An Introduction to Git

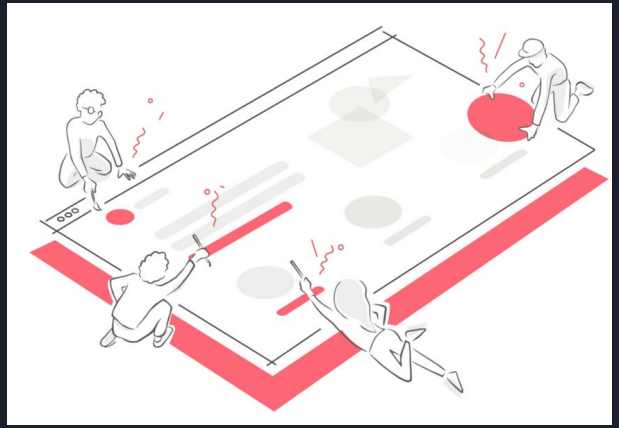
IRIS NITK Bootcamp - Session 0

Introduction

- Let’s say you’re working on a project called “NITKNet” - a social networking platform for college, with a bunch of your friends.
- You and your friends decide all the features that you want to implement in the app - Friends, Messaging, a feature to share course material and assignments, and so on.
- But you immediately face an issue - how will you collaborate on the project in a way that you do not overwrite your friends’ code, how will you maintain different versions of your app?

What is Git?

- Git is a **version control system (VCS)** that makes it easier for organizations and individuals to track and collaborate on changes to files.
- It is used to maintain a complete record of **what** changes were made, **why** these changes were made and **who** made them
- **Note** - Git and GitHub are **not the same** thing, GitHub is a website that is used to host projects that use Git.

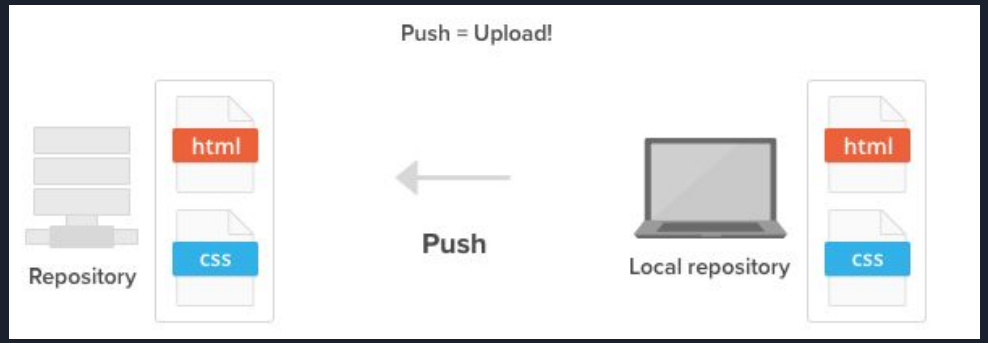


Installation

- https://git-scm.com/downloads - Download Link for Git
- https://git-scm.com/book/en/v2/Getting-Started-Installing-Git - Installation Guide for Git
- https://desktop.github.com/ - GUI Version (Windows and Mac)

