

# Homework 1

For this homework you will create a github repo, set up github pages, clone the repo to your computer as an R project, create a `.qmd` file, and push those changes back to github to create a webpage! You'll submit the link to your github pages site (the one that looks like a nice website).

Sounds like a lot. We'll get through it! (We're going to follow the process outlined for 'render to docs' [from here.](#))

## Step 1

- Head to github and create a new repo.
  - Be sure to make the repo public and **do not** choose a `.gitignore`

### Create a new repository

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


Required fields are marked with an asterisk (\*).

#### Repository template

No template ▾

Start your repository with a template repository's contents.

#### Owner \*

 jbpost2 ▾

#### Repository name \*

Homework1

✔ Homework1 is available.

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

#### 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.

#### Initialize this repository with:

☒ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

#### Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

#### Choose a license

License: Apache License 2.0 ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set `main` as the default branch. Change the default name in your [settings](#).

① You are creating a public repository in your personal account.

## Step 2

- Create a new R project from version control (as we did in the notes/videos) that clones this repository locally.
  - Recall you can click on the green button on the github.com repo website to copy the repo link.
  - A `.gitignore` file may be created in this process. That isn't a worry!

## Step 3

- Create a new `.qmd` document that outputs to HTML. You can give this a title about data science. Save the file in the main repo folder.
- In this document, we want to do two things (separate them with headers):
  - First, write text to answer the question:
    - \* What do you think being a data scientist is about?
    - \* What differences/similarities do you see between data scientists and statisticians?
    - \* How do you view yourself in relation to these two areas?
  - You can write in a conversational tone or more formally (however you want to represent yourself). **Use a markdown list at some point.** There is no word count or anything like that, just make sure you answer the prompts above to receive full credit.
  - Second, include a section with the following R code:

```
y <- density(iris$Petal.Width)
```

- \* Create an R code chunk to determine the class, type, and structure of the object `y`
    - \* Create an R code chunk that uses the `plot` function on `y`. Hide the R code in the final document by setting `echo` to `false`.
    - \* Include some markdown text between these code chunks explaining what you are doing with the R code.
  - You can render the document to check things are looking good.

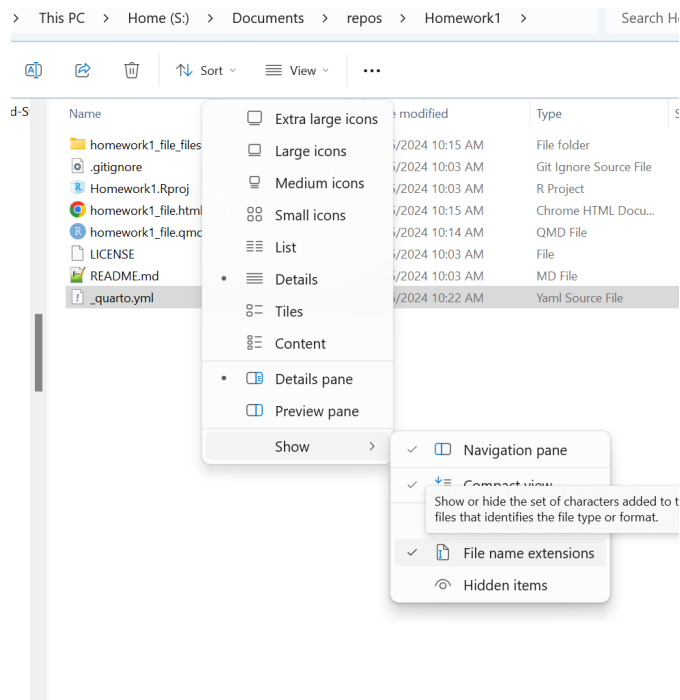
## Step 4

In your repo folder (locally), create a file called `_quarto.yml`. Open this file (perhaps in RStudio or a text editor) and place the following in the file (spacing is important!):

```
project:
  type: website
  output-dir: docs
```

To create the file, you can just navigate to the folder and create a new file.

- On windows, first make sure that file extensions show when you look at files in folders.



- Now right click in the folder area, select ‘New’ –> ‘Text Document’. Change the file name and file extension (the `.txt` part) to `_quarto.yml`. Now you can open the file in RStudio or a text editor and add the text.
- On a mac, you should be able to create a file using TextEdit in your R project folder. Save it as a `.doc` or whatever. Then you should be able to rename the file as `_quarto.yml`. Now you can open the file in RStudio or a text editor and add the text.

