Github Basics class notes

In this course, you learn to use git and GitHub for version control, collaboration, and open source.

GitHub is a website for hosting and collaborating on projects.

Git is a version control system that's typically used with GitHub.

With Git every developer has a local copy of the repository on their computer.

If they make changes they can upload those changes to GitHub and share them with a larger team.

GitHub also has other collaboration features like issues and pull requests that help teams collaborate more effectively.

- go to GitHub.com.
- If you already have an account, you can sign in.
- If you don't, though, it's easy and free to create a new account.
- pick a username.
- also provide my email address and enter a password.
- sign in.

GitHub is free to use for public projects,

you can host projects privately with a paid monthly subscription.

When you've signed up check your profile. It's empty first.

edit your profile and add some information about yourself.

follow some fellow GitHub users, star interesting repositories, and add your own repository.

Soon you'll have a couple green contribution squares on your contribution graph that show your activity.

The more active you are the darker your squares become.

Click Edit Profile and fill in our profile information. You can do it by clicking on Settings.

A repository is the most basic element of GitHub, like a projects folder.

repository contains all the project files, including documentation and stores the history of each file.

GitHub provides features like issues and pull requests. Issues are used to track bugs and feature requests.

We can see that there are several open issues and even more closed issues.

README.md for descriptions

If you have a feature request or a bug for a specific project, the first thing to do would be to search the issues to see if someone else has already reported it.

A pull request represents a change, such as adding, modifying or deleting lines of code or whole files.

Pull requests let you discuss and review changes and are typically used to resolve issues.

Taking a look at pull requests, you can learn a lot about the evolution of a project.

You can click in and see who opened the pull request for conversations surrounding the change, the commits involve and what files changed.

When you're on a project page, you can click the watch button at the top of the page to stay up-to-date with changes.

You'll get notifications for any new poll requests and issues that are created. You can also star projects that you like and find interesting. Starring is the most simple way to contribute to open source and let project owners know you appreciate their work.

To see a list of all the projects you have starred, you can go to your stars page.

A fork is a copy of a repository, similar to a branch. However, it is typically used when you don't have access to a project.

Forking a repository, like branching, allows you to freely experiment with changes without affecting the original project.

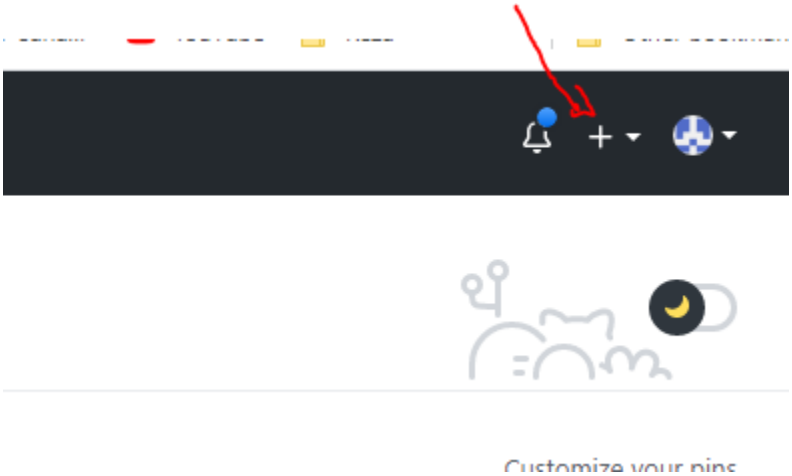
Forks are either used to propose changes to someone else's project, as the user did in the Get It project with the typo per request, or to use someone else's project as a starting point for your own idea.

You can also follow users. When you follow someone on GitHub, you will get notifications on your dashboard about their activity.

Take a moment to star some of the repositories you explored in the last stage and follow some GitHub users.

To create a repository, click the plus icon next to your profile picture and click New Repository.

For this example we'll name our repository Tree House.




write a short description,

Choose public, this means anyone can see this repository.

