

1. Locate to the folder you want to place under git in your terminal.
`$ cd Desktop/ML-1`
2. Initialize Git:
`$ touch README.md` # To create a README file for the repository
Must add commit at this stage to have master branch intact
`$ git init` # Initiates an empty git repository
3. Creating a new branch
`git checkout -b <my branch name>.`
This command will automatically create a new branch and then 'check you out' on it, meaning git will move you to that branch, off of the primary branch.
4. Check which branch you are in -
`$ git branch`
master
* my_new_branch
The branch name with the asterisk next to it indicates which branch you're on at that given time. Default branch is main.
5. If you are in the new branch, paste your files manually in the local repository.
Then we add these files to the Staging Area for commit:
`$ git add.` # Adds all the files in the local repository and stages them for commit

OR if you want to add a specific file (if you have already added all three files, then no need to do this)

`$ git add your_file_name` #To add a specific file
6. Before we commit let's see what files are staged:

`$ git status` # Lists all new or modified files to be committed
7. Now to commit files you added to your git repo:

`$ git commit -m "First commit"`

The message in the " " is given so that the other users can read the message and see what changes you made.

8. Add a remote repository (link of manually created github repository)

```
$ git remote add origin remote_repository_URL
```

```
# sets the new remote
```

```
# now your local repository will point to the github repository
```

9. Check all the remote connections

```
$ git remote -v
```

```
# List the remote connections you have to other repositories.
```

The **git remote -v** command lists the URLs of the remote connections you have to other repositories.

10. Finally push your code to origin (the remote repo)

```
$ git push -u origin master (sending master to upstream)
```

```
$ git push -u origin my_new_branch (sending new branch to upstream)
```

Now the **git push** command pushes the changes in your local repository up to the remote repository you specified as the origin. Also, notice that we couldn't push without step 8 and 9.