# 13th December 2023

In directory ex3

Create text file 1.txt 🡪 touch 1.txt

Type something in the file 🡪 vi 1.txt

Type this is America in that file and exit vim editor

Git init

Git add –A

Git commit –m “first commit”

Edit the 1.txt file again 🡪 vi 1.txt

Change the sentence to “this is USSR” exit vim editor

Commit after making changes message : second commit

Check git log

**GO TO ANY VERSION OF THE FILE USING CHECKOUT**

To go to the committed part of a particular commit 🡪 git checkout copy first 8 digits of the hash value of the wanted log \_filename\_ 🡪 git checkout f2f03fd0 1.txt

(THIS WORKS FOR BOTH PREVIOUS AND LATER VERSIONS OF THE FILES)

1. Go to the first commit part
2. Go to the second commit part

GIT BRANCHING

To when a new branch is made, a copy of the whole main is made in that branch and now we can make changes in this new branch. Provided the master is commited and then we can save the changes in the main by committing the branch

Go to ex4 folder

Create a repository

Create an empty txt file 🡪 touch 1.txt

Git add –A

Git commit –m “first commit”

TO CREATE NEW BRANCH 🡪 git branch br1 (do only after committing master)

TO GO TO THE BRANCH 🡪 git checkout *branchname*

*(in the branch there will be same data from its parent folder)*

Create 2 more empty files in branch br1 (2.txt and 3.txt)

Commit both files git commit -a -m "first branch commit"

Checkout to the default branch using git checkout

TO MERGE THE CHANGES FROM BRANCH TO HEAD

(always go to the default branch and only then merge the wanted branch)

Use command from default 🡪 git merge *branchname*

*And then commit*

Create a new branch br2 and edit file 1.txt, commit in branch then merge changes

Git branch br2

Git checkout br2

Vi 1.txt

Git add .

Git commit –m “changes in 1.txt from br2”

Git checkout master

Ls

Cat 1.txt (see if changes made from br2 reflect over here)

MERGE CONFLICT : when 2 devs are working on same file, one works on line 5 to 10 and another on 7 to 12. There will be conflict on line 7 to 10. We have to

Save in repository 🡪 commit

Create a new directory ex5

Touch one.txt

Vi one.txt 🡪 THIS IS AMERICAA

Git init

Git add .

Git commit –m

Git branch br1

Git branch br2

Git branch command without branch name lists all branches and \* tells on which dir our head is

Go to brznch 1 and go to 1.txt and edit Ukraine wins

Git add .

Git commit –m “br1 one.txt edited”

Git checkout parent

Ls

Cat one.txt

Go to branch 2

Edit one.txt to Russia wins

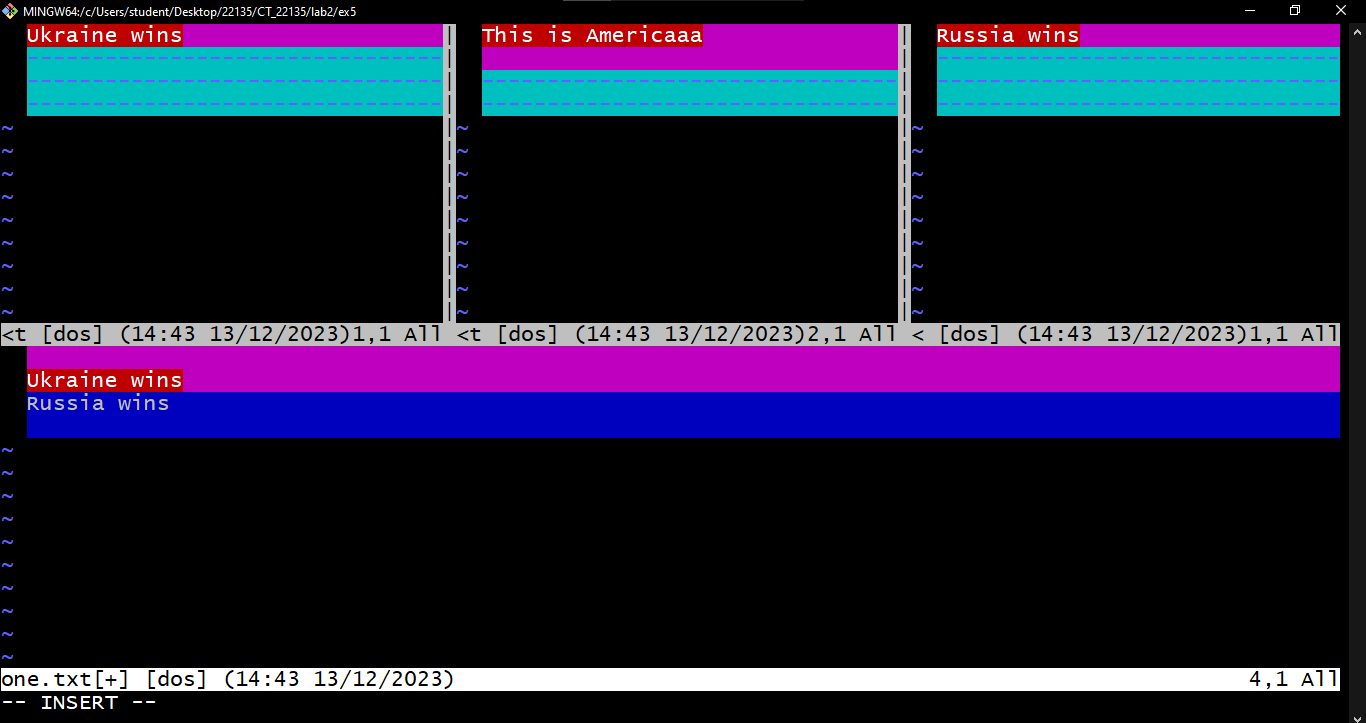
Git add .

Git commit –m

Git checkout parent

Git merge br2 (should get conflict eroor)

TO RESOLVE CONFLICT 🡪 git mergetool then hit enter key (new window will open)



Remove arrows and all of head and branch and keep pressing esc :wq (around 2 or more times) until you are out of that window

Save the conflict resolved changes by committing : git commit -m "conflict of branches resolved"

Do cat one.txt to check if changes are reflected

REMOTE REPOSITORY

CREATE AN ACCOUNT ON GITHUB

Kittycattymat

gitHUB2435

Billu27

On homepage click CREATE REPOSITORY BUTTON and name it

Make it public and name repo1\_13dec and click on add README file

On pc(local), create new folder ex6 and initialize it

On github , go to CODE button and copy the https link

Then on pc(local)

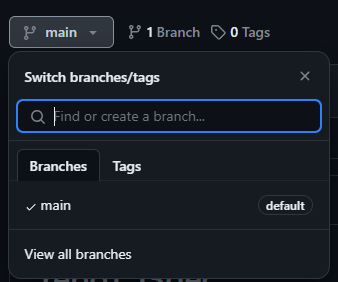
Type git remote add first https://github.com/Billu27/repo1\_13dec.git

First is the variable name with which that link will be identified, you can name it anything

ON PC to pull the repo

Git pull first main

Git pull *name of link*  *name of branch or main*

**

So we have pulled the data and files from remote repo to our local pc

We can name the branch too, that will pull (import) only that branch data

Do ls in pc and see if readme file has been pulled

Then

Touch one.txt

Git add .

git commit -m "first commit"

Then create a new branch

Then command 🡪 git push *var\_name* *branch\_name (*window should open to connect or else resolve using stackoverflow or yt)