

CMU 02-601: Recitation 3 - Matthew Wolff

Git: Version Control

Getting Started

Download

Download [Github Desktop](<https://desktop.github.com/>)

Join

Join [GitHub](<https://github.com/join>) by creating an account
• Use a professional-enough name, since you'll typically include this in your résumé

What is Version Control?



WHY DO WE
NEED IT?



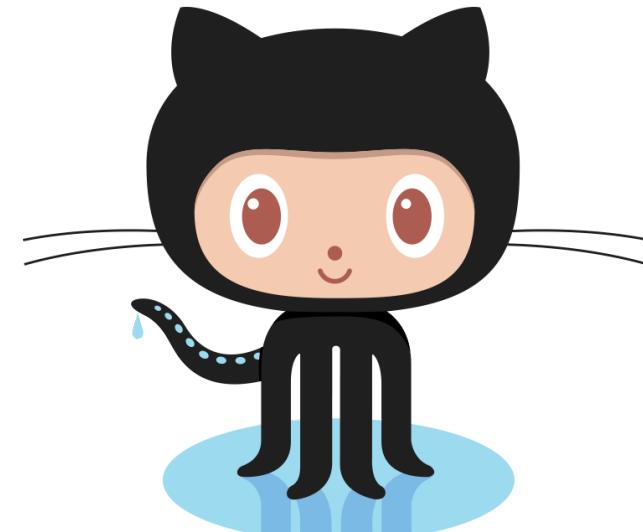
WHY SHOULD
WE USE IT?



WHAT ARE THE
BENEFITS?

What is Git?

- > *"By far, the most widely used modern version control system in the world today is Git. Git is a mature, actively maintained open source project originally developed in 2005 by Linus Torvalds, the famous creator of the Linux operating system kernel. A staggering number of software projects rely on Git for version control, including commercial projects as well as open source."*



What is GitHub?



So what's the
meaningful
difference
between
them?

- **GIT**
 - A command line tool developed by Linus Torvalds
 - Independent from GitHub
- **GITHUB**
 - A Platform built around Git
 - No need to understand Command Line

```
[~/github_workshop] [matthew@mattmoose] [79%] [5749]
[>:] > git init && hub create
Initialized empty Git repository in /Users/matthew/github_workshop/.git/
Updating origin
https://github.com/MatthewWolff/github_workshop
[~/github_workshop] [matthew@mattmoose] [79%] [5750] (git master github_workshop [Matthew])
[>:] >
```

How to use GitHub?

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner * **Repository name ***

Great repository names are short and memorable. Need inspiration? How about [glowing-spork](#)?

Description (optional)

Public
Anyone on the internet can see this repository. You choose who can commit.

Private
You choose who can see and commit to this repository.

Publish GitHub repository

MatthewWolff / **git_workshop**

Visibility: Public Private

/Users/matthew/Desktop/grad_school/ms/second_year/fall/ta_program

▶ Advanced

Cancel **Publish**

Current Repository git_workshop **Current Branch** master **Publish repository**
Publish this repository to GitHub

Changes 4 **History** README.md

4 changed files

imag.../atom_github_support.png

images/create_github_repo.png

index.html

README.md

@@ -0,0 +1,91 @@
1 +# Git: Overview
2 +
3 +<!-- TODO: create repo and commit to it -->
4 +<!-- TODO: create worked example with the <username> repo -->
5 +
6 +## What is Version Control
7 +Imagine you're working on a paper CancerEvolution.docx and you re-write a section but you're not sure if you want to keep it. Therefore, you "Save As" under "CancerEvolutionRewrite.docx". As you're working on this you end up re-writing another section and "Save As" to create a slightly different copy. Or maybe you just create draft after draft and want to have different versions for later. You're essentially creating different version of the same document and manually organizing them on your desktop! Not very effective though.
8 +
9 +## What is Git?
10 +A version control system for code! Gone are the days of "FinalPaperDraft.docx", "FinalPaperDraftFinal.docx", "FinalPaperDraftFinalFinal.docx".
11 +
12 +>> "By far, the most widely used modern version control system in the world today is Git. Git is a mature, actively maintained open source project originally developed in 2005 by Linus Torvalds, the famous creator of the Linux operating system kernel. A staggering number of software projects rely on Git for version control, including commercial projects as well as open source."
13 +
14 +It's extremely useful for tracking and undoing changes to your code over time, as well as sharing your code with others.
15 +
16 +## What is GitHub?
17 +It's an online platform that lets you store and version control your code. It's built around Git. There are others like it, such as GitLab and BitBucket.</p>

