

**Ex. No: 3**

**Date: 28-08-2023**

## **SED AND AWK COMMANDS**

### **Aim:**

To learn the SED and AWK commands and their options in Linux.

### **Commands:**

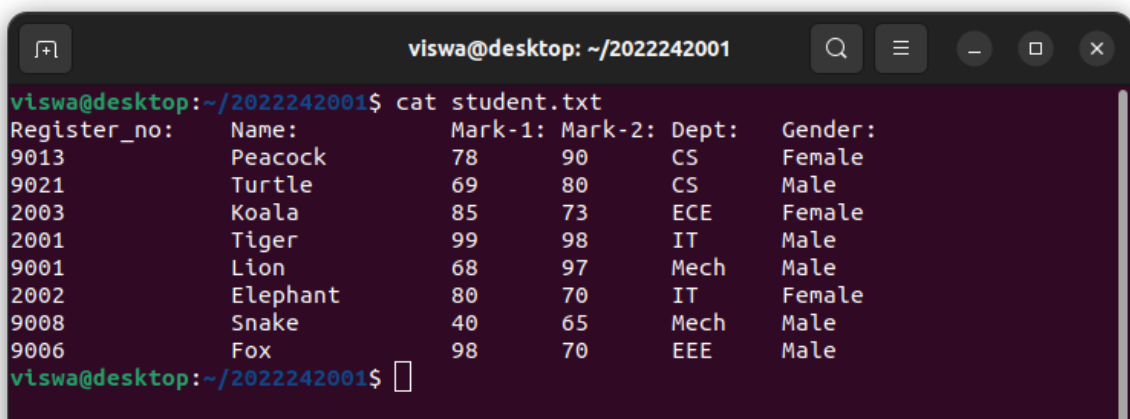
- **sed:**

**Description:** Sed is a stream editor. A stream editor is used to perform basic text transformations on an input stream (a file or input from a pipeline). While in some ways similar to an editor which permits scripted edits (Such as ed), sed works by making only one pass over the input(s), and is consequently more efficient. But it is sed's ability to filter text in a pipeline which particularly distinguishes it from other types of editors.

**Syntax:** ~\$ sed [options...] [script] [input files.....]

### **FILE:**

cat student.txt

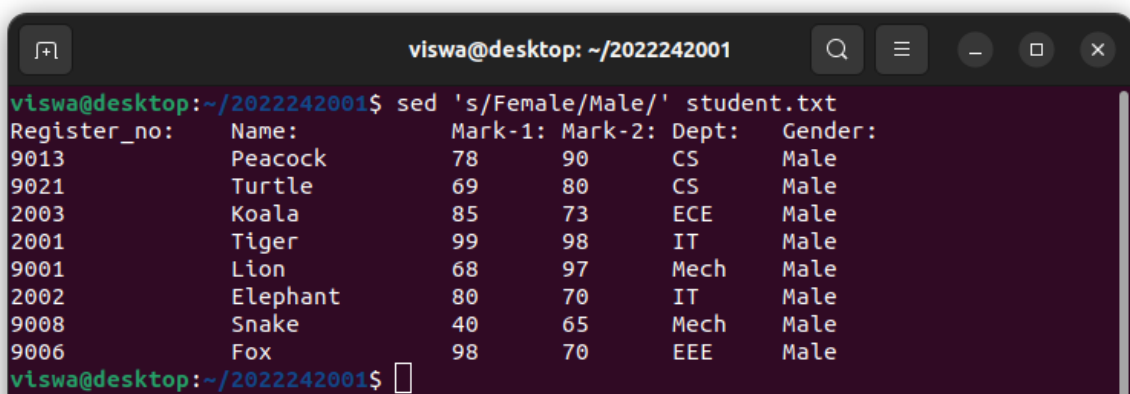


```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ cat student.txt
Register_no:    Name:      Mark-1: Mark-2: Dept:   Gender:
9013            Peacock    78      90      CS      Female
9021            Turtle     69      80      CS      Male
2003            Koala      85      73      ECE     Female
2001            Tiger      99      98      IT      Male
9001            Lion       68      97      Mech   Male
2002            Elephant   80      70      IT      Female
9008            Snake      40      65      Mech   Male
9006            Fox        98      70      EEE     Male
viswa@desktop:~/2022242001$
```

### **Options:**

#### **1. Replacing or substituting string:**

sed 's/Female/Male/' student.txt



```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ sed 's/Female/Male/' student.txt
Register_no:    Name:      Mark-1: Mark-2: Dept:   Gender:
9013            Peacock    78      90      CS      Male
9021            Turtle     69      80      CS      Male
2003            Koala      85      73      ECE     Male
2001            Tiger      99      98      IT      Male
9001            Lion       68      97      Mech   Male
2002            Elephant   80      70      IT      Male
9008            Snake      40      65      Mech   Male
9006            Fox        98      70      EEE     Male
viswa@desktop:~/2022242001$
```

