Merging

What is Merging ?

Definition :

In git, merging is the process of Combining the changes of a branch into different branch.

(Or)

In git, merging will let the developers to take independent lines of development from a branch and integrate the code into a different branch.

Types of Merging :

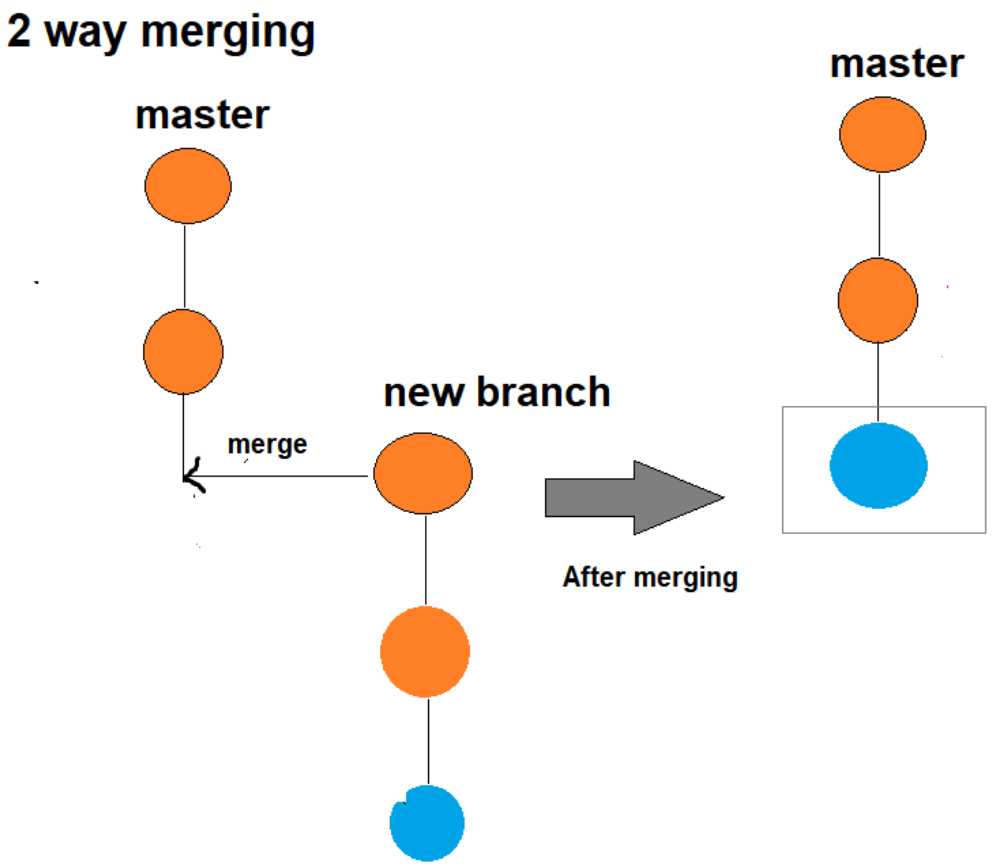
There are two types of merging in git. They are,

1. Two-way merging.
2. Three – way merging.

1. Two-way merging :

Two-way merging takes the list of commits from one branch and merges into another branch and creates a single list of commits of two branches.

- Two -way merge takes place as a Fast-Forward merge in the git terminal.
- The Fast-forward merge will not create any new commit after doing merging.

