GIT

Name: Srima Shetty

TABLE OF CONTENTS

INTRODUCTION

AIM OF PROJECT:

To use Git Bash shell which is part of Git for Windows to the Windows Terminal to use various Git commands like init, status, pull, push etc. to put the project on Github.

GIT:

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. Git is software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows (thousands of parallel branches running on different systems).

VERSION CONTROL:

Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later. Managing the codebase is easier with the help of version control.

GITHUB:

Github is a version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.

GIT COMMANDS:

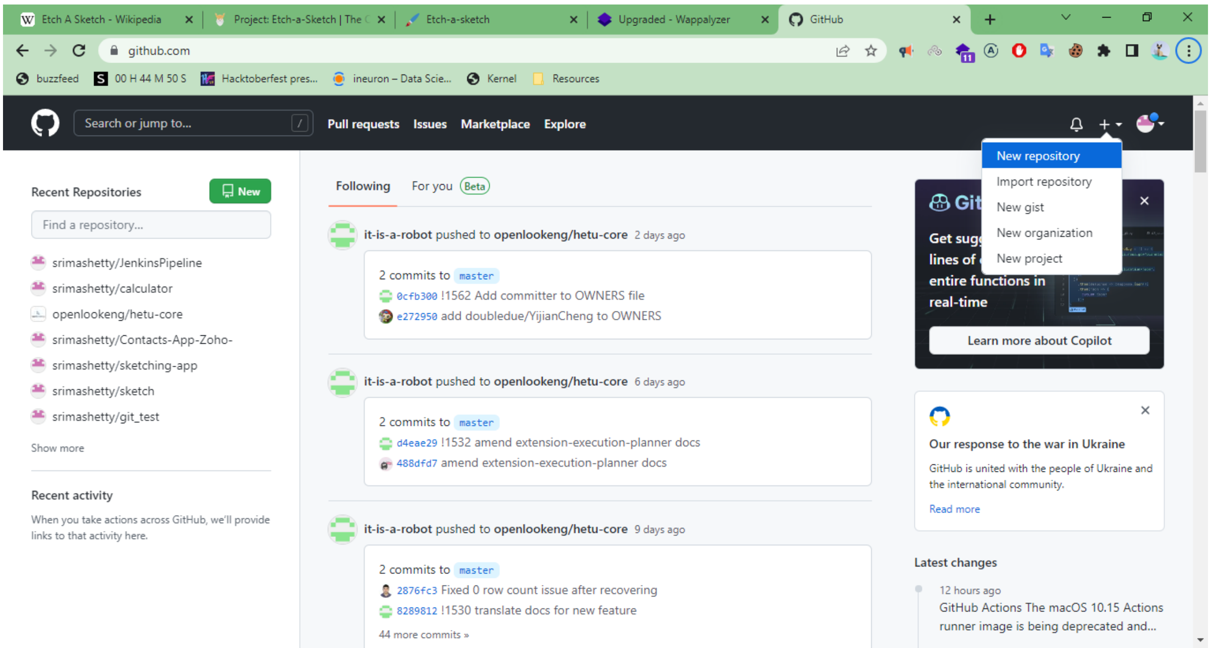
- **Config:** This command configures the user. The Git config command is the first and necessary command used on the Git command line.
- **Init:** This command is used to create a local repository.
- **Clone:** This command is used to make a copy of a repository from an existing URL.

- **Add:** This command is used to add one or more files to staging (Index) area.
- **Commit:** The git commit command captures a snapshot of the project's currently staged changes.
- **Status:** The status command is used to display the state of the working directory and the staging area.
- **Push:** It is used to upload local repository content to a remote repository. Pushing is an act of transfer commits from your local repository to a remote repo.
- **Pull:** Pull command is used to receive data from GitHub. It fetches and merges changes on the remote server to your working directory.

GIT AND GIT REPO

USING GIT BASH:

Step 1: Create a remote repository on Github. Go to the plus sign on the top right corner next to the profile icon, and click new Repository.



When you take actions across GitHub, we'll provide links to that activity here.

it-is-a-robot pushed to openlookeng/hetu-core 2 days ago

2 commits to master
0cfb300 !1562 Add committer to OWNERS file
e272950 add doubledue/YijianCheng to OWNERS

it-is-a-robot pushed to openlookeng/hetu-core 6 days ago

2 commits to master
d4eae29 !1532 amend extension-execution-planner docs
488dfd7 amend extension-execution-planner docs

it-is-a-robot pushed to openlookeng/hetu-core 9 days ago

2 commits to master
2876fc3 Fixed 0 row count issue after recovering
8289812 !1530 translate docs for new feature
44 more commits »

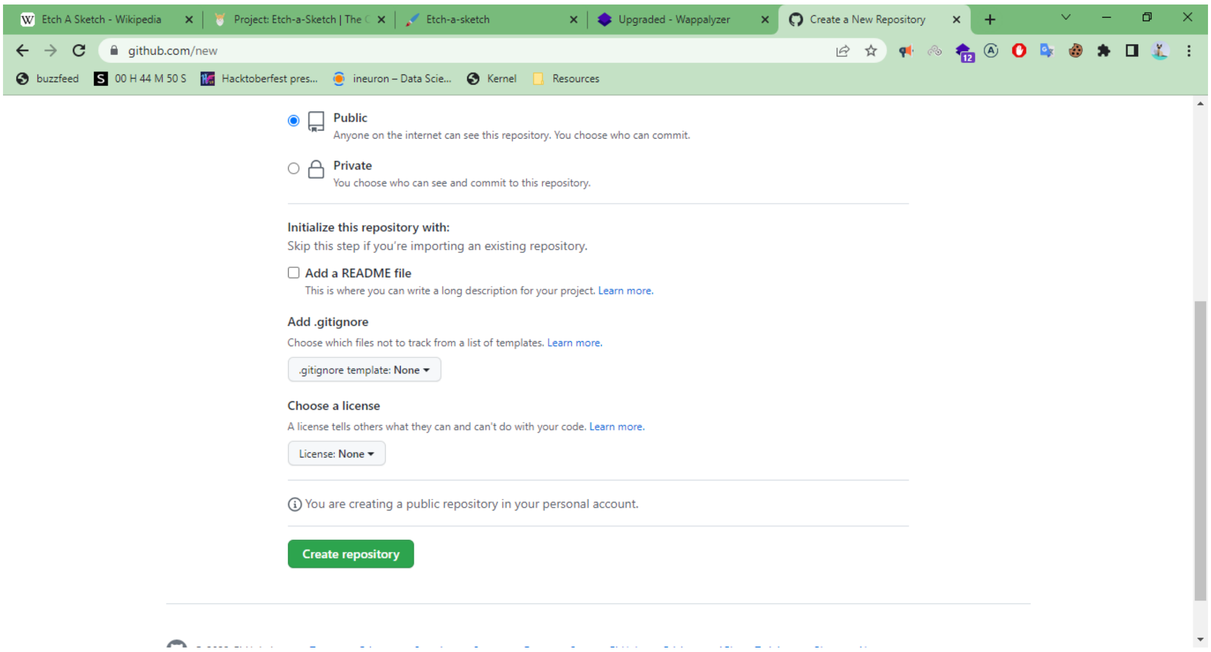
Get sug lines of entire functions in real-time

Our response to the war in Ukraine

GitHub is united with the people of Ukraine and the international community.

12 hours ago
GitHub Actions The macOS 10.15 Actions runner image is being deprecated and...

Step 2: Enter the repository name, specify whether it is public or a private repository. Add readme, gitignore and licence if necessary and click on Create Repository.



● 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.

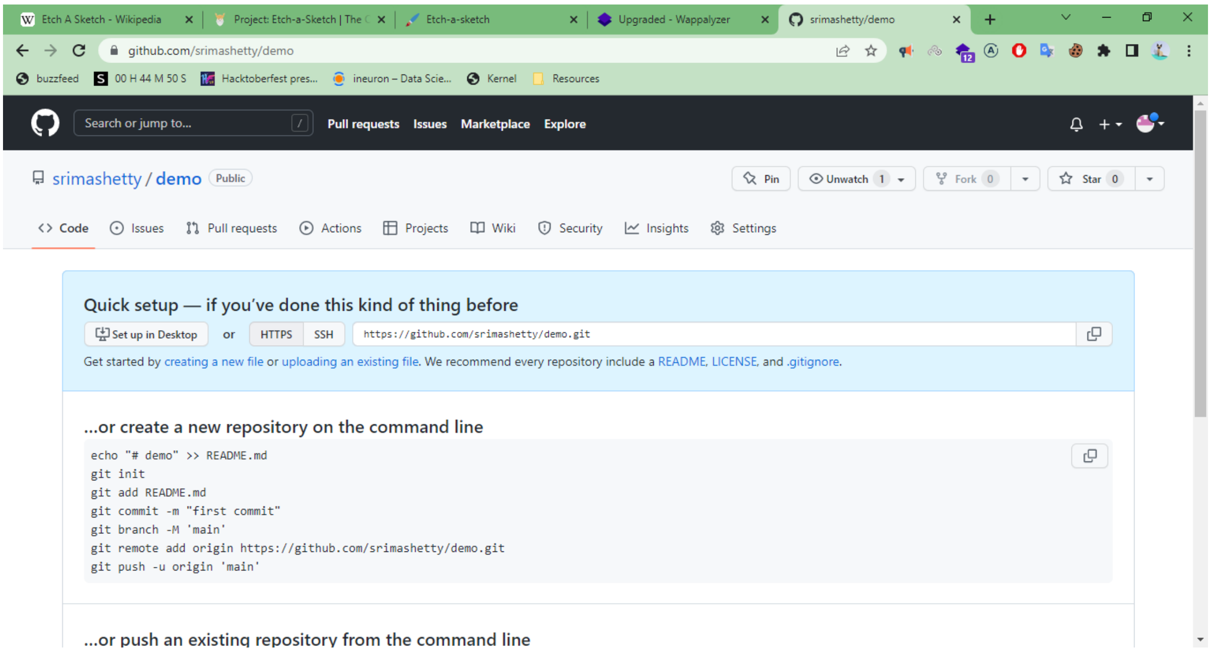
□ Add a README file
This is where you can write a long description for your project. Learn more.

