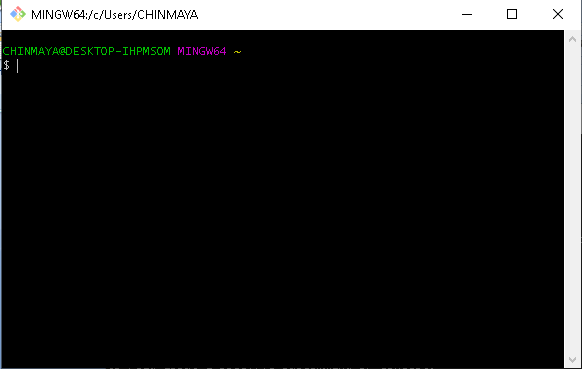




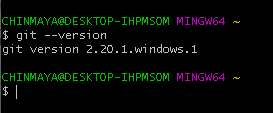


Install git

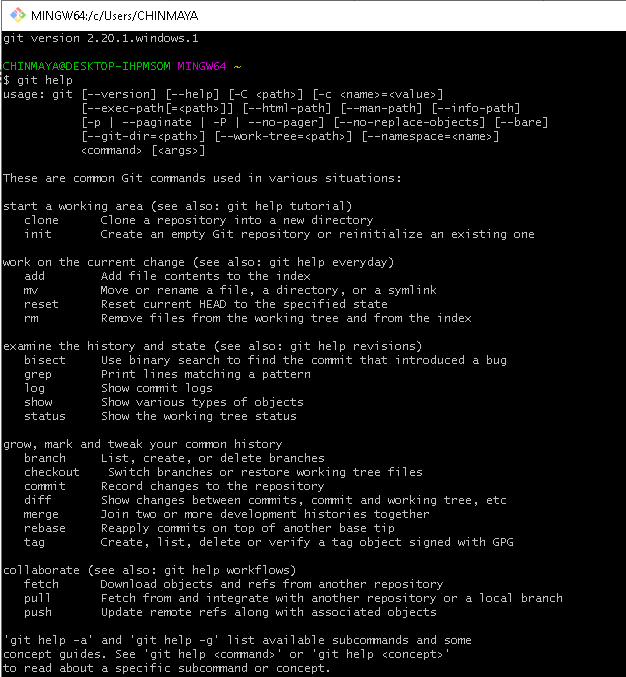
Goto serch and search git =>double clickon it

