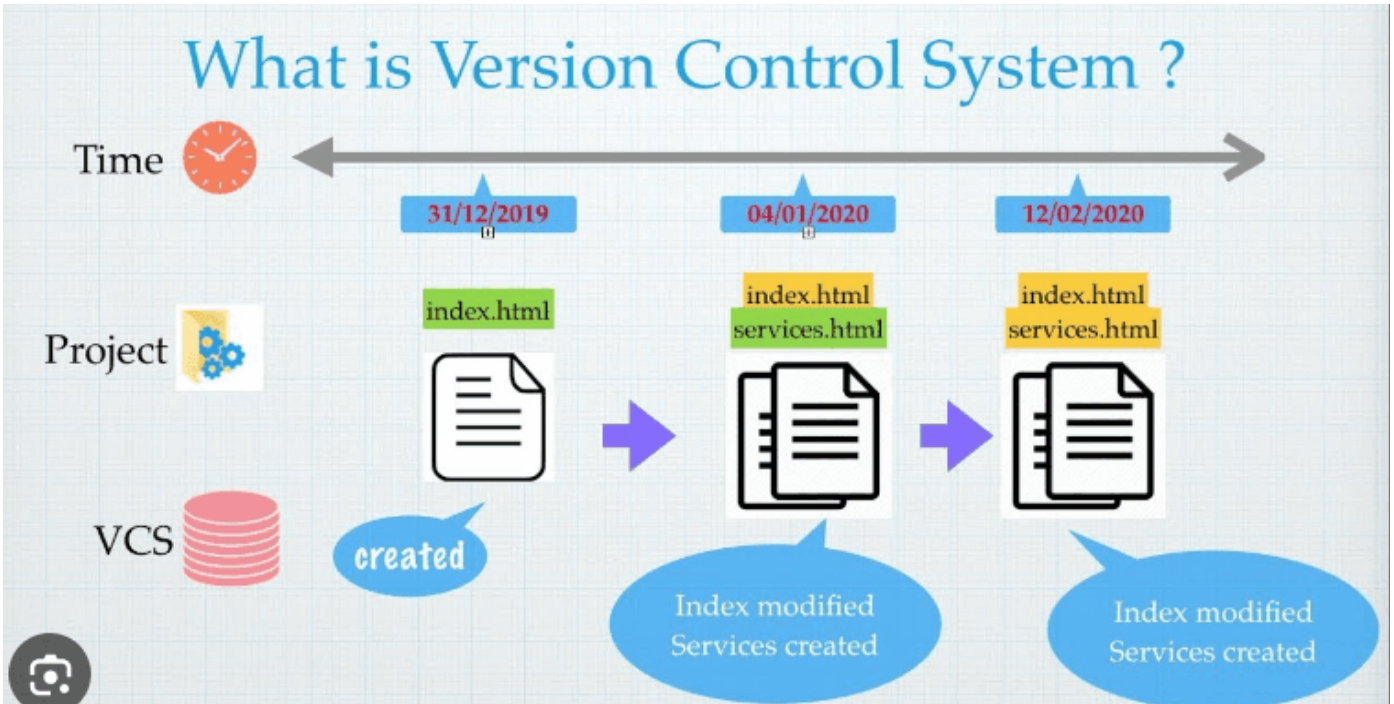
A CASUAL GUIDE TO START GIT

By Deepak

VCS


What is Version Control System ?
Time
31/12/2019
04/01/2020
12/02/2020
Project
index.html
index.html
services.html
index.html
services.html
VCS
created
Index modified
Services created
Index modified
Services created

