**GIT**

**Git Commands:**

**git init**

Initialized empty Git repository in respective folder

**git remote add origin “**[**https://github.com/suraj990kumar/project1.git**](https://github.com/suraj990kumar/project1.git)**”**

**git pull origin master**

**git add –A**

**git commit -a -m "update git"**

**git push –u origin master**

**git remote –v**

**git clone "https://github.com/suraj990kumar/spring-boot.git"**

**git fetch**

**git checkout master**

Change the branch

**Set the proxy:**

**git config --global http.proxy http://username:password@proxyurl:proxyport**

**git config --global https.proxy http://username:password@proxyurl:proxyport**

**Unset the proxy:**

**git config --global --unset https.proxy**

**git config --global --unset http.proxy**

- creating repository

git init - create new repository

git clone - get existing repository

- create central repository

after sign in to Git hub -> click Start project -> create a new repository -> enter repository name -> Initialize this repository with a README -> Add gitignore -> Add a license -> click on Create repository

1. edit file -> commit changes

- create a local repository

go to folder -> right click select git bash -> git init -- local repository created and initialized

- find link of central repository

Clone or download

click and copy the url

-- link local repository to remote repository

git remote add origin "link of central repository"

-- fetch all file from central repository

git pull origin master

- making changes in files

git status -> tell you which files are added to index and are ready to commit

git add -> lets you add files index

git commit -> working copy to local repository

1. create a file in working copy

git status

git add edu1.txt

git status

git commit -m "commit message"

git add -A -> add all file to the index at once

git commit -a -m "commit message" -> commit all file at once

git log -> detail of all git local repository

**https://www.youtube.com/watch?v=WfhRyz3Wf4o**

**Git : Distributed version control system**

**Git is created by Linus Torvald**

**Git is a Distributed Version Control System.**

**Git is famous for project collaboration**

**More than 1 person can work on project at the same time**

**centralized version control system - SVN**

**Distributed version control system -**

**Local repository (at every local PC) and remote reposistory (at centralized location)**

**Trunk Based Development:**

**Master Branch:**

**create child branch -> after completion -> merge to the Master Branch**

**Git Hub: Git lab : bitbucket**

**on cmd -> git**

**1. create an account on github**

**2. login to git hub**

**3. create a repository -> Repositories -> New**

**4. Owner, name, create reposistory**

**About tab**

**<>code -> code**

**5. create a new file**

**6. edit file mutilple times**

**7. go to X Commits to see what has been commited when**

**click on View to go to previous version.**

=== Branch creation and Merging =====

1. if edit and commit then by default go to master branch

Commit changes

Branch creation

2. before click on Propose file change, select the option to create new branch

Merge branch

3. click on Compare & pull Request

compare two branches and merge it.

4. click on Create Pull request

5. if conflict then -> click on Resolve conflict

6. Manually edit the file

7. click on Commit merge

8. Merge pull Request

9. Confirm merge

10 Code -> go to -> master branch verify the code changes.

Git Fork: let others to contribute in your repository

Git is created by Linus Torvald

Git is a Distributed Version Control System.

Git is famous for project collaboration

More than 1 person can work on project at the same time

You can also contribute to public projects

If you fork project, that project comes into your repository

Then you can make change and request to project owner to merge changes.

1. <> Code

2. other person open his repository

3. search for username/project | get the link from ceator (? how to share link to someone else)

4. create a file

5. Create pull request

6. click on Commit

7. in old there is not

8. Pull request

9. verify it

10. Confirm merge

Fork 1 -> one person has fork your request.

Parallel development -> Branching

Local branch

remote branch

git branch firstbranch -> this will contain all files from master branch to firstbranch

git checkout firstbranch -> switch to child branch

git add -A

git commit -a -m "msg"

git checkout master

Merging branch: combine the work of different branch together. should always merge code from child branch to master after development

check out to destination branch

if firstbranch code has to merge to master

then goto master branch

git megre firstbranch -> master branch got code of other branch

-modify the existing file and commit it

git commit -a -m "msg"

-swich to firstbranch

git rebase firstbranch - merge done linearly without any further commit

- push code from local repository to remote repository

--under master

-- first connect through ssh -> generate ssh key

1. ssh-keygen

2. cat (ssh key) -> copy it

3. go to Git hub account -> settings -> SSH and GPG keys -> New SSH key

Title: name

Key: paste the ssh key from 2

4. go to branch in local

ssh -T git@github.com

5. refresh the git hub account you can see green

6. push changes to new branch

git checkout firstbranch

git push origin firstbranch -> it will create a new branch and push all into the new branch in Git hub

7. push changes to master

git checkout master

git push origin master