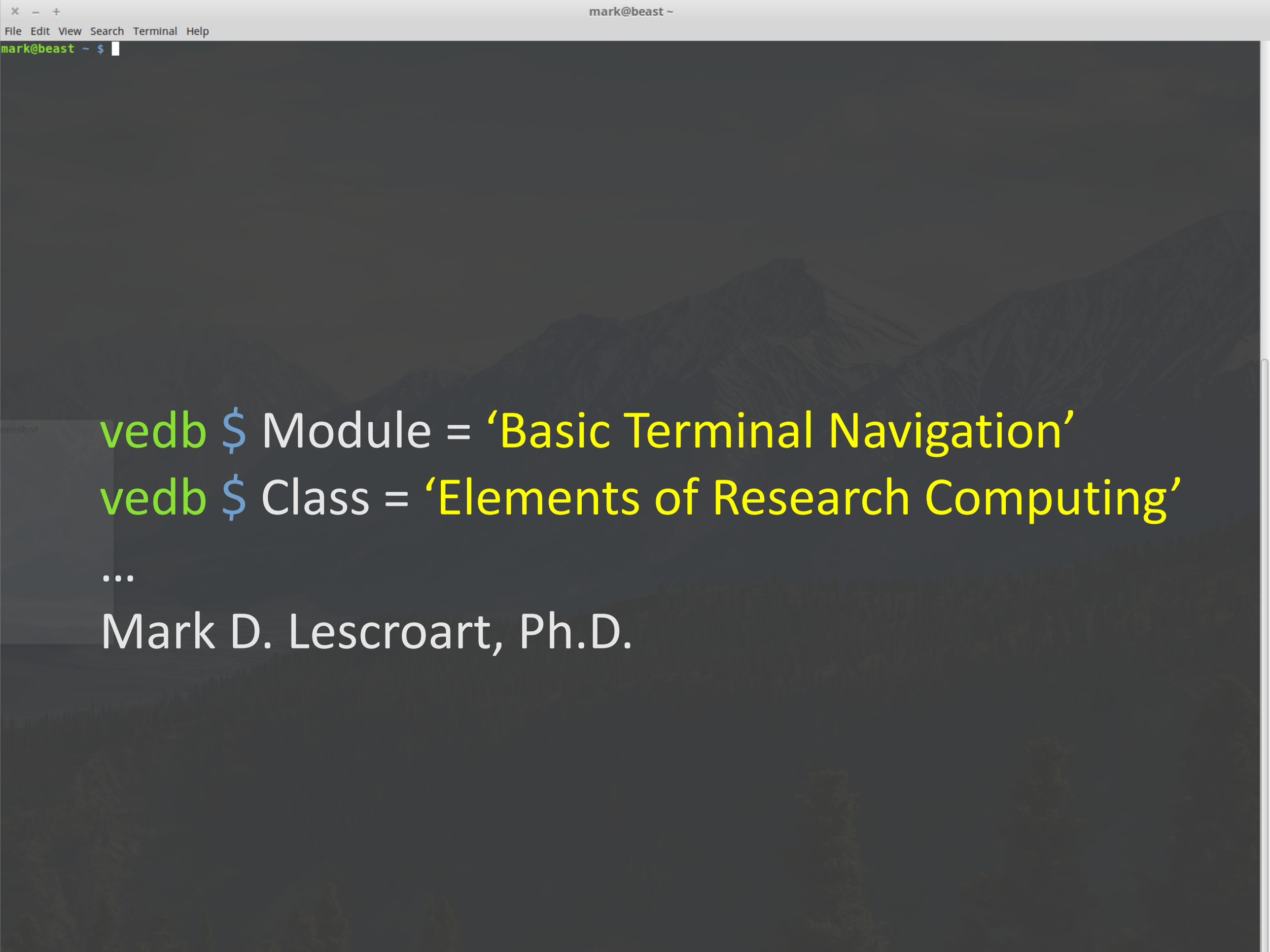


vedb $ Module = ‘Basic Terminal Navigation’

vedb $ Class = ‘Elements of Research Computing’

…

Mark D. Lescroart, Ph.D.

