**Make a new repository in github - demo9**

// do this in VS

// It'll take you to gihub respo where files are

**git remote add origin** [**https://github.com/keno1se/demo9.git**](https://github.com/keno1se/demo9.git)

**git remote add origin** [**https://github.com/keno1se/harmonicascaler.git**](https://github.com/keno1se/demo9.git)

//this base-href comes into 404.html

**ng build --prod --base-href** [**https://keno1se.github.io/demo9/**](https://keno1se.github.io/demo9/)

**ng build --prod --base-href** [**https://keno1se.github.io/harmonicascaler/**](https://keno1se.github.io/demo9/)

//send to git

**ngh --dir dist/demo**

\*\*\* Successfully published!

Now type this in any browser and see you site

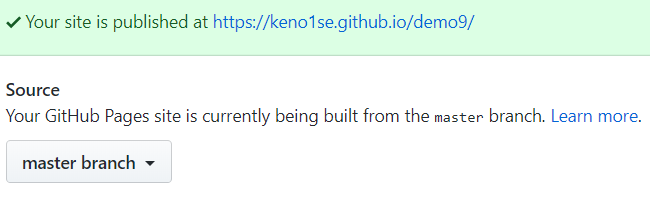
[**https://keno1se.github.io/demo9/**](https://keno1se.github.io/demo9/)

Every time you do some changes you need to do agin ng build and ngh

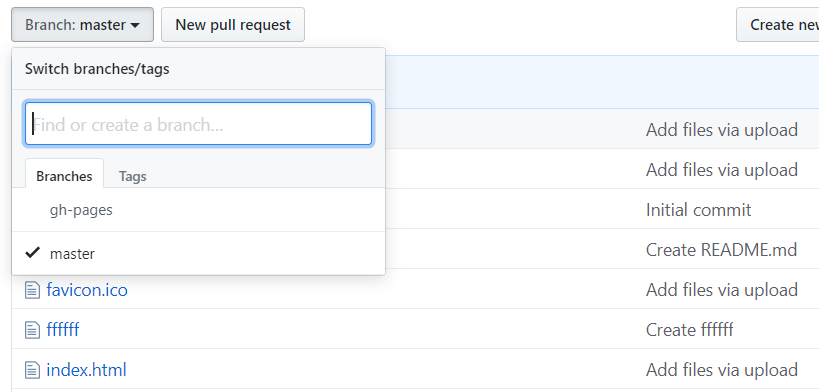
I most cases, the [PROJECTNAME] can be found in the angular.json file at defaultProject. - last in file

**Working and render with master branch files**

Master branch files you upload manually and not using git.   
To **render** those you go settings and in Source select Master branch



To **view** those uploaded files select master.  
 - see Add files via upload

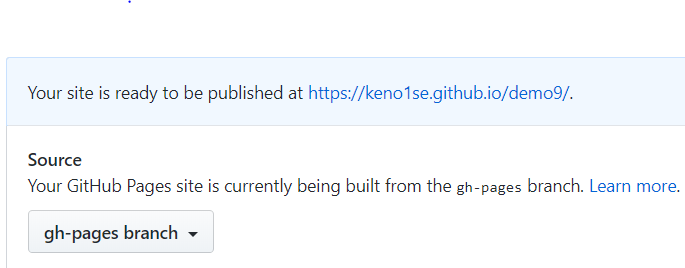


These files are build with  
**ng build --prod**  
and end up in dist folder, copy all dist folder files to master

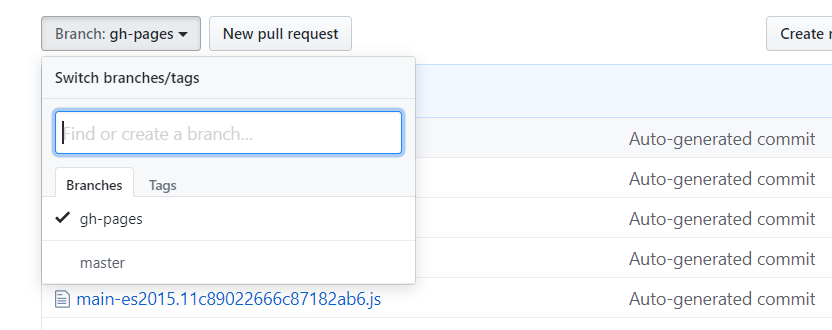
**Working and render with ngh git commited files.**

You don't have to manually upload those files - git does that for you.