Add .gitignore
Choose which files not to track from a list of templates. Learn more.
.gitignore template: None

Choose a license
A license tells others what they can and can't do with your code. Learn more.
License: None

You are creating a public repository in your personal account.

The repository is created and the below page is shown. It also provides a link to the repository, copy the link so you can add a remote repository in the git bash.



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

Set up in Desktop or HTTPS SSH `https://github.com/srimashetty/demo.git`

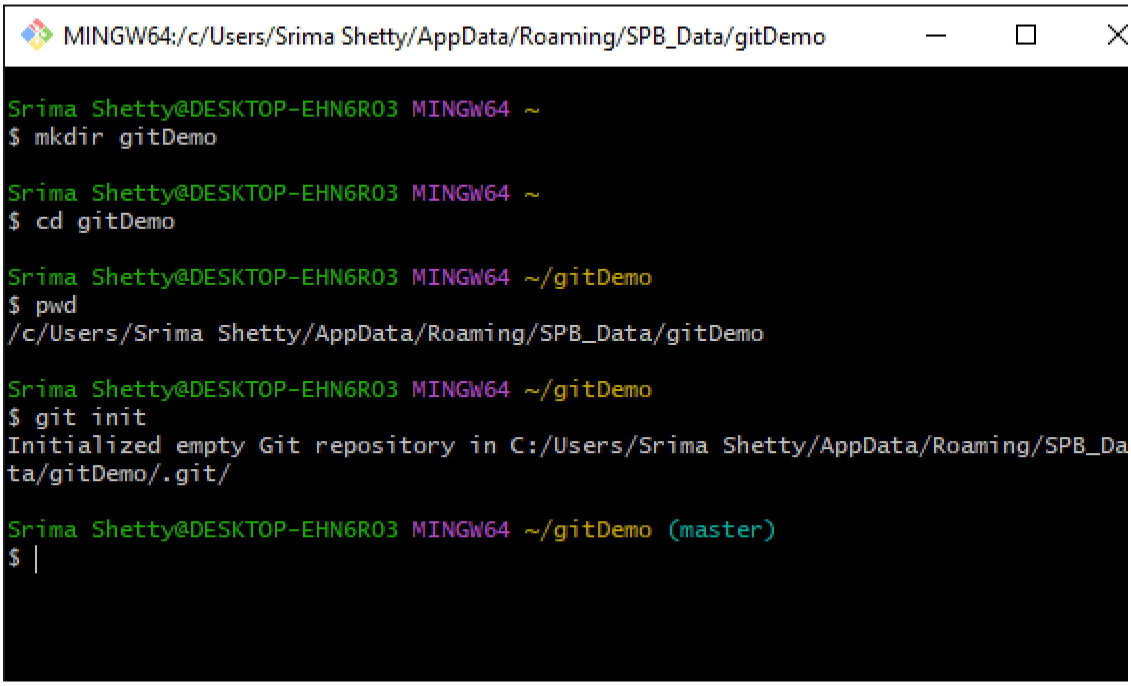
Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

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

```
echo "# demo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M 'main'
git remote add origin https://github.com/srimashetty/demo.git
git push -u origin 'main'
```

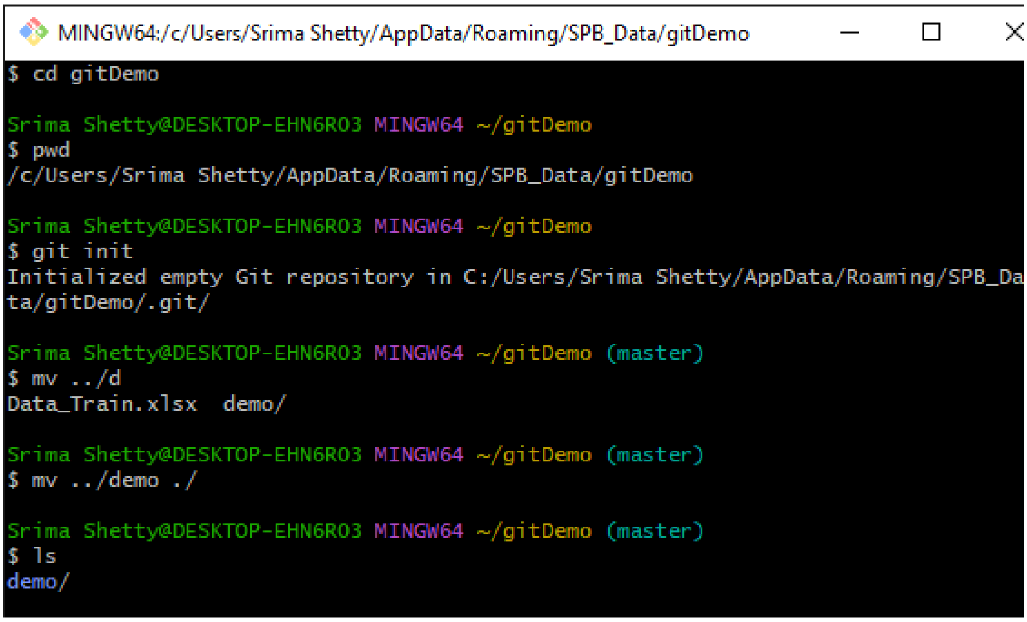
...or push an existing repository from the command line

Step 3: Create a new directory with the command mkdir, and initialise the repository with ‘git init’. Git init is used to create a git repository. We can convert the current unprovisioned directory to a git repository using ‘git init’.


MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo

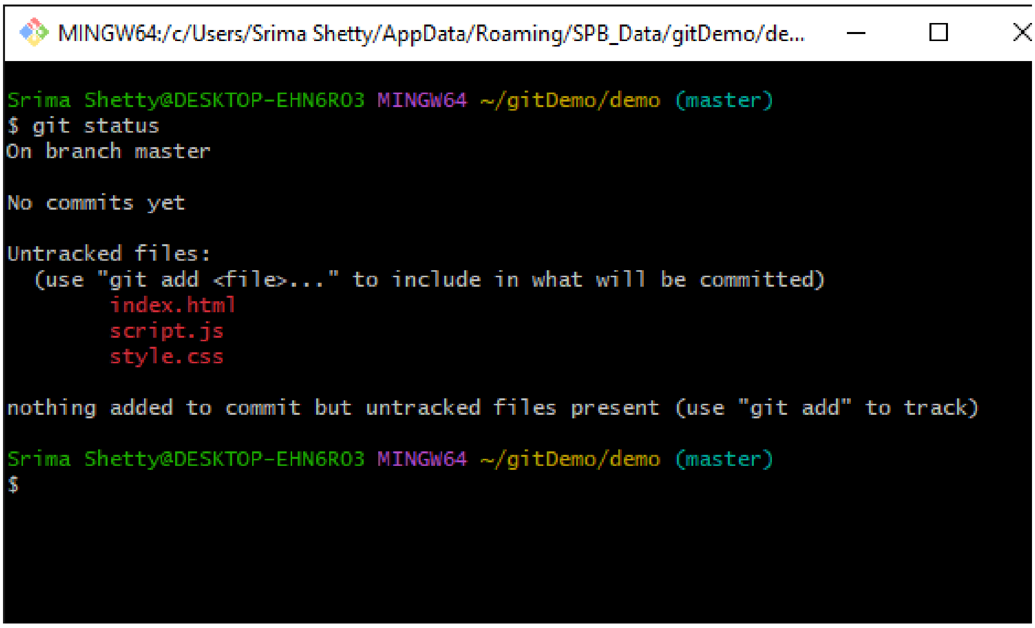
```
Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~
$ mkdir gitDemo

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~
$ cd gitDemo

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo
$ pwd
/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo
$ git init
Initialized empty Git repository in C:/Users/Srima Shetty/AppData/Roaming/SPB_Da
ta/gitDemo/.git/

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo (master)
$ |
```

Step 4: Add files, or make changes to the current git repository. Here I have moved files from another directory to the empty git repository. I have moved a directory of the name ‘demo’ which contains 3 files, index.html, styles.css and a script.js.


MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo

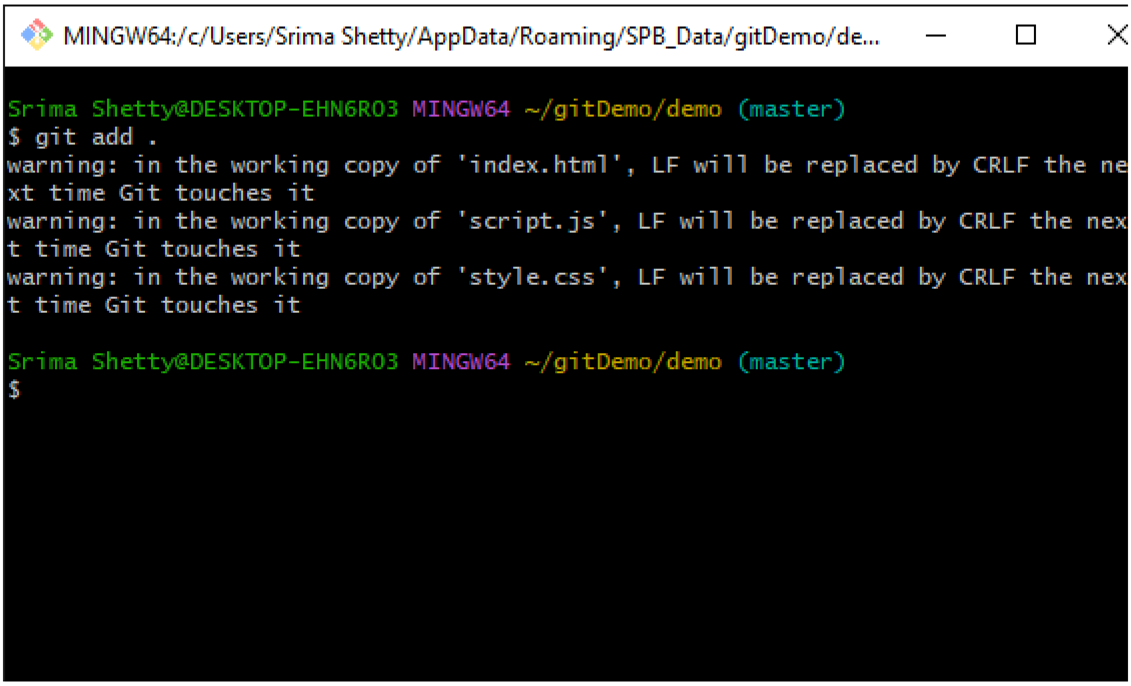
```
$ cd gitDemo

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo
$ pwd
/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo
$ git init
Initialized empty Git repository in C:/Users/Srima Shetty/AppData/Roaming/SPB_Da
ta/gitDemo/.git/

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo (master)
$ mv ../d
Data_Train.xlsx  demo/

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo (master)
$ mv ../demo ./

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo (master)
$ ls
demo/
```

Step 5: Once the changes are made, check the status of the repository using ‘git status’. Git status checks the status of the repository and the staging area. It lets us see which changes are staged and which aren’t. Here I can see that the three files I added are not staged yet., as they are highlighted in red color.


MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

```
Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        index.html
        script.js
        style.css

nothing added to commit but untracked files present (use "git add" to track)

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$
```

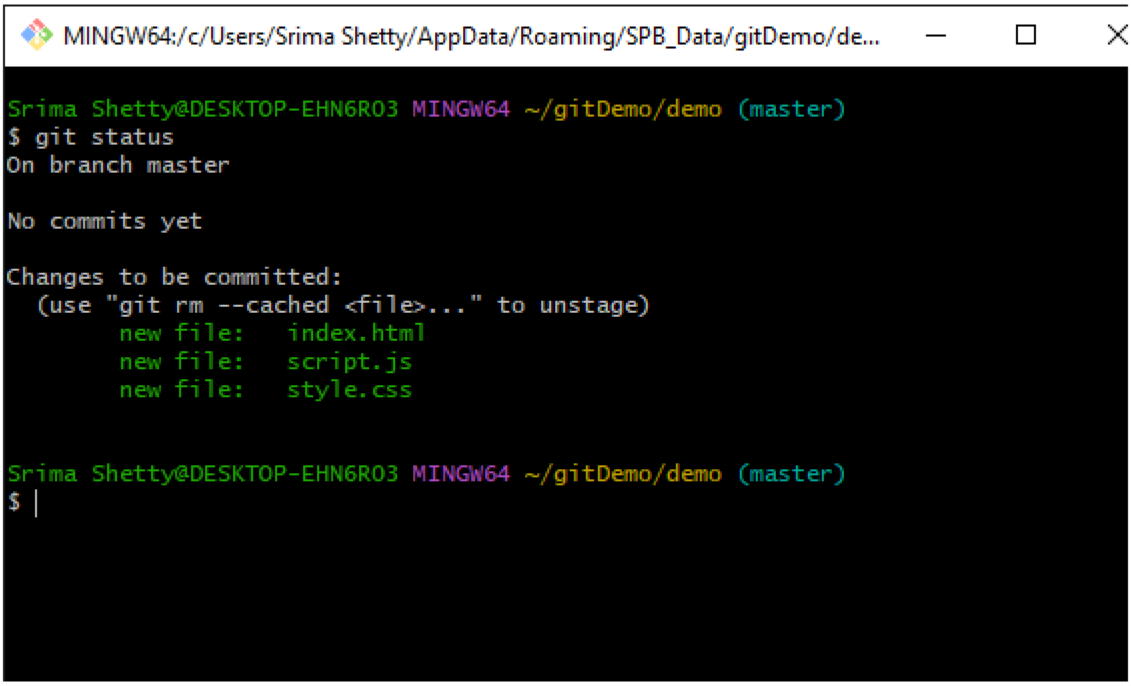
Step 6: Next stage the unstaged changes by using ‘git add .’ Git adds the changes in the working directory to the staging area. The dot indicates to add the current directory and all the files within it.



MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

```
Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$ git add .
warning: in the working copy of 'index.html', LF will be replaced by CRLF the ne
xt time Git touches it
warning: in the working copy of 'script.js', LF will be replaced by CRLF the nex
t time Git touches it
warning: in the working copy of 'style.css', LF will be replaced by CRLF the nex
t time Git touches it

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$
```

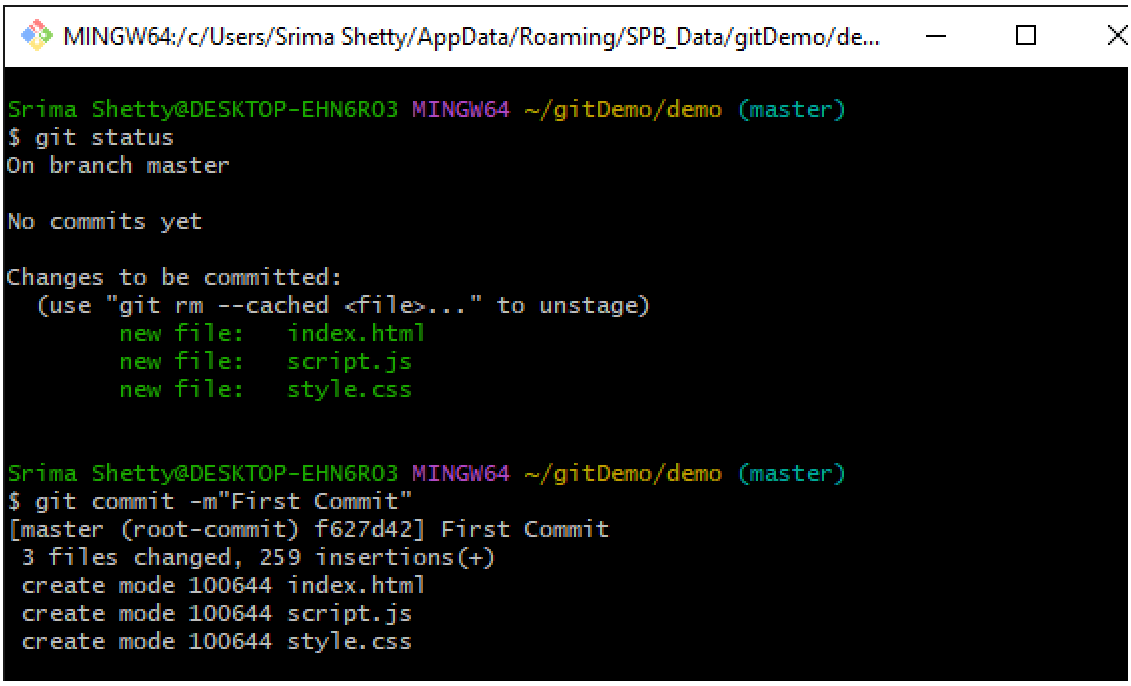
Step 7: After git add, I have checked the status of the repository again using ‘git status’, to confirm if all the changes are added to the staging area.



MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

```
Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   index.html
        new file:   script.js
        new file:   style.css


Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$ |
```

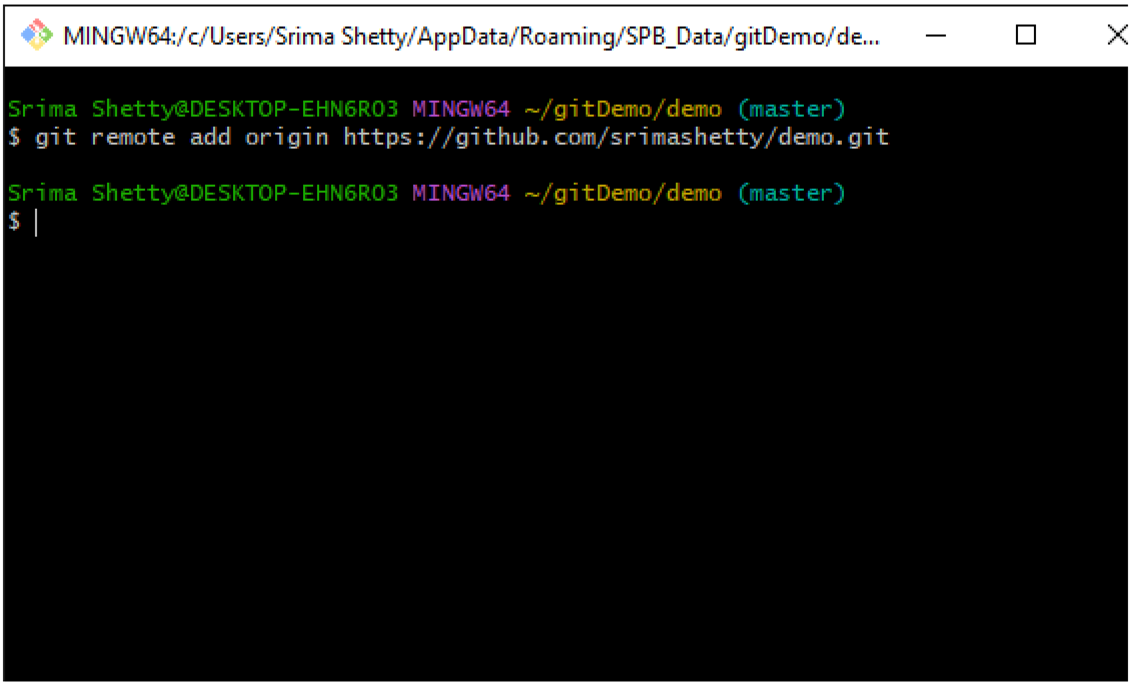
Step 8: Commit all the changes that were added. Git commit basically takes a snapshot of the project's currently staged changes. Using -m, we are giving a message or a label to the commit just made. Here I have given "First Commit" as the message for my commit.



MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

```
Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   index.html
        new file:   script.js
        new file:   style.css


Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$ git commit -m"First Commit"
[master (root-commit) f627d42] First Commit
 3 files changed, 259 insertions(+)
 create mode 100644 index.html
 create mode 100644 script.js
 create mode 100644 style.css
```

Step 9: Connect the github, which would be the remote repository to the local repository on git bash using git remote add and the link of the repository. Here, the link to my remote repository is https://github.com/srimashetty/demo.git.

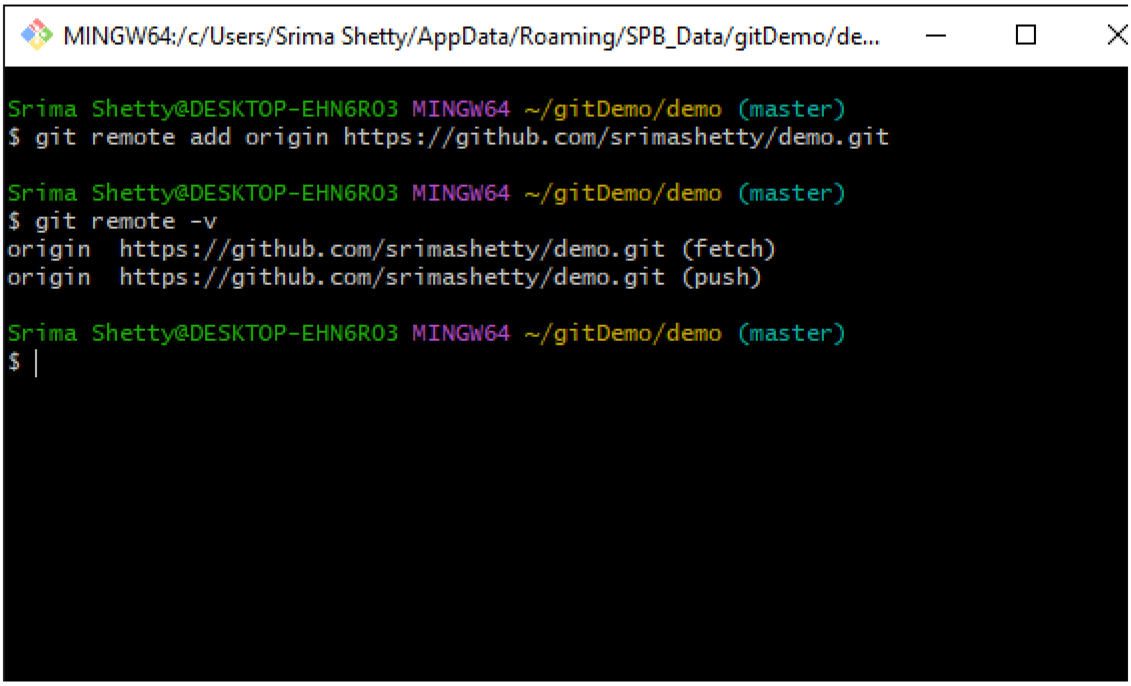


MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

```
Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$ git remote add origin https://github.com/srimashetty/demo.git

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (master)
$ |
```

In

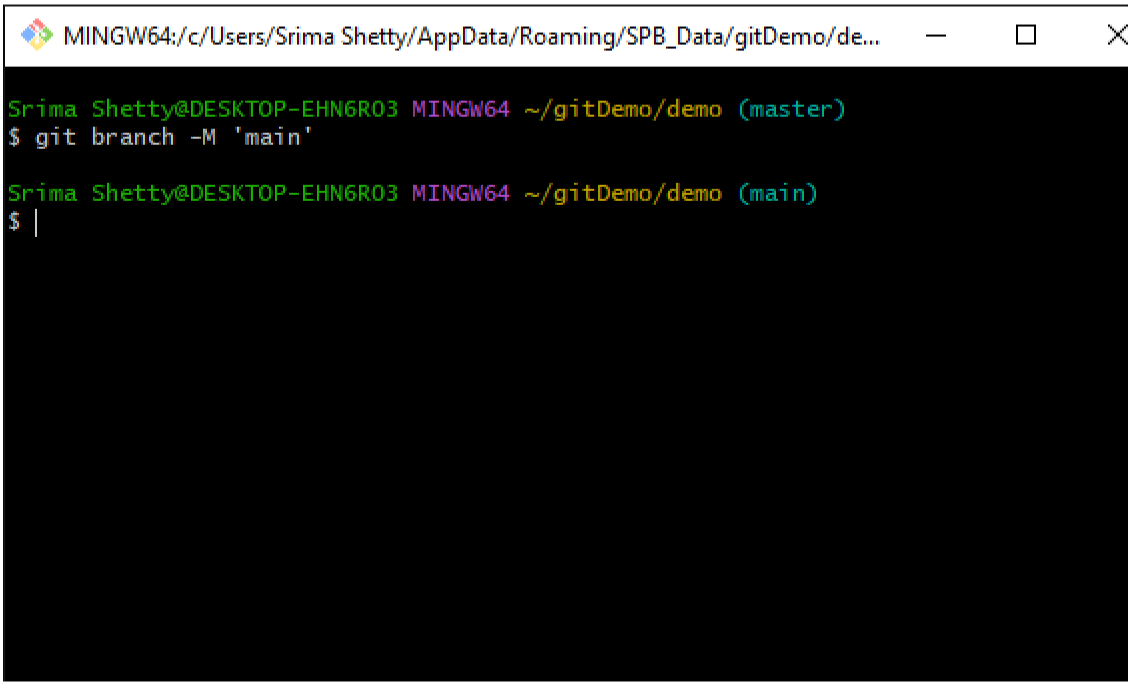
the below snapshot, I am confirming if the remote repository is connected to the local repository by using git remote -v.



MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

```
Srima Shetty@DESKTOP-EHN6RO3 MINGW64 ~/gitDemo/demo (master)
$ git remote add origin https://github.com/srimashetty/demo.git

Srima Shetty@DESKTOP-EHN6RO3 MINGW64 ~/gitDemo/demo (master)
$ git remote -v
origin  https://github.com/srimashetty/demo.git (fetch)
origin  https://github.com/srimashetty/demo.git (push)

Srima Shetty@DESKTOP-EHN6RO3 MINGW64 ~/gitDemo/demo (master)
$ |
```

