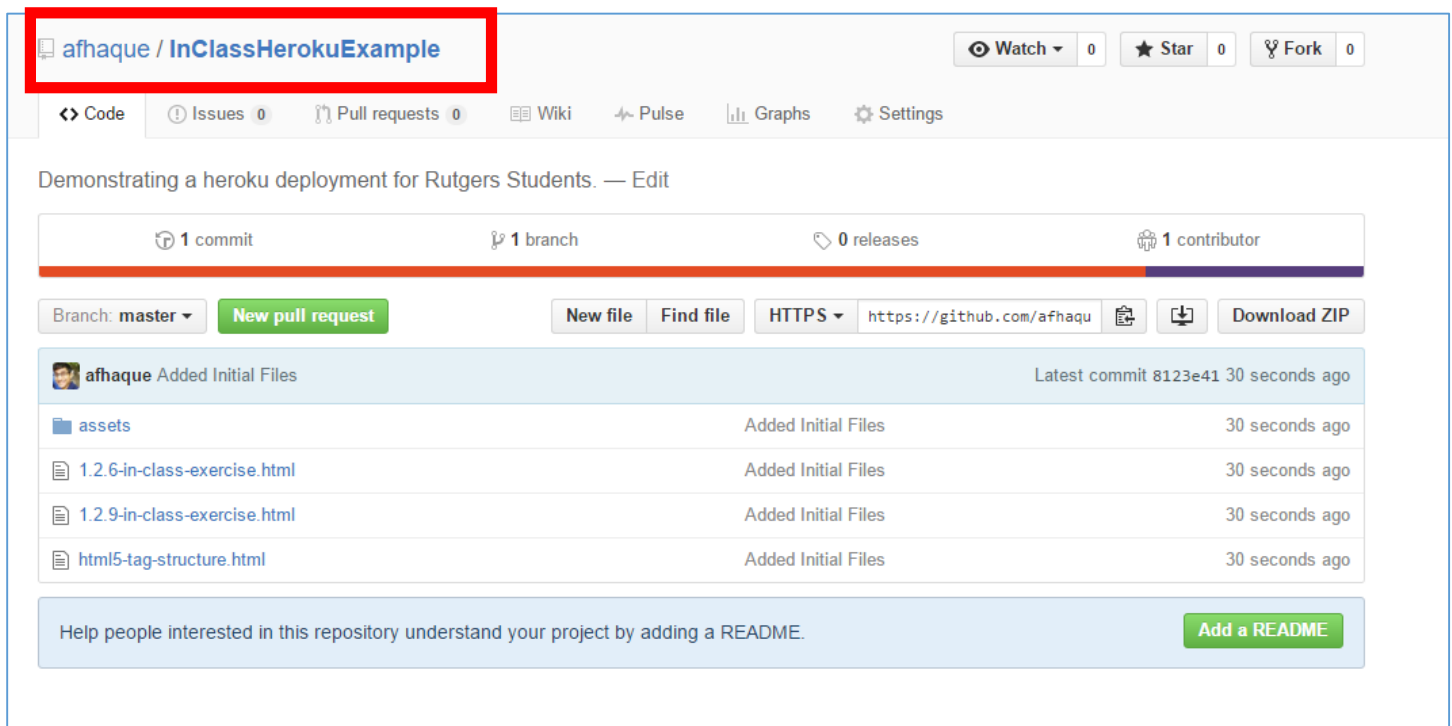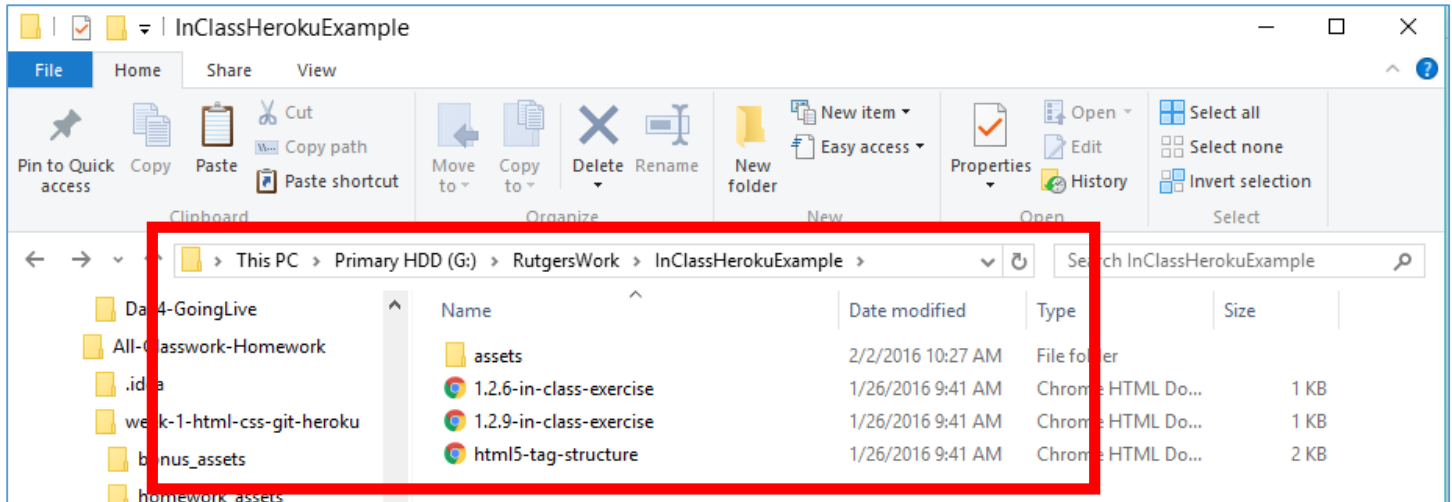# Deploying Your Static Website to Heroku

