GIT MERGE:

It allows us to get the code from one branch to another. This is useful when developers work on the same code and want to integrate their changes before pushing them up in a branch.

command: git merge branch_name

GIT MERGE CONFLICTS

Merge conflicts happen when you merge branches that have competing commits, and Git needs your help to decide which changes to incorporate in the final merge.

GIT makes merging super easy!
CONFLICTS generally araise when two people two people have changed the same lines in a file (or) if one developer deleted a file while another developer is working on the same file!
In this situation git cannot determine what is correct!

Lets understand in a simple way!

cat>file1 : hai all
add & commit
git checkout -b branch1
cat>file1 : 1234
add & commit
git checkout master
cat>>file1 : abcd
add & commit

git merge branch1 : remove it

identify merge conflicts:

see the file in master branch then you will see both the data in a single file including branch names

resolve:

open file in VIM EDITOR and delete all the conflict dividers and save it! add git to that file and commit it with the command (git commit -m "merged and resolved the conflict issue in abc.txt")

merge:

if you have 5 commits in master branch and only 1 commit in devops branch, to get all the commits from master branch to devops branch we can use merge in git. (command: git merge branch_name)

cherry-pick:

if you have 5 commits in master branch and only 1 commit in devops branch, to get specific commit from master branch to devops branch we can use cherry pick in git. (git cherry-pick commit_id).

GIT STASH:

Using the git stash command, developers can temporarily save changes made in the working directory. It allows them to quickly switch contexts when they are not quite ready to commit changes. And it allows them to more easily switch between branches.

Generally, the stash's meaning is "store something safely in a hidden place."

- git stash
- git stash apply
- git stash list
- git stash pop
- git stash clear
- git stash drop stash@{2}