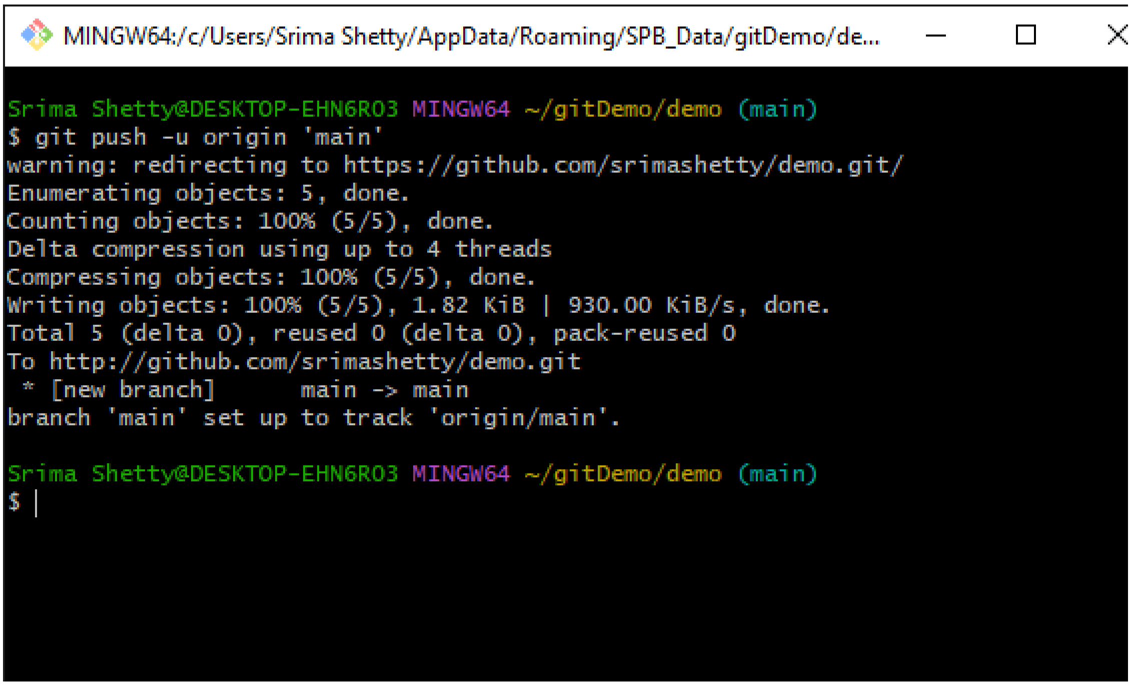
Step 10: I have now created a new branch called ‘main’. I have done so with the command, “git branch -M ‘main’”



MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

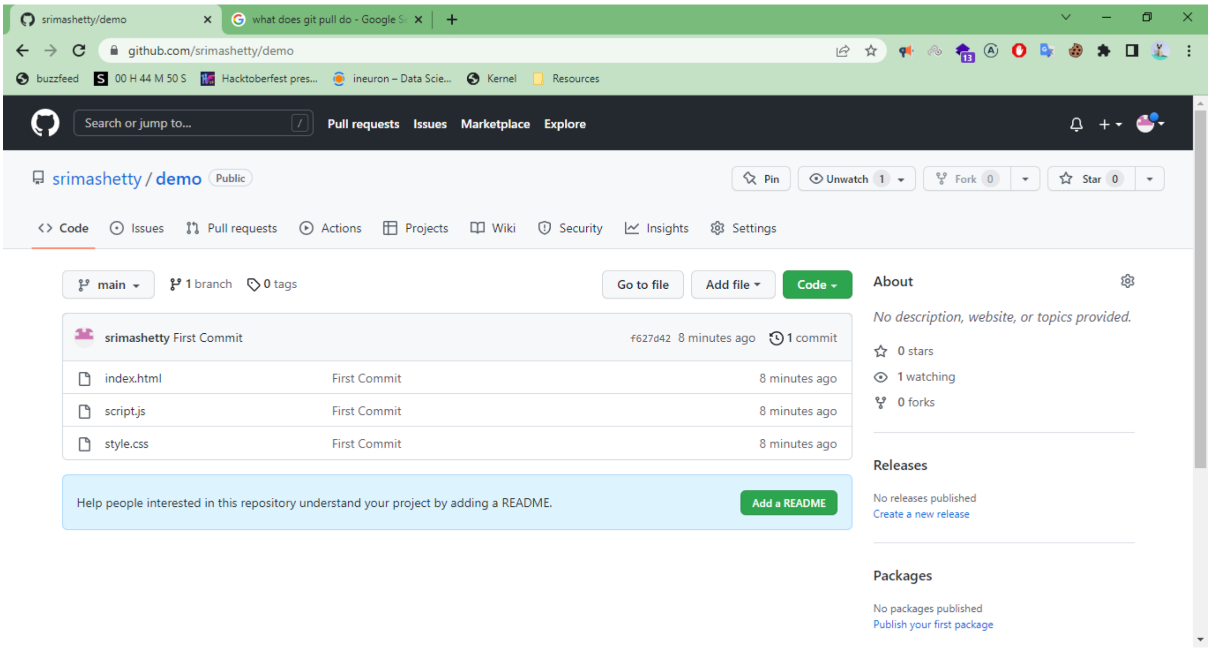
```
Srima Shetty@DESKTOP-EHN6RO3 MINGW64 ~/gitDemo/demo (master)
$ git branch -M 'main'

Srima Shetty@DESKTOP-EHN6RO3 MINGW64 ~/gitDemo/demo (main)
$ |
```

Step 11: I have finally uploaded the changes from the local repository to the remote repository using the git push. I have used the command, ‘git push -u origin ‘main’’, here main is the branch name.


MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

```
Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (main)
$ git push -u origin 'main'
warning: redirecting to https://github.com/srimashetty/demo.git/
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 1.82 KiB | 930.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To http://github.com/srimashetty/demo.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (main)
$ |
```

Step 12: I am now checking on the remote repository on github, to see if the changes are uploaded, as we can see in the snapshot, the changes are done and the commit is made.



srimashetty First Commit f627d42 8 minutes ago 1 commit

index.html	First Commit	8 minutes ago
script.js	First Commit	8 minutes ago
style.css	First Commit	8 minutes ago

Help people interested in this repository understand your project by adding a README.

About

No description, website, or topics provided.

0 stars

1 watching

0 forks

Releases

No releases published

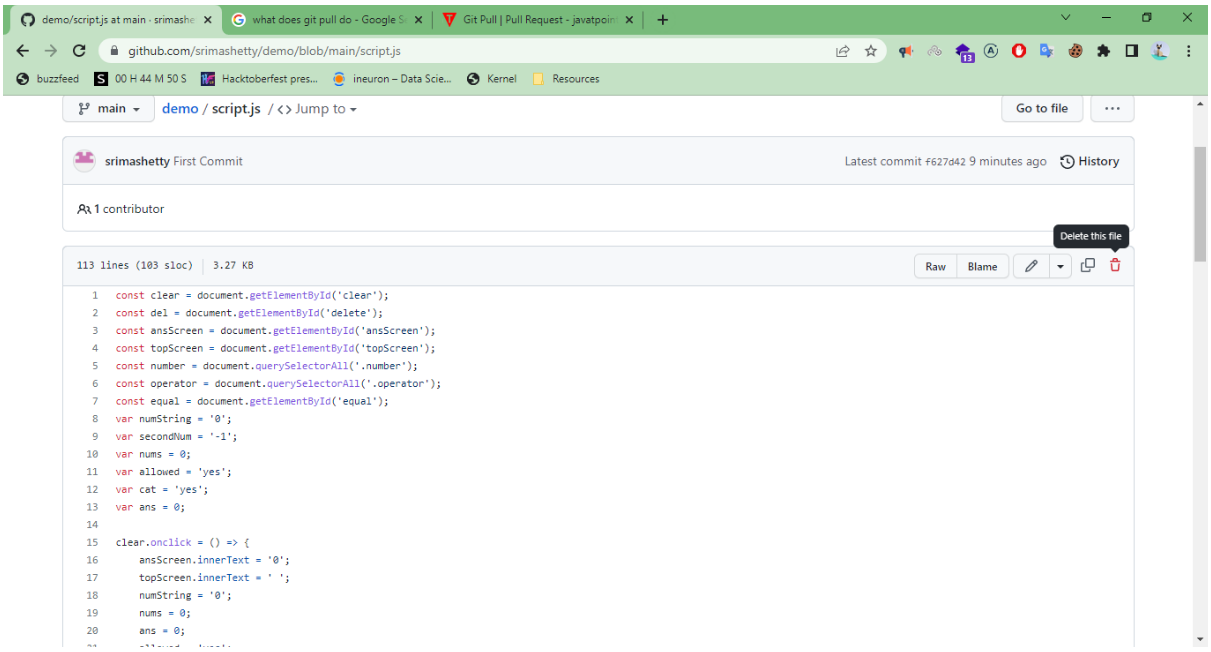
Create a new release

Packages

No packages published

Publish your first package

Step 13: I am now deleting script.js on github, and making that commit.


demo/script.js at main · srimashe
what does git pull do - Google S
Git Pull | Pull Request - javatpoin
github.com/srimashetty/demo/blob/main/script.js
buzzfeed
00 H 44 M 50 S
Hacktoberfest pres...
ineuron – Data Scie...
Kernel
Resources
main
demo / script.js / <> Jump to
Go to file
History
Raw
Blame
Delete this file