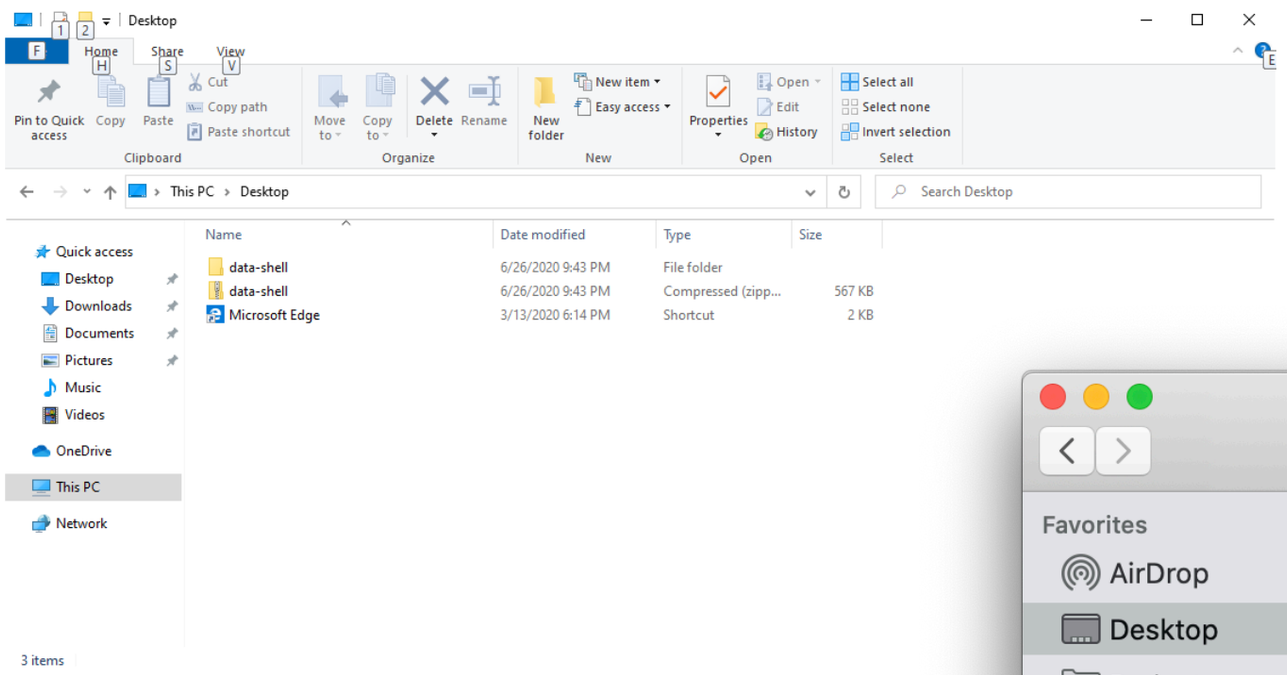
Goals of this lecture:

