Intro to Git, Github and the contribution cycle

Dominique Theodore

Agenda

- Version control
- Review of git concepts
- The forking workflow

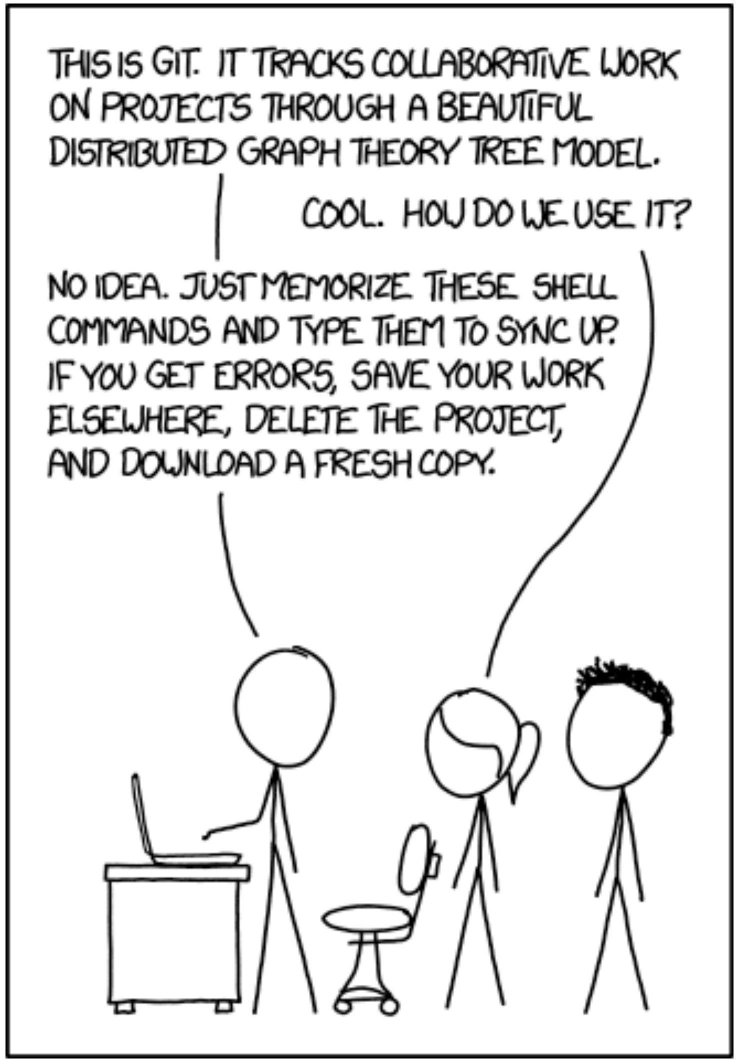
What is version control?

- A **Version Control System** is just software that helps you control (or manage) the different versions...of something (typically source code). Three of the most popular version control systems:
  - Git
  - Subversion
  - Mercurial
- There are two main types of version control system models:
  - the centralized model - all users connect to a central, master repository
  - the distributed model - each user has the entire repository on their computer

Git in a nutshell

- Git is a free and open-source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.
- Git relies on the basis of distributed development of software where more than one developer may have access to the source code of a specific application and can modify changes to it that may be seen by other developers.
- Initially designed and developed by Linus Torvalds for Linux kernel development in 2005.

Git in a nutshell


THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

Git in a nutshell

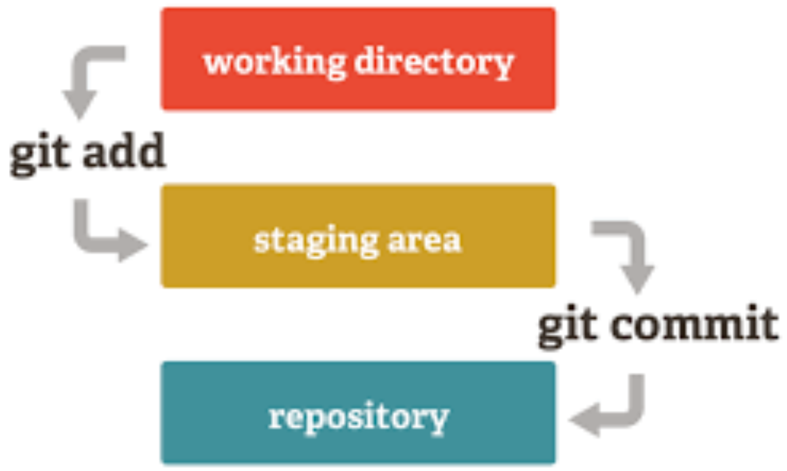
- Every git working directory is a full-fledged repository with complete history and full version tracking capabilities, independent of network access or a central server.
- Git allows a team of people to work together, all using the same files. And it helps the team cope up with the confusion that tends to happen when multiple people are editing the same files.

