Confidentiality Statement

Tata Code of Conduct

We, in our dealings, are self-regulated by a Code of Conduct as enshrined in the Tata Code of Conduct. We request your support in helping us adhere to the Code in letter and spirit. We request that any violation or potential violation of the Code by any person be promptly brought to the notice of the Local Ethics Counselor or the Principal Ethics Counselor or the CEO of TCS.   All communication received in this regard will be treated and kept as confidential.

TATA CONSULTANCY SERVICES

vILP – Level 2

Content Manual   Version 1.1

December 2014

(ILP Guwahati)

Table of Content

1.10. sed

sed is a stream editor used to perform basic text transformations on an input stream (a file, or input from a [pipeline](#)).

1.10.1. *Working methodology*

sed works by making only one pass over the input(s) s called as one execution cycle. Cycle continues till end of file/input is reached.

- Read entire line from stdin/file.
- Removes any trailing newline.
- Places the line, in its pattern buffer.
- Modify the pattern buffer according to the supplied commands.
- Print the pattern buffer to stdout.

1.10.2. Printing Operation in sed

sed allows you to print only specific lines based on the line number or pattern matches. **"p"** is the command for printing the data from the pattern buffer. To suppress automatic printing of patternspace -n option is used with sed. sed -n option will not print anything, unless an explicit request to print is found.

Syntax:

sed -n 'ADDRESS'p filename
sed -n '/pattern/p' filename

Examples:

Let us assume the demo_file has the following content

```
1. Linux - Sysadmin, Scripting etc.
2. Databases - Oracle, mySQL etc.
3. Hardware
4. Security (Firewall, Network, Online Security etc)
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
8. Website Design
9. Software Development
10. Windows- Sysadmin, reboot etc.
```

To prints third line of input file

```
$sed -n '3p' demo_file
3. Hardware
```

To print every nth line starting from the line m

```
 $sed -n 'm~np' filename
```

To print every 2nd line starting from 3rd line:

```
$sed -n '3~2p' demo_file
3. Hardware
5. Storage
7. Productivity (Too many technologies to explore, not much time available)
9. Software Development
```

To print from 4th line to 8th line from input file demo_file:

```
$ sed -n '4,8p' demo_file
4. Security (Firewall, Network, Online Security etc)
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
8. Website Design
```

To print only the last line

```
$sed -n '$'p demo_file
10.Windows- Sysadmin, reboot etc.
```

To print the lines containing the given pattern:

 Syntax:

```
 sed -n /PATTERN/p filename
```

To print the lines with pattern "Sysadmin":

```
$sed -n /Sysadmin/p thegeekstuff.txt
1. Linux - Sysadmin, Scripting etc.
10.Windows- Sysadmin, reboot etc.
```

To print from the nth line of the input, to the line matching the pattern. If the pattern doesn't match, it prints up to end of the input.

Syntax:
sed -n 'N,/PATTERN/p' filename

To prints from 3rd line to the line having the pattern "Security".
```
$sed -n '3,/Security/p' demo_file
3. Hardware
4. Security (Firewall, Network, Online Security etc)
```

To print from the line matching the given pattern to end of the file.

Syntax:
sed -n '/PATTERN/,$p' filename

To print the lines which matches the pattern and next N lines following the matched line.

sed -n '/PATTERN/,+Np' filename

To print the section of file between two regular expression (including the matched line).

sed -n '/P1/,/P2/p' filename
```
$ sed -n '/Storage/,/Design/p' demo_file
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
8. Website Design
```

1.10.3. Deletion operation in sed

In sed the d command is used to delete the pattern space buffer and immediately starts the next cycle.

Syntax:
sed nd filename
'nd' deletes the nth line and prints the other lines.

sed 'ADDRESS'd filename
sed /PATTERN/d filename

The process is
• It reads the first line and places in its pattern buffer
• checks whether supplied command is true for this line , if true, deletes pattern space buffer and starts next cycle and reads the next line.
• If supplied command is not true, it prints the content of the pattern space buffer.

To delete the 3rd line and print other lines from the file demo_file

```
$ sed 3d demo_file
1. Linux - Sysadmin, Scripting etc.
2. Databases - Oracle, mySQL etc.
4. Security (Firewall, Network, Online Security etc)
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
8. Website Design
9. Software Development
10. Windows- Sysadmin, reboot etc.
```

Delete starting from 3rd line and every 2nd line from there.

```
$ sed '3~2d' demo_file
1. Linux - Sysadmin, Scripting etc.
2. Databases - Oracle, mySQL etc.
4. Security (Firewall, Network, Online Security etc)
6. Cool gadgets and websites
8. Website Design
10. Windows- Sysadmin, reboot etc.
```

Delete last line from input.

```
$ sed '$d' thegeekstuff.txt
1. Linux - Sysadmin, Scripting etc.
2. Databases - Oracle, mySQL etc.
3. Hardware
4. Security (Firewall, Network, Online Security etc)
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
8. Website Design
9. Software Development
```

