Fork a repo

A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project.

Mac Windows Linux

GitHub CLI Desktop Web browser

In this article

About forks

Most commonly, forks are used to either propose changes to someone else's project or to use someone else's project as a starting point for your own idea. You can fork a repository to create a copy of the repository and make changes without affecting the upstream repository. For more information, see "Working with forks."

Propose changes to someone else's project

For example, you can use forks to propose changes related to fixing a bug. Rather than logging an issue for a bug you've found, you can:

- Fork the repository.
- Make the fix.
- Submit a pull request to the project owner.

Use someone else's project as a starting point for your own idea.

Open source software is based on the idea that by sharing code, we can make better, more reliable software. For more information, see the "About the Open Source Initiative" on the Open Source Initiative.

For more information about applying open source principles to your organization's development work on GitHub.com, see GitHub's white paper "An introduction to innersource."

When creating your public repository from a fork of someone's project, make sure to include a license file that determines how you want your project to be shared with others. For more information, see "Choose an open source license" at choosealicense.com.

For more information on open source, specifically how to create and grow an open source project, we've created Open Source Guides that will help you foster a healthy open source community by recommending best practices for creating and maintaining repositories for your open source project. You can also take a free GitHub Learning Lab course on maintaining open source communities.

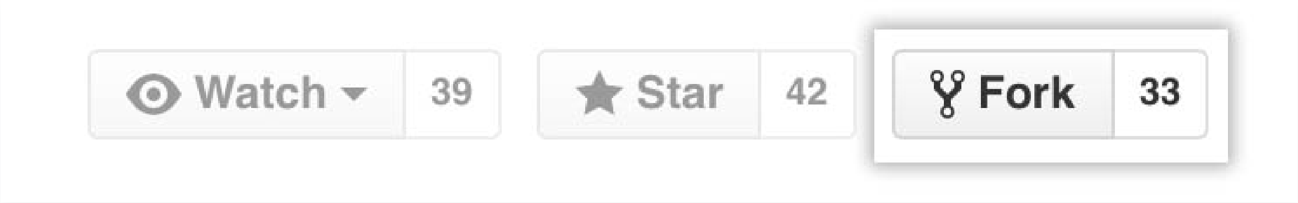
Prerequisites

If you haven't yet, you should first set up Git. Don't forget to set up authentication to GitHub.com from Git as well.

Forking a repository

You might fork a project to propose changes to the upstream, or original, repository. In this case, it's good practice to regularly sync your fork with the upstream repository. To do this, you'll need to use Git on the command line. You can practice setting the upstream repository using the same octocat/Spoon-Knife repository you just forked.

1. On GitHub.com, navigate to the octocat/Spoon-Knife repository.
2. In the top-right corner of the page, click **Fork**.



To learn more about GitHub CLI, see "About GitHub CLI."

To create a fork of a repository, use the `gh repo fork` subcommand.

```
gh repo fork repository
```

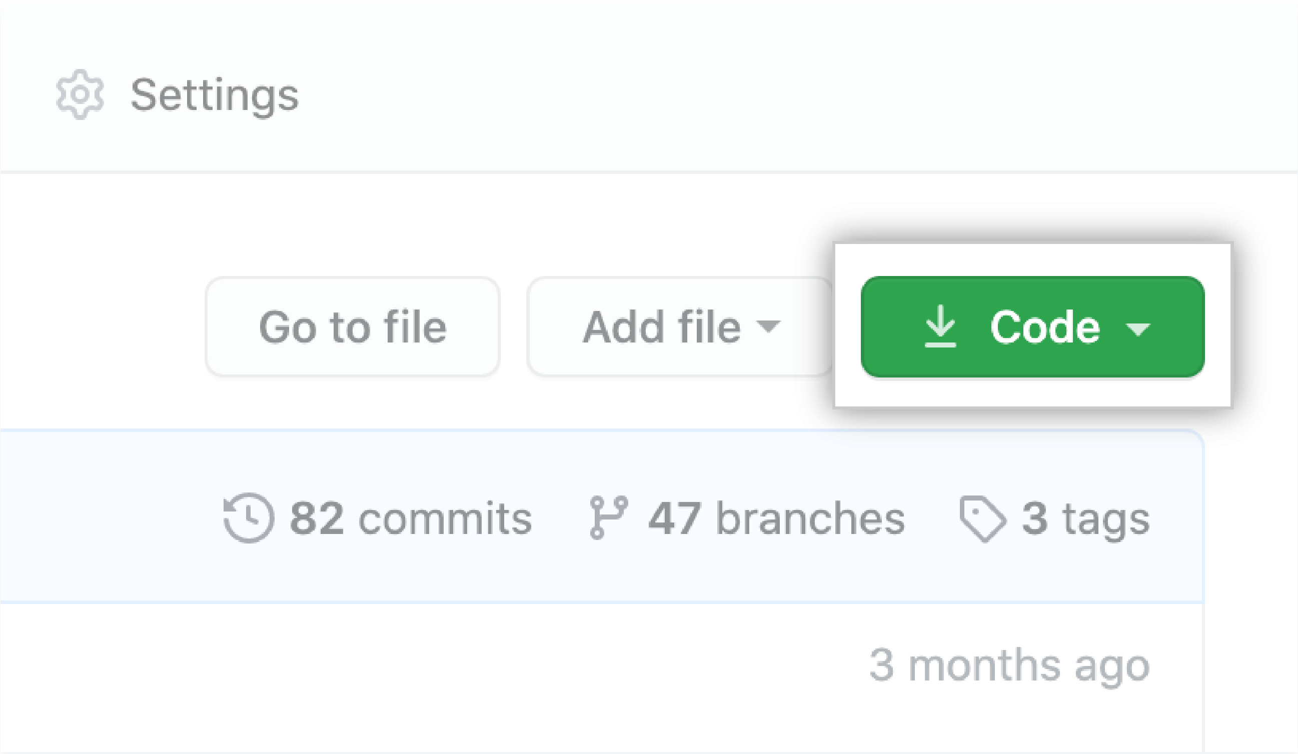
To create the fork in an organization, use the `--org` flag.

```
gh repo fork repository --org "octo-org"
```

