

NOTES 7

cat

Description = cat command is used for displaying the content of a file. cat is shot for `concatenate` which is the cammads intended use. usage: `cat + option + file(s) to display` Example =

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

tac

Description= used for displaying the content of a file in reverse order. usage= `tac + option + files(s) to display` example =

- display the content of a file using absolute path `tac todo.md`

head

Description= displays the top N number of lines of a given file. by default it prints the first 10 lines. if more than one file name is provided then data from each file is preceded by its file name. usage= `head + option + file(s)` example = display the first 10 lines of a file `head ~/Documents/Book/dracula.txt`

tail

Description= displays the last N number of lines of given file. By default, it prints the last 10 line. if more than one file name is provided then data from each file is preceded by its file name. usage= `tail + option + file` example = display the last 10 lines of a file `tail ~/Documents/Book/dracula.txt`

cut

Description= used to extract a specific section of each line of a file and display it to the screen usage= `cut + option + files(s)` example = display a list of all the users in your system `cut -d ':' -f1 /etc/passwd`

sort

Description= used for sorting files. the sort command supports sorting: alphabetically, in reverse order, by number, and by month.

usage= `sort + option + file` example = `sort users.lst`

More examples of sort

- Sort a file and save the output to a new file
 - `sort -o sorted.lst users.lst`
- Sort a file in reverse order
 - `sort -r users.txt`
- Sort by column number
 - `sort -k 2 users.txt`
- Sort a file with numeric data
 - `sort -n phones.txt`
- Check if a file is sorted
 - `sort -c sorted.lst`
- Sort and remove duplicate entries
 - `sort -u users.lst`

Note:
Use the `-t` option
to specify a
delimiter.
For example

```
sort -t";" -k3  
cereal.csv
```

WC

description = used for printing the number of lines, characters and bytes in a file. usage = `wc + option + files(s)` example =

The wc command

- **Description:**
 - The `wc` command is used for printing the number of lines, characters and bytes in a file
- **Usage:**
 - `wc + option + file(s)`
- **Basic Example:**
 - Display the number of characters in a file
 - `wc -m users.txt`
 - Display the number of lines in a file
 - `wc -l users.txt`
 - Display the number words in a file
 - `wc -w users.txt`

description = used for translating or deleting characters from standard output. usage = `standard output | tr + option + set + set` example =

The tr command

- **Description:**
 - The tr command is used for translating or deleting characters from standard output.
- **Usage:**
 - `Standard output | tr + option + set + set`
- **Basic Example:**
 - Translate one character to another (For example a period with a comma.
 - `cat file.txt | tr '.' ','`
 - Translate white space into tabs.
 - `cat program.py | tr "[:space:]" '\t'`
 - Translate tabs into space.
 - `cat file.py | tr -s "[:space:]" ' '`

diff

description = compares files and displays the differences between them. usage = `diff + option + files1 + file2` example =

The diff command

- **Description:**
 - The diff command compares files and displays the differences between them
- **Usage:**
 - `diff + option + file1 + file2`
- **Basic Example:**
 - Display the difference between two files
 - `diff cars.csv cars-backup.csv`
 - Display the difference between two files in a column format:
 - `diff -y cars.csv cars-backup.csv`

grep

description = used to search text in given file. grep works line by line basis. usage and example =

The grep command

- **Description:**
 - Grep is used to search text in given file. Grep works line by line basis (it matches the search criteria in a line by line basis).
- **Usage:**
 - `grep + option + search criteria + file(s)`
- **Basic Example:**
 - Search any line that contains the word "dracula" in the given file:
 - `grep 'dracula' ~/Documents/dracula.txt`