Javascript :-

## Git / Github.

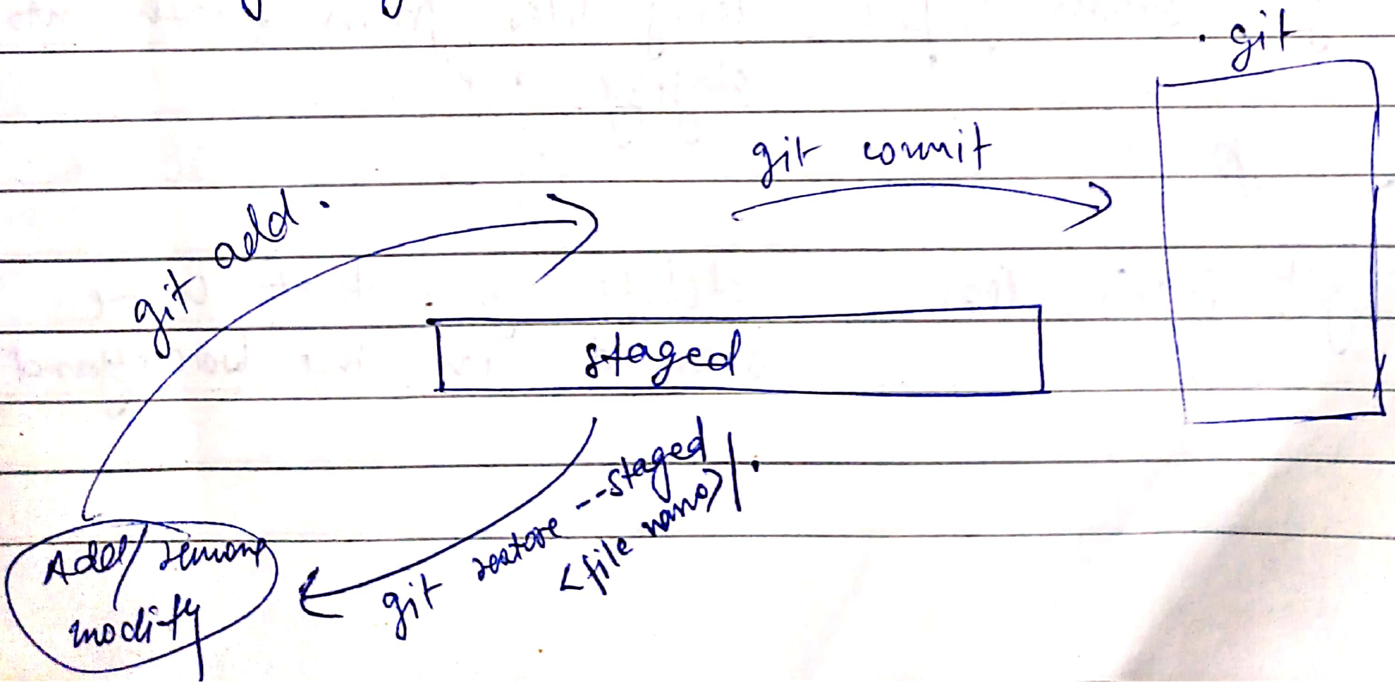
- version control system.
- maintain history of project, time, people wise.

- .git ⇒ this folder contains all the history of the project.

```
ls -a   ⇒ show all hidden files
ls .git → show files inside .git folder
```

real life analogy :-

wedding day



git log → history of all commits ,

# To remove a git, or all previous commits uptill a certain time in history.

- Just copy the hash value of the commit beneath it. (in git log)

& do: git reset <hashvalue>.

⇒ git stash:- shifts staged files into a stash that can be loaded later on.

↓

git stash pop: bring files from stash into staging area.

↓

git stash clear : delete files that were stashed in the backyard,

## * How to attach a repository to a local folder?

→ git remote add origin <url_of_repo>

This name is assigned to this URL.

→ git remote -v // lists all urls attached to this local folder.

→ git push origin master/main → the branch.

↳ the url to which we want to push.

## Inside .git folder:-

ls .git ↳

- HEAD : pointer to the current branch.

# git branch <name> . // creates new branch

# git checkout ~~feature~~ <name> // now this branch is active & currently in use. (head folder points to new branch)

* git merge <name> //

⇒ If a branch already has a pull request associated with it, it will not allow you to create a new pull request.

⇒ Since one branch can only open 1 pull request (which is not merged), ∴ if want to raise multiple pull requests, we would need to create multiple branches.

# avoid commiting on the main branch, Rather raise PR's & then merge to main branch. (← best practice).

→ fork :- used to make a copy of project in your own github account

→ origin is our personal git repo.

* git push origin Shivam -f

(when online repo has a commit ahead of our local, then we need to force push)

* git fetch --all --prune

↓ all branches

↳ fetch deleted ones also.

* git reset --hard upstream/main.

- making our local main branch equal to the upstream. (commits wise)

* git pull upstream main

⇒ pull also does the same thing internally as fetch.

# new branch can be created from any other branch. (nesting of branches)

* Squashing commits:- merge many commits into 1 single commit.

→ git rebase -i <hash-of-that_commit>

↓ interactive environment.

↳ change pick → s accordingly.

## Merge conflicts :-

→ if same lines are changed by 2 commits by 2 diffrent branches. & if both are trying to merge into 1 branch. (like main).