GIT & GITHUB TUTORIAL:

GIT:

What is Git?

- ⇨ It is a Free and open-source version control system
- ⇨ It is a tool that tracks the changes in your code overtime.

What is version control:

- ⇨ The management of changes to documents, computer programs, large web sites and other collections of information.
- ⇨ Some terms regularly used:
    - Directory -> Folder
    - Terminal or Command line -> Interface for Text Commands
    - CLI -> Command Line Interface
    - Cd -> change directory
    - Code Editor -> Word processor for writing code, e.g., VS-code
    - Repository -> Project, or the folder/place where your project is kept.
    - GitHub -> a website to host your repositories online.

Some Basic Git commands:

- ⇨ All the commands are in the lowercase unless explicitly written in capital
    - Clone -> Bring a repository that is hosted somewhere like GitHub into a folder on your local machine.
    - Add -> Track your files and changes in Git.
    - Commit -> Saves your files in Git.
    - Push -> Upload Git commits to a remote repo, like GitHub.
    - Pull -> Download changes from remote repo to your local machine, the opposite of push.
    - Branch -> Shows in which branch you're working.
    - Checkout -> To switch/create new branches.

How to create a new repository from GitHub:

- ⇨ All the commands are in the lowercase unless explicitly written in capital
- ⇨ Step 1:
    - Hover over the (+) drop down and click on new repository
- ⇨ Step 2:
    - Enter your repository name and description.
- ⇨ Step 3:
    - A new repository is created.
    - Always create a README.md file in the repository manually.

How to import, update and push the repository:

- Step 1: All the commands are in the lowercase unless explicitly written in capital
  - Always check git version using the git bash or cmd.
  - **Git –version**
- Step 2:
  - **Git clone (Enter the link of the repository provided on the GitHub)/(SSH link generated)**
- Step 3:
  - Make the changes on the local machine and check the status of the file using the following command.
  - **Git status**
- Step 4:
  - Then if you create a new file in the repository locally; the file is then added to repository locally.
  - This tracks the file in the repository.
  - **Git add .(period) or Git add filename.extension**
- Step 5:
  - We commit the changes made with the command.
  - **Git commit -m "Changes to the Title of the file" -m "Description of the Heading" -m stands for message.**
- Step 6:
  - We push the changes on the internet using the command.
  - **Git push origin (branch)**

Perquisite to push your local repository:

- All the commands are in the lowercase unless explicitly written in capital
- To push your repository under "XYZ" account it needs authentication/proving to GitHub.
- Therefore, connect your local machine to your GitHub account using SSH Keys.
- Step 1:
  - Type the following command:
  - **Ssh-keygen -t rsa -b 4096 -C "email@example.com"**
  - Enter the passphrase as needed or keep it empty.
  - Note the location of the key stored at.
- Step 2:
  - Navigate to the folder/directory where the SSH key is stored.
  - Using "**ls | grep testkey**" on Linux/macOS search for testkey.pub
- Step 3:
  - Copy the testkey.pub
  - Testkey.pub is the key that is going to be uploaded on GitHub's interface
- Step 4:
  - Navigate to SSH/GPG on GitHub
  - Paste the copied public key by adding a new ssh key on the browser.
  - And confirm the access.

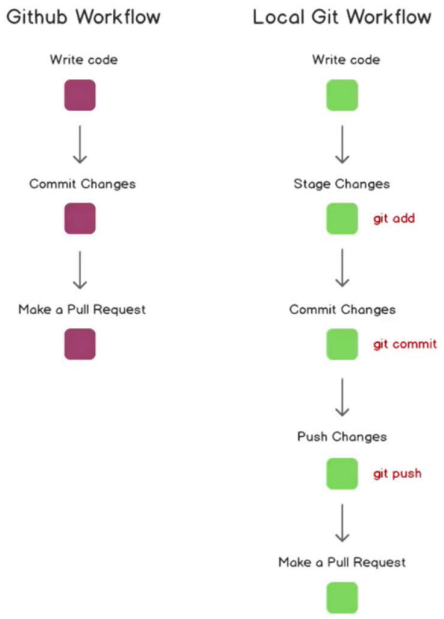
How to create a repository locally and push:

- All the commands are in the lowercase unless explicitly written in capital
- Step 1:
  - Create a new folder and type "**git init**", this initializes an empty git repository.

- Step 2:
  - Fill the details in the local folder/directory and commit the changes locally.
  - To perform this step, refer step 4 and step 5 of how to import, update, and push.
- Step 3:
  - Create a new repository on the GitHub and copy the link.
  - Use "**git remote add origin (copied link)**"
  - This connects the empty repository on the GitHub to the local machine.
- Step 4:
  - Use "**git remote -v**" to check if the connection is made or not.
  - Use "**git push location(origin) branch(master)**" to push the repository on the GitHub.
  - Use "**git push -u location(origin) branch(master)**" to setup an upstream for that branch.

Workflow of the GitHub and Local machine:

- The workflow is shown as follows:



How to create and delete a branch:

- All the commands are in the lowercase unless explicitly written in capital
- Step 1:
  - Use "**git branch**" to see which branches are created.
- Step 2:
  - To create a new branch.
  - "**git checkout -b (branch_name)**"
- Step 3:
  - To delete a branch.
  - "**git branch -d (branch Name)**"
  - "**git branch -D (branch Name)**"
- Note:
  - All changes made to an individual branch will not be reflected in another branch unless merged.

- To switch into different branches:
- "**git checkout branch_name**"

How to create a pull request:

- All the commands are in the lowercase unless explicitly written in capital
- Step 1:
  - Create a new branch
  - Perform all the changes in that branch.
- Step 2:
  - Commit and push the changes on the GitHub.
- Step 3:
  - Check the difference between the main branch and the feature branch.
  - If there are any differences, switch to main branch.
  - "**git diff (branch_name)**"
- Step 4:
  - Merge the main branch with the feature branch.
  - "**git merge (branch_name)**"

How to update your local repository:

- All the commands are in the lowercase unless explicitly written in capital
- Step 1:
  - "**git pull origin master(branch_name)**" to update changes on local machine that are made on GitHub.
  - "**git pull**" if the upstream settings have been mentioned already for that branch.

How to upload large files that are over 100mb:

- All the commands are in the lowercase unless explicitly written in capital
- Step 1:
  - "**git lfs install**" install the git large file storage to upload.
- Step 2:
  - **git lfs track "*.ext"**
  - To track the large files that you wish to upload on GitHub.
- Step 3:
  - To upload the files on the repository, follow the commands below:
    - **git add file.ext**
    - **git commit -m "Add design file"**
    - **git push origin main**