GIT BASICS

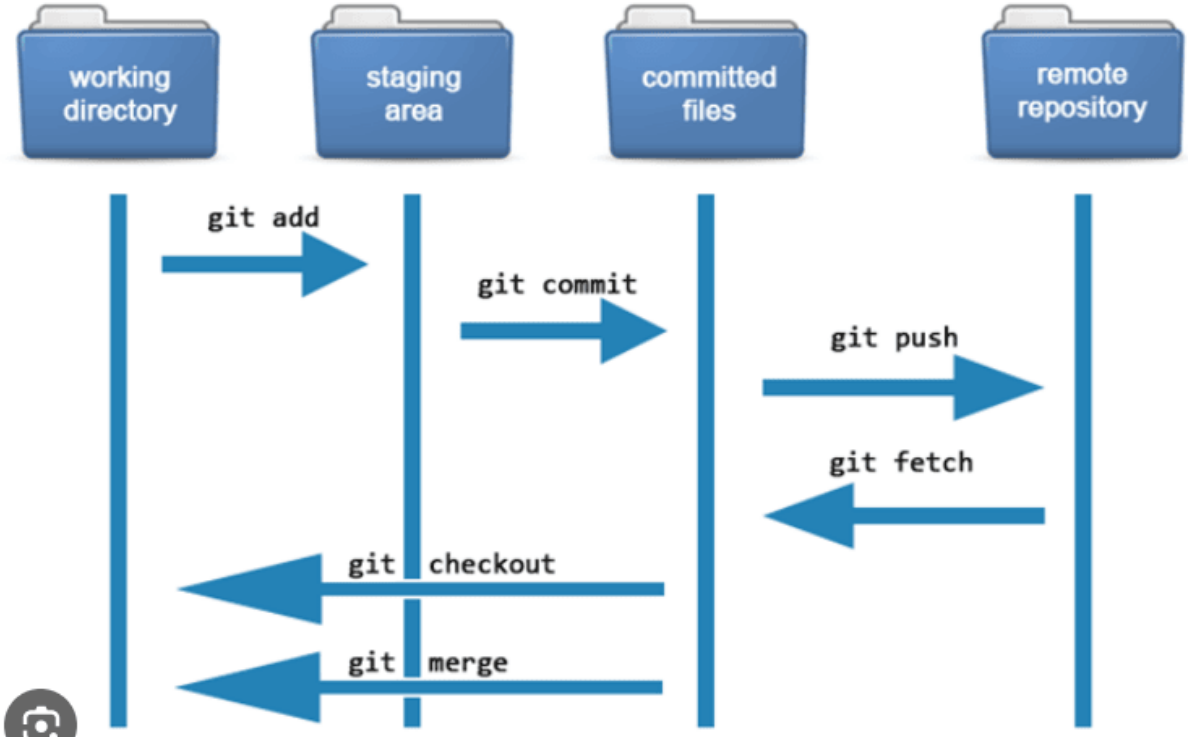
Git is a version control system that allows you to track changes in your code. The basic workflow involves adding changes to the staging area with `git add`, committing changes with `git commit`, and pushing changes to a remote repository with `git push`.

BRANCHING

Branching allows you to work on different versions of your code simultaneously. Use `git branch` to create a new branch and `git checkout` to switch between branches. When you're ready, merge your changes back into the main branch with `git merge`.

COLLABORATION

Git makes it easy to collaborate with others on a project. Use `git clone` to create a local copy of a remote repository, `git pull` to update your local copy with changes from the remote repository, and `git push` to share your changes with others.


working directory
staging area
committed files
remote repository
git add
git commit
git push
git fetch
git checkout
git merge

Git Terminology

Repository (Repo): A repository is a storage location where all your project's files, history, and metadata are stored. It's essentially a folder that contains your project and its version history.

Commit: A commit is a snapshot of changes made to files in the repository. Each commit represents a specific version of the project.

Branch: A branch is an independent line of development within a Git repository. It allows you to work on new features or bug fixes without affecting the main project until you're ready to merge the changes.

Git Terminology Cont.

Master/Main Branch: The default and primary branch in a Git repository. It is commonly used to represent the stable version of the project.

Remote: A remote is a version of the repository hosted on a different server or location. It enables collaboration and allows multiple developers to work together on the same project.

Clone: Cloning is the process of creating a local copy of a remote repository. The clone contains all the history and branches from the remote repository.

Push: Pushing is the act of sending committed changes from your local repository to a remote repository. This updates the remote repository with your changes.

Pull: Pulling is the process of fetching and merging changes from a remote repository into your local repository

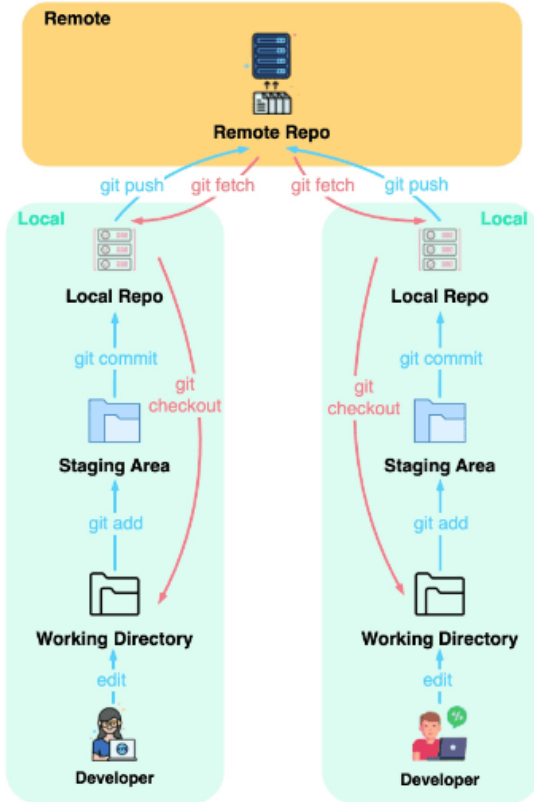
Cont.

