1. Install git software on machine
2. Create GitHub account. Use personal email id
3. What is Git ?

Git is an installable software which will take care of versioning thing. IT provides version control mechanism.

1. What is GitHub ?

GitHub provides a platform where we can put our code. Or it gives a medium where we can see the different versions by different people of the same repository. UI available on web where we can keep our repository of our project and it will be available for whole team.

PR – Pull Request: Going to generate PR on GitHub

1. What is SSH key ?

Git is installed on a local machine while GitHub is available on web, so we need to establish a connection between these two different platforms.

Using SSH key we can establish this connection.

This key is unique encrypted key into local machine using email address which we have used while creating the GitHub account.

Then we put this SSH key in GitHub account.

Open your GitBash and enter below command (here specific folder path is not required)

The SSH key needs to be generated only once if we are using same computer and same account.

1. Generate SSH Key

ssh-keygen -t rsa -b 4096 -C “yourEmailID”

(it uses rsa algorithm to encrypt a key it will going to generate with the mentioned emailed on this machine and it has 4096 bytes)

Need to use the email id which you have used to create gitHub account.

This command will generate a key and saves it in default path (/C/Users/<username>/.ssh/id\_rsa)

Do not enter passphrase (just click enter to skip)

1. Copy key from GitBash

cat ~/.ssh/id\_rsa.pub

Copy the key.

1. Add the copied SSH key on your GitHub account

In GitHub > user icon at right top corner > Settings > SSH & GPG Keys > SSH Key – New SSH Key > Enter copied value > Add key name as “SSH Key” > Click on ‘Add SSH Key”

Set your name and email address on your machine through gitBash. This is required so not to get any warning messages during your first commit

1. git config –global user.email “[you@example.com](mailto:you@example.com)”
2. git config –global user.name “YourName”
3. Create a repository in GitHub

Give meaningful name and keep all parameters to their default values.

Create the repository in different ways

1. We can clone the project
2. Clone the repository
3. Paste the project inside the repository using SSH or HTTPS keys
4. Create repo using below commands

git init

git add <filename>

git commit -m “first commit”

git remote add origin git@ github.com/aiStudent910/GitTrainingByKrishnaXor

git push -u origin master

1. Create a new branch

We cannot work every time on master branch.

Only first time when we are uploading the project we work on master branch

After that every time whatever the code we want to push on github repo, that we are going to linked with our local branch.

Master always holds the final error free copy of project.

Every time for a new assignment, you are going to create a sub-branch/ child branch from the master.

Steps to be followed while working in project,

Checkout master on your machine

Create a new branch

Will do changes / tasks

The raise a PR to merge changes in master

Before creating the branch take a pull from master in our machine

git pull origin master

Then create a new branch

git checkout -b “Chetan\_Branch\_1”

Do the changes you need to do in the local folder

Add two new files and write some texts in them

Update an existing file with some changes in its contents

Now execute below command

git status

Add the changes

git add <firstfile> <secondfile> <thridfile>

git status

git commit -m “<commit message>”

Now execute below command

git push

This will give you command to push the branch, as below

git push –set-upstream origin <branchName>

After successful execution of above command, a new branch gets pushed to the GitHub.

Refresh the GitHub page and you will see ‘compare and pull request’

Click ‘compare and pull request’

Create Pull Request

Merge Pull Request

Confirm Merge

After this, in the master branch you will see all new changes available under the master branch

1. Stash

git stash

This command saves the work in your local machine without committing the work.

i.e. If user needs to switch the branch or account before committing the current changes. Then user can save the current state of work by using the stash command.

Difference between the stash and commit is

Commit tells git that the changes done are final and you are going to push the changes sooner to the gitHub. We can commit only those work which is well tested.

While stash saves temporary. The changes are not even added. Working tree gets cleaned after stash.

So on executing git status command after the stash, it shows “nothing to commit, working tree clean”

git stash list

git stash show

How to handle git conflicts