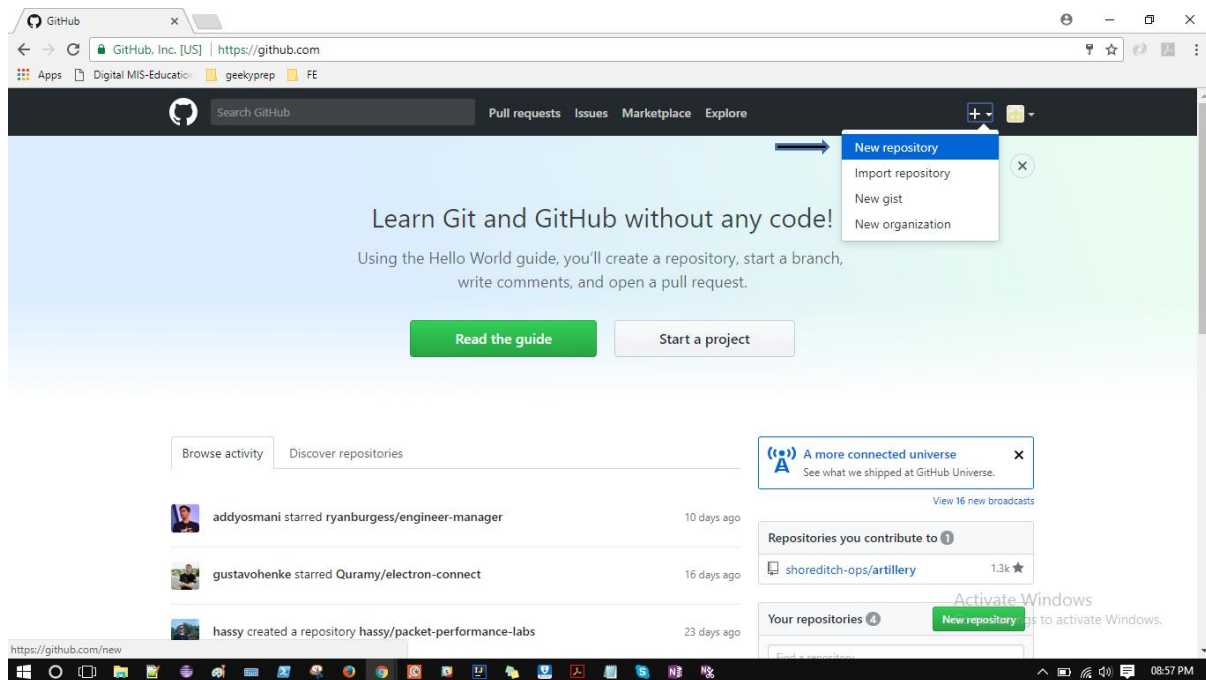
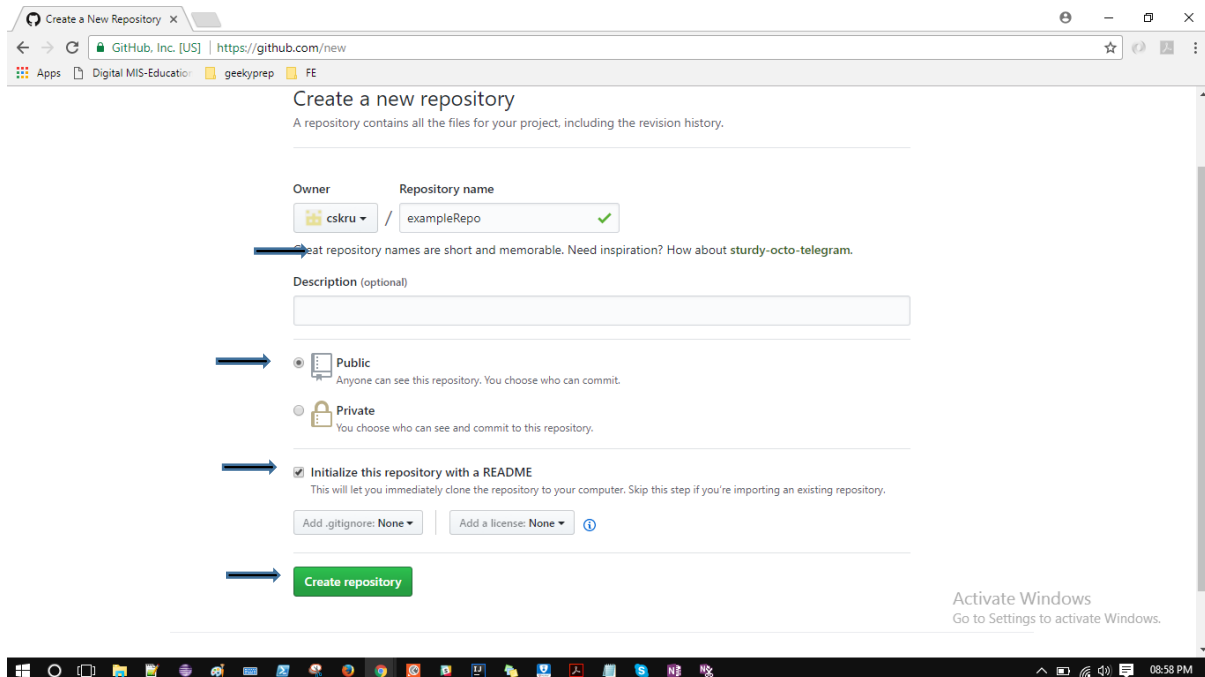


