Comprehensive Guide to Git Commands

Git Commands and Their Detailed Explanations

1. git init
   - Initializes a new Git repository in the current directory.
   - Example: `git init`
   - Additional Options:
     - `--bare`: Creates a bare repository (no working tree).

2. git clone
   - Copies an existing repository to a local directory.
   - Example: `git clone <repository_url>`
   - Additional Options:
     - `--depth <number>`: Creates a shallow clone with a limited commit history.
     - `--branch <branch_name>`: Clones a specific branch.

3. git add
   - Stages changes for the next commit.
   - Example: `git add <file_name>`
   - Additional Options:
     - `--all` or `-A`: Stages all changes, including deletions.
     - `-n` or `--dry-run`: Shows what would be added without actually adding.

4. git commit
   - Records changes to the repository.

- Example: `git commit -m "message"`
- Additional Options:
  - `--amend`: Modifies the last commit.
  - `--no-edit`: Amends without opening an editor.
  - `-a`: Commits all changes directly (skipping the staging area).

5. git status
- Displays the status of the working directory and staging area.
- Example: `git status`
- Additional Options:
  - `--short`: Provides a compact output.
  - `--ignored`: Shows ignored files.

6. git log
- Displays commit history.
- Example: `git log`
- Additional Options:
  - `--oneline`: Shows a summary in one line per commit.
  - `--graph`: Visualizes branches and merges.
  - `--stat`: Shows file changes.

7. git branch
- Manages branches.
- Example: `git branch`
- Additional Options:
  - `-d <branch_name>`: Deletes a branch.
  - `-m <new_name>`: Renames the current branch.

8. git checkout

- Switches branches or restores files.
- Example: `git checkout <branch_name>`
- Additional Options:
  - `-b <branch_name>`: Creates and switches to a new branch.
  - `-- <file>`: Restores a specific file.

9. git merge

- Merges changes from one branch to another.
- Example: `git merge <branch_name>`
- Additional Options:
  - `--no-ff`: Creates a merge commit even for a fast-forward merge.
  - `--squash`: Combines commits into a single commit.

10. git rebase

- Reapplies commits on top of another base tip.
- Example: `git rebase <branch_name>`
- Additional Options:
  - `--interactive` or `-i`: Opens an interactive interface for reordering commits.

11. git remote

- Manages remote repositories.
- Example: `git remote add <name> <url>`
- Additional Options:
  - `-v`: Lists remote repositories with URLs.
  - `remove <name>`: Removes a remote repository.

12. git fetch
    - Fetches changes from a remote repository.
    - Example: `git fetch <remote_name>`
    - Additional Options:
      - `--all`: Fetches all remotes.
      - `--depth <number>`: Limits the commit history.

13. git pull
    - Fetches and integrates changes.
    - Example: `git pull <remote_name> <branch_name>`
    - Additional Options:
      - `--rebase`: Applies changes without a merge commit.

14. git push
    - Uploads local changes to a remote repository.
    - Example: `git push <remote_name> <branch_name>`
    - Additional Options:
      - `--force` or `-f`: Overwrites remote changes (use with caution).
      - `--set-upstream`: Sets the default remote branch.

15. git reset
    - Undoes changes in the working directory or staging area.
    - Example: `git reset <commit>`
    - Additional Options:
      - `--soft`: Moves HEAD, keeps changes in the staging area.
      - `--hard`: Discards all changes.

16. git stash
    - Temporarily saves changes.
    - Example: `git stash`
    - Additional Options:
      - `--include-untracked`: Stashes untracked files.
      - `pop`: Applies and removes the stash.

17. git diff
    - Shows changes between commits, branches, or the working directory.
    - Example: `git diff <branch_name>`
    - Additional Options:
      - `--cached`: Shows staged changes.
      - `--stat`: Displays summary statistics.

18. git tag
    - Creates tags for commits.
    - Example: `git tag <tag_name>`
    - Additional Options:
      - `--annotate` or `-a`: Adds an annotated tag.
      - `--delete`: Removes a tag.

19. git archive
    - Creates a tar or zip file of the repository.
    - Example: `git archive --format=zip HEAD`
    - Additional Options:
      - `--prefix`: Adds a prefix to file paths.

20. git cherry-pick

- Applies a specific commit to the current branch.
- Example: `git cherry-pick <commit_hash>`
- Additional Options:
  - `--no-commit`: Applies without committing.

This document covers core Git commands and extensions. Feel free to refer to the official Git documentation for additional details and advanced usage.