1. Choose a local folder of code that is tied to GitHub. (If you do not have one available, clone a repository you have in GitHub.)

You can see that in my example, I'm working with a folder called InClassHerokuExample. I have this folder both locally (on my machine) and in GitHub.

2. Create a file using sublime called composer.json. Inside of this file include just an empty set of curly brackets {}. Save this file inside of your local Git/Heroku folder.
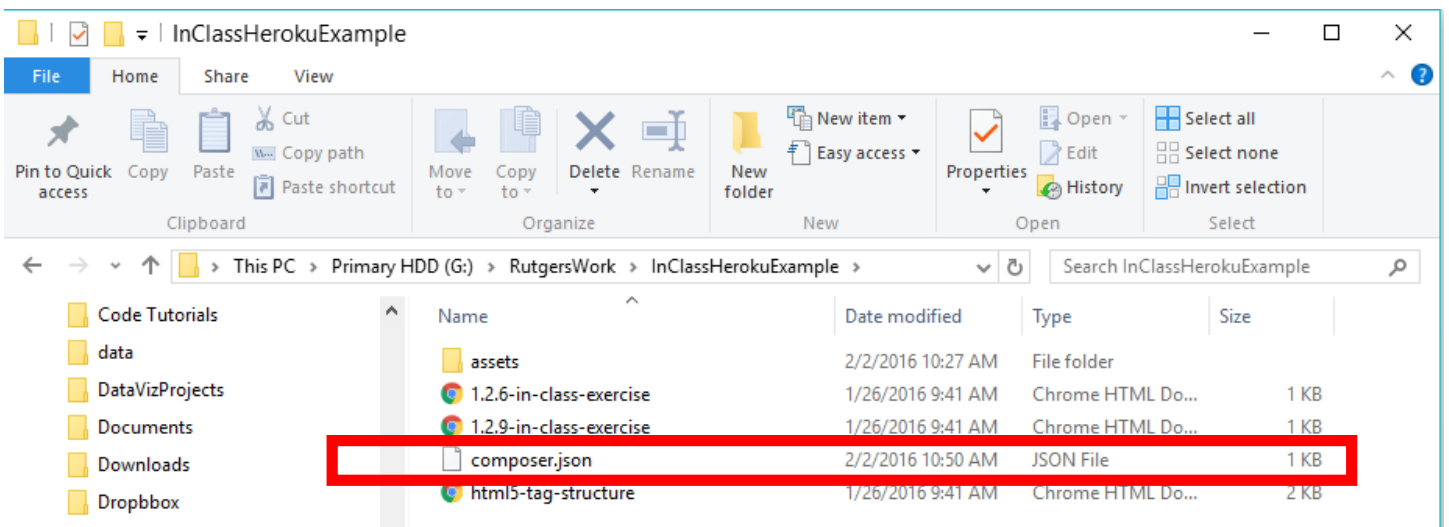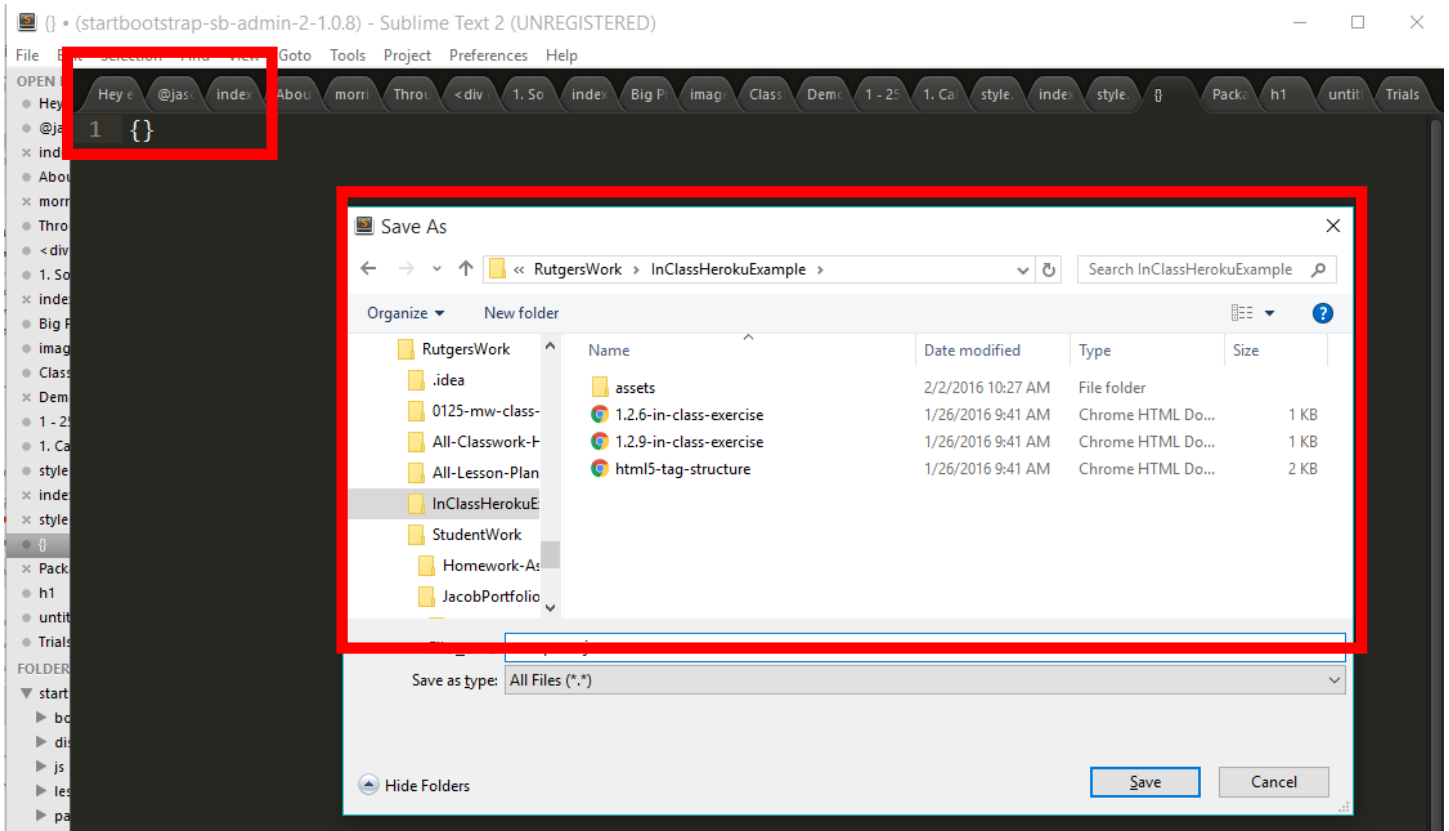
3. Create a file (again in Sublime) called index.php. Inside of this file include the below line.
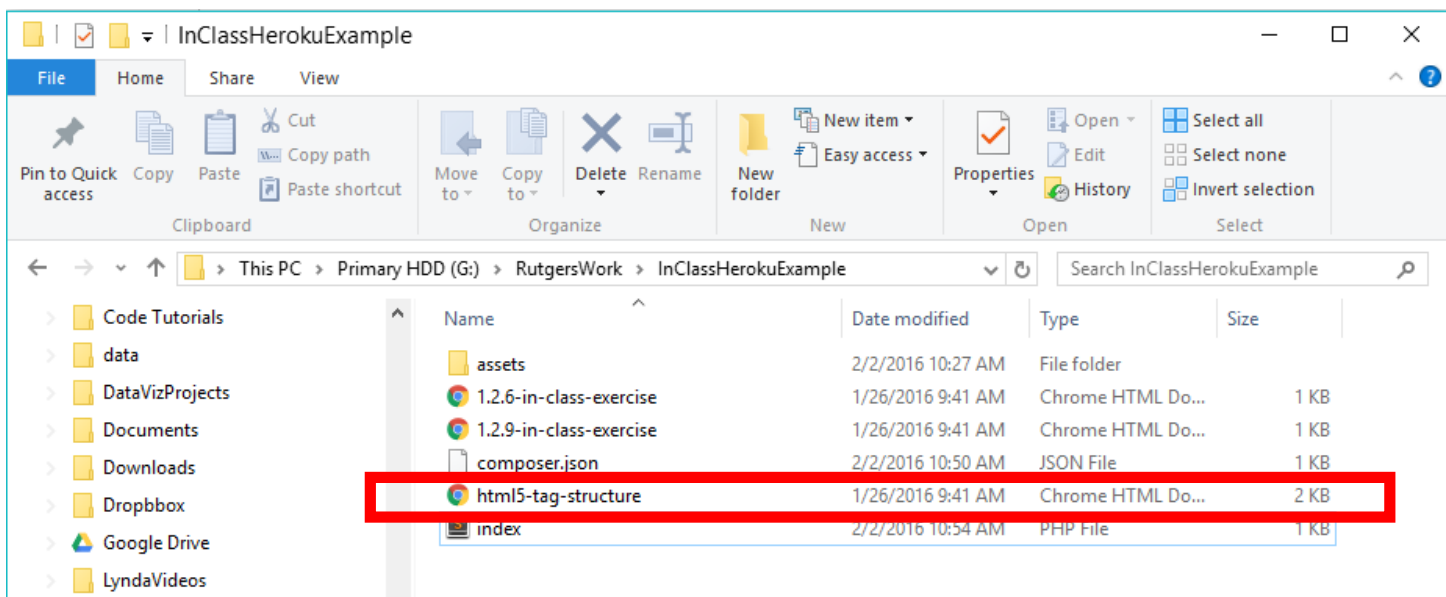
```
<?php include_once("filename of your html file with the extension"); ?>
```