## Creating a default github page, step by step:

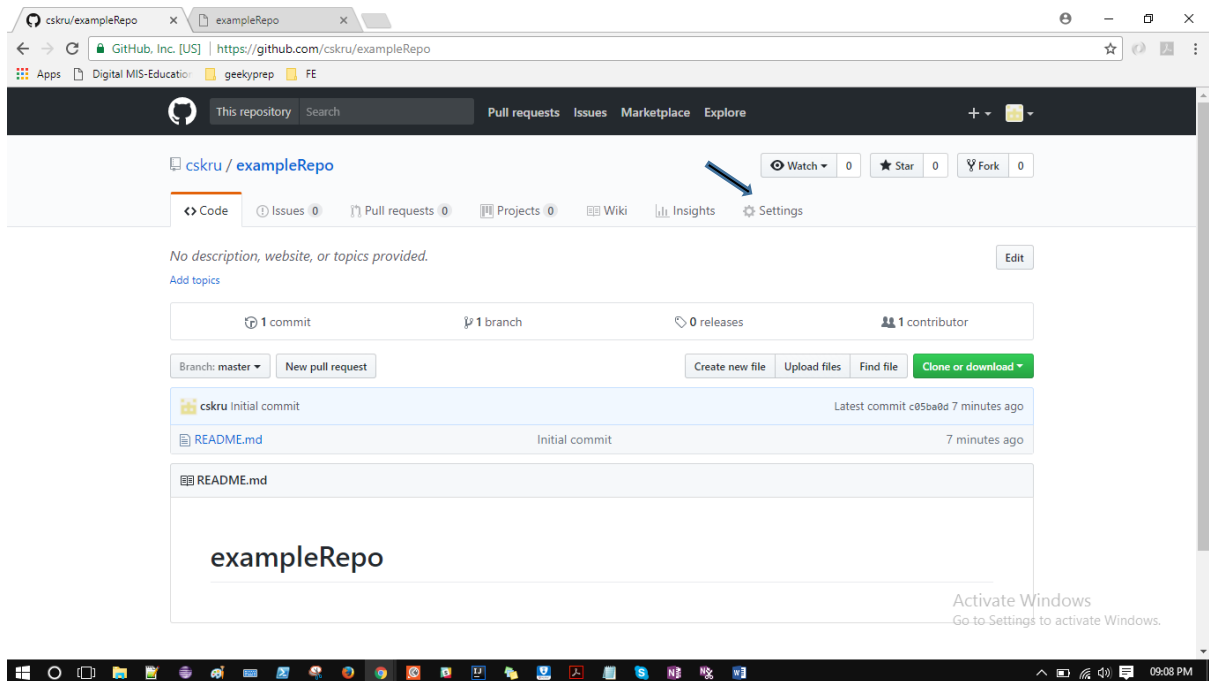
### 1. Login to github and click on new repo



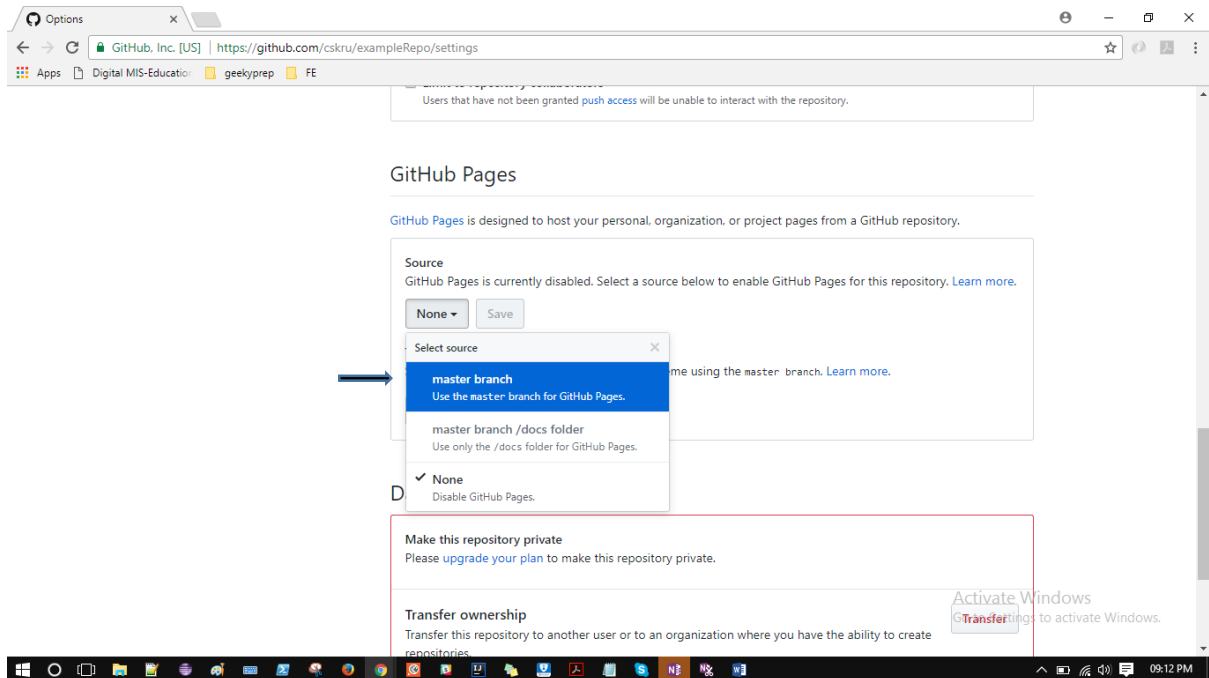
### 2. Give repo name, select public, Check to create Readme and click on Create Repo