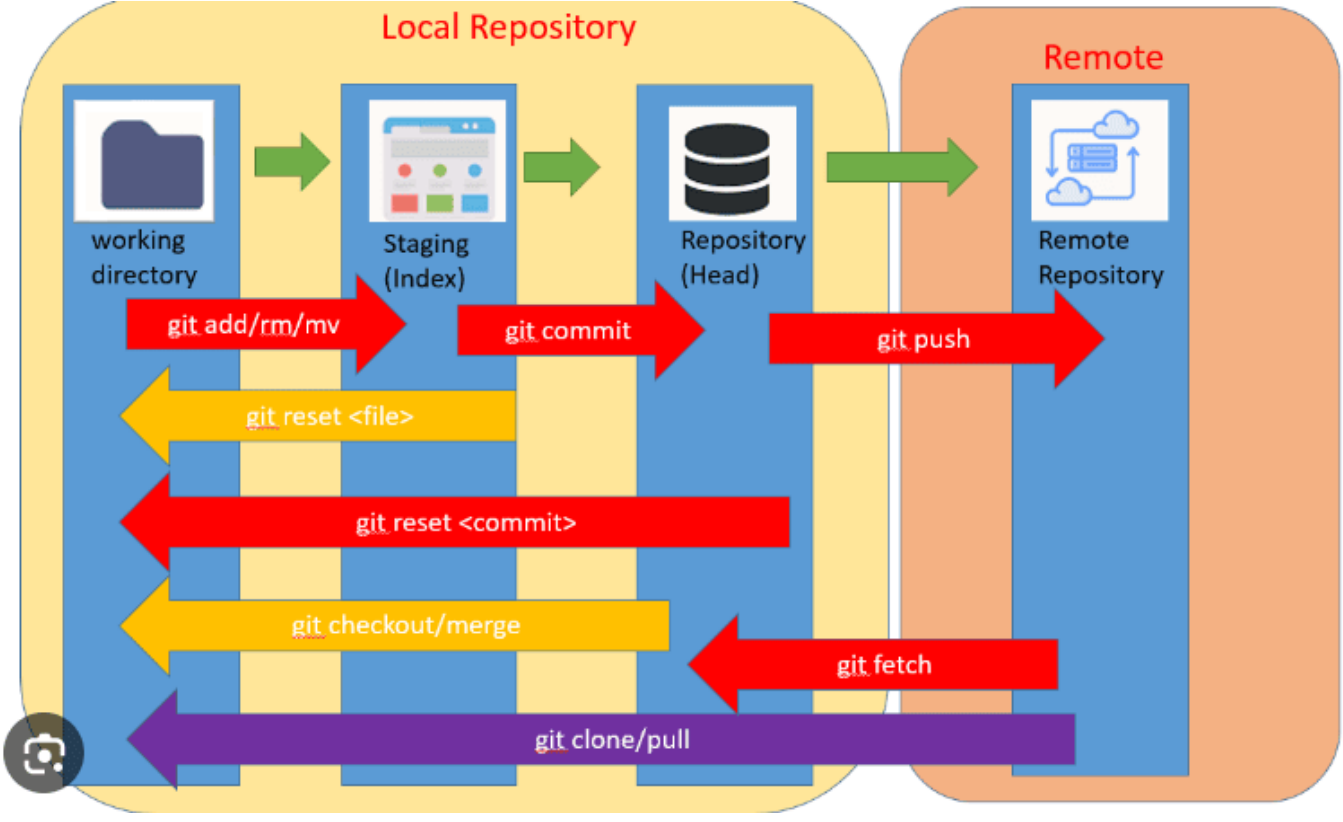
Merge: Merging combines two or more branches together to incorporate changes from one branch into another.

Pull Request (PR): A pull request is a request to merge changes from one branch (usually a feature branch) into another branch (often the master/main branch). It's a common way to propose changes to a project on code hosting platforms like GitHub.

Conflict: A conflict occurs when there are conflicting changes in the same lines of code when merging or pulling. Resolving conflicts requires manual intervention.

