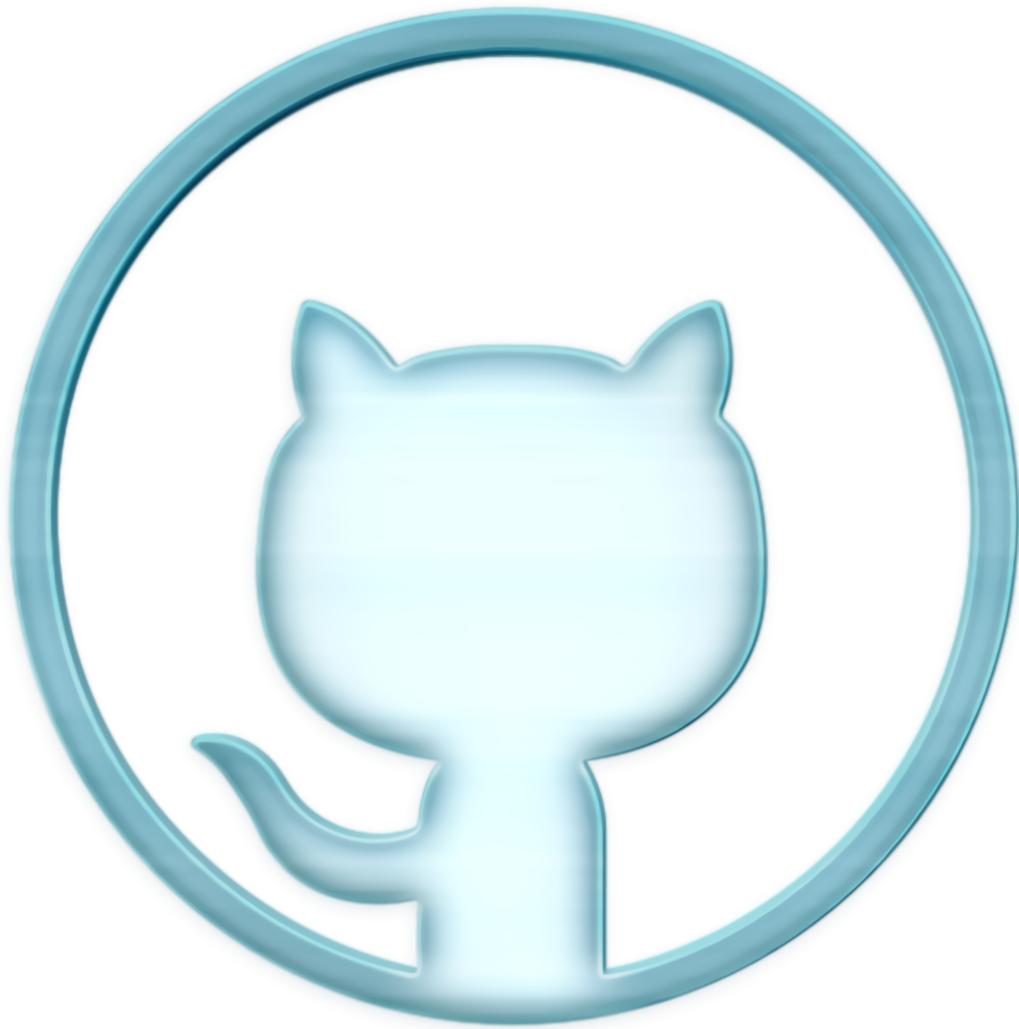


# OFFICIAL USER GUIDE FOR GIT & GITHUB



SHITAL BK, NEW YORK CITY COLLEGE OF TECHNOLOGY

NOVEMBER, 2020

<b>CONTENTS</b>	<b>3</b>
<b>Introduction to Git and GitHub</b>	<b>3</b>
History of Git	3
Purpose	3
Git Online Help	3
<b>Installation of Git</b>	<b>3</b>
Installing on Linux	3
Installing on Mac	4
Installing on Windows	4
<b>Getting Started with GitHub</b>	<b>4</b>
<b>Git Configuration</b>	<b>6</b>
<b>Creating a Git Repository</b>	<b>6</b>
<b>Adding Files to your Git Repository and Checking Git Status</b>	<b>7</b>
<b>Synchronizing your local Git repository with GitHub</b>	<b>11</b>
<b>Summary</b>	<b>14</b>
<b>References</b>	<b>15</b>

# CONTENTS

## Introduction to Git and GitHub

Writing a document is an easy task but maintaining the documents up to its final version is a complicated task. Simply, if you are working on huge projects or tasks, the workflow gets more complex and sometimes the files and documents are impossible to be tracked and managed. To solve this issue, the tool called Git was created.

## History of Git

Git was created by Linus Torvalds in 2005 for development of the Linux kernel, with other kernel developers contributing to its initial development. The development of Git began on 3rd April 2005 and Torvalds announced the project on 6th April and became self-hosting the next day. Finally, on 16th June, Git managed the kernel 2.6.12 release.

## Purpose

Git is a distributed version control system for tracking changes in a source code during software development. It is designed for coordinating work among programmers, but it can be used to track changes in any set of files. Git has become a mandatory tool these days for developers and any successful project in software development is incomplete without the use of git.

