**String Commands**

String related commands will help you to print/search file text as required

***HEAD*:** Head prints the first N number of data of the given input. By default, it prints first 10 lines of each given file.

Example: head file2

head –n 2 file3 #number of lines

|  |
| --- |
|  |
| head -n Command Output |

***SORT:*** Sort is a simple and very useful command which will rearrange the lines in a text file so that they are sorted, numerically and alphabetically. By default, the rules for sorting are:

ü Lines starting with a number will appear before lines starting with a letter.

ü Lines starting with a letter that appears earlier in the alphabet will appear before lines starting with a letter that appears later in the alphabet.

ü Lines starting with a lowercase letter will appear before lines starting with the same letter in uppercase.

Example: sort –r file2

|  |
| --- |
|  |
| sort -r Command Output |

***Options***

-b ignores leading blanks

-d considers only blanks and alphanumeric characters

-f fold lower case to upper case characters

-g compare according to general numerical value

-i consider only printable characters

-M compare (unknown) < `JAN' < ... < `DEC'

-n compare according to string numerical value

-r reverse the result of comparisons

-c check whether input is sorted; does not sort

-k start a key at POS1, end it at POS2 (origin 1)

-m merges already sorted files; do not sort

-o write result to FILE instead of standard output

-s stabilize sort by disabling last-resort comparison

-S use SIZE for main memory buffer

-t use SEP instead of non-blank to blank transition

-T use DIR for temporaries, not $TMPDIR or /tmp

-z end lines with 0 byte, not newline

***UNIQ*:** Uniq command is helpful to remove or detect duplicate entries in a file.

Example: uniq <file name> - it will print uniq values

|  |
| --- |
|  |
| uniq Command Output |

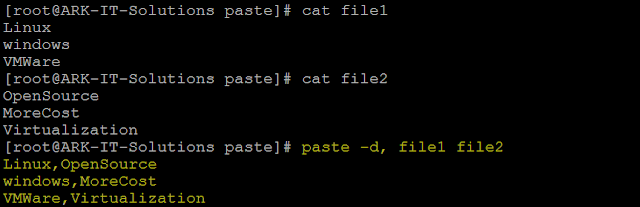
***PASTE:*** It is very useful for merging a single file and also for merging set of files as well.

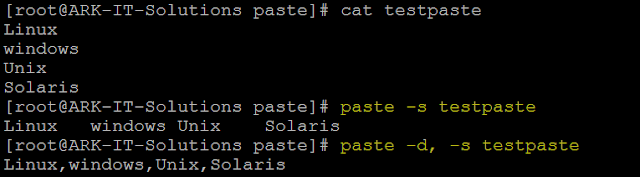
ü paste command examples for single file handling

ü paste command examples for multiple files handling

Example: paste –s file1 #All the separate lines are printed as one line

paste –d, -s file1 #Combined the 'file1' and 'file2' with comma (,) separated.





***CUT:***Cut is used for text processing. You can use this command to extract portion of text from a file by selecting columns.

**Example:** Below screenshot is the best example, first i have used cat command to see the content of file 'testcut'. In next highlighted cut command i have cut the 4 characters from the text.

|  |
| --- |
|  |
| cut Command Output |

***TR*:** It will translate content of the file from one case to another case vice versa. Upper case to Lower case.

Note: if you want to change the file text to caps then redirect the output to another file.

Example: in below example i have used 'file1' content to convert to caps lock characters.

|  |
| --- |
|  |
| tr Command Output |

***SED*:** Sed is a Stream Editor used for modifying the files in unix (or linux). Whenever you want to make changes to the file automatically, sed comes in handy to do this. Most people never learn its power; they just simply use sed to replace text. You can do many things apart from replacing text with sed.

Example: In below screenshot i have replaced the 'linux' string to 'unix' .

|  |
| --- |
|  |
| sed Command Output |

***DIFF*:** To compare the difference between two files text you can use this command

|  |
| --- |
|  |
| diff Command Output |

<https://www.youtube.com/watch?v=RynYYIrFlJ8>