How to create and add to a Github repository

1. Go to your Github profile and find the "new repository" button

2. Enter in all the information about your repository. The name you pick here will be the default name when you clone the repo locally on your computer
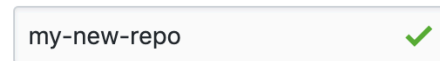
Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.
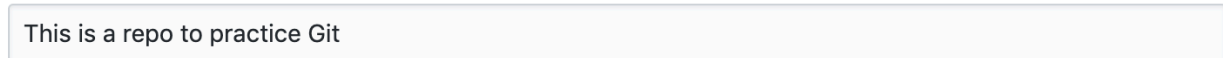
Owner   **Repository name** *

🟫 apeckstl ▾ / my-new-repo   ✓

Great repository names are short and memorable. Need inspiration? How about **improved-disco**?

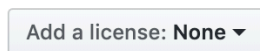**Description** (optional)

This is a repo to practice Git

🔘 📖 **Public**
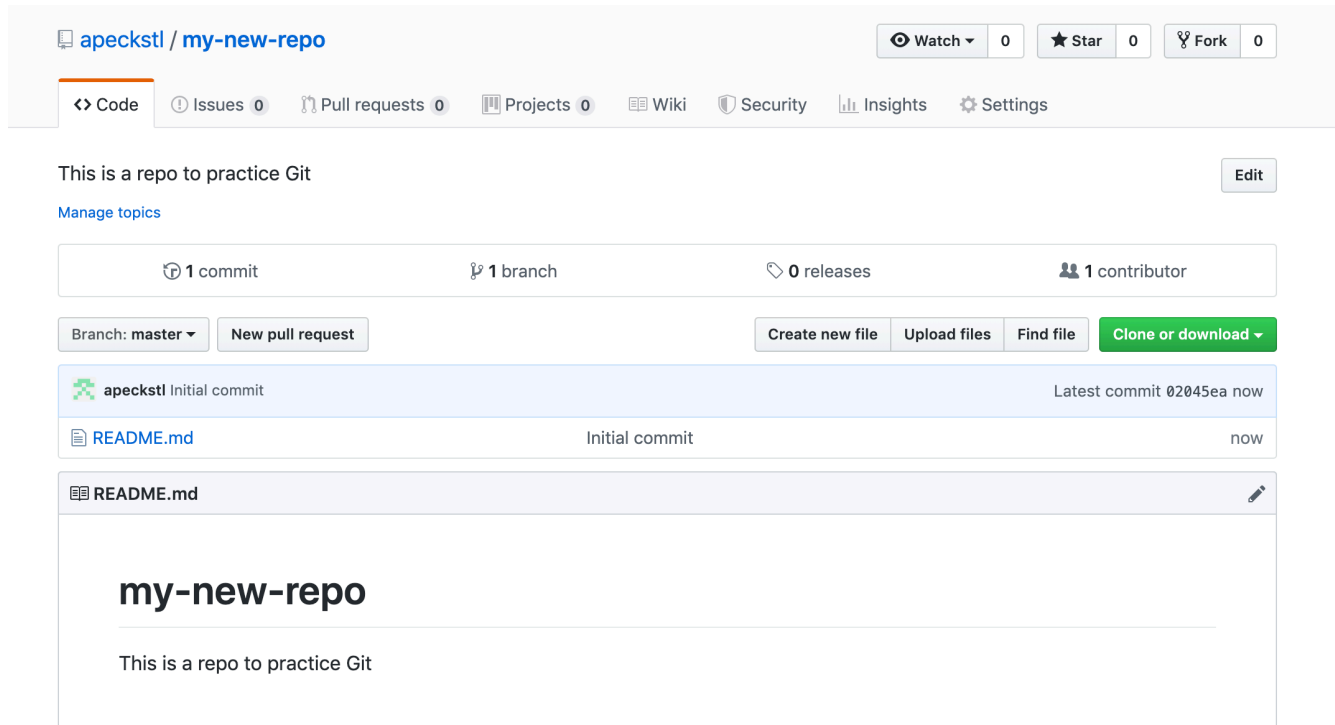Anyone can see this repository. You choose who can commit.

⚪ 🔒 **Private**
You choose who can see and commit to this repository.

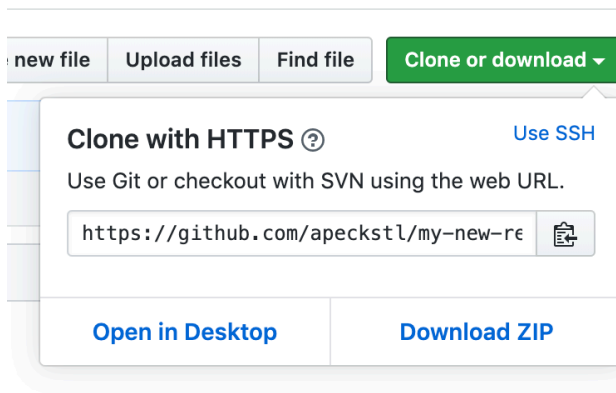Skip this step if you're importing an existing repository.

☑ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: **None** ▾ | Add a license: **None** ▾   ⓘ

Create repository

3. Once you've created the repo, it should look like this:

4. Note how there is one file in your repository, Github automatically adds this if you check the box in the previous step. If you have a file titled README in your base folder, Github will display the contents on the main page of the repo like a nice landing page.
5. Time to clone your repository to your computer so you can make changes!

6. Copy this link to your clipboard.

7. Go to your terminal (Command Prompt on Windows, Terminal on Mac) and type in "git clone THE_URL_YOU_JUST_COPIED. You may be prompted for a username and password to your Github profile. Then change directories so you are *in* your repository.

```
~ git clone https://github.com/apeckstl/my-new-repo.git
Cloning into 'my-new-repo'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
~ cd my-new-repo/
~/my-new-repo (master=)
```

8. Now you have a local copy of the git repository that is stored on Github. Create a file using a basic text editor and put the file in that folder on your computer. You should be able to do this:

```
~/my-new-repo (master ○ =) ls
README.md    newfile.txt
```

9. Now to stage and commit! "git status" is just there to help, you don't need to run it.

```
~/my-new-repo (master ○ =) git add newfile.txt
~/my-new-repo (master ★=) git commit -m "Added first file"
[master ed69212] Added first file
 1 file changed, 1 insertion(+)
 create mode 100644 newfile.txt
~/my-new-repo (master+ ) git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
~/my-new-repo (master+ )
```

10. So your local repository is now 1 commit ahead of the origin (your Github repo), let's fix that.

```
~/my-new-repo (master+ ) git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 291 bytes | 291.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/apeckstl/my-new-repo.git
   02045ea..ed69212  master -> master
~/my-new-repo (master=)
```

11. Now check your changes on the Github website

This is a repo to practice Git   Edit
Manage topics

| ⓘ 2 commits | ⑂ 1 branch | ◌ 0 releases | ⚑ 1 contributor |

| Branch: master ▾ | New pull request | | Create new file | Upload files | Find file | Clone or download ▾ |

🗎 apeckstl Added first file		Latest commit ed69212 33 seconds ago
🗎 README.md	Initial commit	3 minutes ago
🗎 newfile.txt	Added first file	33 seconds ago

📖 README.md   ✎

my-new-repo

This is a repo to practice Git