GitHub is an online platform that allows you to synchronize your local git repository onto the web. Git and GitHub are separate entities but can be linked together.

## Git Online Help

The Git community is very helpful and very active. If you have questions or a problem related to Git, you should be able to get help from the link given below:

<https://git-scm.com/community>.

By visiting the above link, you can get access to the Git community mailing list, report bugs related to Git, access to Git newsletter and also contribute to Git.

## Installation of Git

### Installing on Linux

Check whether Git is already installed by typing the command below:

**git --version**

If git is already installed, it shows you the version number and if git is not installed, it says the command is not found. If you're using RedHat, Fedora or CentOS, type the following command given below for installation:

**sudo yum -y install git**

For other newer Linux distributions, following command can be used shown below:

**sudo dnf -y install git**

For Debian-based distribution, such as Ubuntu, use the command shown below:

**sudo apt-get install git**

Finally, verify that Git is installed correctly

**git --version**

## Installing on Mac

To install Git on Mac, goto the link, [git-scm.com/download/mac](http://git-scm.com/download/mac) and download and install the Git package. You can simply use the command in your terminal as shown below:

**brew install git**

```
[Shitals-MacBook-Pro:~ shitalbk$ brew install git]
```

After installation, verify that Git is installed correctly using command below:

**git --version**

```
[Shitals-MacBook-Pro:~ shitalbk$ git --version  
git version 2.22.0]
```

The version number in the screenshot shows that Git was successfully installed.

## Installing on Windows

To install Git on Windows, goto the link, [gitforwindows.org](http://gitforwindows.org) and simply download and install the Git-bash package.

## Getting Started with GitHub

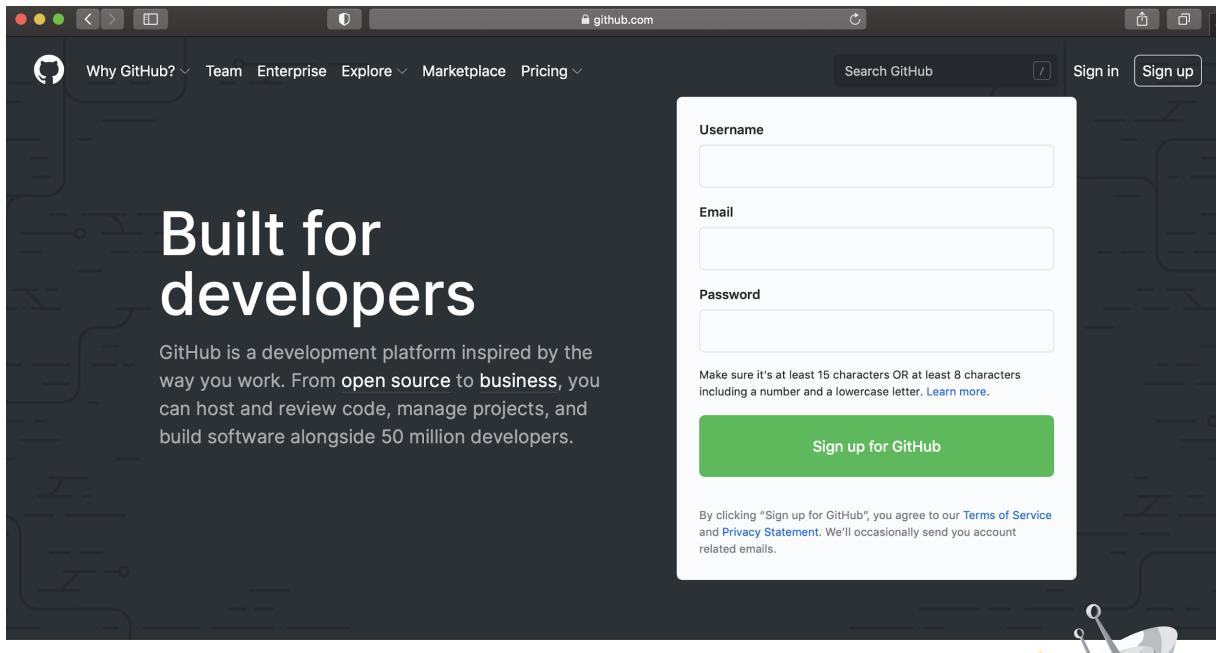
To get started with GitHub, create a GitHub account by visiting the website

<https://github.com/>. GitHub is completely free so you don't have to worry about the monthly subscription for creating an account with GitHub.

