UNIX Slightly Longer Reference Guide (still for newbies)

Shell | Directories | Files | Archiving | Processes | Network | Misc

The BASH Shell	
exit	exit the shell; log you out if it is the login shell.
bash	start the bash shell (in a subshell).

echo [*opt*] [*string*]	display a line of text. **Note:** this is built-in to most shells, so will vary. (POSIX):
	opt:
	-n – suppress newline
	-e – enable backslash characters (\n, \t, \a, \f, ...)
	-E – disable interpretation of backslash characters
	
	EXAMPLES
	`echo Is this a star *?`
	`echo "This is a star *"`
	`echo "This is my userid: $USER"`
	`echo 'This is my userid: $USER'`
	`echo "Today is `date`"`

command > *file*	**redirects** the output (stdout) from *command* to *file*. EXAMPLE `ls -1 > fileList`
command < *file*	**redirects** the input (stdin) to *command* from *file*. EXAMPLE `mail -s "Can't get any sleep" DearAbbey@ap.com < myWifeSnores.txt`

command1 | *command2*	output (stdout) from *command1* is **piped** to the stdin of *command2*.
	EXAMPLEs
	`ls -a | more`
	`ls -l /tmp | grep kschmidt | awk '{print $9}'`

Filesystem / Directories	
ls [*opt*] [*dir*]	lists contents of *dir* (current dir. by default) *opt*: **-a** all **-C** columns **-F** format **-l** long
pwd	**print working directory.** Shows you the current (working) directory.
cd [*dir*]	change directory to *dir* (user's home by default).
mkdir [-p] *dirname*	make directory. Creates *dirname*. **-p** make parent directories, as needed

rmdir *dirname*	remove directory. Removes *dirname*.
cp [*opt*] *src targ* **cp** [*opt*] *src... dir*	copies *src* to *targ*, or copy *src*(s) to *dir*. *opt*: **-i** interactive – if *targ* exists, prompt user before overwriting **-p** preserve file attributes, where possible **-R** copy directories recursively
mv [*opt*] *src target* **mv** [*opt*] *src... dir*	renames *src* to *targ*, or move *src*(s) to *dir*. *opt*: **-i** interactive – if *targ* exists, prompt user before overwriting
rm [*opt*] *targ...*	deletes (removes) *targ*(s). *opt*: **-i** interactive – prompt user before actually unlinking each file **-r** recursive (careful!) **-f** force
Files	
cat [*file*]...	displays (catalogs) the contents of *file*(s) (stdin by default).
more [*file*]...	displays *file*(s) (stdin by default) one screen at a time.
less	enhanced version of `more`. Not available everywhere
vi	the standard UNIX text editor. Know it.
head -*n* [*file*]...	Prints the first *n* (default 10) lines of *file*(s) (default stdin).
tail -*n* [*file*]...	Prints the last *n* (default 10) lines of *file*(s) (default stdin).
wc [*opt*] [*file*]...	**word count** Reports on number of lines, characters, and words in a *file*(s) (default stdin). *opt*: **-c** # of bytes **-m** # of chars (same, if ASCII text file) **-l** # of newlines **-L** length of longest line **-w** # of words
grep *pattern* [*file*]...	Searches *file*(s) (stdin) for *pattern* (a regular expression). Prints matching line, by default. Again, there are different greps out there, so not all options available (-q, -r). *opt*: **-c** just print a count of matching lines **-i** ignore case **-n** prefix output w/line numbers **-q** quiet; no output (just return 0 if match is found) **-r** recursive; read files under listed directories
find *path* [*opt*]	

