

Create a tag in a GitHub repository

Asked 5 years, 11 months ago Active 4 months ago Viewed 488k times



717



I have a repository in GitHub and I need to **tag** it. I tagged in a shell, but on GitHub it is not showing up. Do I have to do anything else?

The command I used in the shell is:

```
git tag 2.0
```



193

And now when I type `git tag` it shows:

```
2.0
```

So it seems like tags are present, correct?

The repository is: <https://github.com/keevitaja/myseo-pyrocms>.

How do I make this tag show up on GitHub? Where are my tags?

[git](#)[github](#)[git-tag](#)

edited Jan 15 at 19:32



Peter Mortensen

14.3k 19 88 116

asked Aug 13 '13 at 18:56



Tanel Tammik

3,939 3 14 22

- 1 You need to push the tag to the repository. Using git tags is same as git branch, it means you have to create/add a tag, push the tag to the repository so that everyone can fetch it. For more basic to advanced, here is the reference [abhisheksharma.online/git-tags](#) – [Abhishek Sharma](#) May 18 at 17:30

7 Answers



You can create tags for GitHub by either using:

1175

- the Git command line, or
- GitHub's web interface.



Creating tags from the command line

To create a tag on your current branch, run this:

```
git tag <tagname>
```

If you want to include a description with your tag, add `-a` to create an [annotated tag](#):

```
git tag <tagname> -a
```

This will create a `local` tag with the current state of the branch you are on. When pushing to your remote repo, tags are NOT included by default. You will need to explicitly say that you want to push your tags to your remote repo:

```
git push origin --tags
```

From the [official Linux Kernel Git documentation for `git push`](#) :

```
--tags
```

All refs under `refs/tags` are pushed, in addition to refsspecs explicitly listed on the command line.

Or if you just want to push a single tag:

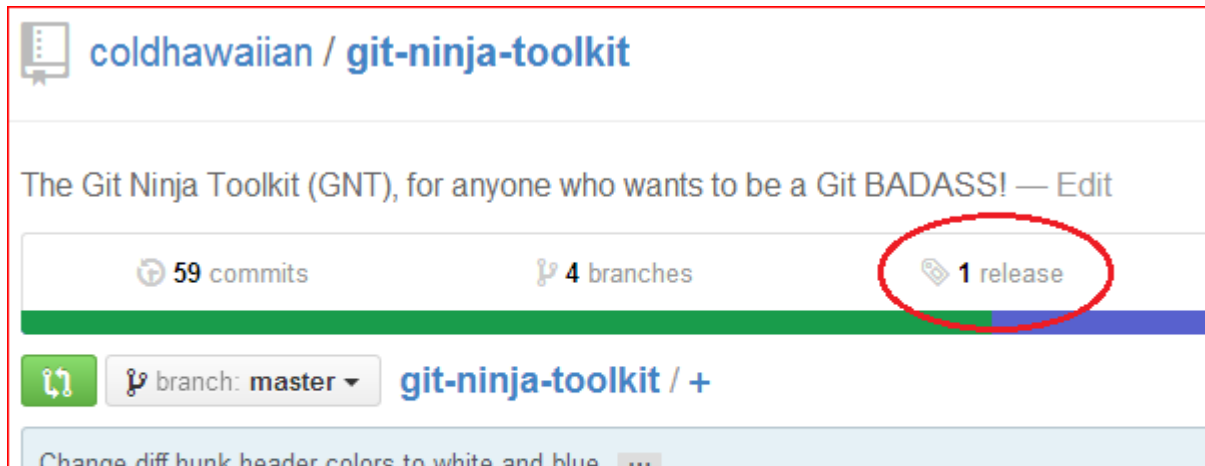
```
git push origin <tag>
```

See also my answer to [How to push a tag to a remote repository using Git?](#) for more details about that syntax above.

Creating tags through GitHub's web interface

You can find GitHub's instructions for this at their [Creating Releases help page](#). Here is a summary:

1. Click the **releases** link on our repository page,




2. Click on **Create a new release** or **Draft a new release**,



3. Fill out the form fields, then click **Publish release** at the bottom,

Tag version

@


 Target: **master** ▼


Excellent! This tag will be created from the target when you publish this release.

Release title

Write

Preview

 **Parsed as Markdown**

 **Edit in fullscreen**

Describe this release

Attach images by dragging & dropping, [selecting them](#), or pasting from the clipboard.