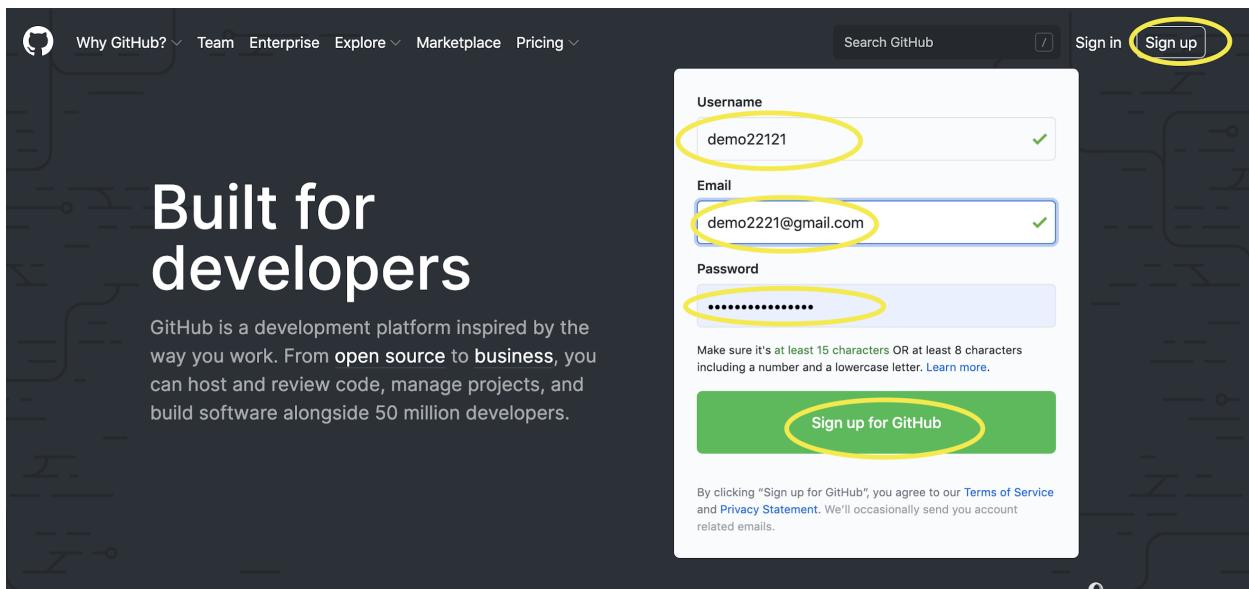
If you are new to GitHub click Sign Up and complete the required processes to create the account and if you are already signed up in GitHub simply login using your username and password as shown below in the screenshot.

### Steps for Sign Up:

1. Goto [github.com](https://github.com) in your browser as shown below:



2. Click the Sign up button and enter the Username, Email and Password as shown below:



3. After the Sign up is completed, GitHub sends the verification email in the email address that you've provided. Simply, login to your email address and click the verification link once it is received in your email and you should be all set to login and start using GitHub.

4. In this tutorial, I've already signed up for GitHub so I'll be simply using my username and password to login after visiting the website [github.com](https://github.com) which will redirect me to my GitHub account.

## Git Configuration

Configuring Git is very simple which allows you to set your email address and your username. You only need to type two lines of command to configure Git and link it to GitHub account. By typing the commands you will be able to replace USERNAME with the username that was created in your GitHub and replace EMAIL with your email address.

Go back to your terminal if you are using MAC, Command Prompt if you are using WINDOWS and type the command given below:

```
git config --global user.name "USERNAME"
```

```
git config --global user.email "EMAIL"
```

```
[Shitals-MacBook-Pro:~ shitalbk$ git config --global user.name "Shital"  
[Shitals-MacBook-Pro:~ shitalbk$ git config --global user.email "shitalbk36@gmail.com"
```

## Creating a Git Repository

Creating a Git repository is like creating a folder for git. If you already have a project directory, goto your terminal and run the command:

```
git init
```

```
[Shitals-MacBook-Pro:~ shitalbk$ git init  
Reinitialized existing Git repository in /Users/shitalbk/.git/
```

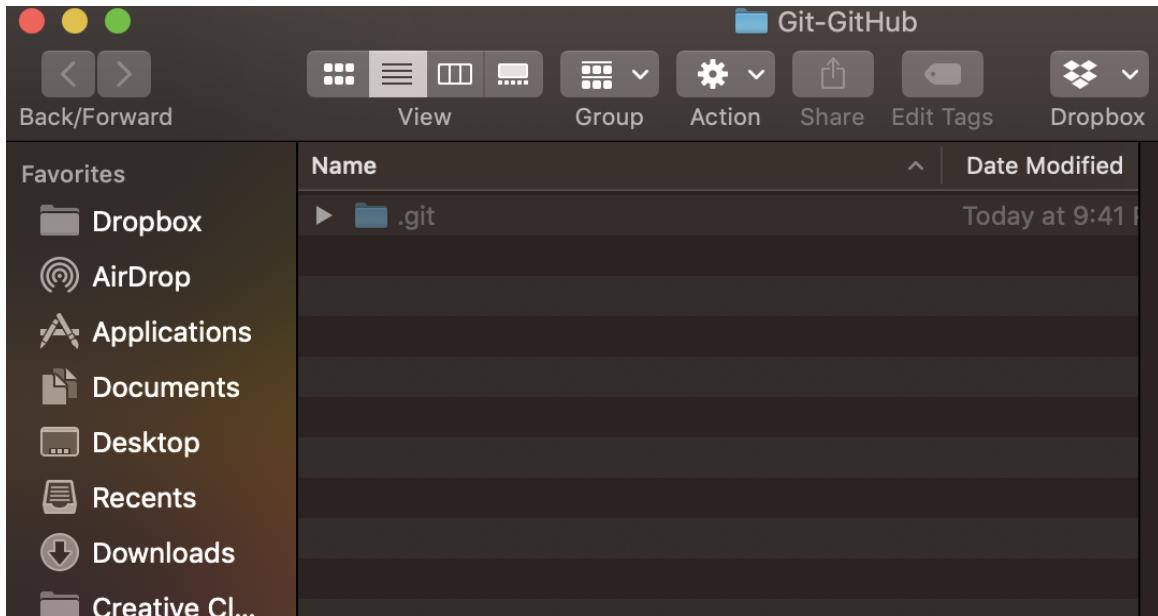
If you have a folder for your project then goto your project directory and type the command as shown below:

```
cd folder name
```

```
git init
```