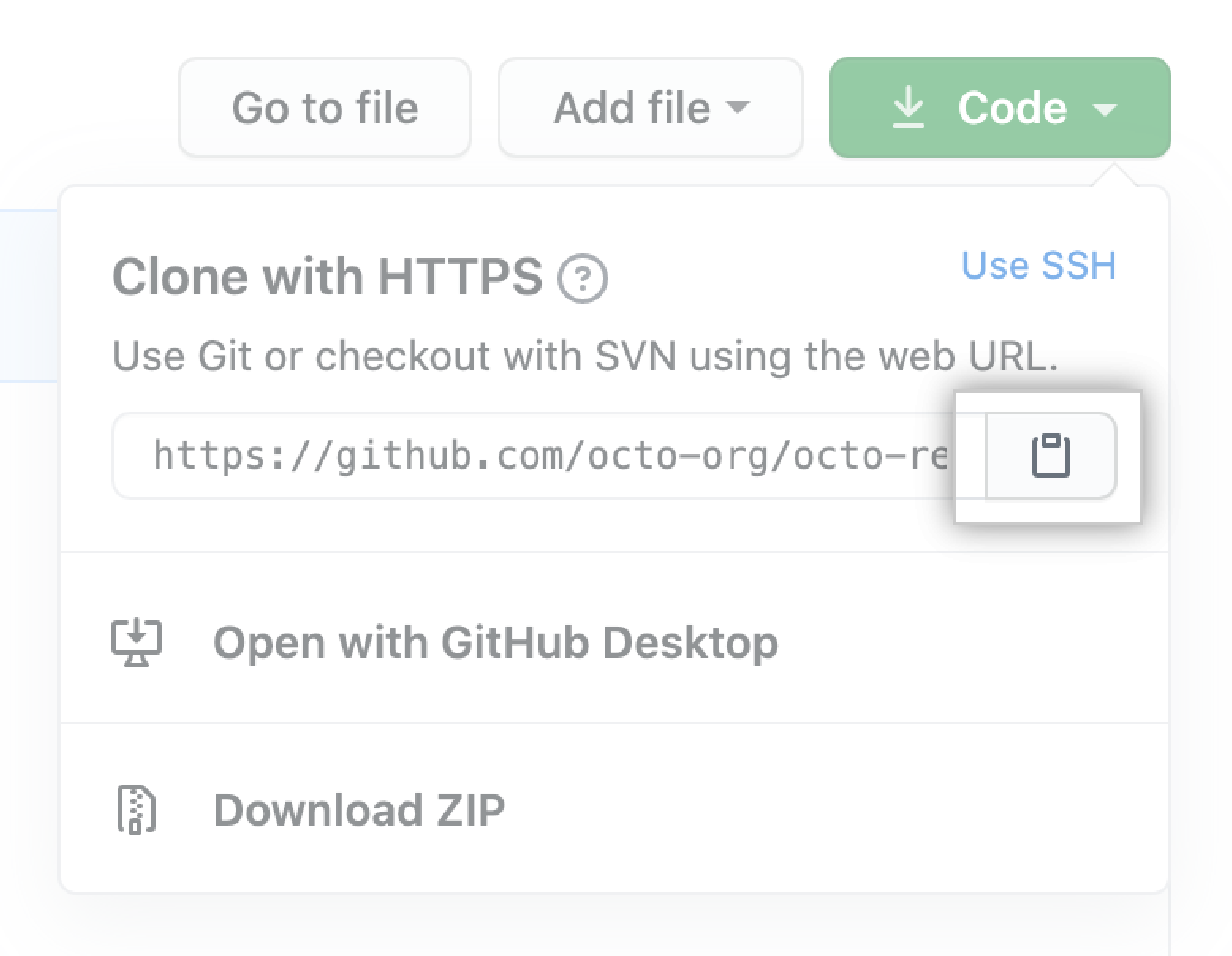
Cloning your forked repository

Right now, you have a fork of the Spoon-Knife repository, but you don't have the files in that repository locally on your computer.

