



# Building Your GitHub Portfolio

Jacob Gladfelter

HPC Systems Admin & Unit Support



# What is Github?

GitHub is a website that allows you to:

- Showcase or share your code.
- Track and manage changes to your code over time.
- Let others review your code, and make suggestions to improve it.
- Collaborate on a shared project, without the fear of overwriting or losing data.

The screenshot displays the GitHub profile of Jacob Gladfelter, known as `um-jglad`. The profile includes a bio, a 'Follow' button, and a list of organizations. The 'Pinned' section shows two repositories: `umich-biostatistics/HPC-R-Examples` and `BDSI-R_Workshop`. The '446 contributions in the last year' section features a contribution graph. The 'Activity overview' section shows a bar chart comparing commits (88%) and issues (1%).

**Profile Information:**

- Name:** Jacob Gladfelter
- Username:** um-jglad
- Followers:** 0
- Following:** 6
- Current Organization:** University Of Michigan | SPH Biostatistics, Ann Arbor

**Pinned Repositories:**

- `umich-biostatistics/HPC-R-Examples` (Public): Example R jobs and scripts for HPC use.
- `BDSI-R_Workshop` (Public): testing out publishing from RStudio

**Contributions:** 446 contributions in the last year.

**Activity Overview:**

- Commits:** 88%
- Issues:** 1%

# Why should I care?

Good question.

Hosting your work on GitHub can be an easy way to show potential employers:

- What you can do
- The quality of your work
- Your ability to collaborate

It's also a great way to help make your code more easily reproducible.

# The Man behind the curtain: Git



**ONE DOES NOT SIMPLY**



**UNDERSTAND GIT**

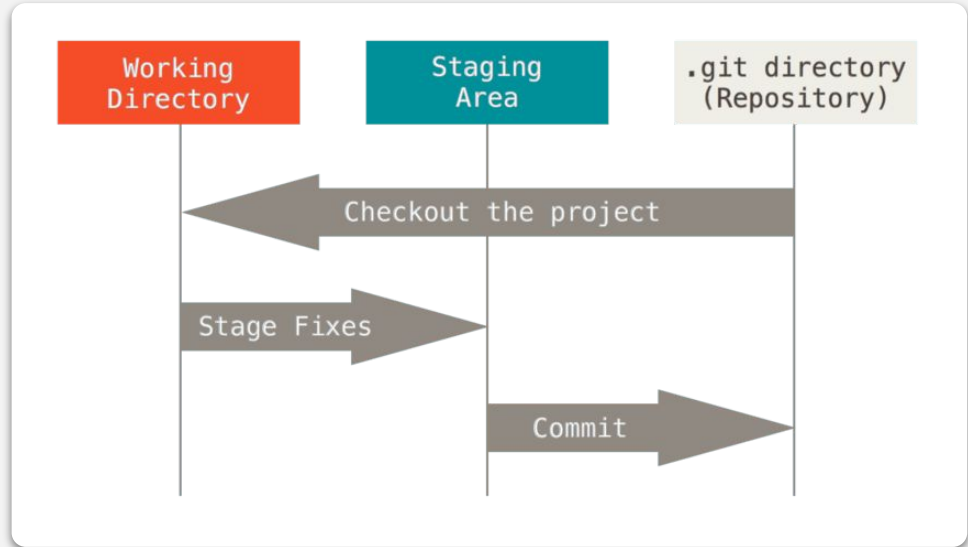
*MemesHappen*

# The Magic of Git

GitHub is built on Git, a **version control system** for tracking file changes.

Git is what enables all of those features listed on the first slide.

You can use Git without GitHub, but **pushing** changes to GitHub allows for easy collaboration so that others can **pull** those changes and continue to **contribute**.



