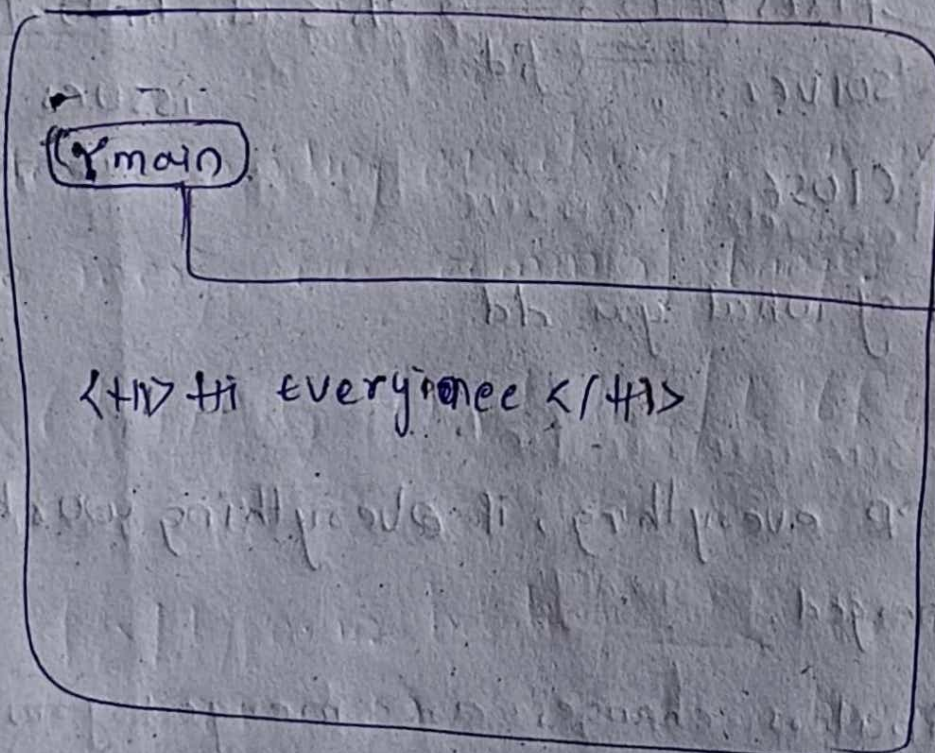


## DAY-2

### Git Branches and Pull requests

Step 1:-



you can see default original repo where  
branch where source code exists



\* By default main is the branch which appears on the repo

\* And in most of the case, the code of default branch (main) <sup>code</sup> exists on production/server.

\* In older repo's master is the default branch name

latest repo's main is the default branch name

⇒ Master & main both are same, which are default branches.

Ques :-

1. can we able to commit in main? Yes ✓
2. committing directly into main is good/not? not ✓

Step 2:-

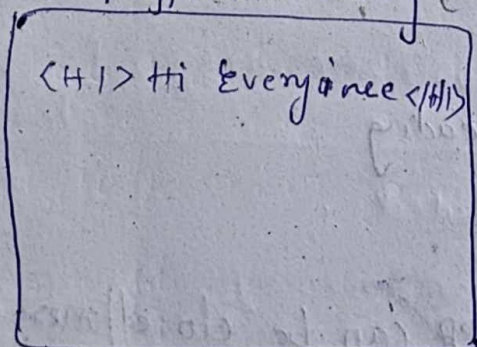
Creating Branch

\* committing changes directly into main is not good practice.

\* so create a branch from main, where you can code/contribute etc

\* Basically copy of entire src code/repo is forking but, copy of some part (like where you want to work & contribute) is known as branches

fix typo in heading (branch name)



main  
|  
typo-in-heading  
(branch name)

\* You will get copy/part



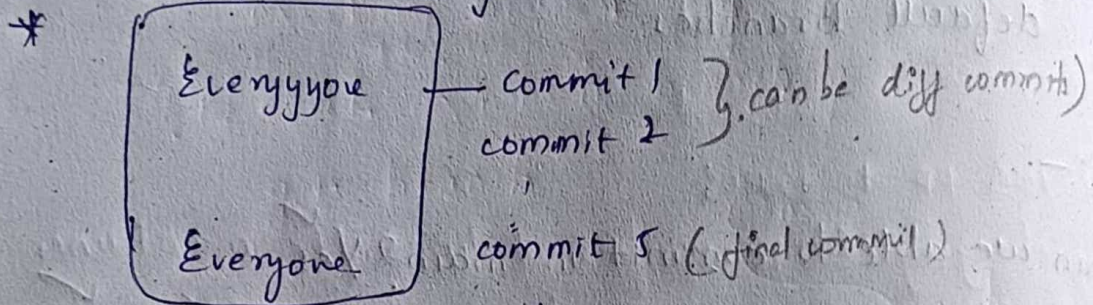
\* Now in this branch, we can as many actions/commits we want.

\* Before working with branch or whatever, first you need to open issue (ticket)

① create issue #1

② create branch, then work

fix/typo-in-heading



All this commits goes into our fix/typo-in-heading branch.

### Step 3:

\* After committing our changes, now we are ready to merge our commits/changes to main branch.

