

DATE:13/07/2022

Documentation for Git

Git Bash is an application for Microsoft Windows environments which provides an emulation layer for a Git command line experience.

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.

COMMANDS USE IN GITBASH :-

1. **git config --global user.name "[name]"** This command sets username.
2. **git config --global user.email "[email address]"** This command sets email.
3. **git init** command creates a new Git repository. It can be used to convert an existing, unversioned project to a Git repository or initialize a new, empty repository. Most other Git commands are not available outside of an initialized repository, so this is usually the first command you'll run in a new project.
4. **mkdir** Creates a directory or subdirectory. Command extensions, which are enabled by default, allow you to use a single mkdir command to create intermediate directories in a specified path. This command is the same as the md command.
5. **git --version** command is used to check your current version of Git.
6. **cd** command, also known as chdir (change directory), is a command-line shell command used to change the current working directory in various operating systems.
 - cd~ :-** takes to home directory.
 - cd.. :-** takes to one level up directory.
 - cd- :-** takes to previous directory.
7. **git config list** command will show all Git config properties throughout all of the variously scoped Git files.
8. **git status** command displays the state of the working directory and the staging area. It lets you see which changes have been staged, which haven't, and which files aren't being tracked by Git.
9. **touch** command creates a file only if the file doesn't already exist.

10. **git add** command adds a change in the working directory to the staging area.
11. **git commit** sends the latest changes of the source code to the repository, making these changes part of the head revision of the repository.
12. **git log** command displays all of the commits in a repository's history.
13. **. gitkeep** command is to solve problem of Git not pushing empty folders to remote DVCS repos like GitHub or GitLab. To get Git to recognize an empty directory, the unwritten rule is to put a file named `. gitkeep` in it.
14. **git pull** command is used to fetch and download content from a remote repository and immediately update the local repository to match that content.
15. **git push** command is used to upload local repository content to a remote repository. Pushing is how you transfer commits from your local repository to a remote repo.
16. **git clone** is primarily used to point to an existing repo and make a clone or copy of that repo at in a new directory, at another location. The original repository can be located on the local filesystem or on remote machine accessible supported protocols. The git clone command copies an existing Git repository.

Q. 1) How to add an existing folder in a Git repository?

ANS :-

1. Create a repository on Github.

2. Go to your project folder :

```
$ cd /path/to/my/project
```

3. Add your project files to the repository by adding these commands in git Bash:

```
$ git init
```

```
$ git add .
```

\$ git commit -m "Commit Message"

4. Write the following command, replace <username> with your Github username

and <repository> with your repository name :

\$ git remote add origin https://github.com/<username>/<repository>.git

5. Push the files on the remote server

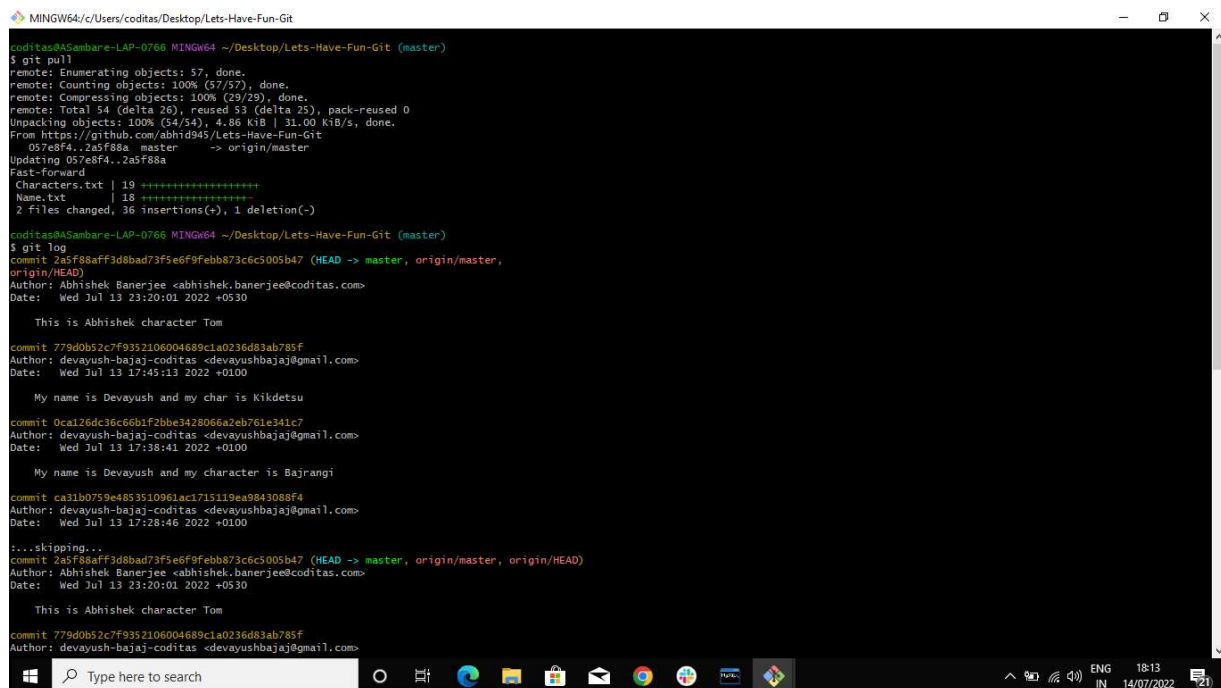
\$ git push -u origin master

GIT FUN ACTIVITY

This activity help us to find how conflicts occur while working on github.

