

# Must Git Commands for every Developer

Create New Repository

**git init**

Returns Staging Area

**git status**

Add File to Staging Area

**git add directive\_guide.txt**

To Create New File

**touch file\_name.extension**

For Adding every Untracked File to Staging Area

**git add .**

Forcefully Remove (Completely Delete) the File

**git rm -f filename.extension**

Remove The File From Staging Area Or Untracked It

**git rm --cached filename.extension**

It Keep Record Of Your Changes

**git commit -m "your messege"**

Committing Untracked Files

**git commit -a -m "your messege"**

# Must Git Commands for every Developer

Go Back In Time Temporary At Any Commit

**git checkout logid**

Switch Back To Original Branch

**git checkout master**

Get Back You In Time Permanently But For A Single Commit

**git revert LOGID**

Displays All The Message Of All The Commit At Once

**git log --oneline**

(Take Back In Time Mulpile Commit) And (Set That As Working Directory) And (Removes Files & ALL THE CHANGES MADE AFTER THE COMMIT LOGID)

**git reset --hard LOGID**

It Makes Your Working Directory In Past Time Without Removing Any File And Changes

**git reset --soft LOGID**

It Makes Your Working Directory In Past Time Without Removing Any File But Changes That Are Made After LOGID

**git reset --mixed LOGID**

Enter The Names Of File You Want To Ignore

**touch .gitignore**

to make .gitignore work and for removing the track of the ignored file

**git rm -r --cached .**

# Must Git Commands for every Developer

Create Newbranch

**git branch branchName**

Create New Branch And Switch To It

**git checkout -b branchName**

List All The Branch

**git branch -a**

Delete The Branch

**git branch -d branchName**

It Merge Branch To The Current One

**git merge branchName**

Playing With Other Repositories

**Forking (available in GUI in website)**

.git is missing from source

**download (available in GUI in website)**

To Create New Folder

**mkdir**

Connecting Local Origin To A Link To Access To Your Repository

**git remote add origin httpURL**

# Must Git Commands for every Developer

Displays Available Function Such As Fetch/Push

```
git remote -v
```

It Will Pull All The Data And Update Our Local Repositories

```
git pull
```

It Will Pull Data From Particular Branch And Will Update Particular Branch In Local

```
git pull <remote> <branchName>
```

Set Tracking Information For This Branch

```
git branch --set-upstream-to=origin/<branch> master
```

To Remove File

```
rm -rf fileName
```

To Remove File

```
rmdir fileName
```

Some Simple but very important Commands

```
git push -u origin master
```

```
git config --global user.email "Your email"
```

```
git config --global user.name "Your Username"
```

```
git --help
```

```
git commit --help
```

```
add README.md
```

```
add LICENSE
```

# Must Git Commands for every Developer

Displays All Your Commit

**git log**

Delete Branch On Github

**git push --origin delete branchName**

Basics Commands are shown here of Git. "How to use them?", "What they mean to us?" Struggling with this kind of question! Don't worry you have many options to learn their usage. I will be providing a video tutorial to help you all in this struggle. Be active on this site for Updates.

Drop mail at "thecreator.3999@gmail.com" for any guidance or any suggestion related to this command sheet.

