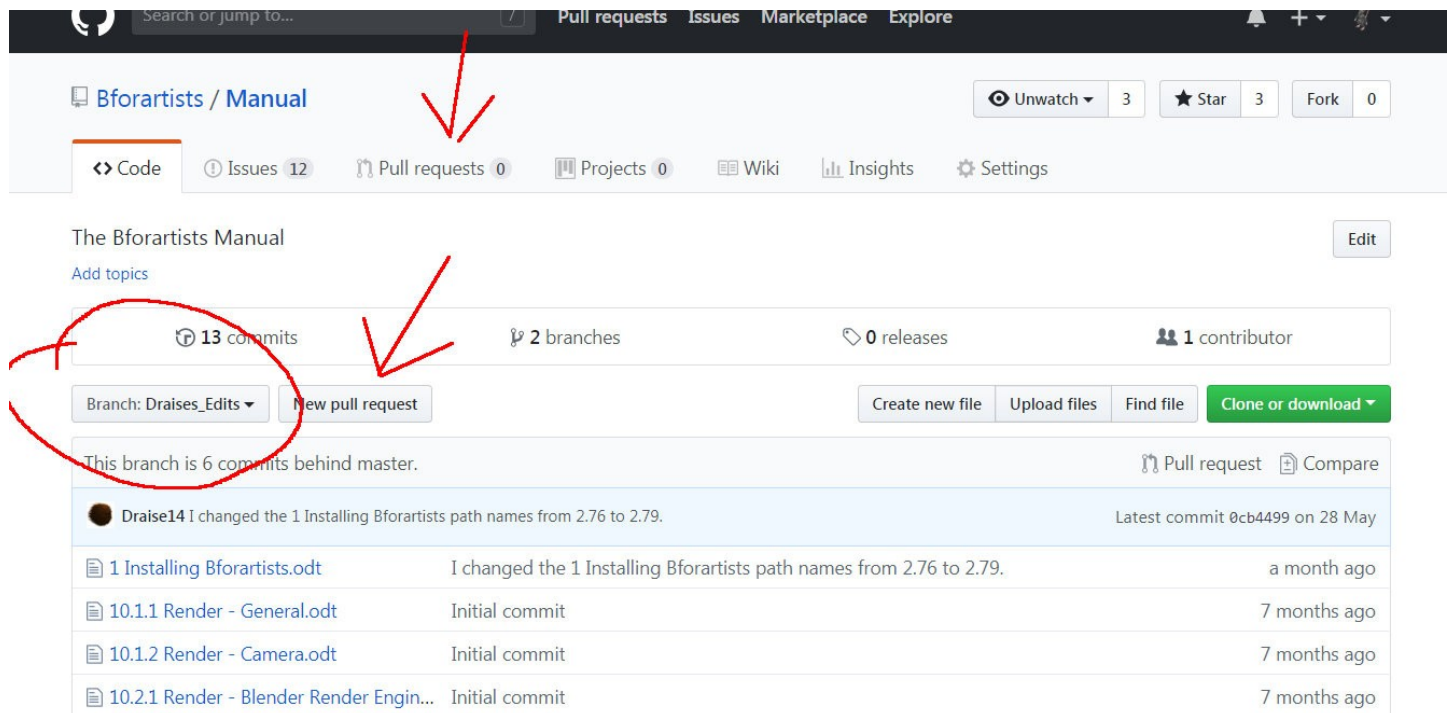


## How to Pull Request

In Bforartists just the Module owners have the right to commit to the master. Other team members needs to develop in branches. And do a pull request when they want their changes to go into the master

This little tutorial is how to do such a pull request. And all further required steps.

When you think that your commit is ready to go into the master, then you can do a Pull Request. See Button New Pull Request. This puts your changes into the Pull Requests queue. The repository owner. will see the pull request then, and can merge it into the master. Or contact the author when he think this commit needs further changes.



The screenshot shows the GitHub repository page for 'Bforartists / Manual'. The page includes a navigation bar with 'Code', 'Issues 12', 'Pull requests 0', 'Projects 0', 'Wiki', 'Insights', and 'Settings'. Below the navigation bar, the repository name 'The Bforartists Manual' is displayed, along with 'Add topics' and an 'Edit' button. A summary bar shows '13 commits', '2 branches', '0 releases', and '1 contributor'. A red circle highlights the 'New pull request' button, which is located next to the 'Branch: Draises\_Edits' dropdown. Below the summary bar, a message states 'This branch is 6 commits behind master.' and provides links for 'Pull request' and 'Compare'. A list of commits is shown, with the most recent commit by 'Draise14' dated '28 May'.

