Git :- 2-3-22

What GIT?

1. For tracking code changes
2. Who made the changes
3. Coding collaboration

Java :- we devlope a part of project say Alpha version is release on 10-Feb-22

Next version we are going to release in March 10

We are having team of 4 peoples> :- some one working on some part and some one working on other part. To combine the code what method one should use if not using the git

1. We should copy the code from one machine and another machines and let combine them together on another machine.

When we talk about the git what we do:- we shared a folder or say directory. By which we can each and everyone is going to put there code in that particular directory. And we get the all code merged in one folder and whenever some one change or new code is added we will get the updated file from that user.

How to work with Git:-

* Initialize git on a folder by making it a repository.
* Git now creates a hidden folder to keep track of changes in that folder
* When a file is changed, added or deleted it is considered modified.
* We have stage facility.
* With staged files are committed, which prompts git to store a file permanent with him.
* Git we get full history of every commit
* Even we can revert back to any previous commit
* Git does not store separate copy of every files or folder that we have committed. But it keep track of changes made in each commit.

GIT commands

Git version :- git –-version

Configure the git for user.

Git config - - global user.name “vimalkumar”

Git config - - global user.email [vimal.jawla@yash.com](mailto:vimal.jawla@yash.com)

Git config - -list

To create a folder for the repository

Mkdir foldername

To go into the folder :- cd foldername

Git init

We have created the file and saved it in our git directory

Now to get list of all files we use ls command

To get the status of git repository

We run git status

Tracked and untracked:-

To commit the file and saved to git repository we use

Git commit -m “message” :- this message is for understanding what we have committed.

Git log

Git log --oneline

Revert to find the commit in log

First thing we need to find the point we want to return to.

One option will revert to latest commit

Second option will revert us to given hash code of the commit

To revert the latest changes we use HEAD

Git revert HEAD --no—edit

Git revert head~NO :- inplace of No we can put the no of reverts for the commits like if we put 1 it will go back to the latest commit and will revert it

To edit the file using vi editor and save them

Press esc key and then press :wq and press enter the file will be saved and you will exit from the vi editor

We are having status flags that can be used with status for printing information on filter basis

?? which only list untracked files

A Files to be added to stage

M print modified files

D deleted files

We have stage the files for adding them in git

We revert the file to previous version

We list all the files

Branching in GIT repository:

Branch is the separate version of the main repository

Suppose we are having files in large projects and we need to update different folders at time

IF we don’t have branch in git what will we do:-

We are going to creates the copy of all files so that our live version will not impact

We have work on the design and find that code depend on code in other files, that need to be changed.

Must copy all of the dependant files.

If I got some error in our current project which is live I need to fix it asap otherwise it will down the application

With branch

With a new branch we can directly edit the code in the branch without impacting to the main branch

We will create the errors branch in order to merge all the errors coming in the code or project

Merge the new code of our branch to the main code and the all live version will work fine

For creating branch run the command

Git branch branchname

To switch the branch I need to command

Git checkout branchname

Mergeing the files from two different branches

Git merge branchname

Branch1

Is our branch where we updated some of the files now we want to merge it to our live session that is we need to merge the code with master branch

First of all goto master branch by checkout method

And then call the merge branchname

GitHub

What we get by GitHub

We can push our local repository to online

We can get data updated and we can pull the our online repository to local repository.

Got to github create an account

Git remote add origin <https://github.com/vimal-yash/march2-yash.git>

Command to push the local master repository to online github repository

Git push –set-upstream origin master(local branch name that need to push )

Sending data to github is push process

And getting data from github is pull process

Pulling data or changes made on server by team(s)

Fetch :- fetch gets the all changes/history of the branch/repo

Pull is combination of merge and fetch