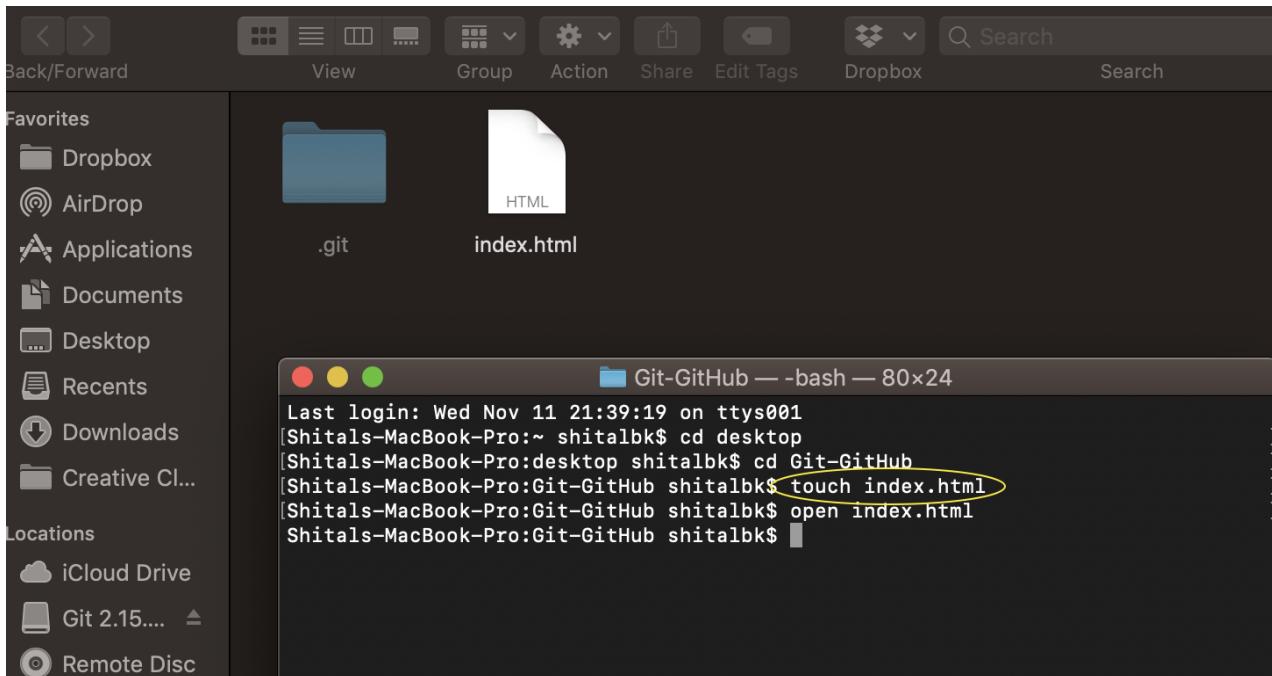
```
[Shitals-MacBook-Pro:desktop shitalbk$ cd Git-GitHub/  
[Shitals-MacBook-Pro:Git-GitHub shitalbk$ git init  
Initialized empty Git repository in /Users/shitalbk/Desktop/Git-GitHub/.git/
```

After creating the git repo inside the folder, you should see the .git folder inside the Git-GitHub folder as shown in the screenshot. The .git folder confirms that the repository was initialized after running the command.



## Adding Files to your Git Repository and Checking Git Status

To add files in your Git repo, first of all create a file if you don't have any files. To create a file using terminal, goto terminal and type the command as shown below:  
**touch index.html**