For private you'd have to upgrade to a paid plan.

Since we wanna share our work and fill out our GitHub profile, we want our work public.

Do not select initialize this repository with a README if you are adding a project we already have.

If I were to select this and then try and push up a Git repository from our local computer, we'd have a conflict.

We have the option to add a .gitignore file or a license file, but we'll skip that for now.

And click Create repository.

some of GitHub's most essential features include markdown, branching, and pull requests.

These features are the backbone of open source on GitHub. Create a repository, and then open terminal and push up a project to GitHub.

Git Commands:

- git init: initialize a git repository in your directory
- git status: check the status of the repository
- git add file_name: add one file
- git add .: add all files
- git commit -m "commit message": commit your files along with a message
- git remote add origin https://github.com/username/reponame.git - the remote url to your GitHub repo
- git push origin master: push your files up to github on the master branch

now we have a repository on GitHub.

issues are typically used to track bugs or features, but issues actually can be used for just about any task you want to track. It could be your to-do list or a list of goals you want to accomplish.

The issue remains open while it's yet to be complete, and then you can close the issue once it's been taken care of.

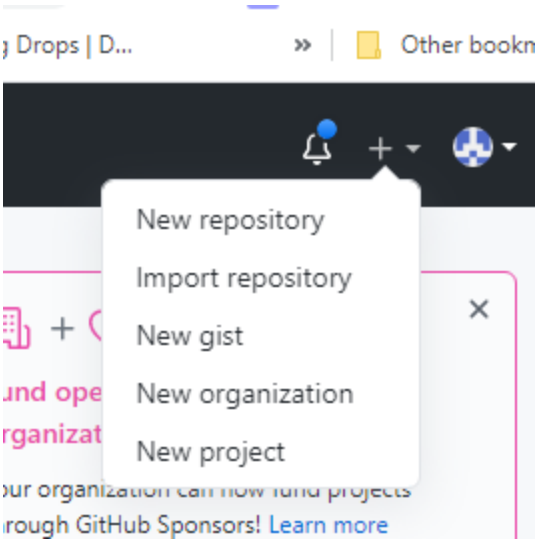
Let's open some issues in our repo. Navigate to the Issues tab from the main view, and click on the New Issue button.

If you have additional comments you can add them below the main issue body.

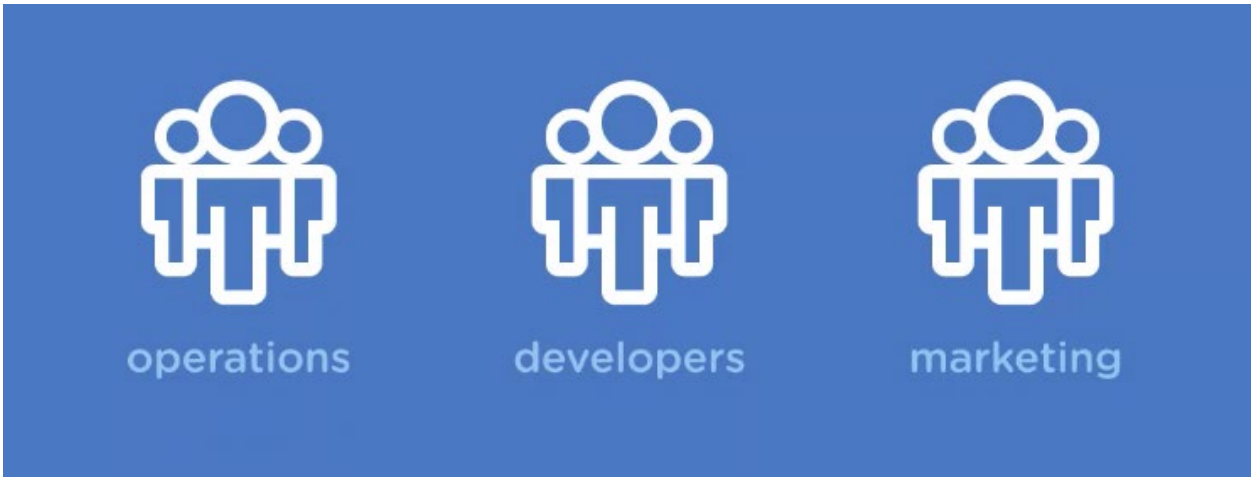
We could also open another issue too.

