**GIT Commands**

1. **git add**

**Syntax** : git add <filename> OR git add .

This command is used to add and update all the modified changes to the file to the staging area, which means that it signifies that the change in the working directory is ready to be staged



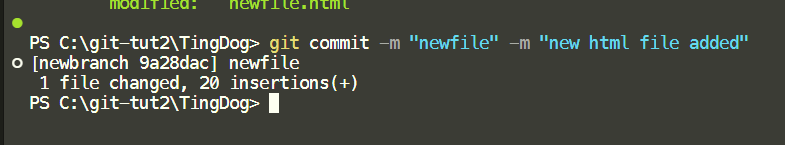
2) **git commit**

This command is used to commit the file changes or stage the changes that have been made to the working directory

**Syntax** : git commit -m “message” -m “message2”

Here the **message1** signifies the Subject of the commit message, regarding a fix or a change

The second commit message **message2** represents a short description of the commit message

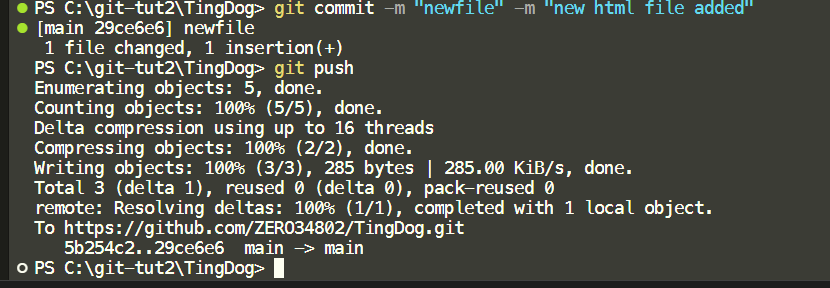


3) **git push**

Git push is used to update or push your local repository changes and content to your remote repository (eg : github)

**Syntax** : git push ( if being done on the main or master branch)

git push <main/origin> <branch\_name> (if being done on a branch)

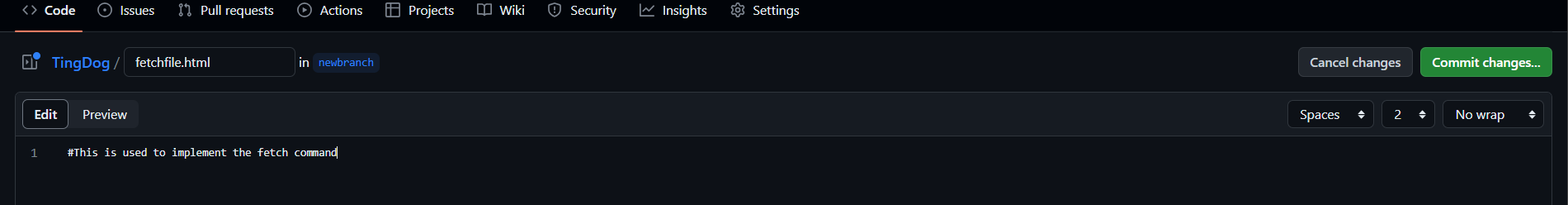


4) **git fetch**

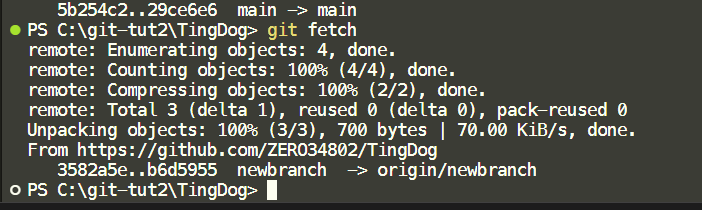
**Syntax** : git fetch

**Description** : This command is used to fetch and update changes made in your remote repository into your local repository without merging them with your working directory

Created a new file in the remote repo



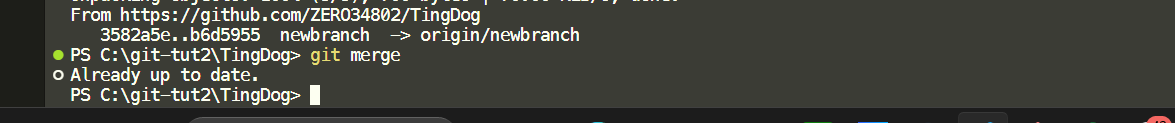
Using fetch to update in the local repo



5) **git merge**

**Syntax :** git merge

**Description** : This command is used to merge the updates bought from the remote repo and placed in the local repo with the working directory.