Review of git commands

- Creating a repository from scratch:
  - `git init`
- Review the commit history for the active branch
  - `git log`
- show modified files in working directory, staged for your next commit
  - `git status`

Staging and committing

- Git has a **staging area** that stores file changes that have not been committed yet.
- This two-step process (staging and committing) gives you the flexibility to choose which files and changes are permanently recorded in your commit history

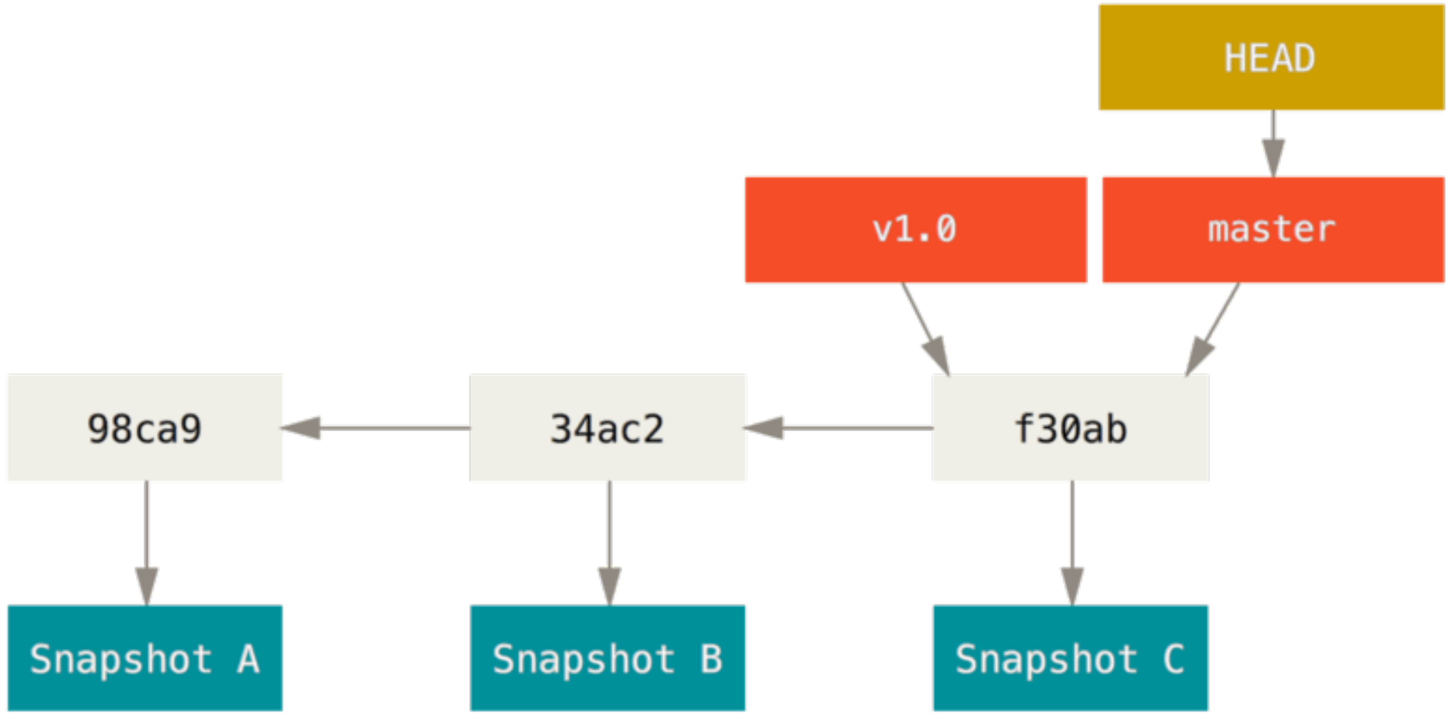


Staging and committing

- A **commit** records changes to one or more files in your branch. Git assigns each commit a unique ID, called a SHA or hash, that identifies:
  - The specific changes
  - When the changes were made
  - Who created the changes

Git branching

- A branch in Git is simply a lightweight movable pointer to a commit.
- The default branch name in Git is master. As you start making commits, you make some changes and commit again, the next commit stores a pointer to the commit that came immediately before it.

