

GABRIELE ROMANATO

Come ordinare un array di stringhe in Bash

In Bash, ordinare un array di stringhe può essere un'operazione utile per organizzare i dati in ordine alfabetico o in base a criteri specifici. In questo articolo, esploreremo diversi modi per effettuare il sorting di un array di stringhe utilizzando i comandi di Bash.

Utilizzare il Comando sort

Il metodo più semplice per ordinare un array di stringhe in Bash è quello di utilizzare il comando `sort`. Ecco un esempio:

```
#!/bin/bash

# Dichiarazione di un array di stringhe
array=("mela" "banana" "arancia" "uva" "ciliegia")

# Ordinamento dell'array e assegnazione a un nuovo array
sorted_array=( $(printf "%s\n" "${array[@]}" | sort) )

# Stampa dell'array ordinato
echo "Array ordinato: ${sorted_array[@]}"
```

In questo script:

- Utilizziamo `printf` per stampare ogni elemento dell'array su una nuova riga.
- Il comando `sort` ordina le righe in ordine alfabetico.
- I risultati vengono riassegnati a un nuovo array.

Ordinamento Inverso

Per ottenere un ordinamento inverso (dalla Z alla A), possiamo usare l'opzione `-r` del comando `sort`:

```
sorted_array=( $(printf "%s\n" "${array[@]}" | sort -r) )
```

Ordinamento Case-Insensitive

Se si desidera un ordinamento che ignori la differenza tra maiuscole e minuscole, si può usare l'opzione `-f`:

```
sorted_array=( $(printf "%s\n" "${array[@]}" | sort -f) )
```

Ordinamento Numerico

Se l'array contiene numeri rappresentati come stringhe e si desidera ordinarli numericamente, si può usare l'opzione `-n`:

```
numeric_array=("10" "2" "33" "5")
sorted_numeric_array=( $(printf "%s\n" "${numeric_array[@]}"
| sort -n) )

echo "Array numerico ordinato: ${sorted_numeric_array[@]}"
```

Conclusione

Il comando `sort` è un metodo potente e flessibile per ordinare array di stringhe in Bash. Grazie alle diverse opzioni disponibili, è possibile personalizzare l'ordinamento in base alle proprie esigenze.