**Using Github details:**

\*Issue: when trying to push to Github from local Git but got rejected.

> git push origin master --force

Or > git push origin master –f

<https://www.pluralsight.com/guides/using-git-and-github-on-windows>

\*The above link teaches how to use Git (and GitHub) on Windows) – setup

1. Create a local repository

2. Configure with your identity and initialize the Git repository on the folder.

3. Create sample txt file and add followed by commit to the local repository.

4. Link to GitHub using SSH key

5. Create a repository at GitHub and remotely add that folder and push the local copy to the GitHub folder.

\* Refer to the link for the command details.

<https://help.github.com/articles/adding-a-file-to-a-repository-using-the-command-line/>

\*The above link teaches how to add a file (text file or pdf files…) to repository:

Stage the file for commit to your local repository.

>> git add . # add all file in the local repository and stages it for commit

>> git commit –m “Add a new file or change xxx”

>> git push origin master

<https://gist.github.com/blackfalcon/8428401>

\*To ensure the local repository is up to date with the remote repo at GitHub before pushing the modified local copy to GitHub

>> git pull origin master # pull updates; combined both git fetch and git merge. When doing a fetch the resulting commits are stored as remote branch allowing you to review the changes before merging. Merging on the other hand can involve additional steps and flags in the command

<https://git-scm.com/book/en/v2/Getting-Started-Git-Basics>

<https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell#ch03-git-branching>

**SCM** is Software Configuration Management and SVN is a Version Control System tool, which is a subset of **SCM**. VCS are also called Revision Control and **Git**, Mercurial are also VCS tools ( Distributed VCS to be more specific. )

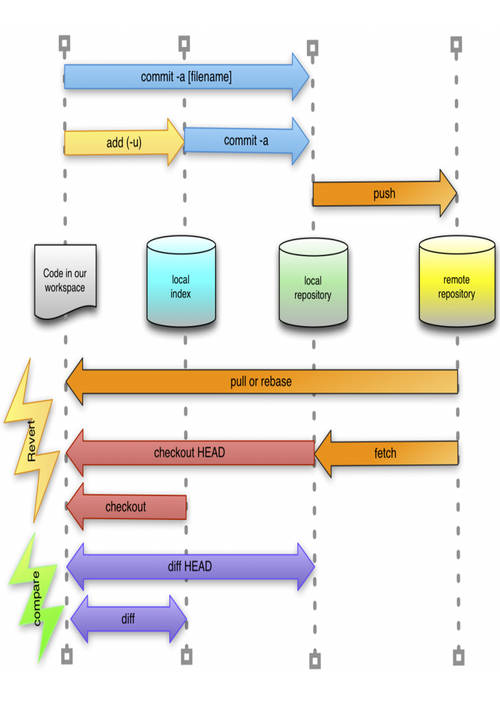
\*You don't need to have a remote repository at all.  
You can have the full git experience, with commits, branches, merges, rebases, etc, with only a local repository.

The purpose of a remote repository (eg, GitHub) is to publish your code to the world (or to some people) and allow them to read or write it.

The remote repository is only involved when you git push your local commits to a remote repository, or when you git pull someone else's commits from it.

 You can commit, branch and party on your own repo on your local machine even without internet access. Then, when you have a connection again, you can push your changes to any other git repo you have access to.





**\*\*To upload an image to README.md,**

The link to be used must be a raw link. To do that, go to the particular repository, click on issue, create new issue, then drag and drop the image file into the Write space given (styling with Markdown), a link with githubusercontent.com will be generated. The link can then be used for the README.md file.

Note: only support .png and other formats but not .tif.

GitHub’s CDN (Content Delivery Network)

Refer: <https://gist.github.com/vinkla/dca76249ba6b73c5dd66a4e986df4c8d>

<a href="Images/Manuscript\_Figure"><img src="Images/Manuscript\_Figure/Manuscript\_Figure1.png" height="300px"/></a>

The above html can be used in the markdown language when referring the image in the same folder.

To Check username and email for Git

git config --list

\*Git and GitHub username and password can be different

**How to create folder in github repository:**

You cannot create an empty folder *and then* add files to that folder, but rather creation of a folder must happen *together with* adding of at least a single file. On github you can do it this way:

* go to the folder inside which you want to create another folder
* click on *New file*
* on the text field for the file name, first write the folder name you want to create
* **then type /**. This creates a folder
* you can add more folders similarly
* finally, give the new file a name, (eg. .gitkeep which is [*conventionally*](https://stackoverflow.com/a/7229996) used to make git track otherwise empty folders, not a git feature though)
* finally click *Commit new file*.