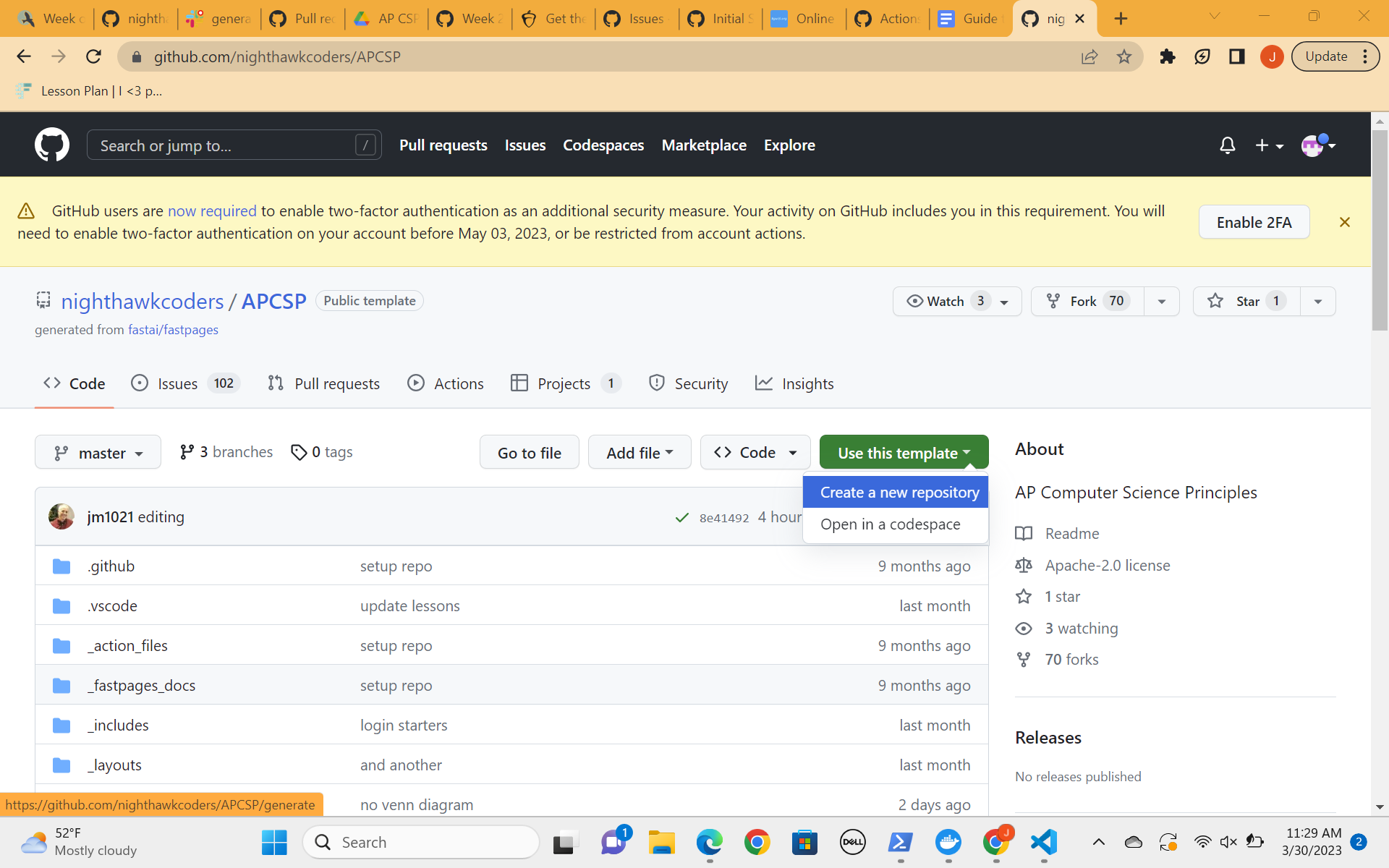
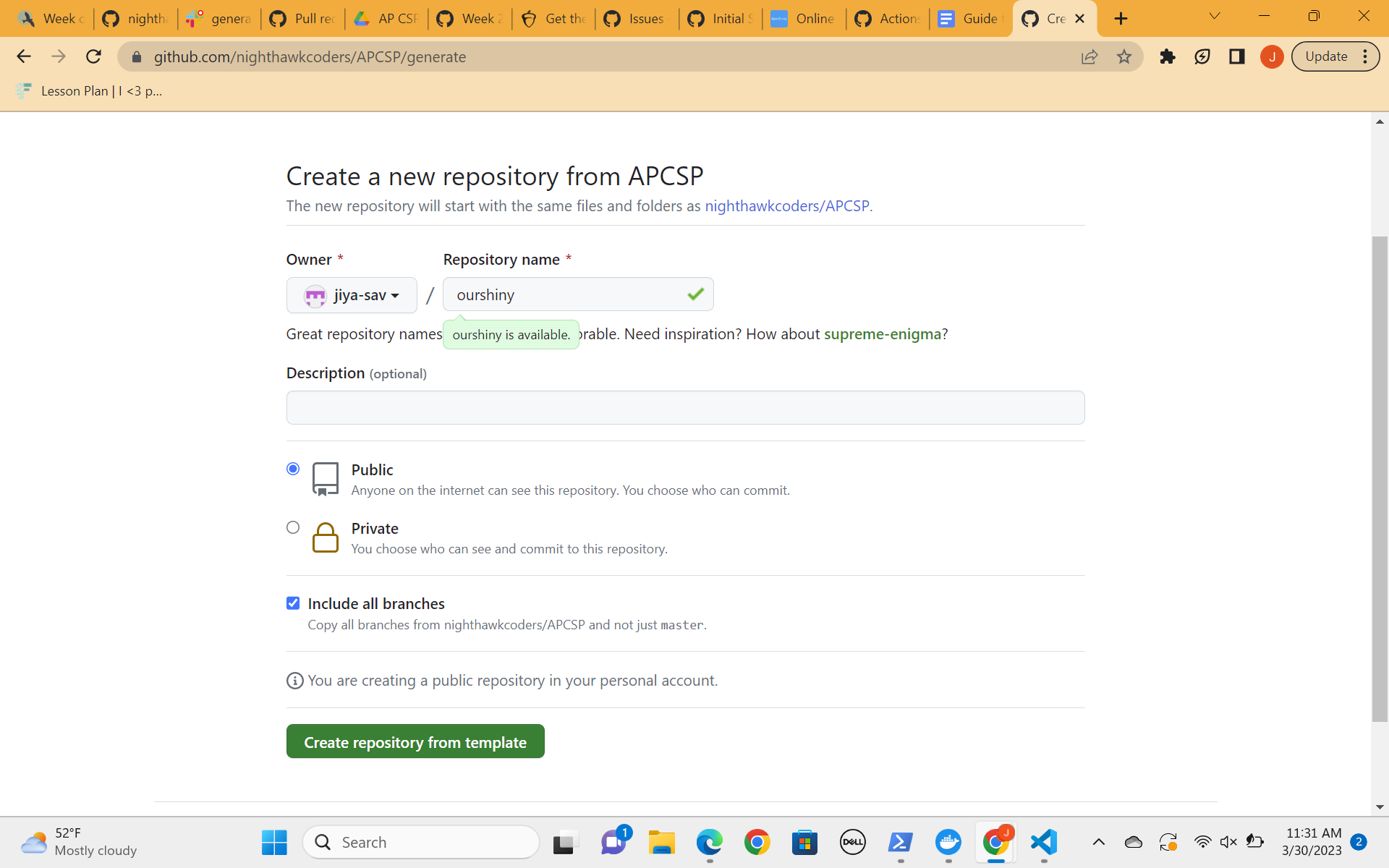
Guide to Making a New Repository

