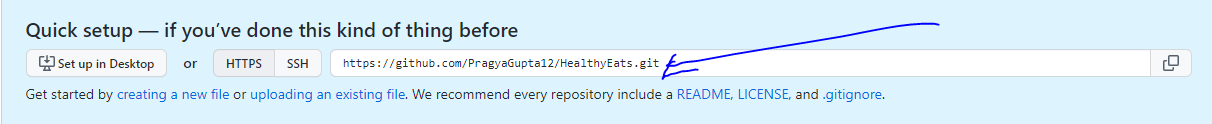
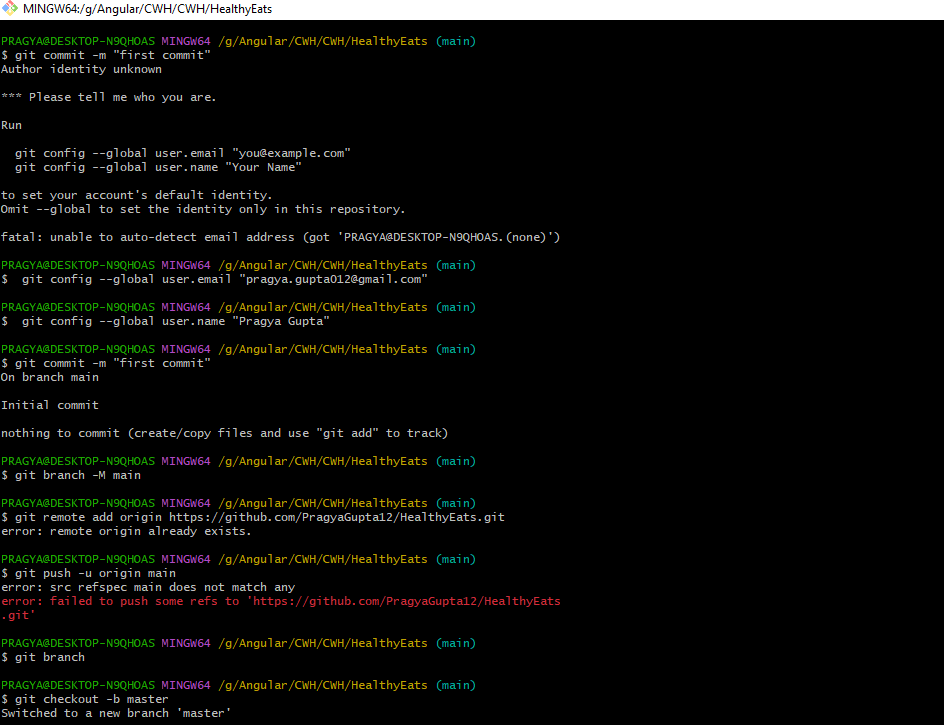
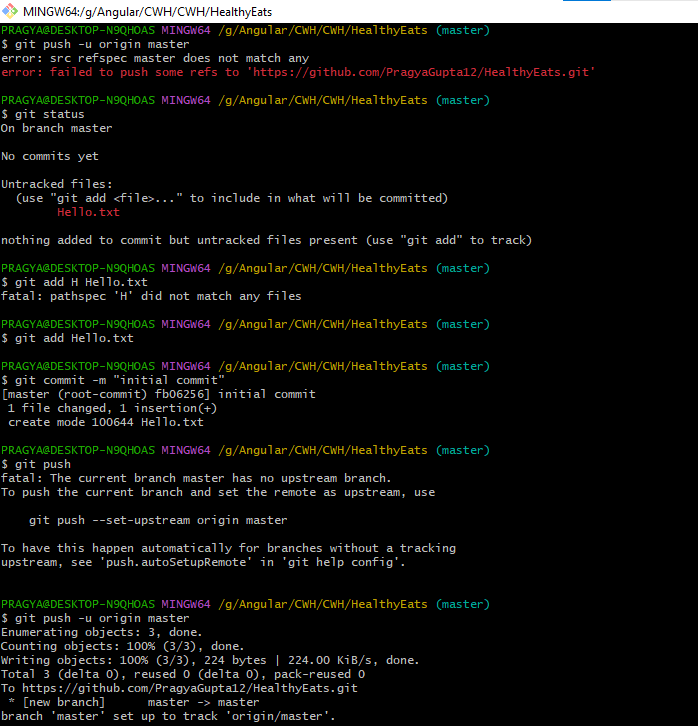
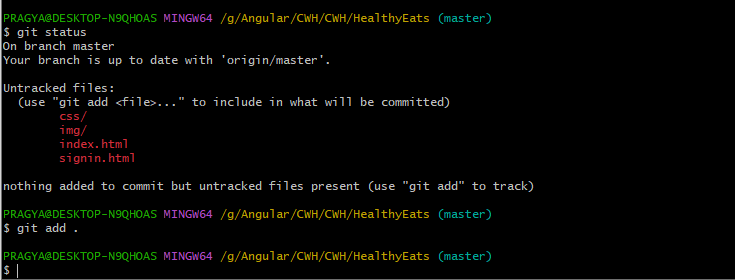
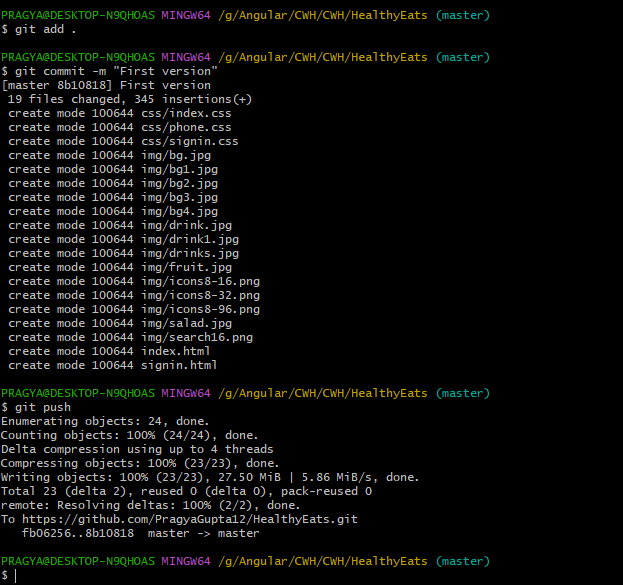
1. Go to the folder where you want to clone your repository and do right click and hit “git bash here”
2. git clone “shift+insert”(repo URL)