☐ **This is a pre-release**
We'll point out that this release is identified as non-production ready.

Publish release

Save draft

4. After you create your tag on GitHub, you might want to fetch it into your local repository too:

```
git fetch
```

edited Mar 14 '18 at 16:36




[Stevoisiak](#)

7,150 11 53 111

answered Aug 14 '13 at 4:42

[user456814](#)

- 4 @RandomDSdevel in github, release is just a tag. You can create tag from command line and push it to github remote. tag will appear as a release on githubs webpage. – [luri G.](#) Feb 16 '17 at 23:23
-
- 4 @luriG.: Well, yes, but shouldn't you be able to create tags that don't automatically get picked up by GitHub as releases...? (Back when I originally created the comment to which you responded and to whose response of yours *this* one is a reply, tags at least *used* to get turned into releases automatically, though I don't know if this is still the case or not going by the workflow explained in your answer hints at this no longer being true. I submitted a support request to GitHub about this a while ago, but they had a bit of a support backlog back then, so my request was closed.) – [RandomDSdevel](#) Feb 20 '17 at 2:38 
-
- Fwiw, for the Github web interface path, I can not see a Create or Draft new release button after clicking the releases link in Step 1. – [Snowcrash](#) Jun 2 '17 at 9:21
-
- @Snowcrash Do you have write access to the repository? – [forresthopkinsa](#) Aug 1 '17 at 15:38
-
- 2 via Github interface you can create only Light-weight tags. – [Anatolii Stepaniuk](#) Dec 5 '18 at 19:42
-



You just have to push the tag after you run the `git tag 2.0` command.

59

So just do `git push --tags now`.



answered Jul 4 '16 at 17:56



[kjdion84](#)

3,980

1

27

40

29 I'd rather suggest to use `git push v2.0` instead of `--tags`, there might be other tags that should not be pushed. – [tschoppi](#) May 15 '17 at 13:26

1 That's only a problem if your local tags differ from your remote tags. They are easy to sync if that ever becomes a problem. – [kjdion84](#) Sep 14 '17 at 10:48

@kjdion84 No, that's also a problem if there are obsolete tags: All tags get always fetched and when someone uses `--tags`, it's hard to get rid of them. – [maartinus](#) Oct 12 '17 at 16:05

1 or `git push origin v2.0` – [prayagupd](#) Nov 15 '17 at 1:33



Creating Tags

57

Git uses two main types of tags: *lightweight* and *annotated*.



Annotated Tags:

To create an annotated tag in Git you can just run the following simple commands on your terminal.

```
$ git tag -a v2.1.0 -m "xyz feature is released in this tag."  
$ git tag  
v1.0.0  
v2.0.0  
v2.1.0
```

The -m denotes message for that particular tag. We can write summary of features which is going to tag here.

Lightweight Tags:

The other way to tag commits is lightweight tag. We can do it in the following way:

```
$ git tag v2.1.0  
$ git tag  
v1.0.0  
v2.0.0  
v2.1.0
```

Push Tag

To push particular tag you can use below command:

```
git push origin v1.0.3
```

Or if you want to push all tags then use the below command:

```
git push --tags
```

List all tags:

To list all tags, use the following command.

```
git tag
```

edited Jan 15 at 19:34



Peter Mortensen

answered Jan 31 '18 at 5:19



Lawakush Kurmi



14.3k

19

88

116



1,321

11

25

Do you add the tag BEFORE or after the `git commit` ? – [garfbradaz](#) Nov 4 '18 at 18:28

I have added tag after git commit. – [Lawakush Kurmi](#) Nov 6 '18 at 5:15



4



CAREFUL: In the command in [Lawakush Kurmi's answer](#) (`git tag -a v1.0`) the `-a` flag is used. This flag tells Git to create an annotated tag. If you don't provide the flag (i.e. `git tag v1.0`) then it'll create what's called a lightweight tag.

Annotated tags are recommended, because they include a lot of extra information such as:

- the person who made the tag
- the date the tag was made
- a message for the tag

Because of this, you should always use annotated tags.

edited Jan 15 at 19:41

[Peter Mortensen](#)

14.3k

19

88

116

answered May 10 '18 at 8:21

[silva blaze](#)

611

4

7



3



It all depends what type of tag you want to create:

- If you want to create Annotated tags, to show extra metadata, you can do it in the following way: `git tag -a v1.0.0` .
- On the other hand, Lightweight tags are used to "bookmark" your commits for private use: `git tag v1.0.0` .

