

Git Commands

1 hour

Basics

| Task | Command |
|--|---------|
| Create a git repository | <hr/> |
| View the status of each file in the repository | <hr/> |
| Stage a file for the next commit | <hr/> |
| Commit the staged files with a descriptive message | <hr/> |
| View repository commit history | <hr/> |
| Define the author name to be used in all repositories | <hr/> |
| Define the author email to be used in all repositories | <hr/> |

Undoing Changes

| Task | Command |
|--|---------|
| View a previous commit | <hr/> |
| Create an annotated tag pointing to the most recent commit | <hr/> |
| Undo the specified commit by applying a new commit | <hr/> |
| Reset tracked files to match the most recent commit | <hr/> |
| Remove untracked files | <hr/> |
| Permanently undo uncommitted changes | <hr/> |

Branching Part I

| Task | Command |
|---|---------|
| List all branches | <hr/> |
| Create a new branch using the current working directory as its base | <hr/> |
| Make the working directory and the HEAD match the specified branch | <hr/> |
| Merge a branch into the checked-out branch | <hr/> |
| Delete a branch | <hr/> |
| Remove a file from the working directory and stop tracking the file | <hr/> |

Branching Part II

| Task | Command |
|---|---------|
| Stage all tracked files and commit the snapshot using the specified message | _____ |
| Force the removal of an unmerged branch | _____ |

Rebasing

| Task | Command |
|--|---------|
| Move the current branch's commits to the tip of new-base, which can be either a branch name or commit ID | _____ |
| Add staged changes to the most recent commit instead of creating a new one | _____ |
| Continue a rebase after amending a commit | _____ |
| Abandon the current interactive rebase and return the repository to its former state | _____ |
| Force a merge commit even if Git could do a fast-forward merge | _____ |

Rewrite History

| Task | Command |
|--|---------|
| Display the local chronological history of a repository | <hr/> |
| Move the HEAD backwards n commits but don't change the working directory | <hr/> |
| Move the HEAD backwards n commits and change the working directory | <hr/> |
| Display the commits reachable from until but not from since (params either commit IDs or branch names) | <hr/> |
| Include extra information about altered files in the log output | <hr/> |

Remotes

| Task | Command |
|---|---------|
| Create a clone of a remote Git repository | _____ |
| List remote repositories | _____ |
| Add a remote repository | _____ |
| Download remote branch information, but do not merge anything | _____ |
| Merge a remote branch into the checked-out branch | _____ |
| List remote branches | _____ |
| Push a local branch to another repository | _____ |
| Push a tag to another repository | _____ |

Patches

| Task | Command |
|---|---------|
| Create a patch for each commit contained in the current branch but not in branch-name | <hr/> |
| Apply a patch to the current branch | <hr/> |

Tips and Tricks

| Task | Command |
|--|---------|
| Stash changes to make a clean working directory | <hr/> |
| Re-apply stashed changes to the working directory | <hr/> |
| View the difference between two commits | <hr/> |
| Difference between the working directory and the staging area | <hr/> |
| View the difference between the staging area of the most recent commit | <hr/> |
| Unstage a file, but don't alter the working directory or move the current branch | <hr/> |
| Create a shortcut for a command and store it in the global configuration file | <hr/> |