

Sed Command

SED Command stands for Stream Editor and it can perform lots of functions on file like searching, find and replace, insertion or deletion. Though most common use of SED Command is for substitution or to find and replace.

Example

```
Cat > test1.txt
```

Unix is great OS. Unix is open source.
Unix is free. Unix is easy to learn.

⇒ Replacing / substituting string. S = substitution

```
Sed 's/unix/linux/' test1.txt
```

Here all the Unix word will be replaced by linux.

Output

linux is great OS. linux is open source.
linux is free.

⇒ Replacing the n^{th} occurrence of a pattern in a line.

```
Sed 's/unix/linux/2' test1.txt
```

output

unix is great OS, linux is open source, Unix
is free OS. linux is easy to learn.

3. Replacing all the occurrence of the pattern
in a file

Sed 's/unix/linux/g' test1.txt.

g = global replacement.

output

linux is great OS, linux is open source,
linux is free OS. linux is easy to
learn

4. Replacing ~~all~~ from n^{th} occurrence to all
occurrence in a line.

Sed 's/unix/line/2g' test1.txt.

output

Unix is great OS, unix is open source,
linux is free OS. linux is easy to learn.

awk Command

awk is a scripting language used for manipulating data and generating reports. Awk is mainly used for pattern scanning and processing.

Cat test1.txt

```

ajay manager account 45000
Sonit Clerk account 25000
Harun manager account 5000
amit manager account 47000
Tarun Peon sales 15000
deepak clerk sales 23000
  
```

⇒ awk '{print}' test1.txt

By default awk command prints the whole data in the file without any changes

⇒ awk ~~'{print}'~~ '{/manager/{print}}' test1.txt

The awk command will print all the line which matches with the word 'manager'

Output

```

Ajay manager account 45000
Harun manager account 5000
amit manager account 47000
  
```


line
⇒ Splitting 4 into field

\$0 - print the whole line

awk '{print \$1, \$4}' test1.txt

output

ajay	45000
Sunil	25000
Varun	50000
amit	47000
tarun	15000
deepak	23000

⇒ Displaying line number.

awk '{print NR, \$0}' test1.txt

1. ajay manager account 45000
2. Sunil clerk account 25000
3. Varun manager sales 50000
4. amit manager sales 47000

⇒ Displaying last field

awk '{print \$1, NF}' test1.txt

ajay	45000
Sunil	25000
Varun	50000