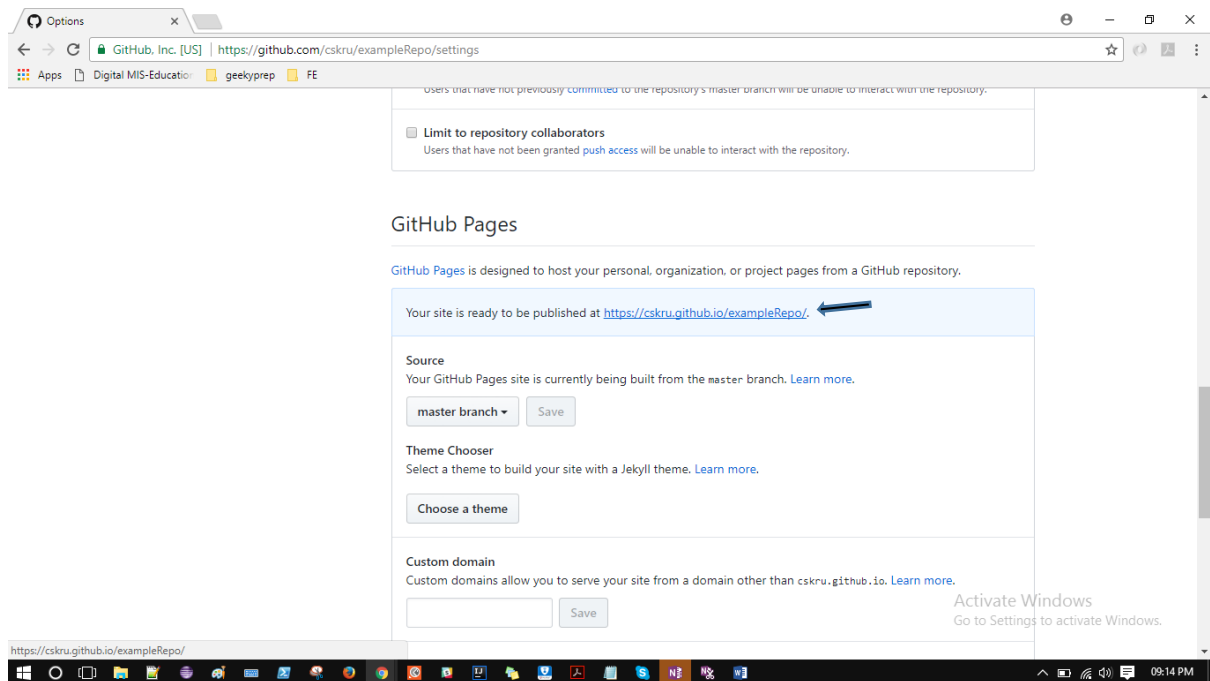
### 3. Click on Settings



### 4. Scroll down to Github Pages, select master branch here and click on Save



- Again scroll down to Github Pages in Settings. Now you can find the url for your repo, by default populated to show readme



- Click on the link, you can see the readme.

### Clone, Add, Commit, Push code to your github repo.

- Install git on your machine and follow default instructions

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

- Configure your git user name and password:

```
$ git config --global user.name "YourGitUserName"
```

```
$ git config --global user.email "YourEmailUsedToCreateGitAccount"
```