Once the Pull request is accepted the changes will go into the master.

Branch: master ▾

Commits on Jun 29, 2018

- Merge branch 'master' of https://github.com/Bforartists/Manual  
ReinerBforartists committed 11 minutes ago 85bd637 <>
- Merge pull request #15 from Bforartists/Draises\_Edits ... Verified 7fd6fc9 <>
- Merge branch 'master' of https://github.com/Bforartists/Manual  
ReinerBforartists committed 16 minutes ago d091f04 <>
- Added chapter 3.6.1 Editors - Text Editor by tools.odt  
ReinerBforartists committed 17 minutes ago 1c6af4b <>

Commits on Jun 25, 2018

- Update README.md  
ReinerBforartists committed 4 days ago 4871853 <>

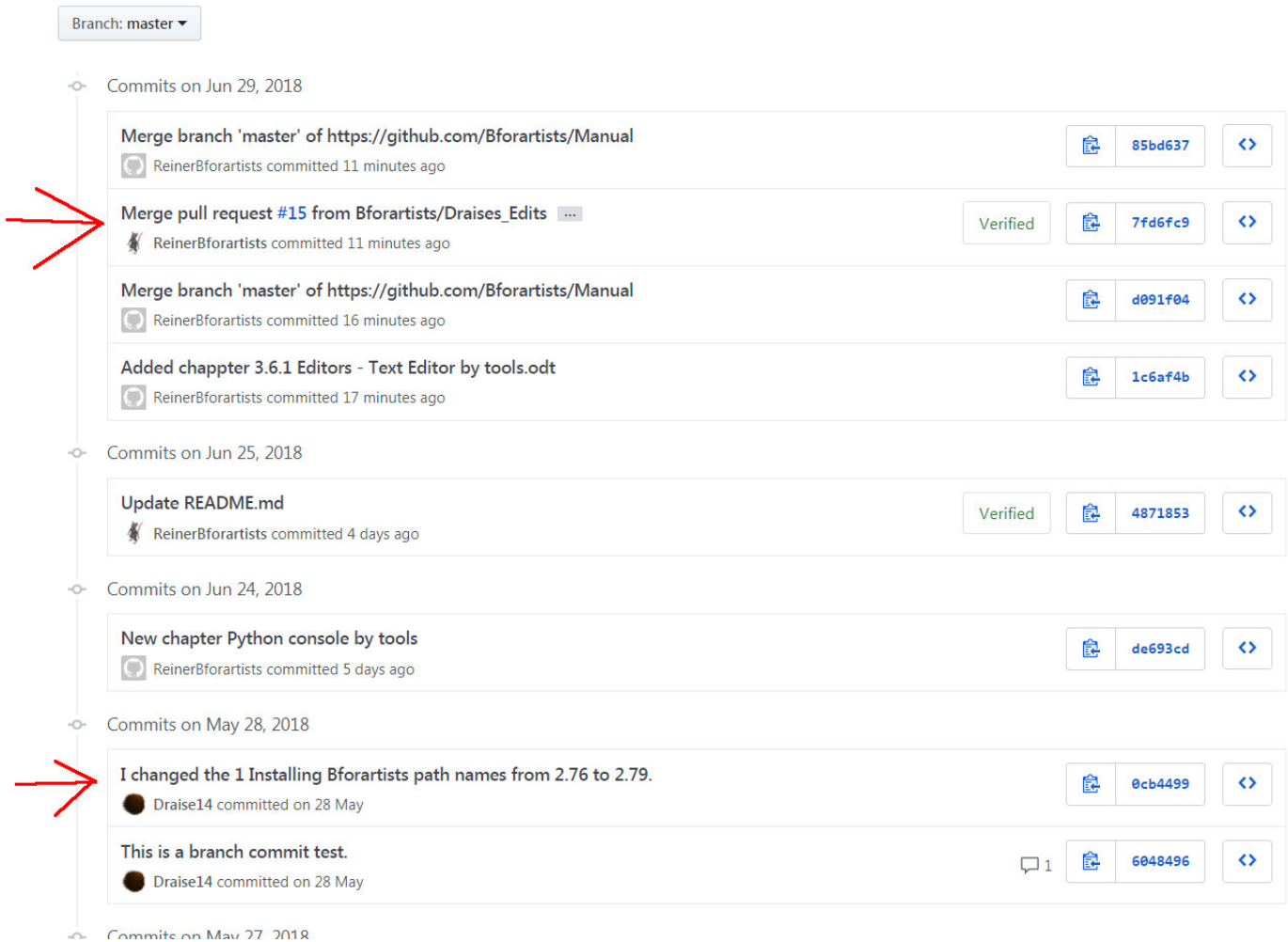
Commits on Jun 24, 2018

- New chapter Python console by tools  
ReinerBforartists committed 5 days ago de693cd <>

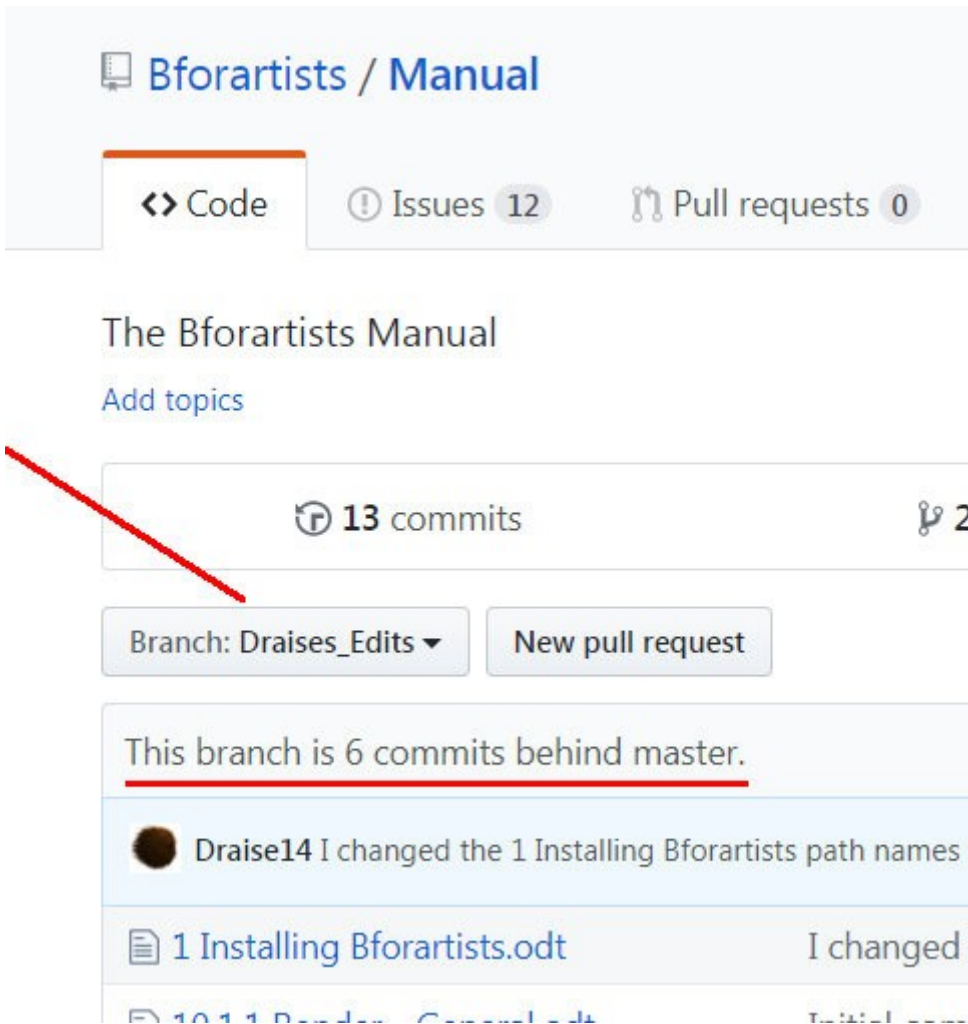
Commits on May 28, 2018

- I changed the 1 Installing Bforartists path names from 2.76 to 2.79.  
Draise14 committed on 28 May 0cb4499 <>
- This is a branch commit test.  
Draise14 committed on 28 May 6048496 <> 1

