GIT

- Git is version control system. Git is a DevOps tool used for source code management.
- it is distributed version control system.
- Git is used to collaborate on code.

- Github- GitHub is a code hosting platform for version control and collaboration.

✓ After successfully install Git into the system.

1)check the version of the git

$git --version

git version 2.36.1.windows.1

(get the current veersion of the git)

2)create repository

create folder in local drive with name anything

open folder right click and see two option

Git GUI here

Git Bash here

click on 'Git Bash here'

3)type command

$git init

Initialized empty Git repository in D:/0_2Python program/Git_Repo/.git/

(Your folder is set as repository. inside folder see the hidden .git folder)

4)then next

$git status

On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

5)File status life cycle-

i)untracked-->every file created is untracked file.

ii)modified -->Any file which is already tracked by git, but it is modified

in working directory is said to be in Modified State.

iii)staged -->These files are ready for commit.

iv)committed-->Any file which is committed is said to be In Repository/Committed State.

type command to create file

$touch Test.txt

(Test.txt file created under your repository)

$git status

On branch master

No commits yet

Untracked files:

(use "git add <file>..." to include in what will be committed)

Test.txt

(here Untracked files is shows)

$ git add Test.txt

(add Test.txt to staged area)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: Test.txt

$ start Test.txt

(Open Test.txt write code and save txt file)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: Test.txt

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: Test.txt

$git add Test.txt

(add to staged)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: Test.txt

6) create two file Test_1 and Test_2 txt in repository folder

$git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: Test.txt

Untracked files:

(use "git add <file>..." to include in what will be committed)

Test_2.txt

Test_3.txt

7)Add multiple untracked file into staged area

$git add .

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: Test.txt

new file: Test_2.txt

new file: Test_3.txt

8)commit the all files

$ git commit -m 'initial commit'

[master (root-commit) 8d63b73] initial commit

3 files changed, 1 insertion(+)

create mode 100644 Test.txt

create mode 100644 Test_2.txt

create mode 100644 Test_3.txt

9) modify the Test_2 file

$ git status

On branch master

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git restore <file>..." to discard changes in working directory)

modified: Test_2.txt

no changes added to commit (use "git add" and/or "git commit -a")

10) Test_2 is modified stage

$git add .

(Changes will reflect on particular files)

11)Create GitHub Account

create repository

Generate ssh key

$ ssh-keygen -t ed25519 -C "your_email@example.com"

12)Add agent to ssh key

$ eval "$(ssh-agent -s)"

Agent pid 1558

i)**$ ssh-add ~/.ssh/id_ed25519**

Identity added: /c/Users/Shreee/.ssh/id_ed25519
(jagtapvijay006@gmail.com)

ii)**$clip < ~/.ssh/id_ed25519.pub**

(Copy key)

New ssh key

13)Config your username and email

$ git config --global user.name vijay

$ git config --global user.email jagtapvijay006@gmail.com

$ git config --global user.username vijaypython2022

14)Check user,username,email register or not

$ git config --global user.name

vijay

$ git config --global user.username

vijaypython2022

$ git config --global user.email

jagtapvijay006@gmail.com

15)current repo commit history

$git log

commit 8d63b73d39f579d093cb1074adb820e185d7be7c

Author: vijaypython2022 <107676033+vijaypython2022@users.noreply.github.com>

Date: Mon Jun 27 15:00:42 2022 +0530

    initial commit

16)repository copy the ssh key

$ git remote add origin git@github.com:vijaypython2022/demo.git

(fetch or push operation you will see remote address)

17)Check remote url

$ git remote -v

origin https://github.com/vijaypython2022/demo.git (fetch)

origin https://github.com/vijaypython2022/demo.git (push)

18)Set url https to ssh

$ git remote set-url origin git@github.com:vijaypython2022/demo.git

(update the url of origin)

$ git remote -v

origin git@github.com:vijaypython2022/demo.git (fetch)

origin git@github.com:vijaypython2022/demo.git (push)

19)upadte the origin name to github (Rename)

$ git remote add github git@github.com:vijaypython2022/demo.git

20)Push the code on remote repo of github account

$ git push origin master

The authenticity of host 'github.com (13.234.210.38)' can't be established.

ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.

This key is not known by any other names

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.

Enumerating objects: 19, done.

Counting objects: 100% (19/19), done.

Delta compression using up to 4 threads

Compressing objects: 100% (10/10), done.

Writing objects: 100% (19/19), 1.38 KiB | 70.00 KiB/s, done.

Total 19 (delta 2), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (2/2), done.

To github.com:vijaypython2022/demo.git

 * [new branch]   master -> master

21)Cloning github repository to local folder

create new folder. right click Git bash type command

$git clone [ssh key of repo]

(Here ssh key copy from remote repo then paste)

22)know the how many branch

$git branch

* master

23)Create New Branch

$git branch student

24)Switch current to new branch

$git checkout student

Switched to branch 'student'

25)Create new branch and switch at one command.

$git checkout -b python

26)create text file in new branch

$touch test.txt

write some code in file and save

27)add stage area

$git add .

28)**git commit -m 'branch added'**

29) check remote url we push exiting repo from another new branch

$git remote -v

github git@github.com:vijaypython2022/demo.git (fetch)

github git@github.com:vijaypython2022/demo.git (push)

origin git@github.com:vijaypython2022/demo.git (fetch)

origin git@github.com:vijaypython2022/demo.git (push)

here are remote list we can push or fetch data

30) Push the data demo.git repo from student

$ git push origin student

31)merge data student to master

$ git merge student

Updating a05c46d..43ced8b

Fast-forward

 python.txt | 2 ++

 1 file changed, 2 insertions(+)

create mode 100644 python.txt

32)After work done with new branch we are deleting student branch from local repo

$ git branch -d student

Deleted branch student (was 43ced8b).

30)delete the branch from github account repo

$ git push origin --delete student

To github.com:vijaypython2022/demo.git

- [deleted] student

31)delete specific file

$ git rm test.txt

32)check where modification if file is untracked stage.

change existing code and save.

$ git diff

diff --git a/Test.txt b/Test.txt

index 63aca56..ea9fa27 100644

--- a/Test.txt

+++ b/Test.txt

@@ -1 +1,3 @@

-version 1.2

\ No newline at end of file

+version 1.2

+

+version 1.3

\ No newline at end of file

$ git add .

33)check where modification done in staged file

$ git diff --staged

diff --git a/Test.txt b/Test.txt

index 63aca56..ea9fa27 100644

--- a/Test.txt

+++ b/Test.txt

@@ -1 +1,3 @@

-version 1.2

\ No newline at end of file

+version 1.2

+

+version 1.3

\ No newline at end of file

34)change status of the file staged to untracked

$ git reset Test.txt

Unstaged changes after reset:

M	Test.txt

35)get file content before modification means (Ctrn+Z)

$ git checkout Test.txt

Updated 1 path from the index

36)ignore the other file not related to me

$ touch .gitignore

one file is created.inside that file add ignore files with extention

$ git status

On branch master

Untracked files:

(use "git add <file>..." to include in what will be committed)

.gitignore

37) Any other developer edit code then we pull data into local repo from remote repo. New added code update in your local repo

$ git pull git@github.com:vijaypython2022/demo.git

38)Git stash command is used to temprorally caches any changes you have made to your working copy so you can switch something else and then come back recover them later.

It take uncommit changes both satged and unstaged save them for later use.

$git stash

39)Git Amend command is way to modify most recent commit

$git commit --amend -m 'update the message'