1. Introduce basic use of the command line (functions, options)

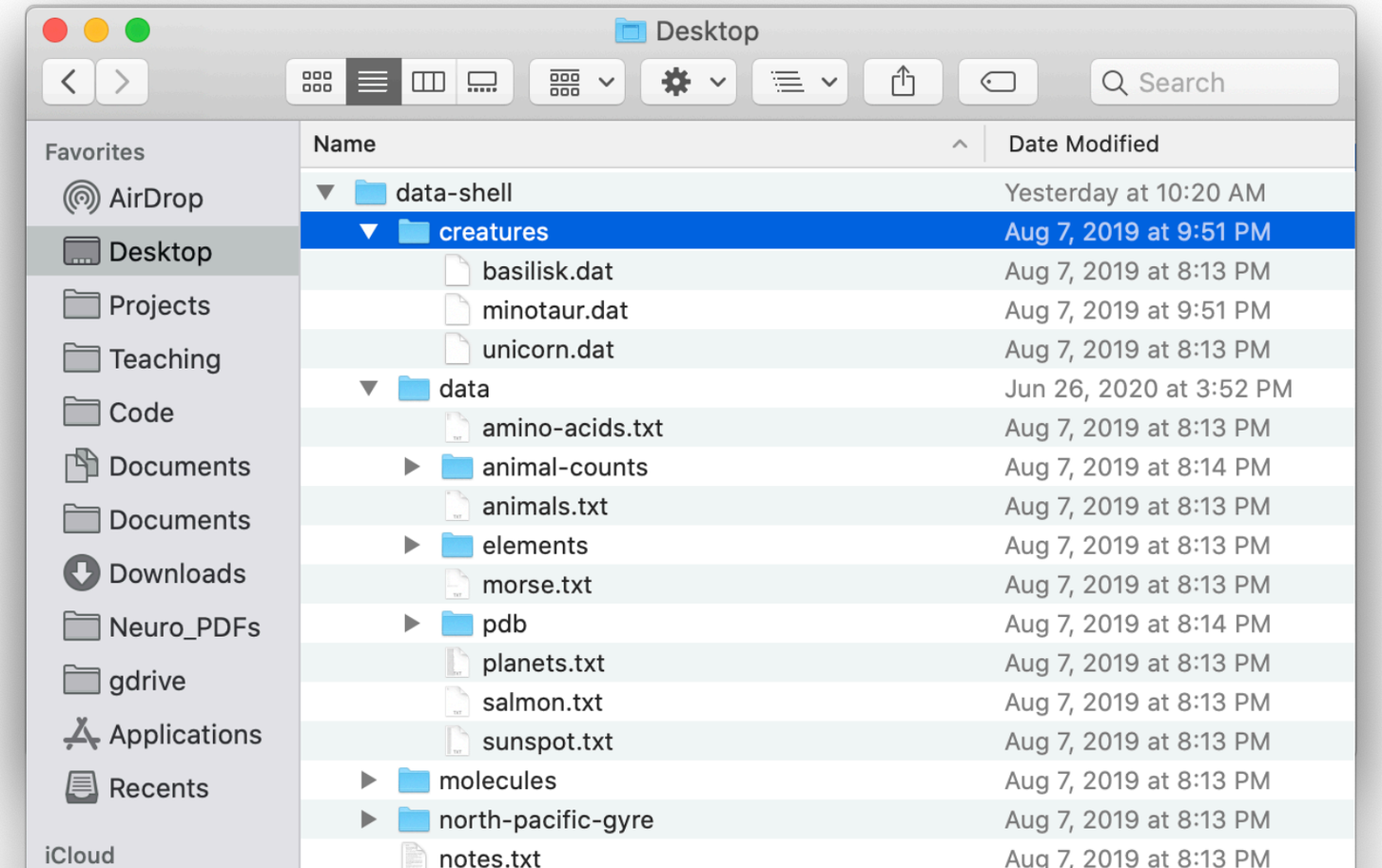
2. Learn functions specific to navigating around your computer