And how to avoid them

Below is the picture of that activity.



```

MINGW64/c:/Users/coditas/Desktop/Lets-Have-Fun-Git
coditas@ASanbare-LAP-0766 MINGW64 ~/Desktop/Lets-Have-Fun-Git (master)
$ git pull
remote: Enumerating objects: 57, done.
remote: Counting objects: 100% (57/57), done.
remote: Compressing objects: 100% (29/29), done.
remote: Total 54 (delta 26), reused 53 (delta 25), pack-reused 0
Unpacking objects: 100% (54/54), 4.86 KiB | 31.00 KiB/s, done.
From https://github.com/abhishek94/Lets-Have-Fun-Git
   057e8f4..2a5f88a  master    -> origin/master
Updating 057e8f4..2a5f88a
Fast-forward
 Characters.txt | 19 +++++
 Name.txt       | 18 +++++
 2 files changed, 36 insertions(+), 1 deletion(-)
coditas@ASanbare-LAP-0766 MINGW64 ~/Desktop/Lets-Have-Fun-Git (master)
$ git log
commit 2a5f88aff3d8bad73f5e6f9febb873c6c5005b47 (HEAD -> master, origin/master,
origin/HEAD)
Author: Abhishek Banerjee <abhishek.banerjee@coditas.com>
Date:   Wed Jul 13 23:20:01 2022 +0530

    This is Abhishek character Tom

commit 779d0b52c7f9352106004680c1a0236d83ab785f
Author: devayush-bajaj-coditas <devayushbajaj@gmail.com>
Date:   Wed Jul 13 17:45:13 2022 +0100

    My name is Devayush and my char is Kikdetsu

commit 0ca126dc36c66b1f2bbe3428066a2eb761e341c7
Author: devayush-bajaj-coditas <devayushbajaj@gmail.com>
Date:   Wed Jul 13 17:38:41 2022 +0100

    My name is Devayush and my character is Bajrangi

commit ca31b0759e4853510961ac1715119ea9843088f4
Author: devayush-bajaj-coditas <devayushbajaj@gmail.com>
Date:   Wed Jul 13 17:28:46 2022 +0100

    ...skipping...
commit 2a5f88aff3d8bad73f5e6f9febb873c6c5005b47 (HEAD -> master, origin/master, origin/HEAD)
Author: Abhishek Banerjee <abhishek.banerjee@coditas.com>
Date:   Wed Jul 13 23:20:01 2022 +0530

    This is Abhishek character Tom

commit 779d0b52c7f9352106004680c1a0236d83ab785f
Author: devayush-bajaj-coditas <devayushbajaj@gmail.com>

```

```
MINGW64/c:/Users/coditas/Desktop/Lets-Have-Fun-Git
commit 2a5f88aff3d8bad73f5e6f9Febb873c6c5005b47 (HEAD -> master, origin/master, origin/HEAD)
Author: Abhishek Banerjee <abhishek.banerjee@coditas.com>
Date: Wed Jul 13 23:20:01 2022 +0530

    This is Abhishek character Tom

commit 779d0b52c7f9352106004680c1a0236d83ab785f
Author: devayush-bajaj-coditas <devayushbajaj@gmail.com>
Date: Wed Jul 13 17:45:13 2022 +0100

    My name is Devayush and my char is Kikdetsu

commit 0ca126dc36c66b1f2bbe3428066a2eb761e341c7
Author: devayush-bajaj-coditas <devayushbajaj@gmail.com>
Date: Wed Jul 13 17:38:41 2022 +0100

    My name is Devayush and my character is Bajrangi

commit ca31b0759e4853510961ac1715119ea9843088f4
Author: devayush-bajaj-coditas <devayushbajaj@gmail.com>
Date: Wed Jul 13 17:28:46 2022 +0100

    My name is Devayush and my character is Zarial

commit e2d49011ea0bebfd0dc969154b135419b485c6bf
Author: devayush-bajaj-coditas <devayushbajaj@gmail.com>
Date: Wed Jul 13 17:21:47 2022 +0100

    This is Devayush and my char is hero

commit 9b2335dc8f6d2108a83d7ad3c235c7162b2bb102
Author: devansh-sharma-coditas <devanshsharmads9935@gmail.com>
Date: Wed Jul 13 17:27:10 2022 +0530

    I am devansh and sher

commit ff9c52e395ae7637e6036614c3b805a5ed61c502
Author: devansh-sharma-coditas <devanshsharmads9935@gmail.com>
Date: Wed Jul 13 17:26:17 2022 +0530

    I am devansh and groot

commit 174434c58bb055ef6332c53b06e6d250106b4b2f
Author: devansh-sharma-coditas <devanshsharmads9935@gmail.com>
Date: Wed Jul 13 17:25:11 2022 +0530

    I am devansh and mlario

commit 2821b04b727b4443f21168c90dc0c0513cdd9050
:
```