We can create a different branch to work in parallel.

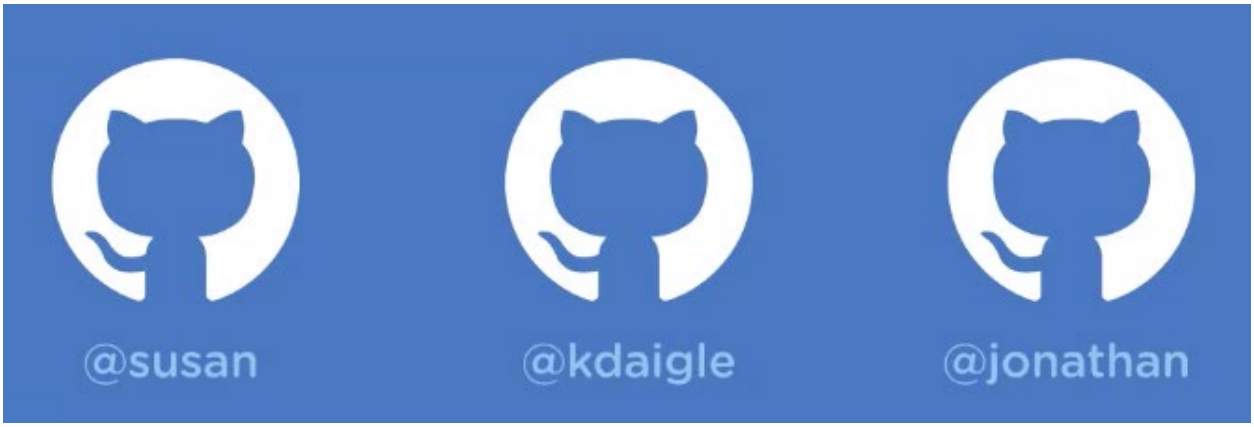
Github organization… using + icon and create a new organization



