

GitHub Basics

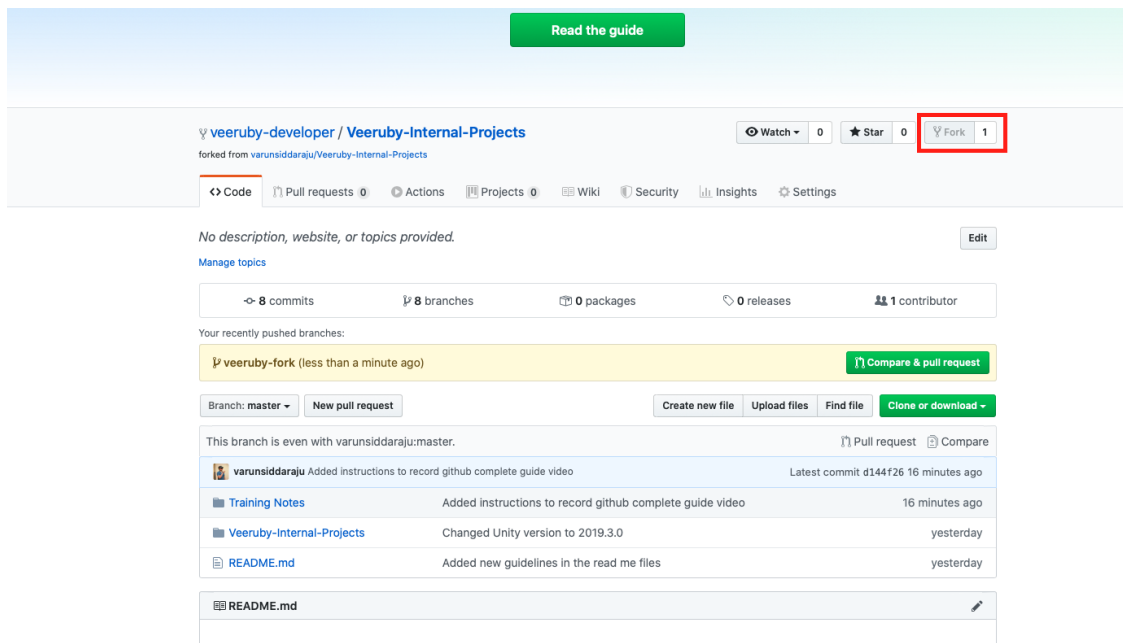
Beginner level

1. What is GitHub?
2. What are the advantages of GitHub?
3. What is commit, push, pull?
4. What is pull request?
5. How to create repository?
6. How to add collaborators?
7. How to merge master to your branch? (Attach screenshots).
8. How to merge your branch into master? (Attach screenshots).
9. How to create pull request? (Attach screenshots).
10. How to create branch? (Attach screenshots).
11. How to delete branch? (Attach screenshots).
12. How to delete repository? (Attach screenshots).

Intermediate level

12. What is fork?

A **fork** is a copy of a repository. **Forking** a repository allows you to freely experiment with changes without affecting the original project.
13. How to create fork? (Attach screenshots).