Checkout: The act of switching between different branches or commits in the repository. It allows you to work with different versions of the project.


Remote
Remote Repo
git push
git fetch
git fetch
git push
Local
Local Repo
git commit
git checkout
Staging Area
git add
Working Directory
edit
Developer
Local
Local Repo
git commit
git checkout
Staging Area
git add
Working Directory
edit
Developer


Local Repository
Remote
working directory
Staging (Index)
Repository (Head)
Remote Repository
git add/rm/mv
git commit
git push
git reset <file>
git reset <commit>
git checkout/merge
git fetch
git clone/pull

GIT COMMANDS

git init: Initialize a new Git repository in the current directory.

git clone <repository_url>: Clone a remote repository and create a local copy.

git add <file>: Add changes in the specified file to the staging area (prepare them to be committed).

git add . or git add --all: Add all changes in the current directory to the staging area.

git commit -m "Commit message": Commit the changes in the staging area with a descriptive commit message.

git status: View the status of the repository, showing the changes made and their current state (staged or not).

Git Commands Cont.

git log: View the commit history, showing all previous commits.

git diff: Show the differences between the working directory and the last commit.

git push: Push committed changes to a remote repository.

git pull: Pull and merge changes from a remote repository to the local repository.

git branch: List all branches in the repository.

git branch <branch_name>: Create a new branch.

git checkout <branch_name>: Switch to the specified branch.

git merge <branch_name>: Merge the specified branch into the current branch.

Git Commands Cont.

git remote -v: Show the list of remote repositories.

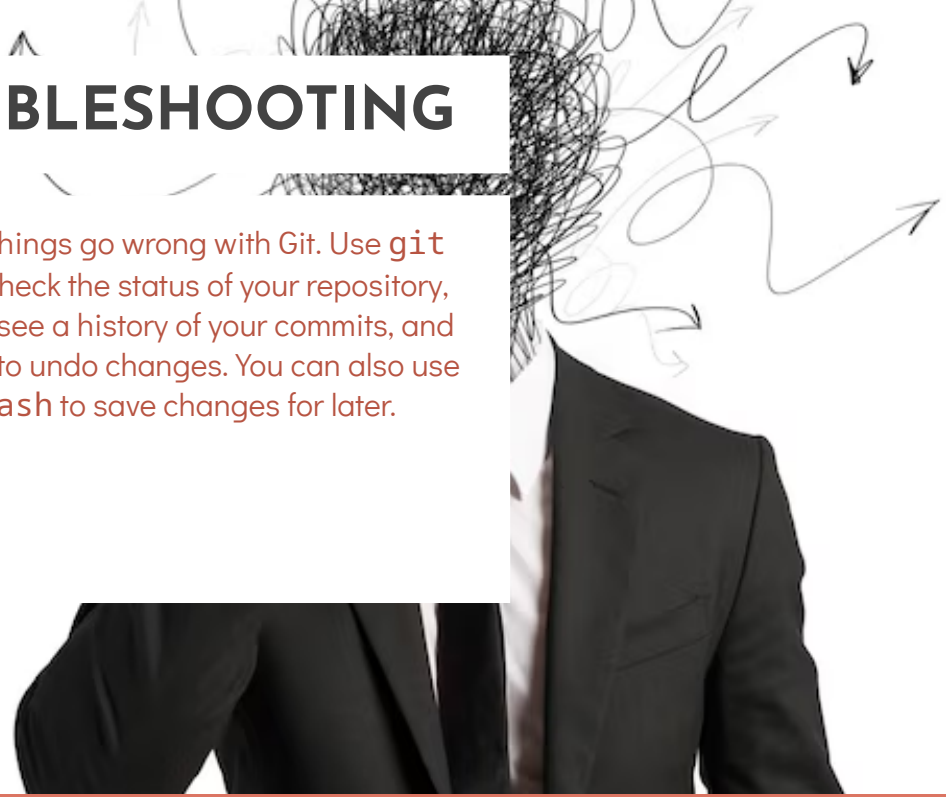
git remote add <remote_name> <repository_url>: Add a new remote repository.

git config --global user.name "Your Name": Set your name for commit messages.

git config --global user.email "you@example.com": Set your email for commit messages.

TROUBLESHOOTING

Sometimes things go wrong with Git. Use `git status` to check the status of your repository, `git log` to see a history of your commits, and `git reset` to undo changes. You can also use `git stash` to save changes for later.



CONCLUSION

Congratulations! You're now on your way to becoming a Git master. Remember to keep practicing and don't be afraid to ask for help. Happy coding!

Thanks!