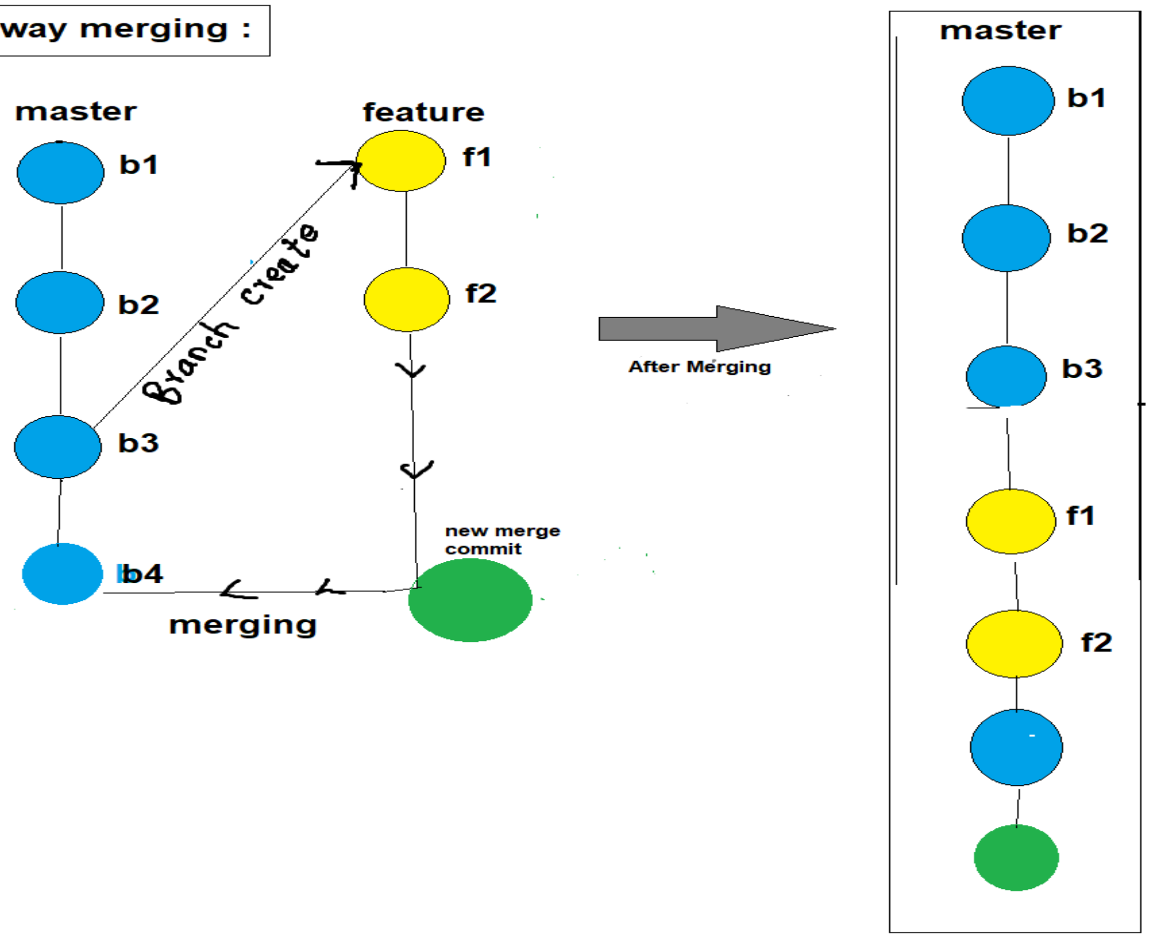


2. Three-way Merging :

Three-way merging takes the list of new commits in both the branches and merge both them recursively by creating new merge commit. The new merge commit contains the changes or new commits which are done in both branches.

- Three-way merging takes place as an ‘ORT’ (Ostensibly Recursive Twin) Strategy in the git terminal.
- ‘ort’ strategy will create a new merge commit while merging.

2. 3 way merging :



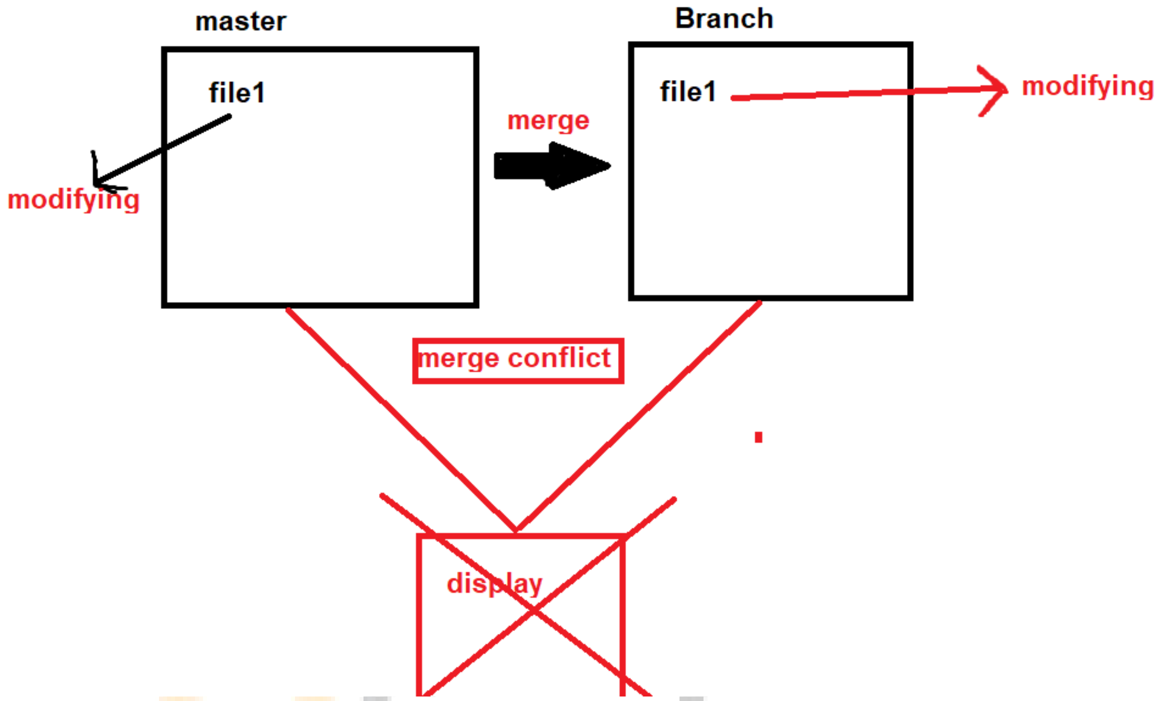
Commands Used for Merging :