14. How to merge changes from fork to master project? (Attach screenshots).

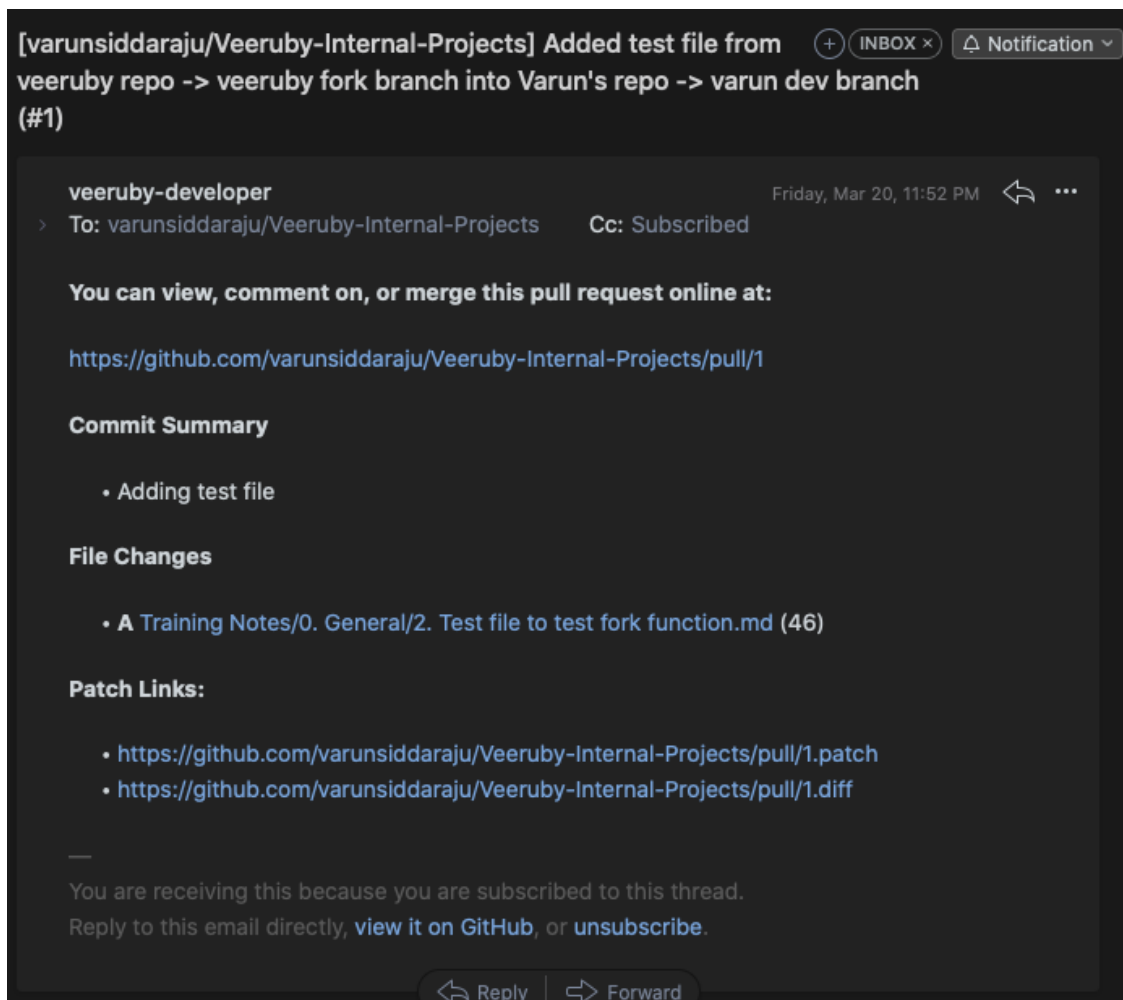
14.1: Create pull request from fork repo.

14.2: Make sure you are merging right fork -> branch in to main repo -> right branch.

14.3: Check whether you are getting merge conflicts are not.

14.4: If there is no merge conflicts then you can proceed and create a pull request.

14.5: Once the pull request is being created the owner of the repo will receive an email to approve or reject the pull request.



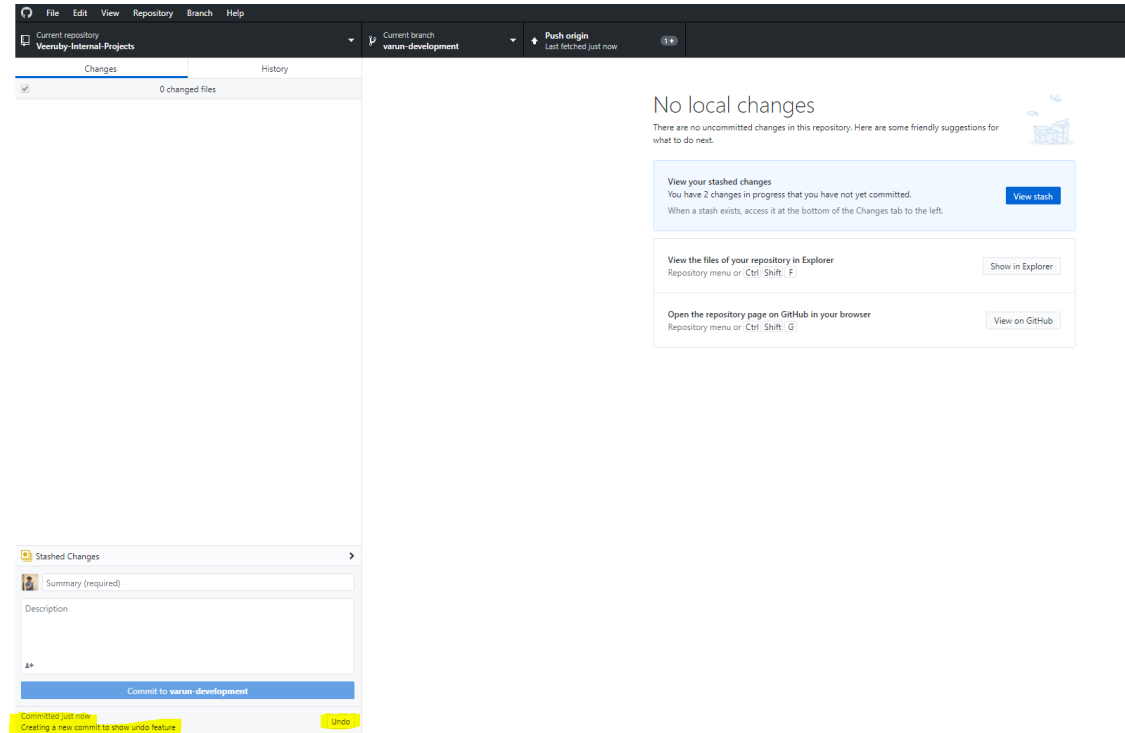
14.6 : Accept the pull request and merge it into main repo.

