

# #!/bin/bash

## File Navigation and Manipulation

ls - list directory contents  
-l long listing format  
-a include hidden (. prefix)  
-d list directories, not contents  
--sort=time/size

Globbering - \* wild card

pwd - print working directory

cd - change directory  
. current, .. up a level, ~ home

cp <src> <dst> - copy files  
-r copy directories recursively

mv <src> <dst> - move files

mkdir <dir> - make new directory

rm - delete files  
-r delete recursively

ln -s <original> <link> - create a link  
\*generally bad idea to not use -s\*

df - report file system disk space usage  
-h human readable

du - estimate file space usage  
-h human readable  
-s summarize

chmod - change file permissions

chown - change file owner

source - run commands in provided file

locate - find a file

find - traverse directory tree

which - locate a command binary

man <cmd> - show manual page for cmd

## awk

pattern scanning and processing language  
-F<x> make x the field delimitator  
(default whitespace)  
NF number of fields (columns) on line  
NR current record number  
\$0 full line  
\$<N> Nth field of line

Examples:

```
print first three fields of file
awk '{print $1,$2,$3}' file
print file without header
awk 'NR > 1 {print $0}' file
at line numbers to file
awk '{print NR,$0}' file
print first field of lines where
second field is greater than zero
awk '$2 > 0 {print $1}' file
```

## Bash History and Shortcuts

up arrow - previous command  
Ctrl-R - search history \*AWESOME\*  
Ctrl-D - EOF/logout/exit  
Ctrl-A - goto start of line  
Ctrl-E - goto end of line  
TAB - autocomplete file or command  
alias <alias>=<cmd> - create a shorter name for a command  
Ex: alias mroe=more

## Text Manipulation

cat - print out file contents  
more - paginate through file contents  
head - show first 10 lines of the file  
-n <N> show first N lines  
tail - show last 10 lines of a file  
-n <N> show last N lines  
-f output lines as file grows  
wc - print line, word, and character counts  
sort - sort contents of file, line by line  
-n numeric sort  
-kN,M sort based on columns N-M  
-r reverse sort  
-R random sort  
-u uniquify  
-o write output to file instead of stdout  
Ex: sort file.txt -o file.txt  
BAD: sort file.txt > file.txt  
uniq - count/omit repeated lines  
-c prefix lines by number of occurrences  
cut - cut out columns from a file  
grep <pattern> <file(s)> - search file contents  
-i ignore capitalization  
-r recursive search  
-I skip over binary files  
-s suppress error messages  
-n show line numbers  
-A<N> show N lines after match  
-B<N> show N lines before match  
sed - filter and transform text  
-i edit files in place  
Ex: sed 's/HIE/HIS/' file.pdb > new.pdb

<http://www.tldp.org/LDP/abs/html/>  
*variable manipulation - Loops and branches*

## I/O Redirection

> - send standard output to file  
Ex: cat \*.txt > all.txt  
>> - append to file  
< - get standard in from file  
| - pipe output to input  
Ex: ls | sort  
2> - send standard error to file  
Ex: cmd > output 2> errors  
>& - send output and error to file

## Loops

```
for i in [list]
do
  <cmd> ... $i ...
done

[list] examples
{1..24}
*.sdf
`cat files`
```

## Process Management

Ctrl-Z - suspect current job  
Ctrl-C - kill current job  
<cmd> & - execute cmd in background  
bg - start executing suspended job in background  
fg - bring suspended job to foreground  
top - display processes  
q to quit  
h show help  
jobs - show your jobs  
nohup <cmd> - don't kill cmd at logout  
disown - nohup running job  
ps - process listing  
pstree - show hierarchical tree of processes  
-p - show process ids  
kill <pid> - kill specific process  
kill -9 <pid> - kill with extreme prejudice  
killall <name> - kill all processes with name  
xargs - build command lines from standard input  
Ex: awk '{print \$1}' dirs.txt | xargs mkdir

## Environment Variables

echo <string> - print string to terminal  
env - print all set environment variables  
.bashrc - initialization file for bash  
export VAR=def - define VAR to def  
\$ - show variable contents  
Ex: echo \$PATH, echo \${PATH}  
\${i%x} - remove x from end of i  
Ex: convert \$i \${i%.jpg}.png  
`cmd` - evaluates to output of cmd  
PATH - where shell searches for executables  
LD\_LIBRARY\_PATH - library search path  
PYTHONPATH - where python searches for modules  
HISTCONTROL - controls history behavior  
export HISTCONTROL=ignoredups  
HISTSIZE - size of history buffer