File paths

Windows Explorer



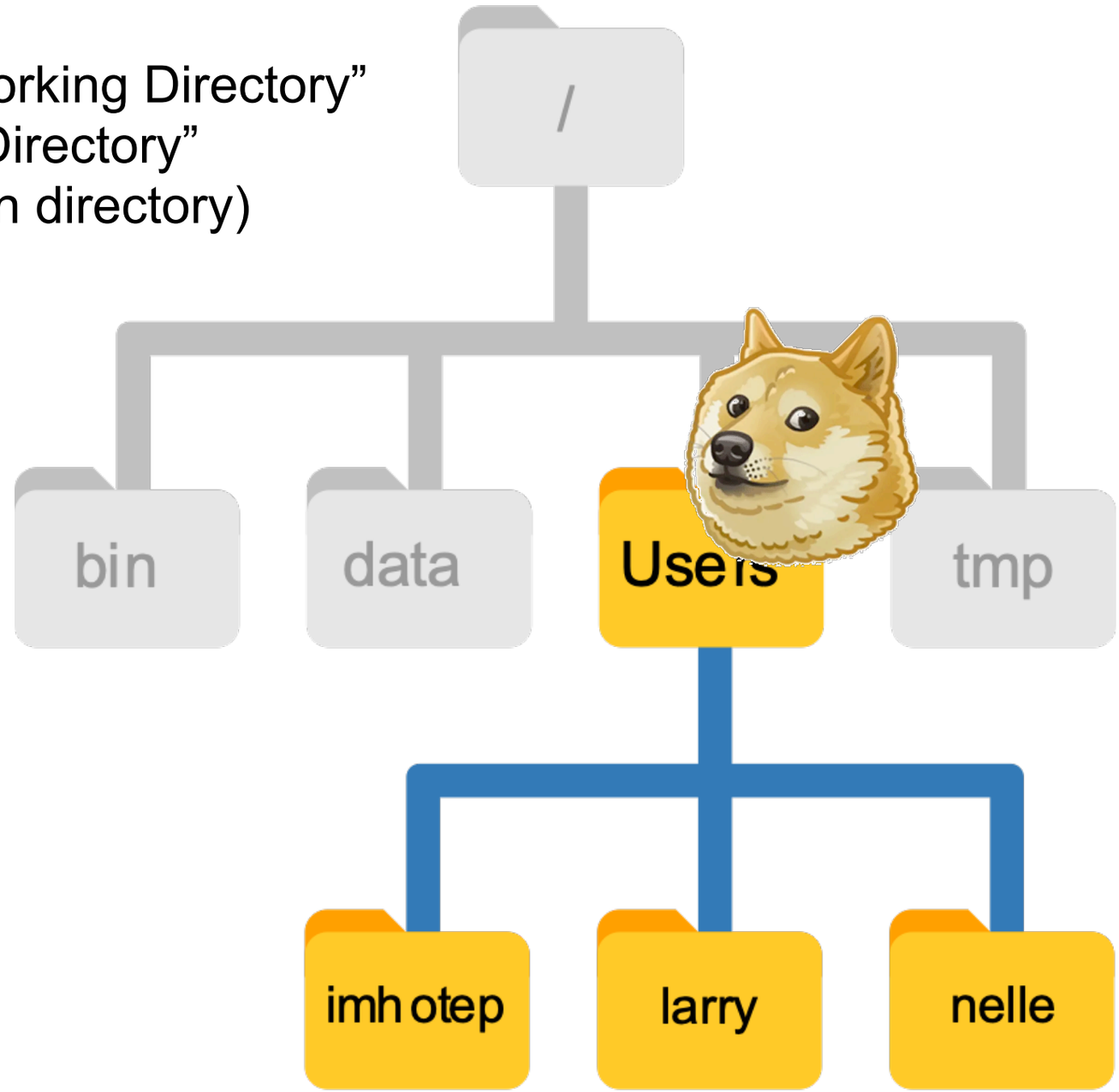
Finder



Basic Navigation commands

pwd - “Print Working Directory”
cd - “Change Directory”
ls - “list” (files in directory)

```
$ pwd
/Users/
$ cd nelle
$ pwd
/Users/nelle/
$ cd ..
$ pwd
/Users
```



[Demo of basic commands:
pwd, ls, cd]

General structure of Unix commands:

Command	Options	Arguments
	Note the dash!	Note the absence of a dash!
ls	-lt	data-shell/
cp	-r	data-shell data-shell2
cd		data-shell2

