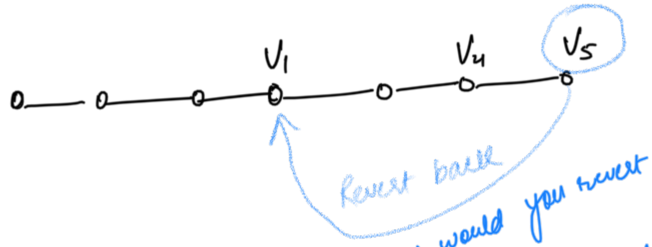
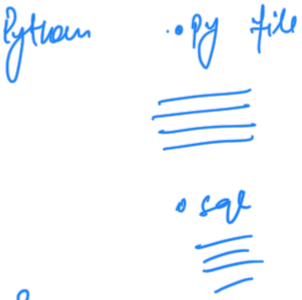
Git & Github

Project





How would you revert back

↳ create checkpoint, how?

Version the project

↳ version control system (vcs)

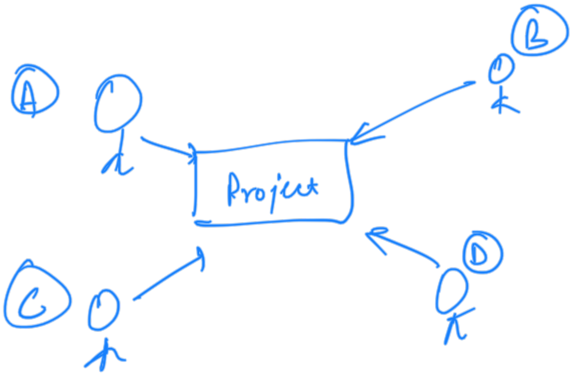
↳ GIT

GIT : Git is a VCS that lets you track all the modifications to your code

Git is not just VCS. It is distributed VCS

① Centralized (skype etc....)

② Distributed



Alternatives to GIT - mercurial / SVN

Distributed VCS means that all of the individuals working will have complete history of the project.

If Ⓐ has made a change, Ⓑ would be having the history of it & so on...

work on different feature & nobody has to

→ People can work on different features. [illegible]
communicate explicity.

Q How many of you have heard of Github & how many of you think that we are talking about this platform Github only ?

Git

→ written in C lang & that is why it is fast

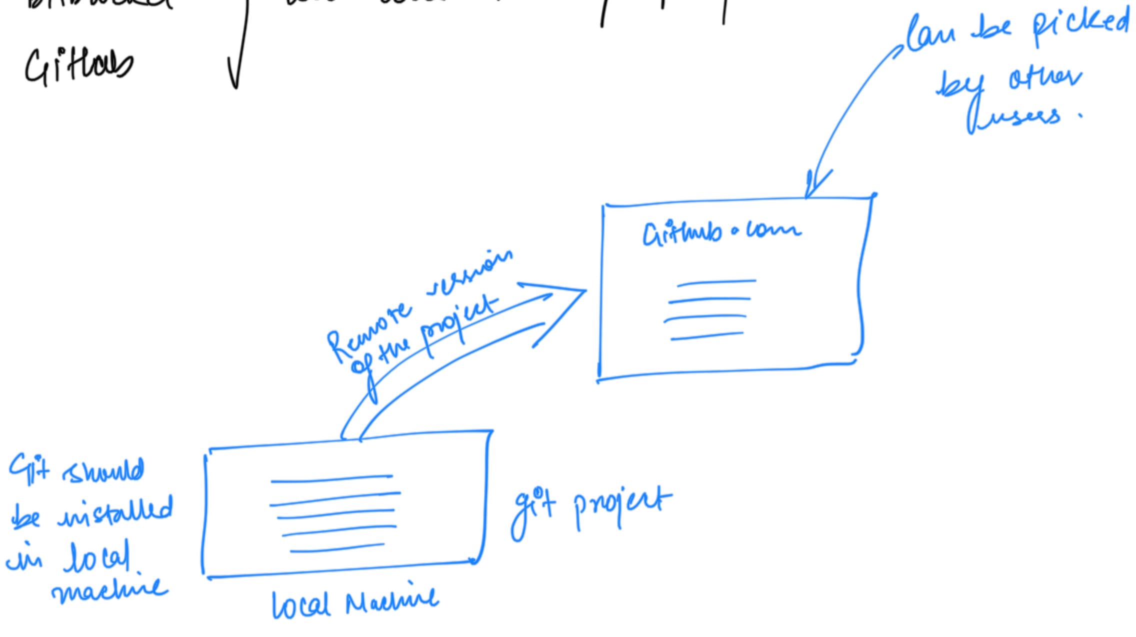
→ It is a VCS

Github

→ Github hosts the code / git repos
↳ code hosting platform
↳ you can share your project with the world
↳ If someone wants to contribute they can do it
↳ the same will be cumbersome to main in google drive