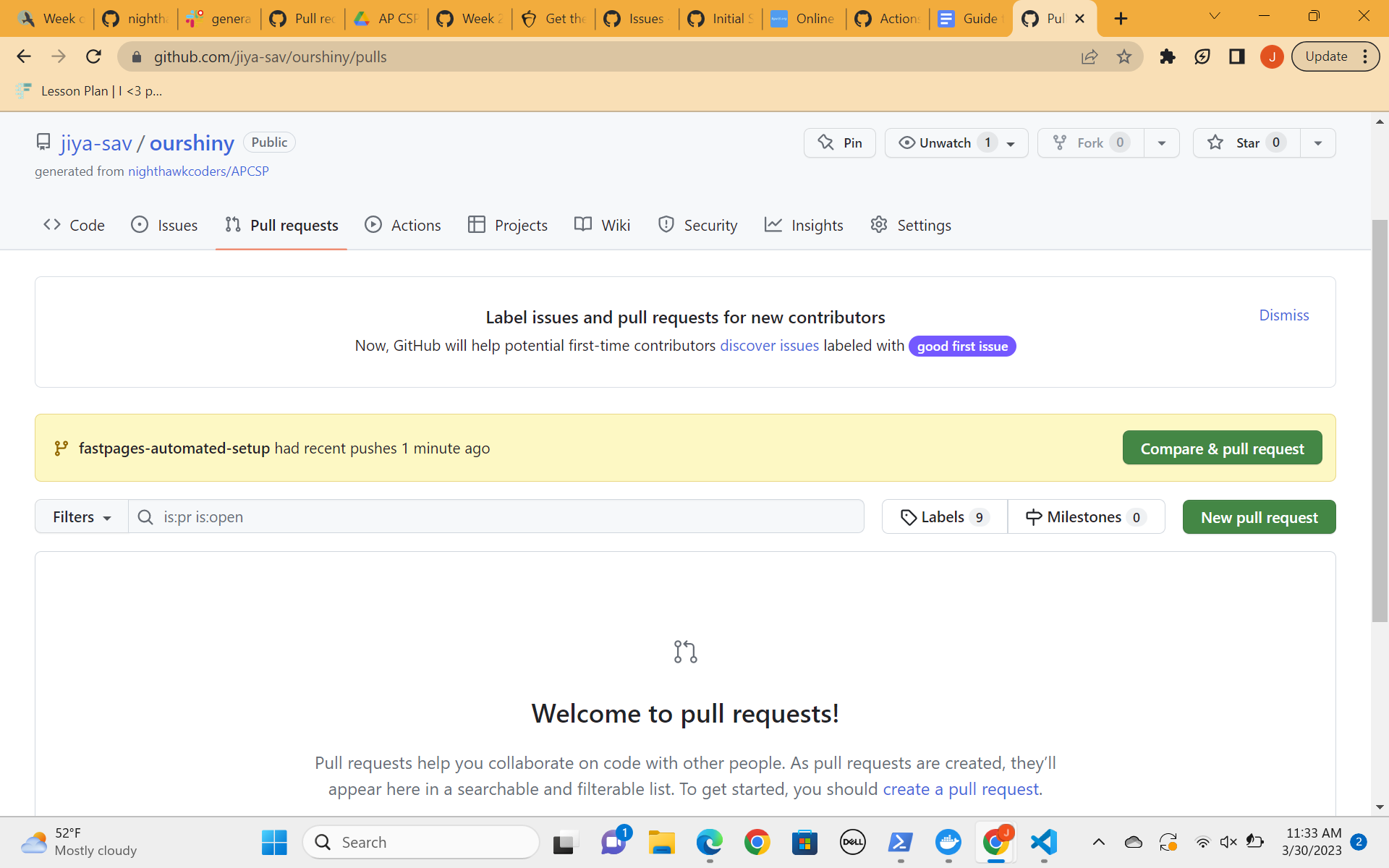
1. Duplicate APCSP, and use it as template for your own repo



