

Contributing Code to SwiftRiver

In this section, we describe how to code finds its way to the `master` branch.

Contribution Workflow

For SwiftRiver development, we use [GitHub flow](#) which can be summarized as follows:

- Before you begin, fork the code to your GitHub account. Thereafter, clone the repository on your development machine.



To clone the code:

```
$ git clone --recursive git@github.com:<github-account-name>/SwiftRiver.git
```

where `<github-account-name>` is your [GitHub](#) username

- Anything on the `master` branch is production ready hence deployable
- To work on something new (bug fix, feature implementations etc), create a descriptive branch off of `master`. This can be done as follows:

```
$ git checkout -b <branch-name> origin/master
```

where `<branch-name>` is the name of your branch. For more information about branching in Git, click [here](#)

- Commit your changes locally and regularly push your work to the same branch on the server
- When you need help or feedback, or you think your branch is ready for merging, open a [pull request](#)
- After someone else has reviewed your work and given it a nod, you may go ahead and merge (your branch) into `master`