Commits on May 27, 2018



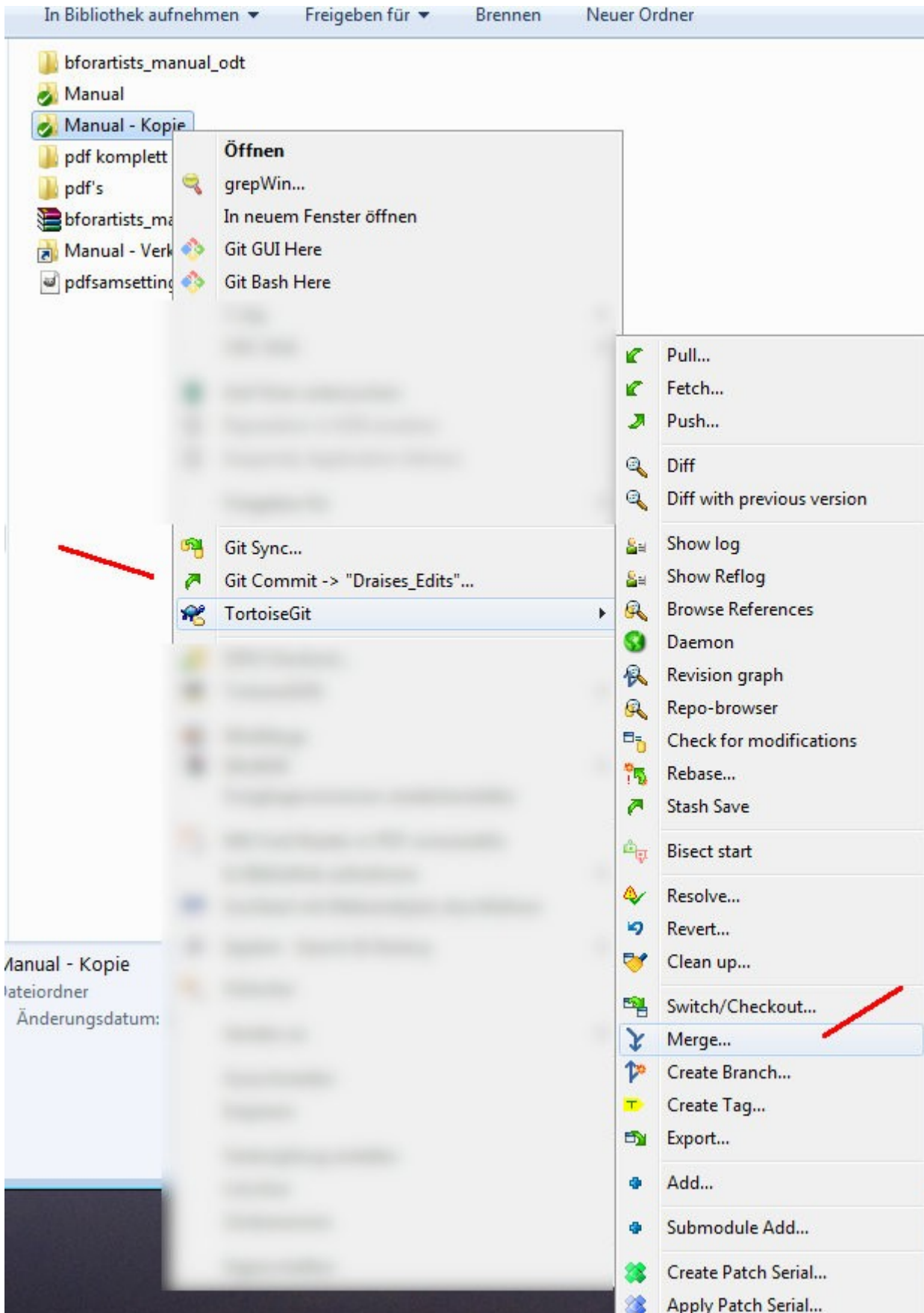
It can also happen that a pull request gets declined, and some further changes are required. But let's say the pull request is accepted. And the merge goes well. Then the branch is usually some commits behind the master.