To **render** those you go settings and in Source select

gh-pages branch

To **view** those ngh git commited files select gh-pages.  
 - see Auto-generated commit



These files are build with  
**ng build --prod --base-href** [**https://keno1se.github.io/demo9/**](https://keno1se.github.io/demo9/)  
and end up in dist folder, upload files with ngh command  
**ngh --dir dist/demo**

**// how to set up git in vs**

[**https://code.visualstudio.com/docs/editor/versioncontrol**](https://code.visualstudio.com/docs/editor/versioncontrol)

**// this is a video about above page**

[**https://www.youtube.com/watch?v=wMqukSKYcvU**](https://www.youtube.com/watch?v=wMqukSKYcvU)

// git hompage site

<https://git-scm.com/docs/git>

//this calculator was my first working deployment to github, demo9, see above

<https://dzone.com/articles/how-to-deploy-an-angular-application-to-github>

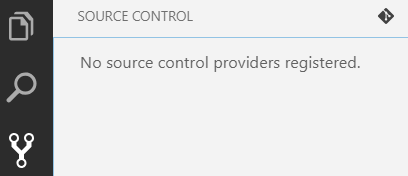
<https://alligator.io/angular/deploying-angular-app-github-pages/>

<https://medium.com/coinmonks/how-to-deploy-an-angular-app-8db1af39f8c1>

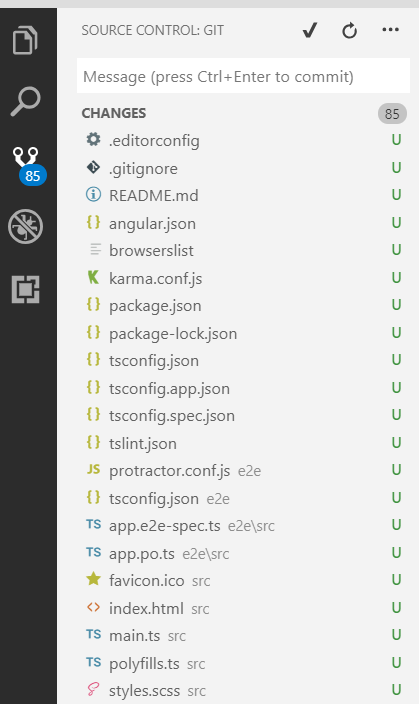
<https://medium.com/code-sketch/how-to-deploy-an-angular-7-app-to-github-pages-9427b609645f>

**How to init git in VS for a project**

You have downloaded and installed git on you pc. From there you can do a git init or you can go to github to clone a repositary. The simplest way is to init a project using VS.   
Open your project in VS, Click on the source control icon - Y, up on the right click that square, a dialog starts where you can init git for this project.



It then looks like this, 85 files i U unstaged state.



Now you can open a terminal and execute all the git commands from there, no need to do it from the downloaded git.

**Setting up a remote repository**

First install the angular-cli-ghpages globally:

**npm install -g angular-cli-ghpages**

This is one way to setting up remote,from vs terminal 

with echo you can read to file.  
echo = write to README.md file

Or easier this way

// It'll take you to gihub respo where files are

**git remote add origin** [**https://github.com/keno1se/harmonicascaler.git**](https://github.com/keno1se/demo9.git)

//this base-href comes into 404.html

**ng build --prod --base-href** [**https://keno1se.github.io/harmonicascaler/**](https://keno1se.github.io/demo9/)

//ngh = send to git  
 //dry run

**ngh --dir dist/harmonicascaler --dry-run**

//dry run ok? then send to git

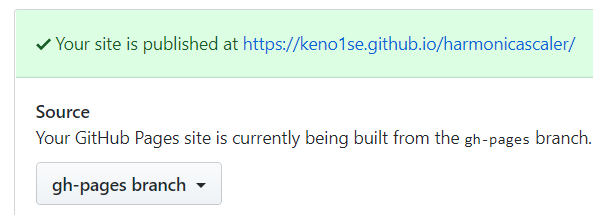
**ngh --dir dist/harmonicascaler**

\*\*\* Successfully published!

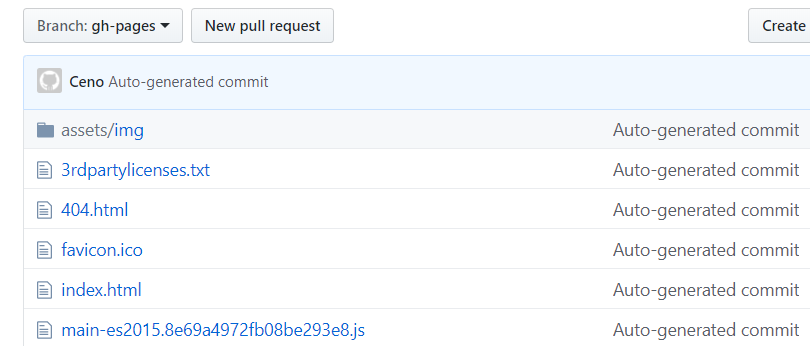
Now type this in any browser and see you site

[**https://keno1se.github.io/harmonicascaler/**](https://keno1se.github.io/demo9/)

**But first make sure that you run files from gh-pages, see below**

****

**You cans have a look at the commited files if you make your branch gh-pages.**

****

**Now remote git hup rendering is working. From no on you can do a ngh command instead of delete, upload files to master all the time. You now got version control!**

**echo "# AAA" >> README.md**  
above will write this to the readme.md file.

�#� �A�A�A�

This is what a dry-run will output