There are a few other tag functionalities such as:

- Listing tags - `git tag -l -n3` . The command lists all existing tags with maximum 3 lines of their tag message. By default `-n` only shows the first line.
- Tag details - `git show <tag_identifier>` . It shows all you need to know about a specific tag.
- Sorting tags - `git tag --sort=<type>`

- Publishing tags - `git push origin v1.0` . You can git push the tag individually, or you can run `git push --tags` which will push all tags at once.

Be sure to check this [tag related article](#) for more relevant information.

answered Jun 14 '18 at 7:54



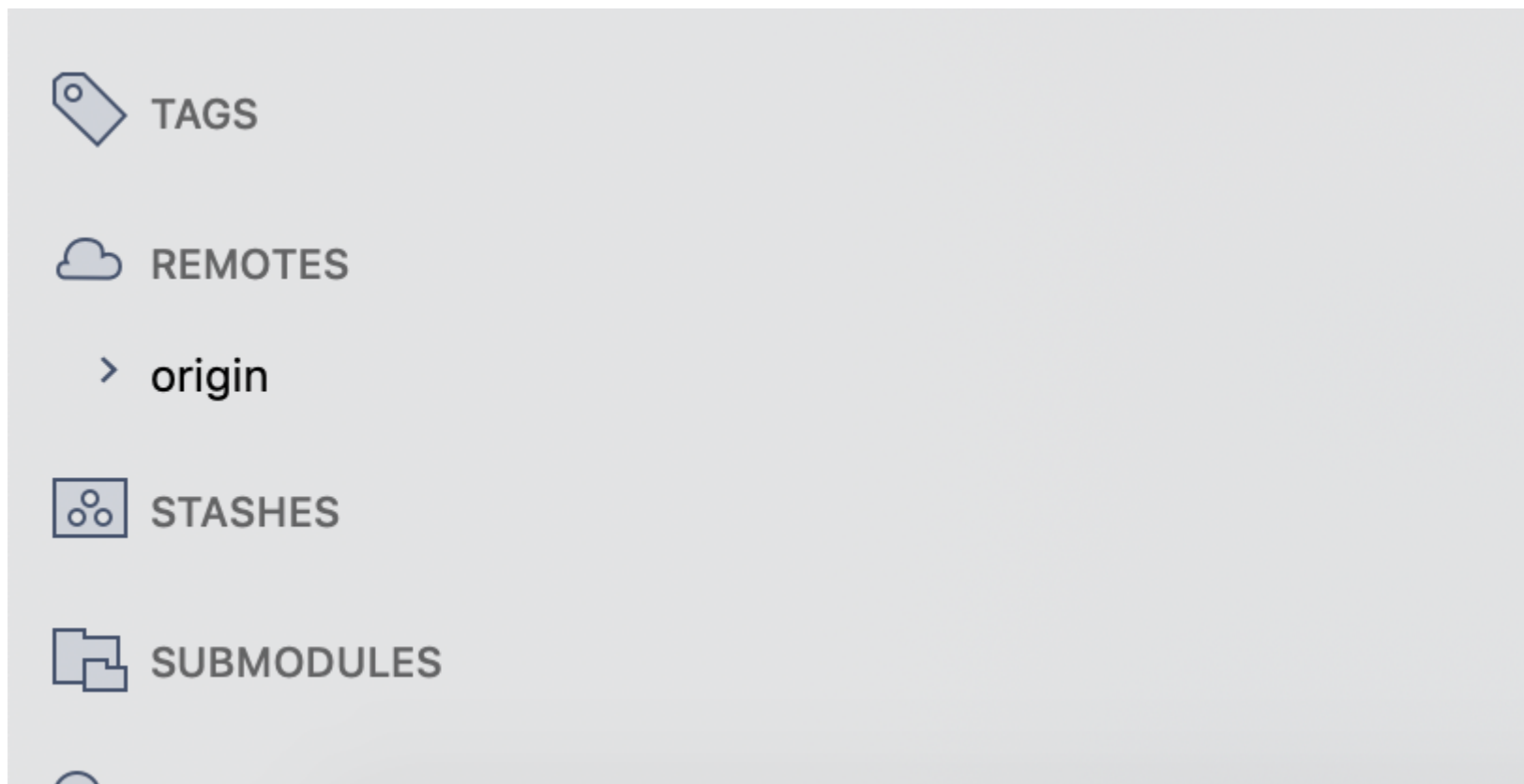
Nesha Zoric

2,251 20 26

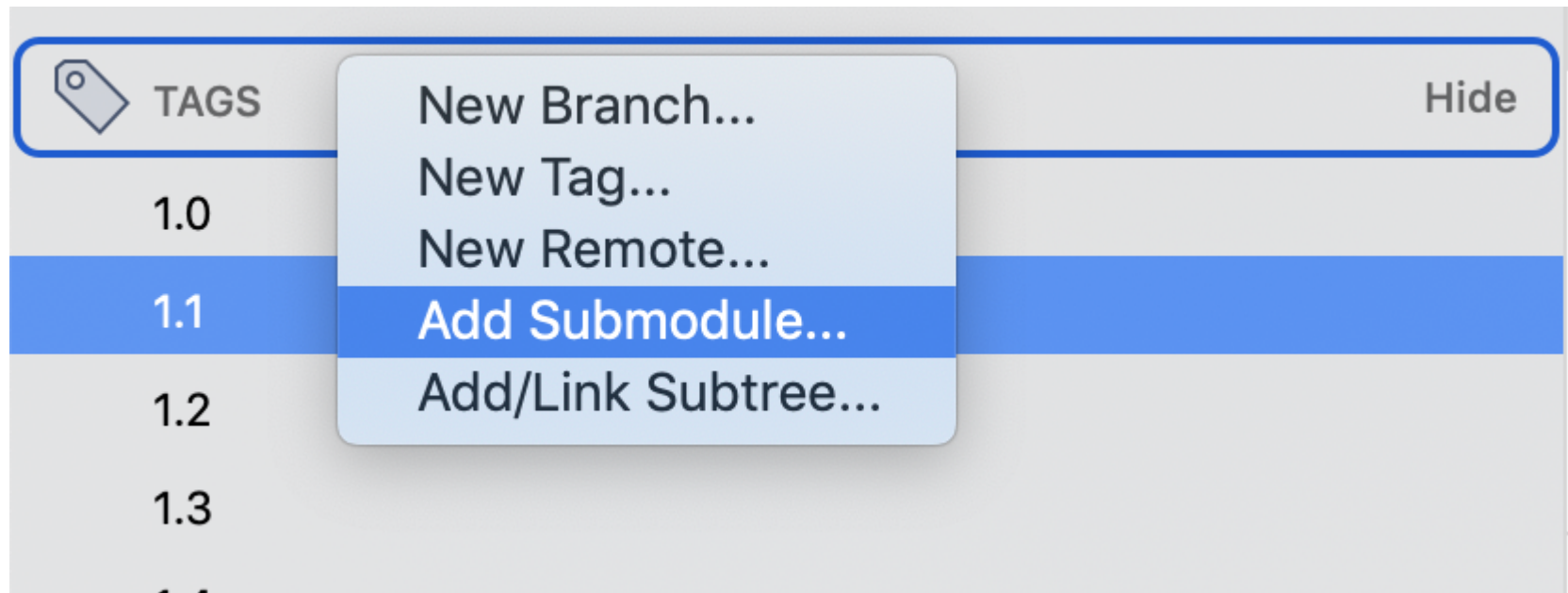
Using [Sourcetree](#)

