

# Riferimento rapido sed

Editing di flusso, sostituzioni, indirizzi, hold space, trasformazioni in-place

## Nozioni di base

### Esecuzione di sed

```
sed 's/old/new/' file.txt # substitute first match per line
sed 's/old/new/g' file.txt # substitute all matches per line
sed -n '5p' file.txt # print only line 5
sed '3d' file.txt # delete line 3
echo "hello" | sed 's/hello/hi/' # pipe input
```

### Flag da riga di comando

<b>-n</b>	Sopprime la stampa automatica; stampa solo con <b>p</b>
<b>-e 'cmd'</b>	Esegue un comando sed (concatena più con <b>-e</b> )
<b>-f script.sed</b>	Legge i comandi da un file
<b>-i[suffisso]</b>	Modifica il file in-place (suffisso di backup opzionale)
<b>-E / -r</b>	Usa espressioni regolari estese

## Sostituzione

### Sintassi della sostituzione

```
sed 's/foo/bar/' f # first occurrence per line
sed 's/foo/bar/g' f # all occurrences
sed 's/foo/bar/3' f # 3rd occurrence only
sed 's/foo/bar/gi' f # all, case-insensitive
sed 's|usr/bin|opt/bin|g' f # alternate delimiter
```

### Flag di sostituzione

<b>g</b>	Sostituisce tutte le occorrenze sulla riga
<b>N (numero)</b>	Sostituisce solo la N-esima occorrenza
<b>p</b>	Stampa la riga se la sostituzione è avvenuta
<b>w file</b>	Scrive le righe sostituite su file
<b>i / I</b>	Corrispondenza case-insensitive (GNU)

## Indirizzi

### Esempi di indirizzi

```
sed '3s/a/b/' f # only on line 3
sed '2,5s/a/b/' f # lines 2 through 5
sed '/^#d' f # delete lines starting with #
sed '/start/,/end/d' f # delete range between patterns
sed '1-2d' f # delete odd-numbered lines (GNU)
```

### Tipi di indirizzo

<b>N</b>	Riga numero N
<b>\$</b>	Ultima riga
<b>N,M</b>	Intervallo di righe da N a M
<b>/regex/</b>	Righe che corrispondono alla regex
<b>/regex1/,/regex2/</b>	Intervallo dalla prima alla seconda corrispondenza
<b>N~step</b>	A partire da N, ogni step righe (GNU)
<b>addr!</b>	Negazione — applica alle righe non corrispondenti

## Elimina e stampa

### Comandi di eliminazione e stampa

```
sed '5d' f # delete line 5
sed '/^$/d' f # delete blank lines
sed -n '10,20p' f # print lines 10-20
sed -n '/error/p' f # print lines matching pattern
sed '/debug!d' f # keep only matching lines
```

### Riferimento comandi

<b>d</b>	Elimina il pattern space, inizia il ciclo successivo
<b>D</b>	Elimina fino al primo newline nel pattern space
<b>p</b>	Stampa il pattern space
<b>P</b>	Stampa fino al primo newline nel pattern space
<b>q</b>	Esce dopo aver stampato il pattern space corrente
<b>Q</b>	Esce senza stampare (GNU)

## Inserisci e aggiungi

### Inserimento, aggiunta, modifica

```
sed '3i\inserted line' f # insert before line 3
sed '3a\appended line' f # append after line 3
sed '3c\replaced line' f # replace line 3
sed '/marker/a\new line' f # append after pattern match
```

### Comandi

<b>i\testo</b>	Inserisce testo prima della riga corrente
<b>a\testo</b>	Aggiunge testo dopo la riga corrente
<b>c\testo</b>	Sostituisce la riga corrente con testo
<b>r file</b>	Legge e aggiunge il contenuto del file
<b>R file</b>	Legge e aggiunge una riga dal file (GNU)
<b>w file</b>	Scrive il pattern space su file

## Hold space

### Comandi dell'hold space

<b>h</b>	Copia il pattern space nell'hold space
<b>H</b>	Aggiunge il pattern space all'hold space
<b>g</b>	Copia l'hold space nel pattern space
<b>G</b>	Aggiunge l'hold space al pattern space
<b>x</b>	Scambia pattern e hold space

