Open bash file via notepad

apurva.misra@IN01L0201000171 MINGW64 ~

* $ notepad++ ~/.bash\_profile

Add editor as notepad for GIT file changes

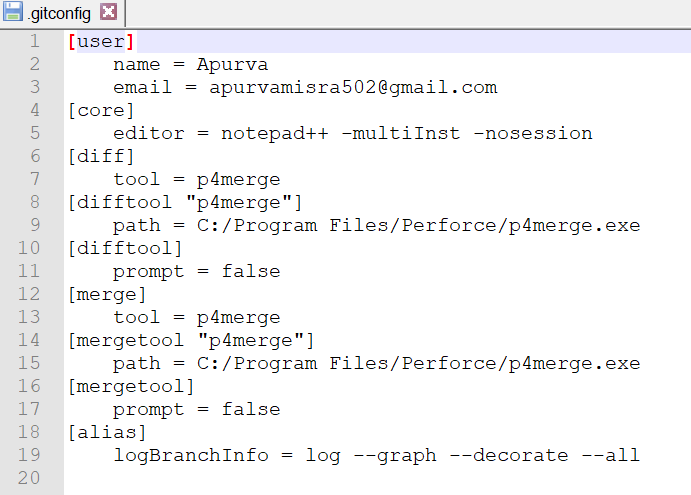
apurva.misra@IN01L0201000171 MINGW64 ~

* $ git config --global core.editor "notepad++ -multiInst -nosession"

Opens .gitConfig file (--global adds changes globally, if not used changes made will be applied to respective git directory)

apurva.misra@IN01L0201000171 MINGW64 ~

* $ git config --global -e



Add diff tool as p4Merge

apurva.misra@IN01L0201000171 MINGW64 ~

* $ git config --global diff.tool p4merge

Provide exe. Location for diff / merge tool

apurva.misra@IN01L0201000171 MINGW64 ~

* $ git config --global difftool.p4merge.path "C:/Program Files/Perforce/p4merge.exe"
* $ git config --global difftool.prompt false // no prompting before executing
* $ git config --global merge.tool p4merge //Add p4 as merge tool
* $ git config --global mergetool.p4merge.path "C:/Program Files/Perforce/p4merge.exe"
* $ git config --global mergetool.prompt false

Git Initialization :

* $ git init // Initialized empty Git repository in D:/workspace/GitLocal/.git/
* $ ls -al // listing file in directory
* $ start notepad++ TextFile1.txt // open a new file with name TextFile1.txt with notepad
* $ git status // check status of local branch

O/P:

On branch master

nothing to commit, working tree clean

* $ git add TextFile1.txt // Add the new created text file to staging area
* $ git commit -m "This is my first commit !" // Add comments regarding the commit
* $ git log // to print logs or check the history of commits

O/P:

commit 649fc56761689cc6f4e6ec9cfa3d00e37056c0db (HEAD -> master)

Author: Apurva <apurvamisra502@gmail.com>

Date: Wed Mar 3 01:09:21 2021 +0530

This is my first commit !

* $ git ls-files // tells which all file will be tracked by git

**O/P:**

TextFile1.txt

* $ touch newFile // creates new files
* $ rm -rf newFile // remove file
* $ git commit -am "This is modified file 2" // add files along with commiting
* $ start notepad++ TextFile1.txt // another modification

**O/P: $ 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: TextFile1.txt

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

* $ git reset HEAD TextFile1.txt // head reset / removes file from the staging area
* $ git checkout -- TextFile1.txt // last commited code on git
* $ cat TextFile1.txt // get content of a file

**O/P:**

hello ! this is textFile1 !

This files is modified !

New modification

* $ git help log // opens documentation section for git log command
* $ git log –decorate // decorated logs
* $ git log –all // show all logs
* $ git log --oneline // show one liner logs

**O/P:**

53cb358 (HEAD -> master) This is modified file 2

f621639 This is modified file 2

649fc56 This is my first commit !