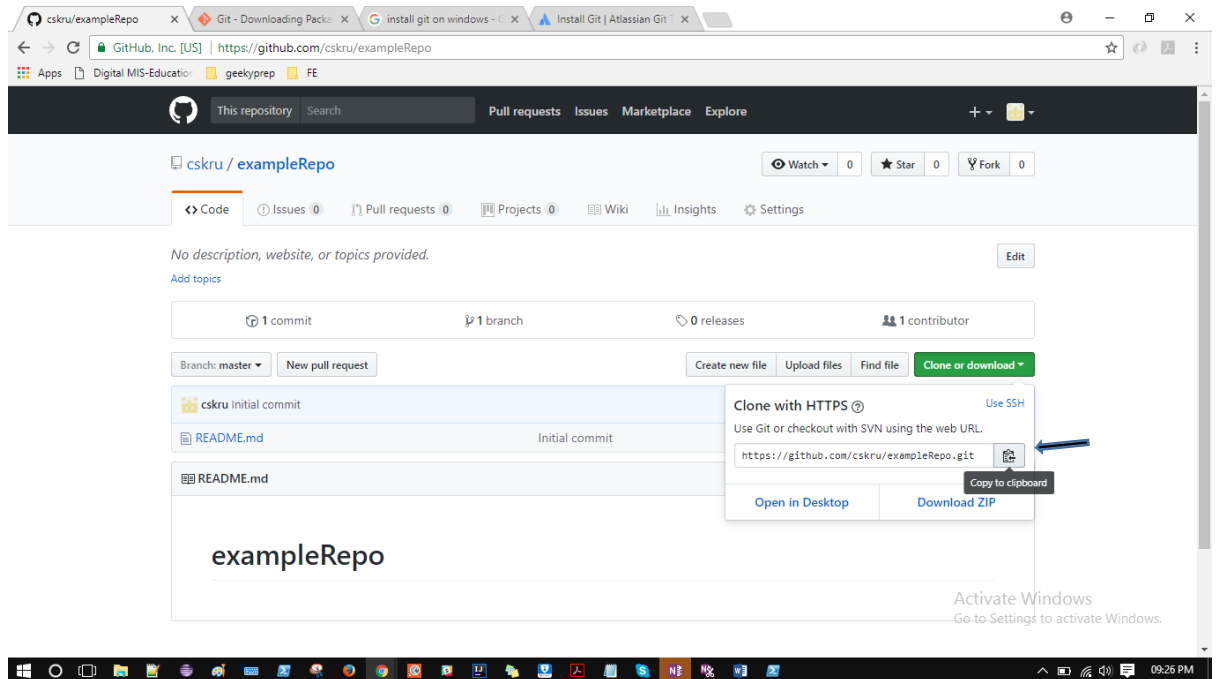
Example:

```
$ git config --global user.name "Emma Paris"
```

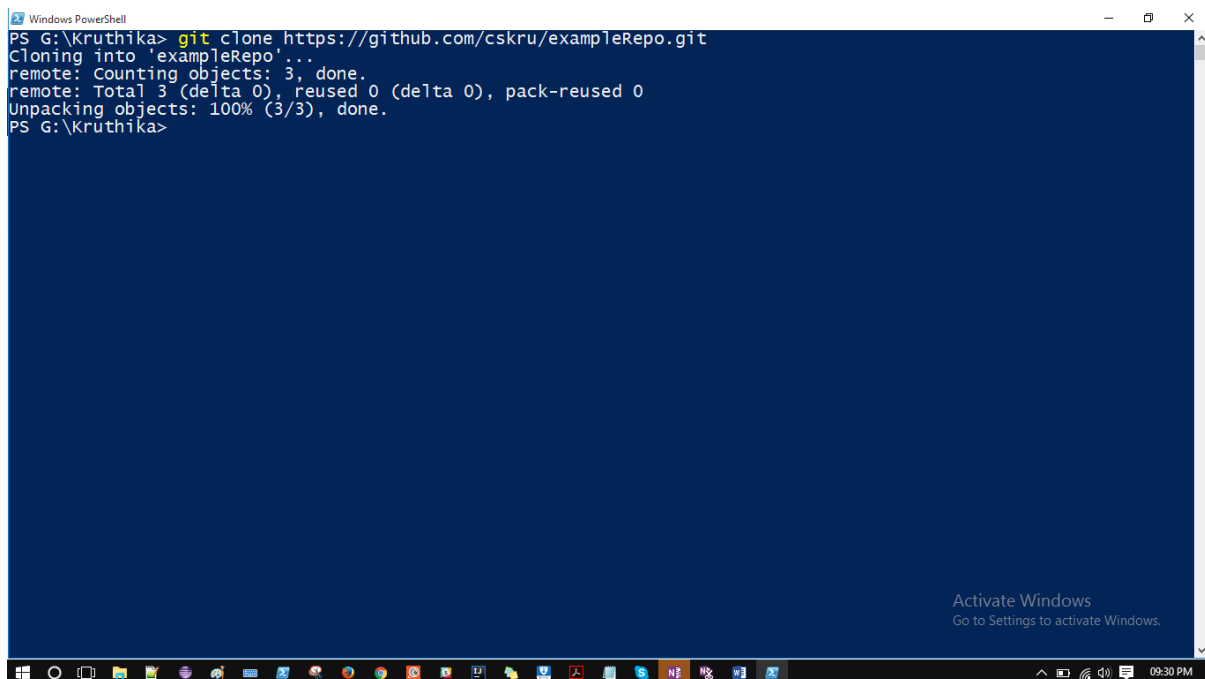
```
$ git config --global user.email eparis@atlassian.com
```

