Since you probably work with your fair share of frameworks libraries, you'll often find yourself downloading these files as you work. Oh, I know: you can just download it from the web, navigate to the folder, uncompress it, and copy the pieces to your project, but doesn't that sound like so much work? It's much simpler to use the command line. To download files, you can use `curl`; proceed as follows:

```
$ curl -O http://www.domain.com/path/to/download.tar.gz
```

The `-O` flag tells `curl` to write the downloaded content to a file with the same name as the remote file. If you don't supply this parameter, curl will probably just display the file in the commmand line (assuming it's text).

```
$ cp originalFile newFile
$ vim newFile #edit newFile
$ diff originalFile newFile
1c1
< This is a sentence.
---
> This is a short sentence.
$ diff originalFile newFile > changes.patch
```

As you can see, the `diff` is just a simple text file that uses a syntax the `diff` and `patch` command will understand. `Patch`? Well, that's the command that goes hand in hand with `diff`. If you've received a patch file, you'll update the original as follows:

As you can see, the `diff` is just a simple text file that uses a syntax the `diff` and `patch` command will understand. `Patch`? Well, that's the command that goes hand in hand with `diff`. If you've received a patch file, you'll update the original as follows:

```
patch originalFile2 changes.patch
```

And now you're all updated.

```
$ ditto -V /old/work/ /new/work/
```

The above command will copy our “work” folder to a new location. Actually, what happens is the command copies the *contents* of our “work” folder to a new “work” folder.

Adding **-V**, meaning *verbose* prints a line to the Terminal window for every file that’s being copied. It’s extremely useful as you can see exactly what file is being copied at any time.

Ditto is a command I use frequently and have often copied many gigabytes of data between hard drives using it because it seems to be so reliable.

If we ever take a screenshot of a window in OS X, by default it will always show a drop shadow, adding wasted pixels. If you'd prefer to have your screenshots drop shadow-free, you can use the following command:

```
$ defaults write com.apple.screencapture disable-shadow -bool TRUE
```

You can either restart your Mac to see the changes or, using another command, see them instantly:

```
$ killall SystemUIServer
```

Change Screenshot File Format

PNG is probably the best format to use for screenshots, but it can certainly take up a bit of space. If you'd prefer to use PDF format, you can use the following:

```
$ defaults write com.apple.screencapture type PDF
```

```
$ killall SystemUIServer
```

To revert the changes, enter the following:

```
$ defaults write com.apple.screencapture type png
```

```
$ killall SystemUIServer
```

If your Mac doesn't currently support AirDrop, you can enter the following command to use it, as well as to use it over ethernet:

```
$ defaults write com.apple.NetworkBrowser BrowseAllInterfaces -boo
```

You'll need to restart the Finder, the quickest way is to use the **killall** command which will restart it:

```
$ killall Finder
```

To revert the changes, enter:

```
$ defaults write com.apple.NetworkBrowser BrowseAllInterfaces -boo
```

Stress Test Your Mac

If you're wanting to run some stress tests on your Mac, there's a simple command we can enter that will instantly utilise 100% of your CPU:

```
$ yes
```

Apple technicians use this frequently when dealing with troublesome Macs that might be crashing under load and it's a very quick way of stressing a Mac. To cancel the command, press **Ctrl-C**.

```
sudo cp /etc/rc.conf{,-old}
```

Putting nothing before the comma will just append `-old` to the filename after copying it with `cp`. If your new file doesn't work out and you want to restore the backed up file to its original location, you can just use:

```
sudo mv /etc/rc.conf{-old,}
```

Moving the comma to the other end of the brace will remove `-old` from the end of the file and restore it to its original name.

Get Your Network IP Address

Sure, we could open *System Preferences*, select *Network* and then view our IP address information, but doing it through the command line is so much cooler!

```
$ ipconfig getifaddr en0
```

The term **en0** represents the network interface to use. Similar to some programming languages, the first network interface starts at zero and then counts up, so if you have two interfaces (such as ethernet and WiFi) then they would be **en0** and **en1**, respectively.

By default, your Mac will always designate a wired network connection as **en0**, unless your Mac doesn't have built-in ethernet. You can substitute **en0** with **en1** if you're not using your wired connection.

