

HOW TO:

# First Commit

in

# GitHub

[www.sminrana.com](http://www.sminrana.com)

# Create A New Repository

- Visit <https://github.com> and signup for a new account. If you already have one, please sign in.
- Create a new repository by clicking this link <https://github.com/new>

## 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 \*



Great repository names are short and memorable. Need inspiration? How about **solid-octo-funicular**?

Description (optional)

 **Public**

Anyone on the internet can see this repository. You choose who can commit.

 **Private**

You choose who can see and commit to this repository.

Initialize this repository with:

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

**Add a README file**

This is where you can write a long description for your project. [Learn more.](#)

**Add .gitignore**

Choose which files not to track from a list of templates. [Learn more.](#)

**Choose a license**

A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository

Don't forget to click the "Create repository" button.

# Install git on your computer

- For Windows user click here <https://git-scm.com/download/win>. After download please install the Git.exe. **Make sure Git Bash also gets installed.**

## Downloading Git



You are downloading the latest (2.28.0) 32-bit version of Git for Windows. This is the most recent [maintained build](#). It was released **about 2 months ago**, on 2020-07-28.

[Click here to download manually](#)

### Other Git for Windows downloads

Git for Windows Setup

[32-bit Git for Windows Setup.](#)

[64-bit Git for Windows Setup.](#)

Git for Windows Portable ("thumbdrive edition")

[32-bit Git for Windows Portable.](#)

[64-bit Git for Windows Portable.](#)

The current source code release is version 2.28.0. If you want the newer version, you can build it from [the source code](#).

- For Ubuntu user run this command on your terminal.  
`sudo apt-get install git`
- For Mac user run this Homebrew command your terminal  
`brew install git`

## 4. Create an SSH Key and Add in your GitHub

On Windows open git bash and run command

```
ssh-keygen -t rsa -C "your_email@example.com"
```

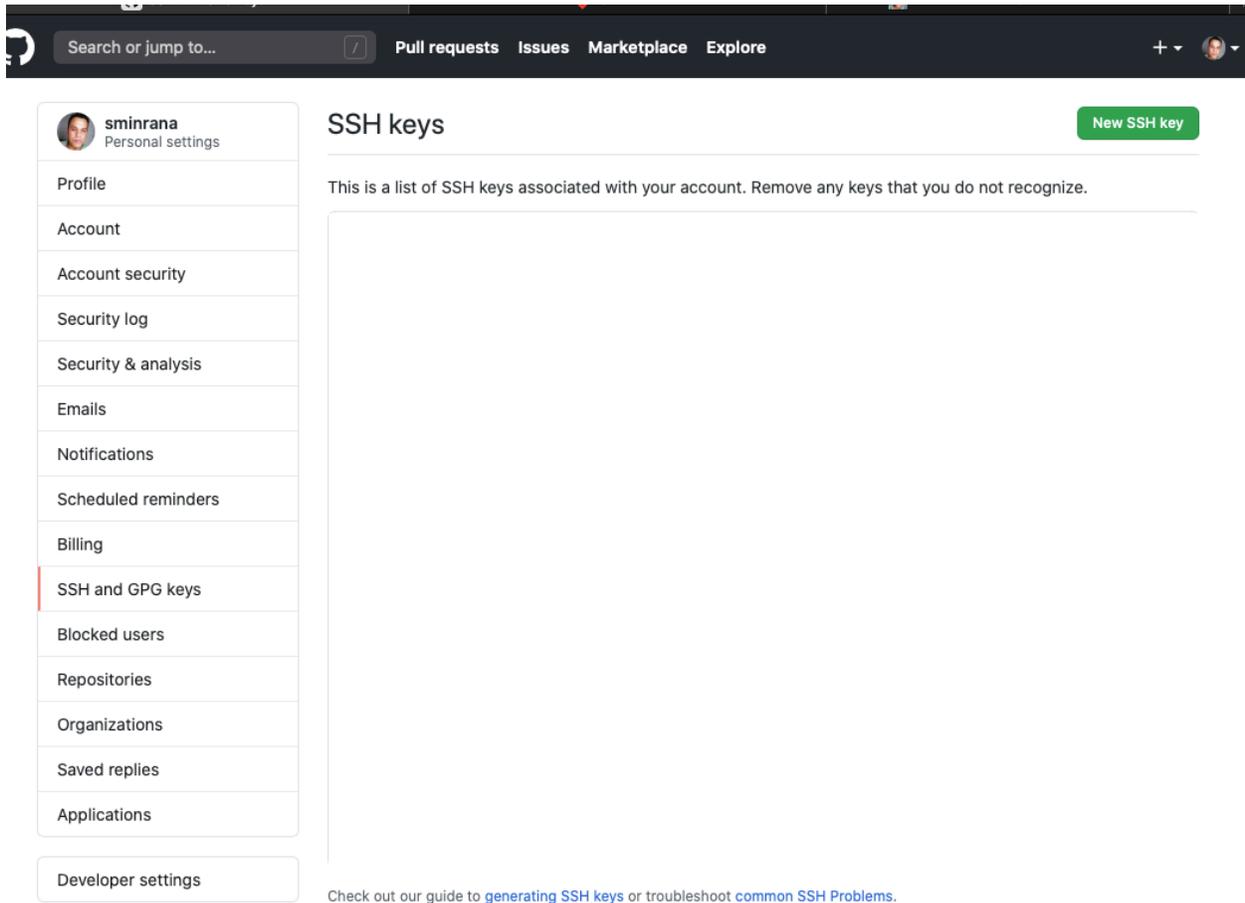
On Linux or Mac run terminal and run command

```
ssh-keygen -t rsa -C "your_email@example.com"
```

