Practical No- 5
AWK Editor

A)Create a file empdata, which contains the following fields:-

Fieldname Datatype Value

1. Employee name character
2. Employee code numeric starts with letter 'E'
3. Department code character MKT, HRD, PUR
4. Grade character A-C
5. Designation character manager, director, gm, executive
6. Years of experience numeric
7. Date of birth dd-mm-yy
8. Region character Pune, Mumbai etc…
9. Basic pay numeric

Insert at least five records in above file; character fields in each record may not be same in the same case. '~' is used as a field separator. Give commands for the following:-

```
Jai~E1~MKT~A~manager~6~01-01-75~Pune~16000
Raj~E2~HRD~B~director~5~01-01-75~Mumbai~15000
Kamal~E3~PUR~C~gm~2~02-03-86~Pune~9000
Anna~E4~MKT~D~manager~9~14-05-75~Mumbai~25000
Ram~E5~HRD~A~executive~7~23-10-80~Mumbai~30000
```

1. Display all employees who are manager in office.

Code: `[tybscit@localhost Desktop]$ awk '/manager/' empdata`

Output:

```
Jai~E1~MKT~A~manager~6~01-01-75~Pune~16000
Anna~E4~MKT~D~manager~9~14-05-75~Mumbai~25000
```

2. List the details of an employee "Jai" in Mumbai office.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '$1=="Raj" &&
$8=="Mumbai"'
empdata
```

Output: `Raj~E2~HRD~B~director~5~01-01-75~Mumbai~15000`

3. Display all employees who are not in the department MKT. display the output sorted on department code.

Code & Output:

```
[tybscit@localhost Desktop]$ awk -F "~" '$3!="MKT" empdata|sort -t
"~" -k3
```

4. Display all employees whose years of experience are more than 5.

Code: `[tybscit@localhost Desktop]$ awk -F "~" '$6>"5"' empdata`

Output:

```
Jai~E1~MKT~A~manager~6~01-01-75~Pune~16000
Anna~E4~MKT~D~manager~9~14-05-75~Mumbai~25000
```

5. List only employee name, department code and basic pay of employees who are executive.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '$5=="executive" {print
$1,$3,$9}' empdata
```
Output:
```
Ram HRD 30000
```

6. Display all employees having grade 'A'.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '$4=="A" {print $1}'
empdata
```
Output:
```
Jai
Ram
```

7. Count total number of employees whose department code is HRD.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '$3=="HRD"
> {cnt=cnt+1}
> end{
> printf "totalnumber of employees whose department code is 'HRD' is
%d
\n",cnt}' empdata
```
Output:
```
Raj~E2~HRD~B~director~5~01-01-75~Mumbai~15000
Ram~E5~HRD~A~executive~7~23-10-80~Mumbai~30000
```

8. Display employee's names with salary above 10,000.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '$9>"10000"{print $1}'
empdata
```
Output:
```
Jai
Raj
Anna
Ram
```

9. Display only designation and basic pay of employees having number of experience between three
and 5.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '"$6>3 && $6<5"{print
$5,$9}'empdata
```
Output:
```
gm 9000
```

10. Find the number of employees in Pune office.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '$8=="Pune"
> {cnt=cnt+1}
> end{
> printf "Number of employees in Pune office is %d/n",cnt}' empdata
```
Output:
```
Jai~E1~MKT~A~manager~6~01-01-75~Pune~16000
Kamal~E3~PUR~C~gm~2~02-03-86~Pune~9000
```

11. Display employees who get basic pay less than 15000, also calculate and display average basic
pay.

Code:

```
[tybscit@localhost Desktop]$ awk 'BEGIN {FS="~"
> OFS=":"
> printf "\nDetails\n"
> } $9<15000{
> cnt=cnt+1; add=add+$9
> print NR,NF,$0
> }END{
> avg=add/cnt
> printf "The average salary of employees is %d\n",avg}' empdata
```

Output:

```
Details
3:9:Kamal~E3~PUR~C~gm~2~02-03-86~Pune~9000
The average salary of employees is 9000
```

12. Locate the employees with same date of birth in Pune office.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '$8=="Pune" && $7=="01-
01-
75"{print NR, $1}' empdata
```