Remove Duplicate “Open With…” Entries

A common bug in OS X is seeing duplicate apps within the “*Open With…*” menu which can get quite full! To fix it:

```
$ /System/Library/Frameworks/CoreServices.framework/Frameworks/Lau
```

The above command will work in OS X Leopard and above. Unlike the other commands, I’d recommend restarting your Mac.

The braces can also work when moving or creating multiple files at once. For example, if you wanted to create three numbered directories, you could just run:

```
mkdir myfolder{1,2,3}
```

This will create three folders: myfolder1, myfolder2, and myfolder3.

Rebuild Spotlight

Spotlight can occasionally not work as well as we'd like. Sometimes, we just need to tell Spotlight to create a new index so it can find files as fast and efficiently as possible. To do this:

```
$ sudo mdutil -E /Volumes/DriveName
```

The above command will delete Spotlight's index, forcing it to start a new one. An easier way of completing the command is to enter the first part of it and, for the location, just drag a hard drive from your desktop to the Terminal window.

Create a new workflow and select Service from the template chooser.

Save it and give it a name that will make sense when you see it in the Services menu.

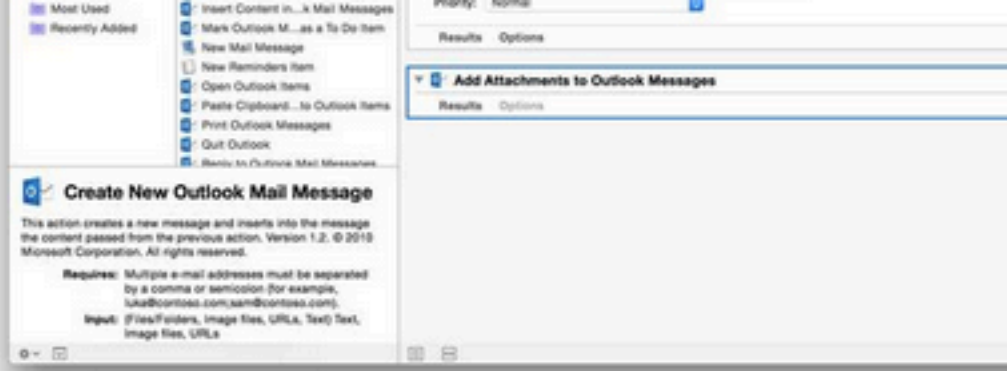
Go to Mail in the Library and choose Create New Outlook Mail Message (you'll only see this as an option if you have Outlook installed).

You'll need to select an email account from which to send the message, but you can leave the rest of the form blank if you want.

If you do, you'll be able to fill in the message details when the workflow runs.

Now drag Add Attachments to Outlook Messages as the second step in the workflow and save it.

Right-click on any file in the Finder, navigate to Services and you'll see the Service you just created.



Test Network Connectivity

You're likely to have heard of the term "ping" at some point. Ping sends very small bits of information over a network to a remote computer, timing how long it takes for a response to be received. It's useful to see if an IP address is working or if a website might be down.

```
$ ping -c 10 www.apple.com
```

Running the above command will send 10 packets of information and provide detailed information about the response time. You can substitute *www.apple.com* for any other domain name or IP address.

Install OS X Software Updates

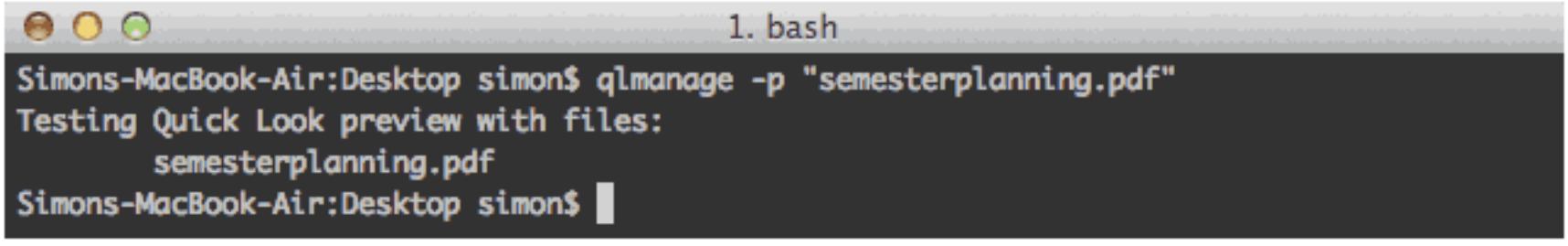
Despite Software Updates moving to the **App Store** in Mountain Lion, we're able to use the command line to install system udates without having to launch it. To see available software updates for your Mac:

```
$ sudo softwareupdate -l
```