1. Go inside the folder created inside the folder you used for git.
2. Go inside the folder created by git then right-click and hit “git bash here”. Follow the below ones
3. 
4. 
5. Git status tells what files are there in your folder.
6. Now we can push all the files that are there in the folder
7. For one file we did “git add filename” but for all files, we did “git add .”
8. 
9. 

Always do the following:

* **Git add .**
* **Git commit -m “message”**
* **Git push**

The changes are saved by committing and a new file version is created in your local system. Now, this doesn't mean that there will be two different files but Git would know what changes are there between the original file and the file you commit.

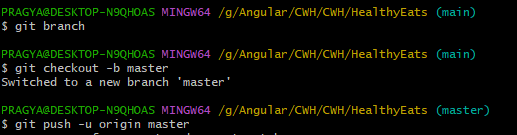
By pushing we push all to the origin (version control system here it is GitHub)

Git push -u origin master.

* Master is the branch name.
* -u is the upstream flag, which means we are linking everything to the server.
* Origin is the remote name. Remote is which server you want to push in (In a way where your code will go).
* So Origin is the repo where you push/pull your code. And if we have multiple repositories then the origin will be different depending on the repo you are using but still it will be called the origin.



* Checkout is for moving in between branches
* -b is creating a new branch flag



* .md- markdown file
* git cherry-pick: when you are working with multiple branches in the same repo and one branch has 2 commits and another branch has 1 commit and you want to move only one commit from the branch one to branch two then we can do it through the cherry pick. But in this case, the commit that has been moved from branch 1 to 2 will no longer exist in branch 1.
* Git stash
* Git stash pop