Certainly! Here's a list of commonly used Git commands along with their descriptions:

1. **git init** - Initializes a new Git repository in the current directory.

2. **git clone \<repository_url>** - Clones a remote repository to your local machine.

3. **git add \<file_name>** - Stages a file for commit.

4. **git add .** - Stages all modified and new files for commit.

5. **git commit -m "Commit message"** - Commits the staged changes with a descriptive message.

6. **git status** - Shows the current status of your repository, including modified, staged, and untracked files.

7. **git log** - Displays a log of all commits, showing commit messages, authors, dates, and commit hashes.

8. **git branch** - Lists all branches in the repository, highlighting the currently active branch.

9. **git branch \<branch_name>** - Creates a new branch with the given name.

10. **git checkout \<branch_name>** - Switches to the specified branch.

11. **git merge \<branch_name>** - Merges the changes from the specified branch into the current branch.

12. **git pull** - Fetches changes from a remote repository and merges them into the current branch.

13. **git push** - Pushes committed changes to a remote repository.

14. **git remote add \<remote_name> \<remote_url>** - Adds a remote repository with a specified name and URL.

15. **git remote -v** - Lists all remote repositories and their URLs.

16. **git diff** - Shows the differences between the working directory and the last commit.

17. **git reset \<file_name>** - Unstages a file, removing it from the staging area.

18. **git reset --hard HEAD** - Discards all changes in the working directory and resets to the last commit.

19. **git stash** - Temporarily saves changes that are not ready to be committed.

20. **git stash pop** - Applies the changes previously stashed using `git stash`.

21. **git config --global user.name "Your Name"** - Sets your global username.

22. **git config --global user.email "youremail@example.com"** - Sets your global email.

23. **git checkout -b \<new_branch_name>** - Creates and switches to a new branch.

24. **git remote remove \<remote_name>** - Removes a remote repository from the list of remotes.

25. **git push -u \<remote_name> \<branch_name>** - Pushes the current branch to a remote repository, setting it as the default upstream branch.

26. **git pull origin \<branch_name>** - Pulls changes from a specific branch of the remote repository.

27. **git fetch** - Fetches changes from the remote repository, updating remote-tracking branches.

28. **git rebase \<branch_name>** - Reapplies the commits on the current branch after moving the base to the specified branch.

29. **git tag -a \<tag_name> -m "Tag message"** - Creates an annotated tag for a specific commit.

30. **git rm \<file_name>** - Removes a file from the working directory and stages the deletion.

These are just a subset of Git commands. Git has many more commands and options for various scenarios. Make sure to refer to the official Git documentation or use `git help` followed by a command to get more information about each command.