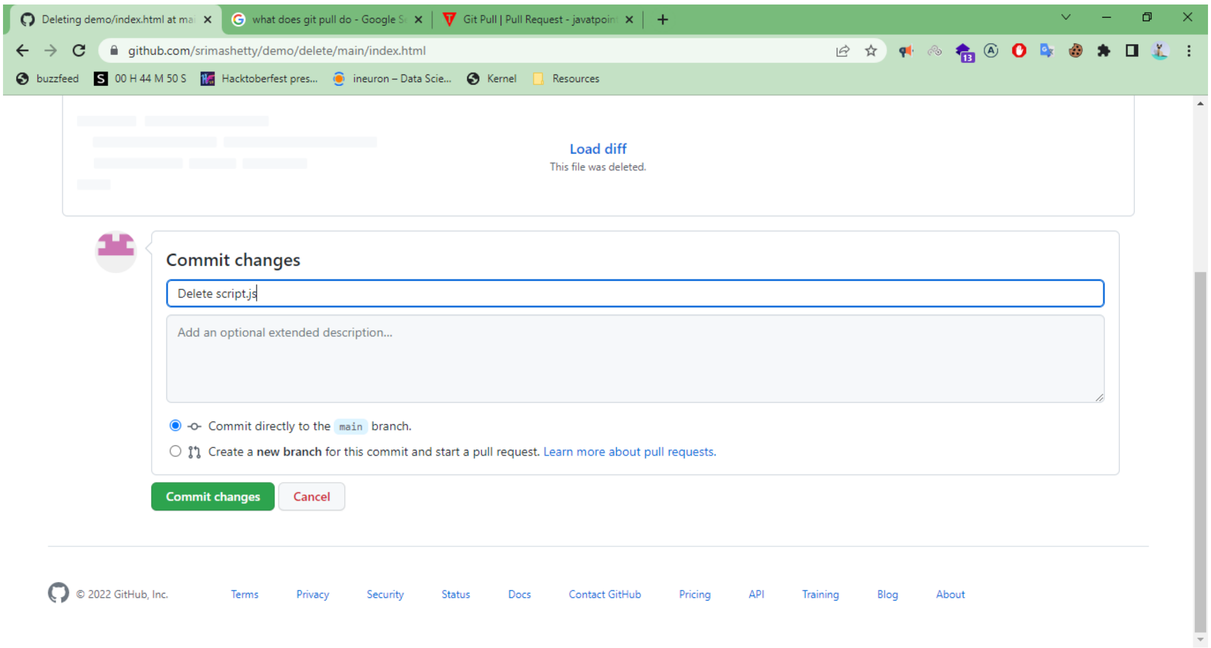
srimashetty First Commit

Latest commit f627d42 9 minutes ago

1 contributor

113 lines (103 sloc) | 3.27 KB

```
const clear = document.getElementById('clear');
const del = document.getElementById('delete');
const ansScreen = document.getElementById('ansScreen');
const topScreen = document.getElementById('topScreen');
const number = document.querySelectorAll('.number');
const operator = document.querySelectorAll('.operator');
const equal = document.getElementById('equal');
var numString = '0';
var secondNum = '-1';
var nums = 0;
var allowed = 'yes';
var cat = 'yes';
var ans = 0;

clear.onclick = () => {
    ansScreen.innerText = '0';
    topScreen.innerText = ' ';
    numString = '0';
    nums = 0;
    ans = 0;
```



Commit changes

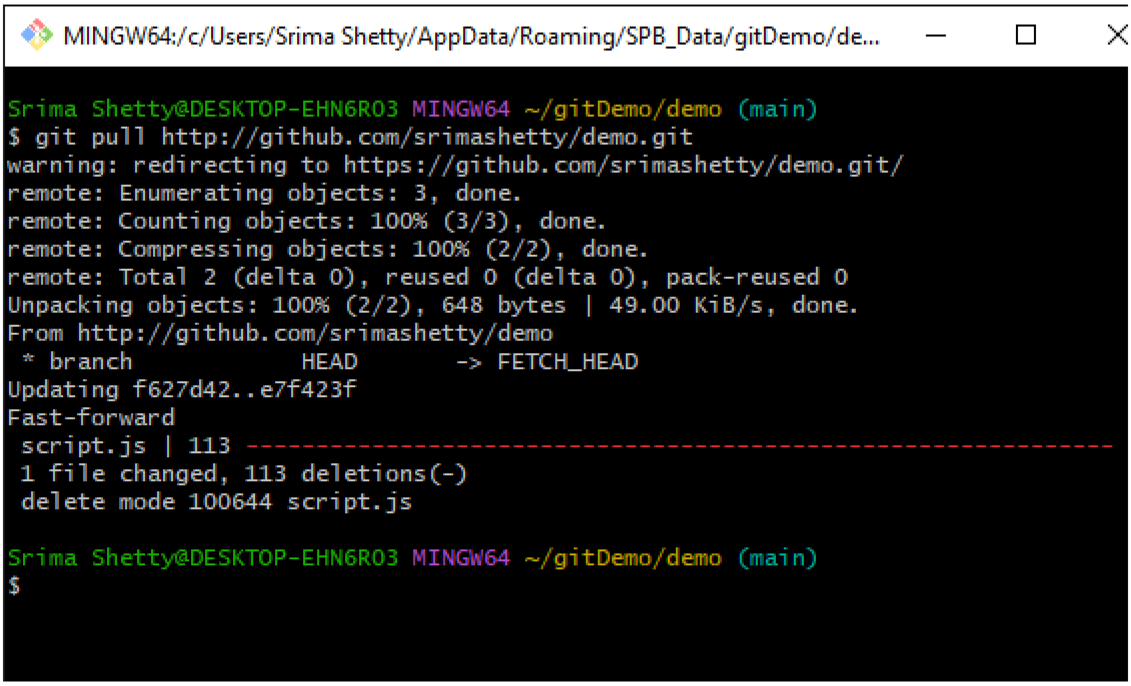
Delete script.js

Add an optional extended description...

● Commit directly to the main branch.

○ Create a **new branch** for this commit and start a pull request. Learn more about pull requests.

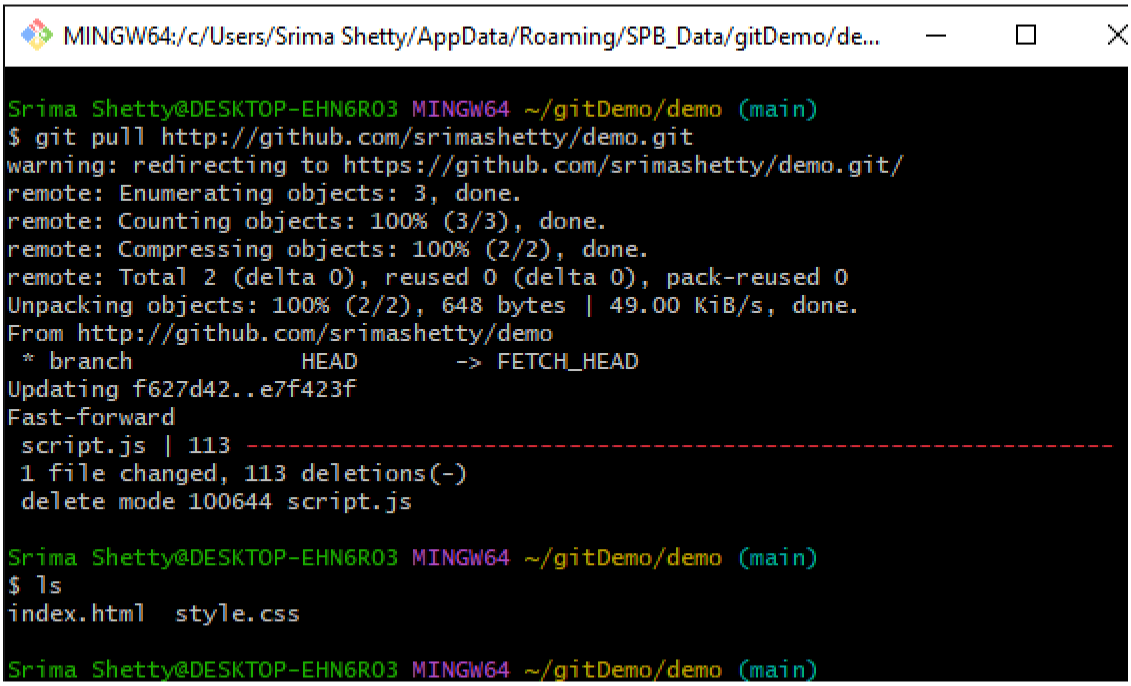
Step 13: But now, I want the commit I made on github to be also applied on my local repository, so I will be using git pull. Git pull is used to fetch and download content from a remote repository and immediately update the local repository to match that content. So here, the script.js is also deleted in the local repository after using git pull.



MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

```
Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (main)
$ git pull http://github.com/srimashetty/demo.git
warning: redirecting to https://github.com/srimashetty/demo.git/
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (2/2), 648 bytes | 49.00 KiB/s, done.
From http://github.com/srimashetty/demo
 * branch            HEAD       -> FETCH_HEAD
Updating f627d42..e7f423f
Fast-forward
 script.js | 113 ---------------------------------------------------------------
 1 file changed, 113 deletions(-)
 delete mode 100644 script.js

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (main)
$
```

Step 14: I am using ls command to see if the file is deleted.

