**Sort**

SORT command is used to sort a file, arranging the records in a particular order.

* The sort command is a command line utility for sorting lines of text files. It supports sorting alphabetically, in reverse order, by number, by month and can also remove duplicates.
* By default, the entire input is taken as sort key. Blank space is the default field separator.
* Lines starting with a number will appear before lines starting with a letter.
* Lines starting with a lowercase letter will appear before lines starting with the same letter in uppercase.

*Syntax :*

$ sort filename.txt

Options with sort function

* -o Option

Using the -o option is functionally the same as redirecting the output to a file.

*Syntax :*

$ sort -o filename.txt inputfile.txt

* -r Option

You can perform a reverse-order sort using the -r flag.

*Syntax :*

$ sort -r inputfile.txt

* -n Option

To sort a file numerically used –n option. -n option is also predefined in unix as the above options are

* -nr option

To sort a file with numeric data in reverse order we can use the combination of two options as stated below.

*Syntax :*

$ sort -nr filename.txt

* -k Option

Unix provides the feature of sorting a table on the basis of any column number by using -k option.

Syntax :

$ sort -k filename.txt

* -c option

This option is used to check if the file given is already sorted or not & checks if a file is already sorted pass the -c option to sort

Syntax :

$ sort -c filename.txt

* -u option

To sort and remove duplicates pass the -u option to sort.

Syntax :

$ sort -u filename.txt

* -M Option:

To sort by month pass the -M option to sort. This will write a sorted list to standard output ordered by month name.

Syntax :

$ sort -M filename.txt

**Cut**

The cut command in UNIX is a command for cutting out the sections from each line of files and writing the result to standard output.

* It can be used to cut parts of a line by byte position, character and field.
* It is necessary to specify option with command otherwise it gives error.

*Syntax*:

cut OPTION... [FILE]...

Options used:

* -b(byte):

To extract the specific bytes, -b option with the list of byte numbers separated by comma. Range of bytes can also be specified using the hyphen(-). Tabs and backspaces are treated like as a character of 1 byte.

* -c (column):

To cut by character use the -c option. This selects the characters given to the -c option.

*Syntax*: $cut -c [(k)-(n)/(k),(n)/(n)] filename

* -f (field):

List of the fields number specified must be separated by comma. Ranges are not described with -f option. cut uses tab as a default field delimiter but can also work with other delimiter by using -d option.

Syntax:

$cut -d "delimiter" -f (field number) file.txt

* –complement:

As the name suggests it complement the output. This option can be used in the combination with other options either with -f or with -c.

$ cut --complement -d " " -f 1 filename

$ cut --complement -c 5 filename

* –output-delimiter:

By default the output delimiter is same as input delimiter that we specify in the cut with -d option. To change the output delimiter use the option –output-delimiter=”delimiter”.

$ cut -d " " -f 1,2 filename --output-delimiter='%'

**Touch**

The touch command is a standard command used in UNIX/Linux operating system which is used to create, change and modify timestamps of a file.

There are two different commands to create a file:

1. **Cat command:** It is used to create the file with content
2. **Touch command**: It is used to create a file without any content. The file created using touch command is empty. This command can be used when the user doesn’t have data to store at the time of file creation.

*Syntax*: touch file\_name

**Touch command to create multiple files:**

*Syntax*:

touch File1\_name File2\_name File3\_name

Options used:

* touch -a: This command is used to change access time only. To change or update the last access or modification times of a file touch -a command is used.

*Syntax*:

touch -a filename

* touch -c : This command is used to check whether a file is created or not. If not created then don’t create it. This command avoids creating files.

*Syntax:* touch -c filename

* touch -c-d : This is used to update access and modification time.

*Syntax*:

touch -c-d filename

* touch -m : This is used to change the modification time only. It only updates last modification time.

*Syntax*:

touch -m filename

* touch -r : This command is used to use the timestamp of another file. Here Doc2 file is updated with the time stamp of File 1.

*Syntax*:

touch -r second\_file\_name first\_file\_name

* touch -t : This is used to create a file using a specified time.

*Syntax*:

touch -t YYMMDDHHMM filename

**Until**

Until command in Linux used to execute a set of commands as long as the final command in the ‘until’ Commands has an exit status which is not zero. It is mostly used where the user needs to execute a set of commands until a condition is true.