**Give a new file path when it says "Enter file in which to save the key". The press enter.**

It will create two keys in your .ssh directory one is private which has no extension and one is public which .pub extension. Now open .pub key in any text editor and select all and copy the content in your clipboard.

Go here <https://github.com/settings/keys> and click the button on the top-right New SSH Key.



The screenshot shows the GitHub settings interface for a user named 'sminrana'. On the left is a navigation sidebar with options like Profile, Account, Account security, Security log, Security & analysis, Emails, Notifications, Scheduled reminders, Billing, SSH and GPG keys (highlighted), Blocked users, Repositories, Organizations, Saved replies, Applications, and Developer settings. The main content area is titled 'SSH keys' and features a 'New SSH key' button in the top right. Below the title, a message states: 'This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.' The list area is currently empty. At the bottom of this section, there is a link to a guide: 'Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH Problems](#).'

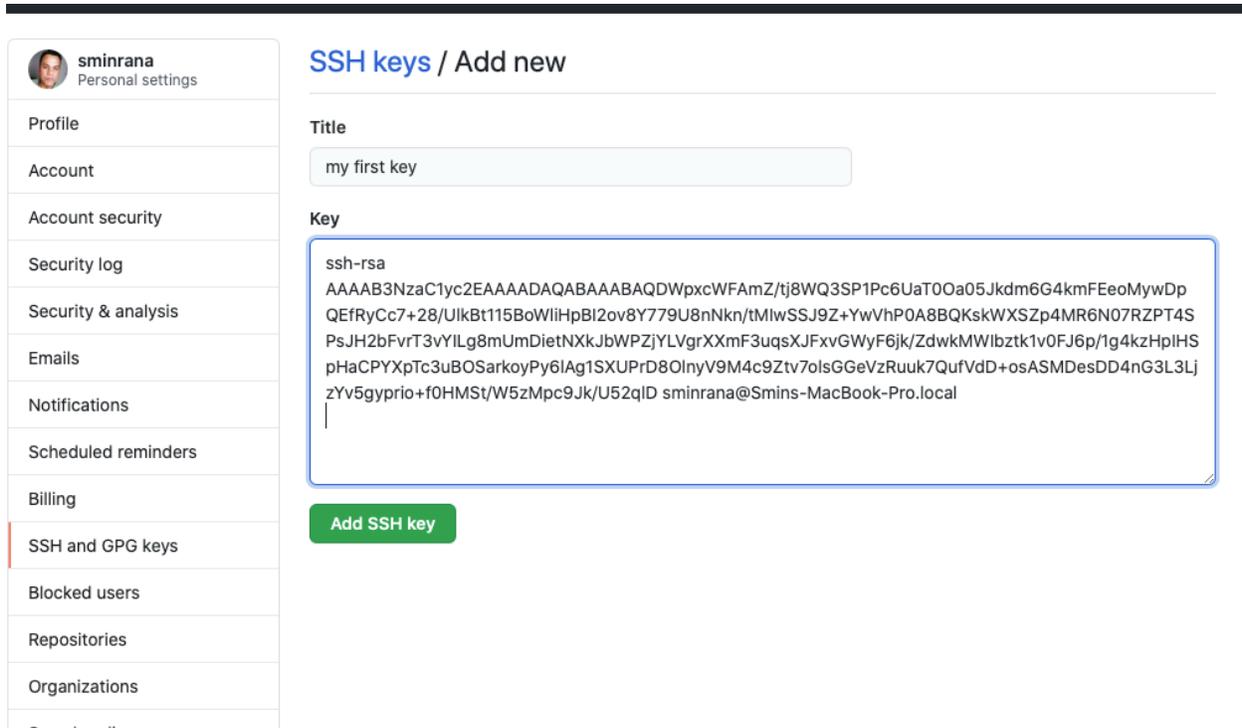
## GPG keys

New GPG key

There are no GPG keys associated with your account.