### Esempi con hold space

```
sed -n '1!G;h;$p' f # reverse lines (tac)
sed '/^$/{ x; s/\n//; x; }' f # collapse hold on blank
sed -n 'H:${x;s/\n/ /g;p;}' f # join all lines with space
```

## Comandi multipli

### Concatenazione di comandi

```
sed -e 's/foo/bar/g' -e 's/baz/qux/g' f
sed 's/foo/bar/g; s/baz/qux/g' f
sed '/header/{ s/old/new/; s/foo/bar/; }' f
sed -f commands.sed input.txt
```

### Raggruppamento e diramazione

<b>{ cmd1; cmd2; }</b>	Raggruppa comandi per lo stesso indirizzo
<b>:etichetta</b>	Definisce un'etichetta di diramazione
<b>b etichetta</b>	Salta all'etichetta
<b>t etichetta</b>	Salta se l'ultimo <b>s///</b> è riuscito
<b>T etichetta</b>	Salta se l'ultimo <b>s///</b> è fallito (GNU)

## Modifica in-place

### Esempi in-place

```
sed -i 's/old/new/g' file.txt # edit in place (GNU)
sed -i.bak 's/old/new/g' file.txt # backup as file.txt.bak
sed -i '' 's/old/new/g' file.txt # macOS in-place (no backup)
sed -i '/^#/d' config.txt # remove comments in place
```

### Note di piattaforma

<b>GNU sed -i</b>	Il suffisso è opzionale; <b>-i</b> da solo modifica senza backup
<b>BSD/macOS sed -i</b>	Richiede argomento suffisso; usa <b>-i ''</b> per nessun backup
<b>-i.bak</b>	Crea un file di backup con estensione <b>.bak</b>
<b>File multipli</b>	<b>sed -i 's/a/b/g' *.txt</b> modifica tutti i file corrispondenti

## Regex

### Regex in sed

<b>.</b>	Qualsiasi carattere singolo
<b>*</b>	Zero o più del precedente
<b>\+</b>	Uno o più (BRE) — <b>+</b> in ERE
<b>\?</b>	Zero o uno (BRE) — <b>?</b> in ERE
<b>^</b>	Inizio riga
<b>\$</b>	Fine riga
<b>[abc]</b>	Classe di caratteri
<b>\( \)</b>	Gruppo di cattura (BRE) — <b>()</b> in ERE
<b>\1, \2</b>	Backreference al gruppo di cattura
<b>&amp;</b>	Intera stringa corrispondente (nella sostituzione)

### Esempi di regex

```
sed 's/[0-9]\+/NUM/g' f # replace numbers
sed -E 's/(foo)(bar)\2\1/g' f # swap groups (ERE)
sed 's./*/(&)/' f # wrap line in parens
sed 's/[ \t]*$/' f # strip trailing whitespace
```

## Pattern comuni

### One-liner

```
sed -n '1p' f # first line (head -1)
sed '$!d' f # last line (tail -1)
sed '/^$/d' f # remove blank lines
sed 's/^[ \t]*$/' f # strip leading whitespace
sed '=' f | sed 'N;s/\n\t/' f # number lines
```

## Ricette

<b>Doppio spazio</b>	<b>sed G</b> — aggiunge riga vuota dopo ogni riga
<b>Rimuovi tag HTML</b>	<b>sed 's/&lt;[&gt;]*//g'</b>
<b>Estrai email</b>	<b>sed -nE 's/.*([a-z]+@[a-z.]+).*/\1/p'</b>
<b>Commenta righe</b>	<b>sed 's/^/# /'</b> — prefissa ogni riga con <b>#</b>
<b>Elimina righe vuote a fine file</b>	<b>sed -e :a -e '/^\n*\$/!\${d;N;ba' -e '}'</b>
<b>Sostituisci riga N-esima</b>	<b>sed 'Nc\nuovo testo'</b> — sostituisce la riga N