Summary (required)

Description

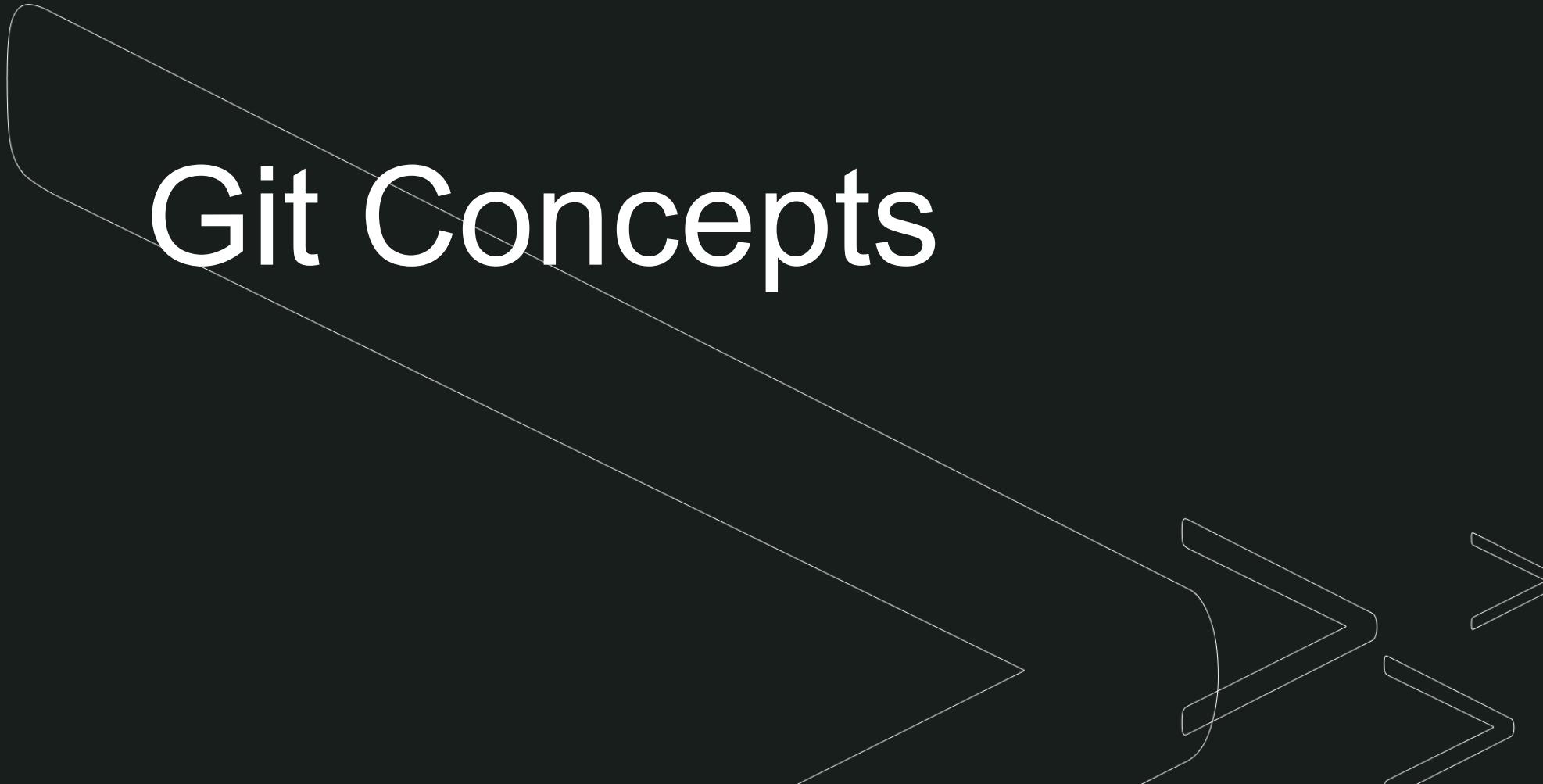
Commit to master



Note: Free GitHub Premium for students

- As a note, being a student gets you a [premium version (Github Student Dev pack) of GitHub for free: <https://education.github.com/pack>
- *>"It includes everything in GitHub Free, unlimited collaborators for private repositories, and