Git Bash Commands

#Git

NIRUPASHREES

2021.03.30

B!

1 0 3

WHY GIT?

In today's scenario of real world problem solving mechanism Git has been so handy in continuous project development, version control and collaboration. Git acts like a portfolio for staging one's project which can be backed up, viewed and shared easily. Promotes easy collaborating from anyone from With version control we can keep track of any changes made with additional information like the person who modified, reason of modification, exact changes made and of course with the timestamp.

WHY GIT BASH?

Bourne Again Shell. Git bash allows the Git features to be used via the command line interface and in addition to that most of the Unix commands are also available. acts like an interface between Git and the local repository (local code copy).

This blog features a couple of the most used Git Bash commands while dealing with a continuous integration of a project.

INSTALLATION

For windows https://gitforwindows.org/ download Git bash from here.

A LIST OF GIT BASH COMMANDS:

SETUP

git config --global user.name “[name]”

This command sets username, which aids in reviewing by whom the changes were made.

git config --global user.email “[email address]”

This command sets an email address, this inturn helps in tracking by whom the commit or merge activity was made.

git config --global color.ui auto

This command sets an automatic command line coloring effect for easy reviewing.

CREATE AND INITIALIZE

git init

This command initialises the existing directory as a git repository

git init [repository name]

git clone [url]

This command is used to obtain the entire repository or download existing source code from the URL provided. Basically makes an identical copy of the latest version of the repository in the local system.

STAGING AND COMMIT

git add [file]

This command adds a file from the working tree to the Staging area / current branch.

git add *

This command adds one or more files from the working tree to the Staging area / current branch.

git add.

Stages all files in the entire repository to the Staging area / current branch.

git rm [file]

This command deletes the existing file from your working directory.

git status

This command lists all the files that have to be committed.

git diff

This command shows the changed contents that aren't staged.

git diff --staged

This command shows the changed contents that are staged but not committed.

git commit -m “[commit message]”

This command records or snapshots the changes permanently in the version history of the repository.

This command commits all the modified or created files to the repository.

git reset [file]

This command unstage a file without any change in the working directory.

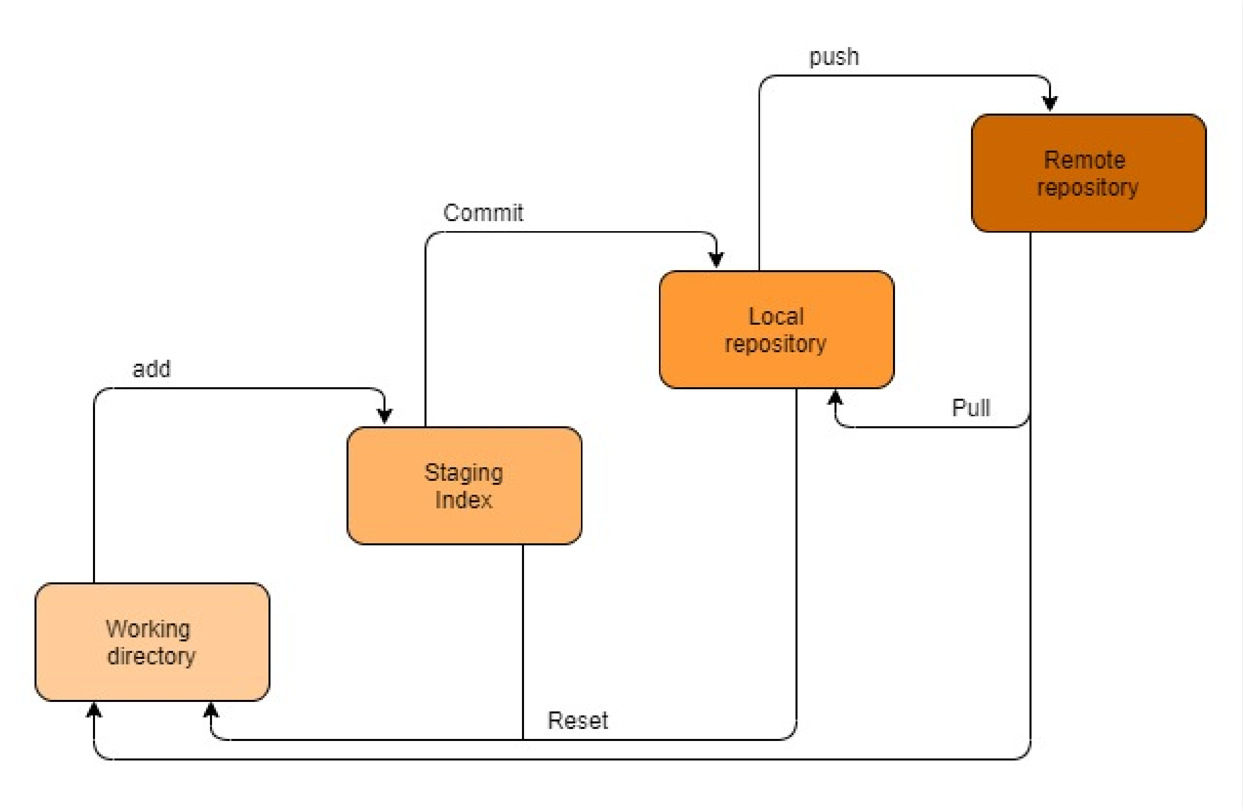
git reset [commit]