1. On GitHub.com, navigate to **your fork** of the Spoon-Knife repository.
2. Above the list of files, click ⤓ **Code**.



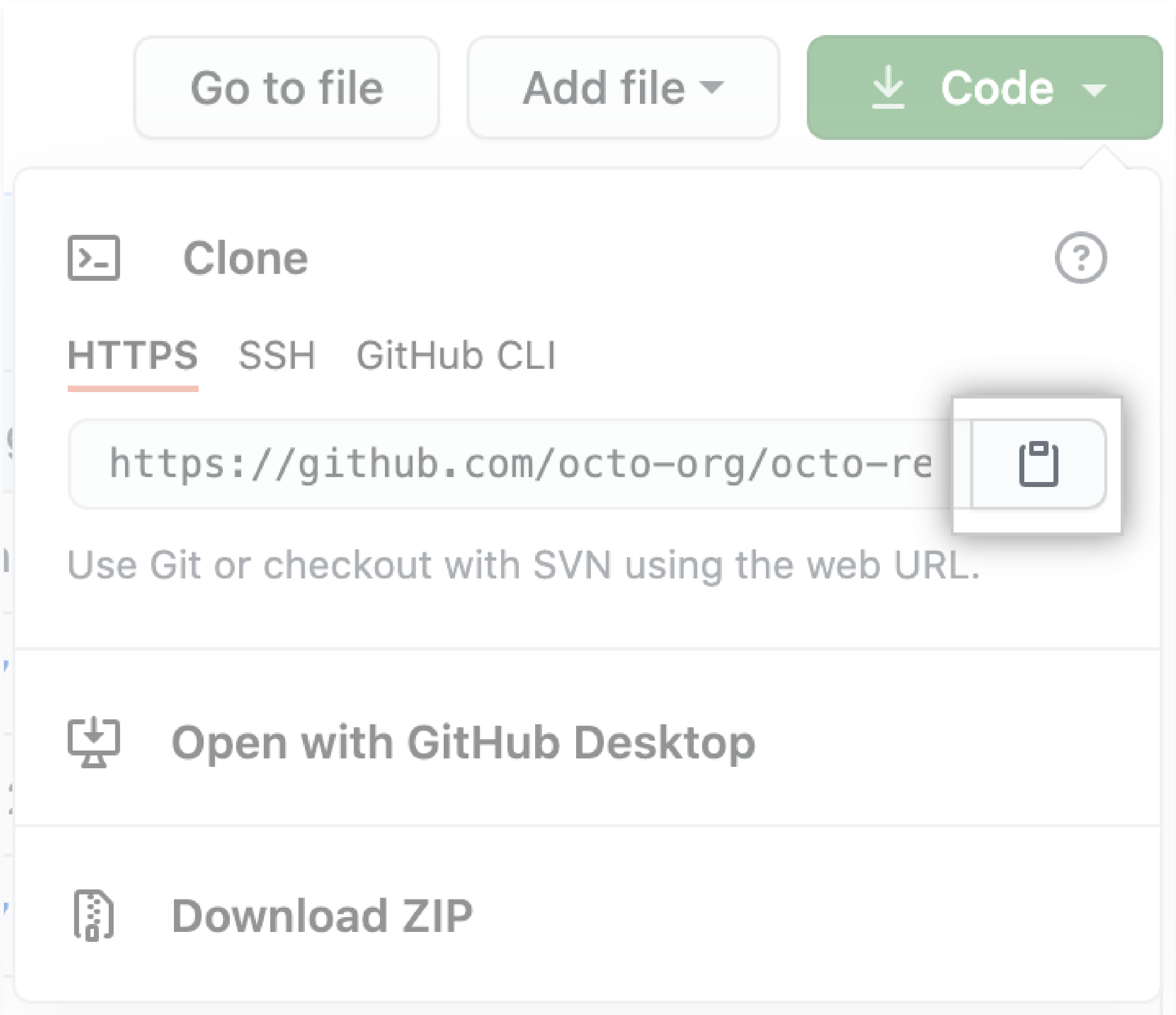
3. To clone the repository using HTTPS, under "Clone with HTTPS", click 📋. To clone the repository using an SSH key, including a certificate issued by your organization's SSH certificate authority, click **Use SSH**, then click 📋. To clone a repository using GitHub CLI, click **Use GitHub CLI**, then click 📋.



Clone with HTTPS

Use Git or checkout with SVN using the web URL.

```
https://github.com/octo-org/octo-re
```


Go to file
Add file
Code
Clone
HTTPS
SSH
GitHub CLI
https://github.com/octo-org/octo-re
Use Git or checkout with SVN using the web URL.
Open with GitHub Desktop
Download ZIP

4. Open Git Bash.

5. Change the current working directory to the location where you want the cloned directory.

6. Type `git clone`, and then paste the URL you copied earlier. It will look like this, with your GitHub username instead of `YOUR-USERNAME`:

```
$ git clone https://github.com/YOUR-USERNAME/Spoon-Knife
```

7. Press **Enter**. Your local clone will be created.

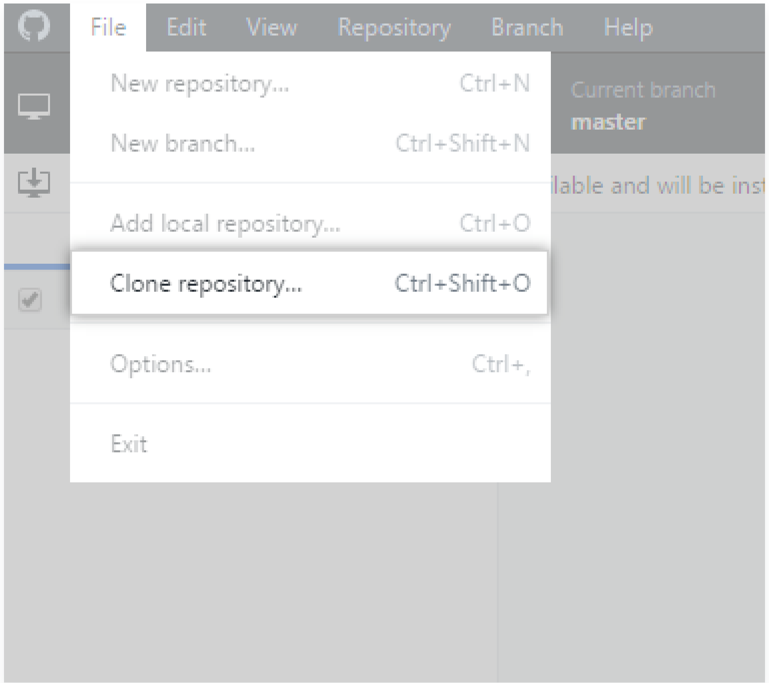
```
$ git clone https://github.com/YOUR-USERNAME/Spoon-Knife
> Cloning into `Spoon-Knife`...
> remote: Counting objects: 10, done.
> remote: Compressing objects: 100% (8/8), done.
> remove: Total 10 (delta 1), reused 10 (delta 1)
> Unpacking objects: 100% (10/10), done.
```

> To learn more about GitHub CLI, see "About GitHub CLI."

