

Assignment-7

Awk with built in variable

Use Case : Create a use case using built-in variable

Point 1: When we want to print the last line of the record in the file, we can use the awk command with the variable NF.

```
bhushan@ubuntu:~/Hardsoft$ cat file4.txt
ID      Name      Salary    Country
[101]   -Rutu     -25000    -India
[102]   -Bont     45000     Belgium
[103]   -Loki     55000     Germany
[104]   -Hina     35000     India
bhushan@ubuntu:~/Hardsoft$ awk '{print $NF}' file4.txt
Country
-India
Belgium
Germany
India
bhushan@ubuntu:~/Hardsoft$
```

Point 2: When we want to count the number of input records. Generally, lines are considered as records.

```
India
bhushan@ubuntu:~/Hardsoft$ awk '{print NR "-" $2}' file4.txt
1-Name
2--Rutu
3--Bont
4--Loki
5--Hina
bhushan@ubuntu:~/Hardsoft$ awk '{print NR "-" $4}' file4.txt
1-Country
2--India
3-Belgium
4-Germany
5-India
bhushan@ubuntu:~/Hardsoft$
```

Point 3: AWK command with OFS variable is used to store the output field separator when AWK prints the output. A blank space character is treated as a default field separator.

```
5 India
bhushan@ubuntu:~/Hardsoft$ date | awk 'OFS="/" {print $2, $3, $6}'
02/Mar/PM
bhushan@ubuntu:~/Hardsoft$ date | awk 'OFS="/" {print $1,$2,$3,$4,$5,$6}'
Thu/02/Mar/2023/08:16:42/PM
bhushan@ubuntu:~/Hardsoft$ date | awk 'OFS="-" {print $1,$2,$3,$4,$5,$6}'
Thu-02-Mar-2023-08:18:00-PM
bhushan@ubuntu:~/Hardsoft$
```

Point 4: To count the number of lines in the input file, we can use the END variable with the awk command

```
bhushan@ubuntu:~/Hardsoft$ awk 'END {print NR}' file4.txt
5
bhushan@ubuntu:~/Hardsoft$ awk 'END {print NR}' file3.txt
10
bhushan@ubuntu:~/Hardsoft$ cat file3.txt
Windows OS
Fedora
Linux
REd-Hat
MAc-OS
Solaris
Android
Sandwish
Hello How are you!

bhushan@ubuntu:~/Hardsoft$ cat file4.txt
  ID      Name      Salary      Country
[101]    -Rutu      -25000      -India
[102]    -Bont      45000      Belgium
[103]    -Loki      55000      Germany
[104]    -Hina      35000      India
bhushan@ubuntu:~/Hardsoft$
```

Point 5: BEGIN variable is used to set actions before any records have been executed. We can also print any data that we want to print before the records are processed

```
bhushan@ubuntu:~/Hardsoft$ awk 'BEGIN {print "Starting of the line"}; {print $2}; ' file4.txt
Starting of the line
Name
-Rutu
-Bont
-Loki
-Hina
bhushan@ubuntu:~/Hardsoft$
```

Point 6: The OFS variable is the space, you can set the OFS variable to specify the separator you need:

```
bhushan@ubuntu:~/Hardsoft$ awk 'BEGIN{FS=":"; OFS="-"} {print $1,$6,$7}' /etc/passwd
root-/root-/bin/bash
daemon-/usr/sbin-/usr/sbin/nologin
bin-/bin-/usr/sbin/nologin
sys-/dev-/usr/sbin/nologin
sync-/bin-/bin/sync
games-/usr/games-/usr/sbin/nologin
man-/var/cache/man-/usr/sbin/nologin
lp-/var/spool/lpd-/usr/sbin/nologin
mail-/var/mail-/usr/sbin/nologin
news-/var/spool/news-/usr/sbin/nologin
uucp-/var/spool/uucp-/usr/sbin/nologin
proxy-/bin-/usr/sbin/nologin
www-data-/var/www-/usr/sbin/nologin
backup-/var/backups-/usr/sbin/nologin
list-/var/list-/usr/sbin/nologin
irc-/var/run/ircd-/usr/sbin/nologin
gnats-/var/lib/gnats-/usr/sbin/nologin
nobody-/nonexistent-/usr/sbin/nologin
systemd-network-/run/systemd-/usr/sbin/nologin
systemd-resolve-/run/systemd-/usr/sbin/nologin
systemd-timesync-/run/systemd-/usr/sbin/nologin
messagebus-/nonexistent-/usr/sbin/nologin
```

Use case : Create a use case to combine patterns using logical operators.

Point 1: Logical And Operator &&

```
bhushan@ubuntu:~/Hardsoft$ awk 'NR>1 && NR<4' file4.txt
[101]      -Rutu      -25000      -India
[102]      -Bont      45000      Belgium
bhushan@ubuntu:~/Hardsoft$ awk 'NR>2 && NR<3' file4.txt
bhushan@ubuntu:~/Hardsoft$ awk 'NR>2 && NR<4' file4.txt
[102]      -Bont      45000      Belgium
bhushan@ubuntu:~/Hardsoft$
```

Point 2: Logical Or Operator ||

```
bhushan@ubuntu:~/Hardsoft$ awk 'NR>2 || NR<4' file4.txt
ID      Name      Salary      Country
[101]    -Rutu      -25000      -India
[102]    -Bont      45000      Belgium
[103]    -Loki      55000      Germany
[104]    -Hina      35000      India
bhushan@ubuntu:~/Hardsoft$
```

Point 3: Logical Not !

```
bhushan@ubuntu:~/Hardsoft$ awk 'NR==2' file4.txt
[101]    -Rutu      -25000      -India
bhushan@ubuntu:~/Hardsoft$ awk 'NR==4' file4.txt
[103]    -Loki      55000      Germany
bhushan@ubuntu:~/Hardsoft$ awk 'NR!=3' file4.txt
ID      Name      Salary      Country
[101]    -Rutu      -25000      -India
[103]    -Loki      55000      Germany
[104]    -Hina      35000      India
bhushan@ubuntu:~/Hardsoft$ awk 'NR!=2' file4.txt
ID      Name      Salary      Country
[102]    -Bont      45000      Belgium
[103]    -Loki      55000      Germany
[104]    -Hina      35000      India
bhushan@ubuntu:~/Hardsoft$
```