Notes 7

cat:

Definition: The cat command is used for displaying the content of a file. **Cat** is short for concatenate.

Usage:

- `cat`+ `option` + `file(s) to display`

Examples:

- `cat` **todo.lst**

- `cat` **~/Documents/todo.lst**

- `cat` -n **~/Documents/todo.lst**

tac:

Definition: The tac command is used for displaying the content of a file in reverse order.

Usage:

- `tac` + `option` + `file(s) to display`

Examples:

- `tac` **todo.md**

- `tac` **~/Documents/todo.md**

- `tac` **cars.cvs**

head:

Definition: The head command 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)`

Examples:

- `head` **~/Documents/Books/dracula.txt**

- `head` -5 **~/Documents/Books/dracula.txt**

- `head` -n 1 *.**csv** *.**py**

tail:

Definition: The tail command displays the last N number of lines of a given file. By default, it prints the last 10 lines. If more than one file name is provide then data from each file is preceded by its file name.

Usage:
`tail` + `option`+`file`

Examples:

- `tail` **~/Documents/Books/dracula.txt**

- `tail` -5 **~/Documents/Books/dracula.txt**

- `tail` -v -n 7 **~/Documents/Books/dracula.txt**

cut:

Definition: The cut command is used to extract a specific section of each line of a file and display it to the screen.

Usage:
`cut`+`option`+`file(s)`

Examples:

- `cut` -d ':' -f1 **/etc/passwd**

- `cut` -d ':' -f1,7 **/etc/passwd**

- `cut` -d ',' -f1 **cars.csv**

sort:

Definition: The sort command is used for sorting files. The sort command supports sorting alphabetically, in reverse order, by number, and by month. Unless specified otherwise:

- Lines staring with a number appear before lines starting with a letter
- lines starting with a letter that appears earlier in the alphabet will appear before lines starting with a letter that appears earlier in the alphabet
- lines starting with a lowercase letter will appear before lines starting with the same letter in uppercase

Usage:
`sort`+`option`+`file`

Examples:

- `sort` **users.lst**

- `sort -o` **sorted.lst users.lst**

- `sort -t ';' -k3` **cereal.csv**

wc:

Definition: The wc command is used for printing the number of lines, character and bytes in a file

Usage:
`wc +option+file(s)`

Examples:

- `wc -m` **users.txt**

- `wc -l`**users.txt**

- `wc -w`**users.txt**

tr:

Definition: The tr command is used for translating or deleting character form standard output

Usage:
`Standard output | tr+option+set+set`

Examples:

- **cat file.txt** `| tr '.' ','`

- **cat program.py** `| tr '[:space:]' '\t'`

- **cat files.py** `| tr -s '[:space:]' ''`

diff:

Definition: The diff command compares files and displays the differences between them

Usage:
`diff+option+file1+file2`

Examples:

- `diff` **cars.csv cars-backup.cvs**

- `diff -y` **cars.csv cars-backup.cvs**

- `diff -u` **cars.csv cars-backup.cvs**

grep:

Definition: 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)`

/

Examples:

- `grep` **'dracula' ~/Documents/dracula.txt**

- `grep -i` **'dracula' ~/Documents/dracula.txt**

- `grep -in` **'dracula' ~/Documents/dracula.txt**

Examples:

- `grep` **'dracula' ~/Documents/dracula.txt**

- `grep -i` **'dracula' ~/Documents/dracula.txt**

- `grep -in` **'dracula' ~/Documents/dracula.txt**