cut

Description

The cut command will grab a specific section and show it on screen.

Syntax

`cut + option + file`

Example

- Display all users in your system:
  - `cut -d ':' -f1`
- Cut a range of bytes:
  - `cut -b 1-5 random/what.txt`
- Find the permissions from the output:
  - `ls -l | cut -d ' ' --complement -s -f1`

cp

Description

This command will copy files and it's directory.

Syntax

`cp + options + directory`

Example

- copy what to random directory:
  - `cp -t random/ what`
- Copy files.tar and hello to random:
  - `cp -t random/ files.tar hello.txt.gz`
- Copy random directory to exp:
  - `cp -r random/ exp/`

mkdir

Description

mkdir is a command to make directories

Syntax

`mkdir + options + directory`

Example

- Make a sonic directory:
  - `mkdir sonic`
- create a tissue directory with another directory name hi in it at the same time:
  - `mkdir -p tissue/hi`
- create more directories at the same time:
  - `mkdir -p paper/cars m/city /forest`

Brace Expansion

Description

The brace expansion {} is a feature that allows you to generate arbitrary strings.

Syntax

```
mkdir + option + name of directory
```

Example

- create a directory in pictures named movies with three folders inside that directory named marvel, dc and disney in a single command line:
  - `mkdir -p movies/{marvel,dc,disney}`
- Now within the pictures create one file in each folder in the movies directory in a single command line:
  - `touch movies/{marvel,dc,disney}/file 1 file 2 file 3`
- Description of example:
  - `mkdir -p food/{favorite/{fav1,fav2}/{cheeseburger,fries},favorite2/{fav1,fav2,fav3}/{yams,steak,pizza}}/`

wc

Description

wc command is used for printing the number of lines, characters and bytes in a file.

Syntax

```
wc + options + file
```

Example

- Print the amount of characters in dracula and it's bytes:
  - `wc -mc dracula.txt`
- Print the maximum of display width:
  - `wc -L dracula.txt`
- Print dracula word count:
  - `wc -w dracula.txt`

grep

Description

Grep is used to search a text in a file by each line to match a world line by line.

Syntax

`grep + options + search criteria + files`

Example

- In pride-and-prejudice search for the only text I and you but by it's self.
  - `grep -Ew "I|you" pride-and-prejudice.txt`
- Display How many how many times I and you is in dracula:
  - `grep -Ew "I|you" dracula.txt | wc -w dracula.txt`
- Search and only match the words army and artillery in war-and-peace.txt:
  - `grep -Eo "army|artillery" war-and-peace.txt`

ls

Description

The ls command is used to list whatever is in the given directory.

Syntax

`ls + options + directory`

Example

- List the files in current directory:
  - `ls`
- list all files and hidden in the directory that you are in:
  - `ls -a`
- List files inside a given directory:
  - `ls -a ~/Pictures`

date

Description

The date command will list the current date.

Syntax

`date + options + directory`

Example

- Use the date command in your home directory:
    - `date`
- Print the date in RFC 5322 format:
    - `date -R`
- Display only the year month and day:
    - `date -I`

uname

Description

This command will display information about your system.

Syntax

`uname + options`

Example

- Display operating system:
    - `uname -o`
- Show kernel release and print your machine name in a single line command:
    - `uname -rm`
- print hardware platform and processor type:
    - `uname -pi`

du

Description

This command is used to gain disk usage information.

Syntax

`cmd + options + directory`

Example

- Display pictures directory usage :
    - `du Pictures/`
- Display pictures disk usage in human readable:
    - `du -h Pictures/`
- Display bytes in it's apparent form:
    - `du -b Pictures/`

echo

Description

echo is used to display a line of text and ot can Echo the string to standard output.

Syntax

`echo + options + string`

Example

- Display hello there:
  - `echo "hello there"`
- if you use just echo it will leave an blank:
  - `echo`
- now echo without the quotation marks:
  - `echo hello world`

tree

Description

the tree command will list the what ever is that directory in a tree format.