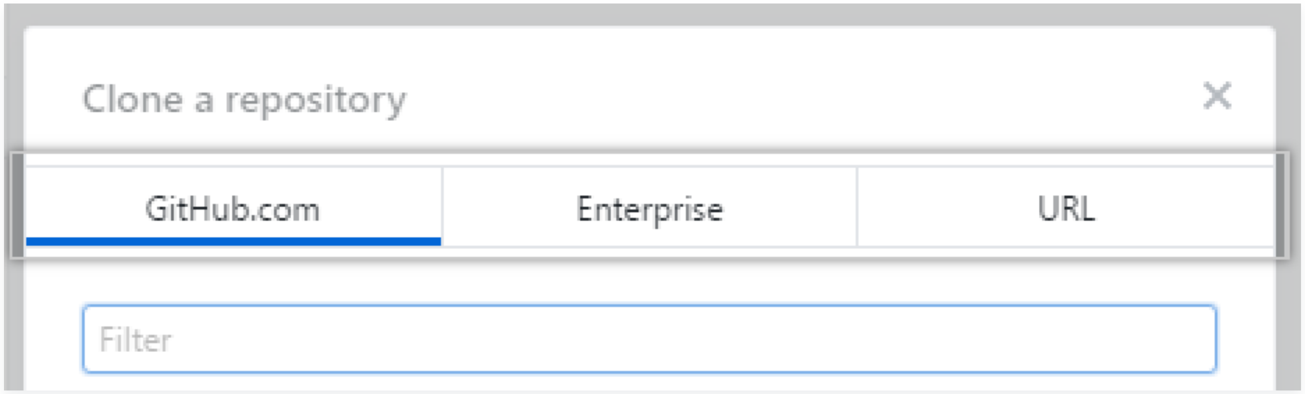
To create a clone of your fork, use the `--clone` flag.

```
gh repo fork repository --clone=true
```

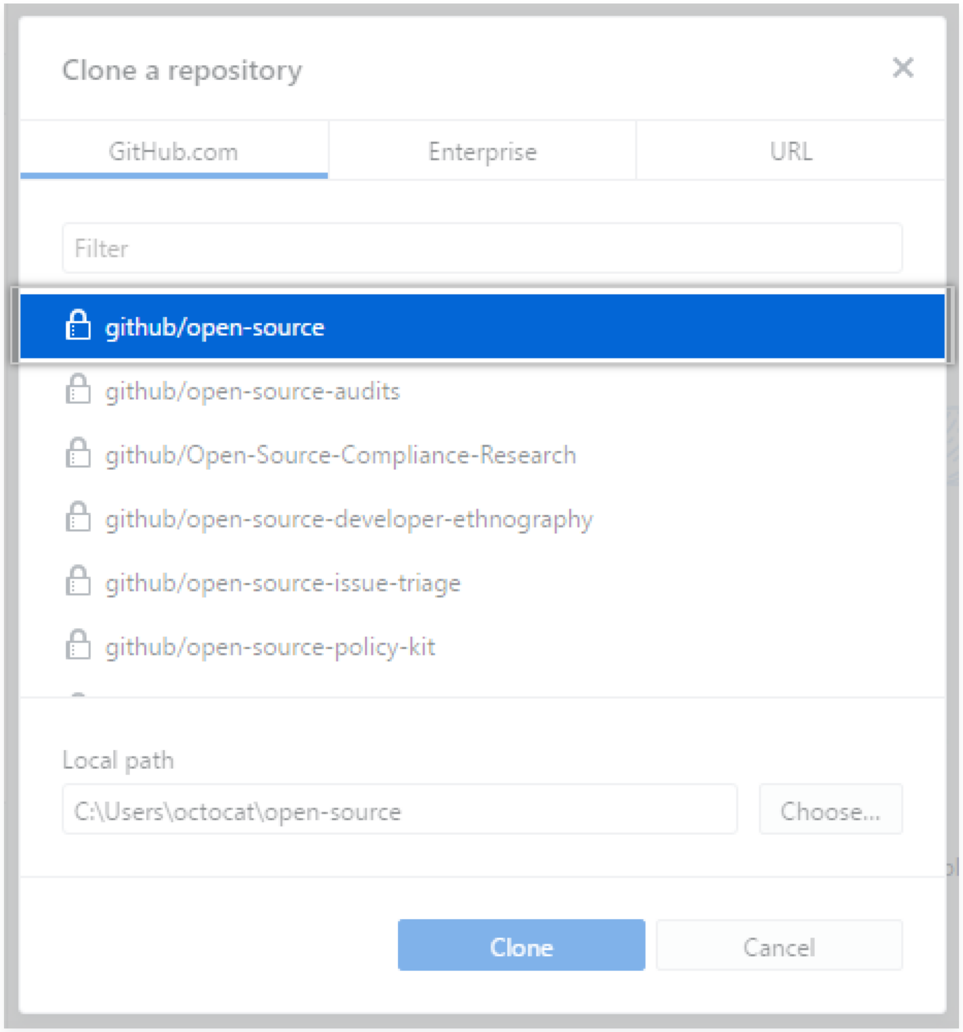
1. In the **File** menu, click **Clone Repository**.



2. Click the tab that corresponds to the location of the repository you want to clone. You can also click **URL** to manually enter the repository location.