* $ git log –graph

**O/P:**

\* commit 53cb3587c232d9511a2ff0dfe6d25273f182cbcc (HEAD -> master)

| Author: Apurva <apurvamisra502@gmail.com>

| Date: Wed Mar 3 02:11:27 2021 +0530

|

| This is modified file 2

|

\* commit f621639159a174dd9660a640941e9fa2c02031e0

| Author: Apurva <apurvamisra502@gmail.com>

| Date: Wed Mar 3 01:15:34 2021 +0530

|

| This is modified file 2

|

\* commit 649fc56761689cc6f4e6ec9cfa3d00e37056c0db

Author: Apurva <apurvamisra502@gmail.com>

Date: Wed Mar 3 01:09:21 2021 +0530

This is my first commit !

* $ git config --global alias.logBranchInfo "log --graph --decorate --all"

// create alias command names

* $ git logbranchinfo // this will call all set properties for that command by the alias names
* $ git config --global –list // list changes of global configuration

**O/P:**

user.name=Apurva

user.email=apurvamisra502@gmail.com

core.editor=notepad++ -multiInst -nosession

diff.tool=p4merge

difftool.p4merge.path=C:/Program Files/Perforce/p4merge.exe

difftool.prompt=false

merge.tool=p4merge

mergetool.p4merge.path=C:/Program Files/Perforce/p4merge.exe

mergetool.prompt=false

alias.logbranchinfo=log --graph --decorate --all

* $ git logbranchinfo -- TextFile1.txT // alias to specific files
* git mv Rename\ File.txt DemoRename.txt // renaming the committed file

Before ::

$ ls

Rename\ File.txt TextFile1.txt

After :

DemoRename.txt TextFile1.txt

* $ git rm DemoRename.txt // removing
* $ mv TextFile1.txt TextRename.txt // renaming outside git
* $ git add -u
* $ touch .gitignore // Git ignore file
* $ notepad++ .git

.git/ .gitignore

* $ notepad++ .gitignore
* $ git diff 50ec055 f621639

**O/P:**

diff --git a/.gitignore b/.gitignore

deleted file mode 100644

index bf0824e..0000000

--- a/.gitignore

+++ /dev/null

@@ -1 +0,0 @@

-\*.log

\ No newline at end of file

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

deleted file mode 100644

index e69de29..0000000

diff --git a/TextRename.txt b/TextFile1.txt

similarity index 75%

rename from TextRename.txt

rename to TextFile1.txt

index b232470..4c3aaa6 100644

--- a/TextRename.txt

+++ b/TextFile1.txt

@@ -1,3 +1,3 @@

hello ! this is textFile1 !

This files is modified !

-New modification

* $ git difftool TextRename.txt // opens the merge tool for differences
* $ git difftool

Branching/merge/delete :

* $ git branch // list all branches
* $ git checkout -b NewBranch // Checkout in branch. Also switches to new branch
* $ git checkout master // switch to master branch

$ git merge NewBranch

Updating 50ec055..1074904

Fast-forward

TextRename.txt | 5 +++++

1 file changed, 5 insertions(+)

* $ git branch -d NewBranch // delete new branch

Deleted branch NewBranch (was 1074904).

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* resolving conflicts \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* Create a new Branch and make a change from that branch and commit those changes

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ git checkout NewMerge

Switched to branch 'NewMerge'

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (NewMerge)

$ notepad++ TextRename.txt

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (NewMerge)

$ git commit -am "Commit from merge branch"

[NewMerge 1dbc90a] Commit from merge branch

1 file changed, 1 insertion(+), 1 deletion(-)

* Switch to master branch and make changes in the same file and commit those changes.

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (NewMerge)

$ git checkout master

Switched to branch 'master'

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ notepad++ TextRename.txt

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ git commit -am "Commit from mater branch"

[master 50aac34] Commit from mater branch

1 file changed, 1 insertion(+), 1 deletion(-)

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ git branch -a

NewMerge

\* master

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ git merge NewMerge

Auto-merging TextRename.txt

CONFLICT (content): Merge conflict in TextRename.txt

Automatic merge failed; fix conflicts and then commit the result.

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master|MERGING)

$ git mergetool

Merging:

TextRename.txt

Normal merge conflict for 'TextRename.txt':

{local}: modified file

{remote}: modified file

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master|MERGING)

$ git merge NewMerge

fatal: You have not concluded your merge (MERGE\_HEAD exists).