- To merge a branch into another branch -----→ **git merge <branch name>.**
- To push the merged branch to remote repo ----→ **git push alias <branch name>.**

Merge Conflict :

If a single file is modified in two branches that to be merged then we will get merge conflict while merging.

If you want to merge both the branches we must solve the conflict and then it will merge the branches automatically.



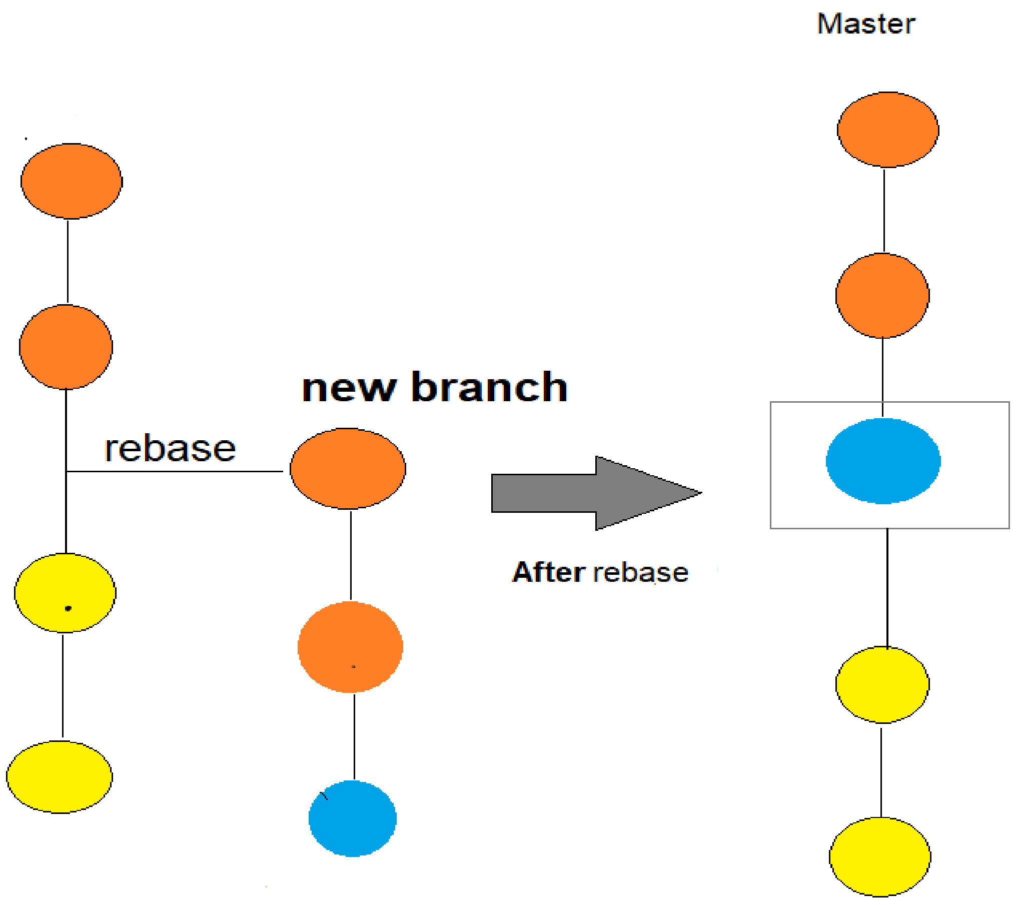
How to solve merge Conflict ?

- ➢ Open the folder that contains your local repository on your windows.
- ➢ Navigate to the file that contains merge conflict.
- ➢ Remove the conflict makers `<<<<<<< HEAD`, `=======` and `>>>>>>> BRANCH-NAME`.
- ➢ Resolve the conflicts and add and commit the file.
- ➢ After committing, The branch will get merged automatically.

Rebase :

Rebase is an alternate command for merging.

- Rebase is used to merge a branch into another branch.
- Rebase will not create a new merge commit after 3 way merging.
- Rebase will maintains a clean history of commits.
- Rebase will create a linear line of commits without adding any new merge commit to the history.



- To reflect a particular commit from one branch to another branch without using a merge command
  - **git cherry-pick commitId**