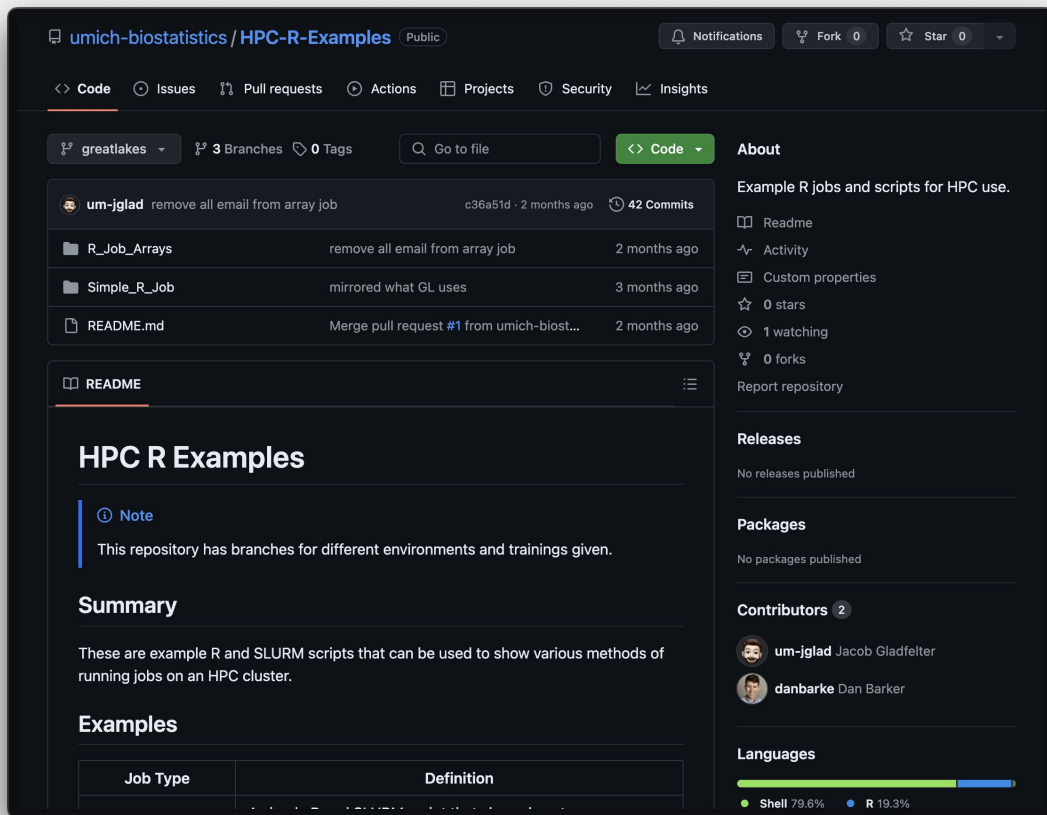
# Repository (Repo)

Think of a GitHub Repository as a directory on a remote server.

This **remote repository** contains all of your code, your files, and each file's revision history.

It can be public or private, and various permissions can be set to allow others to collaborate.

**ALL GOOD REPOS HAVE A README.**

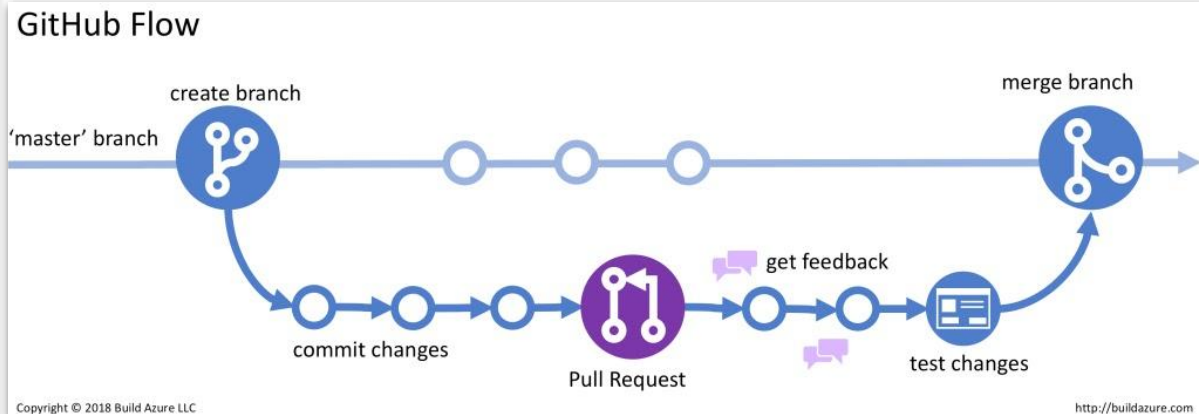


Source: [GitHub Docs | About Repositories](#)