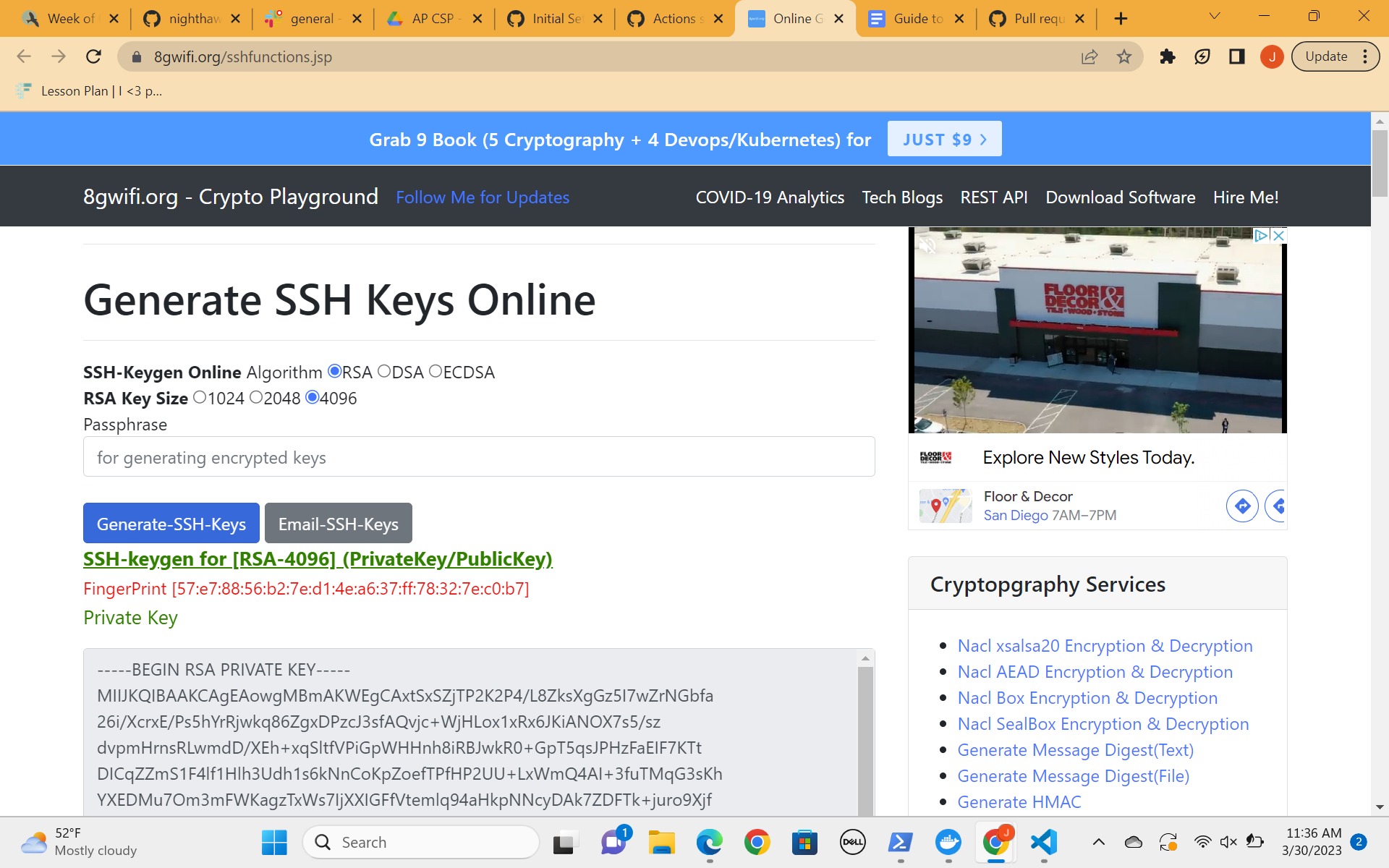
1. Name your repo and include all branches before clicking the green create button.



