

# Week Report

---

## cat

Description: command is used for displaying the content of a file

syntax: cat + option + files to display

### Example

cat todo.lst – display the content of a file located in the pwd cat ~/Documents/todo.lst - display the content of a file using absolute path

## tac

Description: command is used for displaying the content of a file in a reverser order

syntax: tac + option + files to display

### Example

tac cars.lst – display the content of a file located in the pwd tac ~/Downloads/cars.lst - display the content of a file using absolute path

## Head

Description: command display the top N number of lined of a given file.

syntax: Head + option + files

### Example

head doc.txt. - Display the first 10 lines of a file head -n 20 doc.txt - Display a specific number of lines from a file

## tail

Description: command display the last N number of lines of a given file

syntax: tail + option + file

### Example

tail file.txt - Display the last 10 lines of a file tail -n 20 file.txt - Show the last 20 lines of a file

## cut

Description: command is used to extract a specific section of each lone of a file and display it to the screen

syntax: cut + option + files

### example

`cut -f1 doc.txt` - Extract the first field from a tab-separated file  
`cut -c3-6 doc.txt` - Extract characters 3-6 from each line of a file

## Paste command

Description: The paste command is used for joining files horizontally in columns

syntax: `paste + option + files`

Example: `paste car.txt cat.txt` - merge two files  
`paste -d ":" car.txt cat.txt` - merge two files using a different delimiter

## Sort command

The sort command is used for sorting files.

syntax: `sort + option + files`

Example: `sort doc.txt` - sort a file  
`sort -r doc.txt` - sort a file in reverse order

## wc command

Description: The wc command is used for printing the number lines, characters and bytes in a file

syntax: `wc + option + files`

Example: `wc -m doc.txt` - display the number of characters in a file  
`wc -l doc.txt` - display the number of lines in a file

## tr command

Description: tr command is used for translating or deleting characters from standard output

syntax : `standard output | tr + option set + set`

example `cat doc.txt | tr -s "[:space:]" ' '` – translate tab into space

`cat pic.txt | tr "[:space:]" '\t'` translate white space into tab

## diff command

Description: the diff command compares files and displays the difference between them

syntax: `diff + option + file1 + file2`

Example:

`diff doc.txt doc-backup.csv` - display the difference between two files  
`diff -y doc.txt doc-backup.csv`

## grep

Description: grep is used to [search text in given file syntax `grep + option + search criteria + files`

example `grep 'word' file.txt` - Search for a word in a file

`grep -r 'pattern' directory/` → Search recursively in a directory for a specific pattern

## awk

Description: awk is a scripting language used for processing and displaying text

syntax: `awk + option + {awk command} + file + file to save`

example `awk -F: '{print $1}'` – print first field of files

`$ awk '{print}' employee.txt` - Awk prints every line of data from the specified file

`$ awk '/use/ {print}' employee.txt` - Print the lines which match the given pattern.

`awk -F: '{print toupper($1)}' /documents/cars` - convert the first field to upper/lower case

`awk '{print length ($0)}' /documents/doc` - Print the length of a line (record)

## sed command

Description: The sed command is a stream editor that perform operations on files and standard output

syntax: `sed option + sed script + files`

example `sed 's/blue/red/' color-list.lst` - replacing a string in given file

`sed '1,3 s/pizza/rice/' shopping list` - replacing string on a range of lines

`sed '5d' cars-list.txt` - to delete a particular line

`sed '$d' cars-list.txt` - to delete the last line

`sed '2,8d' cars-list.txt` - to delete line from range x to y