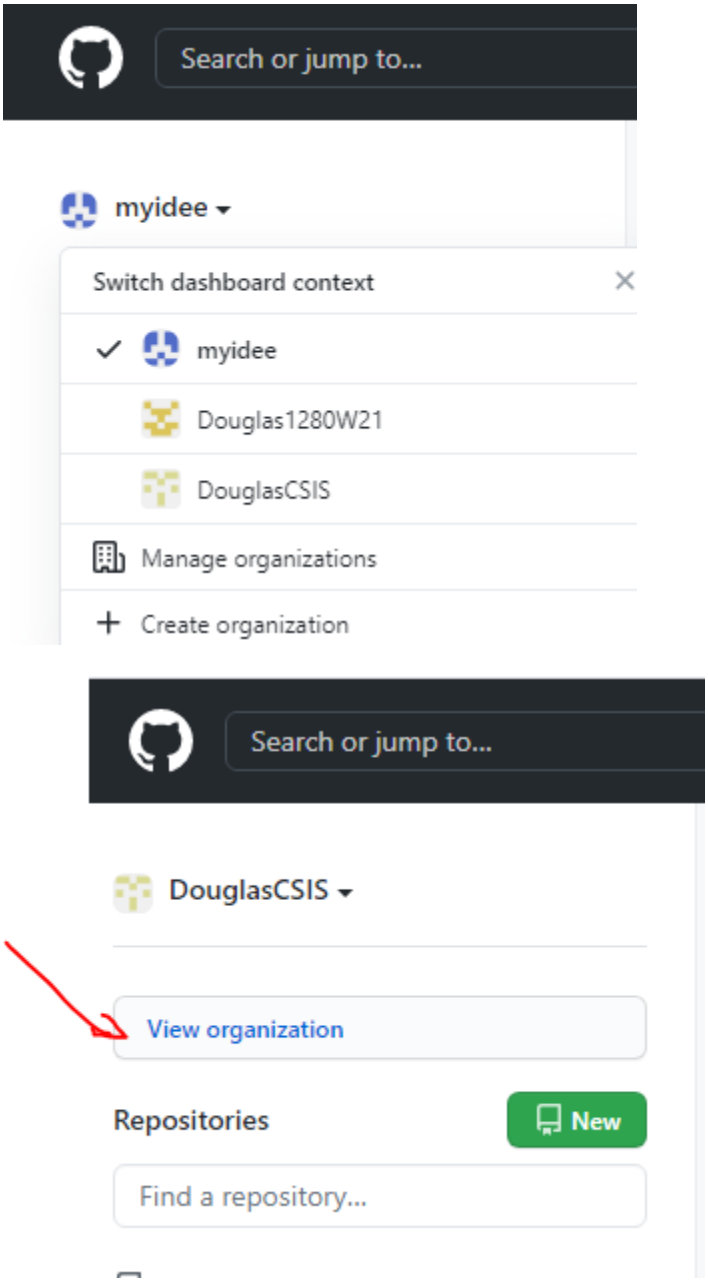

Using Teams




And maybe other individuals on the same projects or different



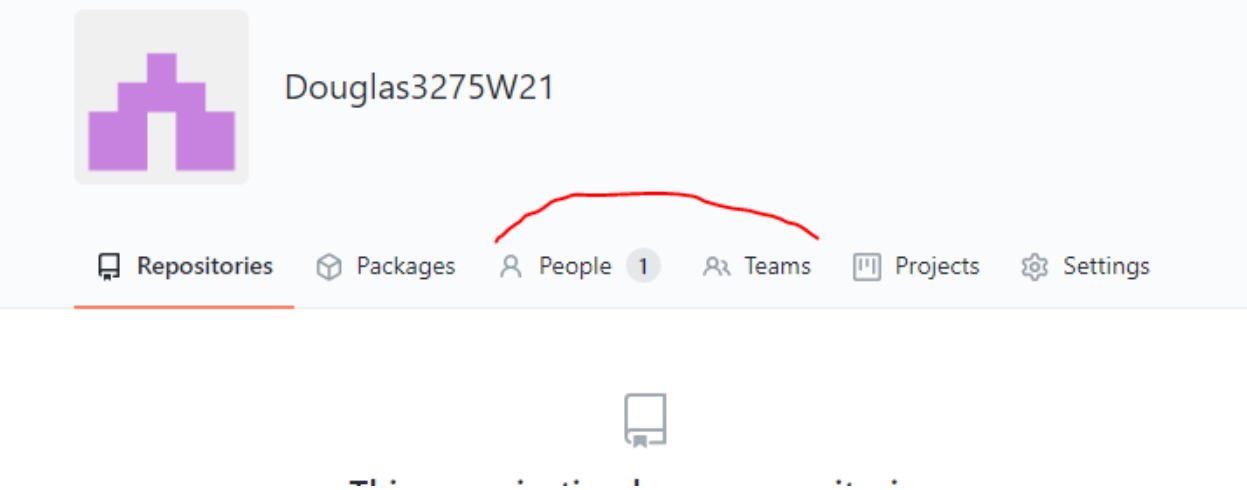

Use contact selector and choose the organization







We can add individuals or teams



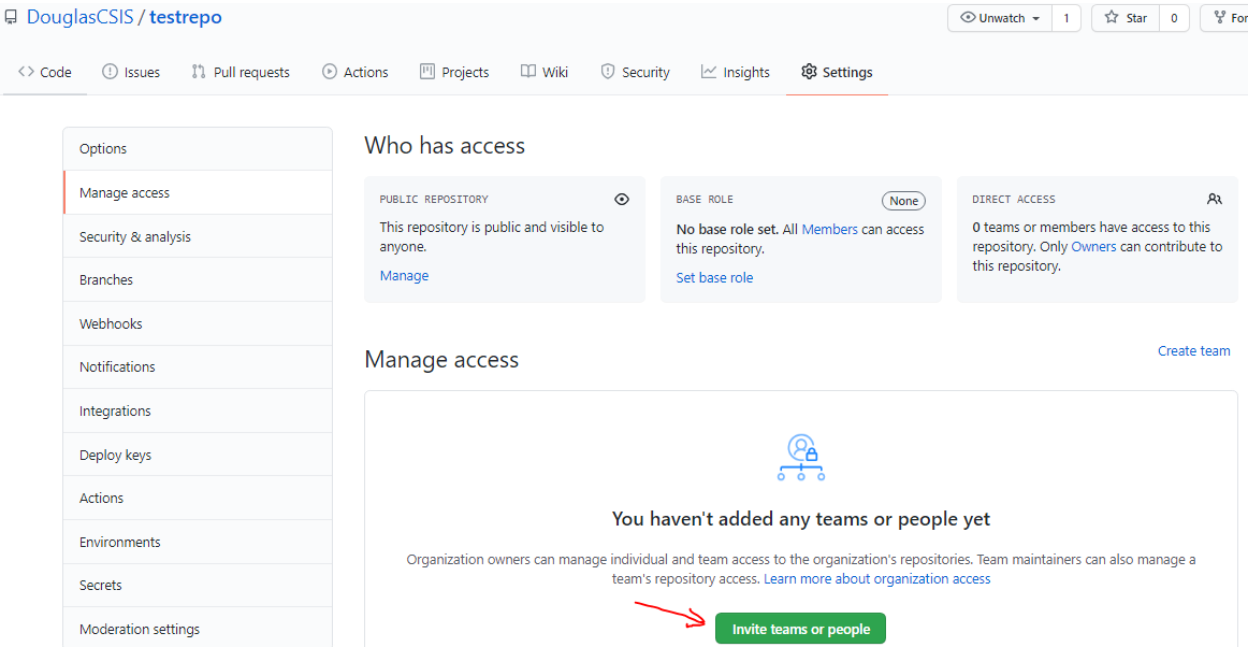

