devCamp (/)

Git Cheat Sheet for Developers

This guide provides common Git commands, their use cases, along with a few workflow recommendations such as how to resolve merge conflicts.

By Jordan Hudgens (/public-profile/jordan)

Basic commands

`git init` **Creates a new git repository in the directory**

`git add <file name>` **Adds a specific file to staging**

`git add .` or `git add —A` **Adds the full directory and its contents to staging**

`git commit —m 'Commit message here'` **Commits the file changes in staging and provides a description for the commit**

`git diff` *Analyze the differences between old files and ones that have been added to staging *

Branching

`git branch` **View all local branches**

`git branch -b <branch name>` **Create a new branch and check into it**

`git branch -av` **View all branches local and remote**

`git checkout <branch name>` **Switch into the specified branch**

`git branch -d <branch name>` **Delete a specific branch**

`git push origin <branch name>` **Push a branch to a remote repository**

`git merge <branch name>` **Merge branch into master branch (ensure that you're inside the master branch)**

Stashing

`git stash` **Stashes changes in the .git file for temporaily hiding changed elements (make sure to run `git add` prior to stashing**

`git stash apply` **Returns the stashed items**

View History

`git log` **View previous commits, their messages, and ids**

`git log <file name>` **View who changed a specific file**

`git blame <file name>` **View who changed a specific file**

Reverting

`git reset --hard HEAD` **Remove all changes and revert to the previous commit**

`git checkout <file name>` **Remove changes made to a specific file, reverting it back to the previous commit**

Merge Conflicts

When you encounter a merge conflict, first... don't freak out. This just means you're being given the ability to choose which code you want to choose to go with.

Sample Conflict

- You run `git pull` on a repo that has changes that conflict with your local copy.
- You're presented with the following error message:

```
error: The following untracked working tree files would be overwritten by merge:
README.md
Please move or remove them before you merge.
Aborting
```

Natural Response

How to Fix a Merge Conflict

> It's actually not that hard to fix merge conflicts, simply run `git status` and it will show you where the conflict exists. An example conflict might look like this:

```
<<<<<<< HEAD
# Git Homepage
=======
# git-guide

> A guide for working with git
>>>>>>> a1dfac148fb86311a453a9de338e4d7b9389ade5
```

The `<<<<<<< HEAD`, `=======`, and `>>>>>>>` symbols are simply showing you where the conflict is. You have the choice to pick what you want to keep, what you want to remove, etc. After picking out the implementation that you want you can run:

- `git add .`
- `git commit -m 'some commit message'`
- `git push`

And you will be good to go.

Cloning

> `git clone <link to repo>` **Clone a repository from a remote source, such as GitHub**

Rebase

> `git rebase master` **While working on a branch you can bring in committed changes from another branch. This is helpful for ensuring that your feature branch can be cleanly merged with the master branch**

Jordan Hudgens (/public-profile/jordan)

I've been a software engineer for the past decade and have traveled the world building applications and training individuals on a wide variety of topics.

View All Posts (/blog)

Recent posts

How to Integrate a Helper Module in Ruby (/site_blogs/how-to-integrate-helper-module-ruby)
March 06, 2017

Build Rock, Paper, Scissors in Ruby with Player vs Computer Gameplay in Ruby (/site_blogs/build-rock-paper-scissors-ruby-player-vs-computer-gameplay-ruby)
April 24, 2017

How to Add a Sudo User to Linux (/site_blogs/how-to-add-sudo-user-linux)
May 23, 2017

Tags

political candidates   professional rails development   application dev portfolio   example of mvc   ruby on rails framework tutorial   rails bootcamp

app examples   learning job meme   portfolio ideas   rails bdd   code management   free github   coding   tdd guide in rails   microservice vs monolith

Content by **Rails Camp (https://rails.devcamp.com/)**