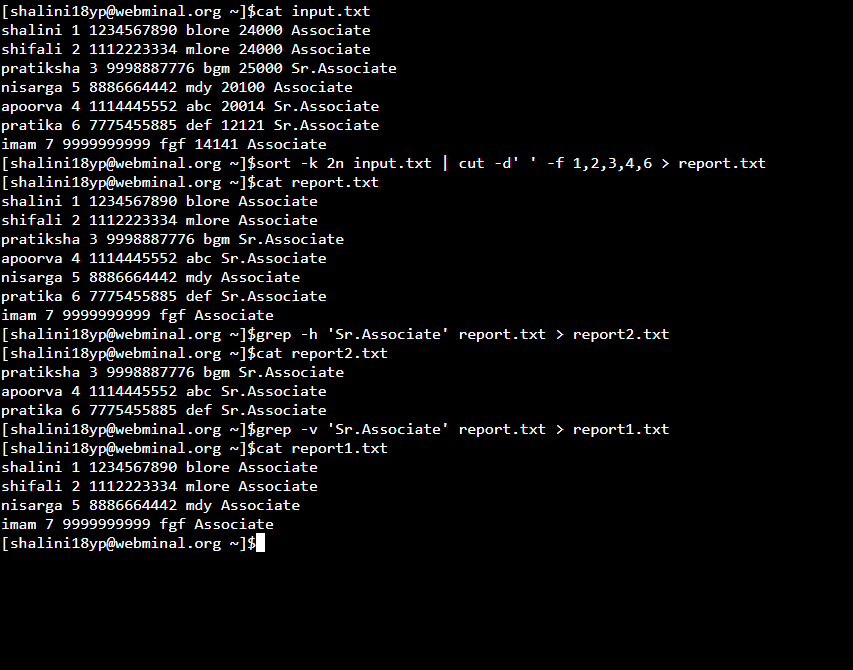
*Syntax*:

until COMMANDS; do COMMANDS; done

* Here, if the COMMANDS get evaluated to false then the statements will be executed. If the COMMANDS get evaluated to true then the no statements will be executed and control will go after the done statement.

Assignment 01

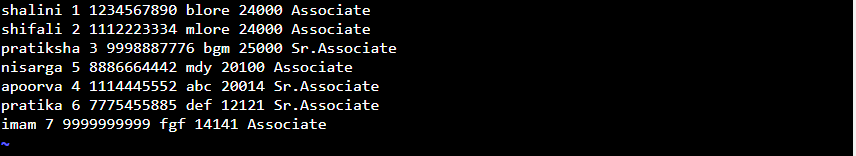
1. Use an input file with 5 columns: Name, SSN, Phone#, City and salary.
2. Pass a parameter which will be a column number.
3. Sort the file based on passed parameter.
4. Create an output file with 1st, 2nd and last column with the sorted data.