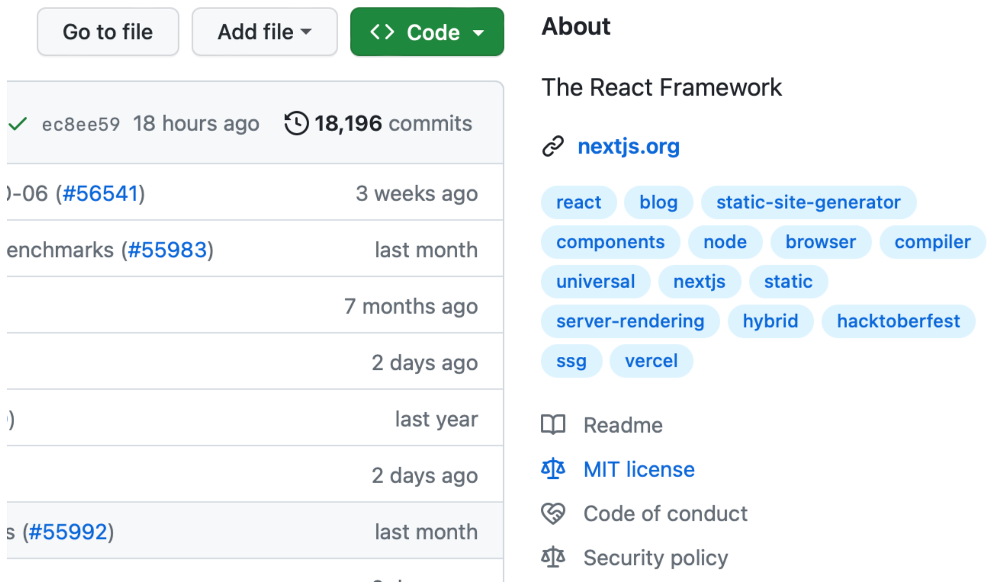


A typical Git workflow

```
# Start a new feature
git checkout -b new-feature main
# Edit some files
git add <file>
git commit -m "Start a feature"
# Edit some files
git add <file>
git commit -m "Finish a feature"
# Merge in the new-feature branch
git checkout main
git merge new-feature
git branch -d new-feature
```

Before you contribute

- Not all projects are open source on Github. Check the license first!
- If a project does not have an open-source license, then it is not open source.
- Read the contribution guidelines. This is often called **`contributing.md`**



The React Framework

nextjs.org

react, blog, static-site-generator, components, node, browser, compiler, universal, nextjs, static, server-rendering, hybrid, hacktoberfest, ssg, vercel

ec8ee59 18 hours ago 18,196 commits

)-06 (#56541) 3 weeks ago
enchmarks (#55983) last month
7 months ago
2 days ago
) last year
2 days ago
s (#55992) last month

Readme
MIT license
Code of conduct
Security policy

MIT License

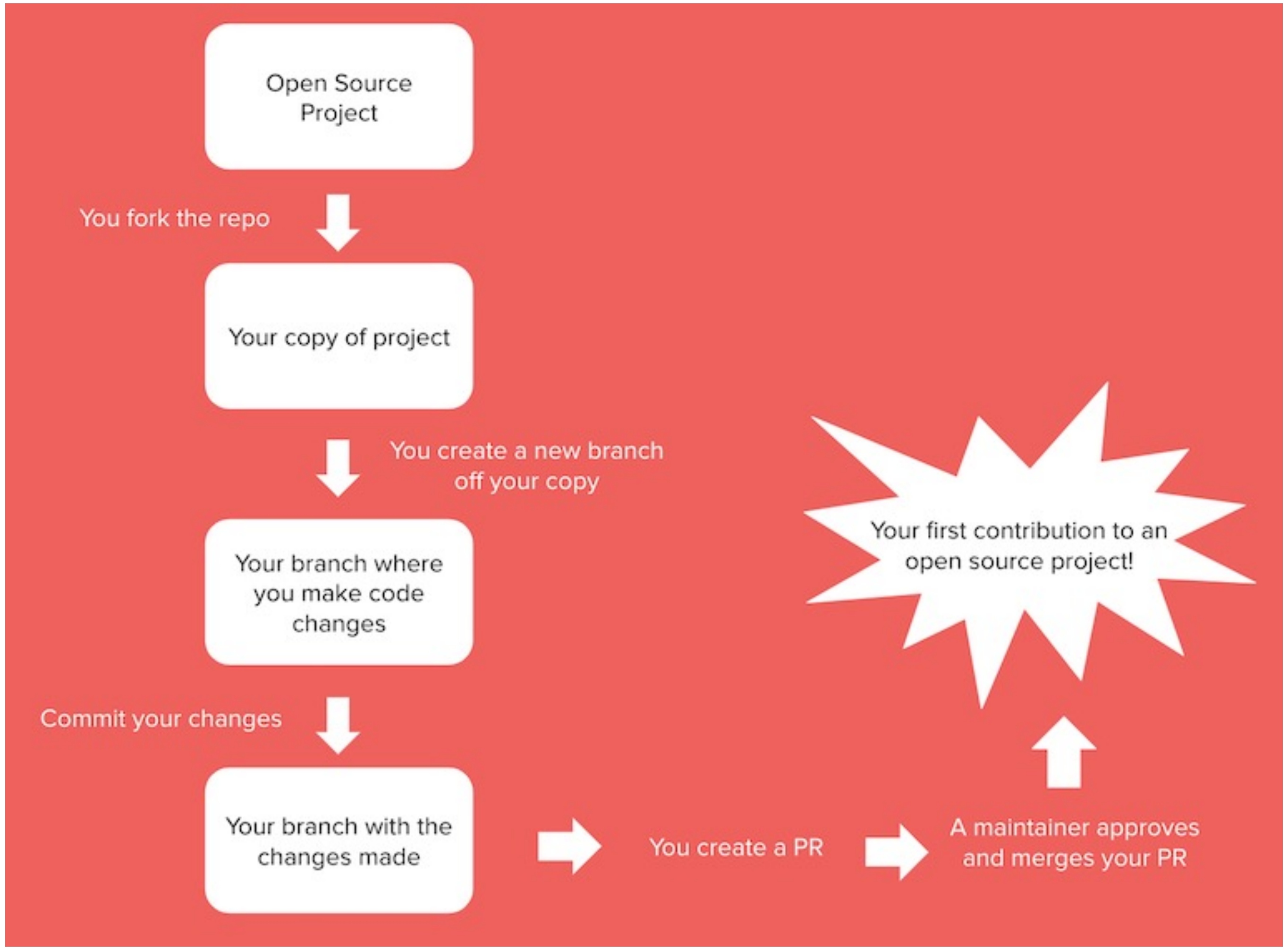
Copyright <YEAR> <COPYRIGHT HOLDER>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