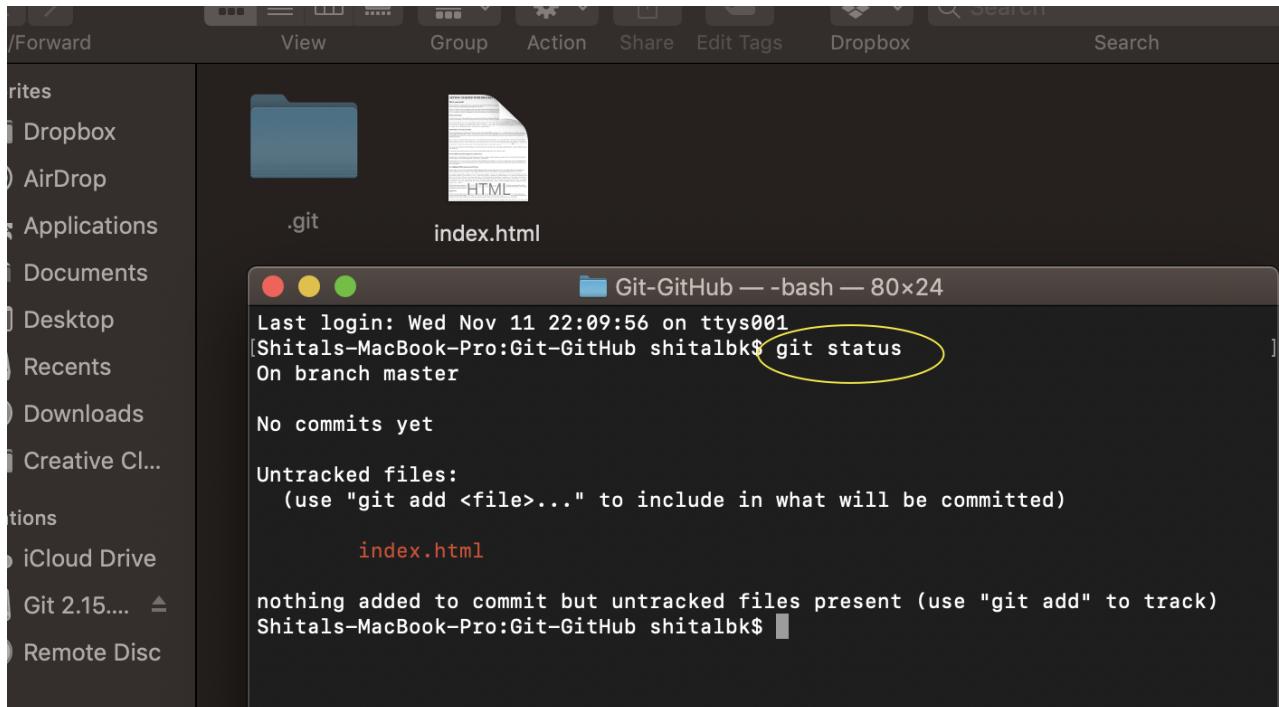


**The index.html file is created as shown in the folder directory. You can add some html code in your file and then save it.**

Once the file is saved after adding the code, add the file to your Git Repository as shown below.

The index.html file is in the Git-GitHub folder but it is not added to the git repo. Git only adds the file after hitting the command.

