

.AWK (AHO WEINBERGER KERNIGHAN) COMMAND

The AWK command is used for Text processing in linux . Although the SED command is also used for text processing , it has some limitations , so the AWK command becomes a handy option for text processing . It provides powerful control to the data . AWK is a powerful scripting language used for text scripting .

```

himanshu@foreman:~/Desktop$ awk
Usage: awk [POSIX or GNU style options] -f progfile [--] file ...
Usage: awk [POSIX or GNU style options] [--] 'program' file ...
POSIX options:          GNU long options: (standard)
    -f progfile          --file=progfile
    -F fs                --field-separator=fs
    -v var=val           --assign=var=val
Short options:          GNU long options: (extensions)
    -b                   --characters-as-bytes
    -c                   --traditional
    -C                   --copyright
    -d[file]             --dump-variables[=file]
    -D[file]             --debug[=file]
    -e 'program-text'    --source='program-text'
    -E file              --exec=file
    -g                   --gen-pot
    -h                   --help
    -i includefile       --include=includefile
    -l library           --load=library
    -L[fatal|invalid|no-ext] --lint[=fatal|invalid|no-ext]
    -M                   --bignum
    -N                   --use-lc-numeric
    -n                   --non-decimal-data
    -o[file]             --pretty-print[=file]
    -O                   --optimize
    -p[file]             --profile[=file]
    -P                   --posix
    -r                   --re-interval
    -s                   --no-optimize
    -S                   --sandbox
    -t                   --lint-old
    -V                   --version

```

To report bugs, see node 'Bugs' in 'gawk.info' which is section 'Reporting Problems and Bugs' in the printed version. This same information may be found at https://www.gnu.org/software/gawk/manual/html_node/Bugs.html. PLEASE do NOT try to report bugs by posting in comp.lang.awk, or by using a web forum such as Stack Overflow.

gawk is a pattern scanning and processing language. By default it reads standard input and writes standard output.

Examples:

```

awk '{ sum += $1 }; END { print sum }' file
awk -F: '{ print $1 }' /etc/passwd

```

```
himanshu@foreman:~/Desktop$ cat > employee.txt
ajay manager account 45000
sunil clerk account 25000
varun manager sales 50000
amit manager account 47000
tarun peon sales 15000
deepak clerk sales 23000
sunil peon sales 13000
satvik director purchase 80000
^C
himanshu@foreman:~/Desktop$
```

```
himanshu@foreman:~/Desktop$ awk '{print}' employee.txt
ajay manager account 45000
sunil clerk account 25000
varun manager sales 50000
amit manager account 47000
tarun peon sales 15000
deepak clerk sales 23000
sunil peon sales 13000
satvik director purchase 80000
himanshu@foreman:~/Desktop$
```

```
himanshu@foreman:~/Desktop$ awk '{print}' employee.txt
ajay manager account 45000
sunil clerk account 25000
varun manager sales 50000
amit manager account 47000
tarun peon sales 15000
deepak clerk sales 23000
sunil peon sales 13000
satvik director purchase 80000
himanshu@foreman:~/Desktop$ awk '/manager/ {print}' employee.txt
ajay manager account 45000
varun manager sales 50000
amit manager account 47000
himanshu@foreman:~/Desktop$
```

```
himanshu@foreman:~/Desktop$ awk '{print}' employee.txt
ajay manager account 45000
sunil clerk account 25000
varun manager sales 50000
amit manager account 47000
tarun peon sales 15000
deepak clerk sales 23000
sunil peon sales 13000
satvik director purchase 80000
himanshu@foreman:~/Desktop$ awk '/manager/ {print}' employee.txt
ajay manager account 45000
varun manager sales 50000
amit manager account 47000
himanshu@foreman:~/Desktop$ awk '{print $1,$4}' employee.txt
ajay 45000
sunil 25000
varun 50000
amit 47000
tarun 15000
deepak 23000
sunil 13000
satvik 80000
himanshu@foreman:~/Desktop$
```

```
himanshu@foreman:~/Desktop$ awk '{print NR,$0}' employee.txt
1 ajay manager account 45000
2 sunil clerk account 25000
3 varun manager sales 50000
4 amit manager account 47000
5 tarun peon sales 15000
6 deepak clerk sales 23000
7 sunil peon sales 13000
8 satvik director purchase 80000
himanshu@foreman:~/Desktop$ awk '{print $1,$NF}' employee.txt
ajay 45000
sunil 25000
varun 50000
amit 47000
tarun 15000
deepak 23000
sunil 13000
satvik 80000
himanshu@foreman:~/Desktop$
```

```

himanshu@foreman:~/Desktop$ awk '{print NR,$0}' employee.txt
1 ajay manager account 45000
2 sunil clerk account 25000
3 varun manager sales 50000
4 amit manager account 47000
5 tarun peon sales 15000
6 deepak clerk sales 23000
7 sunil peon sales 13000
8 satvik director purchase 80000
himanshu@foreman:~/Desktop$ awk '{print $1,$NF}' employee.txt
ajay 45000
sunil 25000
varun 50000
amit 47000
tarun 15000
deepak 23000
sunil 13000
satvik 80000
himanshu@foreman:~/Desktop$ awk 'NR==3, NR==6 {print NR,$0}' employee.txt
3 varun manager sales 50000
4 amit manager account 47000
5 tarun peon sales 15000
6 deepak clerk sales 23000
himanshu@foreman:~/Desktop$ █