3. Choose the repository you want to clone from the list.



github/open-source

github/open-source-audits

github/Open-Source-Compliance-Research

github/open-source-developer-ethnography

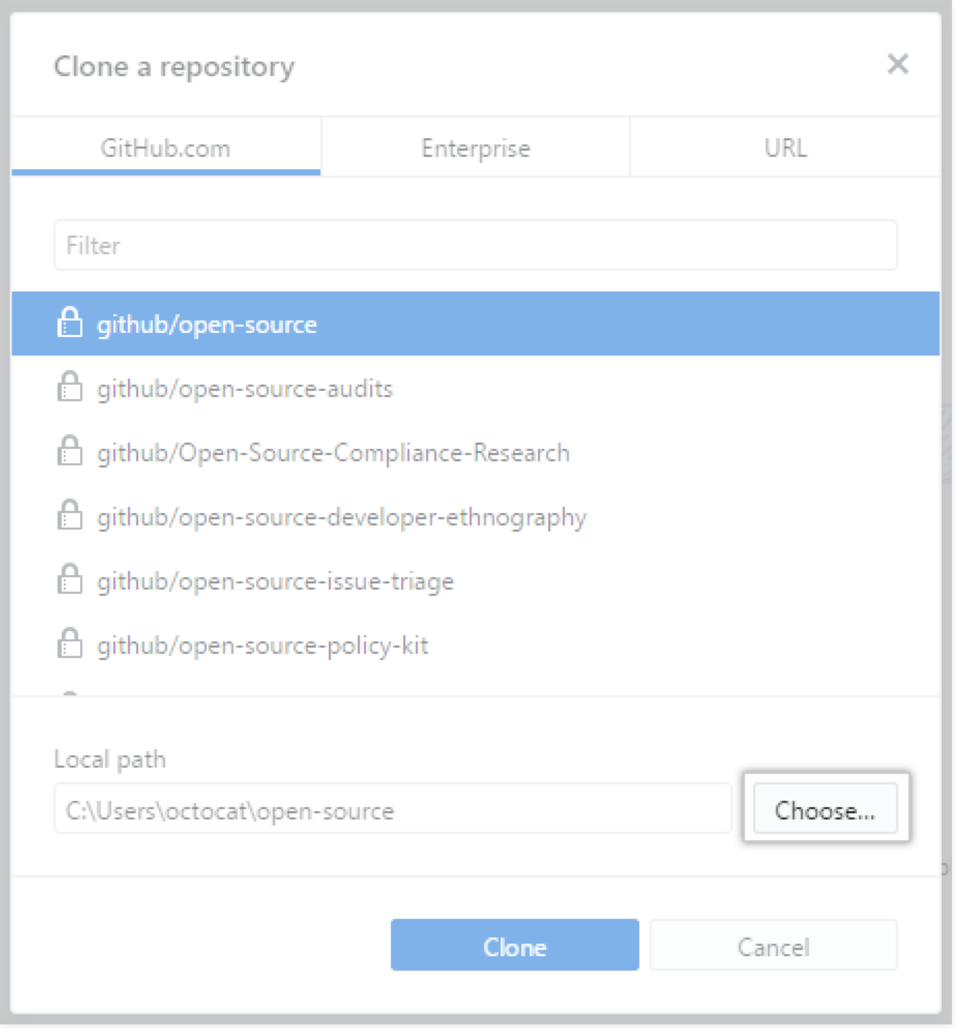
github/open-source-issue-triage

github/open-source-policy-kit

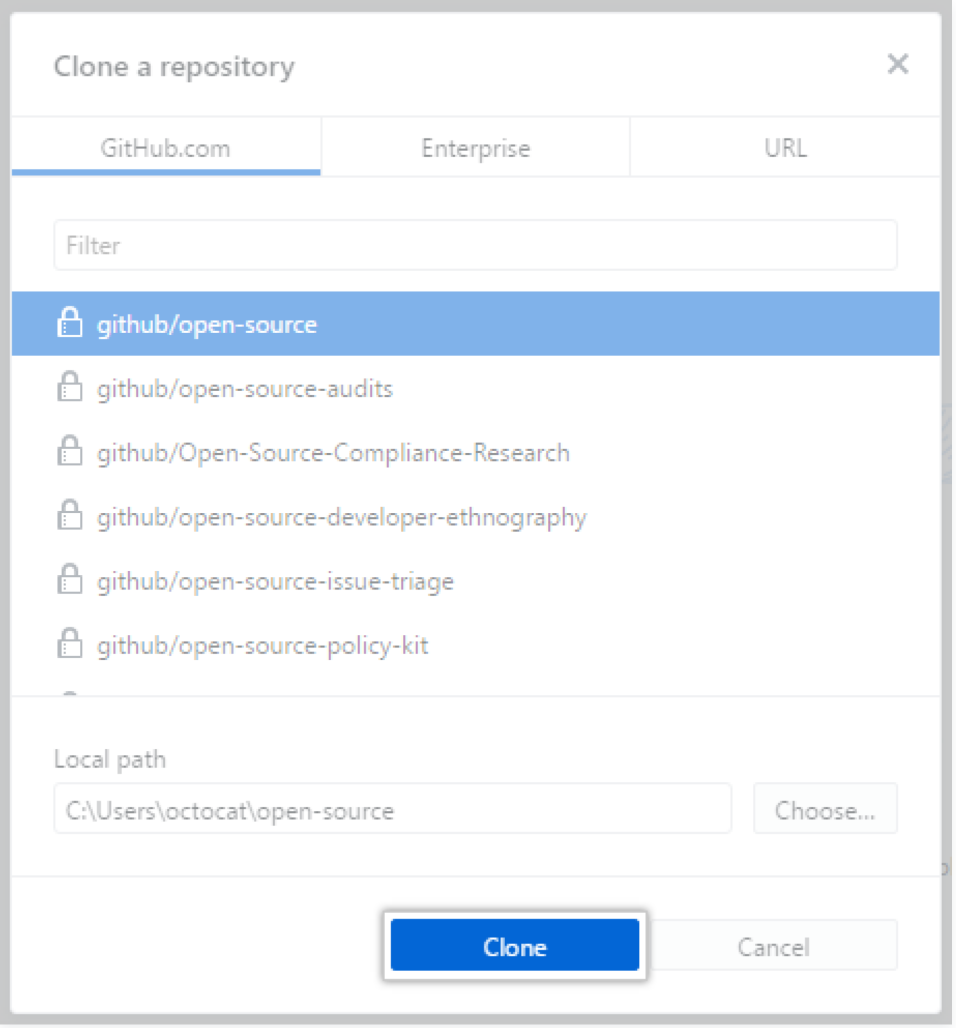
Local path

C:\Users\octocat\open-source

4 Click **Choose...** and navigate to a local path where you want to clone the repository.


Clone a repository
×
GitHub.com
Enterprise
URL
Filter
github/open-source
github/open-source-audits
github/Open-Source-Compliance-Research
github/open-source-developer-ethnography
github/open-source-issue-triage
github/open-source-policy-kit
Local path
C:\Users\octocat\open-source
Choose...
Clone
Cancel

5 Click **Clone**.



github/open-source
github/open-source-audits
github/Open-Source-Compliance-Research
github/open-source-developer-ethnography
github/open-source-issue-triage
github/open-source-policy-kit

