Why Git ?

To make the latest code and changes available for others

* Change Management
* Collaboration

Project :

VCS → Version Control System

Git → (VCS)

- It is distributed
- All members of team can access latest code using Git

GitHub

- GitHub is a Server that uses Git

* Bitbucket
* GitLab

What is Repository?

- A folder/directory where we keep all files related to a project

Local Repository	Remote Repository
• On User's personal computer	On internet, for others to access and store latest updates • Backup • Collaboration

readme.md

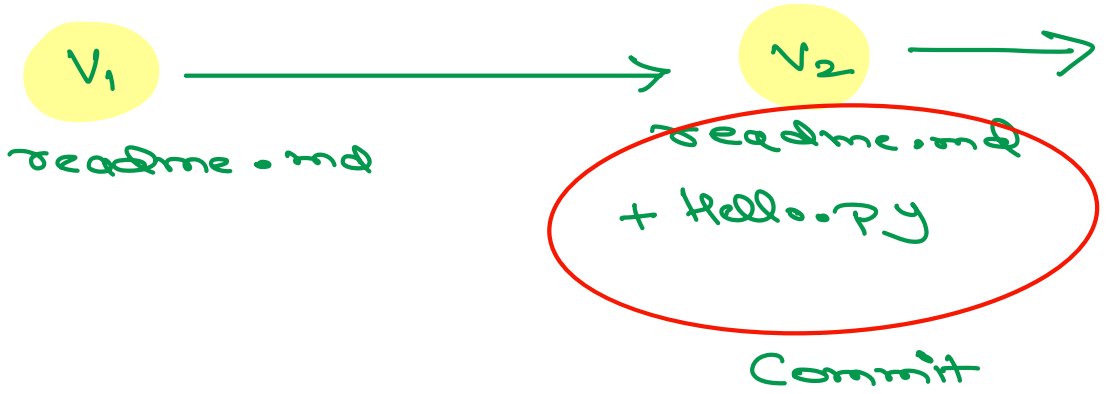
- Minimal information about project, how setup, install and use it with relevant links to documentation paper etc.

gitignore.md

- Name/path of file which we want to ignore/Not become part of repository

.config
.env
model.pkl
*.pyc
__pycache__

* Commit



* Push

① Pushes all the commit to remote repository (origin)

* Clone: Downloads Everything if repo is not present on your local system, you need to clone the repo from github