Refer this site in case of doubts: <https://www.atlassian.com/git/tutorials/install-git>

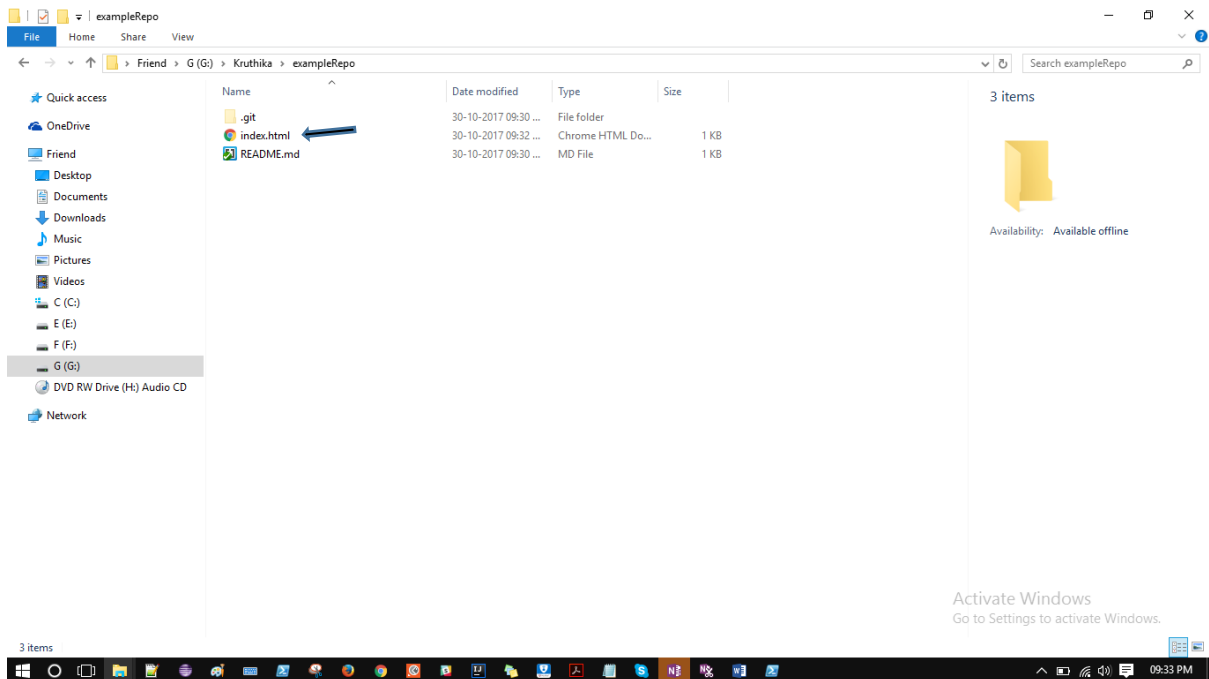
3. Go back to github, Code tab of your repo and copy this clone link:



4. Go to the directory where you want to maintain your project code. Open command prompt / Powershell there, type in `git clone <the link you copied>` and press enter.



5. Once done, go ahead and put your code in the repo (I've created a sample html file. You can paste your entire project directory here)



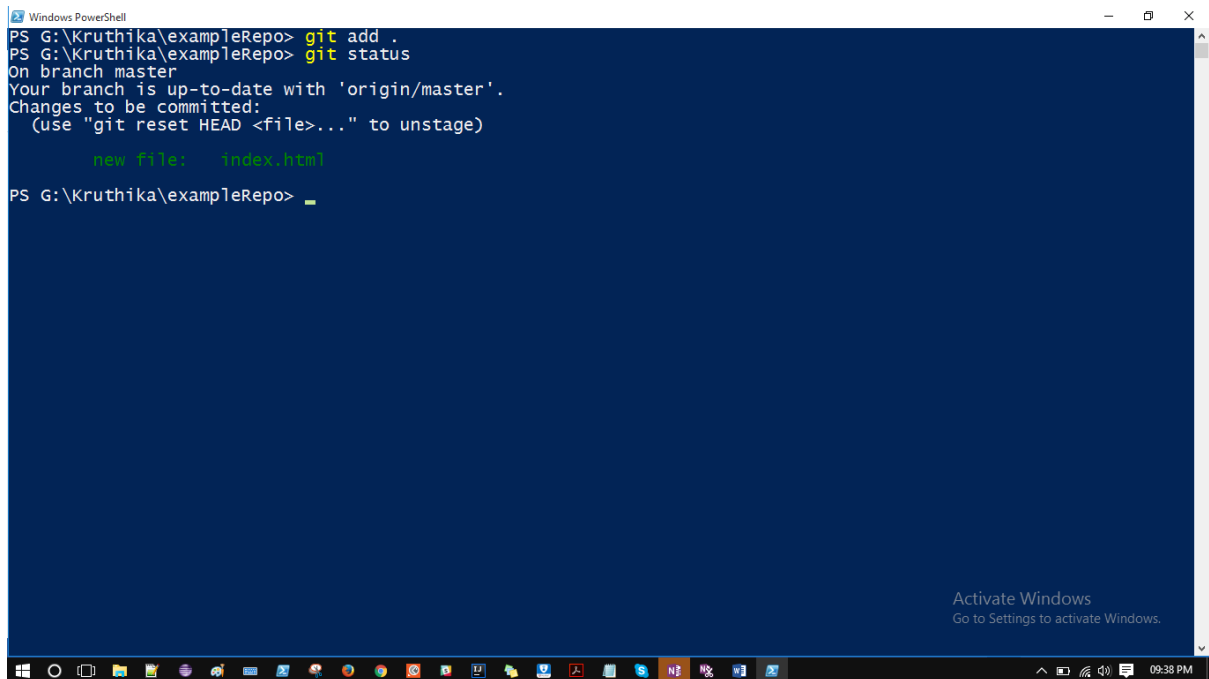
6. Get into your repo (`cd <repoName>`) and do a `git status`. You can see something like this. It means, the version of the project on your local machine has a new file compared to the one on github central server.

```
Windows PowerShell
PS G:\Kruthika> cd .\exampleRepo\
PS G:\Kruthika\exampleRepo> git status
On branch master
Your branch is up-to-date with 'origin/master'.
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)
PS G:\Kruthika\exampleRepo>
```