**git add**



We are checking our repository by hitting the command

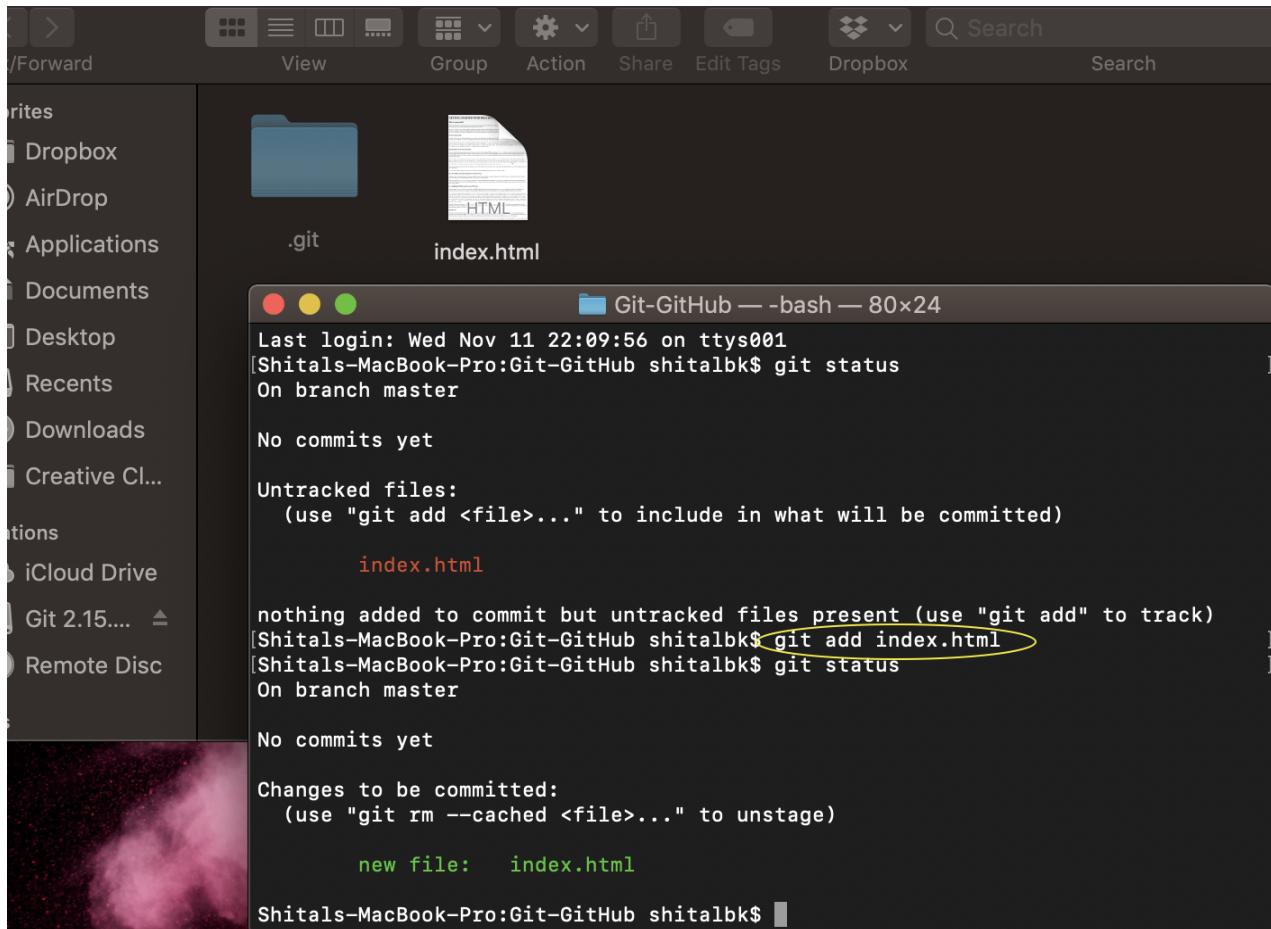
**git status**

This command allows us to track the files in the repository, as a result, the above screenshot says the file was untracked which means the file was not added in the repository and Git wants us to add the file to our Git Repository.

To do so, we can use the command **git add** which adds the file in our repository.

**git add index.html**

After entering this command you can check your git status again and now it doesn't say any files are untracked because the file was already added. Git asks you whether you want the file to be removed or to be committed. Committing inside Git is like saving your changes for the file that you have added. Similarly, once you add the file in the git, you can either remove it or commit it.



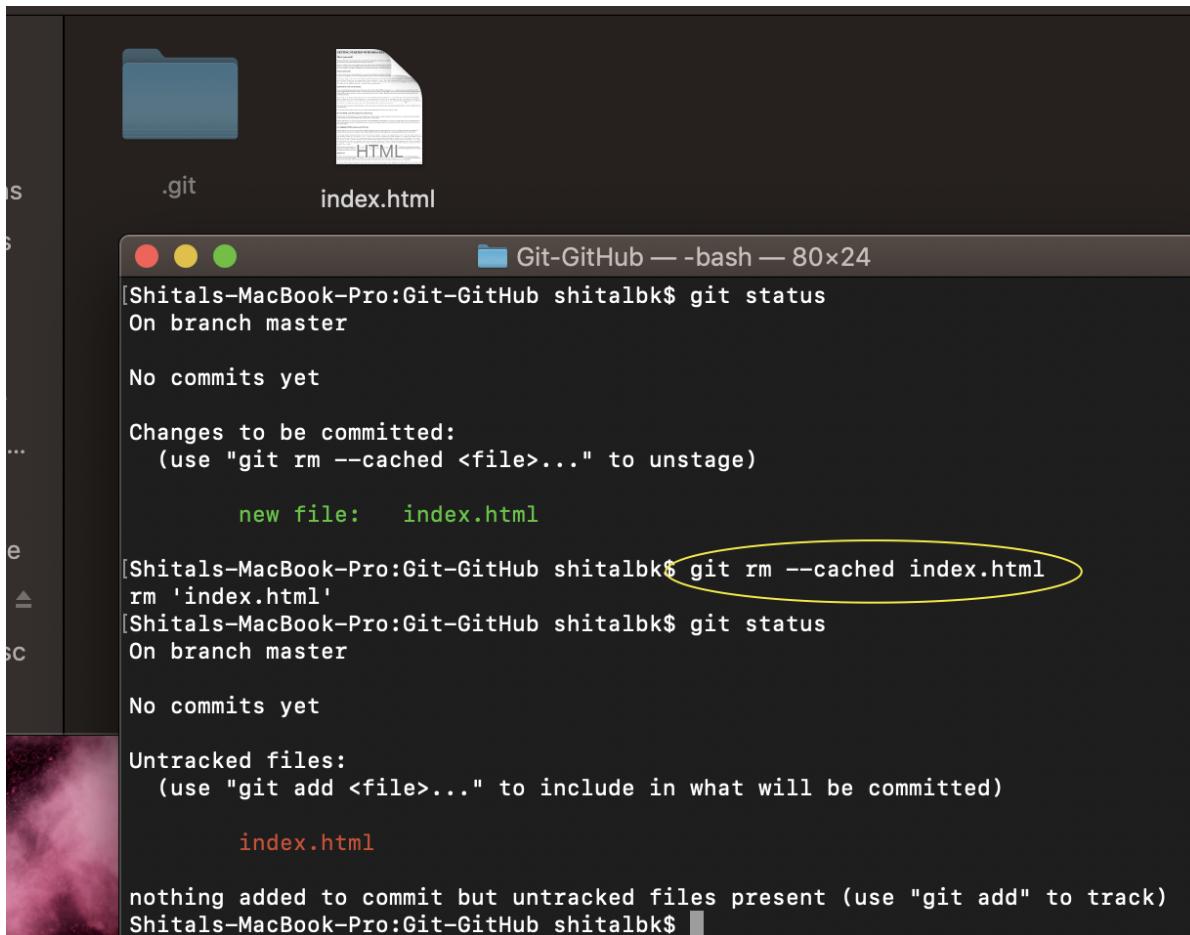
To remove the added file in Git use the command given below:

**git rm --cached index.html**

This command removes the added file from your git repository and you can check your git status again by hitting the command **git status**.

