JAVA

Git and Github:

*Git is a version controller for our code

*Github is used as version controller as well as for collabration.

* "git init" is to initialize a git to store the changes in .git folder.

* "git status" is use to check what are all changes you done to the project but not save. This is called untracked.

* "git add" is to add the file to stage before committing

* "git add ." is to add all file.

* "git commit -m "message-of-what-changed"" is used to commit to what is change.
  ↓ message

* To remove the folder or file outside the stage "git restore --staged file-name".

* To see all the history "git log".

* To remove the commit of unnecessary things copy the commit no. of last next commit "git reset <past the commit no.>". this will go to the commit remove above that.

* If you don't want to commit and don't want to loose it. you can use "git stash". It is like a backstage.

* "git stash pop" will bring the files in backstage to front stage (or) close stage.

* To clear the backstage. use "git stash clear"

* To add github repository to the system to save changes "git remote add origin <repo-link>".

* Origin is like here is the repo-link.

* To push the commit to the repository "git push origin <branch> ~~[illegible]~~"

* You should only commit changes to main (or) master branch if it is finalized.

* "git branch <branch-name>" it will create a branch with branch_name.

* To keep to commit to only one specific branch "git checkout <branch-name>"

* To make the code merge to ~~to~~ main branch "git merge <branch-name>"

* To clone the repository in your system use "git clone <github-link>

* Upstream = forked the other user repo

* one branch have one pull request . if you accept the pull the request and commit , it will directly commit to it.

+ If the pull request is accepted and working on next feature create a new branch and new. pull request .