advanced code review tools for private and public repositories."*

Git Concepts

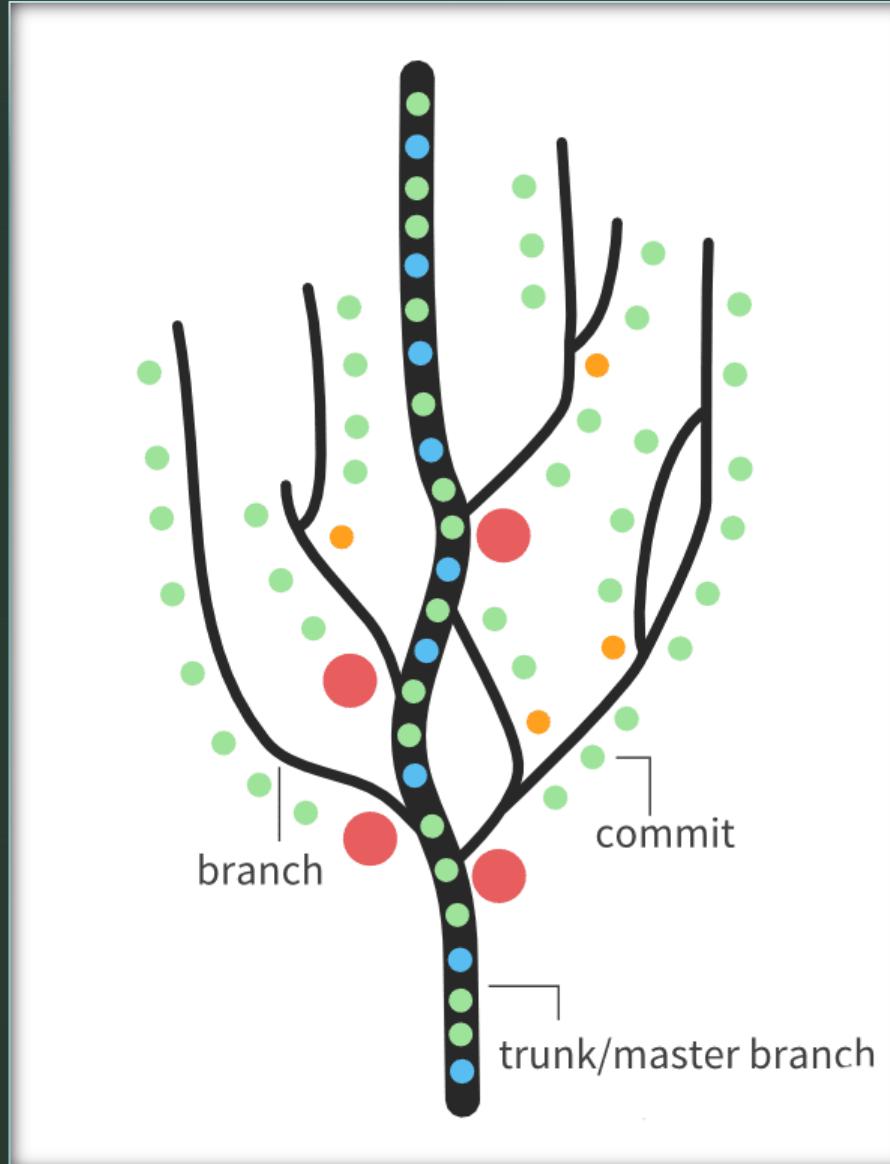


Git Concepts

- The Repository
- Cloning
- Commiting
- Pushing, Pulling
- Branching
- Merging
- Pull Requests
- Forking

Tree Analogy

- Bed of soil
- Seed
- Nutrients
- Growing Branches
- Merging Branches





What's Under the Hood?

