

0. Creating folder and entering there using git command

23 April 2022 00:26

1. Find my present working directory:[the folder I am in now]

pwd

2. Files that are present in my current folder:

Ls

Ls -a[for hidden files]

3. Go to another folder:

cd folder_name

4. Creating a directory/folder:

mkdir folder_name

5. Initialize the folder.

git init

6. Creating file in folder

touch file_name.ext

7. Open the file

open file_name

1.Set user name and email

04 September 2021 23:04

Setting user name and email

```
git config --global user.name nibras  
git config --global user.email nibrasnib1526@gmail.com
```

Showing user name and email

```
git config --global user.name  
Git config --global user.email
```

Change User Name and email

```
git config --global --replace-all user.name any_name  
git config --global --replace-all user.email any_email
```

2.Open Vs code

04 September 2021 23:15

code .

[should be a space after code]

3.Git initialize

04 September 2021 23:21

**Will initialize empty git repository
Or reinitialize existing git repository**

git init

4.hidden folder + create .git

05 September 2021 04:31

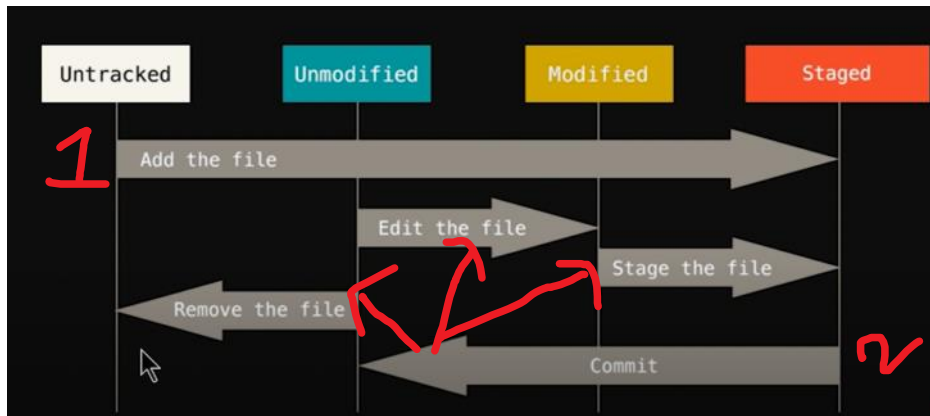
It will show all hidden folder and .git folder

ls -la

5. Untracked and stage area

05 September 2021

04:42



To check file is tracked or untracked:

`git status`

To go to the staging area and track a single file:

`git add file_name.ex-`

To start tracking of project, Do Initial commit:

`git commit -> enter -> i -> any message -> esc+w+q -> enter`

To create blank file:

`touch file_name.ext`

To go to the staging area and track ALL files:

`git add -A`

Stage all changed file in directory but not in subdirectories

`git add .`

Stage Selected types of file only in directory

`git add *.ext`

Stage selected types of file in directory+subdirectory

`git add **/*.ex`

Discard the change in working directory

git restore file_name.txt

Commit all files fastest way:

git commit -m "any messages"

Reset commit and undo it , means take the commit to staging area

git reset --soft HEAD^

Reset commit and also removing from staging area

Git reset HEAD^

Reset commit and also go back to the previous commit [completely deleting the commit]

git reset --hard HEAD^

Deleting number of Commits

git reset -soft HEAD~2

Know the head commit[head=latest commit]

git show head

Know the commit Id

Git log -> then commit will show there, commit id will show there

Navigate commit

git checkout commit id

6. Useful command

05 September 2021 04:47

To bring back previous commit if I mistakenly edit this

`git checkout filename.ext`

`git checkout -f -->for all files`

To know all the commit history of this folder

`git log`

`git log -p -1 -->1 er bodle any number..to watch last 1 ta commit`

To compare between working area and staging area:

`git diff`

[if working area and staging area are same nothing will show]

To compare between working area and last commit:

`git diff --staged`

To check where file is modified

`git status -s --->[In 1st position means modified in stage area,2nd means working area]`

Create .gitignore

`touch .gitignore[creating gitignore file]`

git reset

Makes all files untracked



A diagram illustrating the content of a .gitignore file. On the left, the text ".gitignore" is shown. A yellow arrow points from it to a list of file patterns on the right. The patterns are: "test.txt", ".env", "*.txt", "!main.txt", "test?.txt", and "temp/".

7.Deleting file

05 September 2021 06:30

To untrack the file:

git rm --cached file name

To delete the file

git rm file name

8.Branch

06 September 2021

13:33

Creating a branch

git branch anyname

Check all branches

git branch

Switch to new branch

git checkout branch_name

Come back to master branch

git checkout master

Create and switch to new branch at a time

git branch -b name

9.Remote Repository[public]

07 October 2021 01:25

✓ For Public repository:

-->First create a public Repository in github

★ Connect local repository with remote repository:

git remote add origin [copy the URL]

note: here origin is the name of the URL

★ Check the name of the URL

git remote -->[will show the URL]

★ Find the URL

git remote -v

To make sure you are up to date:

git pull origin master

★ Push the master branch in the URL[means github]

git push origin master

[-->pushing the master branch in github]

Note: if the repository is private you have to take access.

★ Change URL:

git remote set-url origin [copy the URL]

10.Remote Repository[private]

07 October 2021 02:39

✍ We have to use SSH key To Connect Local and Remote Private repository .

How to generate SSH key?

Link: [How to Create and add SSH key to GitHub ? || SSH Key || GitHub](#)



\$

11)push branch

07 October 2021 03:18

★ Push the master Branch:

git push origin master

★ Push another branch:

git push -u origin branch_name



11)Clone

07 October 2021 03:29

★ Clone a repository:

git clone URL any_gitname[for creating a folder]

12)merging 2 branch

December 15, 2023

6:11 PM

- **Scenario:**

- **Theres 2 branches. 1) master 2)chance-1**
- **Something changed in chance-1**

- **Merging the chance-1 branch into master branch:**

- **First go to master branch:**
 - git checkout master
- git branch --merged
- git merge [branch name (chance-1)]
- git push origin master

13)MARKDOWN Tags.

19 April 2022 05:01

Markdown tutorial

1)How to Comment:-

\<!--[Write anything here will be comment]-->

2)How to go new line:-

Give two space and it will start from new line.

3)How to do straight line:-

---[3 dash] will create a straight line like this.

4)Heading:-

'#'###'.....Just like this.

5)Paragraph:-

<p> Here will be the paragraph</p>

6)Italic:-

between two underscore

7)Strikethrough

~~Between these~~

8)Code block:-

1. Single Line:-

`between these`

2. Multiple Line:-

```
```C++
 Your code
```
```

9)Order+Unorder List

- 1.order
- 2.order
- unordered
- unordered

10)Task List:-

- [X] Task[Not Complete]
- [] Task[complete]

11)Link:-

- Direct
<http://www.lol.com>
- Link With Title
[title](http://www.lol.com)
- third type
[Name]: <http://www.lol.com>


```
[lol website][Name]
## 12)IMAGE
![Title](SOURCE)
or
<img src="" width="" />
```

```
## 13)Table:
col1	col2
Mohammad	Nibras
```

```
## 15)BOLD
__Bold__
```

```
## 16)Bold + Italic
***BOLD***
```