## 2. Replacing the nth occurrence of a pattern in a line:

sed 's/Ma/Me/3' student.txt

```
viswa@desktop:~/2022242001$ sed 's/Ma/Me/3' student.txt
Register_no:   Name:      Mark-1: Mark-2: Dept:   Gender:
9013           Peacock    78      90      CS      Female
9021           Turtle     69      80      CS      Male
2003           Koala     85      73      ECE     Female
2001           Tiger     99      98      IT      Male
9001           Lion      68      97      Mech    Male
2002           Elephant  80      70      IT      Female
9008           Snake     40      65      Mech    Male
9006           Fox       98      70      EEE     Male
```

## 3. Replacing all occurrences of the pattern in a line:

sed 's/Ma/Me/g' student.txt

```
viswa@desktop:~/2022242001$ sed 's/Ma/Me/g' student.txt
Register_no:   Name:      Merk-1: Merk-2: Dept:   Gender:
9013           Peacock    78      90      CS      Female
9021           Turtle     69      80      CS      Mele
2003           Koala     85      73      ECE     Female
2001           Tiger     99      98      IT      Mele
9001           Lion      68      97      Mech    Mele
2002           Elephant  80      70      IT      Female
9008           Snake     40      65      Mech    Mele
9006           Fox       98      70      EEE     Mele
viswa@desktop:~/2022242001$
```

## 4. Replacing from nth occurrences to all occurrences in line:

sed 's/M/m/5g' student.txt

```
viswa@desktop:~/2022242001$ sed 's/M/m/5g' student.txt
Register_no:   Name:      Mark-1: Mark-2: Dept:   Gender:
9013           Peacock    78      90      CS      Female
9021           Turtle     69      80      CS      Male
2003           Koala     85      73      ECE     Female
2001           Tiger     99      98      IT      Male
9001           Lion      68      97      Mech    Male
2002           Elephant  80      70      IT      Female
9008           Snake     40      65      Mech    Male
9006           Fox       98      70      EEE     Male
viswa@desktop:~/2022242001$
```

## 5. Parenthesize first character of each word:

echo "Hello Everyone, This is from OS Lab" | sed 's/^(\\b[A-Z]\\)(\\1)/g'

```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ echo "Hello Everyone, This is from OS Lab" | sed 's/^(\\b[A-Z]\\)(\\1)/g'
(H)ello (E)veryone, (T)his is from (O)S (L)ab
viswa@desktop:~/2022242001$
```

## 6. Replacing string on a specific line number:

sed 's/Ma/Me/1g' student.txt

```
viswa@desktop:~/2022242001$ sed 's/Ma/Me/1g' student.txt
Register_no:      Name:      Merk-1: Merk-2: Dept:  Gender:
9013              Peacock    78      90      CS      Female
9021              Turtle     69      80      CS      Mele
2003              Koala     85      73      ECE     Female
2001              Tiger     99      98      IT      Mele
9001              Lion      68      97      Mech    Mele
2002              Elephant  80      70      IT      Female
9008              Snake     40      65      Mech    Mele
9006              Fox       98      70      EEE     Mele
viswa@desktop:~/2022242001$
```

## 7. Duplicating the replaced line with /p flag:

sed 's/G/g/p' student.txt

```
viswa@desktop:~/2022242001$ sed 's/G/g/p' student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:  gender:
Register_no:      Name:      Mark-1: Mark-2: Dept:  gender:
9013              Peacock    78      90      CS      Female
9021              Turtle     69      80      CS      Male
2003              Koala     85      73      ECE     Female
2001              Tiger     99      98      IT      Male
9001              Lion      68      97      Mech    Male
2002              Elephant  80      70      IT      Female
9008              Snake     40      65      Mech    Male
9006              Fox       98      70      EEE     Male
viswa@desktop:~/2022242001$
```

## 8. Printing only the replaced lines:

sed -n 's/Elephant/Mammoth/p' student.txt

```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ sed -n 's/Elephant/Mammoth/p' student.txt
2002              Mammoth  80      70      IT      Female
viswa@desktop:~/2022242001$
```

