Linux Quick Reference Guide

Basic commands

pwd	Prints the current working directory.
cd	Changes to your top-level home directory (usually `/home/`*yourusername*).
cd *dir*	Changes to the specified directory.
ls	Lists the contents of the current directory.
ll	Long directory listing, showing files permissions, ownership, size, and last modification date.
ls -a or **ll -a**	Shows hidden files in addition to regular files. Hidden file names start with a period.
cp *file1 file2*	Makes a copy of *file1* named *file2*.
cp -r *dir1 dir2*	Makes a copy of an entire directory recursively and names it *dir2*.
mv *file1 file2*	Renames *file1* to be *file2* (overwrites *file2* if it already exists).
mkdir *dir*	Creates a new empty directory called *dir*.
rmdir *dir*	Removes a directory (must be empty).
cat *file*	Outputs the entire contents of a text file to the screen.
less *file*	Displays a text file one screenful at a time using a pager.

Pager commands

Space	Advance to the next screenful of information.
q	Quit pager and returnse to Linux shell.
b	Go back one screenful.
h	View help for pager.
Return or **↓**	Move forward one line.
↑	Move backward one line.
/*string*	Search for an occurrence of *string* in the file.
n	Search forward for the next occurrence of *string*.

man *command*	Prints available help for a command; output will come up in a pager.
grep *string file*	Prints out all lines of *file* containing *string*
grep -v *string file*	Prints out all lines of *file* **not** containing *string*
emacs &	Opens the Emacs editor in a separate window.
emacs *file* **&**	Runs Emacs on the specified file in a separate window.
emacs -nw *file*	Runs Emacs within the current terminal window (but **without** mouse support).
python	Starts the Python interpreter.
python -i *file***.py**	Starts the Python interpreter and executes the Python program *file*`.py`
display *graphicsfile*	Displays a graphics file in a pop-up window.
du -sh *file/dir*	Prints a summary of the disk usage for the specified file or directory.
exit or **logout**	Logs out of the current shell.

Compressing and archiving files

gzip *file*	Compresses *file*. The extension `.gz` will be added to the filename.
gunzip *file***.gz**	Uncompresses *file*`.gz`. The extension `.gz` will be removed from the filename.
tar cfz *file***.tgz** *dir*	Creates a compressed archive file named *file*`.tgz` containing all of the files in directory *dir*.
tar tfz *file***.tgz**	Lists the contents of the compressed archive *file*`.tgz`, but do not extract the files.
tar xfz *file***.tgz**	Extracts all of the files archived in *file*`.tgz` to the current directory.

Shorthand symbols for directories

~	User's home directory
~*username*	*username*'s home directory
.	Current directory
..	Parent directory

Examples

```
cd ~
ls ~joeschmoe
cp ~joeschmoe/data.txt .
mv data.txt ..
```

Wildcard symbols for filenames

*****	Matches zero or more characters.
?	Matches exactly one character.
[]	Specifies a range of characters to match.

Examples

```
ls data.*     cp *data* ..     rm *.java
cp data?.txt results/
ls data[0-9].txt
```

Using the shell

- The *shell* is the Linux command-line interpreter. These instructions assume that you are using the `bash` shell. You can check this by typiing the command `echo $SHELL`
- To set an environment variable, type `export` *variable=value*. This sets the value for the current shell session. To set an environment variable for all new shells that you create, add this line to the file `~/.bashrc`
- To avoid typing the same command over again, you can use *Ctrl*-P and *Ctrl*-N (or ↑ and ↓) to cycle through the previous commands you have entered.
- You can press the `TAB` key in many situations to complete filenames or commands for you. Hitting `TAB` twice will show you a list of the available completions in case it is ambiguous.
- You can edit commands directly on the command line using the following Emacs-like keyboard shortcuts:

Ctrl-B and *Ctrl*-F (or ← and →)	Moves cursor to the left or right.
Ctrl-A and *Ctrl*-E (or *Home* and *End*)	Moves cursor to beginning or end of line.
Ctrl-H (or *Backspace*)	Erases character to the left of the cursor.
Ctrl-D (or *Delete)*	Erases character to the right of the cursor.
Ctrl-K	Deletes/cuts everything from the cursor to the end of the line.
Ctrl-U	Deletes/cuts everything from the beginning of the line to the cursor.
Ctrl-C	Cancels the current command.
Ctrl-R	Searches backwards for a previously-entered command.
Ctrl-L	Clears the screen.
Ctrl-Y	Pastes back characters previously cut using *Ctrl*-K or *Ctrl*-U.

Processes and job control

command **&**	Runs a command in the background.
Ctrl-Z	Suspends execution of a program.
jobs	Lists all currently running or suspended jobs (programs).
fg	Runs the most recently suspended job in the foreground, disallowing new shell input.
bg	Runs the most recently suspended job in the background, allowing the shell to accept new input.
fg %1	Runs job number 1 in the foreground.
bg %2	Runs job number 2 in the background.
ps aux	List all processes that are currently running.
ps aux|grep *program*	List all processes named *program* that are currently running.
pgrep *program*	Prints out process ID numbers of all processes named *program*.
pkill *program*	Attempts to kill all processes named *program* (owned by the user).
killall *program*	Attempts to kill all processes named *program* (owned by the user).

Remote logins and file copying

ssh *username***@***hostname*	Log in securely to a remote machine named *hostname*.
wget *URL*	Download a file from a web site specified by *URL*.
scp *file username***@***hostname***:**	Copy *file* securely from the current local directory to a remote machine.
scp *username***@***hostname***:***file* **.**	Retrieve *file* securely from a remote machine to the current local directory.
scp *file1 username***@***hostname***:***file2*	Copy *file1* securely to a remote machine, and name the remote copy *file2*.
scp *username***@***hostname***:***file1 file2*	Retrieve *file1* securely from a remote machine, and name the local copy *file2*.
scp -r *dir1 username***@***hostname***:***dir2*	Copy an entire directory securely to a remote machine.
scp -r *username***@***hostname***:***dir1 dir2*	Retrieve an entire directory securely from a remote machine.