The screenshot shows a Windows PowerShell window with the above commands and output. The taskbar at the bottom shows the time as 09:36 PM.

7. We'll add this file to our local git repo by "git add ." and then do a git status. This means, the file is now in your local git repo and is ready to be committed to central server



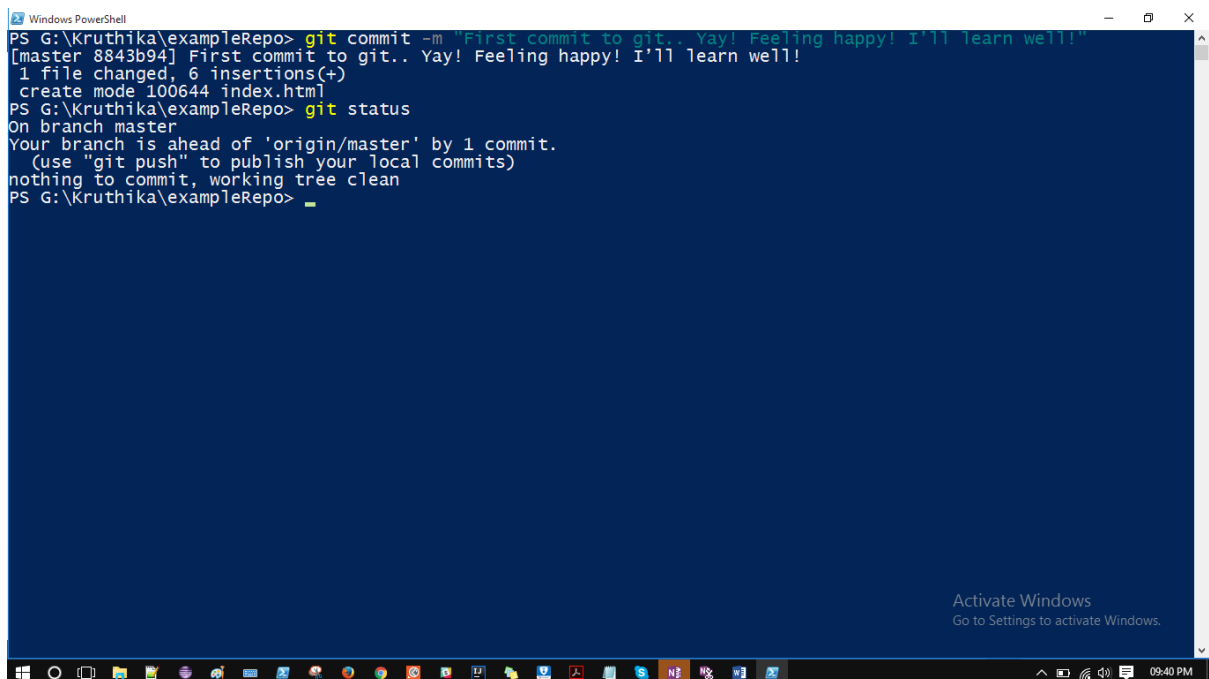
```
Windows PowerShell
PS G:\Kruthika\exampleRepo> git add .
PS G:\Kruthika\exampleRepo> git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   index.html

PS G:\Kruthika\exampleRepo> _
```

Activate Windows  
Go to Settings to activate Windows.