3 Here are the simple steps to create a GitHub **Tag**, when you release build from master.

1. Open source_tree tab



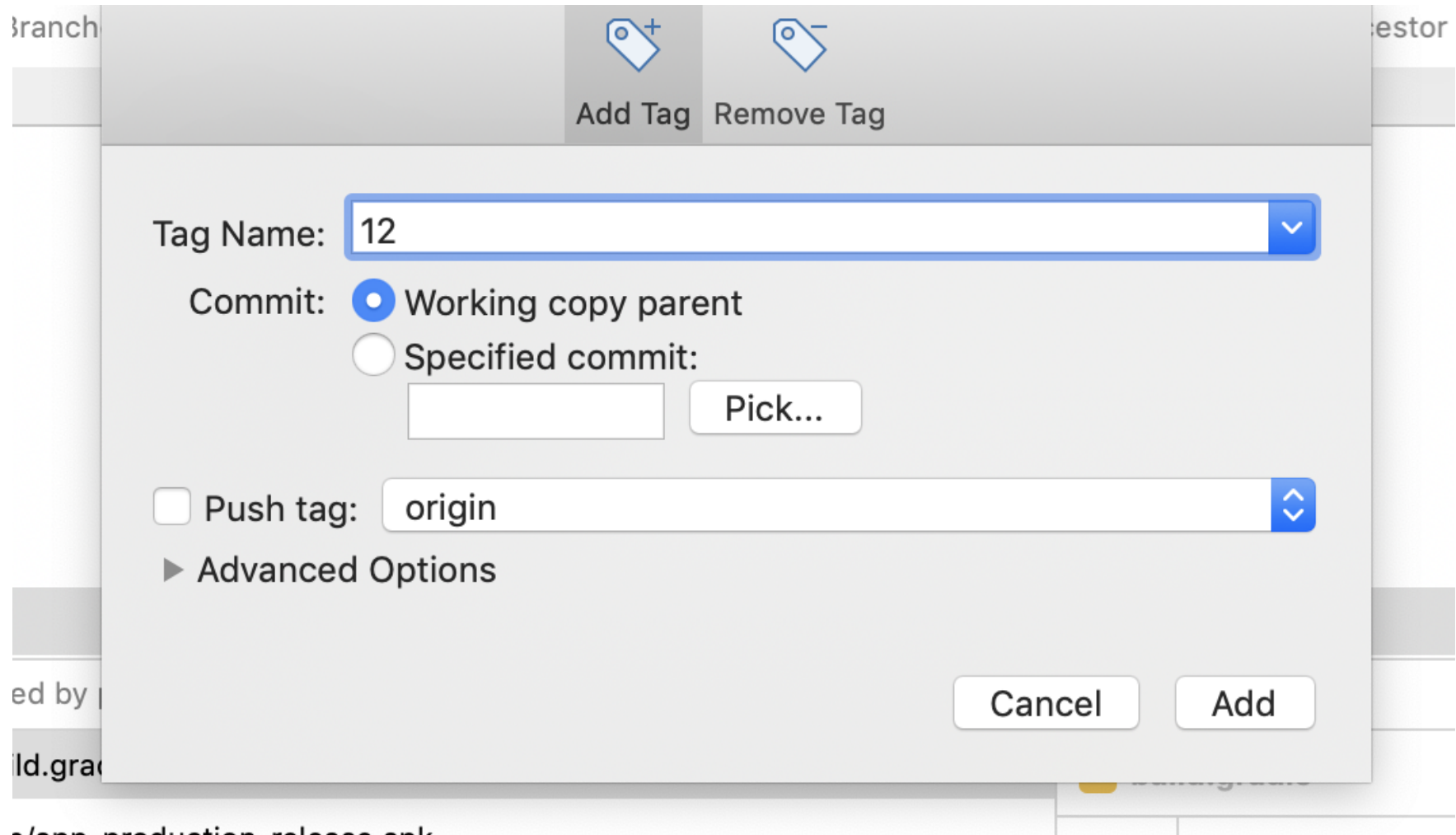
2. Right click on Tag sections from Tag which appear on left navigation section