- Written in C
- Complex
- a Hash Map (like a Python dictionary)
- Graph-structured
- Most people don't worry about the underlying implementation
 - It doesn't drastically improve how you use it

Example Exercise!

1. Open GitHub Desktop



2. Create a new (local) repository

- * Initialize it with a README
 - what is a readme?
 - what is markdown?

Create a New Repository

Name

Description

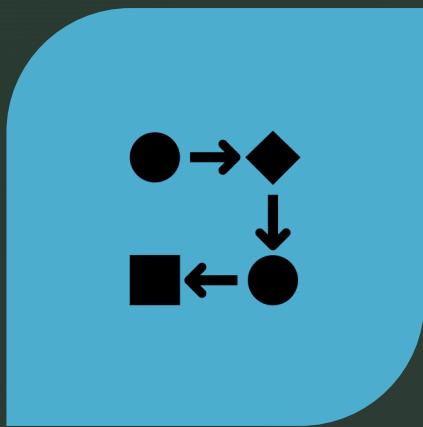
Local Path

Initialize this repository with a README

3. Publish your repository

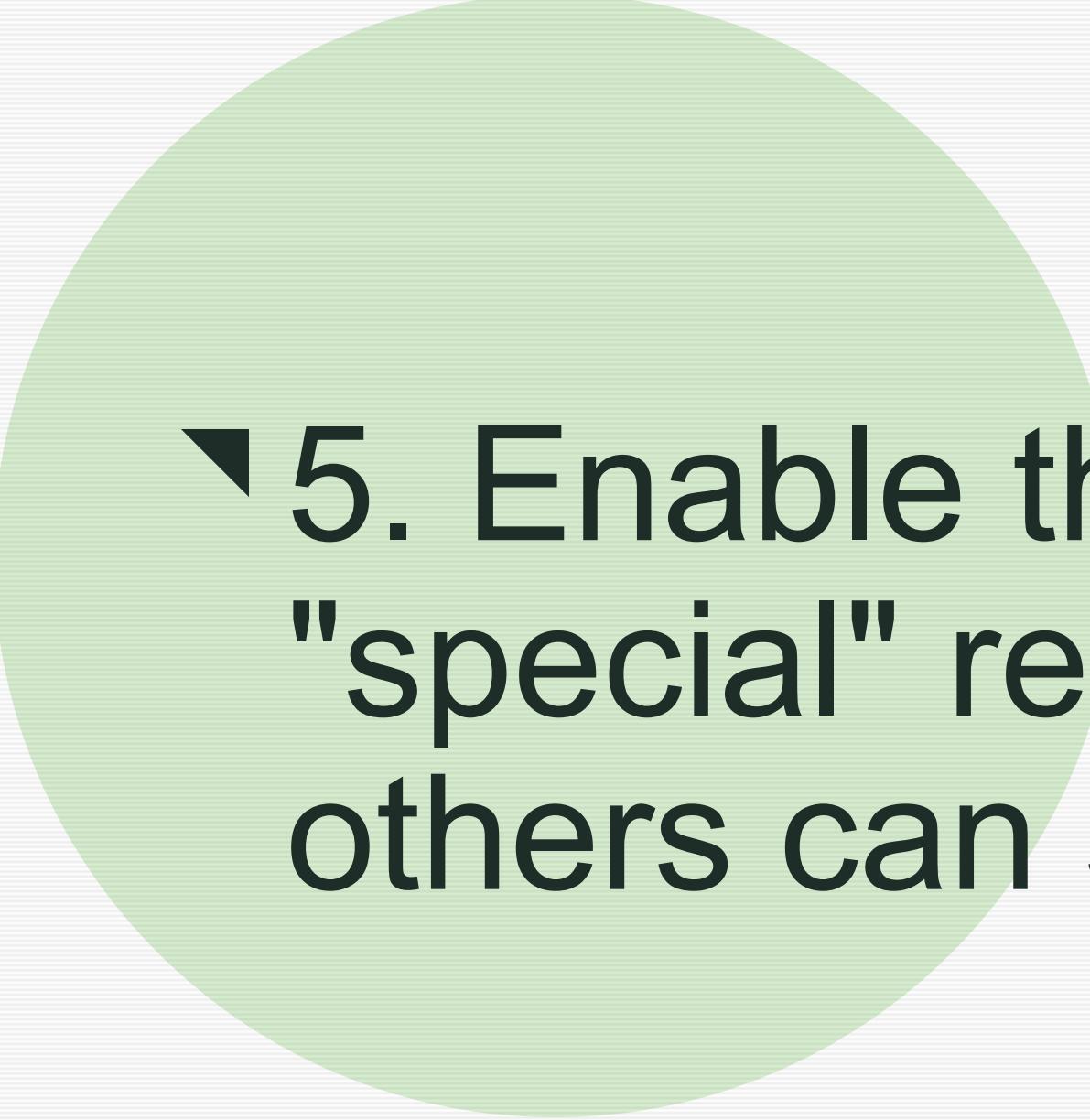


- MAKE SURE IT'S PUBLIC!



- THERE'S ALREADY ONE
AUTOMATIC COMMIT TO IT

4. Go to
[GitHub](<https://github.com>)
on your profile



▼ 5. Enable this "special" repo so others can see it!

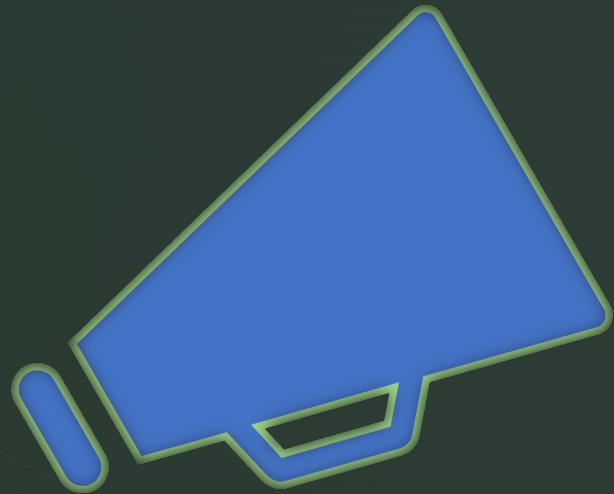
6. Go back to your files. Change the README.

- You now have a local repo. Let's change the README to a small intro for your github account.
- Mine is below!

```
# Hey! 😊
* UW → Computer Science, Genetics & Genomics
* CMU → Computational Biology, Data Science
```

```
![GitHub Stats](https://github-readme-stats.vercel.app/api?username=matthewwolff&include_all=true)
![Top Languages](https://github-readme-stats.vercel.app/api/top-langs/?username=matthewwolff&langs_count=5)
```

7. Commit with
a message and
Push/Publish!



8. Go to on your profile again!

The screenshot shows a GitHub profile page for a user named Matthew Wolff. The profile picture is a Shiba Inu dog wearing a top hat and bow tie. The user's name is Matthew Wolff, and their GitHub handle is MatthewWolff. They are described as a Carnegie Mellon Grad Student. The profile has 15 followers, 3 following, and 9 stars. Contact information includes email (mwolff@cmu.edu) and a website (wolff.sh). The profile is marked as a Highlights contributor to the Arctic Code Vault and is a PRO member.