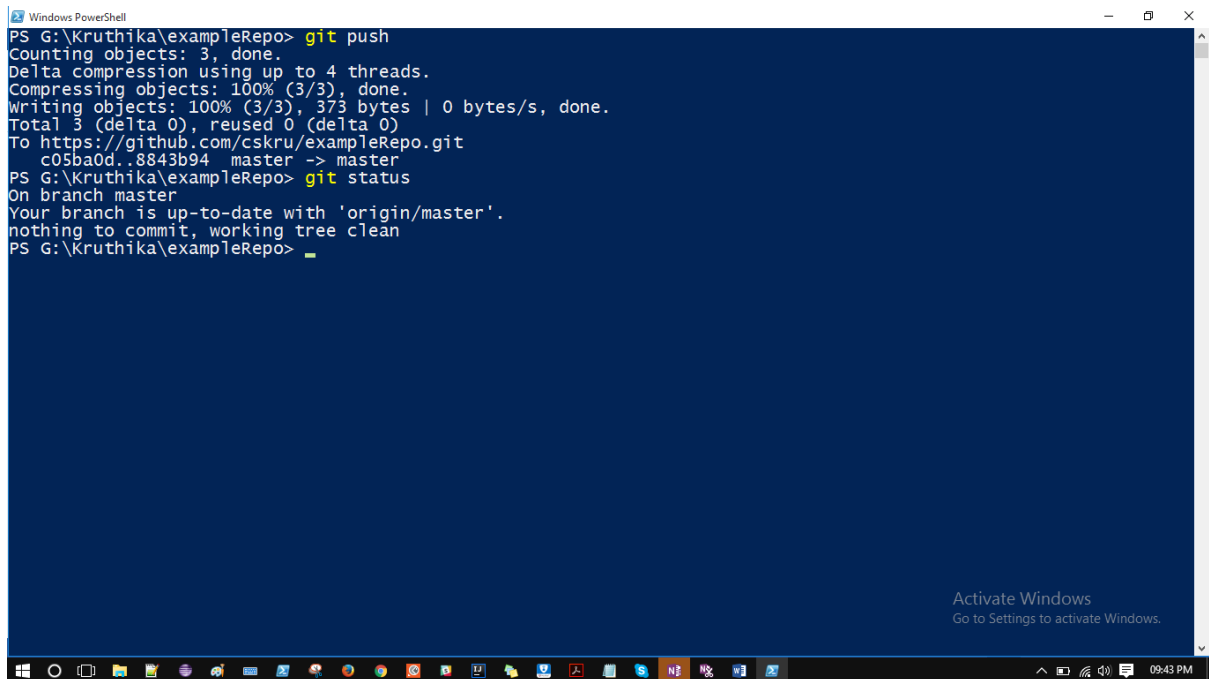
8. Now, we'll commit the file by git commit -m "First commit to git.. Yay! Feeling happy! I'll learn well!"  
Commit should be followed by a message which says the details of what you've done to the files.



```
Windows PowerShell
PS G:\Kruthika\exampleRepo> git commit -m "First commit to git.. Yay! Feeling happy! I'll learn well!"
[master 8843b94] First commit to git.. Yay! Feeling happy! I'll learn well!
 1 file changed, 6 insertions(+)
   create mode 100644 index.html
PS G:\Kruthika\exampleRepo> git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)
nothing to commit, working tree clean
PS G:\Kruthika\exampleRepo> _
```

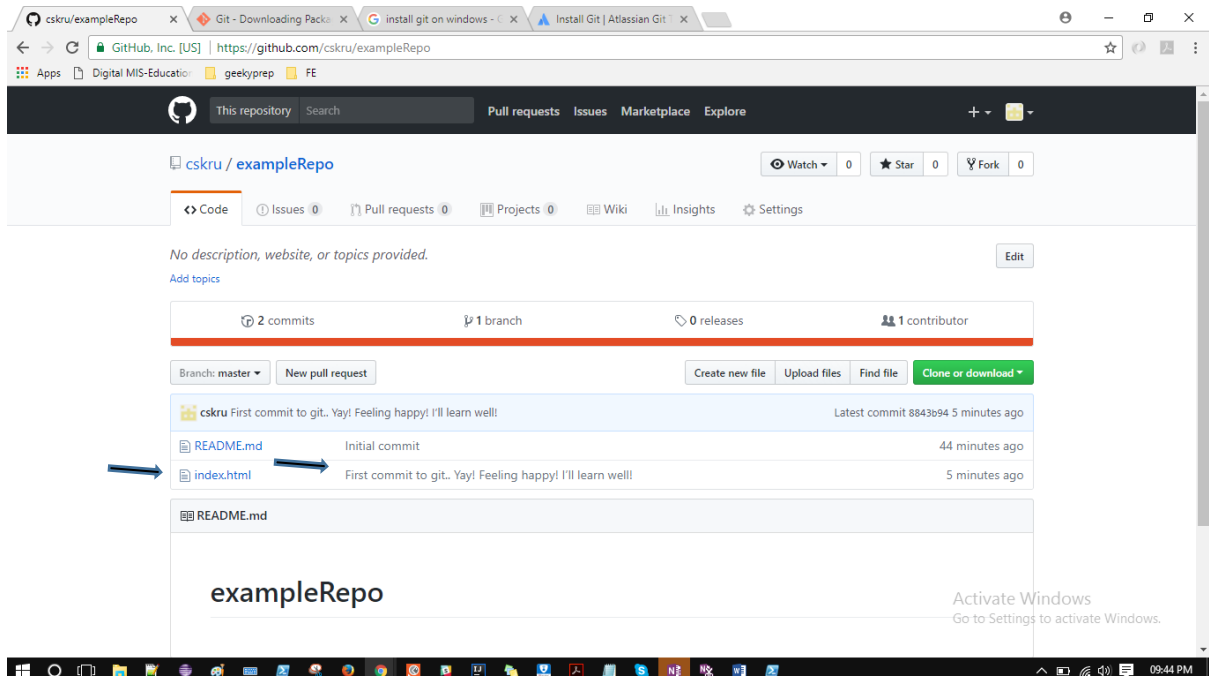
Activate Windows  
Go to Settings to activate Windows.

