presented by DigitalOcean and DEV

GIT_INIT

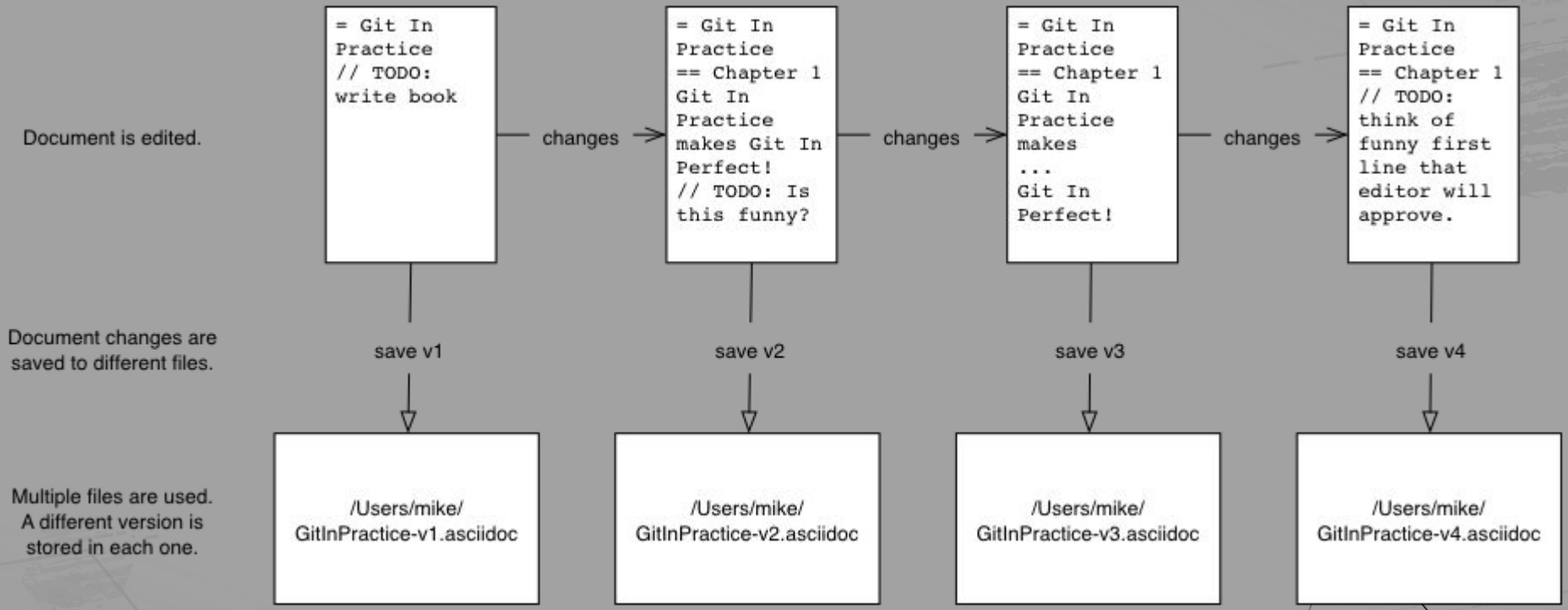
Basics of version control and GIT

IEEE NITK

Topics:

