

Git-Merging

Last updated by | Shakibe Hasan | Jul 18, 2023 at 6:15 PM GMT+6

Git merging is the process of combining changes from different branches into a single branch, typically merging a feature branch into the main branch. Merging integrates the changes made in one branch with the changes in another branch.

Here's how you can perform a merge in Git:

- Switch to the branch where you want to merge the changes into:
`git checkout main`
- Merge the branch you want to bring changes from:
`git merge feature-branch`

This command combines the changes from the "feature-branch" into the current branch (in this case, "main"). Git will automatically merge the changes whenever possible. If there are conflicts, you will need to resolve them manually.

Resolve merge conflicts (if any):

If Git encounters conflicts during the merge, it means that there are conflicting changes in the files being merged. Git will mark the conflicting sections in the affected files, and you'll need to manually resolve these conflicts by editing the files. After resolving the conflicts, you'll need to add the changes and commit the merge.

Commit the merge:

After resolving conflicts (if any), you need to commit the merge to finalize it:

```
git commit -m "Merge feature-branch into main"
```

Push the merged changes:

Once the merge commit is made, you can push the changes to the remote repository if necessary:

```
git push origin main
```