Output:
```
1 Jai
```

13. Locate all for Raj, Jai and Kamal as employee name.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~"
'$1=="Raj"||$1=="Jai"||$1=="Kamal"' empdata
```

Output:

```
Jai~E1~MKT~A~manager~6~01-01-75~Pune~16000
Raj~E2~HRD~B~director~5~01-01-75~Mumbai~15000
Kamal~E3~PUR~C~gm~2~02-03-86~Pune~9000
```

14. Locate all except for Raj, Jai and Kamal as employee name.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '$1!~/Raj|Jai|Kamal/'
empdata
```

Output:

```
Anna~E4~MKT~D~manager~9~14-05-75~Mumbai~25000
Ram~E5~HRD~A~executive~7~23-10-80~Mumbai~30000
```

**15. Find the employees who have designation as director and find the 40% of basic pay as da and
15% of basic pay as hra.**

Code:

```
[tybscit@localhost Desktop]$ awk -F "~" '$5=="director"{
>da=$9*0.04;hra=$9*0.15
> print nr,nf,$1,$9,da,hra}' empdata
```

Output:
```
Raj 15000 600 2250
```

16. Store employee name and date of birth in a file’ nbdata’.

Code:
```
[tybscit@localhost Desktop]$ awk -F "~" '{print
$1,$7}'empdata|tee nbdata
```

Output:

```
Jai 01-01-75
Raj 01-01-75
Kamal 02-03-86
Anna 14-05-75
Ram 23-10-80
```

B) Create a file student with following fields:-

Fieldname Datatype Value

Student code numeric
Student name character
Batch code character B11-B15
No. of modules numeric 1-5
Average marks numeric

Insert at least five records in above file; ':' is used as a field separator. Give commands for the following:-

```
[tybscit@localhost Desktop]$ cat>stud
1:Anna:B11:5:90
2:Ram:B12:7:88
3:Hugh:B13:4:70
4:Tom:B14:1:55
5:Leo:B15:4:65
```

1. Display the details of student in order of their names ignoring case.

Code:
```
[tybscit@localhost Desktop]$ awk '{print}' stud|sort -t ":" -k
2 -f
```

Output:
```
1:Anna:B11:5:90
3:Hugh:B13:4:70
5:Leo:B15:4:65
2:Ram:B12:7:88
4:Tom:B14:1:55
```

2. Display the details of students whose number of modules is greater than 3.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" '$4>3{print "%s\n",$0}'
stud
```

Output:
```
%s
1:Anna:B11:5:90
%s
2:Ram:B12:7:88
%s
3:Hugh:B13:4:70
%s
5:Leo:B15:4:65
```

3. Store the list of rank holders in file 'merit' along with student code and student name, and marks & display its contents.

Code:
```
[tybscit@localhost Desktop]$ sort -t ":" -k5 -r -n stud|head
-3|awk -F
":" '{print $1,$2,$5}'>merit
```

Output:
```
[tybscit@localhost Desktop]$ cat merit
1 Anna 90
2 Ram 88
3 Hugh 70
```

4. Count the number of students in batch B13.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" '$3=="B13"{
```
Output:
```
>cnt=cnt+1
> print NR,NF,$2
> }END{
>printf "Number of students in B13 is %d\n",cnt}' stud
3 5 Hugh
Number of students in B13 is 1
```

5. Display the names of students with same names.

Code & Output:
```
[tybscit@localhost Desktop]$ awk -F ":"
'$2~/[aA][bB][cC]/' stud
```

6. Display the students belonging to batch codes B12 or B15.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" '$3=="B12" ||
$3=="B15"{print
$1}' stud
```
Output:
```
2
5
```

7. Display all the names not starting with ‘a’ or ‘A’.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" '$2~/^[^aA]/' stud
```
Output:
```
2:Ram:B12:7:88
3:Hugh:B13:4:70
4:Tom:B14:1:55
5:Leo:B15:4:65
```

