Using git

Git and github are for version control. Version control is used for tracking changes at various stages so you can undo changes to a specific time in the past and keep record of what you have done.

In this tutorial, we will use git and github to track changes to a document. It can, and is most often, used to track changes to codes. I do not use a code here because I want to focus on something that every possible viewer watching this video is familiar with. But, like I said earlier, the same knowledge learned in this video can be used for codes or any type of work that you would like to track changes.

1. Go to github and sign up
2. Create a new repository
3. Create a read\_me file and commit
4. Install git from <https://git-scm.com/>
5. Open gitbash console
6. Configure your github as follows:
7. git config --global user.name "git username"
8. git config --global user.email "email address used at github"
9. Navigate to the folder in which you would like to put your tracked file
10. Navigate to folder of project you want to version control
11. Rm a file > rm filename.ext

Making a folder linked to git:

1. 3 stages: modified, staged and committed
2. Go into folder you want to add to git and > git init
3. To get the status of all files in the folder> git status

red means changes have been made but not yet staged

1. To stage > git add filename.ext ( to add all file in a folder> git add .)

If you get status now> git status

The files you staged will now be green

To remove from staging area> git rm --cached filename.ext

1. To commit > git commit -m “descriptive message”
2. To get info on all commits > git log –oneline
3. To checkout an earlier commit:
4. Get alphanumeric key from number 3
5. Checkout > git checkout alphanumeric key
6. Go back to master > git checkout master
7. To create a new branch> git branch branchname
8. View all branches> git branch -a
9. To checkout branch> git checkout brachname
10. To create and checkout branch> git checkout -b branchname
11. To delete branch > git branch -D branchname
12. To merge branches: Enter the branch you will like to merge another branch into and merge

:>merge branchname

1. Copy the repository address and push in cmdr from directory containing files to push
2. :> git push address branchname (master or other branch)
3. To avoid using the address every time you will like to push, we can rename it as:
4. git remote add origin [address]
5. then to push , you just use:
6. git push origin [branch to push to github]
7. Github:
8. If no account sign up
9. Create a new repository