Que : What is "stashing" ?

Ans : "Stashing" in the context of version control systems like Git refers to the process of temporarily saving changes that are not ready to be committed or that you want to set aside for a moment. This allows you to switch to a different branch or perform other operations without having to commit incomplete work.

1. \*\*Unfinished Work\*\*: - Let's say you're working on a branch and you have some changes that you haven't finished or aren't ready to commit.

2. \*\*Stash Changes\*\*: - You can use the command `git stash` to save these changes in a "stash". This stash acts as a temporary storage for your changes.

3. \*\*Switch Branches\*\*: - After stashing, you can switch to a different branch or perform other operations without worrying about your unfinished work interfering.

4. \*\*Apply Stash\*\*: - When you're ready to continue working on the original branch, you can use `git stash apply` to reapply the changes from the stash back to your working directory.

5. \*\*Clear Stash\*\*: - Once you've applied the changes, you can use `git stash drop` to remove the stash. If you want to apply and remove the stash in one step, you can use `git stash pop`.

6. \*\*List Stashes\*\*: - You can use `git stash list` to see a list of stashes that you've created. Each stash is assigned a unique identifier.

Stashing is useful in situations where you're in the middle of something but need to switch focus temporarily. It allows you to save your work without having to commit it in an unfinished state.

Remember that stashes are specific to your local repository and are not shared with remote repositories. They're a tool for your local workflow. If you need to collaborate with others, you'll still need to commit and push your changes to the remote repository.