Learn how to [generate a GPG key and add it to your account](#).

Now paste your the content you copied from .pub file and paste in the key. Also give a nice title.



The screenshot shows the GitHub 'SSH keys / Add new' page. On the left is a sidebar with the user's profile 'sminrana' and a list of settings: Profile, Account, Account security, Security log, Security & analysis, Emails, Notifications, Scheduled reminders, Billing, SSH and GPG keys (highlighted), Blocked users, Repositories, and Organizations. The main content area is titled 'SSH keys / Add new'. It has a 'Title' field containing 'my first key' and a 'Key' text area containing an SSH public key. Below the key field is a green 'Add SSH key' button.

**SSH keys / Add new**

**Title**  
my first key

**Key**

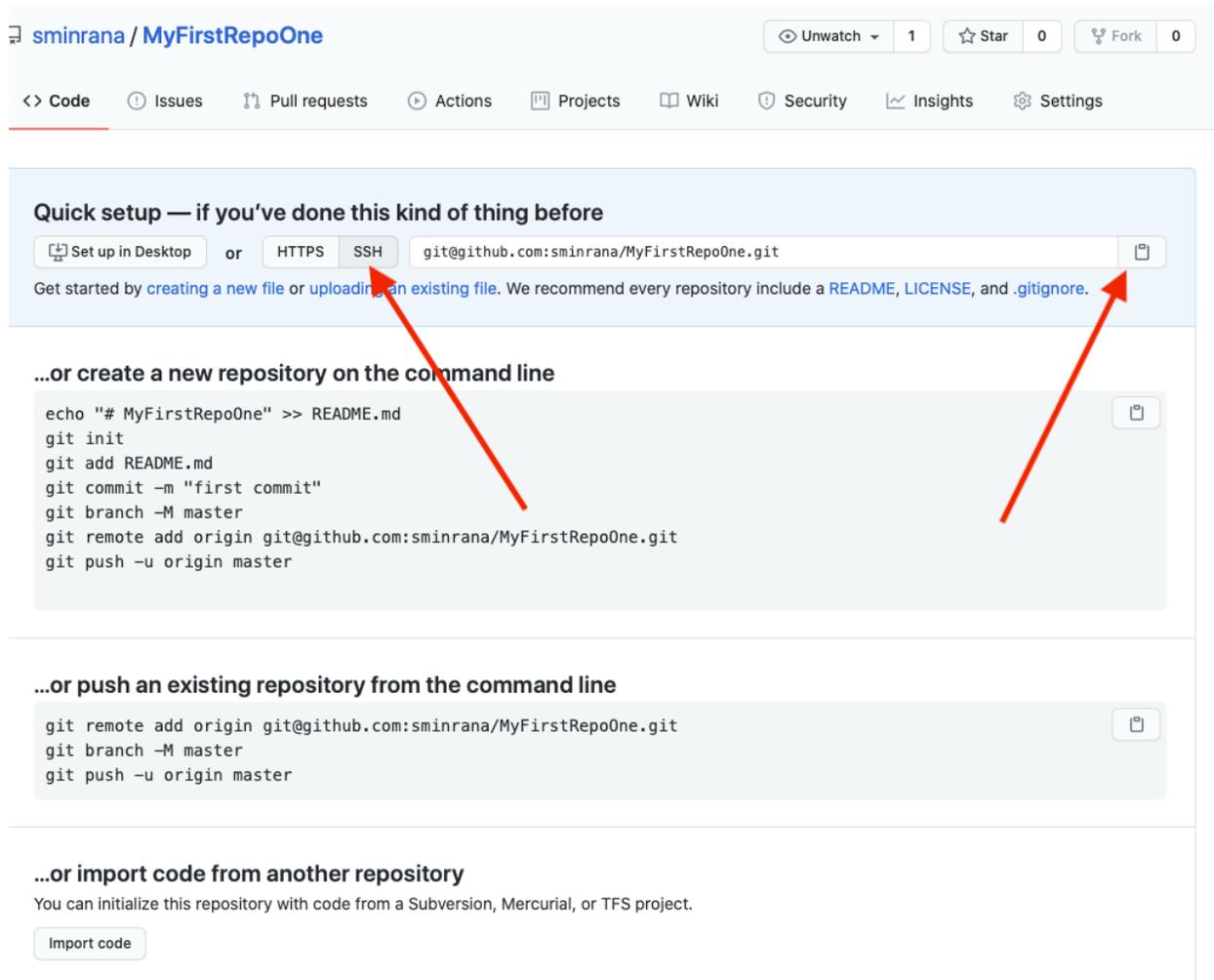
```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDWpxcWFAmZ/tj8WQ3SP1Pc6UaT0Oa05Jkdm6G4kmFEoMywDp
QEfRyCc7+28/UikBt115BoWiiHpBI2ov8Y779U8nNkn/tMlwSSJ9Z+YwVhP0A8BQKskWXSZp4MR6N07RZPT4S
PsJH2bFvrT3vYILg8mUmDietNXkjbWPZjYLVgrXXmF3uqsXJFvxvGWyF6jk/ZdwmMWlbztk1v0FJ6p/1g4kzHplHS
pHaCPYXpTc3uBOSarkopy6IAg1SXUPrD8OlnyV9M4c9Ztv7olsGGeVzRuuk7QufVdD+osASMDesDD4nG3L3Lj
zYv5gyprio+f0HMSt/W5zMpc9Jk/U52qlD sminrana@Smins-MacBook-Pro.local
```