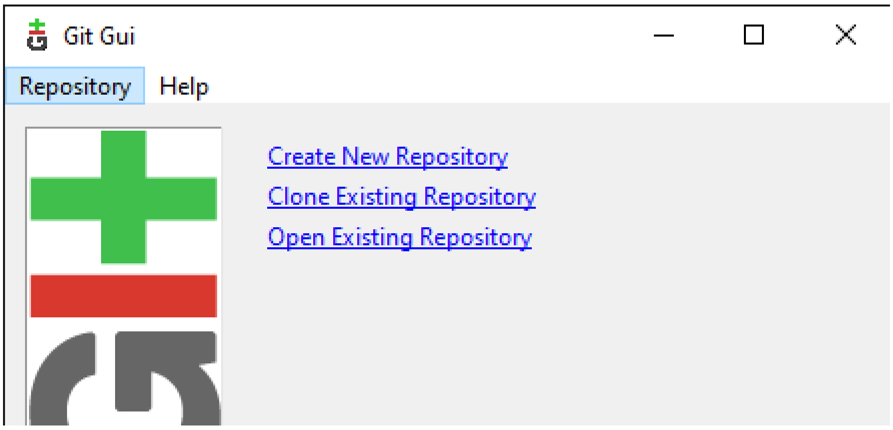


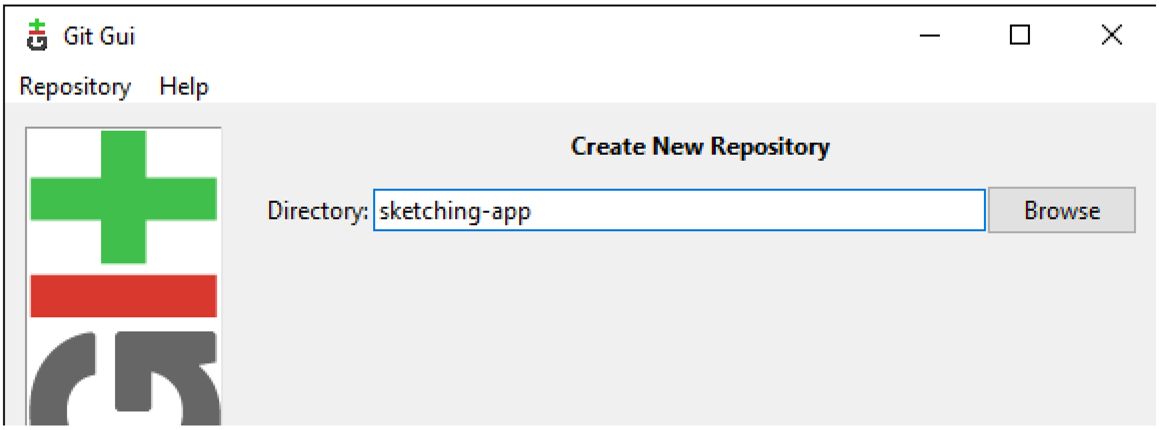
MINGW64:/c/Users/Srima Shetty/AppData/Roaming/SPB_Data/gitDemo/de...

```
Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (main)
$ git pull http://github.com/srimashetty/demo.git
warning: redirecting to https://github.com/srimashetty/demo.git/
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (2/2), 648 bytes | 49.00 KiB/s, done.
From http://github.com/srimashetty/demo
 * branch            HEAD       -> FETCH_HEAD
Updating f627d42..e7f423f
Fast-forward
 script.js | 113 ---------------------------------------------------------------
 1 file changed, 113 deletions(-)
 delete mode 100644 script.js

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (main)
$ ls
index.html  style.css

Srima Shetty@DESKTOP-EHN6R03 MINGW64 ~/gitDemo/demo (main)
```

USING GIT GUI:

