Deleting Git Tags Locally and on Github

08 June 2015

So, I'm currently in the depths of helping a client with getting the front-end of their design language off the ground and also a neat little Yeoman generator to allow anyone working for them to quickly spin up some basic HTML and use the design language.

Git tags can be used to specify points in the history of a project as something important. Usually these are used to mark releases(v0.0.01, and so on). I'm using tags in these projects so that my client can easily update their copy of the Yeoman generator on their machine when a new feature or bugfix has been implemented.

The design language is making use of [Bower](#) so that it gets pulled in (or not) nice and simple into the Yeoman framework that's being created.

Using both Bower and Yeoman has required me tagging some commits to the repo with a version number.

At some point, probably a Friday, I was getting annoyed with the version numbers I was using expotentially on something that wasn't near completion. I decided to nuke all the git tags both locally and remotely so that I could (cheekily) start again.

Delete all Git tags locally

To delete all the git tags locally I happened upon this line of code to enter into the command line. This wil irrecoverably remove any and all of the git tags within the folder you enter the code in terminal.

```
git tag -d `git tag | grep -E '.'`
```

Delete all tags in a remote repo

Finally to delete all of the tags in the remote repo also I found this line of code that did the job. This wil irrecoverably remove any and all of the git tags from the attached remote server of the folder you enter the code in terminal.

```
git ls-remote --tags origin | awk '/^(.*)(s+)(.*[a-zA-Z0-9])$/ {print ":" $2}' | xargs git push origin
```

Some pretty horrendous and scary looking regex'd git codings but it worked perfectly.