A high-level view of the process

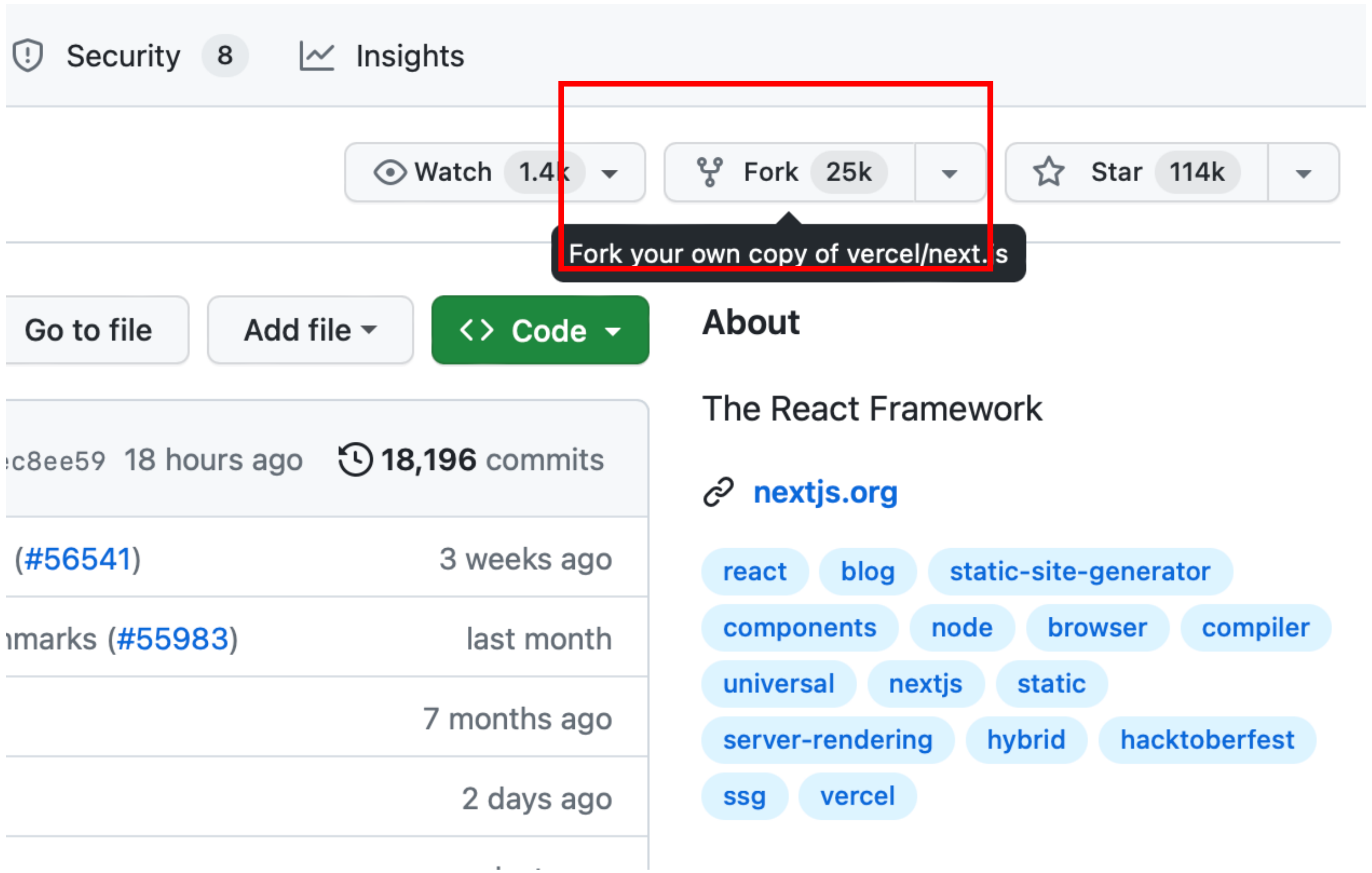

Open Source Project
You fork the repo
Your copy of project
You create a new branch off your copy
Your branch where you make code changes
Commit your changes
Your branch with the changes made
You create a PR
A maintainer approves and merges your PR
Your first contribution to an open source project!