1. If you have an automatically generated pull request, then go ahead and follow the instructions on there. If there is no pull request, as shown below, then move on to step 4.

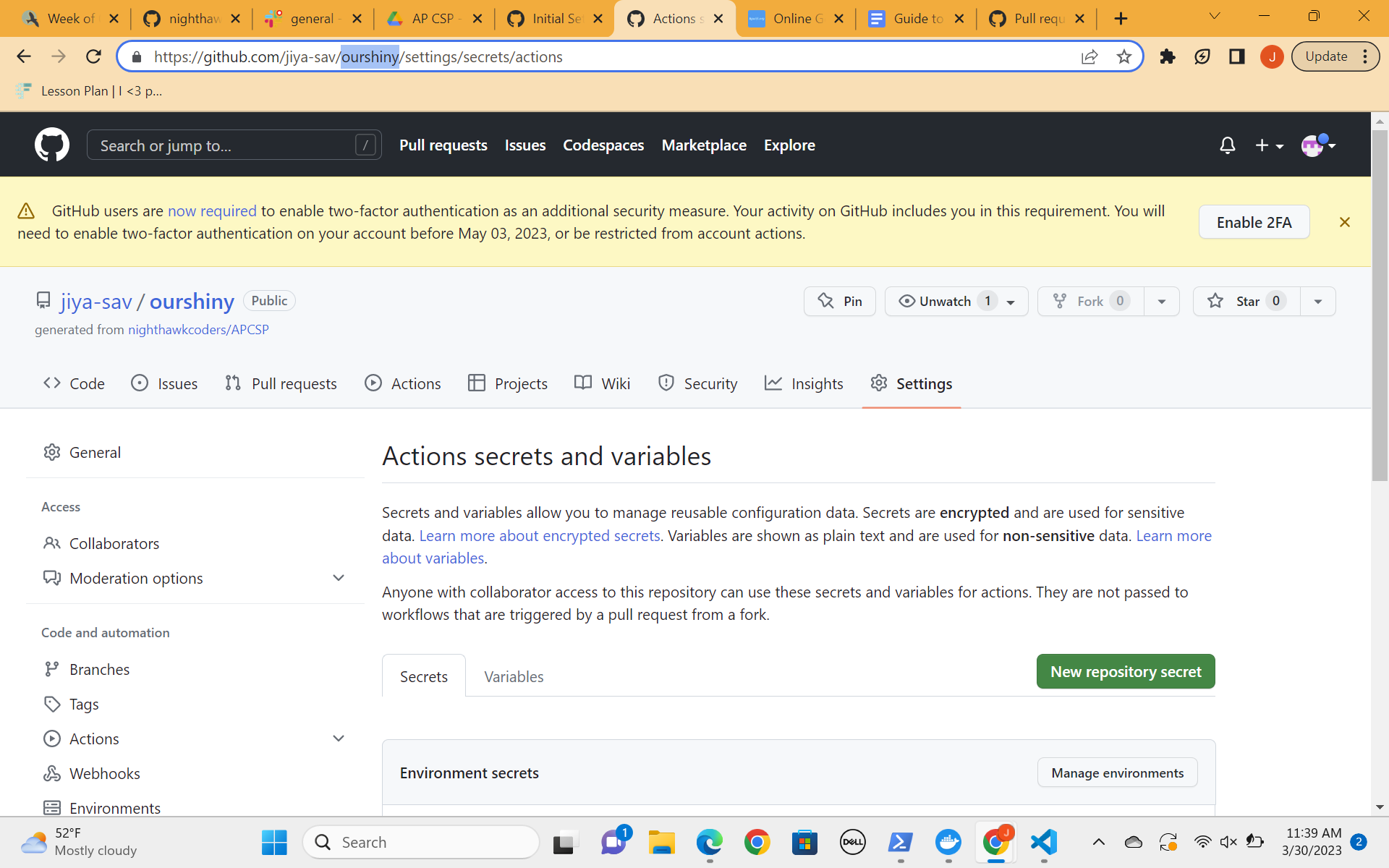


1. Navigate to [this link](https://8gwifi.org/sshfunctions.jsp). Select: RSA and 4096 and leave Passphrase blank. Click the blue button Generate-SSH-Keys. You have just created an ssh key-pair.

