UNIX:

1.CREATE A FILE

cat > filename.txt

enter the values

ctrl+D

cat filename.txt

2.TO DISPLAY FIRST TWO LINES:

head -n filename.txt

3.TO DISPALY LAST LINES:

tail -n filename.txt

4.TO SORT IN REVERSE ORDER

sort -r filenmae.txt

a.TO SORT IN ASCENDING ORDER

sort -r filenmae.txt

5.TO DISPLAY CONTENTS OF A FILE :

cat filename.txt

6. **Default behavior of Awk :**By default Awk prints every line of data from the specified file.

$ awk '{print}' employee.txt

**Output:**

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

7.**Print the lines which matches with the given pattern.**

$ awk '/manager/ {print}' employee.txt

**Output:**

ajay manager account 45000

varun manager sales 50000

amit manager account 47000

**8.Spliting a Line Into Fields :**For each record i.e line, the awk command splits the record delimited by whitespace character by default and stores it in the $n variables. If the line has 4 words, it will be stored in $1, $2, $3 and $4 respectively. Also, $0 represents the whole line.

$ awk '{print $1,$4}' employee.txt

**Output:**

ajay 45000

sunil 25000

varun 50000

amit 47000

tarun 15000

deepak 23000

sunil 13000

satvik 80000

9. **Use of NR built-in variables (Display Line Number)**

$ awk '{print NR,$0}' employee.txt

**Output:**

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

10. **Use of NF built-in variables (Display Last Field)**

$ awk '{print $1,$NF}' employee.txt

**Output:**

ajay 45000

sunil 25000

varun 50000

amit 47000

tarun 15000

deepak 23000

sunil 13000

satvik 80000

In the above example $1 represents Name and $NF represents Salary. We can get the Salary using $NF , where $NF represents last field.

11. **Another use of NR built-in variables (Display Line From 3 to 6)**

$ awk 'NR==3, NR==6 {print NR,$0}' employee.txt

**Output:**

3 varun manager sales 50000

4 amit manager account 47000

5 tarun peon sales 15000

6 deepak clerk sales 23000

**12.For the given text file:**

$cat > geeksforgeeks.txt

A B C

Tarun A12 1

Man B6 2

Praveen M42 3

**a.To print the first item along with the row number(NR) separated with ” – “ from each line in geeksforgeeks.txt:**

$ awk '{print NR "- " $1 }' geeksforgeeks.txt

1 - Tarun

2 – Manav

3 - Praveen

**b. To return the second row/item from geeksforgeeks.txt:**

$ awk '{print $2}' geeksforgeeks.txt

A12

B6

M42

**c. To print any non empty line if present**

$ awk 'NF > 0' geeksforgeeks.txt

0

**d. To find the length of the longest line present in the file:**

$ awk '{ if (length($0) > max) max = length($0) } END { print max }' geeksforgeeks.txt

13

**e. To count the lines in a file:**

$ awk 'END { print NR }' geeksforgeeks.txt

3

**f. Printing lines with more than 10 characters:**

$ awk 'length($0) > 10' geeksforgeeks.txt

Tarun A12 1

Praveen M42 3

**g. To find/check for any string in any column:**

$ awk '{ if($3 == "B6") print $0;}' geeksforgeeks.txt

**h. To print the squares of first numbers from 1 to n say 6:**

$ 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