Source: https://skillcrush.com/

Fork the repo

- Forking creates a personal copy of someone else's project repository on a version control platform like GitHub
- When you fork a repo, you create an exact copy of the original repository in your own GitHub account.
- This allows you to freely experiment with and make changes to the project without affecting the original repository
- **By convention, your forked repository is called the origin repository, while the original repository is the upstream repository.**

Fork the repo


Security 8
Insights
Watch 1.4k
Fork 25k
Star 114k
Fork your own copy of vercel/next.js
Go to file
Add file
Code
About
The React Framework
nextjs.org
react blog static-site-generator components node browser compiler universal nextjs static server-rendering hybrid hacktoberfest ssg vercel
:c8ee59 18 hours ago 18,196 commits
(#56541) 3 weeks ago
ımarks (#55983) last month
7 months ago
2 days ago

Clone your fork locally

- Cloning a repo means creating a copy of a repository in your local environment.
- When contributing to an open source project, you clone your forked repository
- `git clone https://github.com/johndoe/myrepo.git`

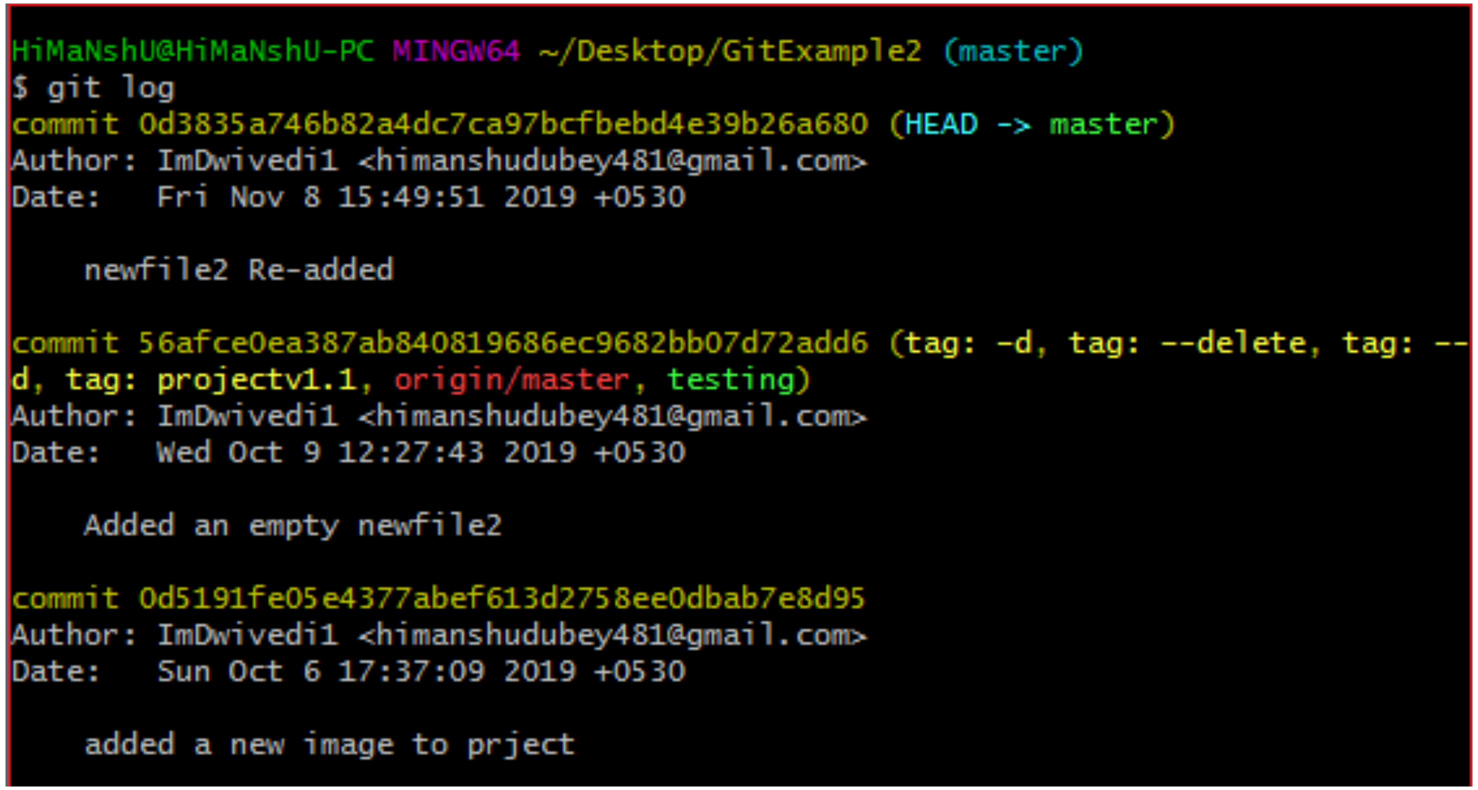
Create your new branch

- Create a new branch for the issue you want to work on
- Run the command below to create a new branch and navigate to it
- `git checkout -b <branch-name>`
- This command automatically switches to the new branch

