SED Quick Reference

Commands	Function
a\	append one or more lines of text to the current line
b *label*	jump to *label*; if label is not specified, then jump to the end of the script. This is an unconditional branch
c\	change (replace) text in the current line with new text
d	delete line
D	delete the first line from pattern space. Control then passes to the top of the script. If command 'D' empties the pattern space, a new line will be read in; otherwise, no new line will be read in
i\	insert text above the current line
g	get what is in the holding buffer and copies it into the pattern buffer, overwriting the pattern space
G	get what is in the holding buffer and copies it into the pattern space, appending to what was there
h	copies the contents of the pattern space to a holding buffer
H	append the contents of the pattern space to a holding buffer
l	list nonprinting characters
n	read the next input line and starts processing the newline with the command rather than the first command
N	append next line to pattern space. Next line is separated from the original pattern space by a newline character
p	print the contents of pattern space
P	print the first line of pattern space
q	print the contents of pattern space then quits or exit *sed*
r	read lines from a file
s	substitute one string for another
t	jump to label if any substitution has been made on the pattern space since the most recent reading of input line or execution of command 't'. If label is not specified, then jump to the end of the script. This is a conditional branch
=	display line number of a line
!	applie the command to all lines except the selected ones
:*label*	label branched to by t or b

Flags

Flags	Function
g	Globally substitutes on a line
i	Ignore case sencitive with substitution
p	Prints lines
w	Writes lines out to a file
x	Exchanges contents of the holding buffer with the pattern space
y	Translates one character to another

Options

Options	Function
-e	Allows multiple edits
-n	Suppresses default output
-f	Precedes a *sed* script filename

Appending: a command

Example

```
sed '/^Elizabeth /a\
Fahd Main:7186794751:Queens:m:56:35:java' empl.dat
```

```
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java
```

Explanation

Append line "Fahn Main ..." after line where "Elizabet" is at the beginning of a line.

Insertin: i command

Example

```
sed '/^Greg/i\
******************************************' empl.dat
```

```
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
Fahd Main:7186794751:Queens:m:56:35:java
******************************************
Greg Norman:7182237890:Queens:m:45:0:java
```

Explanation

Insert line "****** ..." before line where "Greg" is at the beginning of a line.

Replacingt: c command

Example

```
sed '/\*\**/c\
==========================================' empl.dat
```

```
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
Fahd Main:7186794751:Queens:m:56:35:java
==========================================
Greg Norman:7182237890:Queens:m:45:0:java
```

Explanation

Replace line "****** ..." for line "=============...."

Deliting: d command

Example

sed '/==*/d' empl.dat

```
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java
```

Explanation

Delete line that contains two or secquances of "=" characters

Printing: p command

Example

sed '/Fahd/p' empl.dat

```
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
Fahd Main:7186794751:Queens:m:56:35:java
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java
```

Explanation

Prints all lines to standard output by default. If the pattern *Fahd* is found, *sed* will print that line in addition to all the other lines.

Printing: p command with -n option

Example

sed -n '/Fahd/p' empl.dat

```
Fahd Main:7186794751:Queens:m:56:35:java
```

Explanation

The -n option suppresses the default behavior of *sed* when used with the **p** command. Only the lines containing the pattern "Fahd" are printed when -n is used.

Multiple editing: -e option

Example

cat test.txt

```
0000000000000000
```

sed -e 's/00/11/3' -e 's/0/1/12' test.dat

```
0000110000010000
```

Explanation

Substitution third occurrence of "00" character for "11", and twelve's "0" for "1" flag **e** indicates that after execution first substitution, *sed* will execute next command. Numeric flag used for specializing which occurrence of "pattern" to proccess.

Ignoring case sencitive: i command

Example

cat test.dat

```
first line
second Line
LiNe number three
```

sed '1,$s/line/LINE/i' test.dat

```
first LINE
second LINE
LINE number three
```

Explanation

Substitution all occurrences of "line" (ignoring case f.e. Line, LinE), for "LINE". **i** flag is use during substitution.

Translating: y flag

Example

cat test.dat

```
Robert Frost
```

sed 'y/abcdefghijklmnopqrst/ABCDEFGHIKLMNOPQRST/' test.dat

```
ROBERT FROST
```

Explanation

Negation: ! flag

Example

cat empl.dat

```
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java
```

sed '/cobol/!d' empl.dat

```
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
```

Explanation

Delete all lines, exept line(s) contains pattern "cobol";

Next: n command

Example

cat test.dat

```
first line
second line
line number 3
```

sed '/second/{n; s/3/three/}' test.dat

```
first line
second line
line number three
```

Explanation

If pattern "second" is found, substitute "3" for "three" on the next line.

Holding: h command

Example

cat test.dat

```
first line
second line
line number three
forth Line
```

sed -e '/second/{h; d;}' -e '$g' test.dat

```
first line
line number three
forth Line
second line
```

Explanation

If pattern "second" is found, copy this line into ***holding buffer*** (**h;**). Replace containt of holding buffer if it is not empty (lowercase **h;**). Delete line with the pattern (**d;**) from ***pattern space***. Next **-e** option: copy the contents of holding buffer to the stdout (**g**) when last line is reached (**$**).

Holding: H command

Example

cat empl.dat

```
Albert Bronx:7187634623:Manhattan:m:56:32:cobol
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java
```

sed -e '/cobol/{H; d;}' -e '$g' empl.dat

```
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java

Albert Bronx:7187634623:Manhattan:m:56:32:cobol
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
```

Explanation

If pattern "cobol" is found, append (**H**) line with the pattern to the *holding buffer*. Delete this line from ***pattern space*** (**d;**). Second **-e** option: Put contents of the *holdig buffer* to the stdout.
(Replace lines with pattern to the end of file).