6) **git pull**

**Syntax** : git pull (if used for diff branch -> git pull <origin> <brname>)

**Description** : used to perform fetch + merge operation together directly.

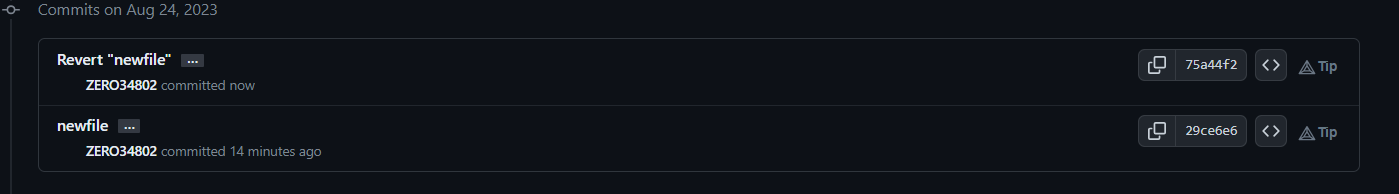
7) **git revert**

**Syntax :** git revert <commit\_hash\_id>

Here the <commit\_hash\_id> refers to the id of the commit that needs to be reverted

**Description** : used to revert either changes that have been committed by mistake or changes that have been committed and pushed by mistake

Commit reverted

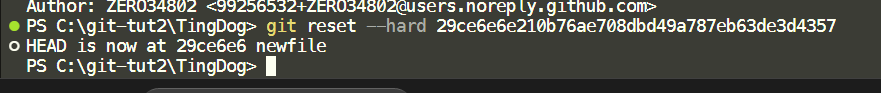


8) **git reset –hard**

**Syntax :** git reset –hard <commit\_id>

<commit\_id> refers to the commit id we wish the state of the head to go back to

**Description** : another git command that is used to hard reset and go back to the state



9) **git log**

**Syntax :** git log

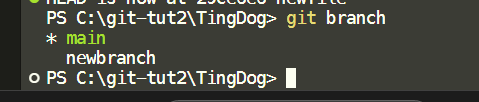
**Description** : This command is used to give a log of all the previous commits and actions along with their reference id



10) **git branch**

**Syntax :** git branch

**Description** : This Command is used to get a list of all the available branches with a pointer pointing to the head of the branch that we are currently on.



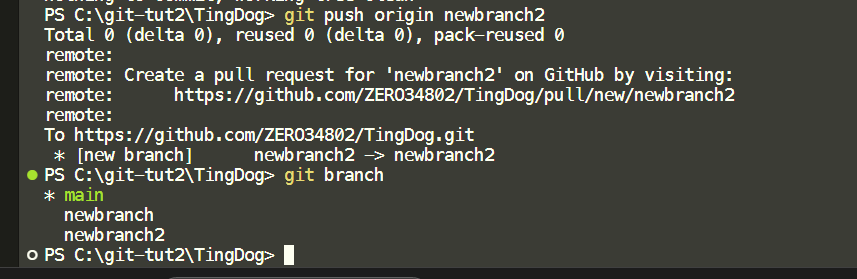
11) **git branch <branch\_name>**

**Syntax :** git branch <branch\_name>

**Description :** used to create a new branch.

**Alternative command** : git checkout -b <branch\_name>

**Note** : It is not just enough to create a new branch for it to be reflected in the remote repository, we have to also push the new branch after adding and committing it to the remote repo.

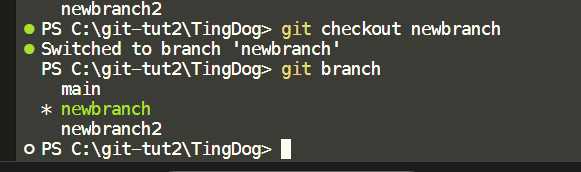


As we can see, the newbranch has been added.

12) **git checkout**

**Syntax :** git checkout

**Description :** This command is used to checkout, or switch the head pointer to another branch and start working on that branch in the remote repo.



As we can see, the head pointer is now pointing at the newbranch.