Starting with teams… we need to create a new team for the first time.

Visible: make the team visible,

Secret: make the team only visible to only the team

Creating repository for the organization is similar process to what was done before

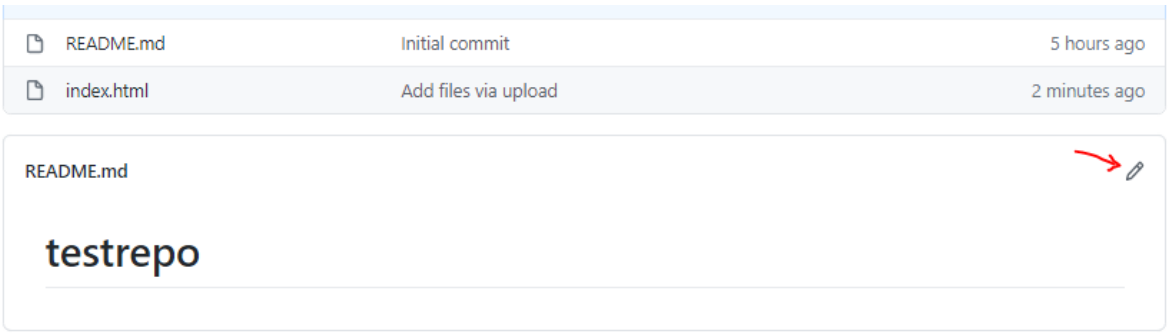
Collaborators and teams can be added in the **Settings**