Syntax

`tree + options + directory`

Example

- Display the downloads folder in a tree format:
  - `tree Downloads/`
- Display downloads in a human readable:
  - `tree -h`
- Print the full path file permissions in human readable size:
  - `tree -fph Downloads/`

mv

Description

Moves files

Syntax

`mv + options + source + directory`

Example

- create a file named pringles once created move that file to car:
  - `mv -t car/ pringles`
- Create a directory named movies and new movies, then move new movies to movies:
  - `mv -t movies/NewMovies/`

- Move files sonic and moonfall to NewMovies:
    - `mv -t movies/NewMovies/ sonic moonfall`

man

Description

The man command is used to display information about a command and it's options you can use beside it

Syntax

`man + options + command`

Example

- Display information on sudo:
    - `man sudo`
- Display the 10th section of passwd:
    - `man 10 passwd`
- Display a section of passwd:
    - `man -f passwd`

apt

Description

use this command to update software, system and download software.

Syntax

`apt + options`

Example

- Download flameshot:
    - `sudo apt install flameshot`
- Update ubuntu:
    - `sudo apt-get update`
- If you don't want to enter your password use minus y:
    - `sudo apt install flameshot -y`

Wildcards (*,?,[])

Description

Wildcard represents letters and characters used to specify a file name for searches.

Syntax

`cmd + options`

Example

- Display all the markdown files:
  - `ls *.md`
- match any character that begins with these sets of characters:
  - `ls [a-fp-z]*`
- list hidden files:
  - `ls .??*`

pwd

Description

This command will show your present working directory.

Syntax

```
pwd + options
```

Example

- Display where you are:
  - `pwd`
- Go to the pictures directory and display it's path:
  - `cd Pictures/ pwd`
- go to Pictures then movies and display it's path:
  - `pwd`

rm

Description

The rm command will remove files and directories

Syntax

```
rm + options + directory or file
```

Example

- Make a directory with names test1 with a file name named test:
  - `mkdir -p test1/`
- Make a file in test1 directory:
  - `touch test1/test`
- remove test file:
  - `rm -r test1/test`

stat

Description

Display file or file system status.

Syntax

`stat + options + FILE`

Example

- display dracula in terse form:
  - `stat -t dracula.txt`
- Display your file system status:
  - `stat -f ~/`
- display file information on dracula:
  - `stat -f dracula.txt`

tar

Description

creates archives by combining files and directories into a singles file.

Syntax

`tar + options + archive name + files`

Example

- make file named food then create a tar file and name it files.tar then add the food file or whatever file you want to tar:
  - `tar -cf files.tar food`
- Extract the files.tar:
  - `tar -xf files.tar`
- Make a file named file1 add that file to files.tar archive:
  - `tar -rf files.tar file1`

vim and nano

Description

these commands will open basic things such as opening a file and editing it then closing it.

Syntax

`vim + options + file name`

Example

- Create a new file and open it in a single command line:
  - `vim.txt`

- Whenever you finish typing what ever you want use the two periods then type the command to save and quit:
  - `:wq`
- Make a file named cart then use nano to edit the file:
  - `nano cart`

gz, bzip2, or xz

Description

These commands are used for file compression.

Syntax

`gz,bzip2 or xz + options + file`

Example

- Make a file called hello.txt then compress it:
  - `gzip hello.txt`
- decompress a file:
  - `bzip2 -d hello.txt`
- create file hellothere.txt hi.txt sonicfast.txt what.txt yo.txt then decompress those files not one my one find the command to decompress multiple files:
  - `xz hellothere.txt hi.txt sonicfast.txt what.txt yo.txt`

cat

Description

The cat command is used to gain information from a file.

