|  |  |  |
| --- | --- | --- |
| **s.no** | Command | Description |
| 1 | **Git config** | Sets the author name and mail id to be used with your commits |
| 2 | **Git init** | Starts new repository |
| 3 | **Git clone** | Obtain the repository from existing url |
| 4 | **Git add .** | Adds files to staging area |
| 5 | **Git commit -m** | Snapshots the file permanently in the version |
| 6 | Git commit -a | Commits any files added with the git add command |
| 7 | Git diff | Shows the diff which r not yet staged |
| 8 | Git diff -staged | Shws differences between staging area and latest version present |
| 9 | Git diff[first branch] [second branch] | Differences between first and second branch |
| 10 | Git reset[file] | Command unstages the files preserves content |
| 11 | Git reset [commit] | Undos the commit after the specified commit and preservers the change locally |
| 12 | Git reset -hard[commit] | Lists all files to be committed |
| 13 | **Git status** | Command lists all the files to be committed |
| 14 | **Git rm [file]** | Deletes the file from working directory |
| 15 | Git log | Lists the version history |
| 16 | Git log -follow[file] | Lists version history |
| 17 | Git show | Shows meta deta |
| 18 | Git tag | Give tags to the specified commit |
| 19 | Git branch | Lists all the branches in current repository |
| 20 | Git branch -d | Deletes the feature branch |
| 21 | Git checkout | Switches from one branch to another |
| 22 | Git merge | Merges specific branch history to current branch |
| 23 | Git remote add | Connect to local repository |
| 24 | Git push | Commited to master branch |
| 25 | Git push -all | Pushes all the branches |
| 26 | Git push[variable name]: [branch name] | Deletes branch in repository |
| 27 | Git pull | Fetches and merges changes on your remote server in wd |
| 28 | Git stash | Temporarily storws all modified tracked files |
| 29 | Git stash save | Temporarily storws all modified tracked files |
| 30 | Git stash pop | Restores most recent stashed files |
| 31 | Git stash list | Stashed all changes |
| 32 | Git stash dropped | Discards recemtly stashed changesheet |
|  |  |  |

