

# UNIX COMMANDS

*Sébastien Renault*

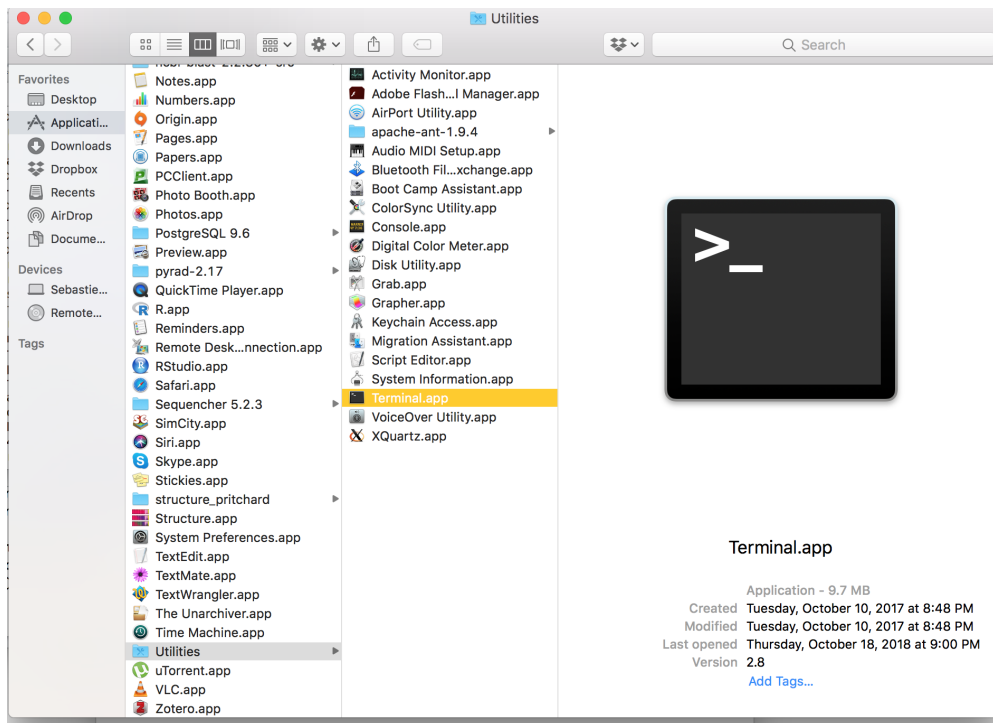
*October 24, 2018*

## Contents

<b>Get started</b>	<b>1</b>
<b>General principles</b>	<b>2</b>
<b>Exploration</b>	<b>2</b>
<b>Moving / removing / creating files</b>	<b>3</b>
<b>Examining files</b>	<b>3</b>
<b>Tricks</b>	<b>3</b>
<b>Working remotely</b>	<b>3</b>
<b>Other useful commands</b>	<b>4</b>
<b>Profile</b>	<b>4</b>
<b>Compression</b>	<b>4</b>
<b>Regular expressions</b>	<b>4</b>
<b>Examples</b>	<b>5</b>
Combining commands: . . . . .	5
Install and run command line programs . . . . .	5
bash . . . . .	5

## Get started

- Open terminal



- Download github repo (`curl -L https://github.com/seb951/basic_unix/archive/master.zip` >master.zip)
- Decompress it (`unzip master.zip`)
- It contains a few examples + .pdf version of tutorial

## General principles

- All commands are short, perform simple tasks & are highly optimized (fast)
- Nearly all commands and options (~99%) work across Unix-based OS
- Commands can take (many) options
- Commands can be combined (see more later)
- Many commands don't require any argument, some do
- `command -option1 -option2 -optionN... parameter object`
- Many commands output directly to terminal

