








Download : <https://www.git-scm.com/download/win>

In VS code use one extension call **git Graph** for picturotical representation

- LIKEWISE

Graph	Description	Date	Author	Comi
	 feature1  origin Adding admin service	29 Jun 2023 1...	navin	fdf874
	experimenting with user avatar	29 Jun 2023 1...	navin	9f66c9
	 main  origin  v1.1 processing user data	27 Jun 2023 1...	navin	2a249
	 v1.0 User service created	26 Jun 2023 1...	navin	ff0ea7
	first commit	26 Jun 2023 1...	navin	3b549

To check the Git install successfully? Use - git

To check list - ls

To change the directory - cd desktop

To make any directory - mkdir

To check all hidden file - > ls -a

To initialize Git : git init

To create new file : touch name.text

To check the status of file or folder which not saved - git status

To remove -> rm -rf fileName OR rm rf --cached fileName

To getting ready for all - >git add . (. means all )

To getting ready for single on - >git add name.text

To take SS - git commit -m "Message"

To input message while committing use -m

To add data in file - vi name.txt

To displayed records in file-> cat name.txt

To check the all commit ( check log) - git log

To roll back you deleted - git rest e50282fa34670c2638550368b04fce01ba8ee0a2

To move fileback offstage - git stash

To fetch stash file - git stash pop

To denia any changes us -> gt stash clear

To check the difference b/w git diff --staged (to check in Staging area)

git show

To check all the Tag use git tag

To check log in line us -> git log --pretty=oneline

## GIT HUB

To add the project with git command - git remote add origin URL (git remote add origin

<https://github.com/Abhishekkumar503/My-Learning.git>)

To check the command which used - git remote -v

To push you work into master - git push origin master

```
Abhishek@DESKTOP-EGDA4B0 MINGW64 ~/desktop/Archit (master)
$ git push origin master
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (9/9), 1.07 KiB | 364.00 KiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/Abhishekkumar503/My-Learning.git
e50282f..dcc5ca7 master -> master
```

Master is main branch

To create - Touch gitTest.txt

To make any changes us - vi fruits.txt

To create feature branch - git branch feature

To switch branch to branch - git checkout feature / main

To merge you changes - git merge features

To **Clone** - git clone URL ( git clone <https://github.com/Abhishekkumar503/students.git>)

```
Abhishek@DESKTOP-EGDA4B0 MINGW64 ~/desktop/Archit/Fork (master)
$ git clone https://github.com/Abhishekkumar503/students.git
Cloning into 'students'...
remote: Enumerating objects: 48, done.
remote: Counting objects: 100% (14/14), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 48 (delta 7), reused 7 (delta 7), pack-reused 34
Receiving objects: 100% (48/48), 285.48 KiB | 2.80 MiB/s, done.
Resolving deltas: 100% (10/10), done.
```

To create new branch - *git branch abis* OR *git checkout -b branch name* OR (with switch and create branch USE ) *git switch -c feature2*

To switch branch to branch - git checkout abis OR git switch branch name

To stag - git add .

To commit - git commit -m “ “

To push - git push origin Abis

IF want to switch last branch USE *git switch -*

To delete the branch use *git branch -d branchname* OR *git -delete branchname*

To push in new branch

```
Abhishek@DESKTOP-EGDA4B0 MINGW64 ~/desktop/archit/fork/students (Abis)
$ git push origin Abis
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 418 bytes | 209.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'Abis' on GitHub by visiting:
remote:   https://github.com/Abhishekkumar503/students/pull/new/Abis
remote:
To https://github.com/Abhishekkumar503/students.git
 * [new branch]      Abis -> Abis
```

To commit forcibly - git push origin Abis -f

To fetch all commits - fetch --all --prune

To reset the main of the upstream main branch -reset --hard upstream/main

Git pull upstream main

To stash and commit at the sametime : git add .; git commit -m "1" { 1,2,3,4,5

To merge in two ways

**FIST** all in single commits - rebase -i 8177941c11162e6aab2959f7fb1cff35d485ae33

**SECOND** : To merge : first switch to main branch ( *git switch main* )then merge it USE *git merge branch name*

PUSh : git push origin main

To move back to parent : cd ../

To rename : mv Archit My folder

Merge conflict :

Pushing from Local : <https://www.youtube.com/watch?v=xwIQimbwJJE>

step 1.

download and install git bash into your local computer according to your operating system

you can download from official website

<https://git-scm.com/downloads>

once you have installed git then Open Git Bash.

Now

step 2. is generate ssh-key, this is used for authentication

---

the command is `ssh-keygen -t rsa -b 4096 -C "tarique.rkl@gmail.com"` and your GitHub email address,

This will Generate public/private rsa key pair.

"Enter a file in which to save the key," press Enter. This accepts the default file location.

Enter passphrase (empty for no passphrase): I will recommend enter here something and also remember that then enter again

Ensure the ssh-agent is running: `-- eval $(ssh-agent -s)`

Now Add your SSH private key to the ssh-agent. --for adding run command `$ ssh-add ~/.ssh/id_rsa`

Now you have to add public SSH key to your GitHub account

so copy your public key by running this command `clip "less then symbol" ~/.ssh/id_rsa.pub`

Now go to your github account click on your profile icon then go to settings click on SSH and GPG key then click on new SSH-key

give a title any thing you like then paste your key here control v then click on "add ssh key" then it will ask for one time password

confirm password, so we have added key successfully

Till now whatever I have done you have to do only once for your system

step 3. Now create a new repository,

---

choose repository name

Now you have to choose either public or private I am leaving it with default public option

Now if you want readme.md file then initialize it otherwise leave it then click on "create repository"

Now leave this screen as it is and go to your local directory where your code is,

open git bash in the same directory

Now initialize git -- git init

then git add . --- it will add all the file into stash ignore warning

then git commit -m "first commit" now it will commit all the changes locally

now add the remote repository url-- git remote add origin git@github.com:Tariqu/test.git

now we are ready to push our code enter commad --- git push origin master

now go to your repository and referesh now you can see all the files and folder here

## SSH Key

```
navin@Navins-Mac-Studio git-course % ssh-keygen -o
```

To open SSH file use → cat .....

## Tagging

Two ways tagging

Lightweight Tagging

Annotated Tagging

```
● navin@Navins-Mac-Studio git-course % git tag
● navin@Navins-Mac-Studio git-course % git tag -a v1.0 -m "1st release"
● navin@Navins-Mac-Studio git-course % git tag
v1.0
○ navin@Navins-Mac-Studio git-course %
```

## To update the tag

```
● navin@Navins-Mac-Studio git-course % git tag v1.1 -m "27th June release"
● navin@Navins-Mac-Studio git-course % git tag
v1.0
v1.1
○ navin@Navins-Mac-Studio git-course %
```

## To push the tag

```
navin@Navins-Mac-Studio git-course % git push origin v1.0
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 157 bytes | 157.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:navinreddy20/git-course.git
 * [new tag]          v1.0 -> v1.0
navin@Navins-Mac-Studio git-course %
```