After a few minutes, you'll be given a list of available updates.

If you'd like to install all available updates, enter:

```
$ sudo softwareupdate -ia
```



To open a file in Quick Look use the following command, substituting **FILE_PATH** for a reference to your file:

`qlmanage -p` **FILE_PATH**

```
nanp /path/to/a/document/buried/deep/in/the/filesystem
```

Instead of retyping the whole thing, you could just run:

```
^nanp^nano
```

This will find the first instance of `nanp` in the last run command and replace it with `nano`.

Continually Monitor the Output of a File

This one is for all the budding system administrators out there. If you'd like to keep monitoring a text file and view any changes to it as they're made, there's a suitable command that will constantly monitor your chosen file and display any new lines as they're added, perfect for monitoring system log files.

```
$ tail -f /var/log/system.log
```

Your Terminal window will constantly watch your specified file (in this case, the *system.log* and every time another line is added, it will print it on the screen.

To cancel, press **Ctrl-C**.

Expansions

When you're working with variations of a file—like backups or different file types—it can get tedious typing out the same commands with small tweaks. Using the brace symbols (`{}`), you can easily perform batch operations on multiple versions of a file.

Say you want to rename just part of a filename. Instead of typing out `mv /path/to/file.txt /path/to/file.xml` , you could just run:

```
mv /path/to/file.{txt,xml}
```

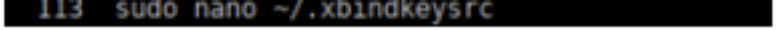
Start a Simple HTTP Server in Any Folder

If you're needing to quickly test some HTML that you're working on, start a simple web server within *any* folder on your Mac. Navigate to the folder to use and enter:

```
$ python -m SimpleHTTPServer 8000
```

The number at the end is the port to use, open your browser and visit *http://localhost:8000* You can use the default of port 80 if you wish and remove the port number entirely.

When you're finished, simply press **Ctrl-C**.




```
history | grep nano
```

You'll get a list that looks something like this:

```
381 sudo nano /etc/NetworkManager/nm-system-settings.conf
387 sudo nano /etc/rc.conf
388 sudo nano /etc/rc.conf
455 sudo nano /boot/grub/menu.lst
```

You can then pick a command out from that list—say I want to run `sudo nano /boot/grub/menu.lst`, which grep lists as command `455`—and run it using:

```
!455
```

Disable Mail's Reply Animation in Mountain Lion

Mail has a great looking animation whereby if you hit reply to any message, a compose window animates into view. It's not for everyone and sometimes just having the window appear instantly. To turn off the animation:

```
$ defaults delete com.apple.mail DisableReplyAnimations -bool TRUE
```

Quit and relaunch Mail for the changes to take effect. To revert the changes:

```
$ defaults delete com.apple.mail DisableReplyAnimations -bool FALS
```

you need to navigate to the directory where you want the link.

```
cd /usr/local/bin
sudo ln "/System/Library/Automator/Combine PDF Pages.action/Contents/Resources/joi
```

Terminal will ask you for your password (you need Admin rights to do this). After the command is run, you will be able to concatenate PDF files anywhere on Terminal by typing

```
PDFconcat -o PATH/TO/YOUR/MERGED/FILE.pdf /PATH/TO/ORIGINAL/1.pdf /PATH/TO/ANOTHEF
```

You can even pass a 'shuffle' argument to make the script take one page per document in turn instead:

```
PDFconcat -s -o PATH/TO/YOUR/MERGED/FILE.pdf  /PATH/TO/DIR/*.pdf
```