Please, commit your changes before you merge.

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master|MERGING)

$ git commit -am "Resolving conflicts~~"

[master e5917c5] Resolving conflicts~~

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ git status

On branch master

Untracked files:

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

TextRename.txt.orig

nothing added to commit but untracked files present (use "git add" to track)

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ notepad++ .gitignore

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ 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: .gitignore

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

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ git commit -am "ignoring orig file to merge"

[master d8b8ce4] ignoring orig file to merge

1 file changed, 2 insertions(+), 1 deletion(-)

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ git status

On branch master

nothing to commit, working tree clean

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* end resolving conflicts \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* tagging in git \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git tag newTag // providing tags to a commits
* $ git log –oneline // added tag can be viewed

**O/P:**

d8b8ce4 (HEAD -> master, tag: newTag) ignoring orig file to merge

e5917c5 Resolving conflicts~~

* $ git tag -d newTag // delete tags
* $ git tag -a v1.0 -m "Snapshot release v1.0" // adding tags with description
* $ git show v1.0 // view details about tags
* $ git log –oneline

**O/P:**

d8b8ce4 (HEAD -> master, tag: v1.0) ignoring orig file to merge

e5917c5 Resolving conflicts~~

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* git stash command \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* Make changes to a file and check its status

$ notepad++ Rename.txt

* $ git stash // Calling stash will remove changes applied

**O/P:**

Saved working directory and index state WIP on master: d8b8ce4 ignoring orig file to merge

* $ git stash list

**O/P:**

stash@{0}: WIP on master: d8b8ce4 ignoring orig file to merge

* Can make changes to other files after stashing previous changes and commit those file first.

$ notepad++ TextRename.txt

$ git commit -am "Stash check in master"

$ git stash pop // pops out the last changes stashed and can be changes/ committed

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* git reset/reflog \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

$ git log --oneline

6af0c1e (HEAD -> master) Stash check in master

d8b8ce4 (tag: v1.0) ignoring orig file to merge

e5917c5 Resolving conflicts~~

50aac34 Commit from mater branch

1dbc90a (NewMerge) Commit from merge branch

6d13a1a Git merge from master commit

1074904 Adding branch changes !

50ec055 adding the ignore file

60c8b32 Delete command / rename command tested

84ae743 After deletion commit

906823d after reanme commit

6c913fc Commit Before rename

53cb358 This is modified file 2

f621639 This is modified file 2

649fc56 This is my first commit !

* $ git reset e5917c5 –soft // reset head to a particular commit

**O/P:**

$ git log --oneline

e5917c5 (HEAD -> master) Resolving conflicts~~

50aac34 Commit from mater branch

* $ git reset 6d13a1a –soft
* $ git reset 6c913fc --hard

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* git reflog \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git reflog // gives logging of different HEAD changes

**O/P:**

6c913fc (HEAD -> master) HEAD@{0}: reset: moving to 6c913fc

6d13a1a HEAD@{1}: reset: moving to 6d13a1a

e5917c5 HEAD@{2}: reset: moving to e5917c5

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* linking git local to remote repo \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git remote add origin <https://github.com/Eelgrey/DemoGit.git>
* $ git remote -v

**O/P:**

origin https://github.com/Eelgrey/DemoGit.git (fetch)

origin https://github.com/Eelgrey/DemoGit.git (push)

* $ git push -u origin master –tag // push origin master with tags

**O/P:**

Enumerating objects: 40, done.

Counting objects: 100% (40/40), done.

Delta compression using up to 8 threads

Compressing objects: 100% (33/33), done.

Writing objects: 100% (40/40), 3.50 KiB | 132.00 KiB/s, done.

Total 40 (delta 14), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (14/14), done.

To https://github.com/Eelgrey/DemoGit.git

\* [new branch] master -> master

\* [new tag] v1.0 -> v1.0

Branch 'master' set up to track remote branch 'master' from 'origin'.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* setting up SSH communication \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal (master)

* $ pwd

/d/workspace/GitLocal

* $ mkdir .ssh // make a .ssh directory
* $ cd .ssh/ // change the directory to .ssh
* $ ssh-keygen -t rsa -C [apurvamisra502@gmail.com](mailto:apurvamisra502@gmail.com) // generate Key

**O/P:**

Generating public/private rsa key pair.