This command undoes all the commits after the specific commit mentioned and preserves the changes locally.

git reset --hard [commit]

This command discards all history and goes back to the specific commit mentioned ..

Bonus: Git commit saves your changes only locally.



Git Bash commands

BRANCH AND MERGE

git branch

git branch [branch name]

This command creates a new branch.

git branch -d [branch name]

This command deletes the feature branch.

git checkout [branch name]

This command is used to checkout to the branch specified.

git checkout -b [branch name]

This command creates a new branch and then checks out to the same.

git merge [branch name]

This command merges specified branches to the current branch.

LOGS AND REVIEW

git log

This command is used to list the version history for the current branch.

git log -follow [file]

This command particularly shows the versions of the file mentioned.

git show [commit]

This command shows the content change and metadata status of the specific commit.

REMOTE UPDATE AND SHARE

git remote add [variable name] [URL]

This command is used to connect your local repository to the remote server.

This command sends the branch commits to your remote server

git push --set-upstream [variable name] [branch name]

or

git push -u origin [branch name]

If branch is newly created, then we can upload with above command

git push [variable name] master

This command sends the committed changes of the master branch to your remote repository.

git push -all [variable name]

This command pushes all branches to your remote repository.

git pull [repository link]

This command fetches and merges changes on the remote server to your working directory.

Note: Git push only uploads changes that are committed.

Conclusion

Git bash is very handy to deal with Git's features and functions. Happy learning!

この記事をシェアする B!

関連記事

[小ネタ] Git履歴から単一ファイルを復元する

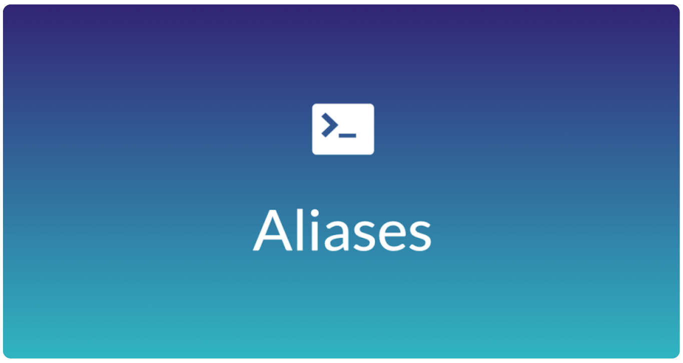
♥ 8

武田隆志 2022.03.30

Github等で利用するSSH keyの作成及び管理を1Passwordで行う

♥ 10

t_o_d 2022.03.17



Zsh のエイリアス: ターミナルと Git をスマートに扱う方法

♥ 0

AdarshParakh 2022.01.25

EC2からCodeCommitを使ったら便利だった

♥ 3

アシ 2021.12.31