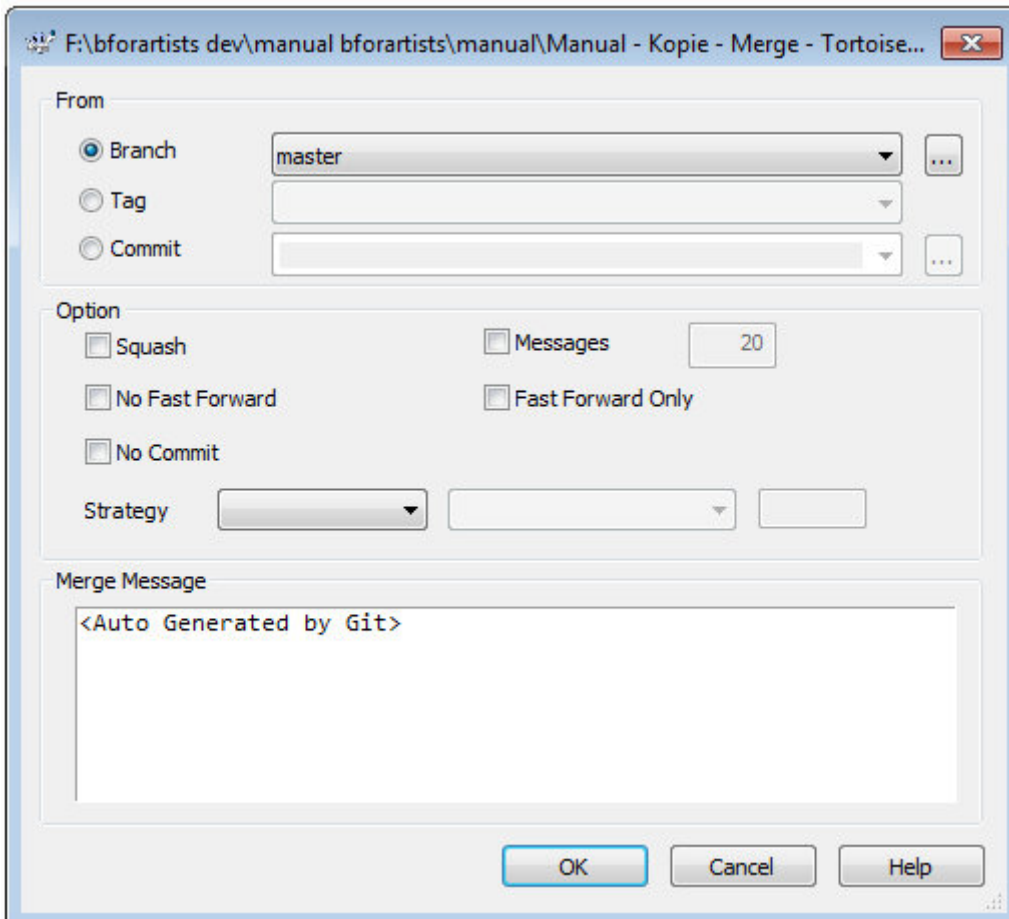
There are two solutions. The easy one is simply to create a new branch for new changes. This one can make sense when you start a new development chapter.

The a bit more complicated solution is to merge the Master into the branch. A repository owner can do this in Github by simply doing a pull request into the other direction. But as a branch owner without access to the master branch you have to do a bit more work. Which can be done at console level. Or with Tortoisegit. Which is what i would prefer you to do. It's two clicks.

Be sure to be in the branch. Not in the master. Choose Merge in the Tortoisegit menu.

