec un caractère qui n'est pas un `#`. Les résultats de la commande afficheront uniquement les lignes du fichier `main.cf` qui ne commencent pas avec un `#` et seront donc actives et utilisées par Postfix.

```
grep ^[^#] /etc/postfix/main.cf
```

— *Jean Serien (jeanserien), CPT*

From:

<https://www.abonnel.fr/> - **notes informatique & technologie**

Permanent link:

<https://www.abonnel.fr/informatique/linux/commandes/grep>

Last update: **2023/09/22 18:07**