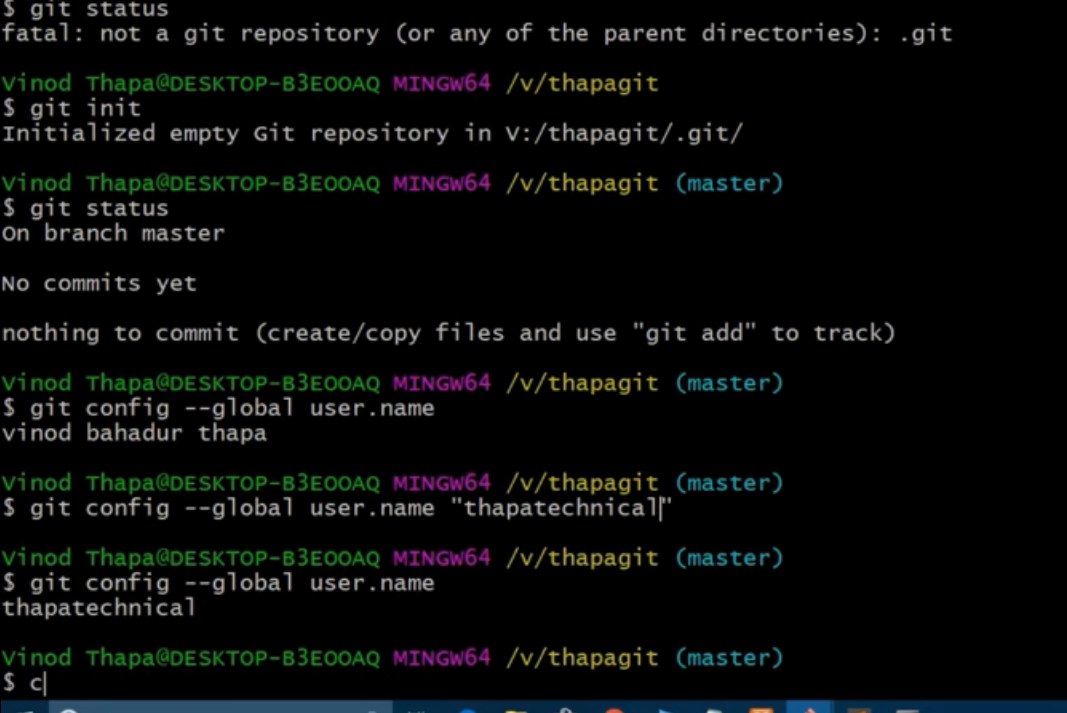


Check version

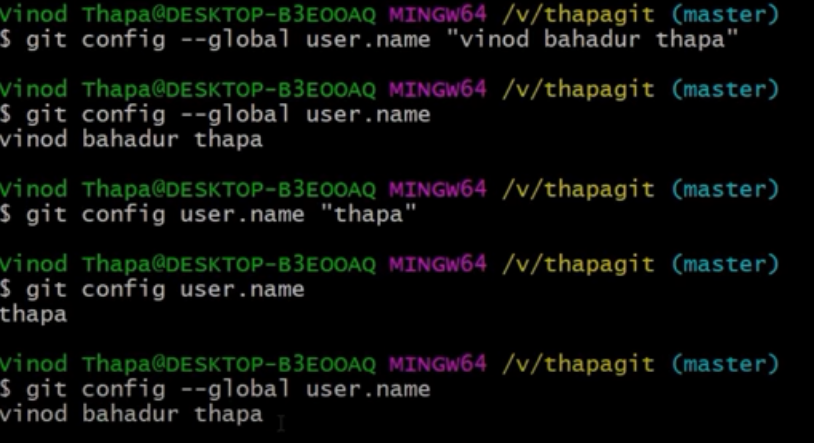


Write Git help

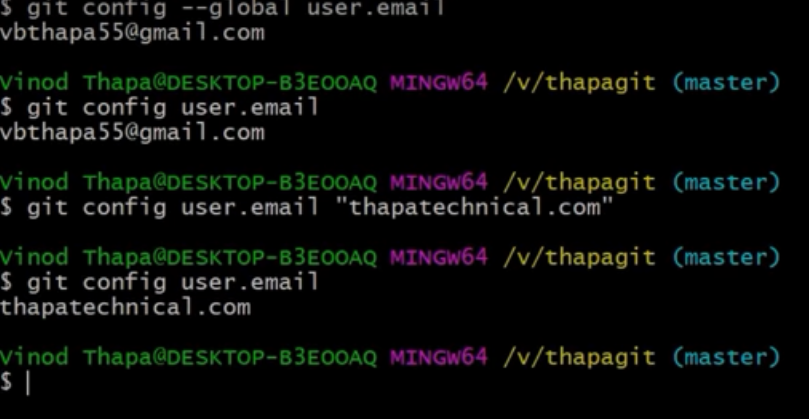




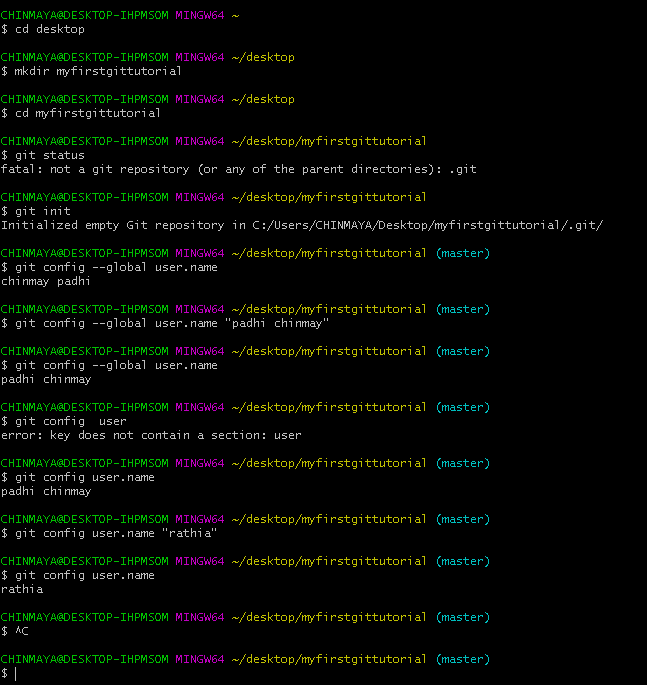
Create and check global and local username