Enter file in which to save the key (/c/Users/apurva.misra/.ssh/id\_rsa):

Created directory '/c/Users/apurva.misra/.ssh'.

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Passphrases do not match. Try again.

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /c/Users/apurva.misra/.ssh/id\_rsa

Your public key has been saved in /c/Users/apurva.misra/.ssh/id\_rsa.pub

The key fingerprint is:

SHA256:ULNzQhUMo2PzNYQ2f7vrbjEzyaswDyP6yZALTTkP4CY apurvamisra502@gmail.com

The key's randomart image is:

+---[RSA 3072]----+

| \*=+. |

| ++=. |

| . \*.+o+ |

| . . o = =... |

|E o = S o o |

| o o = O |

| . + o = B |

| . = o \* + |

| o.+ o=+. |

+----[SHA256]-----+

$ ls -al // shows new files created under .ssh folder

total 4

drwxr-xr-x 1 apurva.misra 1049089 0 Mar 3 22:42 ./

drwxr-xr-x 1 apurva.misra 1049089 0 Mar 3 22:42 ../

$ cd c:/ //change drive

$ cd Users/apurva.misra/.ssh/ // got to .ssh folder

$ ls -al // list all files

$ notepad++ id\_rsa.pub // open this file

$ ssh -T [git@github.com](mailto:git@github.com) // add authenticity of GITHUB

**O/P:**

The authenticity of host 'github.com (13.234.176.102)' can't be established.

RSA key fingerprint is SHA256:nThbg6kXUpJWGl7E1IGOCspRomTxdCARLviKw6E5SY8.

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

Warning: Permanently added 'github.com,13.234.176.102' (RSA) to the list of known hosts.

Hi Eelgrey! You've successfully authenticated, but GitHub does not provide shell access.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*end setting up SSH communication \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Renaming a clone pull from git directly\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal/DemoWebsiteGit/websiteDemo (MASTER)

* $ git remote -v

origin git@github.com:Eelgrey/websiteDemo.git (fetch)

origin git@github.com:Eelgrey/websiteDemo.git (push)

* $ git clone git@github.com:Eelgrey/websiteDemo.git NewWebsiteDemo // cloning to diff name

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* initialize a dummy website \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* Download initializer from browser and then follow-

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal/DemoWebsiteGit/NewWebsiteDemo (MASTER)

* $ cp -R C:/Users/apurva.misra/Downloads/initializr-verekia-4.0/initializr/\* .

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* in case push does not work \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* git push // gives an error beyond vrs 2.0
* git config --global push.default simple

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* fetch/pull \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* make changes in git repo directly is some file and then make changes locally and commit them

$ notepad++ README.md

$ git commit -am "Changes made in the ReadMe.md file"

$ git status

**o/p:**

On branch MASTER

Your branch is ahead of 'origin/MASTER' by 1 commit.

(use "git push" to publish your local commits)

nothing to commit, working tree clean.

* On pushing changes , it fails and shows error.

$ git push

**O/P:**

To github.com:Eelgrey/websiteDemo.git

! [rejected] MASTER -> MASTER (fetch first)

error: failed to push some refs to 'github.com:Eelgrey/websiteDemo.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

* $ git fetch // fetch merges both the changes
* $ git pull
* $ git push
* $ git help config // opens browser with document of help config

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* changing git repo name \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git remote -v

origin git@github.com:Eelgrey/websiteDemo.git (fetch)

origin git@github.com:Eelgrey/websiteDemo.git (push)

* $ git remote set-url origin [git@github.com:Eelgrey/NewName.git](mailto:git@github.com:Eelgrey/NewName.git)
* $ git remote -v

origin git@github.com:Eelgrey/NewName.git (fetch)

origin git@github.com:Eelgrey/NewName.git (push)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*end changing git repo name \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git show c62dd5b9de61555ddaf0a83fa090f43013292054 // show commit info
* $ git remote show origin // show origin info

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* create/push local branch from git \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git checkout -b LocalBranch // Switched to a new branch 'LocalBranch'
* $ git push -u origin LocalBranch

**O/P:**

Enumerating objects: 4, done.