14.7: After all this steps the merge pull request is success and if you want you can delete the fork branch.

15. How to Undo commit changes? (Attach screenshots).

We can un do any of the commits that we created.

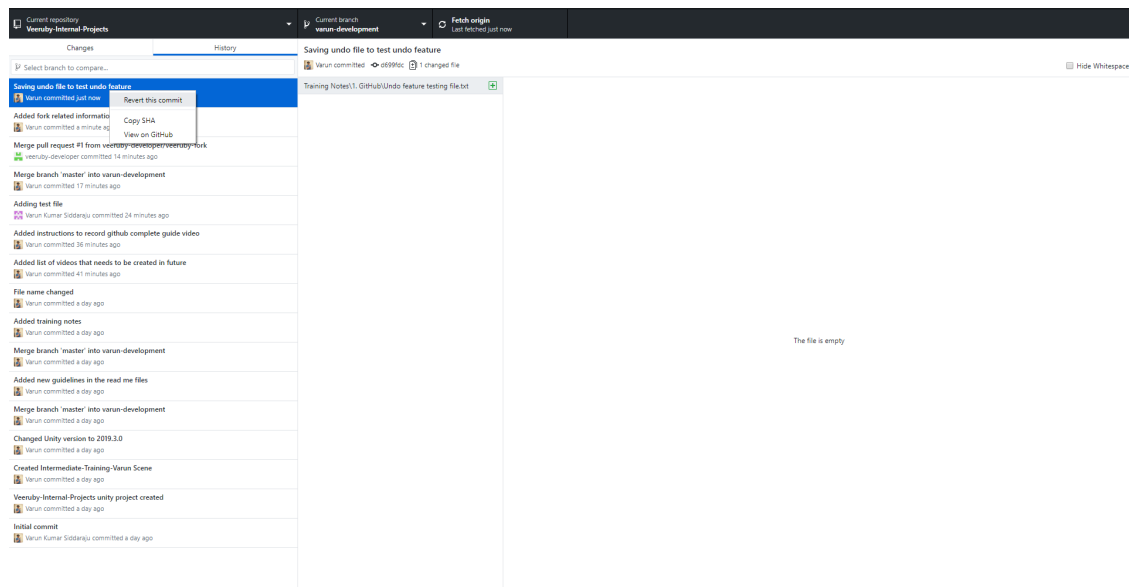
Note: We cannot undo a commit after pushing it to the cloud.