****

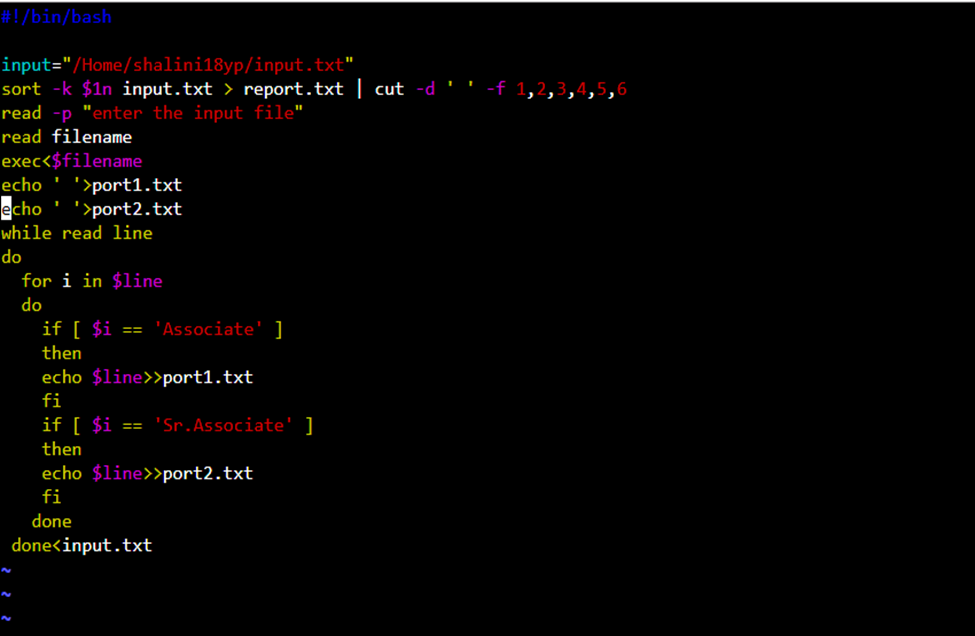
Assignment 02

1. Use  an input file with 6 columns: Name, SSN, Phone#, City , salary,Designation.  
    1.1 Designation should be (Associate and Sr. Associate).
2. Pass a parameter which will be a column number.
3. Sort the file based on passed parameter.
4. Create an output file (Report.txt) with 1st, 2nd, ,3rd, 4th and last column with the sorted data.
5. Create 2 ouput files "Report1" & "Report2,  based on below conditions  
   5.1 If the record in "Report.txt" is having designation as "Associate", then put that record in Report1.txt   
   5.2 If the record in "Report.txt" is having designation as "Sr.Associate", then put that record in Report2.txt