Counting objects: 100% (4/4), done.

Delta compression using up to 8 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 324 bytes | 108.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

remote:

remote: Create a pull request for 'LocalBranch' on GitHub by visiting:

remote: https://github.com/Eelgrey/NewName/pull/new/LocalBranch

remote:

To github.com:Eelgrey/NewName.git

\* [new branch] LocalBranch -> LocalBranch

Branch 'LocalBranch' set up to track remote branch 'LocalBranch' from 'origin'.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* remove branch from git directly and the merging in local \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git checkout MASTER
* $ git branch -d LocalBranch // delete local Branch
* $ git branch -a // list branches

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* update git branch in local using fetch \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git fetch -p

From github.com:Eelgrey/NewName

- [deleted] (none) -> origin/LocalBranch

remote: Enumerating objects: 9, done.

remote: Counting objects: 100% (9/9), done.

remote: Compressing objects: 100% (6/6), done.

Unpacking objects: 100% (7/7), 1.55 KiB | 93.00 KiB/s, done.

remote: Total 7 (delta 3), reused 3 (delta 1), pack-reused 0

c62dd5b..e03dc57 MASTER -> origin/MASTER

\* [new branch] NewExample -> origin/NewExample

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal/DemoWebsiteGit/NewWebsiteDemo (MASTER)

$ git branch -a

\* MASTER

remotes/origin/HEAD -> origin/MASTER

remotes/origin/MASTER

remotes/origin/NewExample

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* delete branch local and push to git hub \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git checkout GitBranch

Switched to a new branch 'GitBranch'

Branch 'GitBranch' set up to track remote branch 'GitBranch' from 'origin'.

* $ git checkout MASTER
* $ git push origin :GitBranch

To github.com:Eelgrey/NewName.git

- [deleted] GitBranch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* rebase code in git \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

$ notepad++ index.html

$ git commit -am "updating before Rebase check"

$ git fetch

$ git log --oneline

aed06a7 35de5a9 (HEAD -> MASTER) updating before Rebase check

e03dc57 Merge pull request #2 from Eelgrey/LocalBranch

* $ git pull –rebase // used to make a common base of multiple branches

Successfully rebased and updated refs/heads/MASTER.

* $ git log --oneline

95e71f3 (HEAD -> MASTER) updating before Rebase check

04d6fe4 (origin/MASTER, origin/HEAD) Rebase check from git repo

e03dc57 Merge pull request #2 from Eelgrey/LocalBranch

* $ git push

\* 95e71f3 (HEAD -> MASTER, origin/MASTER, origin/HEAD) updating before Rebase check

\* 04d6fe4 Rebase check from git repo

\* e03dc57 Merge pull request #2 from Eelgrey/LocalBranch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* git graph \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

$ git log --oneline --graph

\* 95e71f3 (HEAD -> MASTER, origin/MASTER, origin/HEAD) updating before Rebase check

\* 04d6fe4 Rebase check from git repo

\* e03dc57 Merge pull request #2 from Eelgrey/LocalBranch

|\

| \* da13487 This is file from localBranch

|/

\* c62dd5b Making file as .md file

\* aed06a7 Merge pull request #1 from Eelgrey/Eelgrey-BTSInfo

|\

| \* 7ecc780 Create NewAddedFile

|/

\* a18143b Merge branch 'MASTER' of github.com:Eelgrey/websiteDemo into MASTER

|\

| \* 7748f38 Add title on index file

\* | b740b24 Changes made in the ReadMe.md file

|/

\* 8db5cc7 This is first commit to website demo

\* 5dd43db Initial commit

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* change default branch \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

$ cd ..

$ ls -al

total 8

drwxr-xr-x 1 apurva.misra 1049089 0 Mar 4 15:55 ./

drwxr-xr-x 1 apurva.misra 1049089 0 Mar 4 15:50 ../

drwxr-xr-x 1 apurva.misra 1049089 0 Mar 5 17:16 NewWebsiteDemo/

* $ git clone git@github.com:Eelgrey/NewName.git

Cloning into 'NewName'...

remote: Enumerating objects: 65, done.