* Pull

① Only download the difference between Local and remote

② Commits the files as well

* Fetch

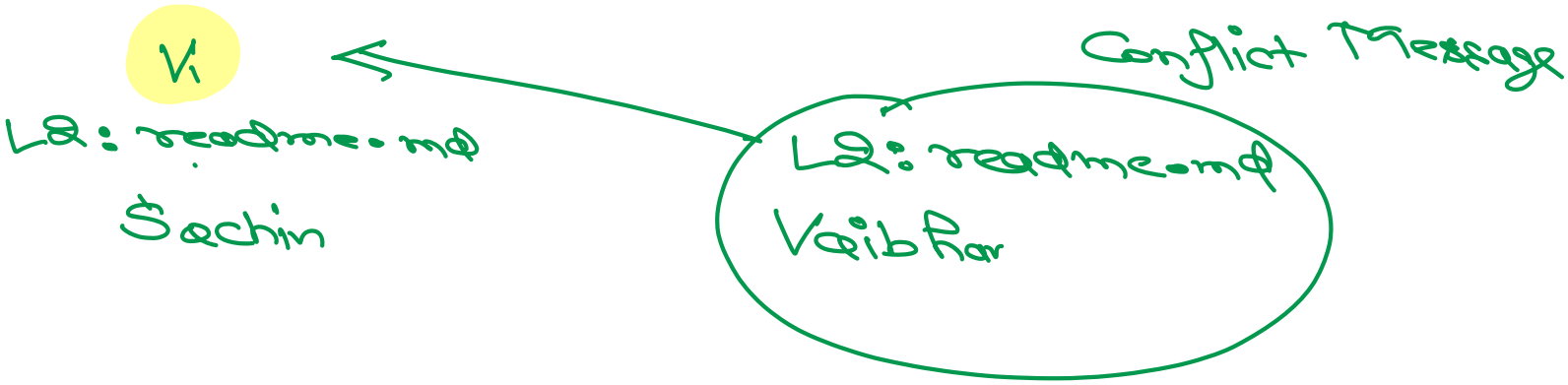
① Also downloads latest changes

② Doesn't Commit

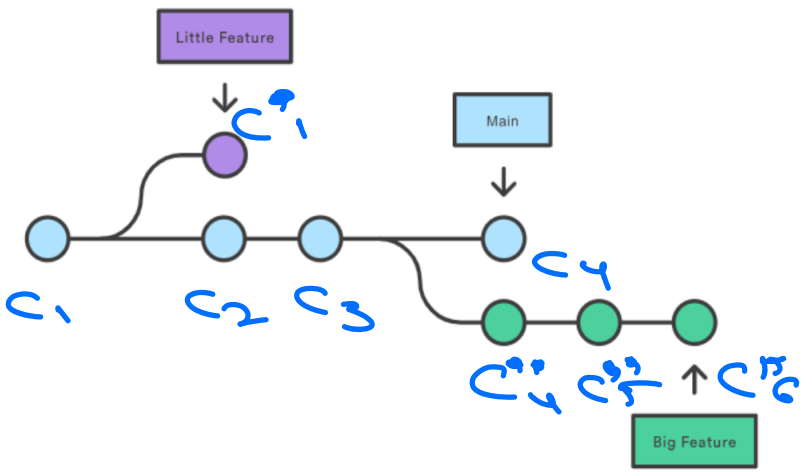
* Conflict: Code changes in same file

V and S

Ex: readme.md



* Branch



* A pointer to the snapshot of your own changes
* Whenever someone want to work on new feature or fix a bug they should create a new Branch
* Once the code is finalized on branch and tested throughly we can MERGE the branch to main

Steps for collaborative Development :

1. Pulling Code → clone / Pull
2. Create Branch
3. Make changes in new Branch
4. Commit and publish / push the new Branch
5. Merge → raise Pull Request
6. Owner can review 'PR' and merge

* Forking

Staging Area

* Modified
* Staged : Added for Next Commit
* Commit : Committed

Token from account

git remote set-url origin https://<githubtoken>@github.com/<username>/<repositoryname>.git

username

repository to Publish into

git push -u origin main

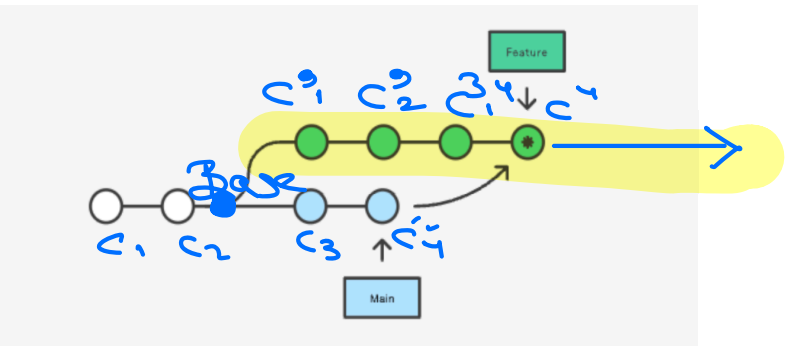
1) First time you need to mention branch