Be sure to replace the filename with the full HTML filename you would like to display. If your html file is inside of a folder remember to use correct relative pathways.

In my case, my index.php file looks like the below.

Save this file again inside of local code folder.

4. If you are a windows user open the cmd.exe (NOT GIT BASH) and type heroku login. Keep this command prompt open in the background.

If you are a mac user stay in terminal and type the command heroku login. Enter your Heroku credentials and proceed with all the below steps in terminal.

5. If you are a Windows User, open Git Bash and navigate to your folder with the code.

```
Ahmed@oatmealcentral MINGW64 /g/rutgerswork
$ cd InClassHerokuExample/

Ahmed@oatmealcentral MINGW64 /g/rutgerswork/InClassHerokuExample (master)
$ |
```

6. Next, run the command: "git remote –v"
   This is to show you that right now, your code is only attached to GitHub (and not Heroku).

```
Ahmed@oatmealcentral MINGW64 /g/rutgerswork/InClassHerokuExample (master)
$ git remote -v
origin  https://github.com/afhaque/InClassHerokuExample.git (fetch)
origin  https://github.com/afhaque/InClassHerokuExample.git (push)
```

7. Next run the command "heroku create". This will create an app instance on the Heroku server and will git connect your code to heroku.

```
Ahmed@oatmealcentral MINGW64 /g/rutgerswork/InClassHerokuExample (master)
$ heroku create
Updating Heroku v4 CLI to 4.27.16-cb94b94 (master)... done
Updating plugins... done
rebuilding plugins cache... done
heroku-cli: Installing core plugins... done
Creating app... done, stack is cedar-14
https://limitless-castle-80639.herokuapp.com/ | https://git.heroku.com/limitless
-castle-80639.git
```

8. Run git remote –v again to confirm that heroku is a git connection option. This time you should see the heroku pathways.

```
$ git remote -v
heroku  https://git.heroku.com/limitless-castle-80639.git (fetch)
heroku  https://git.heroku.com/limitless-castle-80639.git (push)
origin  https://github.com/afhaque/InClassHerokuExample.git (fetch)
origin  https://github.com/afhaque/InClassHerokuExample.git (push)
```