## CREATING A PULL/MERGE REQUEST

fix/typo-in-heading (branch) → main branch

Give proper in PR

title: fix: typo in heading

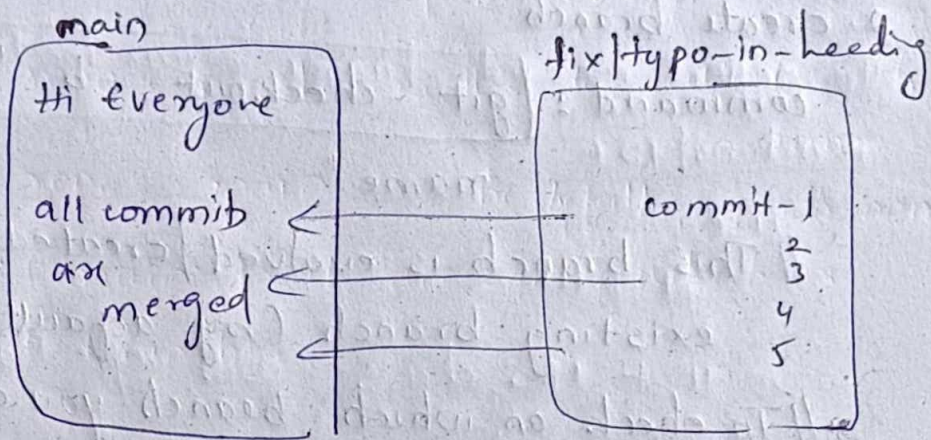
desc: fixes #1

\* Based on our code/changes PR can be closed/merged



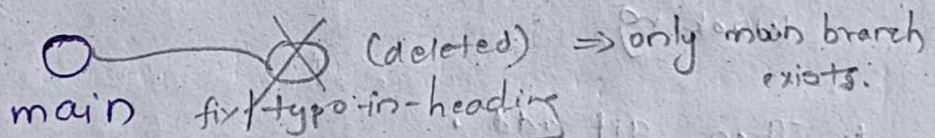
\* If closed our code is not good & many reasons

[ If merged all the commits will merge into main branch



\* changes from fix/typo-in-heading branch to main branch done after raising PR if merged

\* Now we can also delete that branch, if we wish to (bcaz changes are merged)



\* Our code (i.e) src code of main is deployed

change, add, commit -m "changed", push } process when changes from main -> main



#### Step 4: -

As a contributor,

1. create issue.

2. create branch

command: `git checkout -b "new branch name"`

// This branch is evolved/created from the existing branch (by default main)

// To check on which branch you are

`git branch`

○ main

new-branch

Ex: -branch name

fix/typo-in-homepage

3. Now you are on fix/typo-in-homepage, start working.

4. Then `git add`.

`git commit -m "message"`

`git push`.

5. when you try to use push command, you may encounter error like upstream (the main repo url) copy & paste that upstream, & then everything is sorted

Ex: `git push --set-upstream origin fix/typo-in-homepage`

// new branch is created.

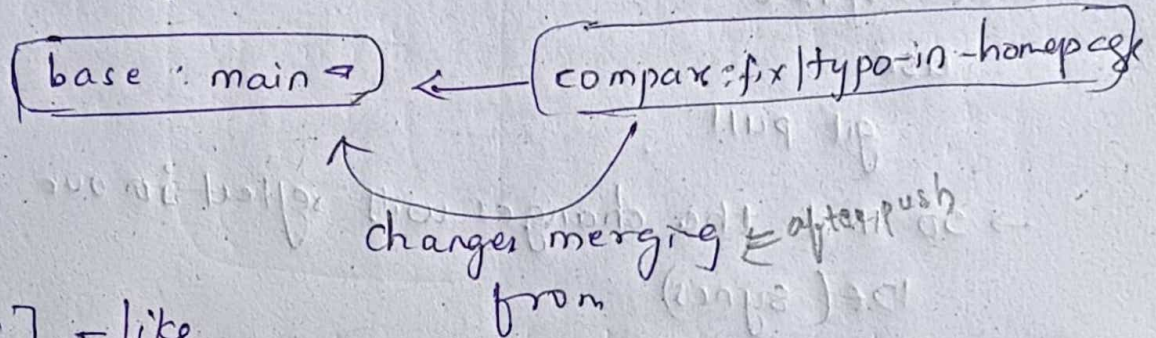
// you can see this branch in repo,



6. Now you will able to see compare and Pull request in your repo tab.

→ You can click on that else

→ click on create a new PR then you will branches option, select it



[x] = like checking ☒ in desc

7. After merging issue is closed

Imp

After merging, you can delete typo branch is github but it locally exist in your IDE/machine

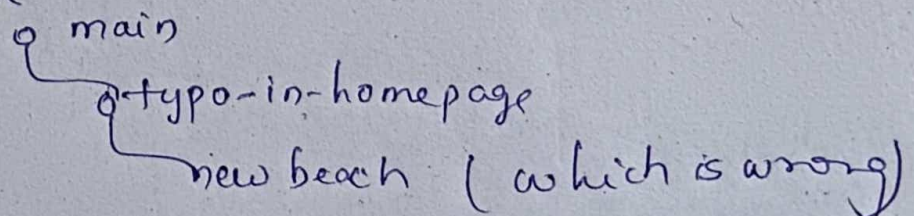
→ Now, before creating another branch for another work

\*\* use `git switch main`

→ This will help you in gng back to main branch & you can create new branches from this.

→ Else without switch to main-branch if you try to checkout another branch,

it will be



→ If you follow this, the commits will be transferred to newbranch also



→ Then you need to git pull (in order to get changes)

Go to main (switch) → Then checkout

To reflect merged changes in our local copy or machine/IDE

use

git pull

→ so that the changes will reflect in our IDE (syncs)