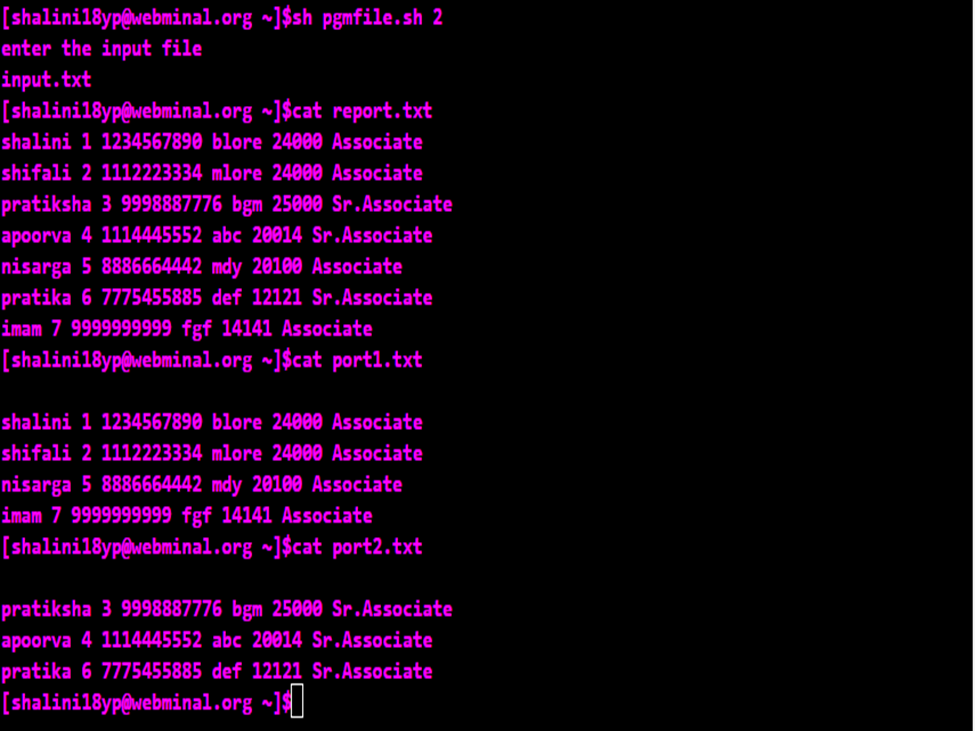
Input file



Shell Script



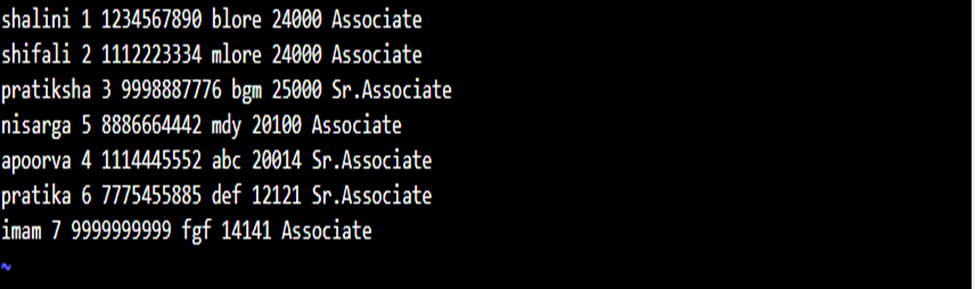
Output



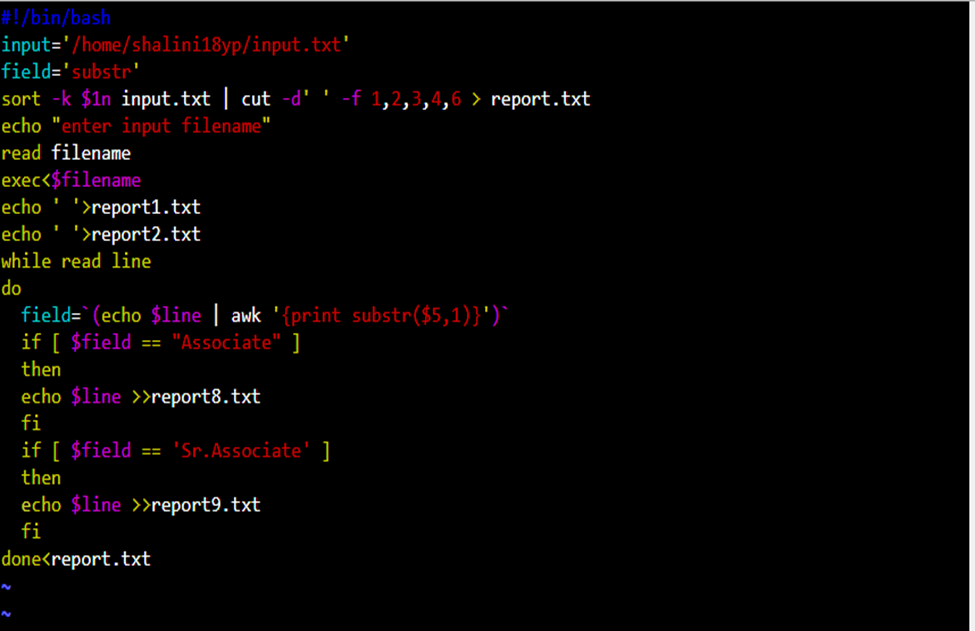
Assignment 03

1. Use  an input file with 6 columns: Name, SSN, Phone#, City , salary,Designation.  
    1.1 Designation should be (Associate and Sr. Associate).
2. Pass a parameter which will be a column number.
3. Sort the file based on passed parameter.
4. Create an output file (Report.txt) with 1st, 2nd, ,3rd, 4th and last column with the sorted data.
5. Create 2 ouput files "Report1" & "Report2,  based on below conditions  
   5.1 If the record in "Report.txt" is having designation as "Associate", then put that record in Report1.txt   
   5.2 If the record in "Report.txt" is having designation as "Sr.Associate", then put that record in Report2.txt
6. Using SUBSTRING.