9. Next run the command "git add –A". This is to add the composer and index.php file to your git repository.

```
Ahmed@oatmealcentral MINGW64 /g/rutgerswork/InClassHerokuExample (master)
$ git add -A
```

10. Next run: git commit –m "COMMENT". This is to save the changes you made in adding those last two files.

```
Ahmed@oatmealcentral MINGW64 /g/rutgerswork/InClassHerokuExample (master)
$ git commit -m "Added index and composer file"
[master f22bcff] Added index and composer file
 2 files changed, 2 insertions(+)
 create mode 100644 composer.json
 create mode 100644 index.php
```

11. Finally, run the command "git push heroku master". A series of processes will be initiated. Once the process is complete note the name of the app.

12. Once the processes are complete, login to your heroku account at www.heroku.com . You will see a list or a (single) app. Note the one that has the same funky name as you saw in bash. Click on it.
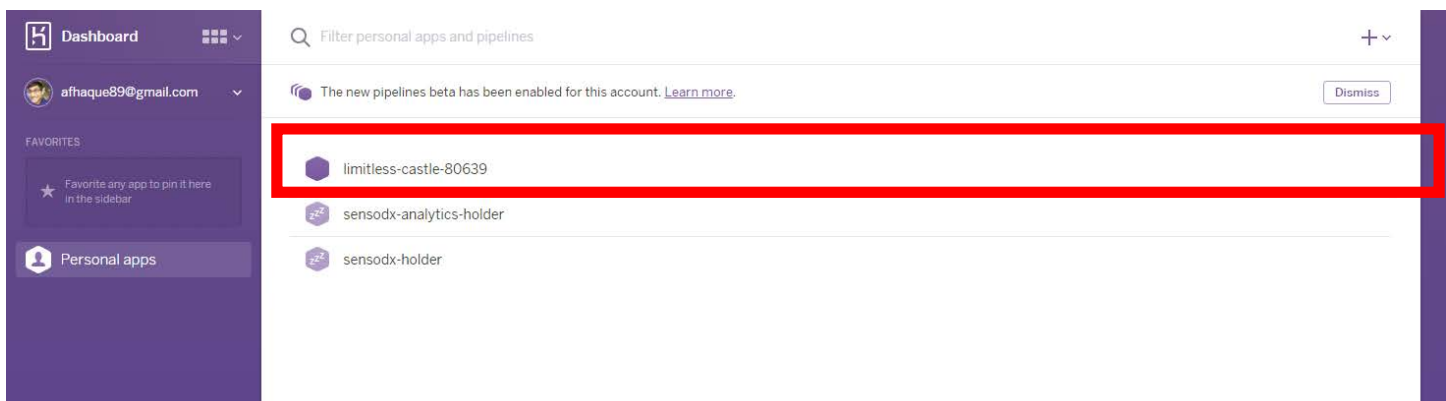
13. Next click on settings. Then scroll down until you see the part that says: "Domains". Note the URL listed under Heroku Domain.