Executable program

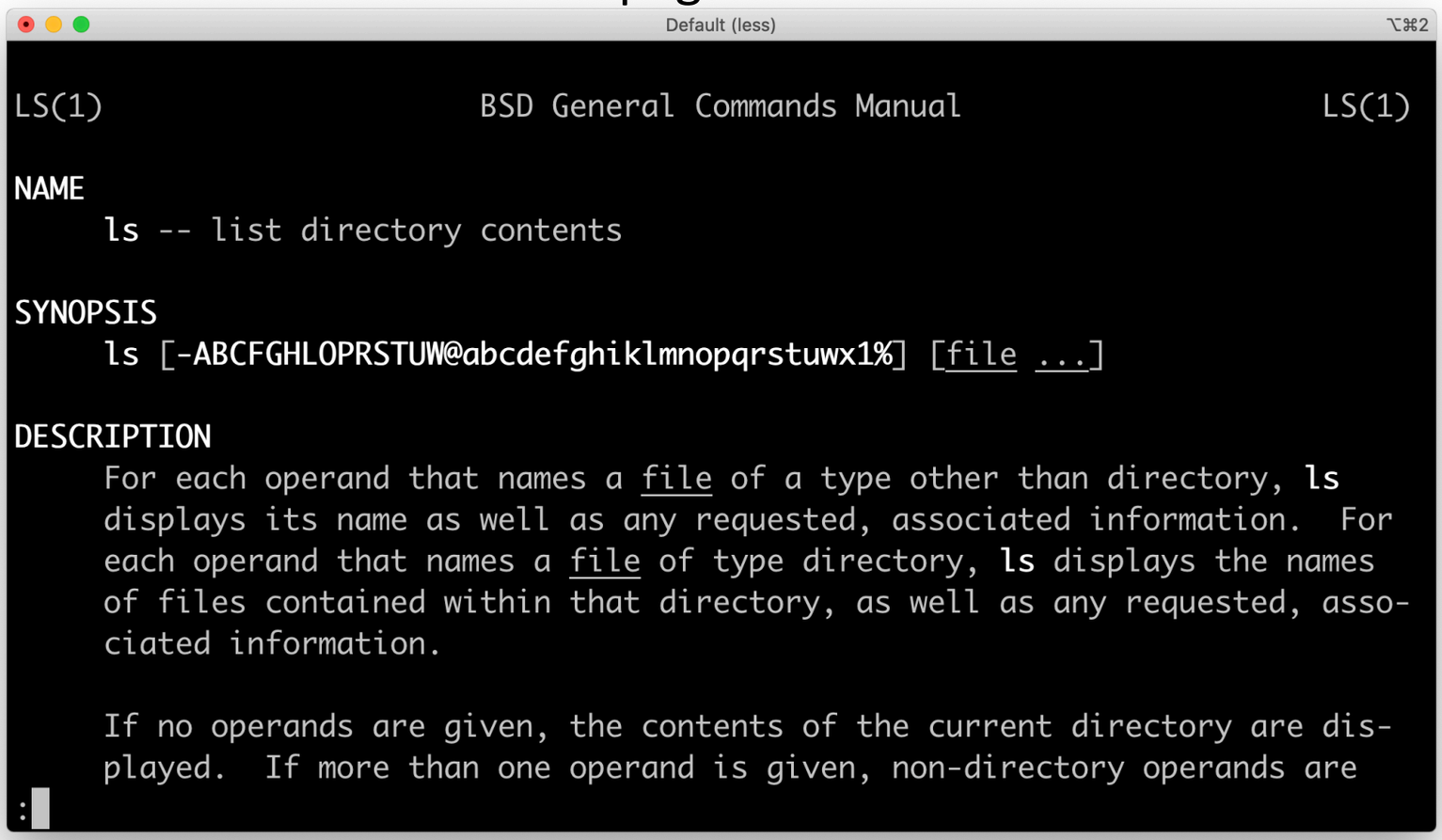
Ways to make it behave differently

Fundamental input or inputs

Manual (*man*) pages

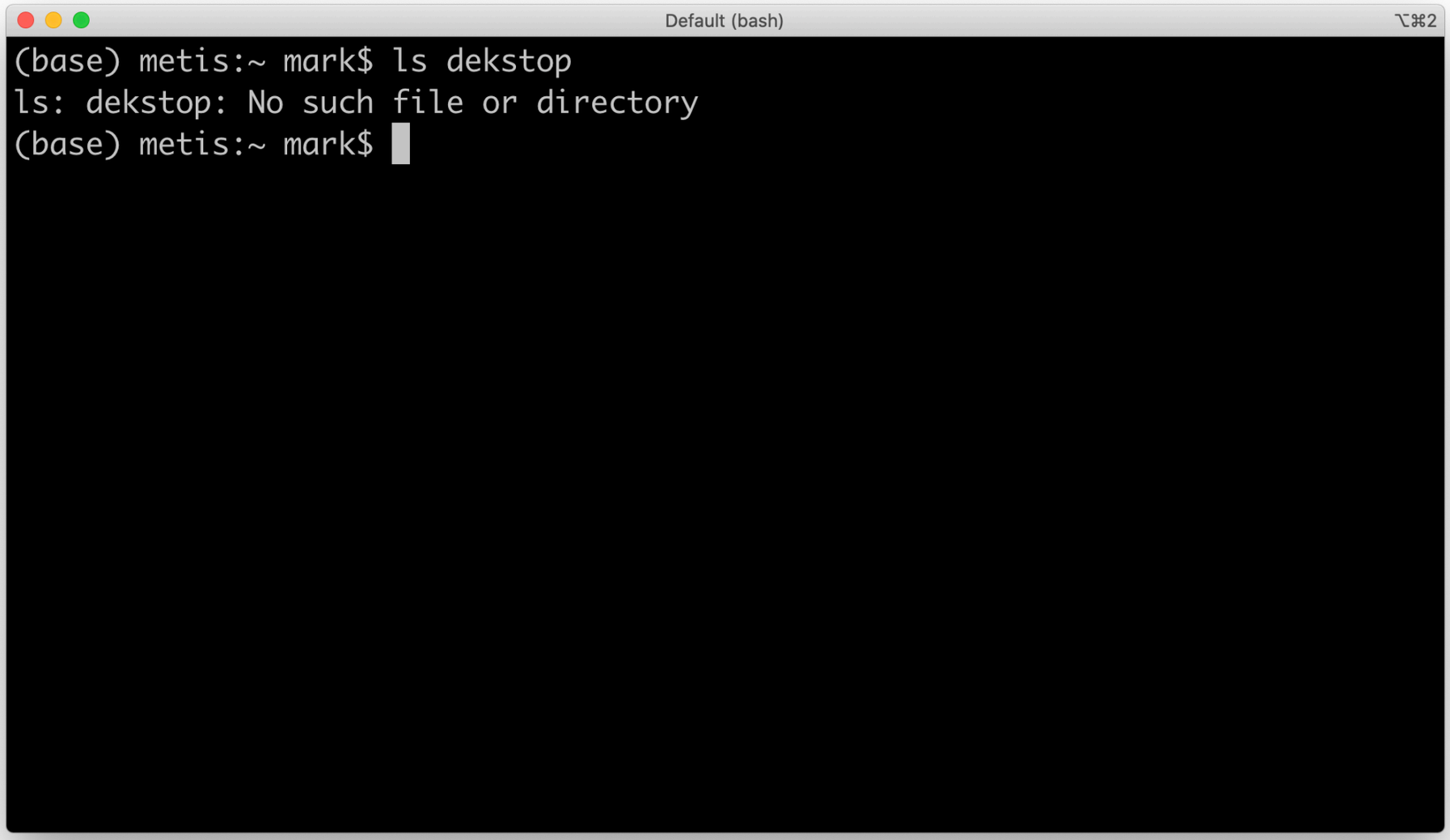
- All Unix commands have built in manual pages, which explain usage and all command line options
- *man* works slightly differently on Windows terminal programs (e.g. mobaxterm)
- Pages are read using a command called **less**; For full usage of less, call : **$ man less**
- Briefly:
  - pages can be advanced using the arrow keys, the space bar (for a page at a time) or in some cases the mouse scroll function
  - Quit out of the manual page by pressing `q`
  - Find things by using `/` and then typing the word or words you would like to search for
- Even intermediate and advanced users OFTEN use man pages

Man page for `ls` - call:
$ man ls



Reading error messages

- Errors are ALWAYS a part of programming. Learning to read error messages - and to google them! - is one of the most useful skills you can acquire as a programmer



Specifying files & directories in Unix ("Paths")

/this/is/an/absolute/path.pdf

e.g.