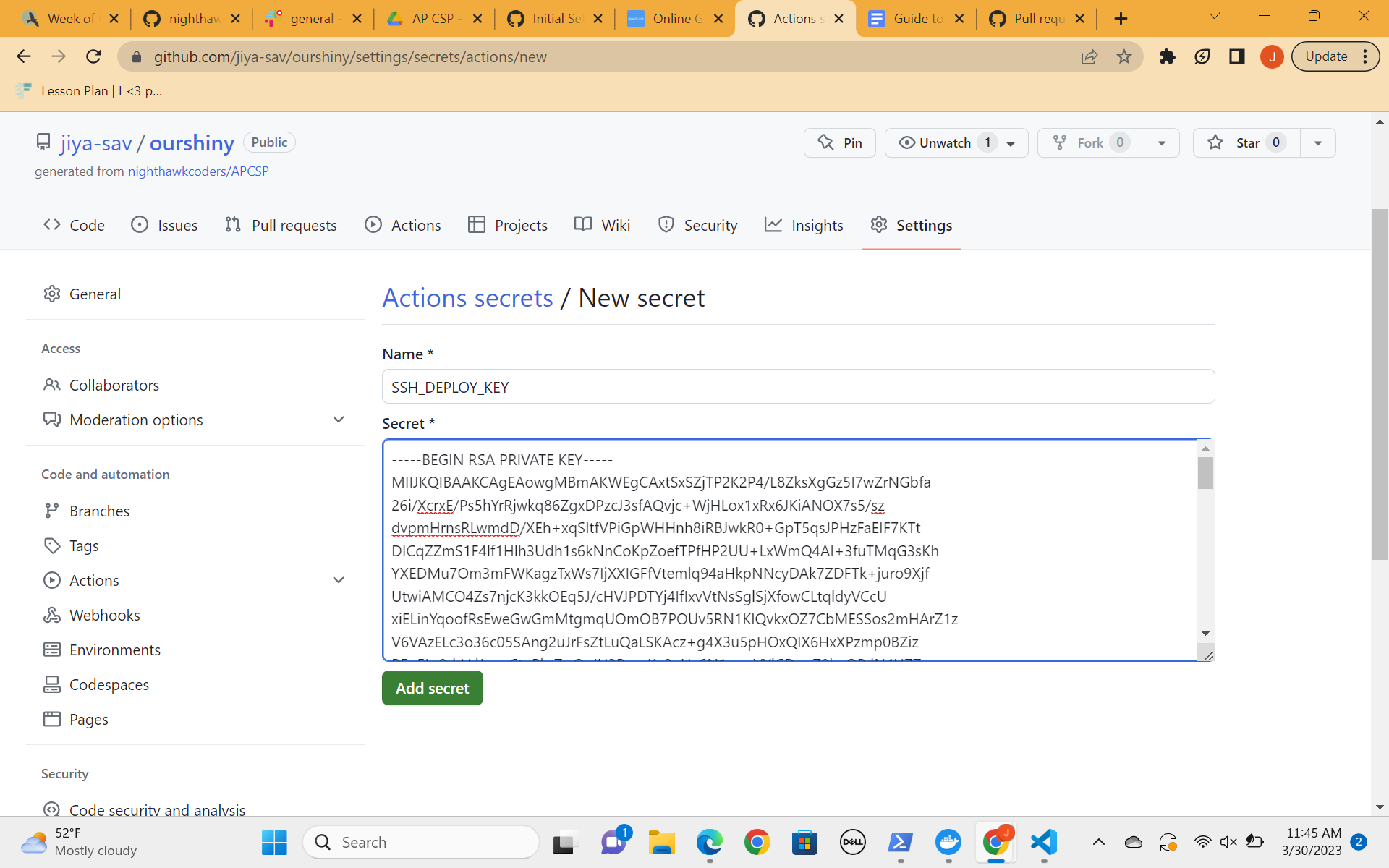


1. Navigate to [this link](https://github.com/jiya-sav/ourshiny/settings/secrets/actions). Replace the section of the link highlighted in blue with your repository name. In the picture, I have inputted “ourshiny,” because it is my repository name. Replace the section of the link highlighted in yellow with your repository name that you just made.

