

\$matthew\$ \$Proano\$ ##Cat Cat meaning to join two strings together ex. cat + file + file2

Display the content of a file with line numbers

```
cat -n /etc/passwd
```

Display the content of a file with line numbers excluding empty lines

```
cat -b /etc/resolv.conf
```

Display a \$ at the end of every line

```
cat -E /etc/group
```

*F
read the
command..
to read ti
passwd, re:*

Display the content of a file suppressing repeating empty lines to a single empty line

```
cat -s /etc/hosts
```

##tac To display the content of file in reverse tac + file + file2

more

basically display the content of a file one page at a time more + file to view more -10 /var/log/syslog

less

faster than more loads 1 page at a time less -N /var/log/syslog less -p "nobody" /etc/passwd ##head displays the first 10 lines of file head + option +file head -5 /etc/passwd ##diff ##tail Display the last ten lines tail + option + file ##Cut cut + option +file cut -d : -f1 /etc/passwd this is to display a list of user in a linux system ##paste uses to merge files together

Merge two files

```
paste users.txt ips.txt
```

Merge two files using a different delimiter

```
paste -d ":" users.txt ips.txt
```

Merge files sequentially instead of horizontally

```
paste -s users.txt ips.txt
```

##sort lets to sort files, goes line by line, sort -o usersSorted.txt users.txt (users.txt is the new file)

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 phoneNumbers.txt
```

Check if a file is sorted

```
sort -c usersSorted.txt
```

Sort and remove duplicate entries

```
sort -u users.txt
```

##wc is used to print number of lines,

characters and bytes in a file

Display the number of bytes in a file

```
wc -c users.txt
```

Display the number of lines in a file

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

Display the number of characters in a file

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

Display the number of words in a file

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

##tr

Translate one character to another. For example a period with a comma.

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

Translate white space into tabs. Useful with python programs.

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

Translate tabs into space. Again, useful with python ~~headaches~~ programs!

```
cat file.py | tr -s "[:space:]" ' '
```

Check out:
<https://robertalberto.com/linuxcommands>

##grep is used to find a string pattern from a file How to use Grep + option + pattern to match + file OR standard output + pipe + grep +pattern to match

Option	Explanation
-i	Turns case sensitivity off
-n	Displays line number of the each line matched
-E	Treats the pattern as an extended regular expression
-G	Treats the pattern as a basic regular expression
-v	Inverts the search
-o	Only display the string matched

Search for a given string in a file

```
grep "IP" data.csv
```

Search for a given string in a file with case insensitivity

```
grep -i "ip" data.csv
```

Search for a given string in multiple files

```
grep "user" file1 file2
```

Search for a string and show line numbers.

```
grep -n "License" /usr/share/doc/bash/README
```

Search and highlight the pattern.

```
grep --color "GNU" /usr/share/doc/bash/README
```

Display all the lines that do not match the pattern

```
grep -v "GNU" /usr/share/doc/bash/README
```

Display only the string match without the line.

```
grep -o "GNU" /usr/share/doc/bash/README
```

file descriptors

used to describe open files in a session allowed 9 descriptors at a time

Example:

- `ls -lax > list_of_files.txt`

To redirect standard error, we use: `2>`

Example:

- `cat badFile.txt 2> error_cat_command.txt`

To redirect standard output and append the output to a file, we use: `>>`

Example:

- `ls -lalh >> list_of_files.txt`

We can use the output redirection to create an empty file:

Example:

- `> newfile`
- `: > newfile` (in older versions of bash)

We can also get rid of output that we do not want:

Example:

- `ls -l ~/Downloads ~/documents 2> /dev/null`

cat command

use cat then type while using cat can redirect the output of cat to any file

```
1:jdoo@c1s106vm: ~
jdoo@c1s106vm:~$ cat > todolist.txt
clean room
pay bills
go to school
jdoo@c1s106vm:~$ cat todolist.txt
clean room
pay bills
go to school
jdoo@c1s106vm:~$
```

Pipe command |

The pipe allows to direct the output of a command into an input command `1 + | + command2 |`

Use grep to look for a string in a particular man page

```
man ls | grep "human-readable"
```

Display only the options of the of any command from its man page

```
man ls | grep "^[[:space:]]*[[[:punct:]]]"
```

Display all IP addresses from the output of the ip command

```
ip addr | grep -Eo '[[[:digit:]]{1,3}\. [[[:digit:]]{1,3}\. [[[:digit:]]{1,3}\. [[[:digit:]]{1,3}']
```

Alias

a shorthand for a complicated command