Viewing your changes

To show the commit history for the currently active branch

- `git log`



```
HiMaNshU@HiMaNshU-PC MINGW64 ~/Desktop/GitExample2 (master)
$ git log
commit 0d3835a746b82a4dc7ca97bcfbebd4e39b26a680 (HEAD -> master)
Author: ImDwivedi1 <himanshudubey481@gmail.com>
Date:   Fri Nov 8 15:49:51 2019 +0530

    newfile2 Re-added

commit 56afce0ea387ab840819686ec9682bb07d72add6 (tag: -d, tag: --delete, tag: --
d, tag: projectv1.1, origin/master, testing)
Author: ImDwivedi1 <himanshudubey481@gmail.com>
Date:   Wed Oct 9 12:27:43 2019 +0530

    Added an empty newfile2

commit 0d5191fe05e4377abef613d2758ee0dbab7e8d95
Author: ImDwivedi1 <himanshudubey481@gmail.com>
Date:   Sun Oct 6 17:37:09 2019 +0530

    added a new image to prject
```

Create your new branch

Some conventions for naming branches:

- **Feature Branch**: If you are working on a new feature for your project
  - feature/add-login
  - feature/user-registration
- **Bug Fix Branch**: If you are fixing a bug in your code:
  - bugfix/fix-login-issue
  - bugfix/resolve-crash-bug
- **Hotfix Branch**: If you need to quickly fix a critical issue in production
  - hotfix/resolve-payment-bug
  - hotfix/patch-security-exploit

Stage and commit your changes

- Show modified files in working directory, staged for your next commit
- `git status`
- To add one or multiple files — but not all — to the staging area, run this command
- `git add <file-name-1> <file-name-2>`
- To add all files to the staging area, run this command:
- `git add .`

Stage and commit your changes

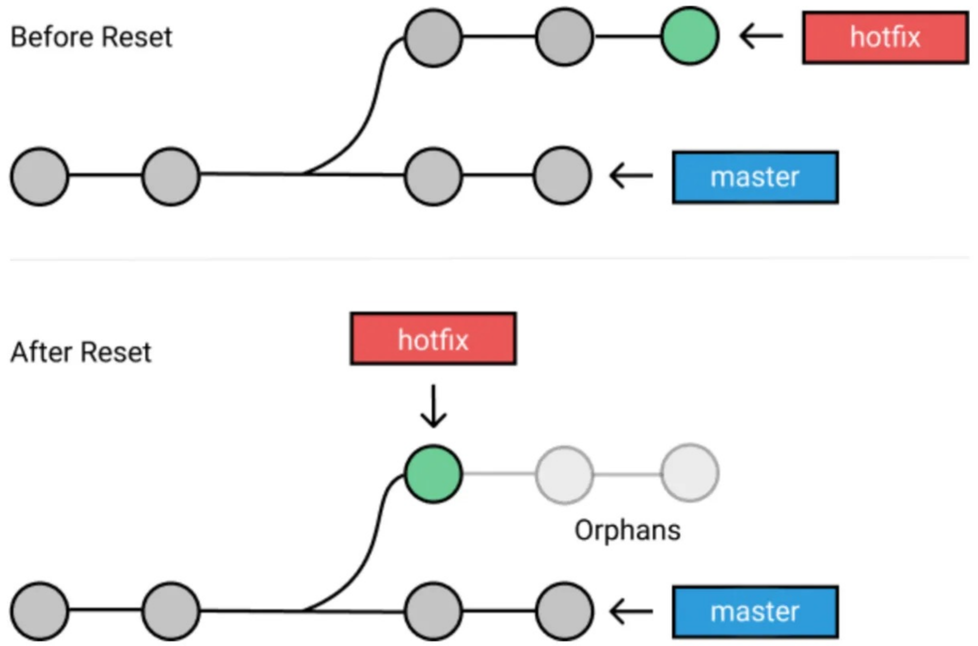
- To make a commit, run this command
- `git commit -m "Your message”`
- Push your changes
- `git push origin HEAD`

Undoing changes

- `git reset --soft`
- The --soft aims to change the HEAD (where the last commit is in your local machine) reference to a specific commit. For instance, if we realize that we forgot to add a file to the commit, we can move back using the --soft with respect to the following format:
- `git reset --soft HEAD~n` to move back to the commit with a specific reference (n). git reset --soft HEAD~1 gets back to the last commit.
- `git reset --soft <commit ID>` moves back to the head with the <commit ID>

Undoing changes

- `git reset --hard` is a Git command used to reset the current branch to a previous commit, discarding any local changes and modifications made to the files. Use with caution!!