## 9. Replacing string on a range of lines:

i) sed '1,2 s/t/T/' student.txt

```
viswa@desktop:~/2022242001$ sed '1,2 s/t/T/' student.txt
RegisTer_no:      Name:      Mark-1: Mark-2: Dept:  Gender:
9013              Peacock    78      90      CS      Female
9021              Turtle     69      80      CS      Male
2003              Koala     85      73      ECE     Female
2001              Tiger     99      98      IT      Male
9001              Lion      68      97      Mech    Male
2002              Elephant  80      70      IT      Female
9008              Snake     40      65      Mech    Male
9006              Fox       98      70      EEE     Male
viswa@desktop:~/2022242001$
```

ii) sed '2,\$ s/Male/MALE/' student.txt

```
viswa@desktop:~/2022242001$ sed '2,$ s/Male/MALE/' student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:   Gender:
9013              Peacock    78      90      CS      Female
9021              Turtle     69      80      CS      MALE
2003              Koala     85      73      ECE     Female
2001              Tiger     99      98      IT      MALE
9001              Lion      68      97      Mech    MALE
2002              Elephant  80      70      IT      Female
9008              Snake     40      65      Mech    MALE
9006              Fox       98      70      EEE     MALE
viswa@desktop:~/2022242001$
```

## 10. Deleting lines from a particular file:

a. To delete a particular line say in file:

sed '4d' student.txt

```
viswa@desktop:~/2022242001$ sed '4d' student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:   Gender:
9013              Peacock    78      90      CS      Female
9021              Turtle     69      80      CS      Male
2001              Tiger     99      98      IT      Male
9001              Lion      68      97      Mech    Male
2002              Elephant  80      70      IT      Female
9008              Snake     40      65      Mech    Male
9006              Fox       98      70      EEE     Male
viswa@desktop:~/2022242001$
```

b. To delete a last line:

sed '\$d' student.txt

```
viswa@desktop:~/2022242001$ sed '$d' student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:   Gender:
9013              Peacock    78      90      CS      Female
9021              Turtle     69      80      CS      Male
2003              Koala     85      73      ECE     Female
2001              Tiger     99      98      IT      Male
9001              Lion      68      97      Mech    Male
2002              Elephant  80      70      IT      Female
9008              Snake     40      65      Mech    Male
viswa@desktop:~/2022242001$
```

c. To delete line from range x and y:

sed '2,3d' student.txt

```
viswa@desktop:~/2022242001$ sed '2,3d' student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:   Gender:
2003              Koala     85      73      ECE     Female
2001              Tiger     99      98      IT      Male
9001              Lion      68      97      Mech    Male
2002              Elephant  80      70      IT      Female
9008              Snake     40      65      Mech    Male
9006              Fox       98      70      EEE     Male
viswa@desktop:~/2022242001$
```

d. To delete from nth to last line:

sed '2,\$d' student.txt

```
viswa@desktop:~/2022242001$ sed '2,$d' student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:      Gender:
viswa@desktop:~/2022242001$
```

e. To delete pattern matching line:

sed '/Male/d' student.txt

```
viswa@desktop:~/2022242001$ sed '/Male/d' student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:      Gender:
9013              Peacock      78      90      CS      Female
2003              Koala      85      73      ECE      Female
2002              Elephant      80      70      IT      Female
viswa@desktop:~/2022242001$
```

## • awk:

**Description:** Awk is a scripting language used for manipulating data and generating reports. The awk command programming language requires no compiling and allows the user to use variables, numeric functions, string functions, and logical operators.

**Syntax:** ~\$ awk options 'selection\_criteria {action}' input-file > output-file

### Options:

#### 1. Default behaviour of AWK command:

awk '{print}' student.txt

```
viswa@desktop:~/2022242001$ awk '{print}' student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:      Gender:
9013              Peacock      78      90      CS      Female
9021              Turtle      69      80      CS      Male
2003              Koala      85      73      ECE      Female
2001              Tiger      99      98      IT      Male
9001              Lion      68      97      Mech      Male
2002              Elephant      80      70      IT      Female
9008              Snake      40      65      Mech      Male
9006              Fox      98      70      EEE      Male
viswa@desktop:~/2022242001$
```

#### 2. Print matching lines with pattern:

awk '/CS/{print}' student.txt

```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ awk '/CS/{print}' student.txt
9013              Peacock      78      90      CS      Female
9021              Turtle      69      80      CS      Male
viswa@desktop:~/2022242001$
```

### 3. Splitting a line into fields

`awk '{print $1,$4}' student.txt`

```
viswa@desktop:~/2022242001$ awk '{print $1,$4}' student.txt
Register_no: Mark-2:
9013 90
9021 80
2003 73
2001 98
9001 97
2002 70
9008 65
9006 70
viswa@desktop:~/2022242001$
```

#### Built in variables:

##### 1. NR:

Record current number of input counts.

`awk '{print NR, $1}' student.txt`

```
viswa@desktop:~/2022242001$ awk '{print NR,$1}' student.txt
1 Register_no:
2 9013
3 9021
4 2003
5 2001
6 9001
7 2002
8 9008
9 9006
viswa@desktop:~/2022242001$
```

##### 2. NF:

Record number of fields in current input.

