**Git Remote Commands**

| **Command** | **Description** |
| --- | --- |
| git remote | Lists all remote repositories linked to the local repo. |
| git remote -v | Shows URLs for all remotes (for fetch and push). |
| git remote add <name> <url> | Adds a new remote repository. |
| git remote remove <name> | Removes a remote repository. |
| git remote rename <old\_name> <new\_name> | Renames a remote repository. |
| git remote set-url <name> <new\_url> | Changes the URL of a remote repository. |
| git remote show <name> | Shows detailed information about a remote. |
| git fetch <remote\_name> | Fetches changes from a remote repository without merging them. |
| git pull <remote\_name> <branch\_name> | Fetches and merges changes from a remote repository branch. |
| git push <remote\_name> <branch\_name> | Pushes local changes to a remote repository branch. |
| git remote prune <remote\_name> | Removes references to branches deleted on the remote repository. |

**Git Config Commands**

| **Command** | **Description** | **Example** |
| --- | --- | --- |
| git config --global user.name "Your Name" | Sets the global user name for all Git commits. | git config --global user.name "John Doe" |
| git config --global user.email "you@example.com" | Sets the global email for all Git commits. | git config --global user.email "johndoe@example.com" |
| git config user.name "Your Name" | Sets the local user name for the current repository. | git config user.name "John Doe" |
| git config user.email "you@example.com" | Sets the local email for the current repository. | git config user.email "johndoe@example.com" |
| git config --global --list | Displays all global configuration settings. | git config --global --list |
| git config --list | Displays all configuration settings (local or global). | git config --list |
| git config --global core.editor "editor\_name" | Sets the default editor for commit messages (e.g., vim, nano). | git config --global core.editor "vim" |
| git config --global alias.<alias> <command> | Creates a shortcut for a frequently used command. | git config --global alias.co checkout |
| git config --global core.autocrlf true | Manages line endings for Windows environments. | git config --global core.autocrlf true |
| git config --global color.ui true | Enables colored output for Git commands. | git config --global color.ui true |
| git config --global init.defaultBranch <name> | Sets the default branch name when initializing new repositories. | git config --global init.defaultBranch main |
| git config --global merge.tool <tool> | Defines the tool for resolving merge conflicts. | git config --global merge.tool vimdiff |
| git config --global credential.helper cache | Caches credentials to avoid repeated prompts. | git config --global credential.helper cache |
| git config --global push.default <behavior> | Sets default behavior for pushing branches (e.g., current). | git config --global push.default current |

**Git branch Commands**

| **Command** | **Description** | **Example** |
| --- | --- | --- |
| git branch | Lists all local branches in the repository. | git branch |
| git branch -a | Lists all local and remote branches. | git branch -a |
| git branch <branch\_name> | Creates a new branch with the specified name. | git branch feature-branch |
| git branch -d <branch\_name> | Deletes the specified branch (if it's already merged). | git branch -d feature-branch |
| git branch -D <branch\_name> | Forcefully deletes the branch, even if it's not merged. | git branch -D feature-branch |
| git checkout -b <branch\_name> | Creates a new branch and immediately switches to it. | git checkout -b new-feature |
| git checkout <branch\_name> | Switches to the specified branch. | git checkout main |
| git branch -m <new\_branch\_name> | Renames the current branch to the specified new name. | git branch -m new-branch-name |
| git branch -m <old\_name> <new\_name> | Renames the specified branch to a new name. | git branch -m old-branch new-name |
| git branch --move <old\_name> <new\_name> | Same as above (another way to rename a branch). | git branch --move old new |
| git branch --set-upstream-to=<remote>/<branch> | Sets the upstream branch for tracking purposes. | git branch --set-upstream-to=origin/main |
| git branch -v | Shows the last commit for each branch. | git branch -v |
| git branch --contains <commit\_hash> | Lists branches that contain the specified commit. | git branch --contains a1b2c3d |
| git branch -r | Lists only the remote branches. | git branch -r |
| git branch --merged | Lists branches that have been merged into the current branch. | git branch --merged |
| git branch --no-merged | Lists branches that have not been merged into the current branch. | git branch --no-merged |

**Git add commands**

| Command | Description | Example |
| --- | --- | --- |
| git add <file> | Adds a specific file to the staging area. | git add file1.txt |
| git add . | Adds all changed files (tracked and untracked) in the current directory and subdirectories to the staging area. | git add . |
| git add -A | Adds all changes (across the repository) to the staging area, including modifications, deletions, and untracked files. | git add -A |
| git add -p | Interactively adds portions of a file (hunk by hunk) to the staging area. | git add -p |
| git add -u | Stages changes to tracked files only, excluding untracked files. | git add -u |
| git add --all | Adds all changes (tracked and untracked) in the entire repository (similar to git add -A). | git add --all |
| git add <directory>/ | Adds all files from a specific directory to the staging area. | git add src/ |
| git add --patch | Opens an interactive prompt to add portions of files (same as git add -p). | git add --patch |
| git add --ignore-errors | Continues adding files even if some errors are encountered, ignoring those errors. | git add . --ignore-errors |
| git add -n | Performs a dry run, showing which files would be added to the staging area without actually adding them. | git add -n |