Creating the gitbash and GitHub repository

**Gitbash**

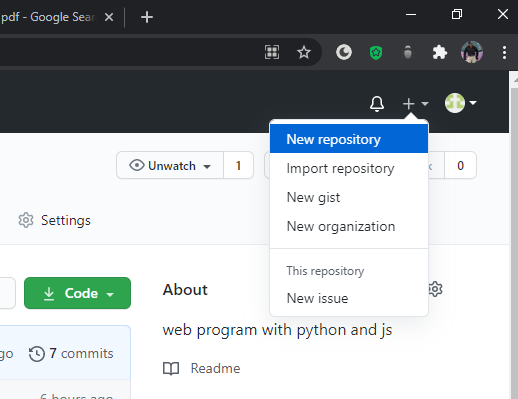
Download the git bash cmd at

<https://git-scm.com/download/win>

TO create an account in git hub go to the following site

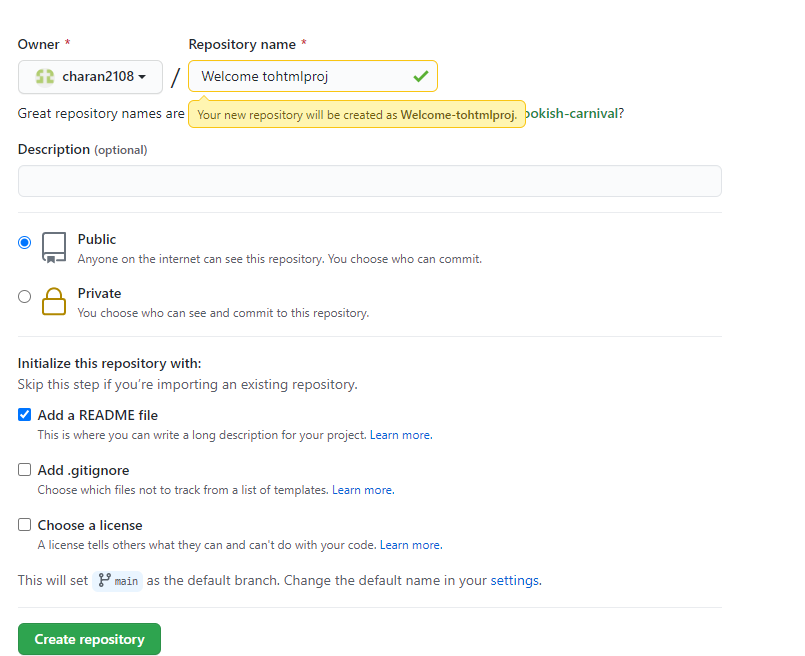
<https://github.com>

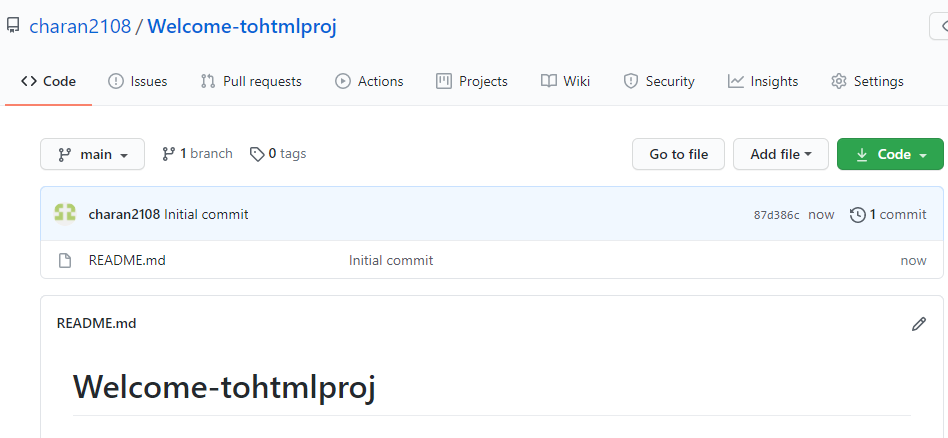
Once registered and after activation click on the + symbol on the right to create the repository as show below



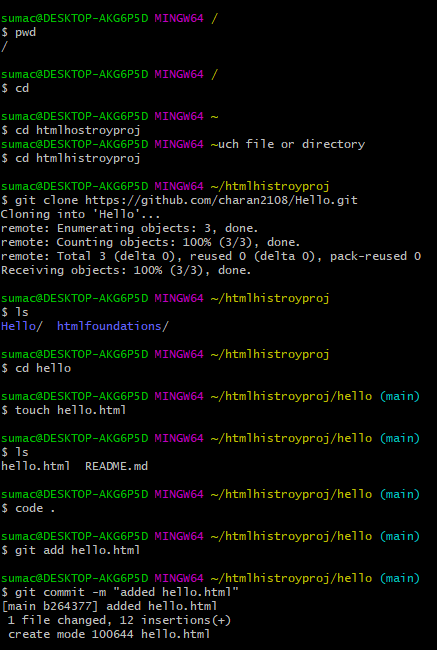
After clicking on the new repository .We are now going to name the repository as Welcome tohtmlproj

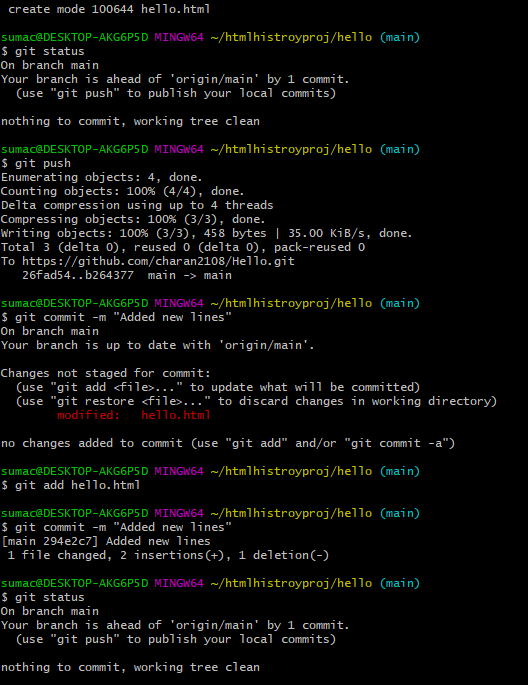
Check mark the add readmefile and click create the repository

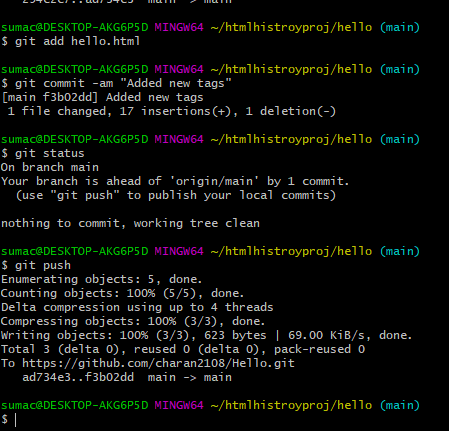


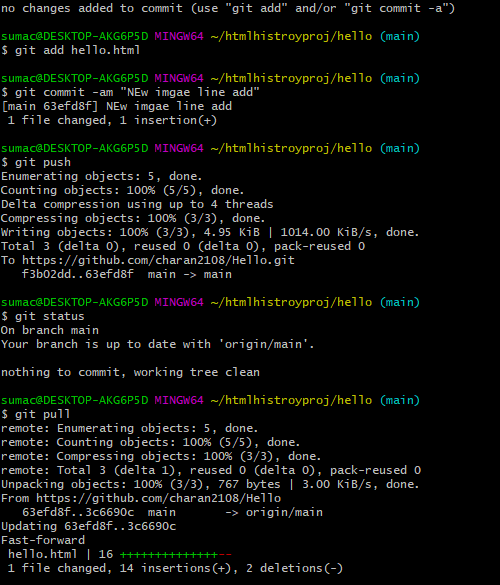


Cloning the project











**Create rep in git hub**

**U will create local repository**

**Clone the terminal to local (url)**

**Add the origin**

**Git status**

**Add the files**

**Commit the files**

**Push the files to the origin**