1 Example Chapter

This is an example chapter from *Building Ruby Gems*, a book from Andy Croll.

You can get a discount by going to http://andycroll.com/building-ruby-gems/ and signing up with your email address. The book will be released in 2014.

Here's a brief overview of the content of the entire book.

Chapter 1: Setup - Describing our example gem, installing Ruby & Bundler

Chapter 2: Creating Our Gem - *You're reading it*

Chapter 3: Gem Structure - Explaining the basic structure of our initial gem

Chapter 4: Testing - Using MiniTest to test drive your gem, Continuous Integration using Travis

Chapter 5: Shipping - Publishing and distributing our gem through rubygems.org, gem versioning, changelog

Chapter 6: Our Code - The code for our example gem, structuring code, internal structure

Chapter 7: Documentation - The README, Badges, using the Yard documentation language, rdoc.info, Github pages

Chapter 8: Stewardship - Running an open source project, issues, pull requests, communication

Throughout we'll be using a (simple) gem I wrote, `movie_titleize` as an example throughout.

2 Initial Gem Setup

You've got Ruby & Bundler installed which means we're only a few simple commands away from getting our gem created.

Let's go

bundle gem movie_titleize

```
bundle gem movie_titleize
```

...gives the following output...

```
      create  movie_titleize/Gemfile
      create  movie_titleize/Rakefile
      create  movie_titleize/LICENSE.txt
      create  movie_titleize/README.md
      create  movie_titleize/.gitignore
      create  movie_titleize/movie_titleize.gemspec
      create  movie_titleize/lib/movie_titleize.rb
      create  movie_titleize/lib/movie_titleize/version.rb
Initializing git repo in /Users/andy/workspace/gems/movie_titleize
```

Git

You'll have noticed the final line in that output talks about intializing a git repository... your tools are guiding you.

You pretty much have to use version control to write a gem, so Bundler gives you helping hand and sets up version control for you. Kind, no?

I'm presuming as a Ruby developer you have at least a passing knowledge of `git`. So the following commands should not look too foreign.

```
cd movie_titleize
git add .
git commit -m 'initial commit'
```

All we've done here is to set up the local repository, the point of all this gem malarky is to share our code, make it public. So... off to Github!

Github

It's pretty standard to host your source code for a gem on Github. Github host public projects for free. Issues and pull requests are a great way to collaborate on your code with other developers.

I'm going to presume you have a Github account and have set things up correctly on your machine.

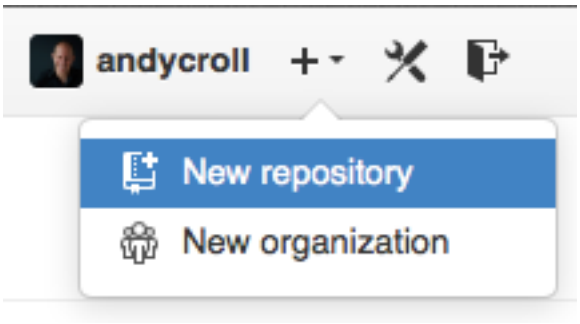
If you're having trouble, then hit up the Github help pages.