**Add SSH key**

Finally click the button Add SSH Key to finish the process.

# Clone Your Empty Repository

Now open your repository page and copy the repository url, make sure you select SSH.



The screenshot shows the GitHub repository page for 'sminrana / MyFirstRepoOne'. At the top, there are navigation tabs for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the repository name, there are buttons for Unwatch (1), Star (0), and Fork (0). The main content area is titled 'Quick setup — if you've done this kind of thing before' and offers three options: 'Set up in Desktop', 'HTTPS', and 'SSH'. The 'SSH' option is selected, and the repository URL 'git@github.com:sminrana/MyFirstRepoOne.git' is displayed in a text box with a copy icon. Below this, there are instructions to get started by creating a new file or uploading an existing file. Two red arrows point from the text 'make sure you select SSH' to the 'SSH' button and the repository URL text box. Below the quick setup section, there are three sections for creating or pushing a repository from the command line, each with a code block and a copy icon. The first section is titled '...or create a new repository on the command line' and contains the following commands: 

```
echo "# MyFirstRepoOne" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M master
git remote add origin git@github.com:sminrana/MyFirstRepoOne.git
git push -u origin master
```

The second section is titled '...or push an existing repository from the command line' and contains the following commands: 

```
git remote add origin git@github.com:sminrana/MyFirstRepoOne.git
git branch -M master
git push -u origin master
```

The third section is titled '...or import code from another repository' and contains the text 'You can initialize this repository with code from a Subversion, Mercurial, or TFS project.' and an 'Import code' button.

**Quick setup — if you've done this kind of thing before**

Set up in Desktop or **SSH** git@github.com:sminrana/MyFirstRepoOne.git

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

**...or create a new repository on the command line**

```
echo "# MyFirstRepoOne" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M master
git remote add origin git@github.com:sminrana/MyFirstRepoOne.git
git push -u origin master
```

**...or push an existing repository from the command line**

```
git remote add origin git@github.com:sminrana/MyFirstRepoOne.git
git branch -M master
git push -u origin master
```