Writing: w command

Example

sed -n '/cobol/ w cobol_empl.dat' empl.dat

cat cobol_empl.dat

```
Albert Bronx:7187634623:Manhattan:m:56:32:cobol
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
```

Explanation

Same as above, but write (**w**) lines with a pattern "cobol" into the file cobol_empl.dat.

Reading: r command

Example

cat line.txt

--

sed '/cobol/r line.txt' test.dat

```
Albert Bronx:7187634623:Manhattan:m:56:32:cobol
---------------------------------------------------
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
---------------------------------------------------
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java
```

Explanation

If pattern "cobol" is fond, read (**r**) the file *line.txt* into the file *test.dat* after each occurrence of the pattern.

Quiting: q command

Example

sed '2q' test.dat

```
Albert Bronx:7187634623:Manhattan:m:56:32:cobol
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
```

Explanation

Print the first two lines on the stdout ("print" is default), and quit (**q**). (Good for big file(s)).

Substitution: s command

Example

sed '1,$s/:/ /g' empl.dat

```
Alex Stachelin 7182347634 Brooklyn m 60 60 unix
Elizabeth Harrington 7183214567 Brooklyn f 42 40 cobol
Fahd Main 7186794751 Queens m 56 35 java
Greg Norman 7182237890 Queens m 45 0 java
```

Explanation

Substitution all occurrences of ":" character for single space. Flaf **g** indicates that the substitution is *global* across the hole line.

Multiple editing: -e option

Example

cat test.txt

```
0000000000000000
```

sed -e 's/00/11/3' -e 's/0/1/12' test.dat

```
0000110000010000
```

Explanation

Substitution third occurrence of "00" character for "11", and twelve's "0" for "1" flag **e** indicates that after execution first substitution, *sed* will execute next command. Numeric flag used for specializing which occurrence of "pattern" to proccess.

Ignoring case sencitive: i command

Example

cat test.dat

```
first line
second Line
LiNe number three
```

sed '1,$s/line/LINE/i' test.dat

```
first LINE
second LINE
LINE number three
```

Explanation

Substitution all occurrences of "line" (ignoring case f.e. Line, LinE), for "LINE". **i** flag is use during substitution.

Translating: y flag

Example

cat test.dat

```
Robert Frost
```

sed 'y/abcdefghijklmnopqrst/ABCDEFGHIKLMNOPQRST/' test.dat

```
ROBERT FROST
```

Explanation

Negation: ! flag

Example

cat empl.dat

```
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java
```

sed '/cobol/!d' empl.dat

```
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
```

Explanation

Delete all lines, exept line(s) contains pattern "cobol";

Next: n command

Example

cat test.dat

```
first line
second line
line number 3
```

sed '/second/{n; s/3/three/}' test.dat

```
first line
```

```
second line
line number three
```

Explanation

If pattern "second" is found, substitute "3" for "three" on the next line.

Holding: h command

Example

cat test.dat

```
first line
second line
line number three
forth Line
```

sed -e '/second/{h; d;}' -e '$g' test.dat

```
first line
line number three
forth Line
second line
```

Explanation

If pattern "second" is found, copy this line into ***holding buffer*** (**h;**). Replace containt of holding buffer if it is not empty (lowercase **h;**). Delete line with the pattern (**d;**) from ***pattern space***. Next **-e** option: copy the contents of holding buffer to the stdout (**g**) when last line is reached (**$**).

Holding: H command

Example

cat empl.dat

```
Albert Bronx:7187634623:Manhattan:m:56:32:cobol
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java
```

sed -e '/cobol/{H; d;}' -e '$g' empl.dat

```
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java

Albert Bronx:7187634623:Manhattan:m:56:32:cobol
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
```

Explanation

If pattern "cobol" is found, append (**H**) line with the pattern to the *holding buffer*. Delete this line from ***pattern space*** (**d;**). Second **-e** option: Put contents of the *holdig buffer* to the stdout.
(Replace lines with pattern to the end of file).

Writing: w command

Example

```
sed -n '/cobol/ w cobol_empl.dat' empl.dat
cat cobol_empl.dat
```

```
Albert Bronx:7187634623:Manhattan:m:56:32:cobol
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
```

Explanation

Same as above, but write (**w**) lines with a pattern "cobol" into the file cobol_empl.dat.

Reading: r command

Example

```
cat line.txt
----------------------------------------
sed '/cobol/r line.txt' test.dat
```

```
Albert Bronx:7187634623:Manhattan:m:56:32:cobol
---------------------------------------------------
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
Elizabeth Harrington:7183214567:Brooklyn:f:42:40:cobol
---------------------------------------------------
Fahd Main:7186794751:Queens:m:56:35:java
Greg Norman:7182237890:Queens:m:45:0:java
```

Explanation

If pattern "cobol" is fond, read (**r**) the file *line.txt* into the file *test.dat* after each occurrence of the pattern.

Quiting: q command

Example

```
sed '2q' test.dat
```

```
Albert Bronx:7187634623:Manhattan:m:56:32:cobol
Alex Stachelin:7182347634:Brooklyn:m:60:60:unix
```

Explanation

Print the first two lines on the stdout ("print" is default), and quit (**q**). (Good for big file(s)).