

http://bl831.als.lbl.gov/~jamesh/powerpoint/unix_commands.ppt

Slightly more advanced:

<http://bl831.als.lbl.gov/~jamesh/elves/manual/tricks.html>

Basic unix commands that everyone should know

(Even if you have a mac)

What the ~*&?!

- ~ "tilde" indicates your home directory: `/home/you`
- * "star": wildcard, matches anything
- ? wildcard, matches any one character
- ! History substitution, do not use
- & run a job in the background, or redirect errors
- #% special characters for most crystallography programs
- ` \ ([" ' back-quote, backslash, etc. special to shell
- _ underscore, use this instead of spaces!!!

Where am I?

pwd

Print name of the “current working directory”

This is the default directory/folder where the shell program will look first for programs, files, etc. It is “where you are” in Unix space.

What is a directory?

/home/yourname/whatever

Directories are places you put files. They are represented as words connected by the “/” character. On Windows, they use a “\”, just to be different. On Mac, they are called “folders”.

Whatever you do...

DO NOT PUT SPACES

In directory/file names!

What have we here?

ls

List contents of the current working directory

```
ls -l      - long listing, with dates, owners, etc.  
ls -lrt   - above, but sorted by time  
ls -lrt /home/yourname/something  
          - long-list a different directory
```

Go somewhere else?

cd

Change the current working directory

```
cd /tmp/yourname/  
          - go to your temporary directory  
cd -     - go back to where you just were  
cd       - no arguments, go back "home"  
          "home" is where your login starts
```

A new beginning...

mkdir

Create a new directory.

```
mkdir ./something - make it  
cd ./something - go there  
ls - check its is empty
```

How do I get help?

man

Display the manual for a given program

```
man ls - see manual for the "ls" command  
man tcsh - learn about the C shell  
man bash - learn about that other shell  
man man - read the manual for the manual
```

to return to the command prompt, type "q"

Move it!

mv

Move or rename a file. If you think about it, these are the same thing.

```
mv stupidname.txt bettername.txt
  - change name
mv stupidplace/file.txt ../betterplace/file.txt
  - same name, different directory
mv stupidname_*.img bettername_*.img
  Will not work! Never ever do this!
```

Copy machine

cp

Copy a file. This is just like “mv” except it does not delete the original.

```
cp stupidname.txt bettername.txt
  - change name, keep original
rm stupidname.txt
  - now this is the same as “mv”
```

“Permission denied” !?

chmod

Change the “permission” of a file.

```
chmod a+r filename.txt
    - make it so everyone can read it
chmod u+rw filename.txt
    - make it you can read/write/execute it
chmod -R u+rw /some/random/place
    - make it so you can read/write everything under
    a directory
```

Destroy! Destroy!

rm

Remove a file forever. There is no “trash” or “undelete” in unix.

```
rm unwanted_file.txt
    - delete file with that name
rm -f /tmp/yourname/*
    - forcefully remove everything in your
    temporary directory.
    Will not prompt for confirmation!
```

less is more

more

Display the contents of a text file, page by page

`more filename.txt` - display contents

`less filename.txt` - many installs now have a replacement for "more" called "less" which has nicer search features.

to return to the command prompt, type "q"

After the download...

gunzip

File compression and decompression

`gunzip ~/Downloads/whatever.tar.gz`

- decompress

`gzip ~/Downloads/whatever.tar`

- compress, creates file with `.gz` extension

Where the %\$#& is it?

find

Search through directories, find files

```
find ./ -name 'important*.txt'
```

- look at everything under current working directory with name starting with "important" and ending in ".txt"

```
find / -name 'important*.txt'
```

- will always find it, but take a very long time!

Did I run out of disk space?

df du

Check how much space is left on disks

```
df - look at space left on all disks
```

```
df . - look at space left in the current working directory
```

```
du -sk . | sort -g
```

- add up space taken up by all files and subdirectories, list biggest hog last

Why so slow?

ps top

Look for programs that may be eating up CPU or memory.

`top` - list processes in order of CPU usage

`jobs` - list jobs running in background of current terminal

`ps -fHu yourname`

- list jobs belonging to your account in order of what spawned what

Die Die Die!

kill

Stop jobs that are running in the background

`kill %1` - kill job [1], as listed in "jobs"

`kill 1234` - kill job listed as 1234 by "ps" or "top"

`kill -9 1234` - that was not a suggestion!

`kill -9 -g 1234` - seriously kill that job and the program that launched it