Create a new repository

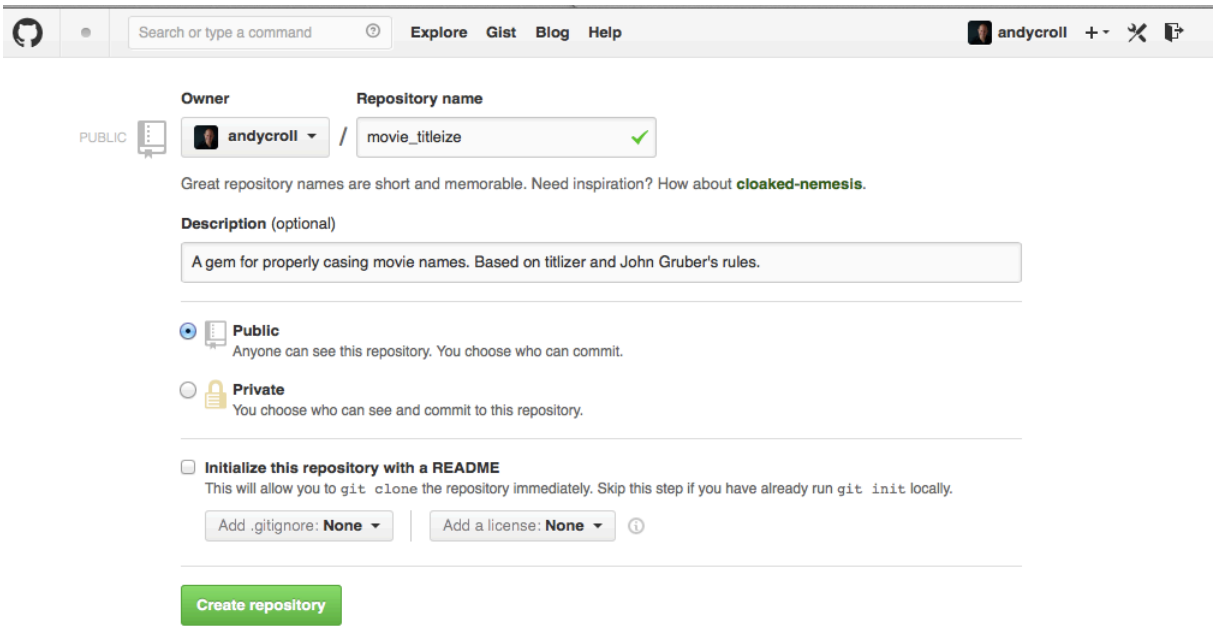
You can always get the latest instructions direct from Github, but it's pretty simple 'clicking about on a website' stuff.

Head on over to Github and sign in.

At the top right, select the 'New Repository' link from the drop down menu.



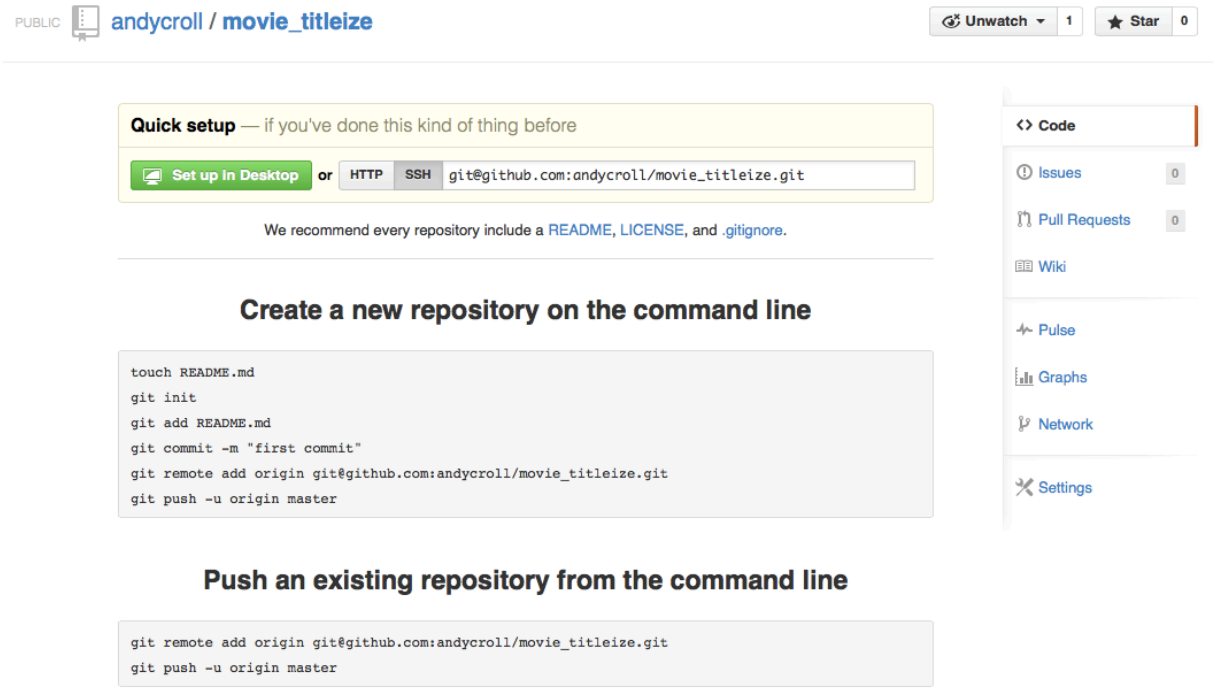
You'll be confronted with the new repository form.