**...or import code from another repository**

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

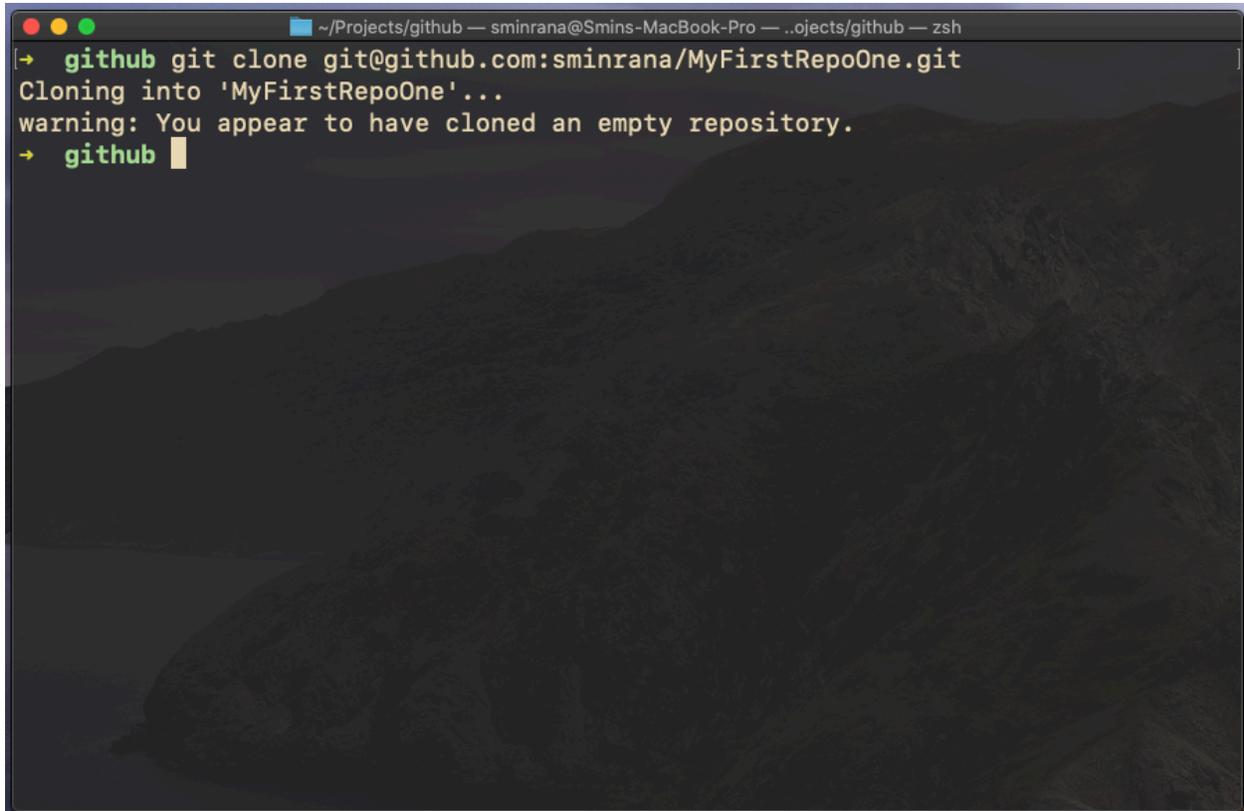
Import code

Now open your terminal and run command

```
git clone git@your_git_url
```

in my case it is

```
git clone git@github.com:sminrana/MyFirstRepoOne.git
```

A screenshot of a terminal window on a Mac. The window title is "~/Projects/github — sminrana@Smins-MacBook-Pro — ..jects/github — zsh". The terminal shows the command `github git clone git@github.com:sminrana/MyFirstRepoOne.git` being executed. The output is `Cloning into 'MyFirstRepoOne'...` followed by `warning: You appear to have cloned an empty repository.` and the prompt `→ github` with a cursor.