9. So, now your new file is committed with its seat belts on to travel to the central server 😊.  
Last step is to push the file “git push”

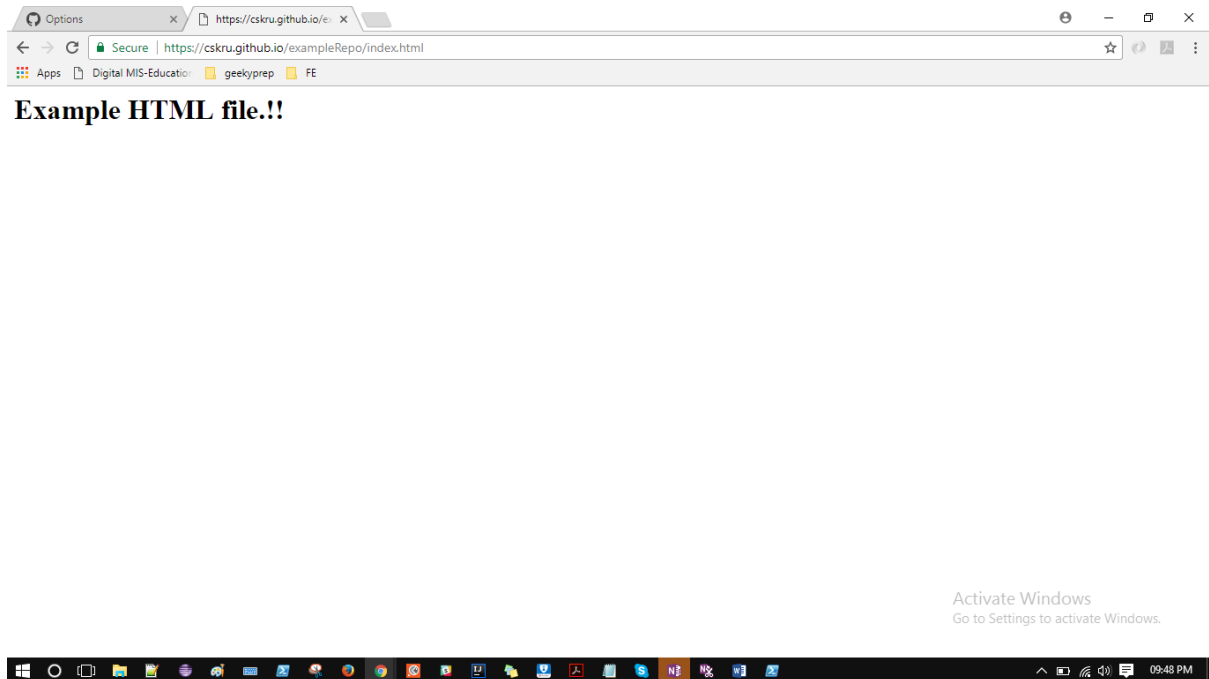


```
PS G:\Kruthika\exampleRepo> git push
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 373 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/cskru/exampleRepo.git
c05ba0d..8843b94 master -> master
PS G:\Kruthika\exampleRepo> git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean
PS G:\Kruthika\exampleRepo>
```

10. The file is now pushed to central server or VCS yay!!!!  
Refresh your github page in a minute. And Boom! Your index.html is here. Yay! With your message 😊



11. Congo! You just created a github repo, github page, pushed your code to repo!  
Hold your breath for the final show, seeing the page as a link 😊
12. If you don't remember the link, go to settings tab, scroll down to Github Pages and grab the link, now append it with index.html  
Example: <https://cskru.github.io/exampleRepo/index.html> and Boom!



**Example HTML file.!!**

13. By default github takes the index.html in the repo, even if you don't mention it.  
I mentioned it for your clarity 😊  
You can have directories here, html pages within it, redirections to it, and what not?
14. Next time you make some changes, do  
git add . (You can mention specific files too)  
git commit -m "message"  
git push  
Generously use git status to know what's going on  
  
So, this is pretty much the basics of git.  
Git is an ocean. Can't cover it in 5 pages.  
But, this is sufficient for you to get started.  
Hope you enjoyed 😊

Warm regards,  
Kruthika

Programming and learning enthusiast 😊

30 Oct 2017, 9:55 PM IST