	Searches subtree rooted at *path*, prints filenames that pass all subsequent tests (left to right) listed in *opt*.
	opt:
	Numeric args can be specified as:
	+n greater than *n*
	n exactly *n*
	-n less than *n*
	Tests:
	-mtime *n* modified *n* * 24 hours ago
	-name *pattern* name matches *pattern* (be careful to quote, if using wildcards).
	-iname like -name, but case insensitive
	-size *n* file used *n* * 512 bytes
	-type *c* if file is of type *c*, where *c* is one of d, f, and l, for directory, regular file, and symbolic link, respectively (partial list).
	Actions
	-exec *command* ; execute *command*; true if 0 is returned. `{}` is replaced by current filename.
	-print true; prints filename to stdout
	
	EXAMPLES
	Find all files in current subtree that have "resume" in the name:
	`find . -iname "*resume*" -print`
	Find all files in my mail directory that mention "motorcycle":
	`find ~/mail -exec grep -iq motorcycle {} \; -print`
	Find all regular files in my Web directory modified in the past 2 days:
	`find ~/public_html -type f -mtime -2 -print`

sort [*opt*] [*file*]...	sorts *file*(s) (default stdin), to stdout. See man pages. *opt*: **-f** ignore case (fold lower to upper) **-k*p1*,[*p2*]** fields to sort, left to right **-m** merge already sorted files **-n** compare in numerical sense **-r** reverse; sort in descending order
Archiving	
tar *oper* [*opt*] [*file*]...	Tape ARchive – to create or restore archives. Makes many files into one, or vise versa. Note, the form of the options is changing, as is default behavior. Do a man, or info. *oper* is one of: **c** – create **x** – extract **t** – table of contents

	opt:
	f *filename* – the name of the archive (default is the tape). Use '-' for stdin/stdout
	v – verbose
	z – read or write through **gzip** (Linux only)
	
	EXAMPLES
	To archive all contents of current directory to file **backup.tar**:
	`tar cvf backup.tar *`
	To extract some archive on the floppy to the current directory:
	`tar xvf /mnt/floppy/backup.tar`
	To archive and gzip all of your public_html files to **~/web.tgz**:
	`tar cvf - ~/public_html | gzip > ~/web.tgz` , or:
	`tar czvf ~/web.tgz ~/public_html` (where available)
	To extract a gzipped tar file:
	`gunzip -c back.tar.gz | tar xvf -` , or:
	`tar xvzf back.tar.gz` (where available)

gzip, gunzip [*file*]...	To compress and decompress *file*(s) (not the `compress` utility). Adds or looks for .gz extension, by default. See man pages. *opt*: **-c** – to stdout NOTE If input is stdin, then output is to stdout
unzip	Decompresses DOS Zip files (made by PKWare).
Processes	
ps [*opt*]	process status. Reports on processes. *opt*: **-e** all users **-f** full listing (more info) **-F** format **-l** long
kill [*opt*] *pid*	Send signals to a process with ID *pid*. *opt*: **-9** SIGKILL (The one to kill a process, and its children.) **-15** SIGTERM (A bit nicer, asks a process to end.)
Network	
telnet *host*	allow you to telnet (log in and start a session) to *host*. Use ssh for CS Dept. machines.
ssh *userId@host*	Provides a terminal i/f, much like telnet, but secure. Not a standard Unix tool (yet).
scp [*userId@host*:]*src* [*userId@host*:]*targ*	Secure copy. Much like FTP, but secure, and does no CR/LF translations. Prompts for a password, if req'd. Not standard.

mail	The standard UNIX mail reader. Don't bother.
elm	A much nicer mail reader.
pine	Another nice mail reader, understands MIME and UUEncode types.
Misc. Unix	
man [*sect*] [-k *keyword*] *item*	man pages. Displays help on *item*, in *sect*. *opt*: **-k** keyword. Search entries for *keyword*.
passwd, **yppasswd**	To change your password.
chsh, **ypchsh**	To change your login shell.

This collection is, by far, incomplete. Just a quick and dirty list to get going, for budding programmers new to Unix. If you think of something that should really be in here (or see any errors), please let me know: *kschmidt@cs.drexel.edu*

Last updated 7/24/2011