/usr/bin/vim

Starts with /

this/is/a/relative/path.txt

e.g.

shell-data/

Starts with

nothing special!**

~/home/directory/path.jpg

e.g.

~/shell-data/

Starts with ~/

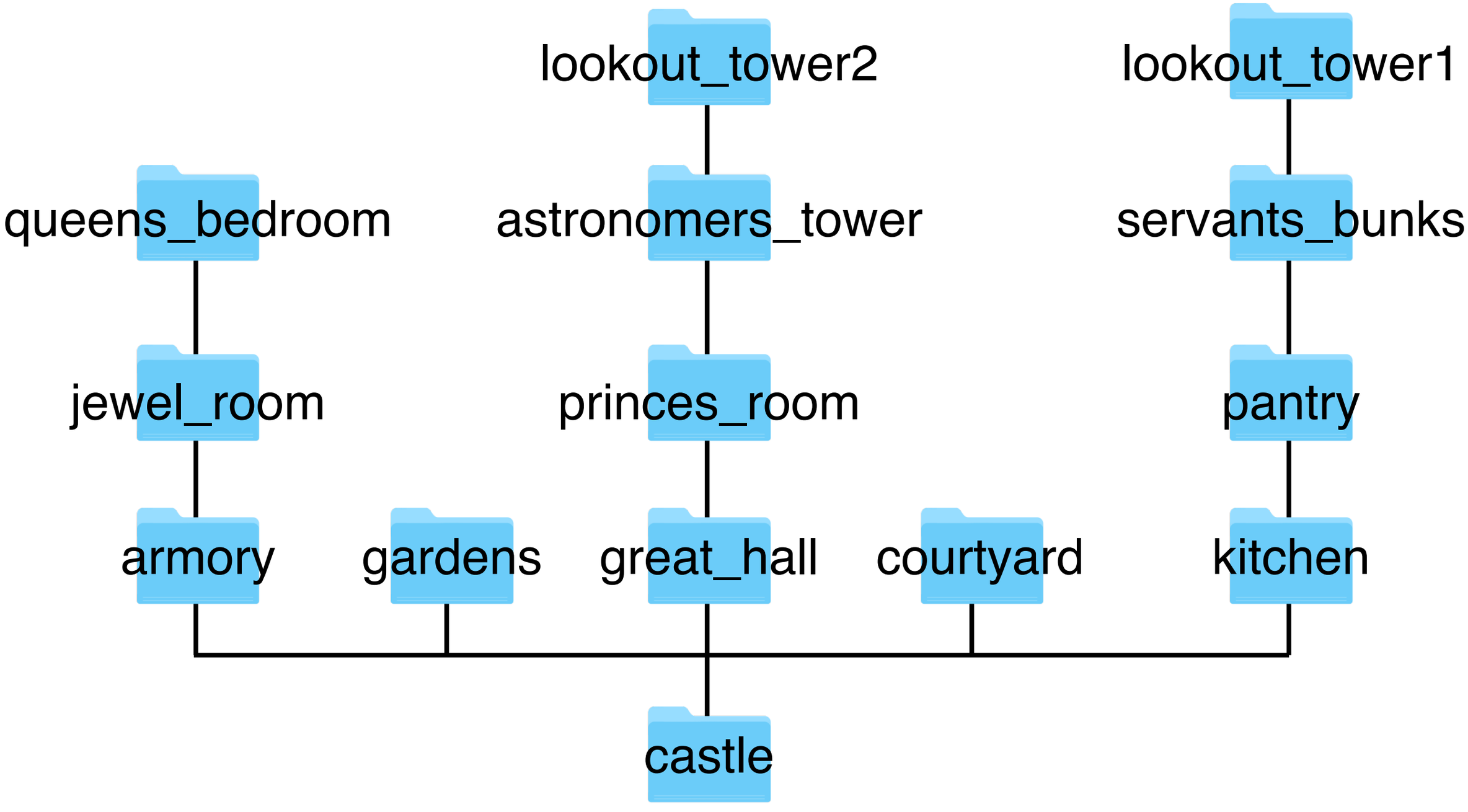
Hidden files and ./ and ../

Directories or files with names beginning with . are hidden by default, e.g. .cache/

BUT, confusingly enough, ./ refers to the CURRENT DIRECTORY, as in **ls ./**

.. Refers to a directory ABOVE the current directory, e.g. **cd ../**

Exercise: Castle assassin



Exercise:

http://swcarpentry.github.io/shell-novice/02-filedir/index.html

- Absolute vs Relative Paths
- Relative Path Resolution
- Reading Comprehension

There is a link to the Unix cheat sheet on the course git page under Day 1

Keep this on hand while you are learning and getting comfortable with these commands!