Matthew Wolff
MatthewWolff
Carnegie Mellon Grad Student
[Edit profile](#)

15 followers · 3 following · ⭐ 9
✉ mwolff@cmu.edu
🔗 wolff.sh

Highlights
★ Arctic Code Vault Contributor
★ PRO

Organizations

Overview **Repositories 81** **Projects** **Packages**

MatthewWolff / README.md

Hey! 🎉

Send feedback

- UW → Computer Science, Genetics & Genomics
- CMU → Computational Biology, Data Science

Matthew Wolff's GitHub Stats

- ⭐ Total Stars: 39
- ⌚ Total Commits: 982
- 💡 Total PRs: 12
- ❗ Total Issues: 8

A+

Most Used Languages

Language	Percentage
Python	55.59%
TeX	20.52%
Jupyter Notebook	16.78%
Shell	7.12%

Pinned

TwitterScraper
Scrape a User's Twitter data! Bypass the 3,200 tweet API limit for a User!
Python ⭐ 21 ⚡ 3

MarkovTweets
Trump Twitter Bot. Uses N-Gram Language Modeling with Markov Chains
Python ⭐ 5

Customize your pins

Let's do a Pull Request!



1. Create a new branch using GitHub desktop



2. Make some changes



3. Commit them! Push them to origin!



4. Create a pull request! (will open GitHub)



5. Merge the pull request

 Current Branch
master

 Publish branch
Publish this branch

Branches **Pull Requests 1**

[New Branch](#)

Default Branch

✓ master 27 minutes ago

Recent Branches

⌚ dev_branch 35 minutes ago

Other Branches

⌚ origin/dev_branch 35 minutes ago

⌚ origin/master 27 minutes ago

Make a change!

example: echo >> README.md



Added a new line

hehe



Commit to **dev_branch**

No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.



Push commits to the origin remote

You have 1 local commit waiting to be pushed to GitHub.

Always available in the toolbar when there are local commits waiting to be pushed or

[Push origin](#)

No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.



Create a Pull Request from your current branch

The current branch (dev_branch) is already published to GitHub.

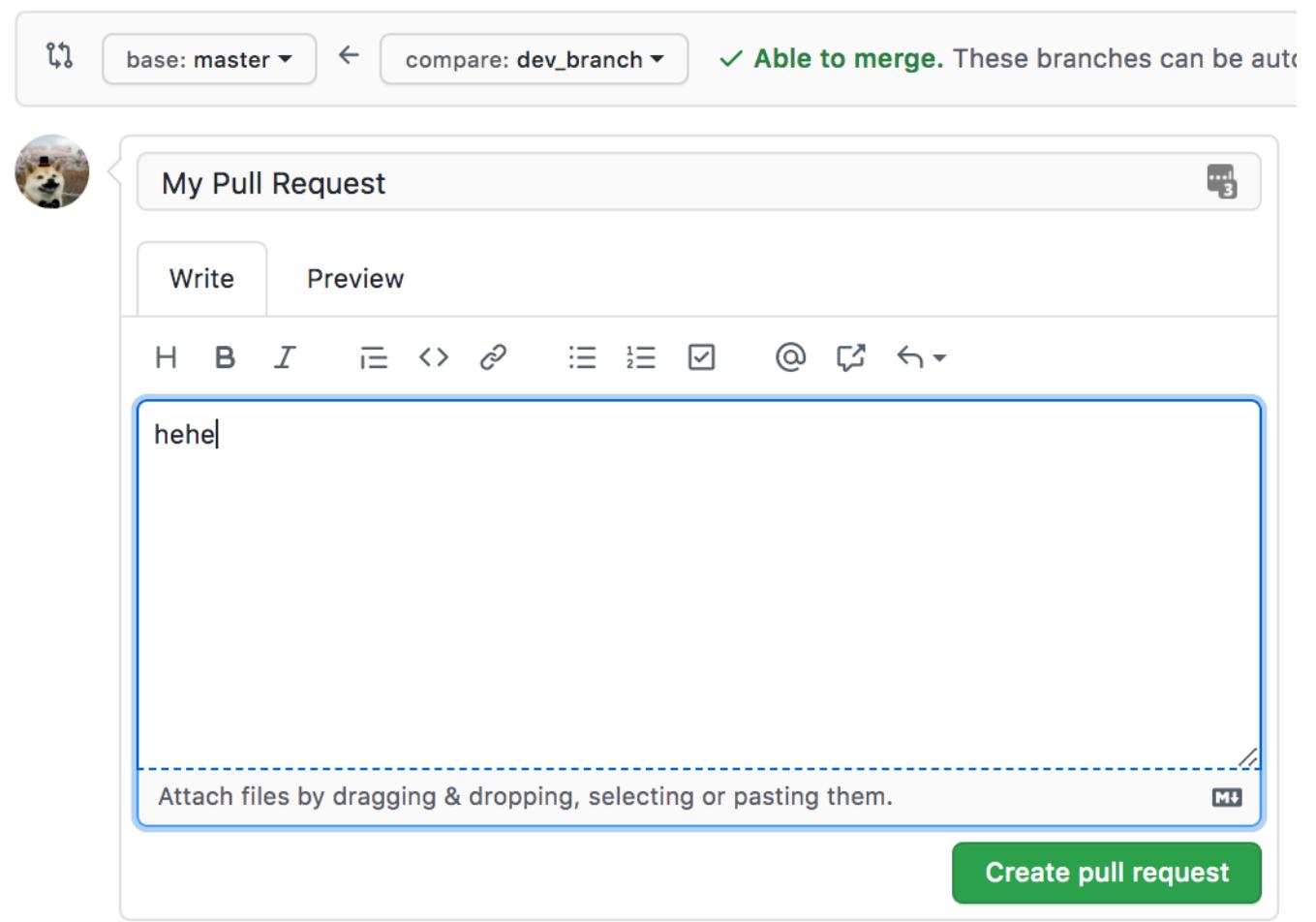
Create a pull request to propose and collaborate on your changes.