Pick a clever and/or witty name. The ruby community does seem to love a pun and/or putting the letter 'r' at the beginning of gem names.

I'm more likely just to pick something that describes what the gem does, but then I lack imagination.

Once you click the 'Create Repository' button you'll be left with the empty repository screen.



As you can see we've already done some of the required commands for creating from teh command line, the last thing we need to do is to link our local repository to this new Github one and push our

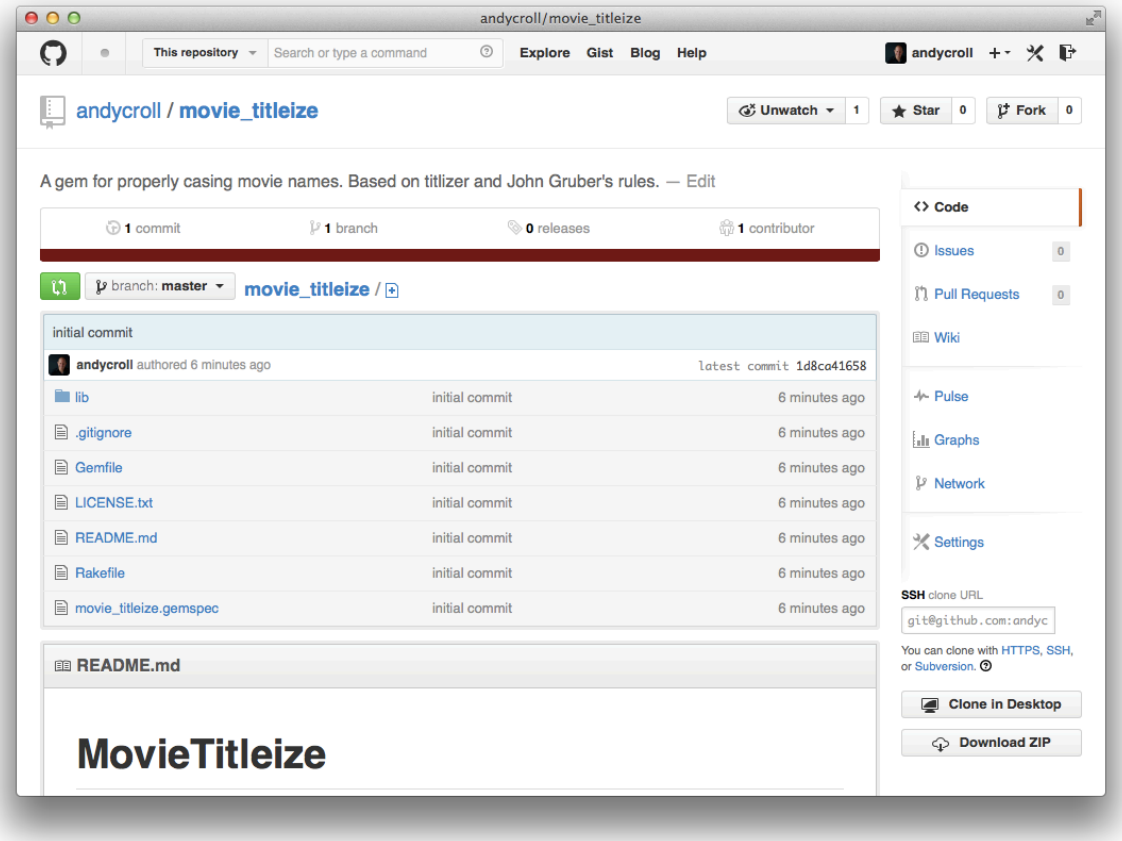
changes up.

```
git remote add origin git@github.com:andycroll/movie_titleize.git
git push -u origin master
```

Job done. If you refresh your browser you'll see the code has been pushed up.

You can see the first commit of the `movie_titleize` project on Github.

commit: 1d8ca41658



Doesn't do much though, we need to fix that.

Up until now I've had you typing commands blindly into a a console and clicking around on Github. So now, I guess it's time to explain myself...