https://github.com/yourgithubname/yourrepositoryname/settings/secrets/actions

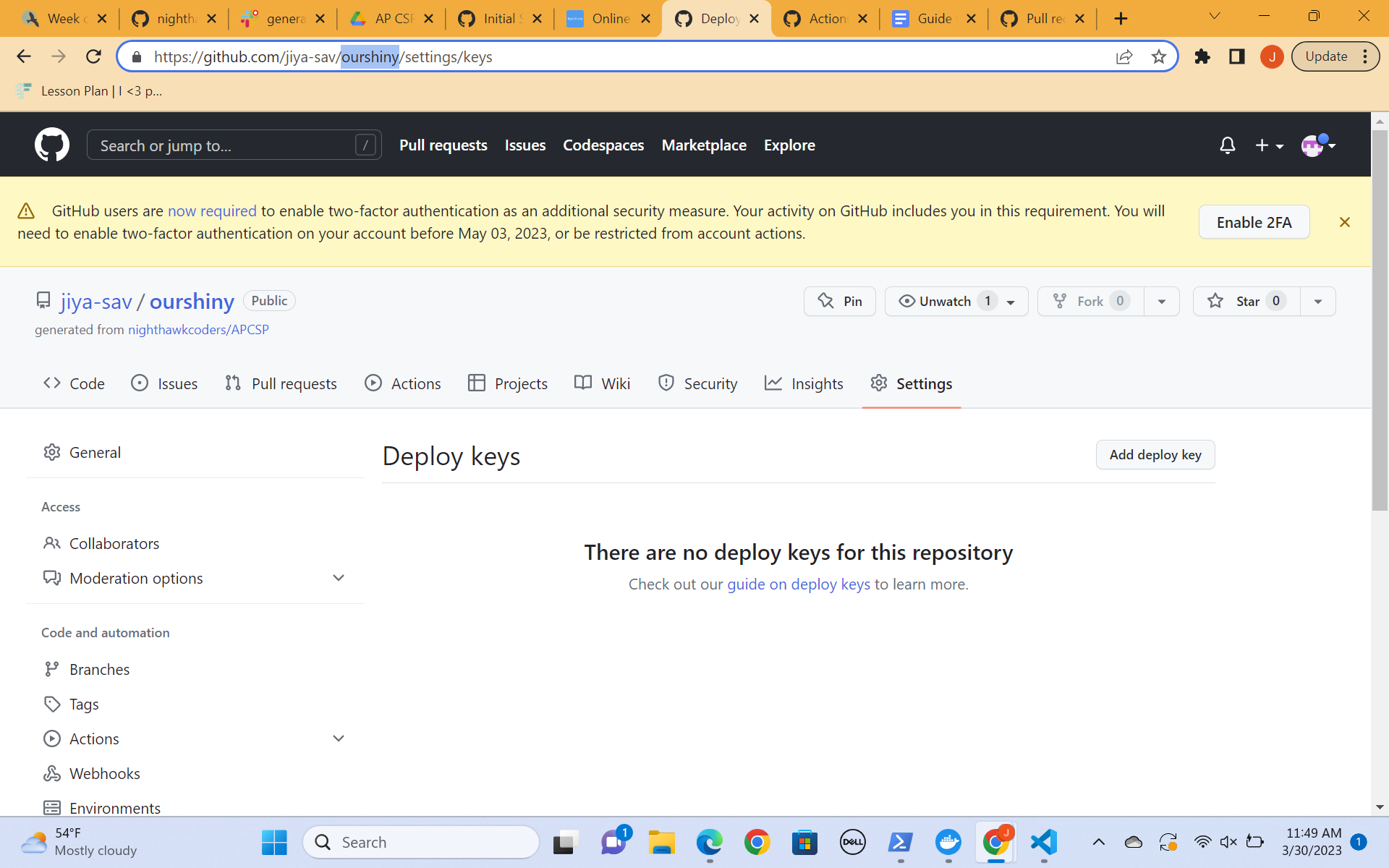


1. Click New repository secret. Copy and paste the Private Key (generated in step 4) into the Value field. This includes the "---BEGIN RSA PRIVATE KEY---" and "--END RSA PRIVATE KEY---" portions. In the Name field, name the secret SSH\_DEPLOY\_KEY. Once done, click the green Add Secret button.

