sed - Stream Editor

What is Sed ?

A “non-interactive” text editor that is called from the unix command line.
Input text flows through the program, is modified, and is directed to standard output.

- A tool usually designed for a short line substitution, deletion and print
- Allows for automation of editing process similar to those you might find in vi or ex (or ed)
- Non-destructive
- Reads in a stream of data from a file, performs a set of actions, & outputs the results

How it Works

- SED reads the data one line at a time, make a copy of the input line & places it in a buffer called the **pattern space**
- Modifies that copy in the **pattern space**
- Outputs the copy to standard output (terminal window)

- The pattern space holds the line of text currently being processed
- You don’t make changes to the original file
- Changes can be captured by redirecting it to another files

sed Commands

We will go over the only some basic sed commands.

a	append
c	change lines
d	delete lines
i	insert
p	print lines
s	substitute

Sed syntax

Syntax

Sed [option] ‘instruction’ file

- Options
  - n → only prints matches
  - i → in line, source file will be modified

sed’s Print Command

USAGE   - without any expression

```
$  sed  -n  ‘p’   <filename>
```

Prints the contents of file. Same like cat command

USAGE - With line numbers

```
$ sed  -n  ‘<lineno1>,<lineno2>p’  <filename>
```

Ex:

```
$ sed –n  ‘5p’  <filename>   ## prints fifth line
$ sed –n ‘3,6p’  <filename   ## prints lines from 3 to 5
$ sed  -n  ‘$p’  <filename>   ## prints last line of the file
$ sed  -n  ‘5,$p’  <filename>  ## prints all lines from line 5th till end of file
$ sed –n  ‘4,+2p’ <filename>   ##  prints 4th line plus next 2 lines
$ sed -n  '1~2p‘  <filename>  prints every second line
```

sed’s Print Command

USAGE - With search pattern

```
$ sed  -n  ‘/<pattern1>/, /<pattern2>/p’  <filename>
Sed  -n  ‘/   /p’
```

Ex:

```
$ sed –n '/abc/p' pritn all line containing word abc
$ sed  -n  ‘/abc/, /xyz/p’  <filename>  ## prints all lines between pattern ‘abc’ and ‘xyz’
```

Delete Command

USAGE

```
$ sed   ‘<address>d’  <filename>
```

Ex:

```
$ sed   ‘5d’  <filename>   ## deletes fifth line
$ sed  ‘3,6d’  <filename   ## deletes lines from 3 to 5
$ sed    ‘$d’  <filename>   ## deletes last line of the file
$ sed   ‘5,$d’  <filename>  ## deletes all lines from line 5th till end of file
$ sed   ‘4,+2d’ <filename>   ##  deletes 4th line plus next 2 lines
$ sed  '1~2d‘  <filename>  deletes every second line
$ sed   ‘/abc/, /xyz/d’  <filename>  ## deletes all lines between pattern ‘abc’
and ‘xyz’
```

Substitute Command

Replaces search pattern with given pattern

USAGE

```
$ sed ‘s/pattern1/pattern2/flag’ <filename>
```

Flags:

- **g** Replace all instances of /*pattern*/ on each addressed line, not just the first instance.

Append Command

Append command is used to add a line after the matching address

USAGE:

```
$ sed  ‘/<pattern>/a\ <text for new line>’   <filename>
$ sed ‘5a\ <text for new line’   <filename>
```

Insert Command

Insert command is used to add a line before the matching address

USAGE:

```
$ sed  ‘/<pattern>/i\ <text for new line>’   <filename>
$ sed ‘5i\ <text for new line’   <filename>
```

Change Command

Change command deletes the matching lines and in their place writes a line with given text

USAGE:

```
$ sed  ‘/<pattern1>/, /pattern2/c\ <text >’   <filename>
$ sed ‘5,8c\ <text’   <filename>
```