* $ ls -al

total 12

drwxr-xr-x 1 apurva.misra 1049089 0 Mar 5 17:49 ./

drwxr-xr-x 1 apurva.misra 1049089 0 Mar 4 15:50 ../

drwxr-xr-x 1 apurva.misra 1049089 0 Mar 5 17:49 NewName/

drwxr-xr-x 1 apurva.misra 1049089 0 Mar 5 17:16 NewWebsiteDemo/

* $ rm -rf NewWebsiteDemo/
* $ cd NewName/

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal/DemoWebsiteGit/NewName (MergeBranch)

$ git branch -a

\* MergeBranch

remotes/origin/HEAD -> origin/MergeBranch

remotes/origin/MergeBranch

remotes/origin/NewExample

remotes/origin/master

* $ git fetch

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* resolving conflicts \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ notepad++ LocalFile
* $ git commit -am "Update localFile with local chages"
* $ git fetch
* $ git pull

Auto-merging LocalFile

CONFLICT (content): Merge conflict in LocalFile

Automatic merge failed; fix conflicts and then commit the result.

* $ git mergetool

Merging:

LocalFile

Normal merge conflict for 'LocalFile':

{local}: modified file

{remote}: modified file

* $ git pull^C
* $ git commit -am "resolving conflicts with p4"

[MergeBranch 78440af] resolving conflicts with p4

* $ git status

On branch MergeBranch

Your branch is ahead of 'origin/MergeBranch' by 2 commits.

(use "git push" to publish your local commits)

Untracked files:

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

LocalFile.orig

nothing added to commit but untracked files present (use "git add" to track)

* $ git commit -am "Add changes"

On branch MergeBranch

Your branch is ahead of 'origin/MergeBranch' by 2 commits.

(use "git push" to publish your local commits)

Untracked files:

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

LocalFile.orig

nothing added to commit but untracked files present (use "git add" to track)

* $ rm LocalFile.orig
* $ git push

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* tagging in GIT \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git tag
* $ git tag stable master // tag master as stable branch
* $ git tag unstable MergeBranch // tag as unstable branch
* $ git tag

stable

unstable

* $ git log --oneline --decorate

78440af (HEAD -> MergeBranch, tag: unstable, origin/MergeBranch, origin/HEAD) resolving conflicts with p4

70b2306 Update localFile with local chages

f283c24 Update local via git

* $ git tag -a v1.0-Beta -m "This is the annotated beta v1.0" 5dd43db
* $ git tag -a v2.0-Alpha -m "This is the annotated alpha v2.0" da13487
* $ git tag -a v3.0-Alpha -m "This is the annotated alpha v3.0" 04d6fe4~
* $ git log --oneline –decorate

**O/P:**

78440af (HEAD -> MergeBranch, tag: unstable, origin/MergeBranch, origin/HEAD) resolving conflicts with p4

70b2306 Update localFile with local chages

f283c24 Update local via git

a6fda25 (tag: v1.0-Alpha) Merge pull request #3 from Eelgrey/NewExample

e03dc57 (tag: v3.0-Alpha) Merge pull request #2 from Eelgrey/LocalBranch

da13487 (tag: v2.0-Alpha) This is file from localBranch

* $ git show
* $ git tag

stable // light weight tags

unstable // light weight tags

v1.0-Alpha // annotated tags

v1.0-Beta // annotated tags

v2.0-Alpha // annotated tags

v3.0-Alpha // annotated tags

* $ git show v1.0-Alpha // show tag info

\*\*\*\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\* push local tag to GIT repo \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git push origin stable

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

To github.com:Eelgrey/NewName.git

\* [new tag] stable -> stable

🡪 $ git push --tags // push all tags

* $ git tag -d v2.0-Alpha // delete tags

Deleted tag 'v2.0-Alpha' (was d10bb99)

* $ git push origin :v1.0-Alpha // push deleted tags to GIT

To github.com:Eelgrey/NewName.git

- [deleted] v1.0-Alpha

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* floating tag \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git log --oneline --decorate

e2e4c5b (HEAD -> MergeBranch) Check floating tag

78440af (tag: unstable, origin/MergeBranch, origin/HEAD) resolving conflicts with p4

70b2306 Update localFile with local chages

* $ git tag -f unstable e2e4c5b

Updated tag 'unstable' (was 78440af)