Github
Bitbucket
Gitlab
} all code hosting platform

Github


Can be picked by other users.
Github.com
Remote version of the project
Git should be installed in local machine
git project
local Machine

Platform walkthrough

↳ social platform for coders

↳ see other's work / projects

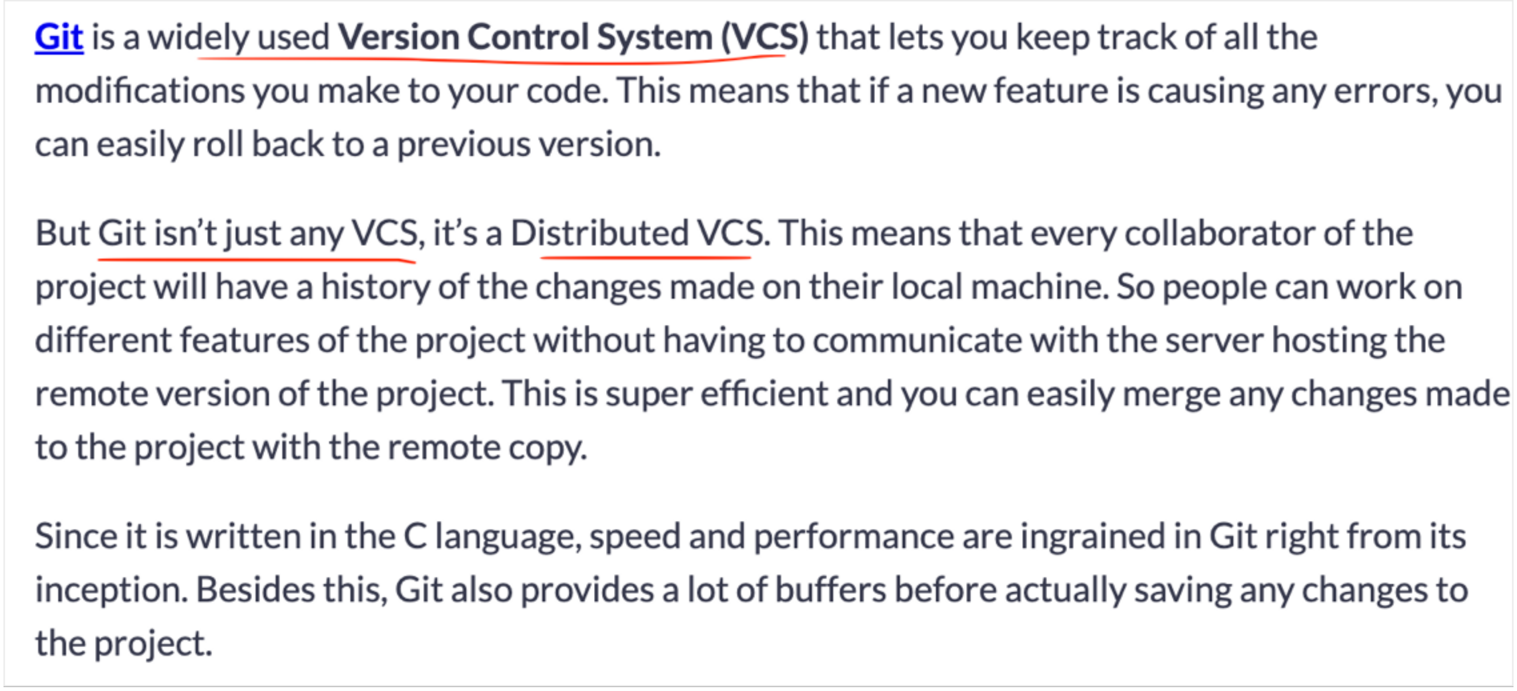
↳ contribute

GIT TERMINOLOGIES

(A) Repository : Folder that contains all the project files & the history of all the revisions made to each file.

↳ A git Repository

↳ initilize .git



Git is a widely used **Version Control System (VCS)** that lets you keep track of all the modifications you make to your code. This means that if a new feature is causing any errors, you can easily roll back to a previous version.