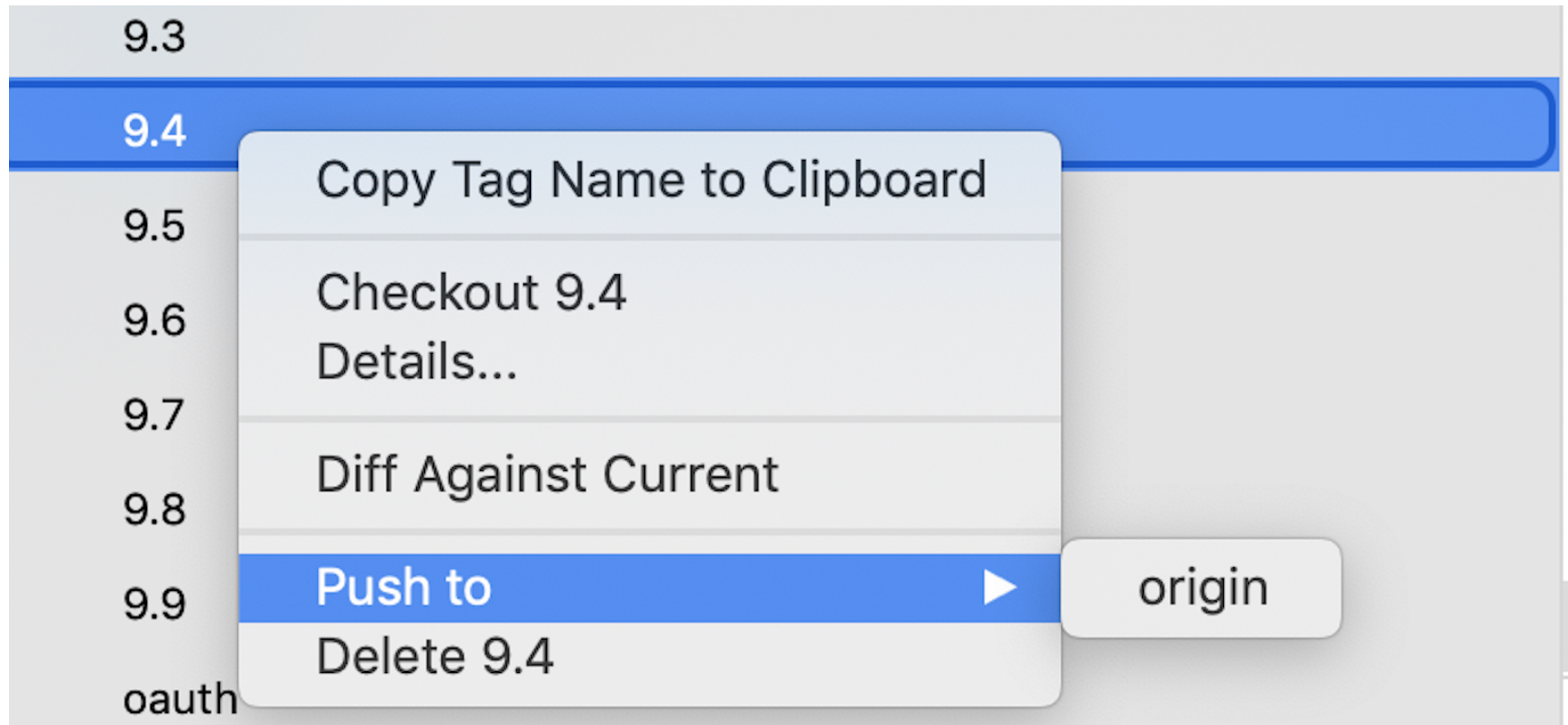
3. Click on New Tag()

4. A dialog appears to Add Tag and Remove Tag

5. Click on Add Tag from give name to tag (preferred version name of the code)



6. If you want to push the TAG on remote, while creating the TAG ref: step 5 which gives checkbox **push TAG to origin** check it and pushed tag appears on remote repository
7. In case while creating the TAG if you have forgotten to check the box **Push to origin**, you can do it later by right-clicking on the created TAG, click on **Push to origin**.



edited Mar 17 at 13:28

answered Feb 8 at 16:38



Suresh Maidaragi

494 7 14

This answer explains how to create a local tag using Sourcetree, but the question is all about how to push the tag to the remote repository (GitHub in this case). You say to do this in step 6, but you left out the instructions for it. – [Rory O'Kane](#) Feb 17 at 19:11

@RoryO'Kane please check it now I have updated my answer, Thanks for pointing it out – [Suresh Maidaragi](#) Mar 17 at 13:29



0



For creating git tag you can simply run `git tag <tagname>` command by replacing with the actual name of the tag. Here is a complete tutorial on the basics of managing git tags: <https://www.drupixels.com/blog/git-tags-create-push-remote-checkout-and-much-more>

answered Feb 26 at 7:54

Gulab Bisht



protected by [Josh Crozier](#) Dec 27 '18 at 22:47

Thank you for your interest in this question. Because it has attracted low-quality or spam answers that had to be removed, posting an answer now requires 10 [reputation](#) on this site (the [association bonus](#) does not count).

Would you like to answer one of these [unanswered questions](#) instead?