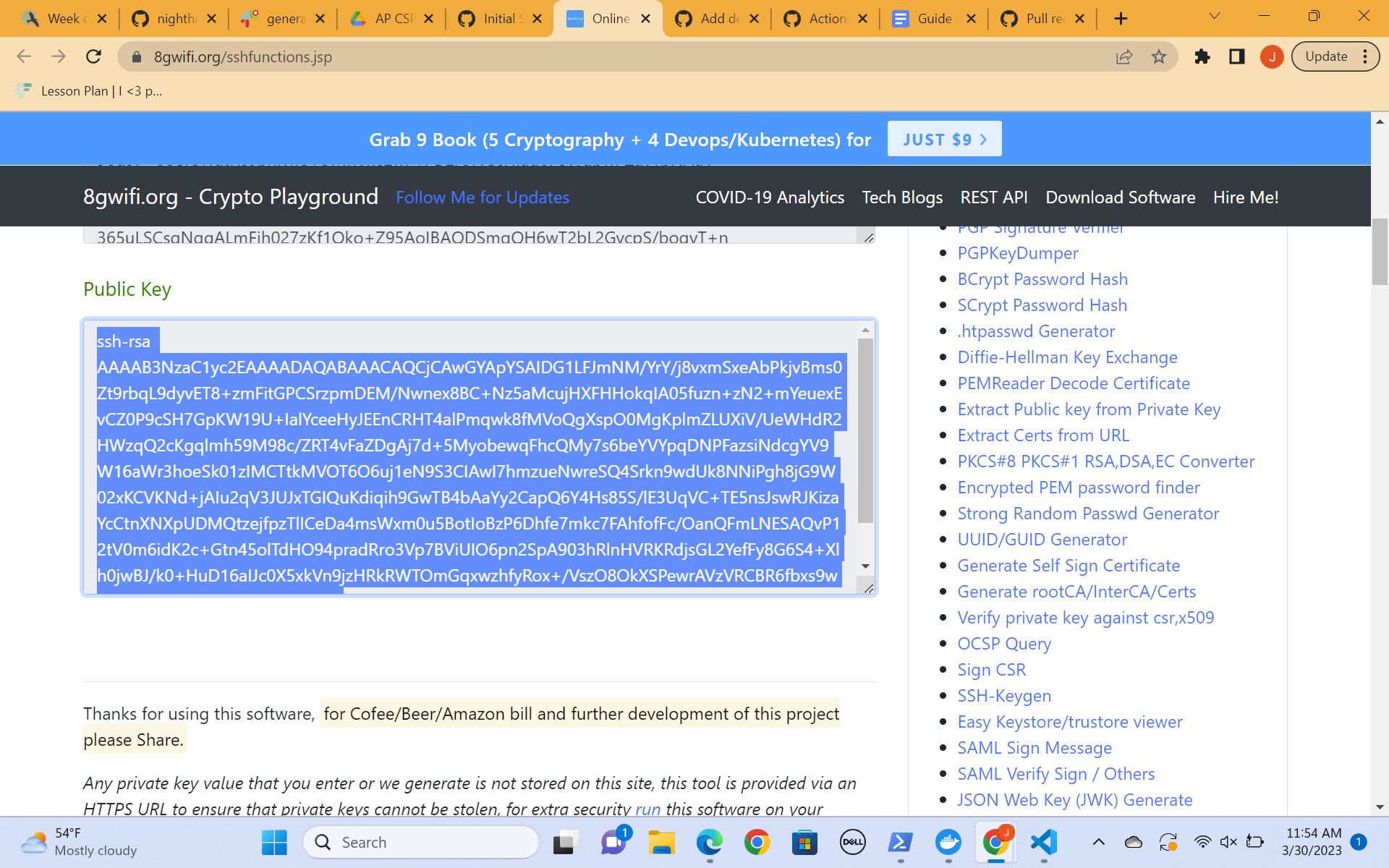


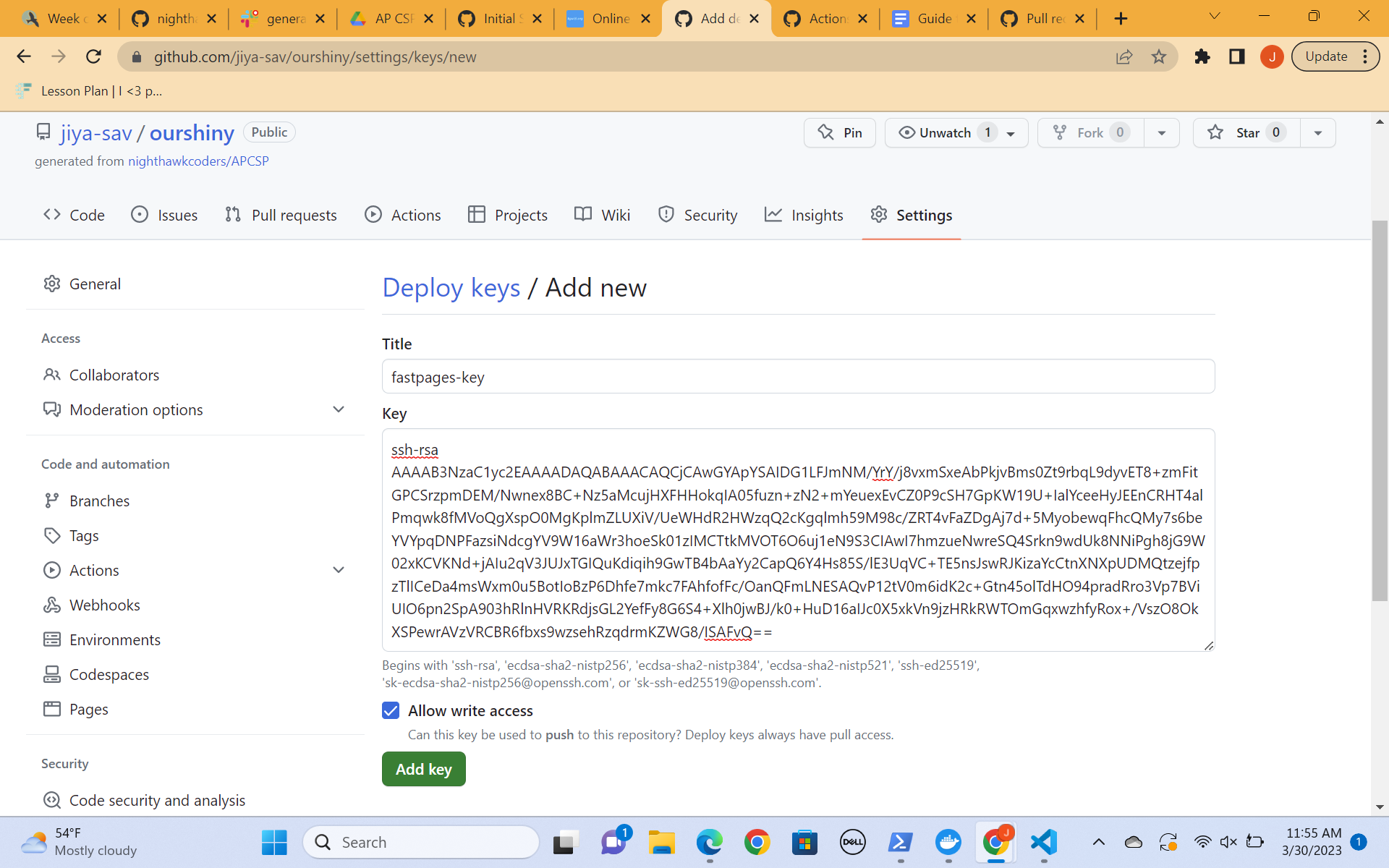
1. Navigate to [this link](https://github.com/jiya-sav/ourshiny/settings/keys), then replace the portions below with your github and repository names. My github name in the image below is jiya-sav, and my repository name is ourshiny.

https://github.com/yourgithubname/yourrepositoryname/settings/keys



1. Click the Add deploy key button. Paste your Public Key (generated in step 4) into the Key box. In the Title, name the key anything you want, for example fastpages-key. Finally, make sure you click the checkbox next to Allow write access (pictured below), and click the green Add key button to save the key.





1. GitHub Actions will build your site, which will take 2-3 minutes to complete. This will happen anytime you push changes to the master branch of your repository. You can monitor the logs of this if you like on the Actions tab of your repo.

Your GH-Pages Status badge on your README will eventually appear and be green, indicating your first successful build.

You can monitor the status of your site in the GitHub Pages section of your repository settings.

Now that you have made your repository secure, you can clone it into vscode and CODE, CODE, CODE!!!