# Sessions 1-5

## #01 - What's Git & GitHub & Why U Should Learn Part 1

- **Git** is the most commonly used version control system. Git tracks the changes you make to files, so you have a record of what has been done, and you can revert to specific versions should you ever need to. Git also makes collaboration easier, allowing changes by multiple people to all be merged into one source.
- **Repository** is the folder where your project is kept
- **Github** is a website to host your repositories online
- Github simplify using Git
- Git has GUI

### Why you must learn Git?

- Developers contribute to the same project, in other words devs work on the same project and not on a separated one
- You can revert specific changes
- You can collaborate to fix issues
- You can collaborate to create new features
- You can solve conflicts
- You can organize features

## #02 - Whats Git & GitHub & Why U Should Learn Part 2

### Words You will Hear

- Repository
- Branch
- Local Repo

- Remote Repo
- Commit [Snapshot or Checkpoint in your local repo]
- Clone [From Local or Remote]
- Push [Upload local changes to remote]
- Pull [You pull changes from remote repo to your local]
- Pull Request [Tell other about your changes to pull it from local to remote. You can discuss changes with collaborators and add follow-up commits before your changes are merged into the base branch]

## Important Notes

- Create Repository for every project, don't merge all projects in one repo.
- Create a new branch for every feature or enhancement.
- No need to connect to remote repo when working.
- anyone can *push* and *pull* depending on permissions.

# #03 - Create Github Repository And Clone It

- First create the repo on Github

Note: .md is a markdown file

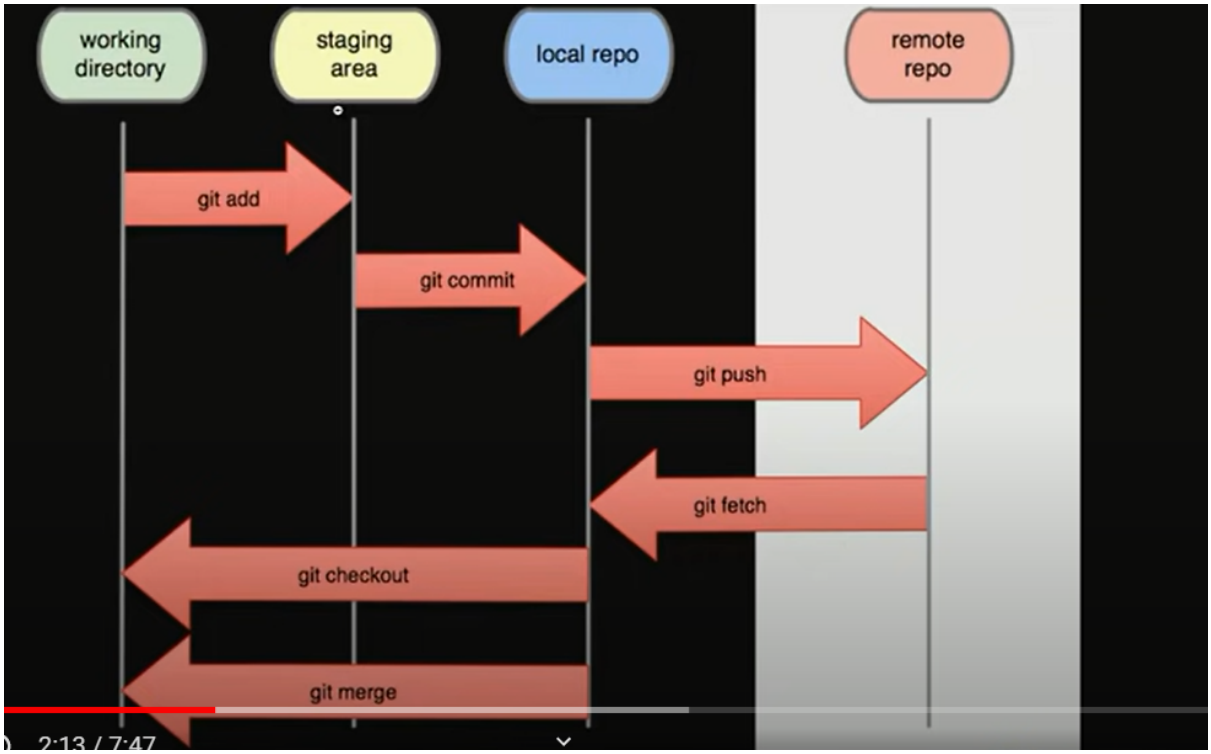
- You can check basic writing and formatting syntax for ReadMe page https://docs.github.com/en/github/writing-on-github/getting-started-with-writing-and-formatting-on-github/basic-writing-and-formatting-syntax
- In your directory clone the github repo

```
git clone repo-link
```

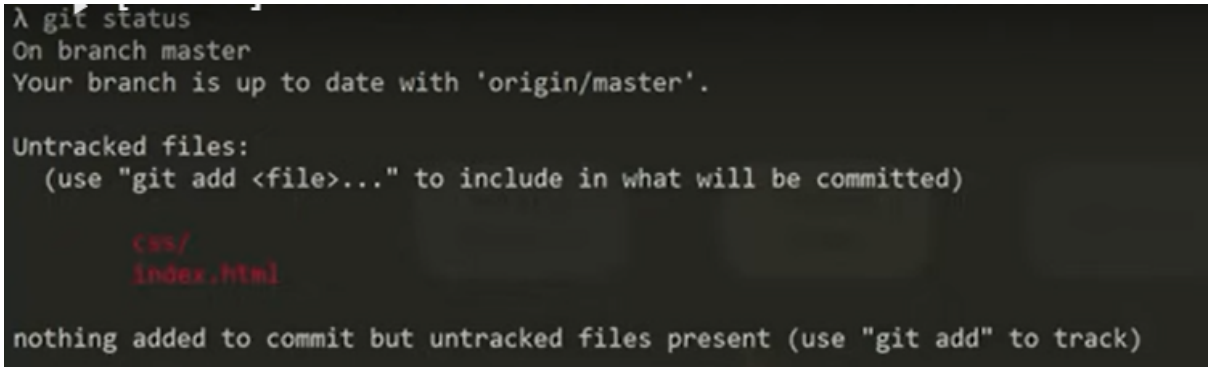
## #04 - Add And Reset And Commit & Explain Progress

- To check what is going on in the working directory, from new added folders to modified files, we use a command called

```
git status
```




- To add untracked files




```
#1st option you can add each file individually
git add css\main.css index.html

#Or add all files at one time
git add .
```

- After the branch is up to date we can now commit the changes in the local repo

- If you want to unstage file you can use

```
#This command will bring the index.html file back to the untracked file
git reset head index.html
```

- Now that we added the need files to the staging area, we need to commit them in our local repo

```
git commit -m "I created the new files main.css and index.html"
```

## #05 - Push Local Changes To Remote Repository

- Now that we made our commits, we need to push those commits to our remote repo
- But before pushing, we need to know what is the name of our branch and what is the name of our remote repo

```
#to know the name of our branches in the directory
git branch

#to know the name of our remote repo
git remote -v
```

- Now you can push to the branch(master) at the remote repo(origin)

```
git push origin master
```

- At this stage you have to enter your credentials