Set email







CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~

$ cd desktop

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop

$ mkdir myfirstgittutorial

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop

$ cd myfirstgittutorial

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop/myfirstgittutorial

$ git status

fatal: not a git repository (or any of the parent directories): .git

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop/myfirstgittutorial

$ git init

Initialized empty Git repository in C:/Users/CHINMAYA/Desktop/myfirstgittutorial/.git/

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop/myfirstgittutorial (master)

$ git config --global user.name

chinmay padhi

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop/myfirstgittutorial (master)

$ git config --global user.name "padhi chinmay"

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop/myfirstgittutorial (master)

$ git config --global user.name

padhi chinmay

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop/myfirstgittutorial (master)

$ git config user

error: key does not contain a section: user

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop/myfirstgittutorial (master)

$ git config user.name

padhi chinmay

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop/myfirstgittutorial (master)

$ git config user.name "rathia"

CHINMAYA@DESKTOP-IHPMSOM MINGW64 ~/desktop/myfirstgittutorial (master)

$ git config user.name

rathia

now I am going to create my first git local directory

cd desktop

mkdir chingit

cd chingit

git config --global user.name “chinmay padhi”

git config --global user.name

git status

git init

git status

touch index.html

notpad index.html (write something here and ssave)

git status

git add . //Or git add filename

git status

git commit –m ‘commit 1’

to know id of each commit

git log

git log –oneline

git log –oneline 3 (it provide the 3 no commit)

now we are going to work with remote repository like (github,bitbucket ,gitlabetc..)



Go to github.com and create your account

Create a repository on github

And copy HTTP link

Again go to git commandprompt

Git remote //(there is no remote now you just create a remote)

Git remote add origin <https://github.com/chinmay/myripo.git>

//(here origin is the allias name. you can choose here any name )

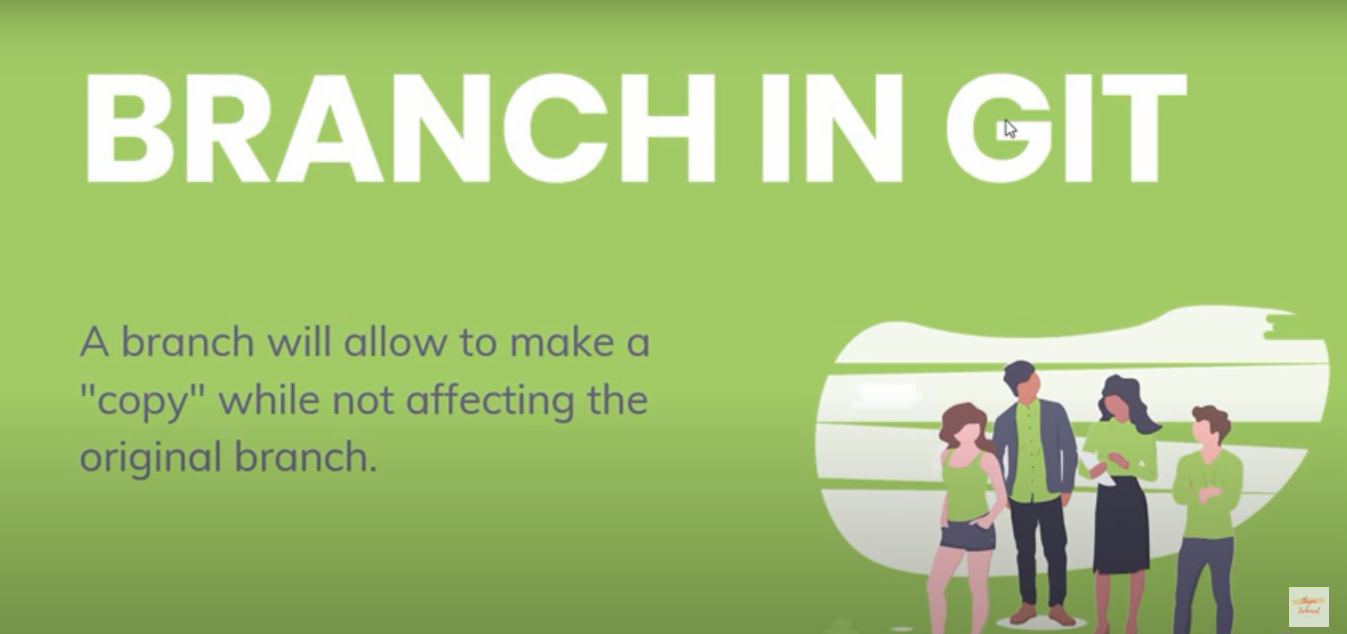
Git remote

Git remote --version

Git push –u origin master //-u create the tracking between local to remote and then push the local commit.