# GitHub Flow

The basics of using a GitHub repo can be boiled down to:

1. Creating a “Branch”
2. Making Changes
3. Creating “Pull Requests”
4. Reviewing Changes
5. “Merging”





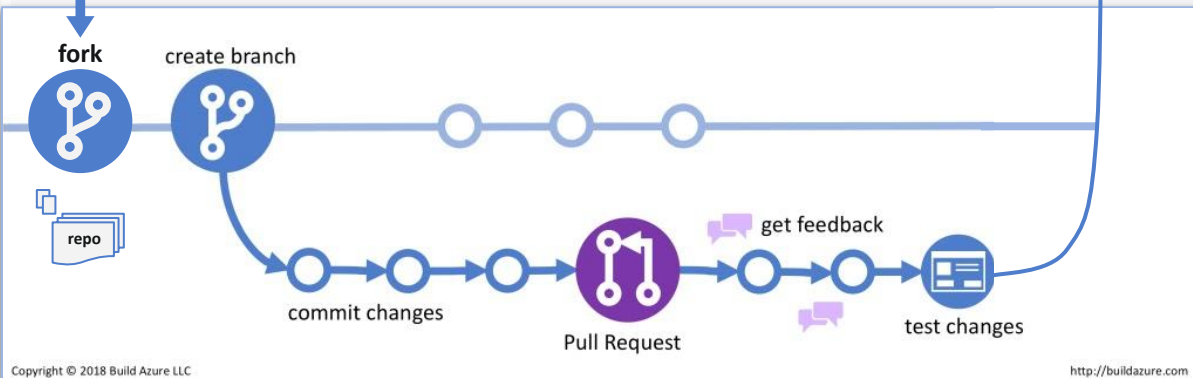
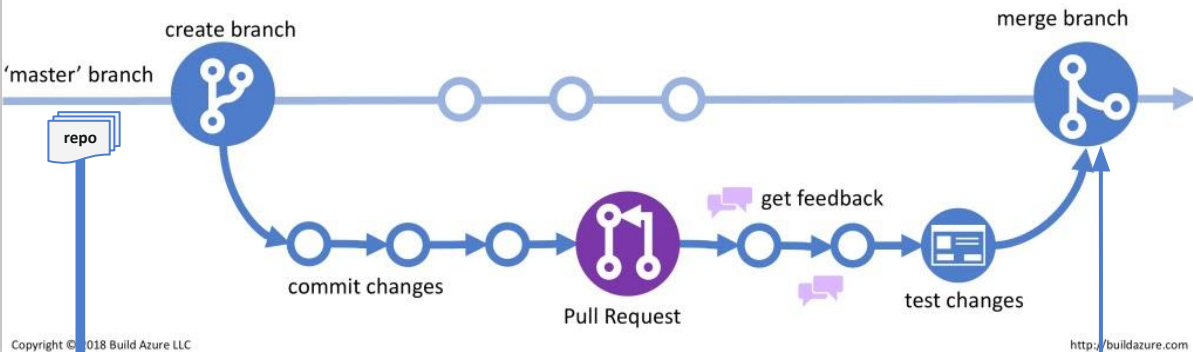
# Contributing

If you want to contribute to someone else's project where you're not already a contributor, you can create a **"fork"**.

Forking creates a copy of the repo under your name that is tied back to the original.

This allows you to make a branch, test changes, and then submit a pull request.

## GitHub Flow



# Creating a Presence

Make yourself, your code, and  
your contributions visible!

1. Sign up for an account
2. Create a Repo
3. Upload your Code
4. Update your profile
5. Profit?

# Resources

## Documentation

- [GitHub Docs](#)
- [Git - Documentation](#)
- [GitHub Pages](#)
- [GitHub and RStudio](#)
- [rmarkdown's site generator | R Markdown: The Definitive Guide](#)

## Examples

- [GitHub Profile README Examples](#)
- [coatless \(James J Balamuta\) · GitHub](#)
- [BDSI ggplot Example](#)