8. Display all the names starting with ‘a’ or ‘A’.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" '$2~/^[aA]/' stud
```
Output:
```
1:Anna:B11:5:90
```

**9. Display and count the number of students having marks in the range 40 to 60. Also display the
total and average marks.**

Code:
```
[tybscit@localhost Desktop]$ awk 'BEGIN{FS=":"
> OFS=":"
>printf "Details\n"
> }$5>=40 && $5<=60{
>cnt=cnt+1
>tot=tot+$5
> print NR,$2
> }END{
>avg=tot/cnt
>printf "Total marks:%d\n",tot
>printf "Average marks :%d\n",avg
>printf "Total students :%d\n",cnt}' stud
```
Output:
```
Details
4:Tom
```

```
Total marks:55
Average marks :55
Total students :1
```

10. Display the student's records from line number 2 to 4.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" 'NR==2,NR==4{printf
"%d%s\n",NR,$0}' stud
```
Output:
```
2 2:Ram:B12:7:88
3 3:Hugh:B13:4:70
4 4:Tom:B14:1:55
```

11. Display the student's records that are having number of fields 5.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" 'NF==5{print $0}' stud
```
Output:
```
1:Anna:B11:5:90
2:Ram:B12:7:88
3:Hugh:B13:4:70
4:Tom:B14:1:55
5:Leo:B15:4:65
```

12. Display the student's records that are having number of fields less than or equal to 4.

Code & Output:
```
[tybscit@localhost Desktop]$ awk -F ":" 'NF<=4{printf
$0}' stud
```

13. Display the student code, student name and marks that are having number of fields greater than 5.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" 'NF<=5{print $1,$2,$5}'
stud
```
Output:
```
1 Anna 90
2 Ram 88
3 Hugh 70
4 Tom 55
5 Leo 65
```

14. Display the student's name having the length greater than 3.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" 'length($2)>3{print
$1,$2,$5}'
stud
```
Output:
```
1 Anna 90
2 Ram 88
3 Hugh 70
```

15. Display the student's records having the length of student name less than or equal to 3.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" 'length($2)<=3{print
$1,$2,$5}'
stud
```
Output:
```
4 Tom 55
```

```
5 Leo 65
```

16. Display the student’s records having the length less than or equal to 15.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" 'length($2)<=15{print
$1,$2,$5}'
stud
```
Output:
```
1 Anna 90
2 Ram 88
3 Hugh 70
4 Tom 55
5 Leo 65
```

17. Display the student’s records having the length greater than or equal to 15.

Code & Output:
```
[tybscit@localhost Desktop]$ awk -F ":" 'length($2)>=15{print
$1,$2,$5}' stud
```

18. Display the student’s records having the length is in the range 5 to 15.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" 'length >=5 &&
length<=15{print
$0}' stud
```
Output:
```
1:Anna:B11:5:90
3:Hugh:B13:4:70
4:Tom:B14:1:55
5:Leo:B15:4:65
```

19. Display the line number and index having character ‘b’ in student name.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" '{print
NR,index($2,"b")}' stud
```
Output:
```
1 0
2 0
3 0
4 0
5 0
```

20. Display the line number, student name and index having character ‘b’ in student name.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" '{print
NR,index($2,"b"),$2}'
stud
```
Output:
```
1 0 Anna
2 0 Ram
3 0 Hugh
4 0 Tom
5 0 Leo
```

21. Display the index, student name and marks having character ‘b’ in student name.

Code:
```
[tybscit@localhost Desktop]$ awk -F ":" '{print
index($2,"b"),$2,$5}'
```

```
stud
```

Output:

```
0 Anna 90
0 Ram 88
0 Hugh 70
0 Tom 55
0 Leo 65
```

22. Display the line number, index, student name and marks having character ‘b’ in student name.

Code:

```
[tybscit@localhost Desktop]$ awk -F ":" '{print
NR,index($2,"b"),$2,$5}'
stud
```

Output:

```
1 0 Anna 90
2 0 Ram 88
3 0 Hugh 70
4 0 Tom 55
5 0 Leo 65
```