Syntax

`cmd + options + file to display`

Example

- Display dracula.txt:
  - `cat dracula.txt`
- Display the file suppressing repeating empty lines to single lines with $ at the end of the line:
  - `cat -sE ~/Documents/Books/dracula.txt`
- Display one of you labs:
  - `cat Lab1/lab1-submission.md`

tac

Description

This command is used for to display a file in reverse order

Syntax

`tac + options + files to display`

Example

- Display one of your markdown files in reverse order:
    - `tac Lab1/lab1-submission.md`
- Display markdown lab file in a interpret regular expression :
    - `tac -r lab1-submission.md`
- Display the content of a file using the absolute path:
    - `tac ~/Documents/Books/dracula.txt`

head

Description

This command will display the first numbers of lines of a given file if you were to put the command regular it print the first 10 lines.

Syntax

`cmd + options + files`

Example

- Display only 3 lines of dracula:
    - `head -3 dracula.txt`
- Display the first 10 lines of dracula.txt:
    - `head -10 dracula.txt`
- Display the first 15 lines of dracula.txt:
    - `head -15 dracula.txt`

tail

Description

just like head but instead of printing the first lines tail will print the last lines

Syntax

`tail + options + file`

Example

- Print the last five lines in pride-and-prejudice.txt:
    - `tail -5 pride-and-prejudice.txt`
- Print the last ten lines in pride-and-prejudice.txt:

    - `tail -10 pride-and-prejudice.txt`
- Print the last twenty lines in pride-and-prejudice.txt:
    - `tail -20 pride-and-prejudice.txt`

paste

Description

The paste command is used for joining files horizontally in columns.

Syntax

`paste + options + files`

Example

- merge tom1 tom2:
    - `paste tom1.lst tom2.lst`
- Show the serial of dracula.txt:
    - `paste -s dracula.txt`
- Merge two files:
    - `paste -d ":" tom1.lst tom2.lst`

sort

Description

The sort command will sort files and in reverse alphabetically order by number and by month.

Syntax

`sort + options + file`

Example

- sort a file in reverse order:
    - `sort -r skill.txt`
- sort a file in numeric data:
    - `sort -n skill.txt`
- sort and remove what ever duplicate entries:
    - `sort -u skill.txt`

tr

Description

This command is used for translating or deleting characters from standard output.

Syntax

`Standard output | tr + options + set + set`

Example

- deletes characters:
    - `cat skill.txt | tr -d ','`
- Put a comma after a repeated character:
    - `cat skill.txt | tr -s ','`
- Translate white spaces into tabs and squeeze repeats:
    - `cut skill.txt | tr -s "[:space:]" '\t'`

diff

Description

This command will compare file and show what are the difference between them.

Syntax

`diff + options + file1 + file2`

Example

- in file skill.txt make sure there is some words inside of the file then make food.txt and have a list of food then show the difference between the two:
    - `diff skill.txt food.txt`
- Display the difference in a column:
    - `diff -y skill.txt food.txt`
- expand tabs in output:
    - `diff -s skill.txt food.txt`

awk

Description

This command displays line by line.

Syntax

`awk + options + {awk command} + file + file to save (optional)`

Example

- Display first and 3rd field of etc passwd:
    - `awk -F: '{print $1,3}' /etc/passwd`
- Print the first and last field of the etc passwd:
    - `awk -F: '{print $1," = ",$NF}'`
- Print a file name but do not include the first two:
    - `awk 'NR > 3 { print }' /etc/passwd`

Saving the output of a command

Description

We can input and output of commands to and from files and combining multiple commands into a powerful pipelines

Syntax

```
Command output + > + file
```

Example

- save an output of a command:
  - `ls -lA > skill.txt`
- Saves the errors messages by a command to a file:
  - `ls lA downloads 2> error-of-ls`
- Does not display errors and sends the errors to deserted place:
  - `ls -lA downloads/ 2> /dev/null`

output redirection

Description

The pipe will allow you to redirect standard output of a command to another output.

Syntax

```
command_1 | command_2 | command_3 | .... | command_N
```

Example

- The command grep will help look for a specific sting in for a command option:
  - `man ls | grep "size"`
- Another example look for sort:
  - `man ls | grep "sort"`
- Search for the option to list by entries:
  - `man ls | grep "colum"`