* $ git log --oneline --decorate

e2e4c5b (HEAD -> MergeBranch, tag: unstable) Check floating tag

78440af (origin/MergeBranch, origin/HEAD) resolving conflicts with p4

* $ git push origin MergeBranch
* $ git push --force origin unstable

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* comparison using expresion \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

MergeBranch@{3days}

HEAD@{3}

HEAD-1/HEAD^

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* forking \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

go to some other repo --> click fork--> creates copy

--> clone repo into local--> create changes in a branch

--> push the new branch into repo -->create pull request // this pulls your current chnges in the upstream or fork branch

--> add comments and allow to pull changes #1637

--> again changing and pushing the file on git will automatically update the fork repo with your new changes on reloading

--> any user having write-in right on fork repo can accept your pull and merge

--> status changes from open to close to merge

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* create upstream origin for fork repo \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $ git remote -v

origin git@github.com:Eelgrey/starter-web.git (fetch)

origin git@github.com:Eelgrey/starter-web.git (push)

* $ git remote add upstream [git@github.com:scm-ninja/starter-web.git](mailto:git@github.com:scm-ninja/starter-web.git)
* $ git remote -v

origin git@github.com:Eelgrey/starter-web.git (fetch)

origin git@github.com:Eelgrey/starter-web.git (push)

upstream git@github.com:scm-ninja/starter-web.git (fetch)

upstream git@github.com:scm-ninja/starter-web.git (push)

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal/DemoWebsiteGit/starter-web (Feature)

* $ git pull upstream master

From github.com:scm-ninja/starter-web

\* branch master -> FETCH\_HEAD

\* [new branch] master -> upstream/master

Already up to date.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* creating gist \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* gist is basically a code snippet/ file that works as an independent repo on its own
* works same as repo--> copy-->clone--delete--add files

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* changes done at REPO level \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* issues : works on lable / milestone/ assigness
* label : to mark the issue/ can add or edit or delete label
* milestone : can mark milestone timelines | can edit /delete/ close/sort based on priority

apurva.misra@IN01L0201000171 MINGW64 /d/workspace/GitLocal/DemoWebsiteGit/Gist\_Demo (master)

* $ git commit -am "This is trial for issue closing from terminal, close #5"

[master ea02c4c] This is trial for issue closing from terminal, close #5

1 file changed, 2 insertions(+), 1 deletion(-)

\*\*\*\*\*\*\* close issue using terminal

// this associates commits no. to issues. ----> not working

//can refeernce issue id #ref\_no to comments in commits to associate it to issue closed

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* contributors \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* Study about it.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* organization \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* Create org like gist--> add name --> transfer repo to org
* fork account into ur own repo using fork technique

\*\*\*\*\*\*\* upstream fork org to local

* $ git remote -v

origin git@github.com:Eelgrey/NewName.git (fetch)

origin git@github.com:Eelgrey/NewName.git (push)

* $ git remote add upstrOrg [git@github.com:OrgEelgrey/DemoGit.git~](mailto:git@github.com:OrgEelgrey/DemoGit.git~)
* $ git remote -v

origin git@github.com:Eelgrey/NewName.git (fetch)

origin git@github.com:Eelgrey/NewName.git (push)

upstrOrg git@github.com:OrgEelgrey/DemoGit.git~ (fetch)

upstrOrg git@github.com:OrgEelgrey/DemoGit.git~ (push)

* $ git fetch --all

Fetching origin

Fetching upstrOrg

From github.com:OrgEelgrey/DemoGit

\* [new branch] master -> upstrOrg/master

\* [new branch] shared -> upstrOrg/shared

🡪 $ git branch -a

\* master

remotes/origin/HEAD -> origin/master

remotes/origin/master

remotes/origin/shared

remotes/upstrOrg/master

remotes/upstrOrg/shared

\*\*\*\*\*\*\*\*\*\*\* command to tell to create shared for which origin \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* $git checkout -b shared origin/shared

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* teams \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* Create team in org url
* search members
* multiple owner can be for a team
* invite/ cancel invite
* manage teams/ permissions/ access
* move people within teams / export info or org in json format

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* XXXXXXXX \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*