Stage and commit your changes

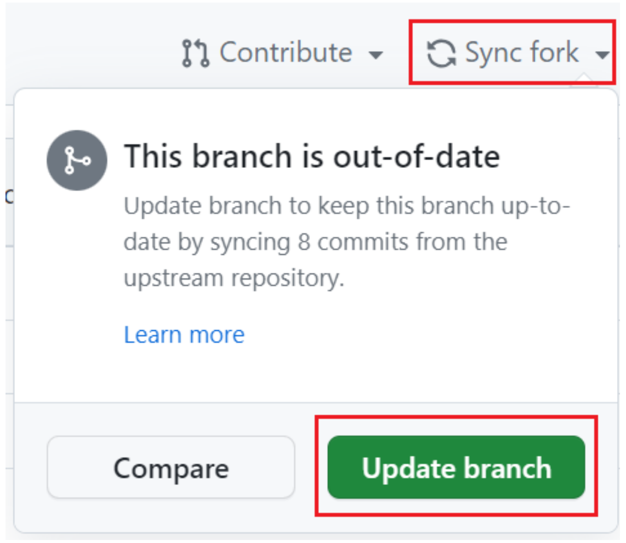
- Use a descriptive commit message as far as possible e.g
  - “Fix bug causing login page to crash on invalid input”
  - “Add feature to sort products by price in the shopping cart”

	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAANDS	2 HOURS AGO

AS A PROJECT DRAGS ON, MY GIT COMMIT MESSAGES GET LESS AND LESS INFORMATIVE.

Update the origin repository

- When you are working on changes, the state of the origin and your local repositories at this time will sometimes no longer be the same as the upstream.
- To update the origin repository on Github
- Click the Sync fork dropdown button.
- Click the green Update branch button.
- You should then pull the changes from the main branch in origin
- `git pull origin main`



This branch is out-of-date

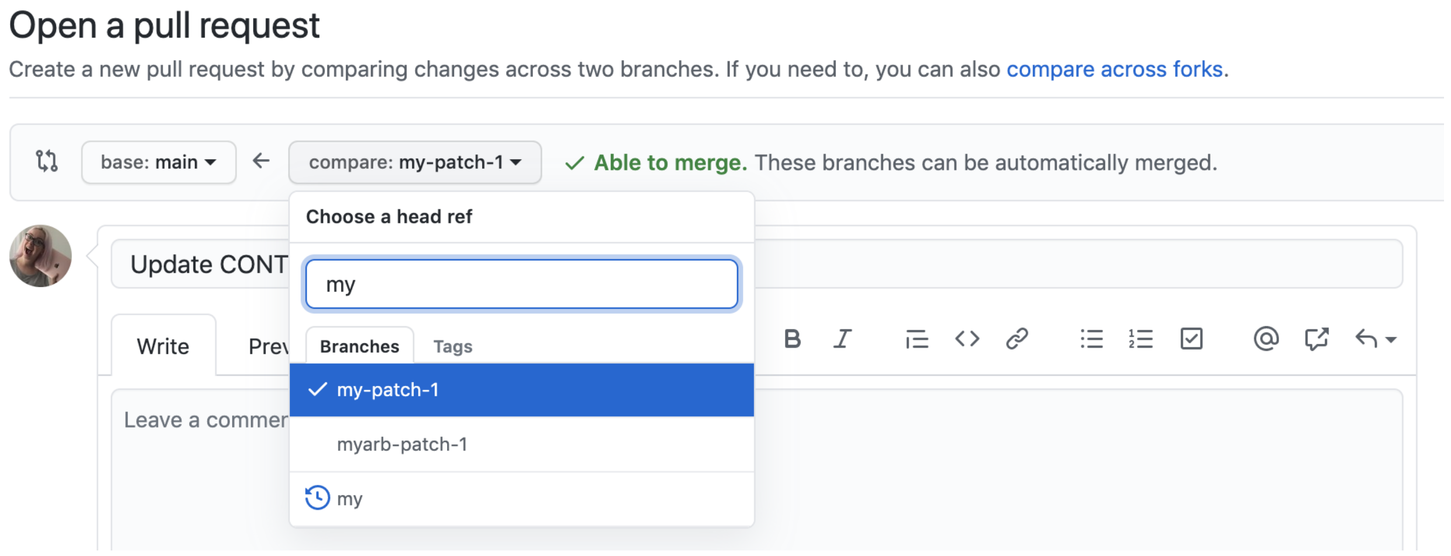
Update branch to keep this branch up-to-date by syncing 8 commits from the upstream repository.

Push your changes

- Push your changes: This means moving changes from the local to the remote repository. Run this command:
- `git push origin <branch-name>`

Submit a Pull Request

- Once you are confident in your changes, push your branch to your forked repository and submit a pull request (PR) to the original project's repository.



Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.

base: main ← compare: my-patch-1 ✓ Able to merge. These branches can be automatically merged.