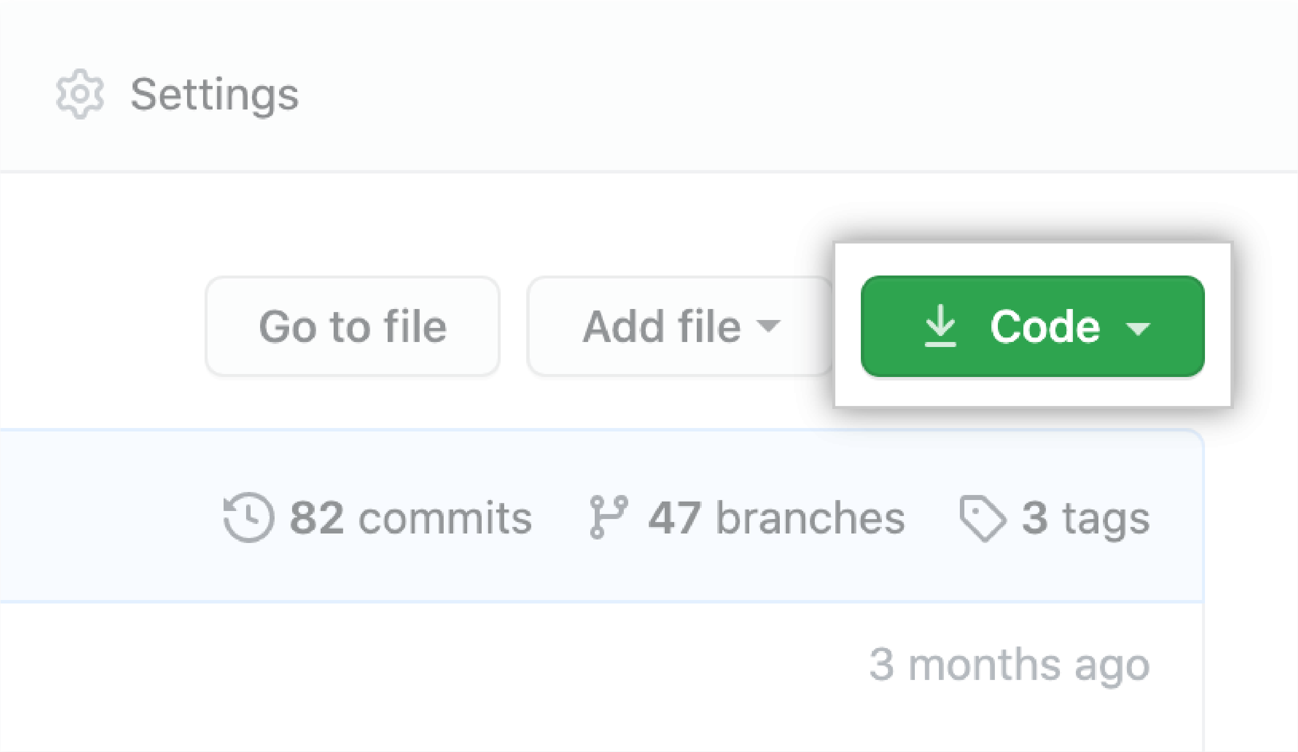
Local path
C:\Users\octocat\open-source

Configuring Git to sync your fork with the original repository

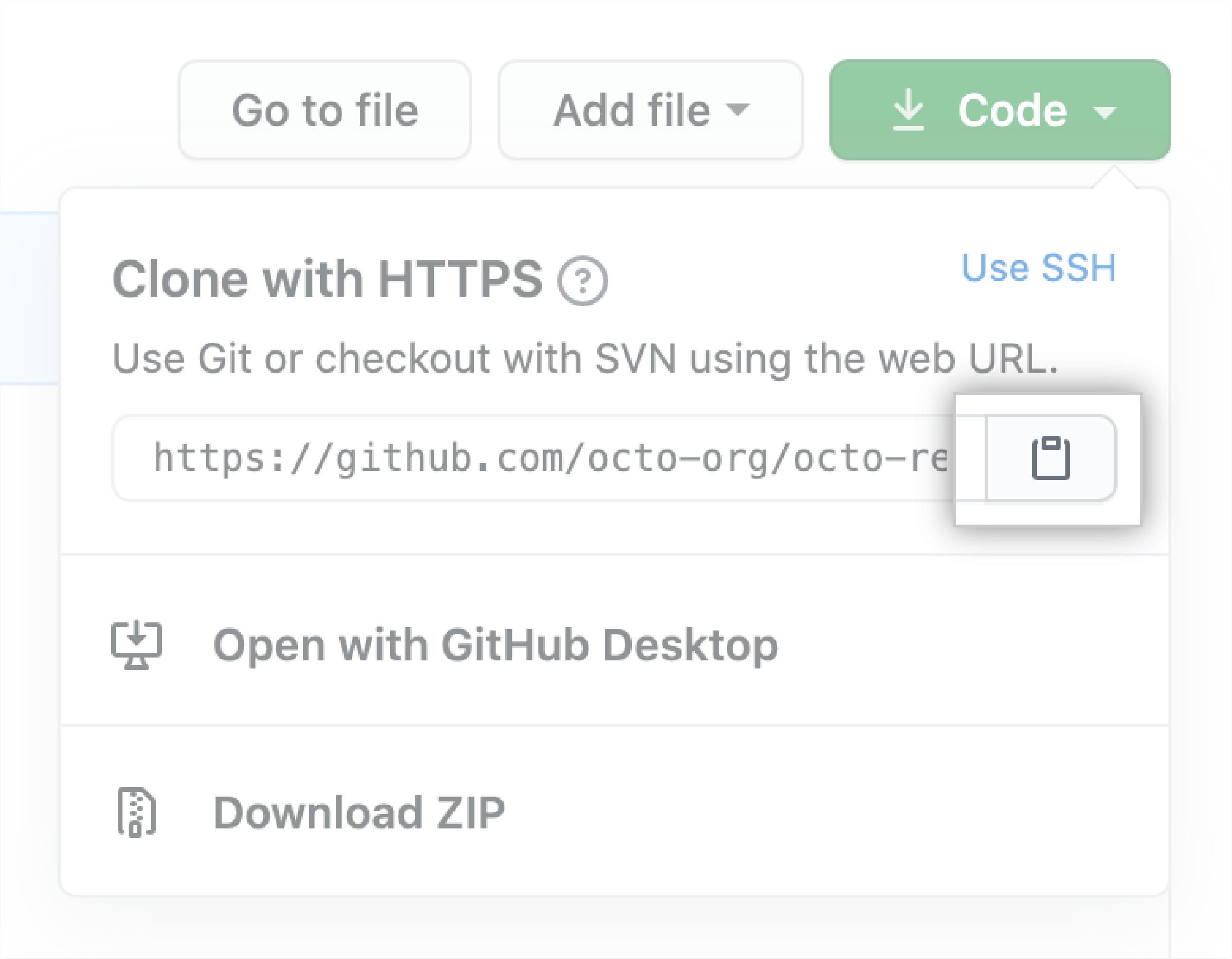
When you fork a project in order to propose changes to the original repository, you