1. git init: Initialize a new Git repository.
2. git clone [url]: Clone a repository into a new directory.
3. git add [file]: Add a file or directory to the staging area.
4. git commit -m "[message]": Commit changes with a descriptive message.
5. git status: Show the status of working directory and staging area.
6. git diff: Show the difference between the working directory and the index.
7. git diff --staged: Show the difference between the staged changes and the last commit.
8. git branch: List all local branches.
9. git branch [branch-name]: Create a new branch.
10. git checkout [branch-name]: Switch to a different branch.
11. git checkout -b [branch-name]: Create and switch to a new branch.
12. git merge [branch-name]: Merge changes from one branch into the current branch.
13. git remote: List all remote repositories.
14. git remote add [name] [url]: Add a new remote repository.
15. git push [remote] [branch]: Push local changes to a remote repository.
16. git pull [remote] [branch]: Fetch changes from a remote repository and merge them into the current branch.
17. git fetch [remote]: Fetch changes from a remote repository without merging.
18. git log: Show commit logs.
19. git log --oneline: Show compact commit history.
20. git log --graph: Show commit history with ASCII graph.
21. git reset [file]: Unstage a file.
22. git reset --hard: Reset the index and working directory to the last committed state.
23. git reset --hard HEAD^: Reset to the previous commit.
24. git reset --hard HEAD~2: Reset to two commits ago.
25. git revert [commit]: Revert a commit.
26. git rm [file]: Remove a file from the repository.
27. git mv [old-path] [new-path]: Move or rename a file.
28. git tag [tag-name]: Create a new tag at the current commit.
29. git tag: List all tags.
30. git show [commit]: Show information about a commit.
31. git stash: Stash changes in a dirty working directory.
32. git stash apply: Apply the most recent stash.
33. git stash list: List all stashes.
34. git stash drop: Remove the most recent stash.
35. git stash pop: Apply and drop the most recent stash.
36. git cherry-pick [commit]: Apply the changes introduced by a specific commit.
37. git clean: Remove untracked files from the working directory.
38. git config --global user.name "[name]": Set your username.
39. git config --global user.email "[email]": Set your email address.
40. git config --global color.ui auto: Enable colored output.
41. git remote -v: Show remote URLs after adding a remote.
42. git blame [file]: Show who last modified each line of a file.
43. git bisect: Use binary search to find the commit that introduced a bug.
44. git grep [pattern]: Search the working directory for a pattern.
45. git archive [branch-name]: Create a zip or tarball archive of a branch.
46. git rebase [branch]: Reapply commits on top of another base tip.
47. git submodule add [url]: Add a submodule repository.
48. git submodule update: Update submodules to the latest commit.
49. git help: Get help on any Git command.
50. git version: Display the installed Git version.

These commands cover a wide range of Git functionalities, from basic repository management to more advanced operations like branching, merging, and collaborating with remote repositories.