C:\Users\nchin\OneDrive\Desktop\git>git push --set-upstream origin master

git@github.com: Permission denied (public key).

fatal: Could not read from remote repository.

First you create a GitHub account you have to authenticate your machine to your GitHub by using ssh public key

First perform ssh-keygen. It will generate two keys 1. Public. 2. Private

Then copy the public key by using a notepad. Then place the code into your GitHub account.

1. Go to settings!
2. Go to ssh and gpg keys.
3. Then place your key in to ssh key
4. That's it. Now you can authenticate your GitHub account to your server.

A screenshot of a computer

Description automatically generated

Go to the settings!

A white background with black text

Description automatically generated

When we will create a new remote repository we have to add the origin to the local repo

A screenshot of a computer program

Description automatically generated

A whiteboard with writing on it

Description automatically generated

A close-up of a whiteboard

Description automatically generated

1.Git Init

 initialize repository in our locally and perform the clone push branch commands

2. Git clone

Downloading the content from repo to our locally

After finishing the changes add to the staging area by using add

3. Git add <filename>

It will go waiting in the staging area until we do the commit to the local repository.

4. Git commit -m “commit message.”

It will go merging to our local repo.

5.Git push

It will send our data in our local repo to remote repo.

 all the above we will use in single branch.

Multiple branches

6. Git branch <branch name>

Create a new branch {whenever you are creating a new branch it will copy all data in master/main branch.

7. Git checkout <branch name>

Switch between the branches

8. Git branch

Checking the number of branches

9. Git status

Checking status

Git add, git commit, git push after we have to do the pull request to merge our code from our branch to master branch.

Conflicts

10. Git pull

Perform in master branch it compares master branch in local and remote repos and pull all the date from remote repo

11. Git merge origin/master

Then go to your working branch and perform merge command it will merge updated data from master

We have a master branch.

We created two branches.

1. employ1

            2. Employ2

So, it will copy all the data from the master branch.

We performed some changes in the employ1 branch and we performed add, commit, push and raised the pull request. It is also accepted.

Ex: we changed hi- hiiiii

Then we made some changes in employ2 branch, and we also performed add, commit, push and raised a pull request, but here we will get some conflicts like below.

So, click on resolve conflicts

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Then you must perform these things{commands}

1.Go to the master branch and pull the data from the remote master branch. So the date       is up to date in your local master branch.

      Git pull.

2.Then go to employ2 branch and merge the master branch by using

   Git checkout employ2

   Git merge origin/master

3. Allow both changes in vs code editor

4. Git add

5. Git commit

6. Git push

It will allow the merger on your previous pull request after you did fix your conflicts.