Delete the line which matches the given pattern from input.

```
$ sed /Sysadmin/d demo_file
2. Databases - Oracle, mySQL etc.
3. Hardware
4. Security (Firewall, Network, Online Security etc)
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
8. Website Design
9. Software Development
```

To deletes the line from which matches the given pattern to end of the file.

```
$ sed '/Website/,$d' demo_file
1. Linux - Sysadmin, Scripting etc.
2. Databases - Oracle, mySQL etc.
3. Hardware
4. Security (Firewall, Network, Online Security etc)
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
```

To deletes the line from which matches the given pattern and 2lines next to that.

```
$ sed '/Storage/,+2d' demo_file
1. Linux - Sysadmin, Scripting etc.
2. Databases - Oracle, mySQL etc.
3. Hardware
4. Security (Firewall, Network, Online Security etc)
8. Website Design
9. Software Development
10.Windows- Sysadmin, reboot etc.
```

To delete blank Line from a file using sed

```
$ sed '/^$/d' demo_file
1. Linux - Sysadmin, Scripting etc.
2. Databases - Oracle, mySQL etc.
3. Hardware
4. Security (Firewall, Network, Online Security etc)
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
```

```
8. Website Design
9. Software Development
10.Windows- Sysadmin, reboot etc.
```

1.10.4. Substitution operation in sed

In sed the **S** command is used to substitute the pattern. The `s' command attempts to match the pattern space against the supplied expression/ pattern; if the match is successful, then that portion of the pattern space which was matched is replaced with the replacement given.

Syntax:

```
$sed 'ADDRESSs/REGEXP/REPLACEMENT/FLAGS' filename
$sed 'PATTERNs/REGEXP/REPLACEMENT/FLAGS' filename
```

- s is substitute command
- / is a delimiter
- REGEXP is regular expression to match
- REPLACEMENT is a value to replace

FLAGS can be any of the following:

- g Replace all the instance of REGEXP with REPLACEMENT
- n Could be any number,replace nth instance of the REGEXP with REPLACEMENT.
- p If substitution was made, then prints the new pattern space.
- i match REGEXP in a case-insensitive manner.
- w file If substitution was made, write out the result to the given file.
- We can use different delimiters (one of @ % ; :) instead of /

Example:

To substitute Word "Linux" to "Linux-Unix" in thr file demo_file

```
$ sed 's/Linux/Linux-Unix/' thegeekstuff.txt

1. Linux-Unix Sysadmin, Linux Scripting etc.
2. Databases - Oracle, mySQL etc.
3. Security (Firewall, Network, Online Security etc)
4. Storage in Linux-Unix
5. Productivity (Too many technologies to explore, not much time available)
# Additional FAQS
```

6. Windows- Sysadmin, reboot etc.

Substitute all Appearances of a Word, flag "g" is used.

$ sed 's/Linux/Linux-Unix/g' demo_file

1. Linux-Unix Sysadmin, Linux-Unix Scripting etc.
2. Databases - Oracle, mySQL etc.
3. Security (Firewall, Network, Online Security etc)
4. Storage in Linux-Unix
5. Productivity (Too many technologies to explore, not much time available)
Additional FAQS
6. Windows- Sysadmin, reboot etc.

To Substitute Only 2nd Occurrence of a Word

$ sed 's/Linux/Linux-Unix/2' demo_file
# Instruction Guides
1. Linux Sysadmin, Linux-Unix Scripting etc.
2. Databases - Oracle, mySQL etc.
3. Security (Firewall, Network, Online Security etc)
4. Storage in Linux
5. Productivity (Too many technologies to explore, not much time available)
Additional FAQS
6. Windows- Sysadmin, reboot etc.

To Write Changes to a File and Print the Changes

$ sed -n 's/Linux/Linux-Unix/gpw output' demo_file
1. Linux-Unix Sysadmin, Linux-Unix Scripting etc.
4. Storage in Linux-Unix
$ cat output
1. Linux-Unix Sysadmin, Linux-Unix Scripting etc.
4. Storage in Linux-Unix

To combine multiple sed commands we have to use option -e
Syntax:
$sed -e 'command' -e 'command' filename

Example:

To delete 4th and 2nd line from the input

```
$ sed -e '4d' -e '2d' demo_file
1. Linux - Sysadmin, Scripting etc.
3. Hardware
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
8. Website Design
9. Software Development
10. Windows- Sysadmin, reboot etc.
```

To Delete the first,last and all the blank lines from input

```
$ sed -e '1d' -e '$d' -e '/^$/d' demo_file
2. Databases - Oracle, mySQL etc.
3. Hardware
4. Security (Firewall, Network, Online Security etc)
5. Storage
6. Cool gadgets and websites
7. Productivity (Too many technologies to explore, not much time available)
8. Website Design
9. Software Development
```