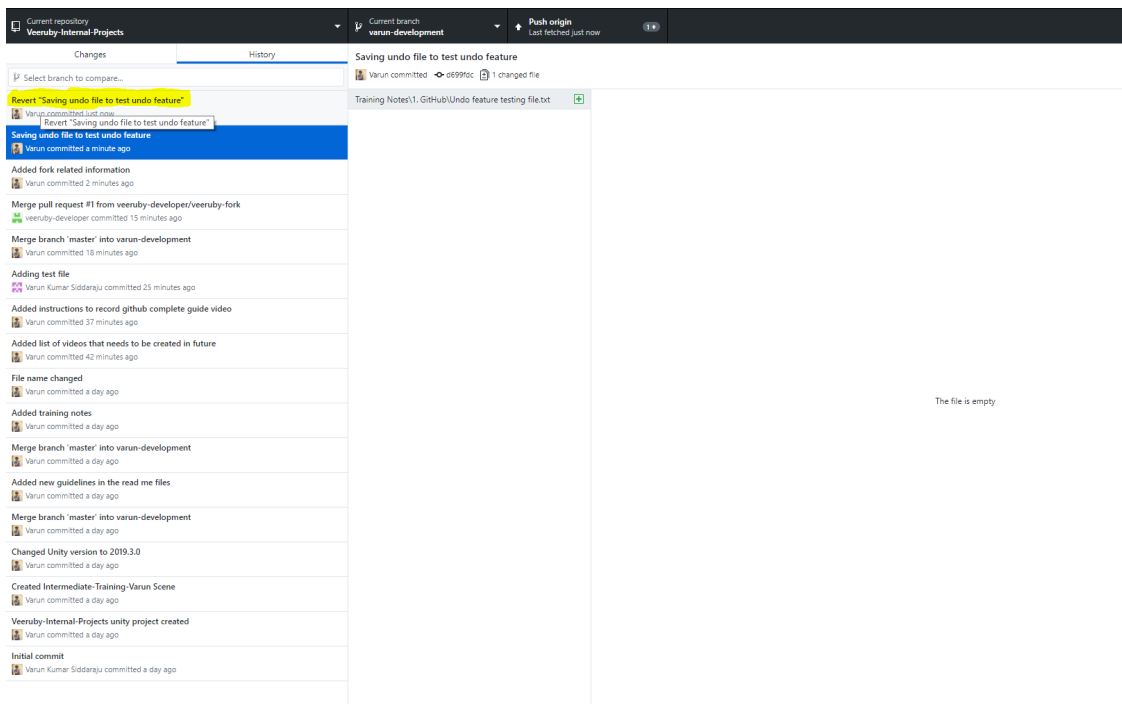


16. What is revert?

The git revert command is a forward-moving undo operation that offers a safe method of undoing changes. Instead of deleting or orphaning commits in the commit history, a revert will create a new commit that inverses the changes specified.

17. How to revert changes? (Attach screenshots).





18. What is stash?

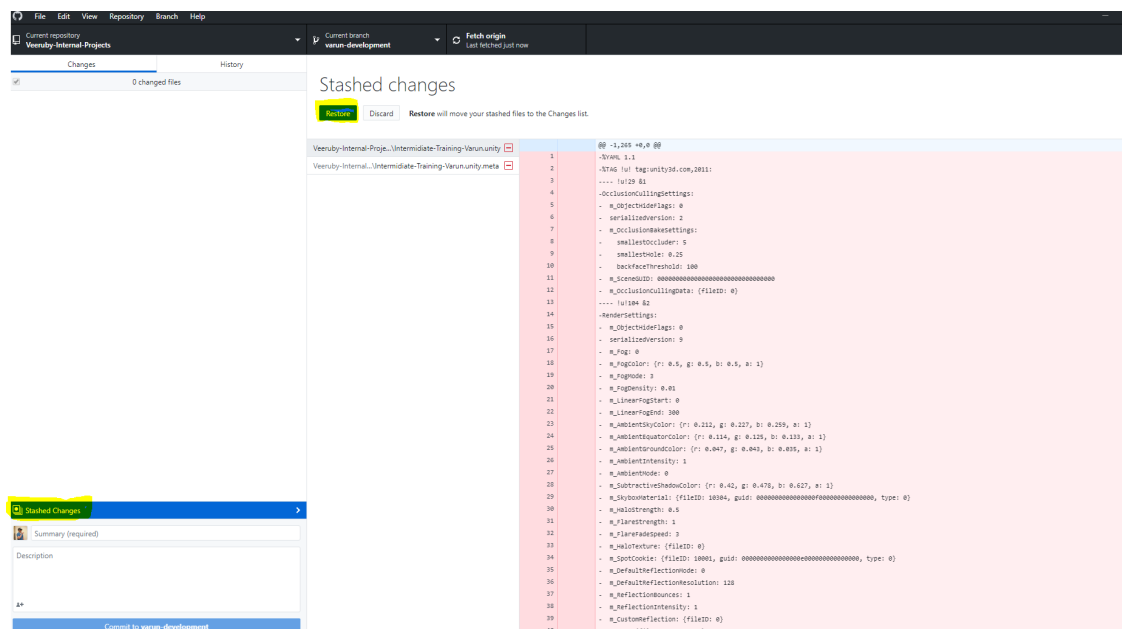
The changes we made in our current branch is called as stash.

These stashes can be moved to other branches, if we don't need to commit in the current branch.

We can keep these stash in the current branch and move to other branches and still can come back and access the stash and make further changes or commit these changes in the initial branch.

19. How to use stash?

When you shift from new branch to previous branch, look for stashed changes and restore it if needed.



Advanced level

20. What is LFS?

21. How to install LFS and add LFS file into it? (Attach screenshots).

22. How to push LFS commits? (Attach screenshots).