

RIASSUNTO SED

In rete esiste già (quasi) tutto su sed, e' inutile riscrivere le cose ovvie. Mostrero' solamente una serie di esempi di uso comune e di uso meno comune, senza pretesa di completezza. Questo scritto e' semplicemente una somma di appunti, e serve solo a non dedicare 1 ora alla lettura della pagina "man" di sed. Le prove sono state svolte su slackware 14.2. Useremo un file di testo piano per le nostre prove, tale file avra' come nome "provased" , ed il suo contenuto sara' il seguente:

```
prima riga
seconda riga
terza riga
quarta riga
quinta riga
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

Sostituiamo la parola "riga" con la parola "linea"

```
sed 's/riga/linea' provased
prima linea
seconda linea
terza linea
quarta linea
quinta linea
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

Possiamo indicare due sostituzioni in contemporanea in un'unica linea, in questo caso diventa obbligatorio indicare l'opzione "-e"

```
sed -e 's/riga/linea/; s/quinta/ultima/' provased
prima linea
seconda linea
terza linea
quarta linea
ultima linea
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
```

5

Ora utilizziamo come prova il file di testo piano "provased2" , il quale si compone di 5 linee , ed ogni linea si compone di una sola parola , ripetuta 5 volte, la parola e' "kernel", quindi il file "provased2" e' il seguente

```
kernel kernel kernel kernel kernel
kernel kernel kernel kernel kernel
kernel kernel kernel kernel kernel
kernel kernel kernel kernel kernel
kernel kernel kernel kernel kernel
```

Riproviamo il comando di sostituzione: vogliamo sostituire la parola "kernel" con la parola Linux

```
sed 's/kernel/linux/' provased2
linux kernel kernel kernel kernel
linux kernel kernel kernel kernel
linux kernel kernel kernel kernel
linux kernel kernel kernel kernel
linux kernel kernel kernel kernel
```

Se invece volessimo sostituire tutte le parole "kernel" con la parola "linux" allora dovremmo usare il seguente comando

```
sed 's/kernel/linux/g' provased2
linux linux linux linux linux
linux linux linux linux linux
linux linux linux linux linux
linux linux linux linux linux
linux linux linux linux linux
```

Al lettore il compito di trovare il modo di sostituire solo 2 (oppure 3) volte la parola "kernel". Torniamo al nostro file di esempio "provased". Ora sostituiamo la parola "riga" con la parola "linea", e stampiamo a monitor soltanto le sostituzioni eseguite, tralasciando le righe non modificate

```
sed -n 's/riga/linea/p' provased
prima linea
seconda linea
terza linea
quarta linea
quinta linea
```

Come sopra, ma con il salvataggio delle righe modificate , all'interno del file di testo piano "output.txt"

```
sed -n 's/riga/linea/p w output.txt' provased
```

modifichiamo soltanto la seconda linea

```
sed '2s/riga/linea/' provased
prima riga
```

```
seconda linea
terza riga
quarta riga
quinta riga
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

modifichiamo le linee da 1 a 4

```
sed '1,4s/riga/linea/' provased
prima linea
seconda linea
terza linea
quarta linea
quinta riga
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

modifichiamo dalla terza linea fino alla fine del file

```
sed '3,$s/riga/linea/' provased
prima riga
seconda riga
terza linea
quarta linea
quinta linea
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

cancelliamo la terza linea del file

```
sed '3d' provased
prima riga
seconda riga
quarta riga
quinta riga
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

cancelliamo dalla quarta riga fino alla fine del file

```
sed '4,$d' provased  
prima riga  
seconda riga  
terza riga
```

inseriamo una riga prima di ogni riga gia' esistente nel file "provased"

```
sed 'i/ prima di ogni riga ' provased  
/ prima di ogni riga  
prima riga  
/ prima di ogni riga  
seconda riga  
/ prima di ogni riga  
terza riga  
/ prima di ogni riga  
quarta riga  
/ prima di ogni riga  
quinta riga  
/ prima di ogni riga  
0 1 2 3 4 5  
/ prima di ogni riga  
1 2 3 4 5  
/ prima di ogni riga  
2 3 4 5  
/ prima di ogni riga  
3 4 5  
/ prima di ogni riga  
4 5  
/ prima di ogni riga  
5
```

inseriamo un riga dopo ogni riga gia' esistente nel file "provased"

```
sed 'a/append' provased  
prima riga  
/append  
seconda riga  
/append  
terza riga  
/append  
quarta riga  
/append  
quinta riga  
/append  
0 1 2 3 4 5  
/append  
1 2 3 4 5  
/append  
2 3 4 5  
/append  
3 4 5
```

```
/append
4 5
/append
5
/append
```

modificare una singola linea del file "provased"

```
sed '4c _linea_modificata__' provased
prima riga
seconda riga
terza riga
_linea_modificata__
quinta riga
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

trasformiamo i caratteri della parola "riga" con i seguenti numeri "1234"

```
sed -e 'y/riga/1234/' provased
p12m4 1234
second4 1234
te1z4 1234
qu41t4 1234
qu2nt4 1234
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

elimina tutte le righe vuote del file "provased". non vi sono righe vuote, quindi sembra che non succeda nulla

```
sed '/^$/d' provased
prima riga
seconda riga
terza riga
quarta riga
quinta riga
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

mette un trattino basso "_" all'inizio di ogni linea del file "provased"

```

sed 's/^/_/' provased
_prima riga
_seconda riga
_terza riga
_quarta riga
_quinta riga
_0 1 2 3 4 5
_1 2 3 4 5
_2 3 4 5
_3 4 5
_4 5
_5

```

mette un trattino basso alla fine di ogni linea del file "provased"

```

sed 's/$/_/' provased
prima riga_
seconda riga_
terza riga_
quarta riga _
quinta riga_
0 1 2 3 4 5 _
1 2 3 4 5 _
2 3 4 5 _
3 4 5 _
4 5 _
5 _

```

mettiamo le parentesi tonde attorno ad ogni numero, sempre nel file "provased" in questo caso il simbolo "&" corrisponde al risultato dell'espressione regolare [0-9][0-9]*

```

sed 's/[0-9][0-9]*/(&)/g' provased
prima riga
seconda riga
terza riga
quarta riga
quinta riga
(0) (1) (2) (3) (4) (5)
(1) (2) (3) (4) (5)
(2) (3) (4) (5)
(3) (4) (5)
(4) (5)
(5)

```

invertiamo le parole nelle righe testuali del file "provased". In questo caso le notazioni " \2 " e " \1 " fanno riferimento agli "spezzoni" di testo da invertire

```

sed 's/\([a-z]*\) \([a-z]*\)\/\2 \1/' provased
riga prima
riga seconda
riga terza
riga quarta

```

```
riga quinta
0 1 2 3 4 5
1 2 3 4 5
2 3 4 5
3 4 5
4 5
5
```

vediamo le parole duplicate di un determinato file

```
sed -n '/\([a-z][a-z]*\) \1/p' provased2
kernel kernel kernel kernel kernel
kernel kernel kernel kernel kernel
kernel kernel kernel kernel kernel
kernel kernel kernel kernel kernel
kernel kernel kernel kernel kernel
```

fonti:

<http://www.pluto.it/files/ildp/guide/abs/sedawk.html>

<https://www.grymoire.com/Unix/Sed.html>

