**Git Interview Q&A**

### 1. What does git add do and why is it important before committing?

* git add moves changes from the working directory to the staging area (index).
* Importance:
  + Lets you decide which changes to include in a commit.
  + Prevents accidental commits of unwanted files.

### 2. How do you stage only specific changes in Git?

* Stage a specific file:
* git add file1.yaml
* Stage multiple specific files:
* git add file1.yaml file2.yaml
* Stage certain lines interactively:
* git add -p file.yaml

### 3. If you have to update 2 YAML files in a remote repo as a new joiner, how will you do it?

Steps:

1. Clone the repository:

* git clone <repo-url>  
  cd <repo-name>

1. Create a new branch:

* git checkout -b update-yaml-files

1. Edit the YAML files locally.
2. Stage only the updated files:

* git add file1.yaml file2.yaml

1. Commit with a meaningful message:

* git commit -m "Updated configuration in file1.yaml and file2.yaml"

1. Push your branch to remote:

* git push origin update-yaml-files

1. Raise a Pull Request (PR) for review.

### 4. What is Git stash & when do you use it?

* git stash temporarily saves uncommitted changes and cleans your working directory.
* Use cases:
  + Switch branches quickly without committing.
  + Pull latest changes without losing local progress.
* Commands:
* git stash # save changes  
  git stash list # view stashes  
  git stash apply # reapply last stash  
  git stash pop # reapply & remove from stash

### 5. Difference between git fetch & git pull

* git fetch: Downloads new commits from remote **without merging**.
* git pull: Fetches + merges changes into your local branch.
* Formula: git pull = git fetch + git merge

### 6. What is git cherry-pick & when would you use it?

* git cherry-pick <commit-hash> applies a specific commit from one branch to another.
* Use cases:
  + Apply a bug fix from a feature branch to main.
  + Copy urgent hotfixes to another branch.

Example:

git checkout main  
git cherry-pick a1b2c3d