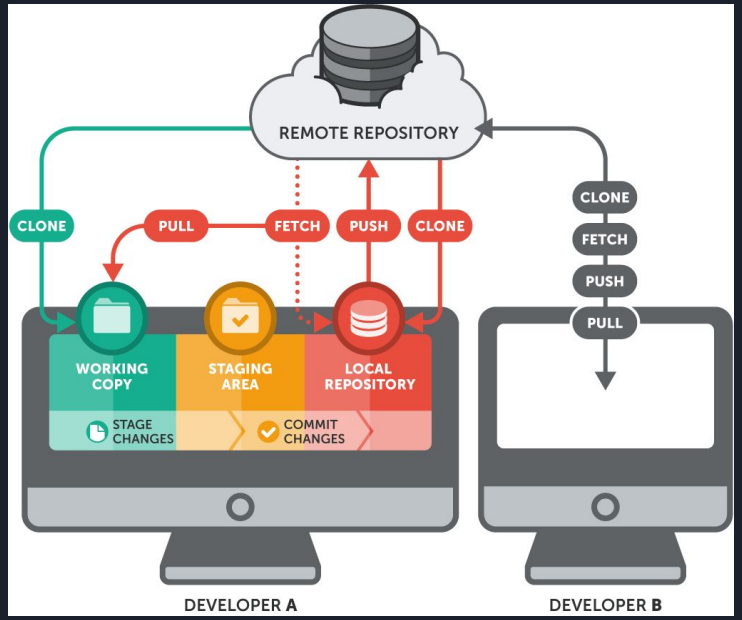
Basic Commands - 1

- A **repository** is the same as your project folder - which also contains a record of all the changes that have been made, who made them and why they were necessary.
- Two types of repositories - **local repo** (Present on your local computer system) and **remote repo** (Common repository that resides on a Web server)
- ***git init -*** This command is used to initialize or create a .git folder, which converts the working folder into a Git repository. Any changes made to this folder from now onwards can be tracked using Git.



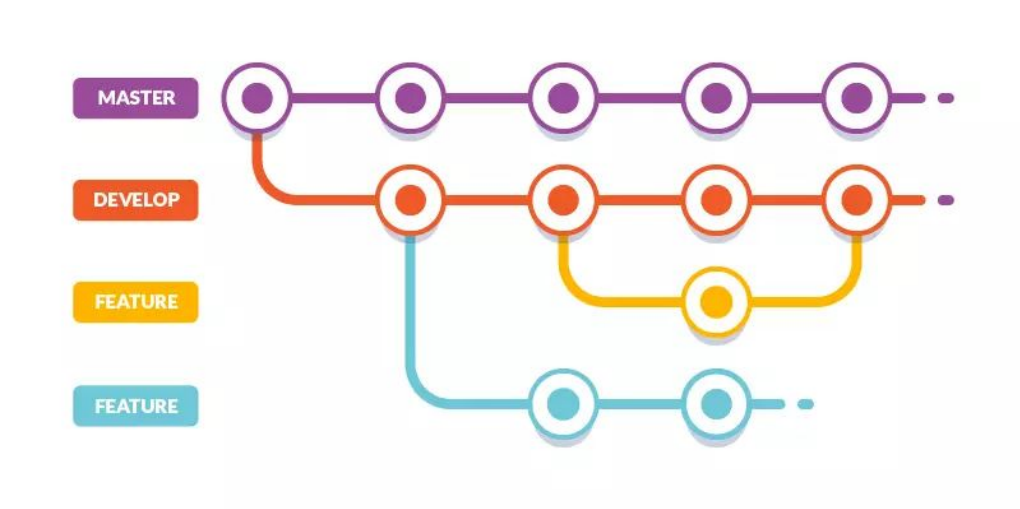
Basic Commands - 2

- ***git add <file_names> -*** After you have made changes to your local copy of the project, you need to **stage** these changes.
- It can be thought of as telling Git what changes you want to upload to the remote repository.
- ***git status -*** Displays some basic information regarding which files have been modified, staged, etc
- ***git commit -*** You can **commit** your **staged** changes - a commit message can also be given to explain why the commit has been made



Basic Commands - 3

- ***git add remote origin <remote_repo_URL>*** - This command tells Git to link your local repo and the remote repo present at the specified URL.
- ***git push -*** Pushes or uploads the commit to the remote repository.
- **Branches -** Git Branches allow the collaborators to work on separate copies of the repository.



Basic Commands - 4

- ***git clone -*** Downloads a remote Git repo to your computer.
- ***git pull*** - Pulls (Downloads) any new changes that have been made to the remote repository to your computer, and updates your local repo.
- ***git branch <branch_name> -*** Creates a new branch with the given branch name
- ***git checkout <existing_branch_name>*** - Changes the working branch
- ***git merge <branch_name> -*** Takes all the commits that exist on <branch_name> and merge them into your current branch