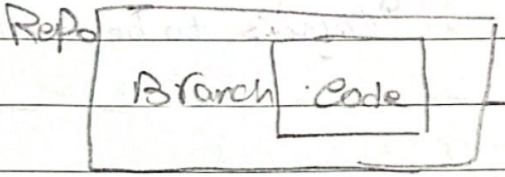
⇒ Before starting in Project → we must know git and github:-
↳ With git basics

→ We start with git and github → they are different → Web based → Hosting and Collaborating Provider.
↳ Local
Local Version Control System

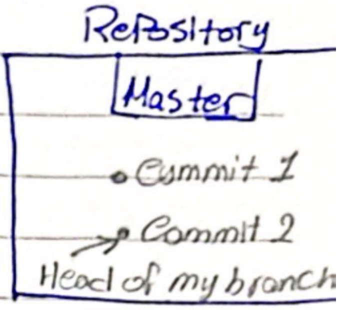
Repositories, Branches, Commits (inside)
- Repositories: The location where your code History is stored.
- Branches: Folder inside Repository contain Commits
- Commits: Different stages of your code ↓ Inside Branch



→ this is How it Works

⇒ First: git init → to initialize a repository in your project path.
⇒ Second: git status → to check for files in your project whether they are added or Not. (Not tracked files).
⇒ Third: git add. → it will add all of our files of the folder.
⇒ Fourth: git commit -m → means add this code as version of the code we can save in our branch and we have to give it a name
↳ message

Note: Now our code is at master branch and saved in it



⇒ We know this By → (git branch) command
↳ we Have git log → to get History of Commits.

⇒ if you want to get back to a commit ⇒
Head also pointing to this Commit ← git checkout "ID" → you get it from (log)

⇒ by git Master → we go back to our Last Commit.
↳ OR git Checkout master.

⇒ To Delete a Commit ⟹ git reset --hard "ID" (of the Previous Commit)
↳ Permently.

⇒ Delete unwanted Code done without Saving by ⇒ git Checkout
-- .

Branch Features

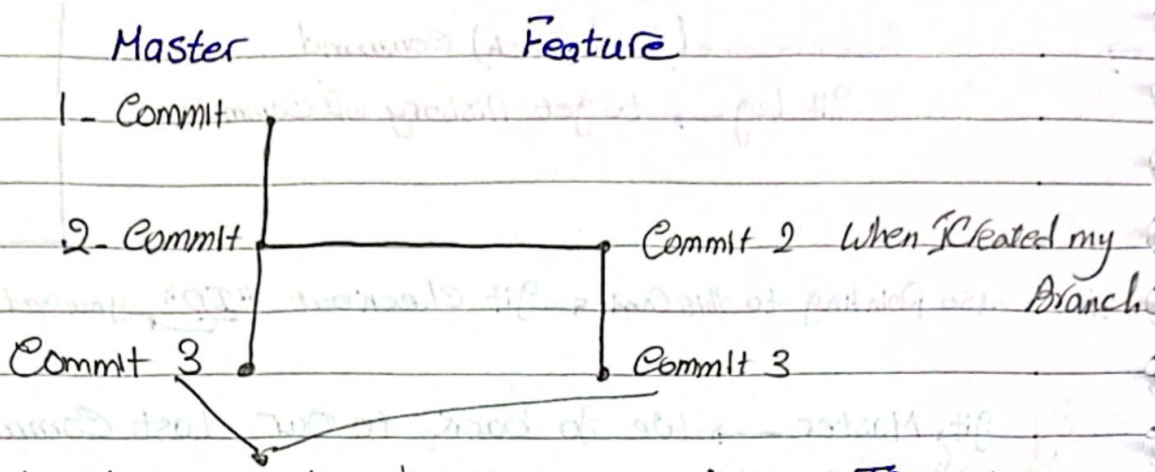
⇒ To Create a Branch :- git Checkout -b "Name" ⇒ and Switch to it.
↳ it have all Commited files in it.

⇒ To Switch Between branches:-
" git Checkout "Branch name" " Master Feature/branch

⇒ To merge files of Branch and master, 2 Commit —— 2 Comm
We make Head on master then → git merge "Branch Name" (w
ـــــــ> لو ضفنا اى Commit للبرانش الجديد هيكون كدا. 3 Com

⇒ To Delete a branch ⇒ git branch -D "name of branch"

Note :-



⇒ When I try to merge them it will make conflict → IDE will make us choose only one in our Branch.