

# Developing code for ICON

[How to merge a branch back to icon-mpim:master](#) documents how to merge your changes back to mpim master. However before embarking on this procedure please clarify with your supervisor if the proposed developments are desired.

## Working on your own feature branch

### 1) In case you do not have the repository at all

```
git clone --recursive git@gitlab.dkrz.de:icon/icon-mpim.git
```

```
cd icon-mpim
```

You are now automatically on 'master', which is our main development branch and serves as reference.

Checkout a new ( feature-/bugfix- ) branch

```
git checkout -b feature-<my-branch>
```

**In case the feature branch already exists you have to skip the -b!** to simply change into the branch

Use the status command to check your status

```
git status
```

The result should look like

```
On branch feature-<my-branch>
nothing to commit, working directory clean
```

### 2) Start doing your changes....

...use "git add/rm <file>..." to update what will be committed → your files will be staged

Check again with

```
git status
```

### 3) Commit your changes

```
git commit
```

The commit message has to begin with [icon-mpim:feature-<my-branch>]. This should be followed by some useful information regarding your commit. See “Tips and Tricks”: [https://wiki.mpimet.mpg.de/doku.php?id=models:icon:tips\\_and\\_tricks#add\\_the\\_current\\_repository\\_and\\_branch\\_name\\_to\\_the\\_commit\\_message\\_automatically](https://wiki.mpimet.mpg.de/doku.php?id=models:icon:tips_and_tricks#add_the_current_repository_and_branch_name_to_the_commit_message_automatically) to build this string automatically

#### 4) Publish your local branch with a different name in our repository:

```
git push origin feature-<my-branch>:feature-<my-branch>-test001
```

So this is like pushing your actual branch to an intermediate copy of your actual branch.

This is necessary for further rebasing of your local feature branch with icon-mpim:master and has some more advantages: \* you can check this version of your branch with buildbot independent of further changes in your local copy \* you have a copy of this state of work \* you can run test experiments with a fixed version of your development

#### 5) Update your (local) branch with master regularly:

First you have to add all changes done in the central icon-mpim.git repository to your local copy

```
git fetch --all
```

Then you have to rebase your changes to icon-mpim:master. This might lead to conflicts, as the work on icon-mpim:master is ongoing. It might also lead to differences in the output!

```
git rebase origin/master
```

Make changes and commit as in 3) and 4). “Publish” your local branch with a new different name in our central repository, for testing or storage, e.g.:

```
git push origin feature-<my-branch>:feature-<my-branch>-test002
```

---

— [Monika Esch](#) 2022/11/09 10:41

From:  
<https://wiki.mpimet.mpg.de/> - MPI Wiki

Permanent link:  
[https://wiki.mpimet.mpg.de/doku.php?id=models:icon:code\\_development\\_aes](https://wiki.mpimet.mpg.de/doku.php?id=models:icon:code_development_aes)

Last update: **2022/11/09 10:42**