But Git isn't just any VCS, it's a Distributed VCS. This means that every collaborator of the project will have a history of the changes made on their local machine. So people can work on different features of the project without having to communicate with the server hosting the remote version of the project. This is super efficient and you can easily merge any changes made to the project with the remote copy.

Since it is written in the C language, speed and performance are ingrained in Git right from its inception. Besides this, Git also provides a lot of buffers before actually saving any changes to the project.

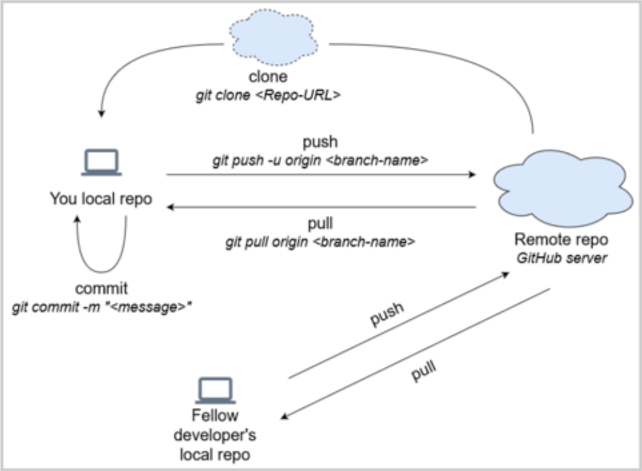
GitHub is a widely used platform for version control that uses Git at its core. It lets you host the remote version of your project from where all the collaborators can have access to it. Not just your own team members, but any member of GitHub can contribute to your code (that is of course if you choose to accept the changes made). We will discuss all of this in detail in this article.

GitHub is like a social platform where you can find a plethora of open-source projects with their codes. All the new and emerging technologies can be found on this platform. You can collaborate on amazing projects and have discussions on your contributions! This is the best open-source platform you'll find and is a data scientist's dream!

You can check out our monthly collection of the best open-source data science projects on GitHub here.

There is surely a lot that you can do on GitHub, so let's get started.

What is a Repository?


clone
git clone <Repo-URL>
push
git push -u origin <branch-name>
You local repo
pull
git pull origin <branch-name>
Remote repo
GitHub server
commit
git commit -m "<message>"
push
pull
Fellow developer's local repo

Repository or **Repo** is a folder that contains all the project files and the history of the revisions made to each file. There are two repositories of your project that you will work with throughout the lifetime of your project – Remote repo and Local repo:

- **Remote repo** contains your project that can be accessed from anywhere and by anyone. Your remote repository lives on the GitHub server and anybody can access them
- **Local repo** is a copy of the remote repo that resides on your local machine. All the changes you make will be saved in your local repo. Your fellow developers will not be able to see them yet until you push them to the remote repo

Scenarios

1) You start your own project

2) Clone the project onto your local system

3) If I have a repo, should you be allowed to push changes in my repo? → If access request → Yes

→ But if it is high stake, you can allow someone to create a copy of the repository on his account.

⊢ Do rough work & all & if things are working

You can ask to original author to incorporate the same changes

optional

FORKING

Workflow

Till Now

① You have made the changes to your folder on your local repository

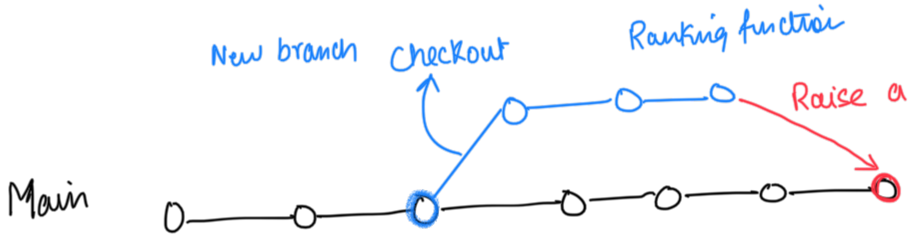
② You need to commit. Save all the changes you made & make it as a version,

③ Push | Publish it to the server

④ If I have pushed it & you also have to work on it as a team. You will clone & work on it & if i have to incorporate those changes, I will Pull from the remote repo.

Fetch command tells you whether your version is same as that of remote or not.

Collaborative Development - Branching



Raise a pull request (to merge it with the main branch)

Manager has to approve, review & then merge to the main code