

La commande de lecture Linux

lire La commande dans Bash lit une ligne de l'entrée standard et la divise en champs.

Exemples de commandes de lecture Linux

La commande `read` sous Linux est une commande intégrée utilisée dans les scripts shell et les lignes de commande pour lire une ligne d'entrée à partir de l'entrée standard (stdin) et l'attribuer à une variable. Voici quelques exemples d'utilisation de la commande `read` :

Basic Usage:

```
read varname
echo "You entered: $varname"
```

This script waits for the user to enter some text, which is then stored in the variable `varname`. When the user presses Enter, the script displays the entered text.

Reading Multiple Values:

```
echo "Enter two values:"
read var1 var2
echo "You entered: $var1 and $var2"
```

This allows the user to enter two separate values, which are stored in `var1` and `var2`. The values should be separated by space.

Silent Input (Useful for Passwords):

```
read -s -p "Enter your password: " password
echo
echo "Password entered."
```

The `-s` option makes `read` silent, meaning it does not echo the input back to the terminal. This is useful for sensitive information like passwords. The `-p` option allows you to display a prompt.

Reading a Whole Line Including Spaces:

```
IFS= read -r line
echo "You entered: $line"
```

By setting `IFS` (Internal Field Separator) to an empty value and using `-r` (to prevent backslash escapes from being interpreted), the entire line of input, including spaces, is read into the variable `line`.

Setting a Timeout:

```
read -t 5 -p "Enter your name (you have 5 seconds): " name
echo "Hello, $name"
```

The `-t` option sets a timeout. In this example, `read` will wait for 5 seconds for the user to enter their name.

Reading from a File:

```
while IFS= read -r line
do
  echo "Line: $line" done < filename.txt
```

Options de ligne de commande de la commande de lecture

Voici quelques-unes de ses options couramment utilisées :

- `-r` : Cette option empêche les barres obliques inverses d'agir comme caractères d'échappement. Avec `-r`, les barres obliques inverses sont traitées comme des caractères littéraux. Ceci est généralement recommandé pour garantir que l'entrée est lue avec précision, sans aucune interprétation involontaire des caractères échappés par une barre oblique inverse.
- `-p` : Cette option vous permet de spécifier une invite qui s'affiche à l'utilisateur. Il est couramment utilisé lorsque vous interagissez avec l'utilisateur et attendez sa contribution. Par exemple, `read -p "Entrez votre nom : " name` afficherait "Entrez votre nom : ", puis stockerait l'entrée de l'utilisateur dans le nom de la variable.
- `-a array` : cette option lit à partir de l'entrée standard et attribue les mots lus dans un tableau indexé. Ceci est utile pour lire une ligne d'entrée dans un tableau, où chaque mot devient un élément du tableau.
- `-d delim` : l'option `-d` spécifie un caractère délimiteur pour terminer la ligne d'entrée, plutôt qu'une nouvelle ligne. Par exemple, `read -d ":" var` lira

- -n num : Cette option lit les caractères numériques plutôt qu'une ligne complète. C'est utile lorsque vous devez limiter la saisie à un nombre spécifique de caractères.
- -s : Mode silencieux. Lors de l'utilisation de -s, l'entrée saisie par l'utilisateur n'est pas affichée à l'écran. Ceci est généralement utilisé pour les entrées sensibles telles que les mots de passe.
- -t timeout : Spécifie un délai d'expiration en secondes. Si l'utilisateur ne fournit pas d'entrée dans le nombre de secondes spécifié, la commande de lecture se terminera. Ceci est utile pour les scripts dans lesquels vous ne souhaitez pas attendre indéfiniment la saisie de l'utilisateur.
- -u fd : Cette option indique à read de lire l'entrée du descripteur de fichier fd au lieu de l'entrée standard. Il s'agit d'une fonctionnalité avancée utilisée dans les scénarios dans lesquels vous manipulez des descripteurs de fichiers dans votre script.
- -N num : similaire à -n, mais il lit les caractères numériques plutôt que les octets. Ceci est particulièrement pertinent dans le contexte de caractères multi-octets.
- -E : utilisée avec -p, cette option désactive l'interprétation des échappements de barre oblique inverse et de la continuation de ligne dans la chaîne d'invite.
- -i texte : Cette option fournit un texte initial pour la lecture. C'est particulièrement utile pour fournir une valeur par défaut que l'utilisateur peut modifier.

La commande read est un outil puissant dans les scripts Bash pour gérer les entrées utilisateur et analyser le texte. Vous pouvez combiner ces options selon vos besoins pour répondre à vos besoins spécifiques.