14. Finally, go to the URL listed under the Heroku Domain in Chrome. If all went well – you should see your website!
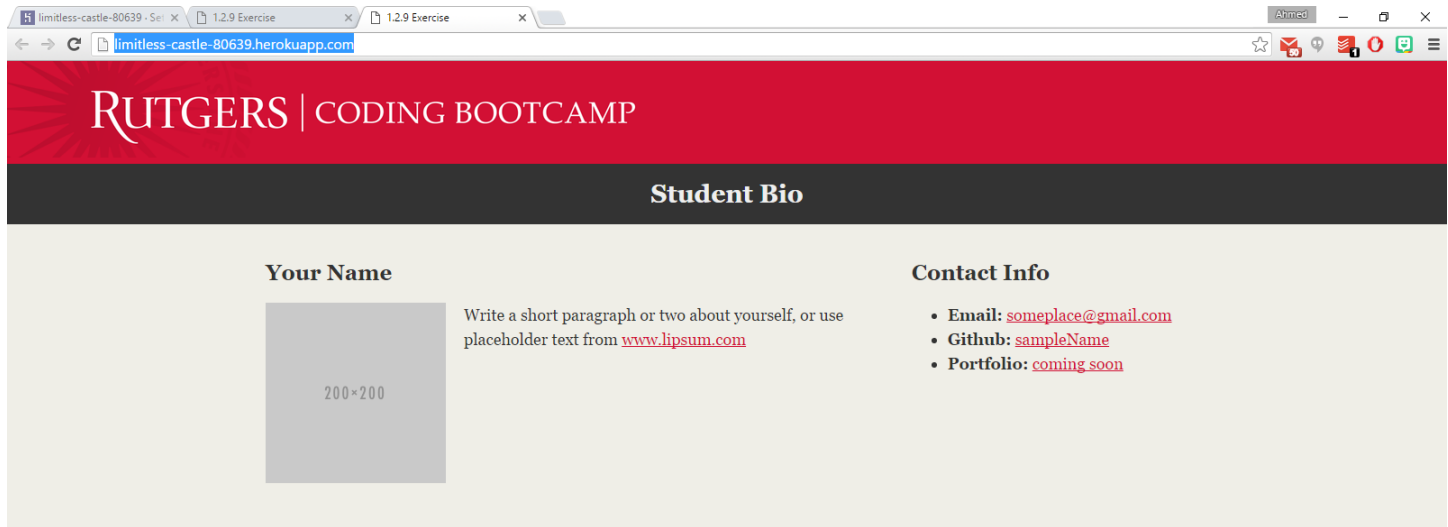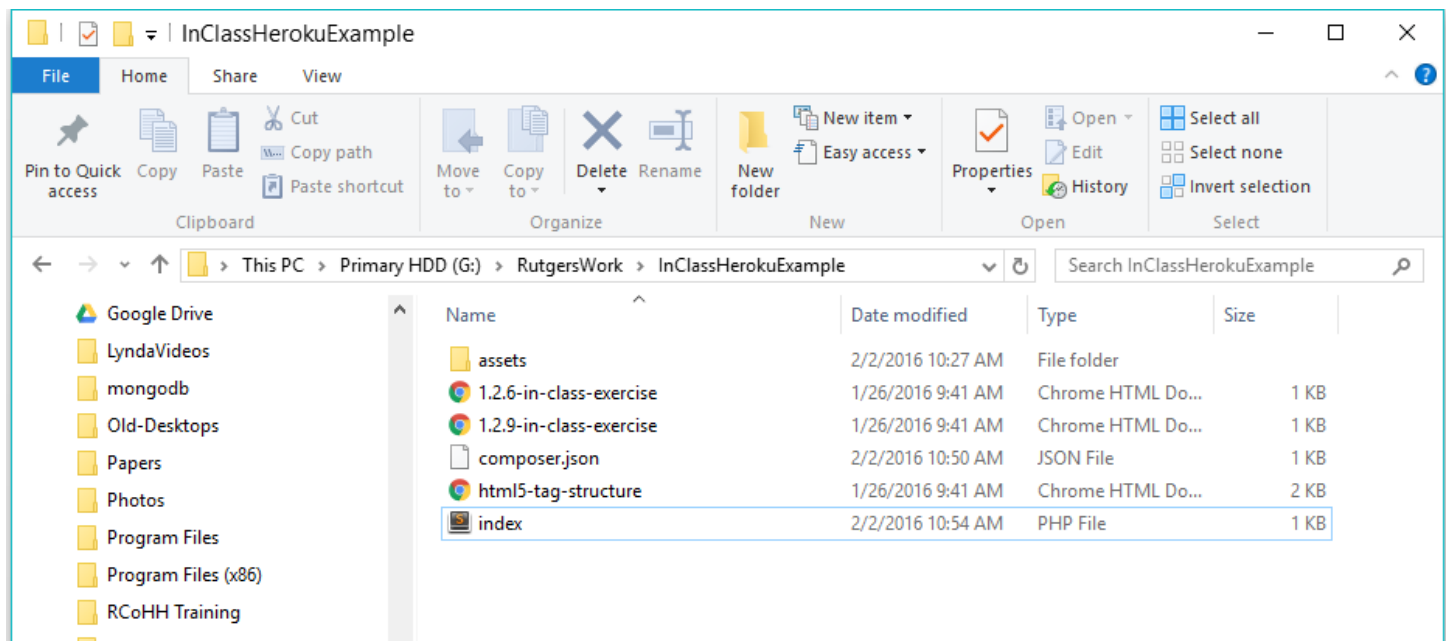

VICTORY!!!

Troubleshooting:

15. **The Case of the Blank Screen:**

Even after completing all of these steps, you may end up with a blank screen in chrome if you navigate to your website. As was discovered by the great Nate Hawley III, this may have to do with relative pathways. Try experimenting with code where your .html file isn't inside of a separate folder in your git folder.

See example below:

GOOD

BAD (WILL NOT WORK)