**Git PUSH Commands**

| **Command** | **Description** | **Example** |
| --- | --- | --- |
| git push <remote> <branch> | Pushes the specified branch to the specified remote repository. | git push origin main |
| git push | Pushes the current branch to its configured upstream remote branch (often used after git pull). | git push |
| git push -u <remote> <branch> | Pushes the specified branch and sets upstream tracking for future pushes and pulls. | git push -u origin main |
| git push --all <remote> | Pushes all local branches to the specified remote repository. | git push --all origin |
| git push --force or git push -f | Force pushes the current branch, overwriting the remote branch if it diverges from the local branch. | git push --force origin main |
| git push --tags | Pushes all tags to the remote repository. | git push --tags |
| git push <remote> --delete <branch> | Deletes the specified branch from the remote repository. | git push origin --delete feature-branch |
| git push --dry-run | Shows what would be pushed to the remote without actually pushing. | git push --dry-run origin main |
| git push <remote> <local\_branch>:<remote\_branch> | Pushes a local branch to a branch with a different name on the remote repository. | git push origin feature:feature-remote |
| git push --set-upstream <remote> <branch> | Sets the upstream branch for tracking and then pushes the current branch to it (similar to -u). | git push --set-upstream origin feature |

**GIT PULL COMMANDS**

| Command | Description | Example |
| --- | --- | --- |
| git pull | Fetches and merges changes from the remote repository to the current local branch. | git pull |
| git pull <remote> | Fetches and merges changes from the specified remote repository to the current branch. | git pull origin |
| git pull <remote> <branch> | Fetches and merges changes from the specified branch of the remote repository to the current branch. | git pull origin main |
| git pull --rebase | Fetches and rebases the local commits on top of the upstream changes instead of merging. | git pull --rebase origin main |
| git pull --ff-only | Only allows fast-forward merges (no merge commits). | git pull --ff-only origin main |
| git pull --no-commit | Pulls changes but does not commit them automatically after the merge. | git pull --no-commit origin main |
| git pull --no-rebase | Ensures that no rebase is performed during the pull (forces a merge strategy). | git pull --no-rebase origin main |
| git pull --verbose | Displays additional information during the pull process. | git pull --verbose origin main |
| git pull --quiet | Suppresses output messages during the pull process. | git pull --quiet origin main |
| git pull --depth <depth> | Pulls a limited number of commits from the history (useful for large repositories). | git pull --depth 1 origin main |

**GIT INIT commands**

| **Command** | **Description** | **Example** |
| --- | --- | --- |
| git init | Initializes a new Git repository in the current directory (creates a .git folder). | git init |
| git init <directory> | Initializes a new Git repository in the specified directory (creates a .git folder in that directory). | git init project-directory |
| git init --bare | Initializes a bare Git repository (used for shared repositories, without working directories). | git init --bare |
| git init --quiet | Suppresses output during repository initialization. | git init --quiet |
| git init --separate-git-dir <path> | Initializes a repository, but stores the .git directory in a different location. | git init --separate-git-dir /path/to/gitdir |
| git init --template=<template> | Initializes a repository using specified template directory, customizing hooks or config. | git init --template=/path/to/template |

**UPSTREAM COMMANDS**

| **Command** | **Description** | **Example** |
| --- | --- | --- |
| git push --set-upstream <remote> <branch> | Sets the upstream (tracking) branch for the current local branch. This allows for simplified future pulls and pushes. | git push --set-upstream origin main |
| git branch --set-upstream-to=<remote>/<branch> | Sets the upstream branch for an existing local branch to track a specific remote branch. | git branch --set-upstream-to=origin/main |
| git pull | Pulls changes from the upstream (tracked) branch for the current branch. | git pull |
| git fetch <remote> | Fetches updates from the upstream remote repository without merging them into your local branch. | git fetch origin |
| git remote show <remote> | Displays detailed information about the upstream branch, including its tracking configuration. | git remote show origin |
| git push -u <remote> <branch> | Pushes the branch and sets the upstream tracking (same as --set-upstream). | git push -u origin feature-branch |
| git status | Shows the current status of your repository, including whether the current branch is tracking an upstream branch. | git status |

**GIT SHORTCUTS**

| **Shortcut Command** | **Full Command** | **Description** |
| --- | --- | --- |
| git st | git status | Shows the current status of your repository. |
| git co | git checkout <branch> | Switches to a different branch. |
| git br | git branch | Lists all the branches in the repository. |
| git ci | git commit | Commits the changes in the staging area. |
| git ci -m "<message>" | git commit -m "<message>" | Commits with a message in a single command. |
| git lg | git log --oneline --graph --all --decorate | Displays a simplified Git history with branch and merge information. |
| git unstage | git reset HEAD <file> | Removes a file from the staging area (undoes git add). |
| git uncommit | git reset --soft HEAD^ | Uncommits the last commit but keeps the changes in the working directory. |
| git amend | git commit --amend | Edits the last commit (useful for fixing or updating commit messages). |
| git rb | git rebase | Rebase the current branch on top of another branch. |
| git pu | git pull | Fetches and merges changes from the remote repository. |
| git pl | git pull --rebase | Fetches and rebase the current branch with the remote. |
| git pom | git push origin main | Pushes the changes from the current branch to the main branch on the remote repository. |
| git po | git push origin | Pushes the current branch to the origin remote repository. |
| git df | git diff | Shows the differences between the working directory and the index or between two commits. |
| git last | git log -1 HEAD | Shows the most recent commit. |
| git alias | git config --global alias.<shortcut> '<full\_command>' | Command to create a new Git alias (shortcut). Example: git config --global alias.st 'status'. |