1. Creating a new repository

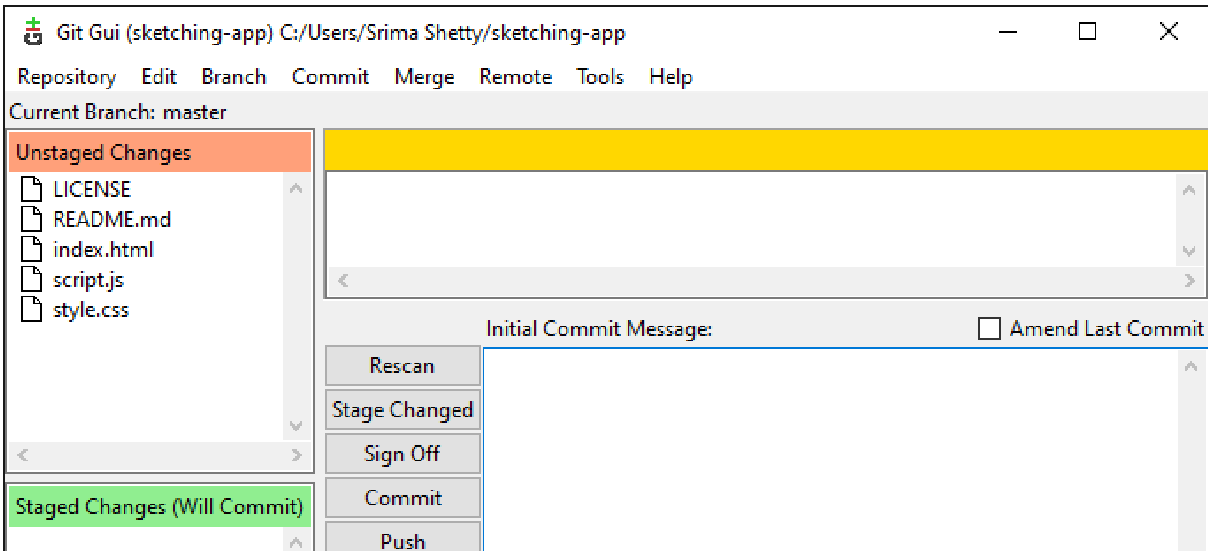




Create New Repository

Directory: sketching-app

2. Adding unstaged files

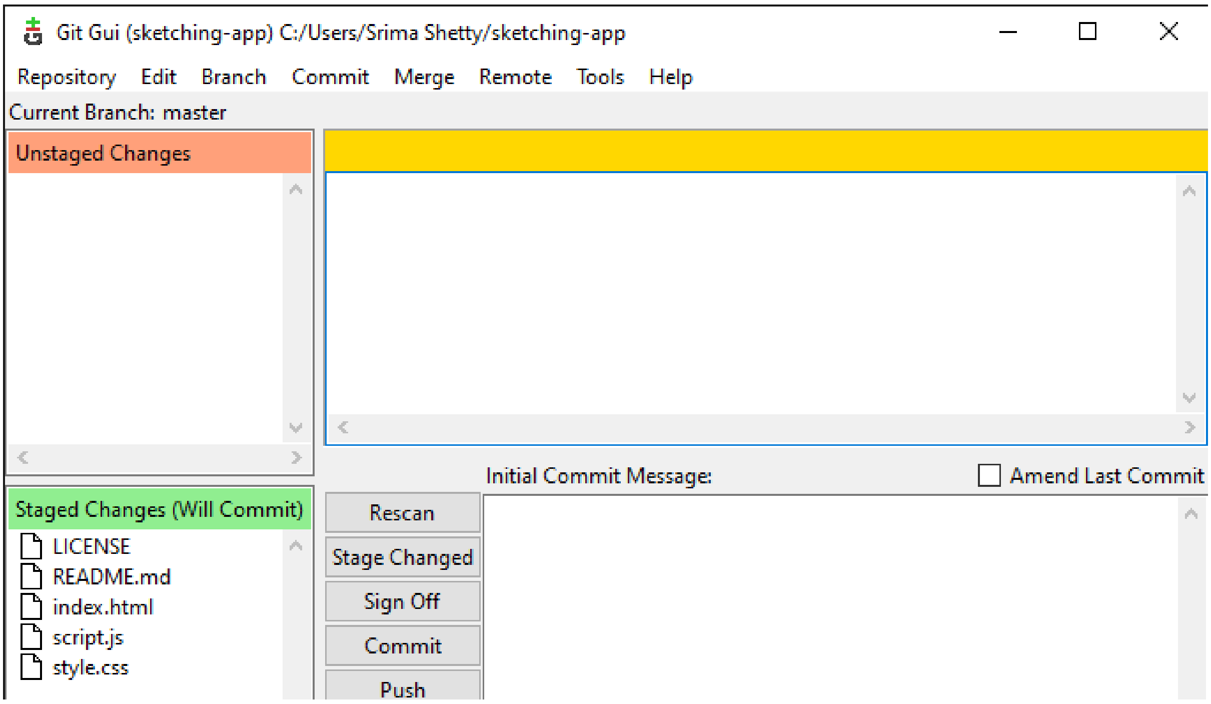


Git Gui (sketching-app) C:/Users/Srima Shetty/sketching-app

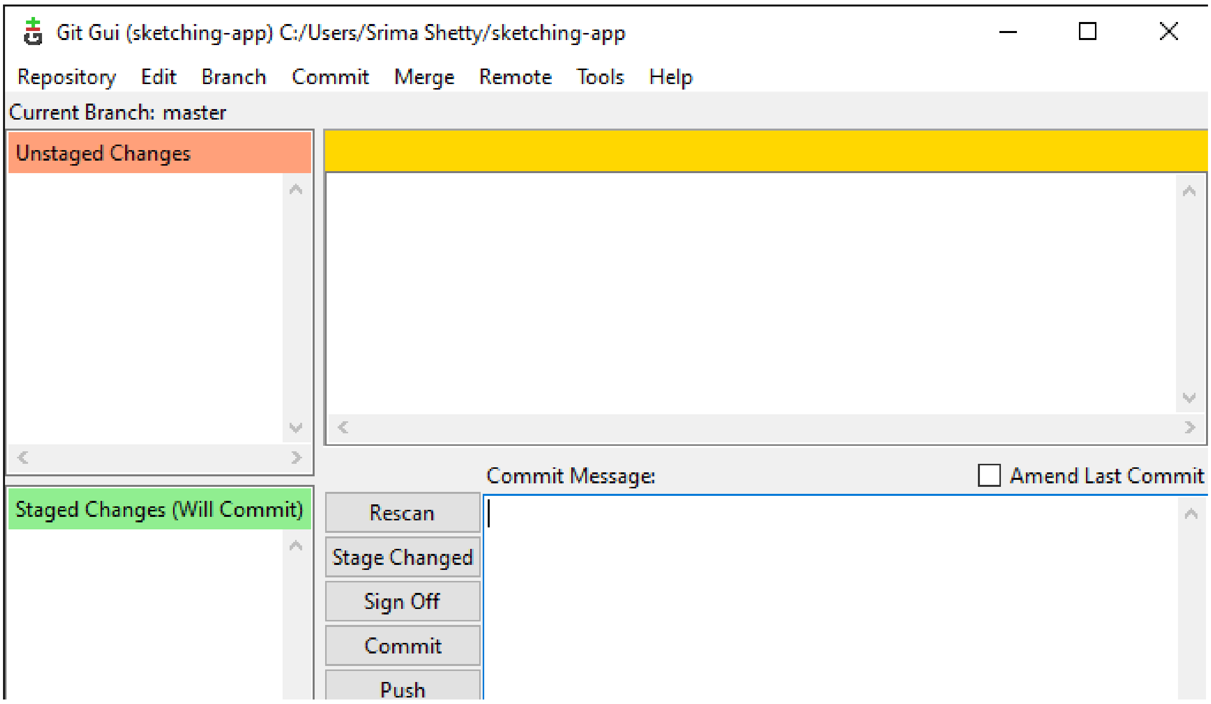
Current Branch: master

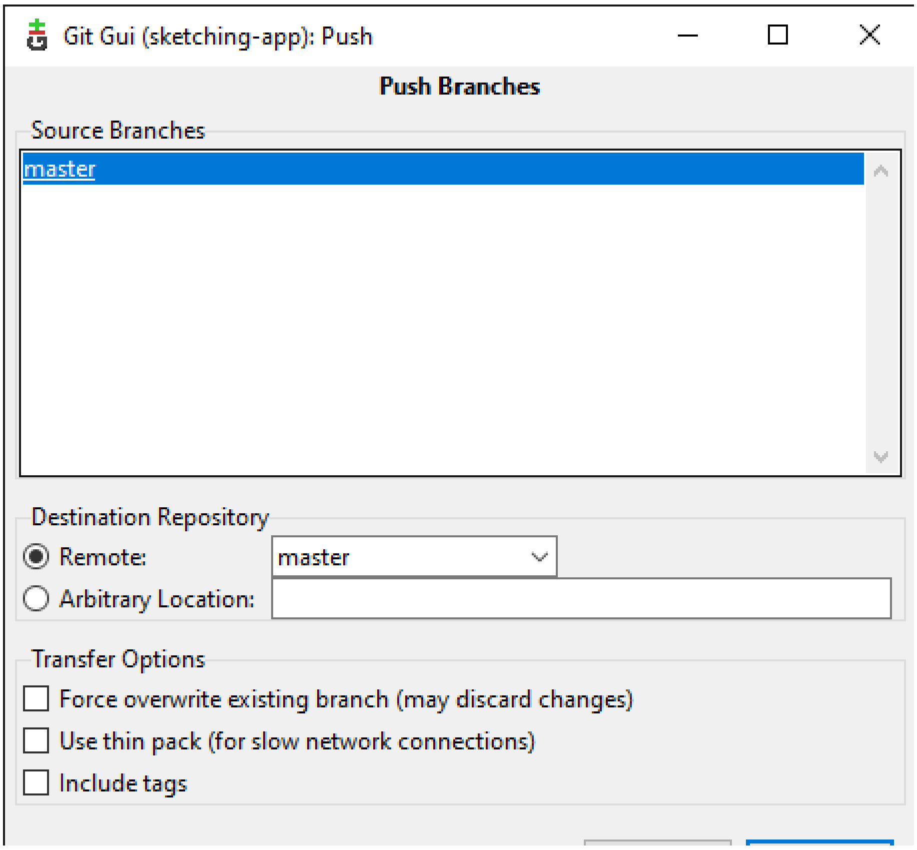
- LICENSE
- README.md
- index.html
- script.js
- style.css

3. Commit



4. Pushing file to remote Repository





Git Gui (sketching-app): Push

Push Branches

Source Branches

master

Destination Repository

● Remote: master

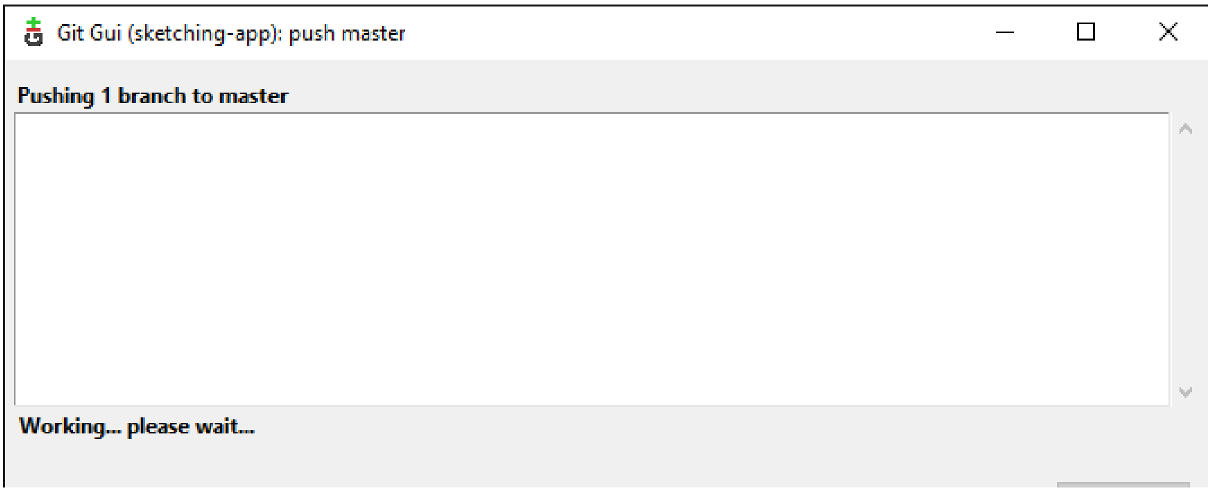
○ Arbitrary Location:

Transfer Options

□ Force overwrite existing branch (may discard changes)

□ Use thin pack (for slow network connections)

□ Include tags

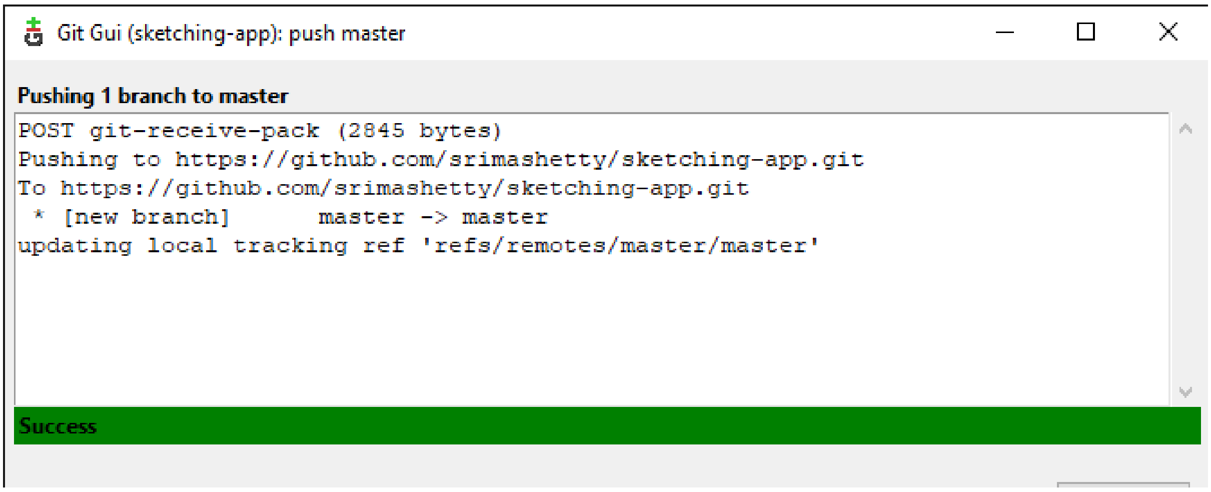


Git Gui (sketching-app): push master

Pushing 1 branch to master

Working... please wait...

5.



Pushing 1 branch to master

```
POST git-receive-pack (2845 bytes)
Pushing to https://github.com/srimashetty/sketching-app.git
To https://github.com/srimashetty/sketching-app.git
 * [new branch]      master -> master
updating local tracking ref 'refs/remotes/master/master'
```

Success

Conclusion

In this project, I learned how to use Git and Github. I used the terminal in Git bash, and learned the functionalities of various git commands like, git init, git init, git status, git add, git branch, git commit, git push, git pull etc.
I also learned how to create a repository on github. I learned what various files in github meant, like the licences, .gitignore and the .readme file. I learned how to change settings in the github repository as well.