**Git Adding an existing project to GitHub using the command line**

1. Create a new repository on GitHub. To avoid errors, do not initialize the new repository with README, license, or gitignore files. You can add these files after your project has been pushed to GitHub.
2. Open Git Bash.
3. Change the current working directory to your local project.
4. Initialize the local directory as a Git repository.

$ git init

1. Add the files in your new local repository. This stages them for the first commit.

$ git add .

# Adds the files in the local repository and stages them for commit. To unstage a file, use 'git reset HEAD *YOUR-FILE*'.

1. Commit the files that you've staged in your local repository.

$ git commit -m "First commit"

# Commits the tracked changes and prepares them to be pushed to a remote repository. To remove this commit and modify the file, use 'git reset --soft HEAD~1' and commit and add the file again.

1. At the top of your GitHub repository's Quick Setup page, click to copy the remote repository URL.
2. In the Command prompt, add the URL for the remote repository where your local repository will be pushed.

$ git remote add origin *remote repository URL*

# Sets the new remote

$ git remote -v

# Verifies the new remote URL

1. Push the changes in your local repository to GitHub.

$ git push origin master

# Pushes the changes in your local repository up to the remote repository you specified as the origin

**Deleting File from remote repository**

Use [git rm](https://git-scm.com/docs/git-rm).

If you want to remove the file from the Git repository **and the filesystem**, use:

git rm file1.txt git commit -m "remove file1.txt"

But if you want to remove the file only from the Git repository and not remove it from the filesystem, use:

git rm --cached file1.txt git commit -m "remove file1.txt"

And to push changes to remote repo

git push origin branch\_name

### [**Pulling changes from a remote repository**](https://help.github.com/en/github/using-git/getting-changes-from-a-remote-repository#pulling-changes-from-a-remote-repository)

git pull is a convenient shortcut for completing both git fetch and git mergein the same command:

$ git pull *remotename* *branchname*

# Grabs online updates and merges them with your local work

Because pull performs a merge on the retrieved changes, you should ensure that your local work is committed before running the pull command. If you run into [a merge conflict](https://help.github.com/en/articles/resolving-a-merge-conflict-using-the-command-line) you cannot resolve, or if you decide to quit the merge, you can use git merge --abort to take the branch back to where it was in before you pulled.

**Git Init** - To create a new repo, you'll use the git init command. git init is a one-time command you use during the initial setup of a new repo. Executing this command will create a new .git subdirectory in your current working directory. This will also create a new master branch.

Once you have a remote repo setup, you will need to add a remote repo url to your local git config, and set an upstream branch for your local branches. The git remote command offers such utility.

git remote add <remote\_name> <remote\_repo\_url>

This command will map remote repository at <remote\_repo\_url> to a ref in your local repo under <remote\_name>. Once you have mapped the remote repo you can push local branches to it.

git push -u <remote\_name> <local\_branch\_name>

This command will push the local repo branch under <local\_branc\_name> to the remote repo at <remote\_name>.

**Git clone** - git clone is used to create a copy or clone of remote repositories. You pass git clone a repository URL. If you used git clone to set up your local repository, your repository is already configured for remote collaboration. git clone will automatically configure your repo with a remote pointed to the Git URL you cloned it from. This means that once you make changes to a file and commit them, you can git push those changes to the remote repository.

Git add git commit - see 5 & 6 above