## Exploration

Command	Meaning
<code>ls</code>	what's there
<code>ls -l</code>	with an option
<code>ls -t -h</code>	with more options
<code>ls -lthor</code>	combining even more options
<code>man ls</code>	how to run command (q = quit; d = next page; ; arrows & fn arrows = scrolling)
<code>pwd</code>	where are we (full path)
<code>cd</code>	move somewhere else (default to home)
<code>cd ..</code>	move back

Command	Meaning
cd	move back, then forward
../Applications	
cd ~	move home
cd /	move to root

## Moving / removing / creating files

Command	Meaning
mv	move
cp	copy
rm	<b>careful</b> remove a file
rmdir	<b>careful</b> remove an empty directory
rm -r	<b>VERY careful</b> remove directory recursively
mkdir	create empty directory
touch	create empty file
echo	print message to screen

## Examining files

Command	Meaning
head	print first 10 lines to screen
tail	print last 10 lines to screen
less	print file line by line (use arrows + fn arrows to move up/down)
cat	print file content to screen
sort	sort a file
uniq	unique lines in a file
clear	clear terminal
wc	wordcount
du	disk usage

## Tricks

Command	Meaning
>	redirect
>>	append to file
	pipe
*	anything (special character: for literal *, use \*)
?	any character (special character)
Ctrl+A	go to beginning of line in terminal
Ctrl+E	end of line
tab key	autocomplete

## Working remotely

Command	Meaning
<code>ssh</code>	secure shell (remote connection)
<code>scp</code>	secure copy (remote connection)

## Other useful commands

Command	Meaning
<code>chmod</code>	change mode of file
<code>sudo</code>	superuser do
<code>top</code>	what's running
<code>kill</code>	kill a process
<code>curl</code>	get stuff from web (MACOS native) or <code>wget</code> (MACOS non-native)
<code>history</code>	see recent history (use <code>!</code> with nb to use previous cmd)
<code>open</code>	open a file
<code>ln</code>	symbolic link
<code>which</code>	where is a command
<code>alias</code>	create own commands
<code>nano</code>	edit text (whole file loaded in memory)
<code>nano +123</code>	specify location in file

## Profile

- Let's look at `.bash_profile` to modify options (e.g. `nano ~/.bash_profile`)
- Usefull options are here: `examples/bash_example_commands`

## Compression

Command	Meaning
<code>gzip</code>	compression/decompression of files
<code>gzip -cd file.gz</code>	decompress to standard out (use a <code>&gt;</code> redirect)
<code>gunzip</code>	compression/decompression of files
<code>tar</code>	compression/decompression of archives
<code>tar -xvzf file tar.gz</code>	decompress a tar.gz (tarball)

## Regular expressions

Command	Meaning
<code>grep</code>	regular expression
<code>grep -options 'pattern' file</code>	regular expression

## Examples

### Combining commands:

- Let's look at the file `examples/sequence.fasta`
- print line 17 by piping `head` and `tail`
- Count lines starting with '`>`' (pipe `grep` and `wc`)
- Count unique lines starting with '`>`' (pipe `grep`, `sort`, `uniq` and `wc`)
- How many files is there in the directory (pipe `ls` & `wc`)

### Install and run command line programs

- Let's look at this example: <https://github.com/rec3141/rec-genome-tools/blob/master/bin/fastagrep.pl>
- Notice the `#!/` (shebang) on the first line
- It is a short `perl` script that *greps* sequences in a *.fasta* formatted file
- Copy-paste the text in a new file (e.g. `touch fastagrep.pl`, `nano fastagrep.pl`, paste and save)
- Make it executable (`chmod +x fastagrep.pl`)
- Let's look at it (`./fastagrep.pl`)
- Let's try it (`./fastagrep.pl 'deserticola' examples/sequence.fasta`)
- Let's symbolic link it so the computer knows where to find it (`ln source target`)

### bash

- The terminal runs (by default) the `bash` command language
- We can write or use more complex executable `bash` code
- Let's look at an example `less examples/bash_cmd2`