command line

The programs you call on a terminal are not so different from their graphical interface you are used to on windows/mac.

You need to know these commands:

`pwd more less cp mv mkdir ls cd chmod find`

Two useful tips:

- use **TAB** on your keyboard for command and name completion.
- the **up** arrow allows to browse your history (also available with `history`)

Exercise 1

go to your home folder

`cd`

create a fake file by using

`echo "hello world" > filetest` see if this file is present `ls -l`

read it

`more fileTest`

Of note, `less` is an alternative to `more`

rename it

`mv fileTest test`

check

`ls -l`

create a folder

`mkdir TEST`

Of note, all commands are case-sensitive `ll`

`ll` is a classic alias for `ls -l` move the file in this folder

`mv test TEST`

check, and see if present in the folder

`ll TEST`

copy it in the current folder

`cp TEST/test .`

`.` is the current folder, `..` is the folder one level close to the root `/` now we have the same file, with same name, one in the TEST folder, one in the current. use the up arrow, you should see 'cp TEST/test .' and change it for

`cp TEST/test test2`

the first and last field of `ls -l` should provide

```
drwxr-xr-x TEST
-rw-r--r-- test
-rw-r--r-- test2
```

trash test

`rm test`

if this command doesn't ask for confirmation, let me know we may change this behavior.

`chmod` allows to change permissions

try to read the file `test` after

`chmod 222 test`

`r` stands for read, `w` for write and `x` for execution for files and browsing for folders.
the first pattern is the owner
the second pattern is for the group
the third pattern is for everyone else

text editor

Let's have a look at a text editor, there is plenty of them, the one I use is `vim`, why? Because

- it's commonly installed on servers
- extremely powerful

enter the editor

`vim test2`

you have two modes

- command
- insert

By default you are in the command mode, let's enter in the editor mode with either `i` or `insert` on your keyboard. You should see `--INSERT--` at the bottom. Now you can edit you file. When its finished, press `ECHAP` to return in the command mode. You must enter `:` for each command. The useful ones

- `w` save changes to the file

- `:q!` quit without saving changes
- `:wq` write and quit

Exercise 2

`find` is great but not at all user-friendly. Try to find all your files which are bigger than 1 Go. then which are older than 1 year. Imagine doing this with windows...

Exercise 3, using a FASTA file

Get all sequences in genbank with the keyword trnl
http://www.ncbi.nlm.nih.gov/sites/entrez?db=nuccore&cmd=search&term=trnl

but download only sequences from one class like*mammals* as a FASTA file. You should obtain a 1.1 Mo file. Otherwise, you can use `/home/users/aginolhac/trnl_mammals.fasta`

- how to obtain the first 500 lines of a file?
  see the command `head` and its manual `man head`. You can redirect the output to a file with 'command > file_first500.fasta' for example.
- how can you obtain from line 400 to 560?
  Think of piping `head` and `tail`
- count how many lines in this file. See `wc`
- count how many sequences you have in the FASTA file. look at `grep`
- you should have obtained the 500 first lines in a file and the 500 last in a second file. How can you merge these two files? look at `cat`

Extra questions

- there is a empty file after each sequence. Try to remove them (my favorite is `sed`, but check the `-v` option of grep)
- Extract all headers (start with `>`) then the *gi number* (see `cut`). Redirect to a file.
- Look if there some double gi, look at `sort` and `uniq`.