```

```

himanshu@foreman:~/Desktop$ cat > geeksforgeeks.txt
A      B      C
Tarun   A12    1
Man     B6      2
Praveen M42    3
^C
himanshu@foreman:~/Desktop$ awk '{print NR "- " $1 }' geeksforgeeks.txt
1- A
2- Tarun
3- Man
4- Praveen
himanshu@foreman:~/Desktop$ █

```

```

himanshu@foreman:~/Desktop$ cat > geeksforgeeks.txt
A      B      C
Tarun  A12    1
Man    B6      2
Praveen M42    3
^C
himanshu@foreman:~/Desktop$ awk '{print NR "- " $1 }' geeksforgeeks.txt
1- A
2- Tarun
3- Man
4- Praveen
himanshu@foreman:~/Desktop$ awk '{print $2}' geeksforgeeks.txt
B
A12
B6
M42
himanshu@foreman:~/Desktop$ awk 'NF < 0' geeksforgeeks.txt
himanshu@foreman:~/Desktop$ ls
employee.txt  geeksforgeeks.txt
himanshu@foreman:~/Desktop$ awk '{ if (length($0) > max) max = length($0) } END { print max
}' geeksforgeeks.txt
19
himanshu@foreman:~/Desktop$ █

```

```

himanshu@foreman:~/Desktop$ awk 'END { print NR }' geeksforgeeks.txt
4
himanshu@foreman:~/Desktop$ awk 'length($0) > 10' geeksforgeeks.txt
A      B      C
Tarun  A12    1
Man    B6      2
Praveen M42    3
himanshu@foreman:~/Desktop$ awk '{ if($3 == "B6") print $0;}' geeksforgeeks.txt
himanshu@foreman:~/Desktop$ awk 'BEGIN { for(i=1;i<=6;i++) print "square of", i, "is",i*i;
}'
square of 1 is 1
square of 2 is 4
square of 3 is 9
square of 4 is 16
square of 5 is 25
square of 6 is 36
himanshu@foreman:~/Desktop$ █

```

. SED (STREAM EDITOR) COMMAND

SED is a text editor used on unix systems to edit files quickly and efficiently . The tool searches Through , Replaces , adds , and deletes lines in a text file without opening the File in a text editor . Learn how to use the SED command and its options through easy - to - Follow examples .

```
himanshu@foreman:~/Desktop$ sed
Usage: sed [OPTION]... {script-only-if-no-other-script} [input-file]...

  -n, --quiet, --silent           suppress automatic printing of pattern space
      --debug                     annotate program execution
  -e script, --expression=script  add the script to the commands to be executed
  -f script-file, --file=script-file
                                  add the contents of script-file to the commands to be executed
  --follow-symlinks               follow symlinks when processing in place
  -i[SUFFIX], --in-place[=SUFFIX] edit files in place (makes backup if SUFFIX supplied)
  -l N, --line-length=N          specify the desired line-wrap length for the 'l' command
  --posix                         disable all GNU extensions.
  -E, -r, --regexp-extended      use extended regular expressions in the script
                                  (for portability use POSIX -E).
  -s, --separate                 consider files as separate rather than as a single,
                                  continuous long stream.
      --sandbox                  operate in sandbox mode (disable e/r/w commands).
  -u, --unbuffered               load minimal amounts of data from the input files and flush
                                  the output buffers more often
  -z, --null-data                separate lines by NUL characters
      --help                    display this help and exit
      --version                 output version information and exit
```