can configure Git to pull changes from the original, or upstream, repository into the local clone of your fork.

1. On GitHub.com, navigate to the octocat/Spoon-Knife repository.

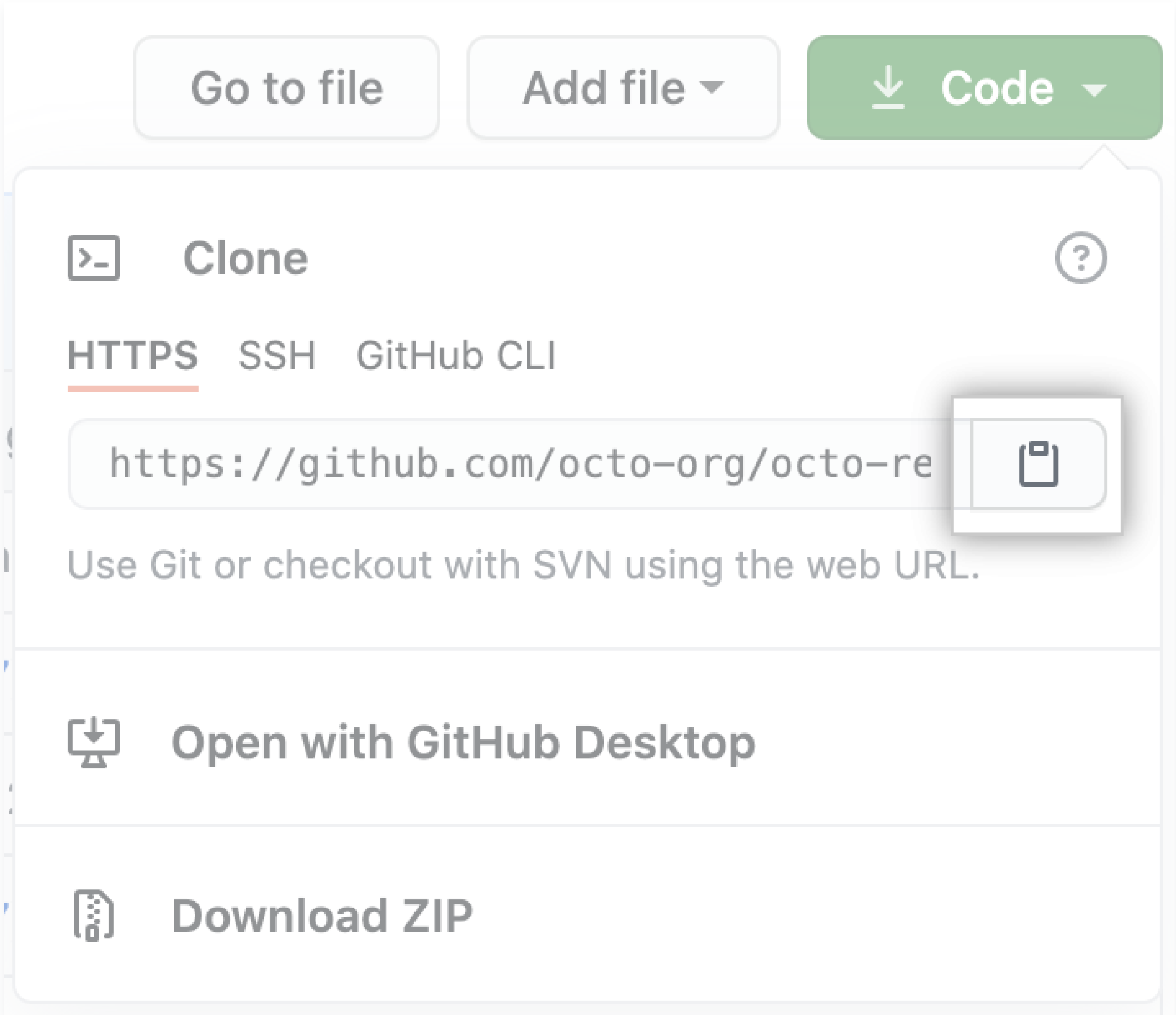
2. Above the list of files, click ⤓ **Code**.