After you remove the added file, git no longer has your file. As a result, git goes to the previous state where no files are added which shows the result as an untracked file and further it says to use "git add <file>..." to add your file again in git as shown below in the screenshot.

Since we have removed our file named **index.html**, we will add our file again and learn how to commit the added file with a message.



```
[Shitals-MacBook-Pro:Git-GitHub shitalbk$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

    new file:   index.html

[Shitals-MacBook-Pro:Git-GitHub shitalbk$ git rm --cached index.html
rm 'index.html'
[Shitals-MacBook-Pro:Git-GitHub shitalbk$ git status
On branch master

No commits yet

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

    index.html

nothing added to commit but untracked files present (use "git add" to track)
Shitals-MacBook-Pro:Git-GitHub shitalbk$ ]
```

To add your file again, use **git add index.html** and to check the git status use **git status**.

Now our file is ready to be committed. To commit, use the command shown below:  
**git commit -m “message”**

The command saves your changes to the file and you can add any message that includes inside the quotes as shown below in the screenshot.

Now we have saved our changes using the command **git commit**.

After the commit is made, it shows the changes and insertions made with the file as shown below.

To add all the files at once, use  
**git add .**

```

Git-GitHub — bash — 80x24
No commits yet

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

    index.html

nothing added to commit but untracked files present (use "git add" to track)
[Shitals-MacBook-Pro:Git-GitHub shitalbk$ git add index.html
[Shitals-MacBook-Pro:Git-GitHub shitalbk$ git status
On branch master

No commits yet

Changes to be committed:
(use "git rm --cached <file>..." to unstage)

    new file:   index.html

[Shitals-MacBook-Pro:Git-GitHub shitalbk$ git commit -m "Adding index file"
[master (root-commit) 7d902d5] Adding index file
 1 file changed, 207 insertions(+)
 create mode 100644 index.html
Shitals-MacBook-Pro:Git-GitHub shitalbk$ 

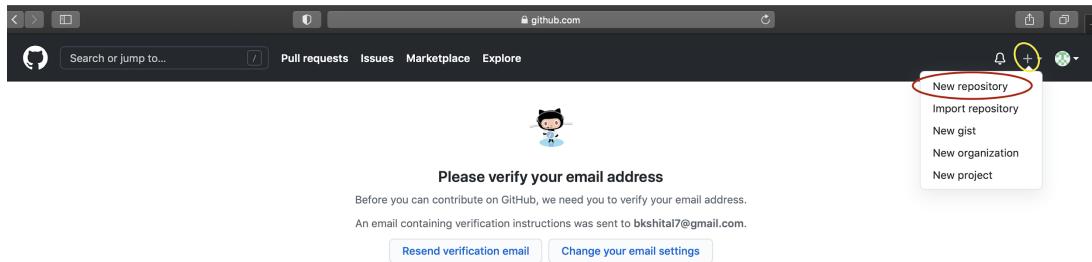
```

## Synchronizing your local Git repository with GitHub

To synchronize your local repository with GitHub, proceed as follows:

Go to the [GitHub.com](https://github.com) web site and make sure you are logged in.

- In the top right corner, you should find a + sign. Click the + sign highlighted in yellow and then click New Repository.



**GitHub**

Product	Platform	Support	Company
Features	Developer API	Help	About
Security	Partners	Community Forum	Blog
Team	Atom	Professional Services	Careers
Enterprise	Electron	Learning Lab	Press
Customer stories	GitHub Desktop	Certifications	Social Impact
The README Project		Status	Shop
Pricing		Contact GitHub	
Resources			
Roadmap			

- Enter the name of your repository (Git-GitHub), you can enter any name of your choice.
- Add a description of your repository — it is optional so I'm not adding right now.
- You have the option to make your repository public or private on the web, I'm making it public so anybody can access my repository. You can make it private if you want nobody to access your repository.
- Do not select the option to “Initialize this repository with a README”.
- Click the button to create the repository.

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-guide](#)?