2) git push

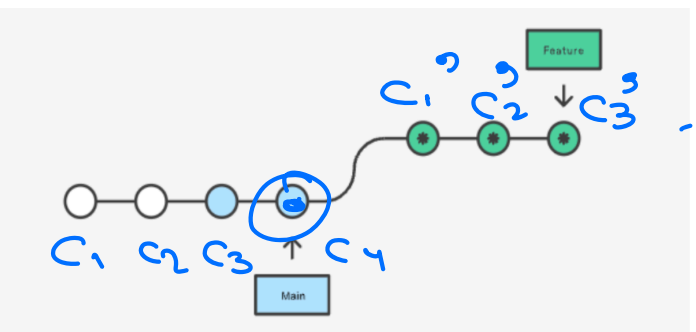
Git Pull

Merge Scenario

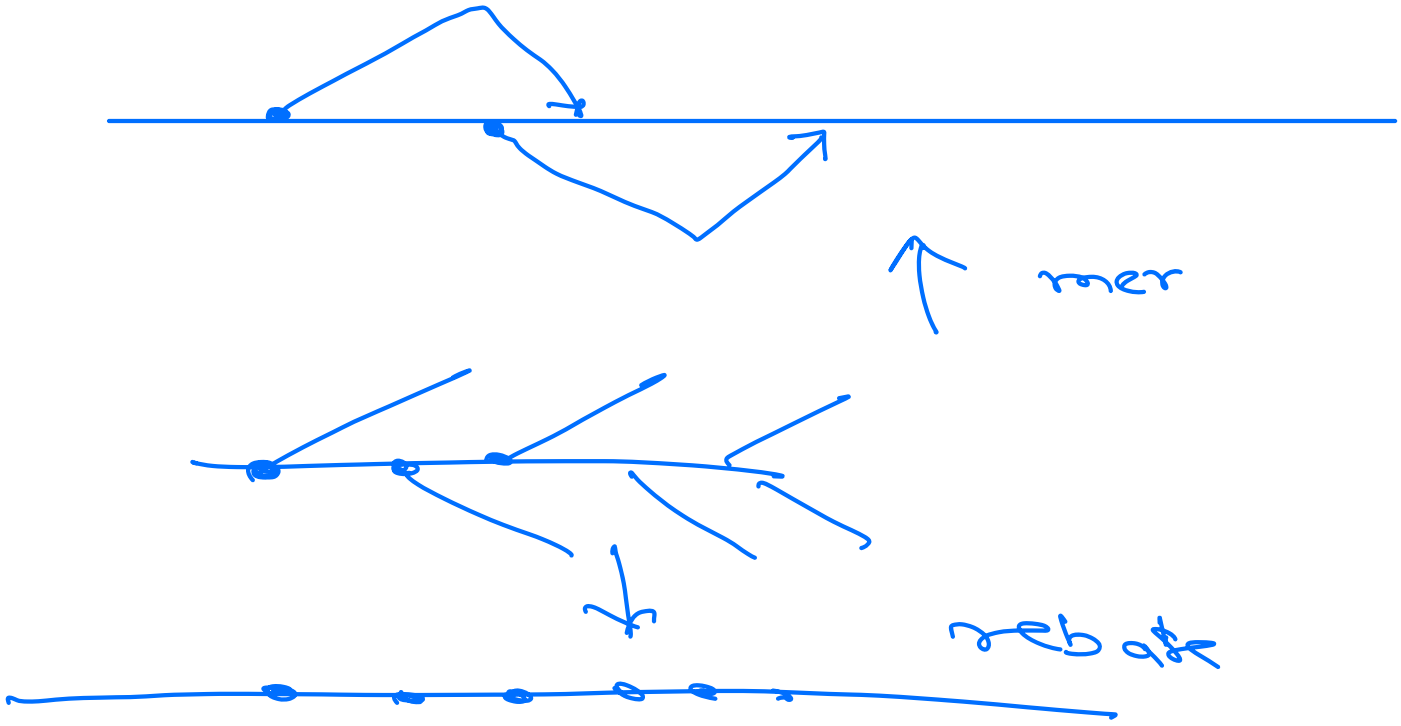
1)



Git Rebase



* We shift the base of our F branch to new commit in main branch



H.W.

* Implement all the task we did using Github Desktop via github cli

* Collaborate and simulate Merge Conflict

* Resolve and Merge PR