Unix Command Line Reference

Borrowed from; all items in *italics* are placeholders, and can be any file name or directory

File commands

ls – directory listing
ls -al – formatted listing with hidden files
cd *dir* - change directory to dir
cd – change to home
pwd – show current directory
mkdir *dir* – create a directory dir
rm *file* – delete file
rm -r *dir* – delete directory dir
rm -f *file* – force remove file
rm -rf *dir* – force remove directory *dir* USE WITH CARE!
cp *file1 file2* – copy *file1* to *file2*
cp -r *dir1 dir2* – copy *dir1* to *dir2*; create *dir2* if absent
mv *file1 file2* – rename or move *file1* to *file2* if *file2* is an existing directory, moves *file1* into directory *file2*
ln -s *file link* – create symbolic link *link* to *file*
touch *file* – create or update file
cat > *file* – places standard input into file
more *file* – output the contents of file
head *file* – output the first 10 lines of file
tail *file* – output the last 10 lines of file
tail -f *file* – output the contents of file as it grows, starting with the last 10 lines

Process management

ps – display your currently active processes
top – display all running processes
kill *pid* – kill process id pid
killall *proc* – kill all processes named proc *
bg – lists stopped or background jobs; resume a stopped job in the background
fg – brings the most recent job to foreground
fg *n* – brings job n to the foreground

File Permissions

chmod *octal file* – change the permissions of file to octal, which is specified separately for user, group, and world by adding: 4: read(r) 2: write (w) 1: execute (x)
Examples: (For more, see **man chmod**)
chmod 777 – read, write, execute for all
chmod 755 – rwx for owner, rx for group and world

SSH

ssh *user@host* – connect to *host* as *user*
ssh -p *port user@host* – connect to *host* on *port* as *user*
ssh-copy-id *user@host* – add your key to *host* for *user* to enable a keyed or passwordless login

System Info

date – show the current date and time
cal – show this month's calendar
uptime – show current uptime
w – display who is online
whoami – who you are logged in as
uname -a – show kernel information
htop – cpu & memory information [must install!]
man *command* – show the manual for *command*
df – show disk usage
du – show directory space usage
ncdu – navigable, hierarchical disk usage [must install!]
whereis *app* – show possible locations of app
which *app* – show which app will be run by default
whatis *app* – show basic description of app

File Compression

tar cf *file.tar files* – create a tape archive (tar) named *file.tar* containing *files* (See man tar for more options)
tar xf *file.tar* – extract the files from *file.tar*
tar czf *file.tar.gz files* – make tar with Gzip compression
tar xzf *file.tar.gz* – extract a tar using Gzip
gzip *file* – compresses file and renames it to *file.gz*
gzip -d *file.gz* – decompresses file.gz back to *file*

Network

ping *host* – ping host and output results
wget *file* – download file
wget -c *file* – continue a stopped download

Installation

sudo apt install *program* – installs *program* from an online package repository (a PPA)
sudo apt remove *program* – uninstalls *program*
dpkg -i *pkg.deb* – install from an installer file (Debian)
rpm -Uvh *pkg.rpm* – install from installer file (RPM)

Shortcuts

Ctrl+C – halts the current command
Ctrl+Z – stops the current command, resume with fg in the foreground or bg in the background
Ctrl+D – log out of current session, similar to exit
Ctrl+W – erases one word in the current line
Ctrl+U – erases the whole line
Ctrl+R – type to bring up a recent command
!! - repeats the last command
exit – log out of current session