Choose a head ref

my

Branches Tags

✓ my-patch-1

myarb-patch-1

my

Update CONT

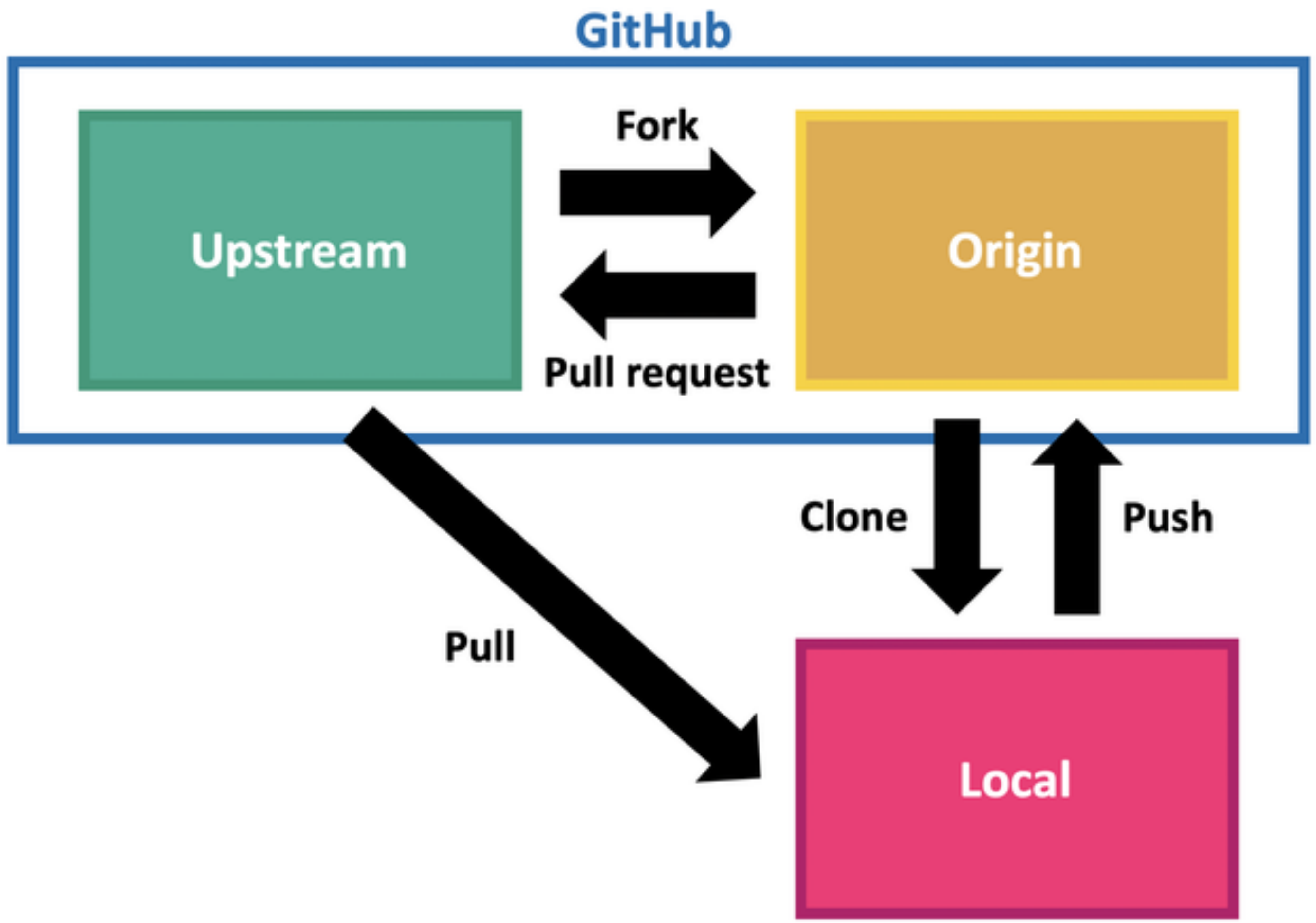
Write Prev

Leave a commer

Submit a Pull Request

- Pull requests are an essential part of the collaborative development process
- The main purpose of a pull request is to propose changes to a codebase and initiate a discussion among team members before merging those changes into the main branch.
- When making pull requests, ensure you include a detailed description of your changes, any relevant documentation updates, and reference any related issues.

The forking workflow



Resources

- Github Cheat Sheet https://drive.google.com/file/d/1LwkLF7HT9pU4JnHmmb4RwdGGdwKPhoNb/view?usp=sharing
- Getting Git right https://www.atlassian.com/git
- A Step-by-Step Guide: How to Contribute to Open Source as a Developer https://medium.com/@laasrisaid34/a-step-by-step-guide-how-to-contribute-to-open-source-as-a-developer-1fe7a195ec2a
- Forking workflow https://www.atlassian.com/git/tutorials/comparing-workflows/forking-workflow