Git commands Cheat Sheet

Basics and Initialization

Initialize a new git repository: *git init*

Set configuration values for your username and email:
git config --global user.name <your-name>
git config --global user.email <your-email>

Clone a repository:
git clone <repository-url>

Add a file to the staging area:
git add <file>

Add all files changes to the staging area: *git add .*

Check the unstaged changes: *git diff*

Staging and Commits

Commit the staged changes:
`git commit -m “Message”`

Reset staging area to the last commit:
`git reset`

Check the state of the working directory and the staging area:
`git status`

Remove a file from the index and working directory:
`git rm <file>`

List the commit history:
git log

Check the metadata and content changes of the commit:
`git show <commit-hash>`

Branch Management

Lists all local branches:
`git branch`

Create a new branch:
`git branch <branch-name>`

Rename the current branch:
`git branch -m <new-branch-name>`

Delete a branch:
`git branch -d <branch-name>`

Switch to another branch:
`git checkout <branch-name>`

Merge specified branch into the current branch:
`git merge <branch-name>`

Remote and Cleanup

Create a new connection to a remote repository:
`git remote add <name> <repository-url>`

Push the committed changes to a remote repository:
`git push <remote> <branch>`

Download the content from a remote repository:
`git pull <remote>`

Cleanup unnecessary files and optimize the local repository:
`git gc`

Temporarily remove uncommitted changes and save them for later use:
`git stash`

Reapply previously stashed changes
`git stash apply`

Get this cheat sheet as a PDF at:
markodenic.com/git-cheat-sheet