

alias



En Bash, un **alias** est un raccourci ou une abréviation qui est utilisé pour remplacer une commande ou une séquence de commandes plus longue par un seul mot ou une expression.

Par exemple, si vous utilisez fréquemment la commande `ls -al`, vous pouvez créer un alias nommé `ll` pour cette commande en utilisant la syntaxe suivante dans le terminal :

```
alias ll='ls -al'
```

Maintenant, chaque fois que vous tapez `ll` dans le terminal, Bash exécutera automatiquement la commande `ls -al`.

Les alias peuvent être utiles pour gagner du temps et améliorer l'efficacité lorsque vous travaillez avec la ligne de commande en réduisant la quantité de texte que vous devez taper pour exécuter une commande particulière.

Vous pouvez enregistrer des alias dans le fichier `.bash_aliases` si ce dernier existe.

Dans certains systèmes, le fichier `.bash_aliases` n'existe pas par défaut, mais vous pouvez le créer manuellement dans votre répertoire d'utilisateur en utilisant la commande suivante :

```
touch ~/.bash_aliases
```

Ensuite, vous pouvez ajouter vos alias dans ce fichier en utilisant la même syntaxe que celle utilisée pour définir des alias dans le terminal. Par exemple :

```
alias ll='ls -al'
```

Une fois que vous avez ajouté vos alias dans le fichier `.bash_aliases`, vous devez exécuter la commande suivante pour les charger dans votre session Bash en cours :

```
source ~/.bashrc
```

Cette commande rechargera le fichier `.bashrc` qui charge normalement le fichier `.bash_aliases` dans votre session Bash.

Notez que le nom et l'emplacement des fichiers de configuration Bash peuvent varier selon le système. Par conséquent, il est possible que vous deviez ajuster les instructions ci-dessus en fonction de votre configuration spécifique.

Quelques exemples

Voici quelques exemples d'alias de commandes qui peuvent être utiles :

- `ll` pour `ls -al` : comme expliqué précédemment, cela affichera le contenu d'un répertoire en mode long, avec les permissions et les détails de propriétaire affichés.

```
alias ll='ls -al'
```

- `grep` avec couleur : pour ajouter de la couleur à la sortie de la commande `grep` pour améliorer la lisibilité, vous pouvez utiliser l'alias suivant :

```
alias grep='grep --color=auto'
```

- `mkdir -p` : si vous créez souvent des répertoires imbriqués, vous pouvez utiliser l'option `-p` pour créer des répertoires parents si nécessaire. Vous pouvez également créer un alias nommé `mkdirp` pour simplifier l'opération.

```
alias mkdirp='mkdir -p'
```

- `ps` avec tri inversé : si vous utilisez souvent la commande `ps` pour afficher les processus en cours d'exécution sur votre système, vous pouvez ajouter un alias nommé `pss` pour trier les résultats par ordre décroissant d'utilisation de la mémoire.

```
alias pss='ps aux --sort=-rss'
```

- du trié par taille : si vous voulez afficher les dossiers par taille décroissante, vous pouvez ajouter un alias nommé `dus` pour trier la sortie de la commande `du` par ordre décroissant de taille.

```
alias dus='du -hs * | sort -hr'
```

- `cp` avec confirmation : pour ajouter une confirmation avant de copier des fichiers, vous pouvez créer un alias nommé `cp` qui inclut l'option `-i` pour demander une confirmation avant d'écraser les fichiers existants.

```
alias cp='cp -i'
```

- `rm` avec confirmation : de même, pour ajouter une confirmation avant de supprimer des fichiers, vous pouvez créer un alias nommé `rm` qui inclut l'option `-i` pour demander une confirmation avant de supprimer chaque fichier.

```
alias rm='rm -i'
```

- `git` avec couleur : si vous utilisez Git pour gérer des projets de développement de logiciels, vous pouvez ajouter de la couleur à la sortie de la commande `git` pour améliorer la lisibilité.

```
alias git='git --color=auto'
```

- `cd` avec retour rapide : pour revenir rapidement au répertoire parent, vous pouvez créer un alias nommé `..` qui exécute la commande `cd ..` pour revenir au répertoire parent.

```
alias ..='cd ..'
```

- free avec unités de mémoire lisibles : si vous utilisez souvent la commande free pour surveiller l'utilisation de la mémoire, vous pouvez ajouter l'option -h pour afficher les tailles de mémoire dans des unités lisibles pour l'homme (comme "1.23 GiB" au lieu de "1289311232").

```
alias free='free -h'
```

- grep avec numéros de ligne : pour afficher les numéros de ligne dans la sortie de la commande grep, vous pouvez ajouter l'option -n à l'alias grep.

```
alias grep='grep -n'
```

- ls avec tri inversé : pour afficher les fichiers et répertoires triés par ordre inverse de date de modification, vous pouvez créer un alias nommé lsr.

```
alias lsr='ls -ltr'
```

- tar avec compression gzip : pour créer une archive tar compressée avec gzip, vous pouvez créer un alias nommé targz.

```
alias targz='tar -cvzf'
```

- df avec taille et espace libre en unités lisibles : pour afficher l'espace disque disponible sur les partitions de votre système dans des unités lisibles pour l'homme, vous pouvez créer un alias nommé dfh.

```
alias dfh='df -hT'
```

Ces exemples montrent comment vous pouvez personnaliser vos alias de commandes pour automatiser des tâches courantes, faciliter la navigation dans les répertoires et améliorer la lisibilité de la sortie de la commande.

From:

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

Permanent link:

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

Last update: **2023/02/19 08:03**

