

# Contributor's Guide

---

This document describes instructions and best practices for contributing to this repository. If you are a student unfamiliar with Git and GitHub, please consult the [official Git guides](#), the professor's Git notes, or ask an instructor for assistance.

NOTE: Most files in this repository are written in a syntax called Markdown. For reference, see this [Markdown Cheatsheet](#).

## Step 1: Fork the Repository

First, "fork" the course repository via the GitHub.com online interface to create a copy under the ownership of your own GitHub user. After doing so, you should be able to view your fork online at [https://github.com/YOUR\\_GITHUB\\_USERNAME/intro-to-python](https://github.com/YOUR_GITHUB_USERNAME/intro-to-python).

Next, use the command-line or GitHub Desktop to "clone", or download, your fork onto your computer:

```
git clone https://github.com/YOUR_GITHUB_USERNAME/intro-to-python.git # this
is the HTTPS address, but alternatively you can use the SSH address
```

After cloning it, navigate to the local repo from the command-line:

```
cd intro-to-python
```

## Managing Remote Addresses

As a one-time setup step which will allow you to keep your fork up-to-date in the future, we need to ensure the local repo has an association with each of two different remote addresses (called "origin" and "upstream", respectively).

As a result of the cloning process, the remote address called "origin" should have already been established:

```
git remote -v
#> origin  https://github.com/YOUR_GITHUB_USERNAME/intro-to-python.git
(fetch)
#> origin  https://github.com/YOUR_GITHUB_USERNAME/intro-to-python.git
(push)
```

If you cloned via GitHub Desktop, the remote address called "upstream" should have already been established as well (see output of the `git remote -v` command below). Otherwise, you may need to first configure the

"upstream" remote address manually:

```
git remote add upstream https://github.com/prof-rossetti/intro-to-python.git
# this is the HTTPS address, but alternatively you can use the SSH address
```

When you see associations with both remote addresses, you are ready to move on:

```
git remote -v
#> origin    https://github.com/YOUR_GITHUB_USERNAME/intro-to-python.git
(fetch)
#> origin    https://github.com/YOUR_GITHUB_USERNAME/intro-to-python.git
(push)
#> upstream https://github.com/prof-rossetti/intro-to-python.git (fetch)
#> upstream https://github.com/prof-rossetti/intro-to-python.git (push)
```

## Refreshing your Fork

If you have previously forked the course repository and now your fork is out of date / behind the "upstream" course repository, then follow these steps to refresh your fork from the command-line (Terminal or Git Bash):

```
git branch # make sure you are on the "master" branch, or switch to it

# update your local repository to reflect the contents of the "upstream"
remote repository:
git pull upstream master

# update your "origin" remote repository to reflect the contents of your
local repository:
git push origin master
```

NOTE: if you see a message about "conflicts" when you pull or push, run: `git reset --hard upstream master` to do a hard reset

WARNING: refreshing your fork will delete any previous changes you have committed there that aren't reflected in the "upstream" course repository, so make sure any outstanding Pull Requests have been merged before refreshing your fork!

## Step 2: Revise File Contents

Revise and save the desired file(s). You can do this locally, or remotely via the GitHub.com online interface by navigating to the file and pressing the "pencil" icon to reveal an online text editor.

When done editing the file, "commit" to save the changes. If you've committed locally, make sure to push or sync your changes back up to your remote fork.

After committing (and perhaps pushing / synching if necessary), you should be able to view your changes reflected online in your remote fork.

## Step 3: Submit a Pull Request

After your remote fork contains the changes you'd like to be included into the course repository, create a Pull Request (PR) against the the "upstream" course repository using the GitHub.com online interface. In the PR message, describe what changes you made and why.

An instructor should review your PR within a timely manner. If the instructor accepts your changes, the instructor will merge them into the course repository's master branch, at which point you should be able to view your changes reflected in the remote "upstream" course repository. Else if there are issues with the submission, the instructor should provide further instruction by commenting on the PR, and may close it.

Congratulations! 🎉 Thanks! 🙏