3. To clone the repository using HTTPS, under "Clone with HTTPS", click 📋. To clone the repository using an SSH key, including a certificate issued by your organization's SSH certificate authority, click **Use SSH**, then click 📋. To clone a repository using GitHub CLI, click **Use GitHub CLI**, then click 📋.



Go to file
Add file
Code
Clone with HTTPS
Use SSH
Use Git or checkout with SVN using the web URL.
https://github.com/octo-org/octo-re
Open with GitHub Desktop
Download ZIP


Go to file
Add file
Code
Clone
HTTPS
SSH
GitHub CLI
https://github.com/octo-org/octo-re
Use Git or checkout with SVN using the web URL.
Open with GitHub Desktop
Download ZIP

4. Open Git Bash.

5. Change directories to the location of the fork you cloned.
   - To go to your home directory, type just `cd` with no other text.
   - To list the files and folders in your current directory, type `ls`.
   - To go into one of your listed directories, type `cd your_listed_directory`.

- To go up one directory, type `cd ..`.

6. Type `git remote -v` and press **Enter**. You'll see the current configured remote repository for your fork.

```
$ git remote -v
> origin  https://github.com/YOUR_USERNAME/YOUR_FORK.git (fetch)
> origin  https://github.com/YOUR_USERNAME/YOUR_FORK.git (push)
```

7. Type `git remote add upstream`, and then paste the URL you copied in Step 2 and press **Enter**. It will look like this:

```
$ git remote add upstream https://github.com/octocat/Spoon-Knife.git
```

8. To verify the new upstream repository you've specified for your fork, type `git remote -v` again. You should see the URL for your fork as `origin`, and the URL for the original repository as `upstream`.

```
$ git remote -v
> origin    https://github.com/YOUR_USERNAME/YOUR_FORK.git (fetch)
> origin    https://github.com/YOUR_USERNAME/YOUR_FORK.git (push)
> upstream  https://github.com/ORIGINAL_OWNER/ORIGINAL_REPOSITORY.git (fetch)
> upstream  https://github.com/ORIGINAL_OWNER/ORIGINAL_REPOSITORY.git (push)
```

Now, you can keep your fork synced with the upstream repository with a few Git commands. For more information, see "Syncing a fork."

> To learn more about GitHub CLI, see "About GitHub CLI."

To configure a remote repository for the forked repository, use the `--remote` flag.

```
gh repo fork repository --remote=true
```

To specify the remote repository's name, use the `--remote-name` flag.

```
gh repo fork repository --remote-name "main-remote-repo"
```

Next steps

You can make any changes to a fork, including:

- **Creating branches:** *Branches* allow you to build new features or test out ideas without putting your main project at risk.
- **Opening pull requests:** If you are hoping to contribute back to the original repository, you can send a request to the original author to pull your fork into their repository by submitting a pull request.

Find another repository to fork

Fork a repository to start contributing to a project. You can fork a repository to your user account or any organization where you have repository creation permissions. For more information, see "Roles in an organization."

If you have access to a private repository and the owner permits forking, you can fork the repository to your user account or any organization on GitHub Team where you have repository creation permissions. You cannot fork a private repository to an organization using GitHub Free. For more information, see "GitHub's products."

You can browse Explore to find projects and start contributing to open source repositories. For more information, see "Finding ways to contribute to open source on GitHub."

Celebrate

You have now forked a repository, practiced cloning your fork, and configured an upstream repository. For more information about cloning the fork and syncing the changes in a forked repository from your computer see "Set up Git."

You can also create a new repository where you can put all your projects and share the code on GitHub. For more information see, "Create a repository."

Each repository in GitHub is owned by a person or an organization. You can interact with the people, repositories, and organizations by connecting and following them on GitHub. For more information see "Be social."

GitHub has a great support community where you can ask for help and talk to people from around the world. Join the conversation on Github Support Community.

Terms Privacy Security Status Help Contact GitHub Pricing Developer API Training Blog About