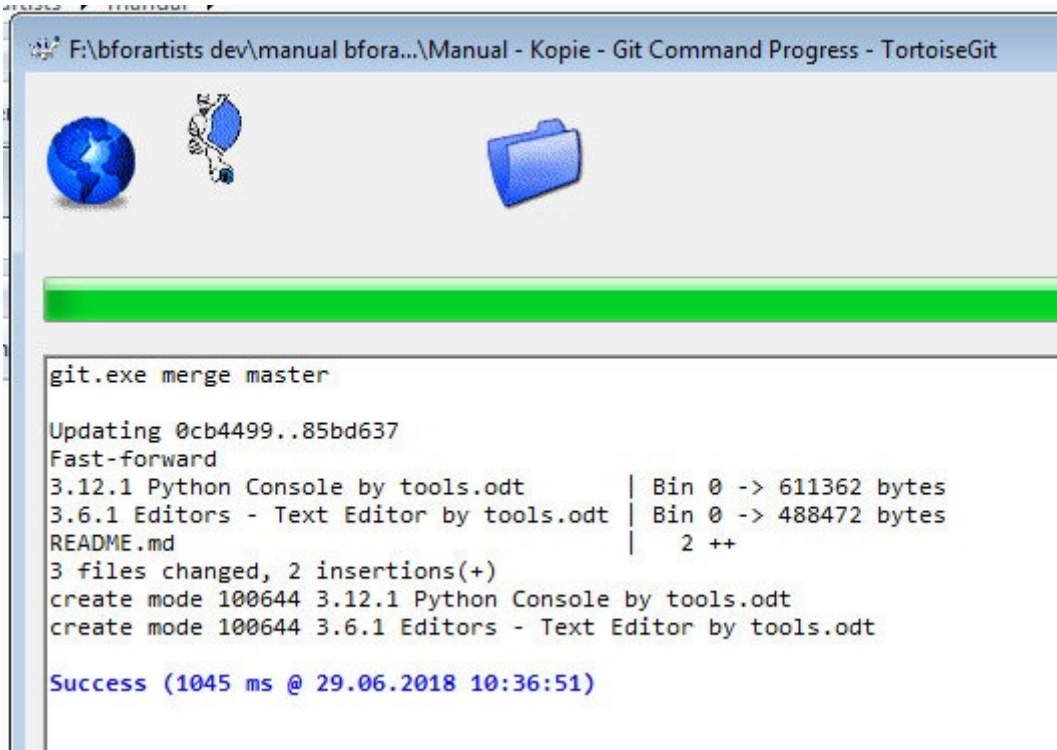


Now change the "FROM" branch to Master. And click OK.



Now the files from the master will be packed into the branch.

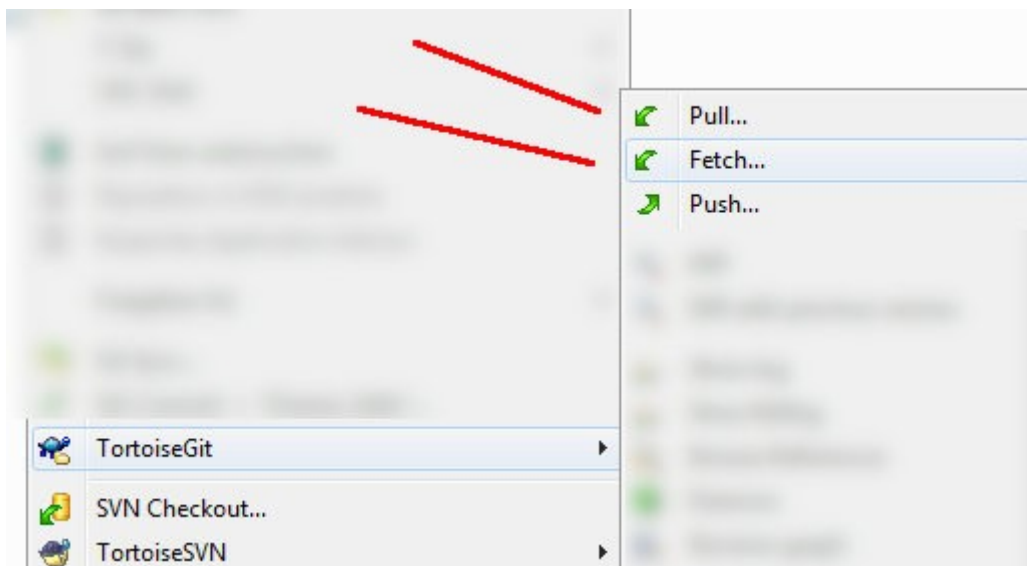
This does the changes at your local repository. What is left now is to commit the changes by a Push, so that the changes goes also into the Github repository. And then you can go on with further changes.



```
git.exe merge master
Updating 0cb4499..85bd637
Fast-forward
 3.12.1 Python Console by tools.odt      | Bin 0 -> 611362 bytes
 3.6.1 Editors - Text Editor by tools.odt | Bin 0 -> 488472 bytes
 README.md                               | 2 ++
3 files changed, 2 insertions(+)
create mode 100644 3.12.1 Python Console by tools.odt
create mode 100644 3.6.1 Editors - Text Editor by tools.odt

Success (1045 ms @ 29.06.2018 10:36:51)
```

What you should also do from time to time is to do a Fetch and a Pull to keep your local repository up to date with the global repository at Github. This avoids to get "Git did not exit clean" warnings. First fetch, then pull ...



What is not covered here is Merge conflicts, which is a different chapter then ...