Now you can open pull request for your team to review

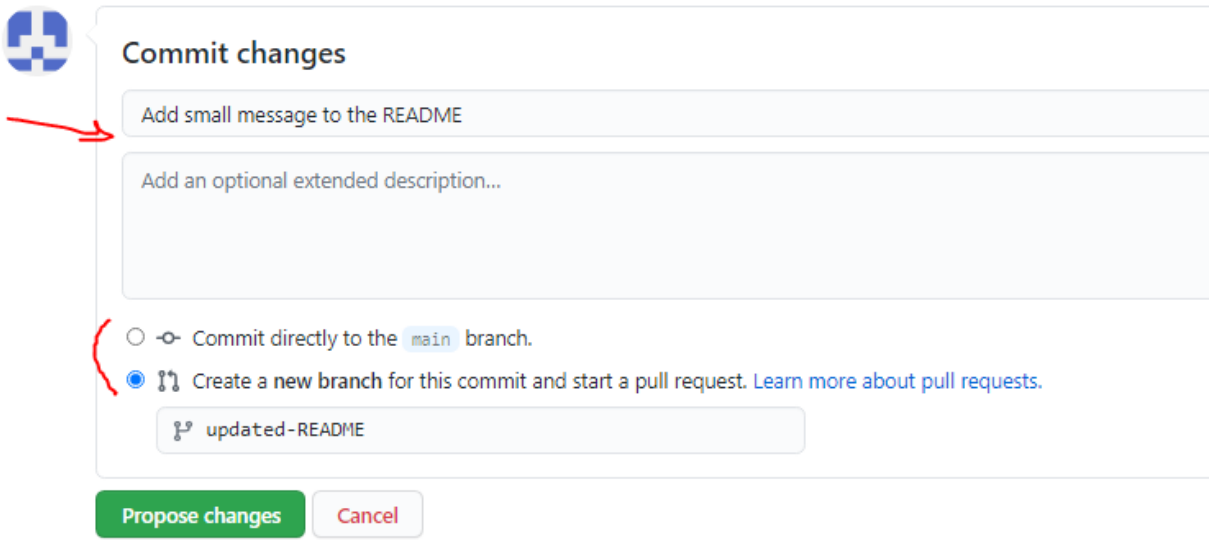
A well-built pull request is more important for larger organizations and teams