[Create Pull Request](#)

Branch menu or

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [cor](#)



rewrote first part, added more #1

 Open

MatthewWolff wants to merge 2 commits into `master` from `dev_branch` 

 Conversation 0

-o Commits 2

 Checks 0

 Files changed 1



MatthewWolff commented 18 minutes ago

 ...

No description provided.



MatthewWolff added 2 commits 27 minutes ago



rewrote first part, added more

623855e



Merge branch 'master' into dev_branch

 Verified

6dc6505

Add more commits by pushing to the `dev_branch` branch on **MatthewWolff/.sandbox**.



Continuous integration has not been set up

GitHub Actions and [several other apps](#) can be used to automatically catch bugs and enforce style.



This branch has no conflicts with the base branch

Merging can be performed automatically.

 Merge pull request



You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Add more commits by pushing to the dev_branch branch on GitHub



Merge pull request #2 from MatthewWolff/dev_branch

I did it

mwolff@cmu.edu



Choose which email address to associate with this commit

Confirm merge

Cancel

My Pull Request #2

Merged

MatthewWolff merged 2 commits into `master` from `dev_branch` now



Conversation 0

Commits 2

Checks 0

Files changed 1



MatthewWolff commented 5 minutes ago

...

hehe

New commit

df9543c

another

75fbc40

MatthewWolff merged commit `8acfb8d` into `master` now

Revert



Pull request successfully merged and closed

Delete branch

You're all set—the `dev_branch` branch can be safely deleted.

.gitignore

```
# Created by https://www.toptal.com/developers/gitignore/api
# Edit at https://www.toptal.com/developers/gitignore?templa

### IntelliJ ####
# Covers JetBrains IDEs: IntelliJ, RubyMine, PhpStorm, AppCode, PyCharm, DataGrip, WebStorm, CLion, GoLand, IdeaVim, Rider, Unity
# Reference: https://intellij-support.jetbrains.com/hc/en-us/articles/206544839

# User-specific stuff
.idea/**/workspace.xml
.idea/**/tasks.xml
.idea/**/usage.statistics.xml
.idea/**/dictionaries
.idea/**/shelf

# Generated files
.idea/**/contentModel.xml

# Sensitive or high-churn files
.idea/**/dataSources/
.idea/**/dataSources.ids
.idea/**/dataSources.local.xml
.idea/**/sqlDataSources.xml
.idea/**/dynamic.xml
.idea/**/uiDesigner.xml
.idea/**/dbnavigator.xml

# Gradle
.idea/**/gradle.xml
.idea/**/libraries
```

Command Line Git

Key Commands

<http://commands.wolff.sh>

Questions?