//instead of origin you can write the previous URL because origin working as a alias name

What is branch in git repository ?



Git branch //default branch is master

Git branch chinmaybranch // Command for create a branch

Git checkout chinmaybranch // Command for Switch to this branch

//now go to previous file or code do some changes here

Again

Git add .

Git commit –m ‘message’

Git push origin chinmaybranch //local branch name

Git checkout master //command for switch to master

git merge chinmaybranch //now chinmay branch is merge to the master branch by using first forwared method

Git push –u origin master



Create a folder

Go inside this folder

Go to your github remote repository click on clone button and copy the path and write

Git clone <https://github.com/chinmay/myripo.git>

Cd chinmay

git log –oneline //you can watch here all the previous commit

GIT IGNORE

Touch gitignore //create gitignore file

Add externally that file inside this directory

Git status

Git add .

Git commit -m ‘git ignore added’

Git push

A .gitignore file is a list of files to ignore when pushing files to Github.com. It tells git which files should be ignored when changes are staged (added) and committed. Some files are commonly not tracked by git so they are not committed nor sent up to repositories. Sending and receiving too many files can slow down development and takes up unnecessary space.

From the Github specs...

The purpose of gitignore files is to ensure that certain files not tracked by git remain untracked.

Create a .gitignore file all your own:

These steps use vim to update the .gitignore file

1. Navigate to the folder that contains the files for your project.
2. If you have not yet created a .git file, run the git commit command.
3. Create a .gitignore file by running touch .gitignore. The file name ".gitignore" is case sensitive and the name of the file matters. Git will look for a file with that name and will not stage (add) or commit files that is is told to ignore.
4. Use vim to open the file by running vim .gitignore. This will open a text editor called "Vim" inside your console.
5. Press the escape key to enter and exit text-entry mode.
6. Once you have updated your .gitignore files you can save and exit vim by pressing escape, entering :wq, and pressing return/enter

Example:

# Don't track content of these folders

node\_modules/

someOtherfoler/

# Compiled source #

###################

\*.com

\*.class

\*.dll

\*.exe

\*.o

\*.so

\*.so

# Packages #

############

# it's better to unpack these files and commit the raw source

# git has its own built in compression methods

\*.7z

\*.dmg

\*.gz

\*.iso

\*.jar

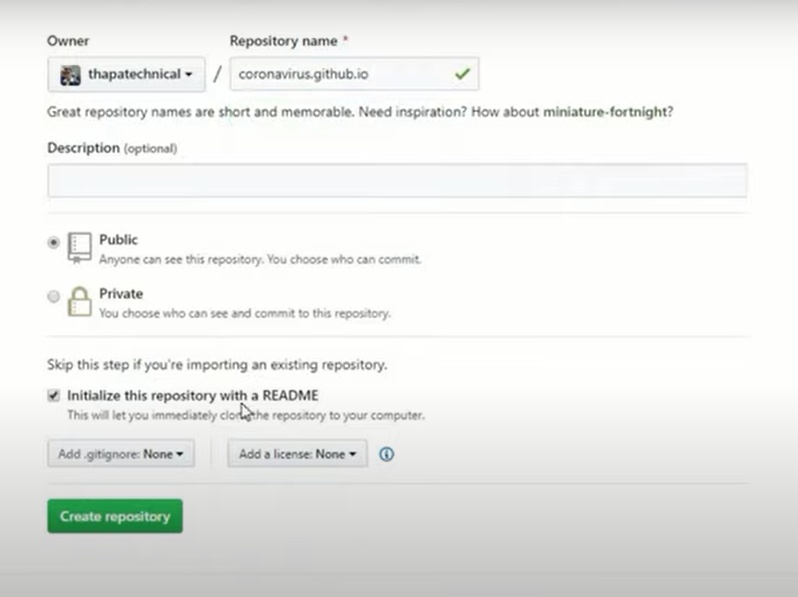
\*.rar

\*.tar

\*.zip

How to host our application on github

First youhave to give your remote repository name like that



Then download github desktop for window 10

You can do all the things here