If no -e, --expression, -f, or --file option is given, then the first non-option argument is taken as the sed script to interpret. All remaining arguments are names of input files; if no input files are specified, then the standard input is read.

GNU sed home page: <<https://www.gnu.org/software/sed/>>.
General help using GNU software: <<https://www.gnu.org/gethelp/>>.

```
himanshu@foreman:~/Desktop$ cat > geekfile.txt
unix is great os. unix is opensource. unix is free os.
learn operating system.
unix linux which one you choose.
unix is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.
^C
himanshu@foreman:~/Desktop$ sed 's/unix/linux/' geekfile.txt
linux is great os. unix is opensource. unix is free os.
learn operating system.
linux linux which one you choose.
linux is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.
himanshu@foreman:~/Desktop$
```



```
himanshu@foreman:~/Desktop$ sed 's/unix/linux/2' geekfile.txt
unix is great os. linux is opensource. unix is free os.
learn operating system.
unix linux which one you choose.
unix is easy to learn.linux is a multiuser os.Learn unix .unix is a powerful.
himanshu@foreman:~/Desktop$
```

```
himanshu@foreman:~/Desktop$ sed 's/unix/linux/2' geekfile.txt
unix is great os. linux is opensource. unix is free os.
learn operating system.
unix linux which one you choose.
unix is easy to learn.linux is a multiuser os.Learn unix .unix is a powerful.
himanshu@foreman:~/Desktop$ sed 's/unix/linux/g' geekfile.txt
linux is great os. linux is opensource. linux is free os.
learn operating system.
linux linux which one you choose.
linux is easy to learn.linux is a multiuser os.Learn linux .linux is a powerful.
himanshu@foreman:~/Desktop$ sed 's/unix/linux/3g' geekfile.txt
unix is great os. unix is opensource. linux is free os.
learn operating system.
unix linux which one you choose.
unix is easy to learn.unix is a multiuser os.Learn linux .linux is a powerful.
himanshu@foreman:~/Desktop$
```

```
unix is easy to learn.unix is a multiuser os.Learn linux .linux is a powerful.
himanshu@foreman:~/Desktop$ echo "Welcome To The Geek Stuff" | sed 's/\(\b[A-Z]\)/\(\1\)/g'
(W)elcome (T)o (T)he (G)eek (S)tuff
himanshu@foreman:~/Desktop$ sed '3 s/unix/linux/' geekfile.txt
unix is great os. unix is opensource. unix is free os.
learn operating system.
linux linux which one you choose.
unix is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.
himanshu@foreman:~/Desktop$
```

```
himanshu@foreman:~/Desktop$ sed 's/unix/linux/p' geekfile.txt
linux is great os. unix is opensource. unix is free os.
linux is great os. unix is opensource. unix is free os.
learn operating system.
linux linux which one you choose.
linux linux which one you choose.
linux is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.
linux is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.
himanshu@foreman:~/Desktop$ sed -n 's/unix/linux/p' geekfile.txt
linux is great os. unix is opensource. unix is free os.
linux linux which one you choose.
linux is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.
himanshu@foreman:~/Desktop$ sed '1,3 s/unix/linux/' geekfile.txt
linux is great os. unix is opensource. unix is free os.
learn operating system.
linux linux which one you choose.
linux is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.
himanshu@foreman:~/Desktop$ sed '2,$ s/unix/linux/' geekfile.txt
unix is great os. unix is opensource. unix is free os.
learn operating system.
linux linux which one you choose.
linux is easy to learn.unix is a multiuser os.Learn unix .unix is a powerful.
himanshu@foreman:~/Desktop$
```