**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](#).

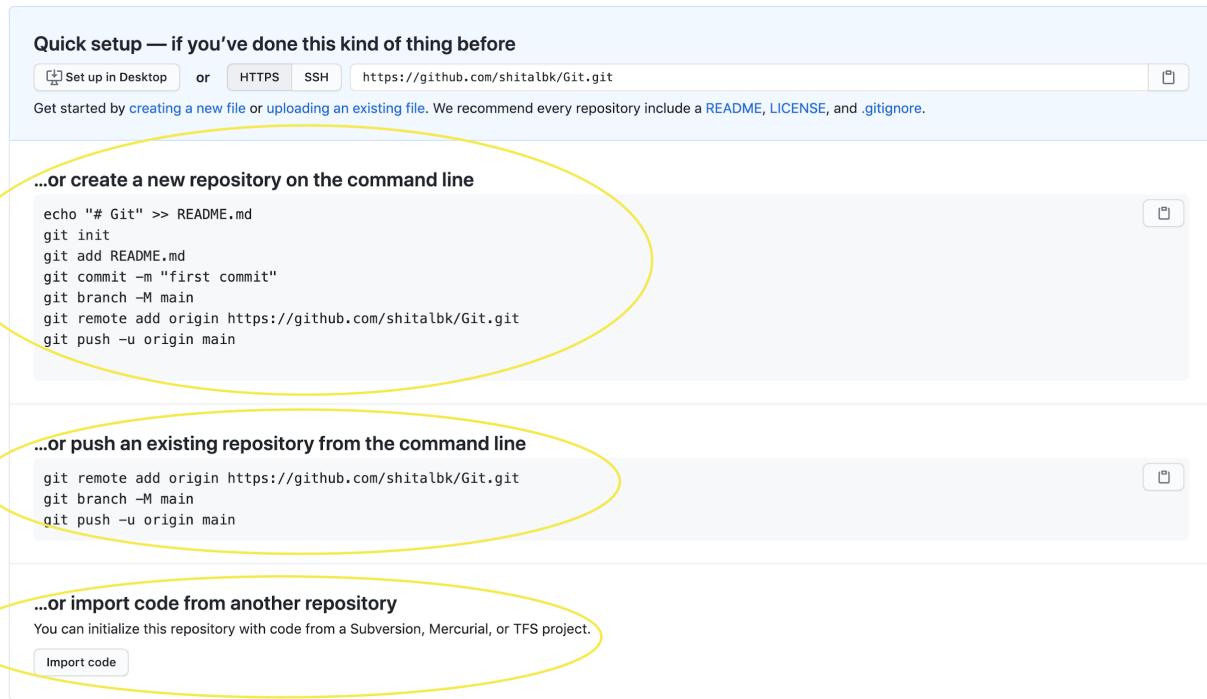
**Add .gitignore**  
Choose which files not to track from a list of templates. [Learn more](#).

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

**Create repository**

These steps have created an empty repository on GitHub, you now need to synchronize your local repository with this.

After creating the repository, you should see something like this shown below in the screenshot.



From the above screenshot, we can see three ways to synchronize the repository from Git to GitHub which is from local environment “terminal/computer” to online platform “Github.com”.

- **The first method is for creating a new repository on the command line.**
- **The second method is pushing an existing repository on the command line.**
- **The last one is importing code from another repository.**

The first method should be used when you have no repository created in your local environment using git and terminal and the highlighted commands can be used.

The second method should be used when you have already created a repository in your local environment using git and terminal. As a result, we should be using the highlighted three lines of code in the command line.

Finally, the last one is used when you are importing code from another repository and simply click the button Import code to use the third method.

In our case, we have already created a repository in our local environment using git and terminal so we will be using the second method to push our repository in the online version which is GitHub. To do this go back to terminal in your computer and type the following command shown below:

```
git remote add origin https://github.com/shitalbk/Git.git
git branch -M main
git push -u origin main
```

```
Shitals-MacBook-Pro:Git-GitHub shitalbk$ git remote add origin https://github.com/shitalbk/Git.git
Shitals-MacBook-Pro:Git-GitHub shitalbk$ git branch -M main
Shitals-MacBook-Pro:Git-GitHub shitalbk$ git push -u origin main
[ Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 3.82 KiB | 3.82 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/shitalbk/Git.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
```

After the command is done in the terminal the above result should be seen in the terminal's screenshot.

Go back to your GitHub account and refresh the page, you should now see your local repository synchronized into your github account.

The screenshot shows a GitHub repository page for 'shitalbk / Git'. The repository is private. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the navigation, there are buttons for Go to file, Add file, and Code. The main content area displays a single commit:

- Branch: main
- Commits: 1 branch
- Tags: 0 tags
- Author: shitalbk Adding file
- Date: 41d0827 7 minutes ago
- Commits: 1 commits
- File: index.html
- Message: Adding file
- Date: 7 minutes ago

At the bottom, there is a button to Add a README.

You can add a Readme file from your github or git.

## Summary

This tutorial gives the basics of how to use Git and GitHub. There are many other commands and tasks that can be done using Git and GitHub which are not covered in this tutorial since it covers only the basics. To learn in depth you can visit the site

<https://git-scm.com/book/en/v2> which covers additional information about Git. I hope you enjoyed my tutorial for learning the basics of Git and GitHub.

## References

Bonner, A. (2019, September 03). Getting started with Git and GitHub: The complete beginner's guide. Retrieved October 28, 2020, from

<https://towardsdatascience.com/getting-started-with-git-and-github-6fcd0f2d4ac6>

(she/her), H., (He/Him), R., & (She/Her), N. (n.d.). An Intro to Git and GitHub for Beginners (Tutorial). Retrieved October 28, 2020, from

<https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners>

MV, T. (2019, November 07). The beginner's guide to Git & GitHub. Retrieved October 28, 2020, from <https://www.freecodecamp.org/news/the-beginners-guide-to-git-github/>