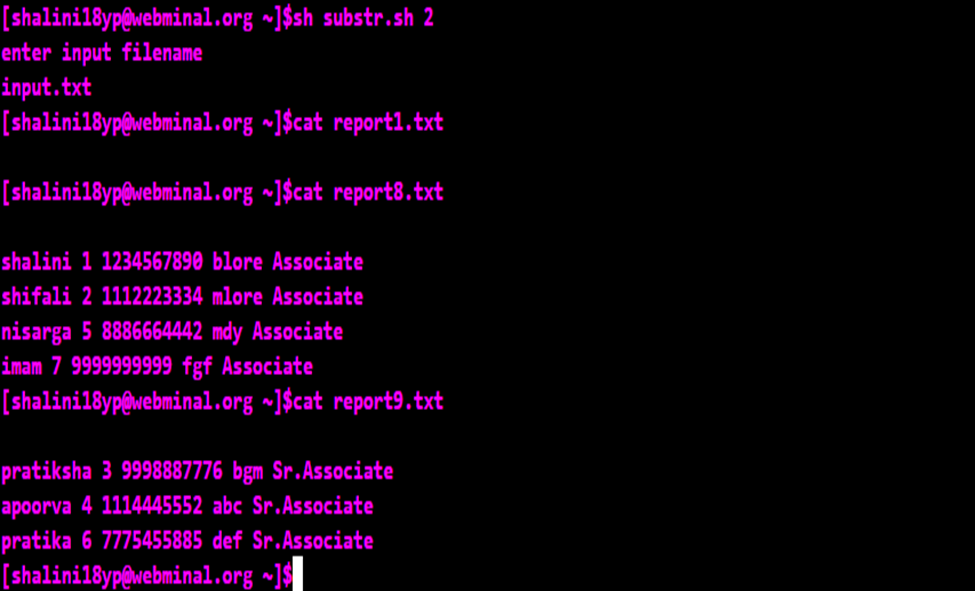
Input file



Shell Script



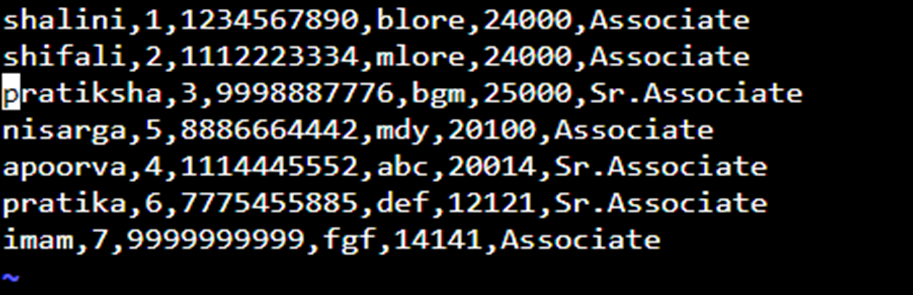
Output



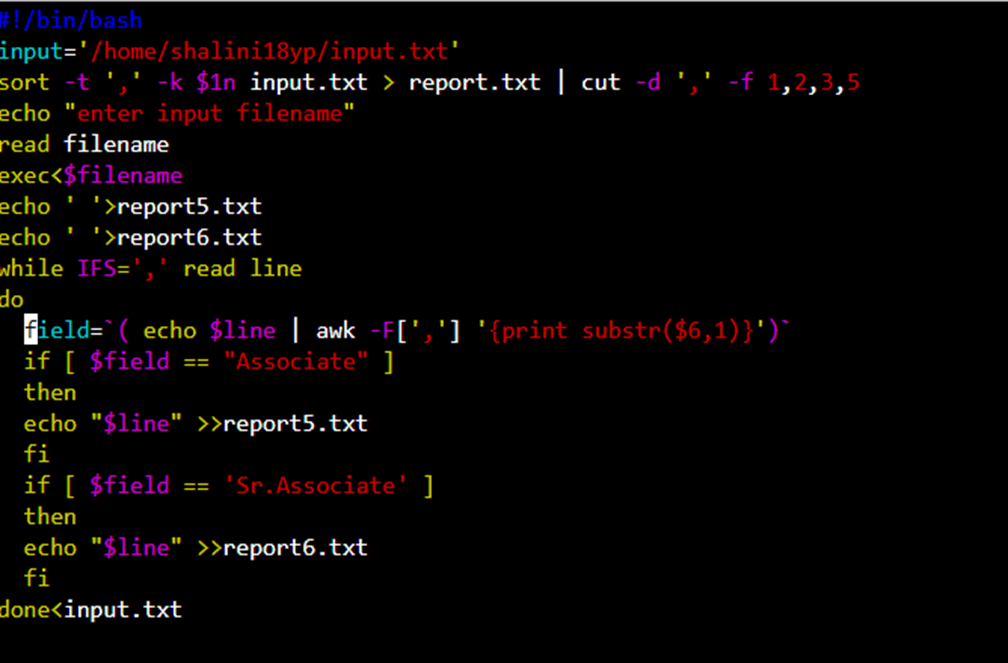
Assignment 03

1. Use  an input file with 6 columns: Name, SSN, Phone#, City , salary,Designation Delimited by ‘,’ (comma)  
    1.1 Designation should be (Associate and Sr. Associate).
2. Pass a parameter which will be a column number.
3. Sort the file based on passed parameter.
4. Create an output file (Report.txt) with 1st, 2nd, ,3rd, 4th and last column with the sorted data.
5. Create 2 ouput files "Report1" & "Report2,  based on below conditions  
   5.1 If the record in "Report.txt" is having designation as "Associate", then put that record in Report1.txt   
   5.2 If the record in "Report.txt" is having designation as "Sr.Associate", then put that record in Report2.txt
6. Using SUBSTRING.

Input file



Shell Script



Output

