

Git workflow with RStudio

Aoife McCarthy

AUTHOR

Git commands

Git setup

Git copying repositories

Have you already copied the repository into a local directory? (i.e. your 'Git' folder in CancerStats1)

Checkout the main branch and pull any changes

Create a new feature branch to do your piece of work on, include your initials and the purpose of the branch in the branch name

Carry out your work, making edits or writing new code, and commit and push changes as you go – remember to write descriptive commit messages!

Ready for your work to be reviewed? Create a Pull Request and assign a reviewer (hopefully someone with a bit of info on the project already)

Copy the repository from Git using HTTPS or SSH URLs, setup new project in RStudio

```
$ git checkout main
$ git pull
```

```
$ git checkout -b <new_branch_name>
$ git push origin <new_branch_name>
```

```
$ git add <file_name>
$ git add . (-to add all files changed)
$ git push
```

Pull requests tab >

New pull request

>

Reviewers

DitteA

Reviewer looks over the changes on GitHub/Gitea, then runs the code in RStudio to ensure it still works in practice

Reviewer submits review, either requesting changes or giving approval

Once approved, reviewer merges the feature branch with the main branch, and delete the feature branch

Once your branch has been merged, checkout the main branch and pull changes to update your local version

Is this a main version of the code? i.e. used to produce annual publication, or bi-annual KPIs

Assign a tag to the merge commit and push the tag

```
$ git tag -a <tag_name> -m "<message>"
$ git push origin <tag_name>
```

Delete old branches in RStudio to ensure you don't accidentally keep working on them locally

```
$ git remote prune origin
$ git branch -d <branch_name>
```



REVIEWER

AUTHOR