`awk '{print NF, $1}' student.txt`

```
viswa@desktop:~/2022242001$ awk '{print NF,$1}' student.txt
6 Register_no:
6 9013
6 9021
6 2003
6 2001
6 9001
6 2002
6 9008
6 9006
viswa@desktop:~/2022242001$
```

##### 3. Display a range of lines with NR:

`awk 'NR==2, NR==5 {print NR, $0}' student.txt`

```
viswa@desktop:~/2022242001$ awk 'NR==2,NR==5{print NR,$0}' student.txt
2 9013          Peacock          78      90      CS      Female
3 9021          Turtle           69      80      CS      Male
4 2003          Koala            85      73      ECE     Female
5 2001          Tiger            99      98      IT      Male
viswa@desktop:~/2022242001$
```

4. To print first item with '-' separation:

```
awk '{print NR '-' $1}' student.txt
```

```
viswa@desktop:~/2022242001$ awk '{print NR '-' $1}' student.txt
1
-9011
-9018
-1999
-1996
-8995
-1995
-9000
-8997
viswa@desktop:~/2022242001$
```

5. To return second row/item:

```
awk '{print $2}' student.txt
```

```
viswa@desktop:~/2022242001$ awk '{print $2}' student.txt
Name:
Peacock
Turtle
Koala
Tiger
Lion
Elephant
Snake
Fox
viswa@desktop:~/2022242001$
```

6. To count the lines in a file:

```
awk '{print NR}' student.txt
```

```
viswa@desktop:~/2022242001$ awk '{print NR}' student.txt
1
2
3
4
5
6
7
8
9
viswa@desktop:~/2022242001$
```

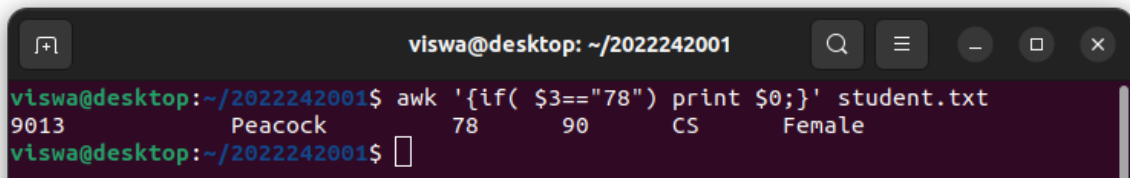
7. Print lines more than n characters:

```
awk 'length($0)>15' student.txt
```

```
viswa@desktop:~/2022242001$ awk 'length($0)>15' student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:  Gender:
9013      Peacock      78      90      CS      Female
9021      Turtle      69      80      CS      Male
2003      Koala      85      73      ECE      Female
2001      Tiger      99      98      IT      Male
9001      Lion      68      97      Mech      Male
2002      Elephant      80      70      IT      Female
9008      Snake      40      65      Mech      Male
9006      Fox      98      70      EEE      Male
viswa@desktop:~/2022242001$
```

8. To check for any string in any column:

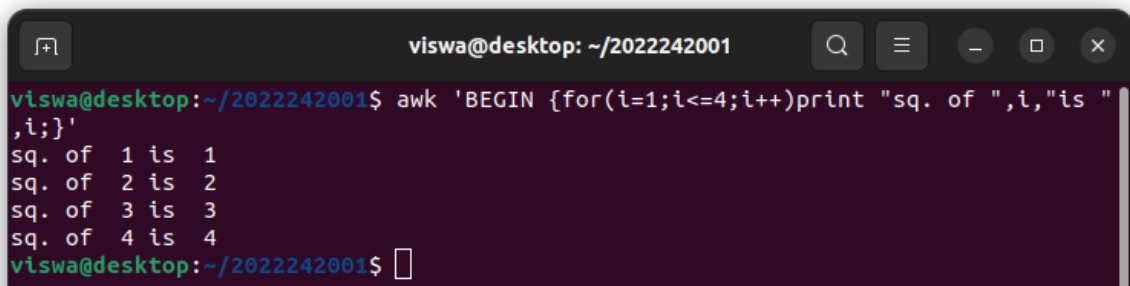
```
awk '{if( $3== "78")print $0;}' student.txt
```



```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ awk '{if( $3=="78") print $0;}' student.txt
9013          Peacock          78          90          CS          Female
viswa@desktop:~/2022242001$
```

9. To print square of 1st n numbers

```
awk 'BEGIN{for(i=1;i<=4;i++)print "sq. of " i "is" i*i;}'
```



```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ awk 'BEGIN {for(i=1;i<=4;i++)print "sq. of ",i,"is "
,i;}'
sq. of  1 is  1
sq. of  2 is  2
sq. of  3 is  3
sq. of  4 is  4
viswa@desktop:~/2022242001$
```

## Result:

Thus, the various options in Sed and Awk commands were learned in Linux.