- Version Control
- Git
- Github
- Demo

Version Control



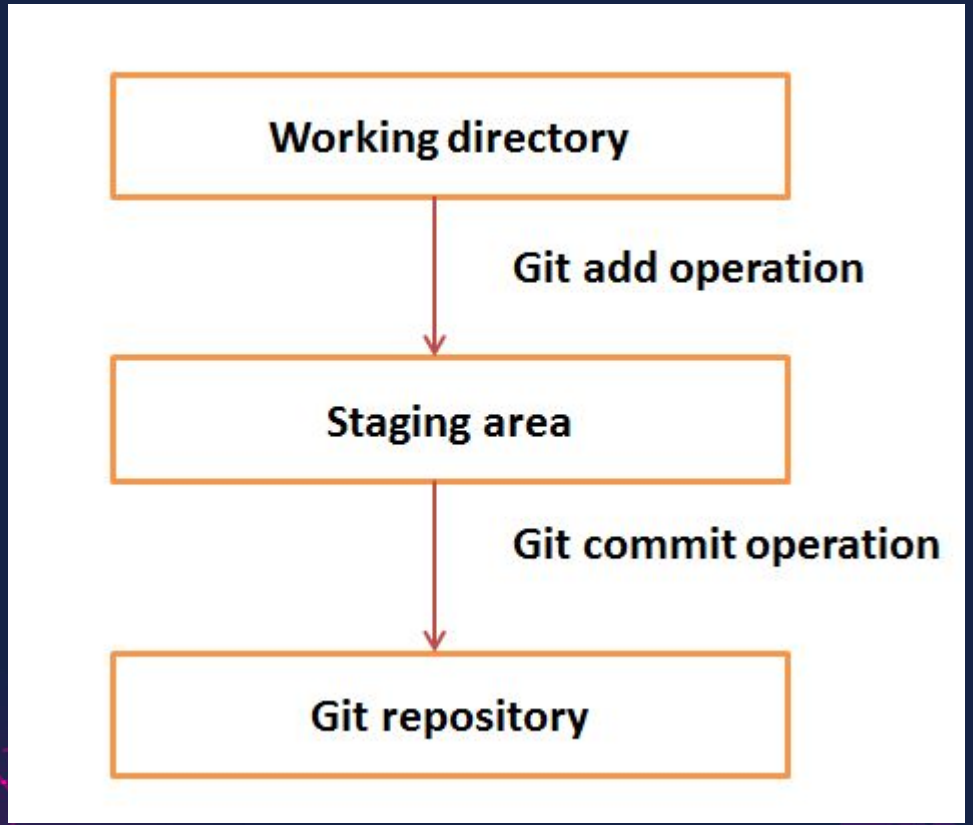
Version Control System

- Version control system records changes to a file or set of files over time so that you can recall specific versions later.
- It is also used to keep a track of who made a particular change.
- Eg. Git, SVN, Mercurial, etc.

Learning Git

What is git?

- A distributed Version Control System
- Open Source
- Takes “Snapshots” of your files/code.
- Keeps track of code history.
- Coordinates work between multiple developers.



Demo time! Yes!

Demo - 1

Step 0: Install git
https://www.atlassian.com/git/tutorials/install-git

Step 1:

git init

- Initialize a new, empty repository.
  OR
- Convert an existing directory into a new Git repository.

Step 2:

git add

- git add <filename>
- git add .
- Tells Git about what files need to be added to the staging area

Step 3:

git commit

- **git commit - m “Commit-Message-Goes-Here”**
- **All the changes from the staging area are added into the local repository**

Extra step:

git status

- Tells us what files are in the staging area, check status of the working tree.
- Tells us about the tracked and untracked changes.

Extra step:

git log

- git log
- Shows your commit history.
- Shows a list of your commits with the commit message, timestamp and a hash value of that commit

Github

What is Github

- Seperate from Git. Git does not need Github to exist.
- Github is a web based platform that acts as a server where people can host their repository.
- You can host your git repo on your own web server without using github.
- Github has become the most popular destination for most of the open source projects and hence it has become synonymous to git.
- Alternatives to github: Gitlab, Bitbucket etc

How to use Github:

- As simple as creating an account on the website.
- You can create repos in your account
- You can also “clone” public repo to your local system
- You push your changes to the “remote” github repo
- We shall discuss all these concepts in a demo next

Demo - 2

First step: **fork**

- Fork the repo you want to work with
- Go to github.com and fork the following repo:
    - github.com/webclub-nitk/hacktoberfest-2k19

- This will create a copy of the repo in your account.

Second step: **clone**

- Clone the repo you just forked
- Go to github.com and clone the following repo:
    - github.com/<your-username>/hacktoberfest-2k19

- Go to terminal and type this:
- git clone https://github.com/<your-username>/hacktoberfest-2k19

Third step: **Pick up an issue**

- Pick a good first issue.
- For example, we can pick issue #1 -> Introduce yourself
- Follow the code of conduct and contributing guidelines.
- Claim the issue and start working

Fourth Step: Add the files and commit

- Go to the introductions folder
- Add a text file with your roll-no. Eg. 17CO221.txt
- Add your details like name, username, branch, etc. (Check the README.md file for an example)

- Add and Commit
- git add .
- git status (Should display the file you create above)
- git commit -m “Details of 17CO221 added”

Fifth Step: Push the changes to remote github

- Command: git push <local> <branch-name>
- By default we will have <local> called origin
- <remote> as master branch (could be other branch name as well)

- git push origin master

Create a Pull request (PR)

Go the your forked repo:

- https://github.com/<your username>/hacktoberfest-2k19

There will be an option to initiate a PR. Go ahead and submit a PR.

References

Version Control:
https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control

Git Cheat Sheet:
https://www.atlassian.com/git/tutorials/atlassian-git-cheatsheet

THANK YOU

Special Thanks to
CSD NITK
for providing us with the
required technical support.


Hacktober
FEST
2019
presented by
DigitalOcean
and
DEV