First for the repo, first you want to post some description in README.md

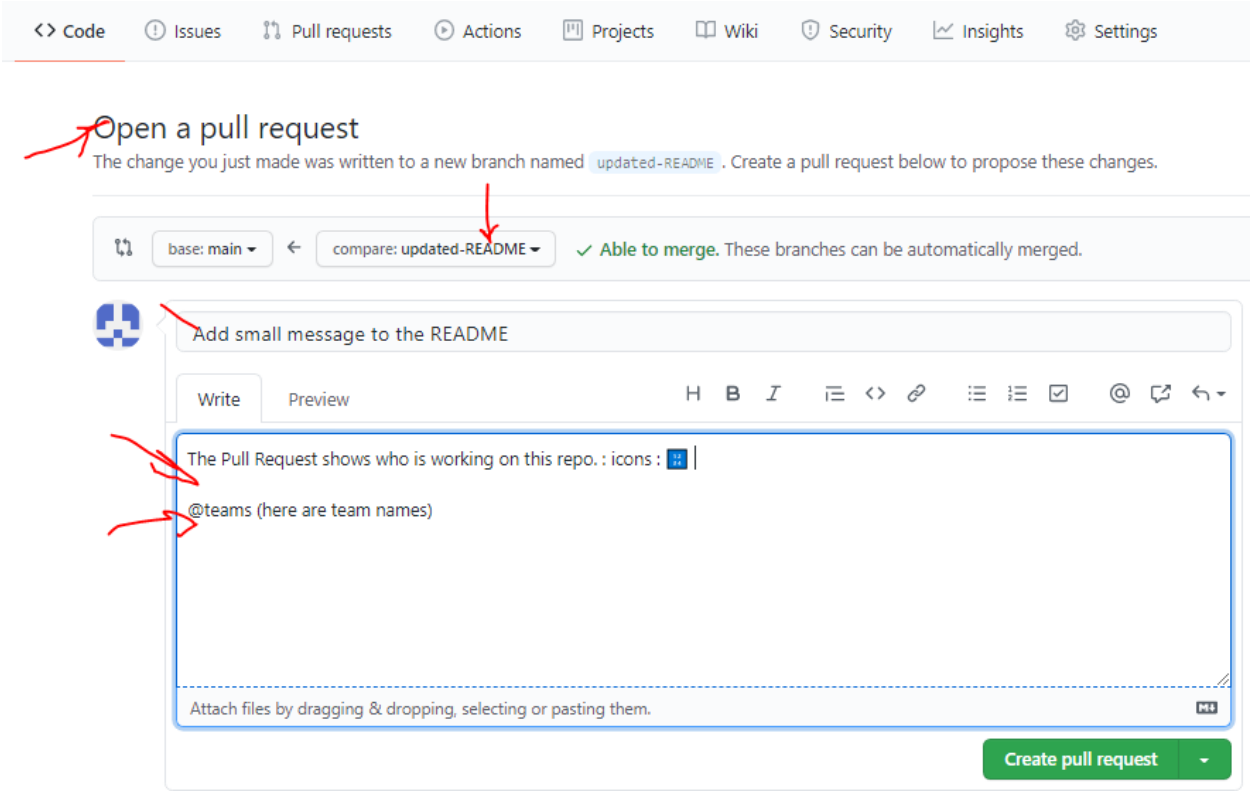



Use the pen and make some notifications.

You will have an option to commit it to existing branch or make a new branch.



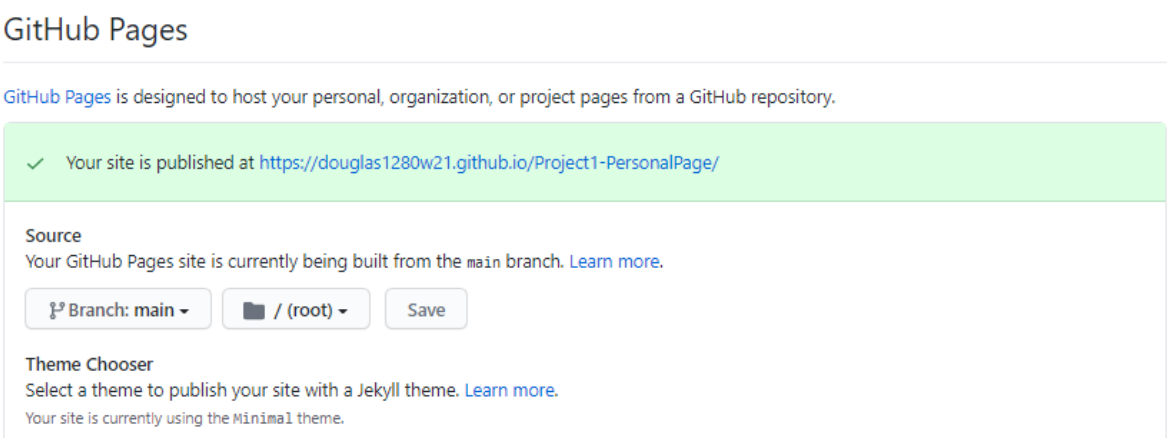

The following screen will appear.




Now you can pull a request. When working in large groups a proper pull request is critical, so first think why you want to create a pull request.

**Github Pages**

You can create a home page or project page that can be hosted for free by github.




Summary of git commands for PR on Git Bash

1- git branch new-branch
2- git branch
3- git checkout new-branch (git checkout -b new-branch to do branching and checkout)
4- make changes to a file
5- git add .
6- git commit -m "…"
7- git push origin new-branch (may need logging in again)