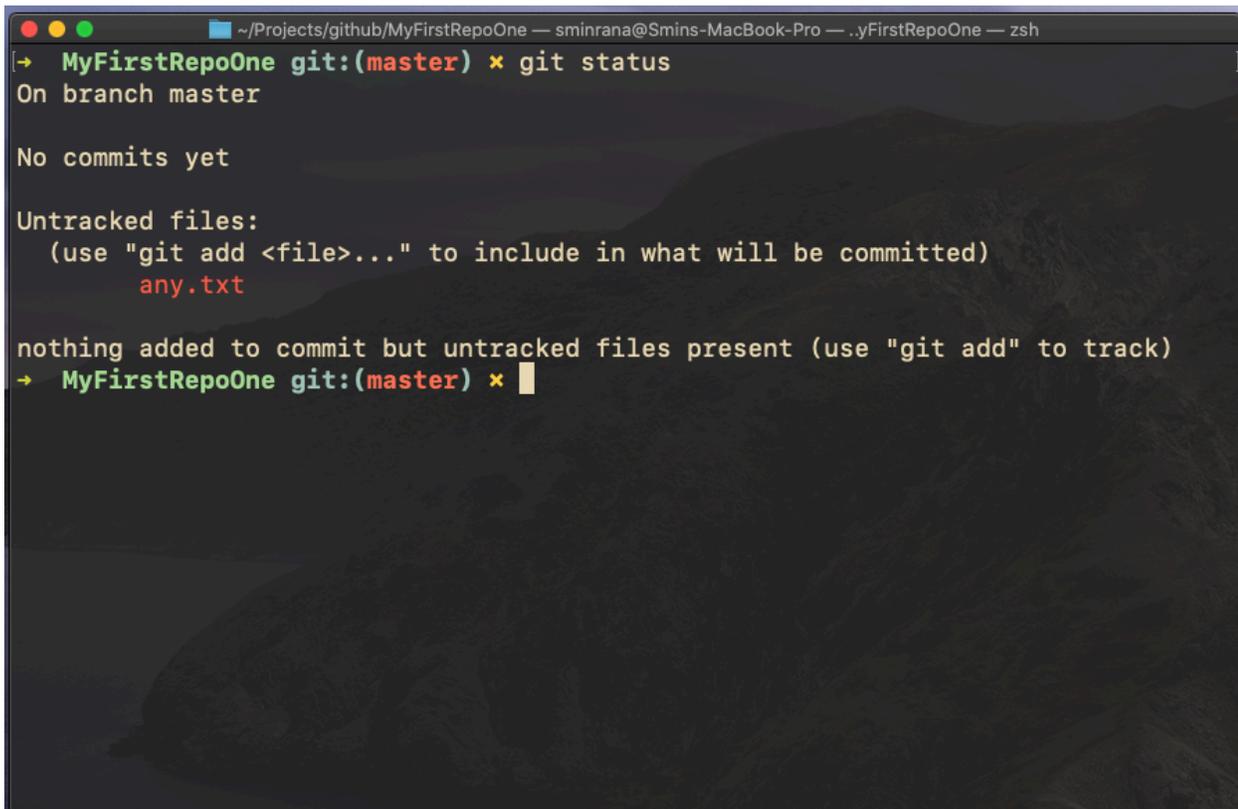
```
~/Projects/github — sminrana@Smins-MacBook-Pro — ..jects/github — zsh
[→ github git clone git@github.com:sminrana/MyFirstRepoOne.git
Cloning into 'MyFirstRepoOne'...
warning: You appear to have cloned an empty repository.
→ github
```

Now create a txt file in the MyFirstRepoOne directory and add any text.

Run this command to know your git status

`git status`

Mine looks like this, my txt file name is any.txt.

A terminal window screenshot showing the output of the 'git status' command. The window title is '~/Projects/github/MyFirstRepoOne — sminrana@Smins-MacBook-Pro — .yFirstRepoOne — zsh'. The prompt is 'MyFirstRepoOne git:(master) x git status'. The output shows 'On branch master', 'No commits yet', and 'Untracked files: (use "git add <file>..." to include in what will be committed) any.txt'. A final line says 'nothing added to commit but untracked files present (use "git add" to track)'. The prompt is now 'MyFirstRepoOne git:(master) x' with a cursor.

```
→ MyFirstRepoOne git:(master) x git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
       any.txt

nothing added to commit but untracked files present (use "git add" to track)
→ MyFirstRepoOne git:(master) x
```

Add this txt file in git by running this command

`git add any.txt`

Then run git commit command

`git commit -m "My first git commit message"`

Finally push your changes to the GitHub server

`git push origin master`

```
~/Projects/github/MyFirstRepoOne — sminrana@Smins-MacBook-Pro — .yFirstRepoOne — zsh
→ MyFirstRepoOne git:(master) ✘ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  any.txt

nothing added to commit but untracked files present (use "git add" to track)
→ MyFirstRepoOne git:(master) ✘ git add any.txt
→ MyFirstRepoOne git:(master) ✘ git commit -m "My first git commit message"
[master (root-commit) 2cc4d32] My first git commit message
1 file changed, 1 insertion(+)
 create mode 100644 any.txt
→ MyFirstRepoOne git:(master) git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 237 bytes | 237.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:sminrana/MyFirstRepoOne.git
 * [new branch]      master -> master
→ MyFirstRepoOne git:(master) █
```