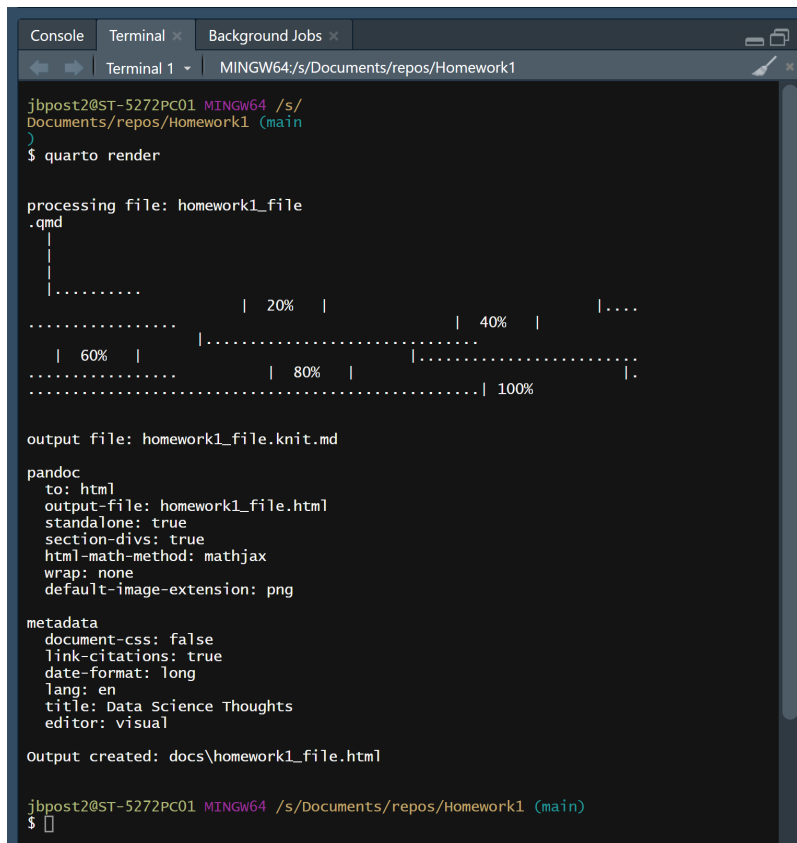
## Step 5

Now create a file called `.nojekyll` in your project repo. This file doesn’t need to have anything in it! You just need that file there (it may be a hidden file after you create it. Github should still track it.)

## Step 6

Open the terminal in RStudio and run the following code:

```
quarto render
```



```
jbpost2@ST-5272PC01 MINGW64 /s/
Documents/repos/Homework1 (main)
$ quarto render

processing file: homework1_file
.qmd
|
|.....
|.....| 20% |.....| 40% |.....|
|.....| 60% |.....| 80% |.....| 100%
|.....|

output file: homework1_file.knit.md

pandoc
to: html
output-file: homework1_file.html
standalone: true
section-divs: true
html-math-method: mathjax
wrap: none
default-image-extension: png

metadata
document-css: false
link-citations: true
date-format: long
lang: en
title: Data Science Thoughts
editor: visual

Output created: docs\homework1_file.html

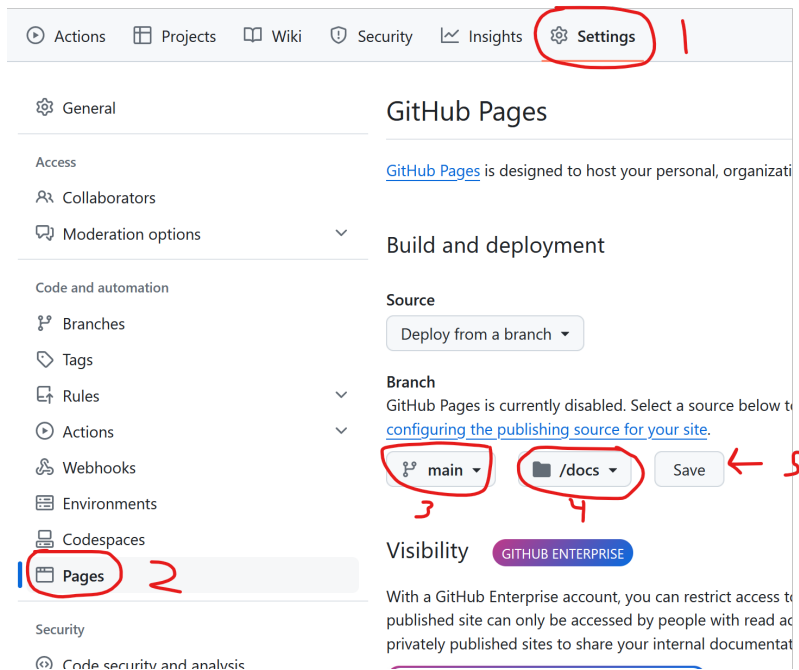
jbpost2@ST-5272PC01 MINGW64 /s/Documents/repos/Homework1 (main)
$
```

## Step 7

Push all changes up to your repo! You can do this via menus or the command line (or via the github web interface).

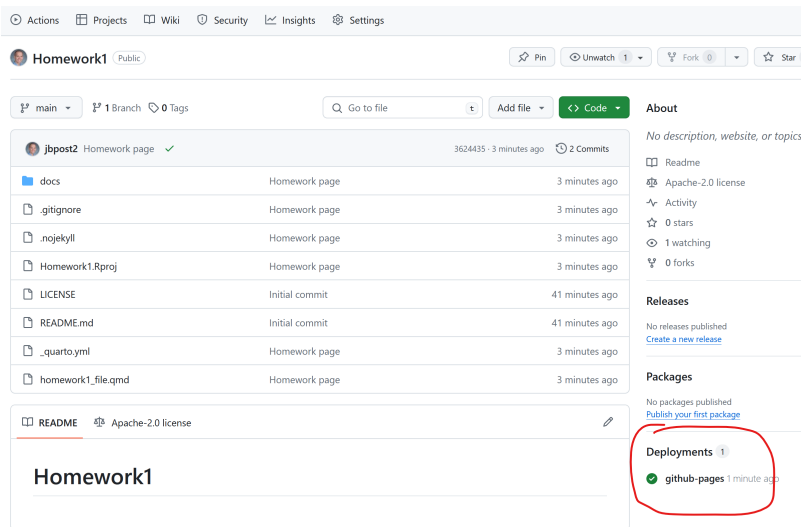
## Step 8

Head to your github repo page. Go to settings, choose pages, and under “Branch” choose ‘main’ and change the folder to /docs. Then hit save!



## Step 9

Wait about 2 minutes... Head back to your main github repo page. You'll now see a 'Deployments' section on the bottom right.



Click on that. Hopefully, after a minute you see a green check and something that says your site is ready!



## github-pages deployments

Latest deployments

 **github-pages**  
Last deployed 1 minute ago  
<https://jbpost2.github.io/Homework1/>

 Filter Filter deployments

1 deployments

 **Homework page** Active  
Deployed to github-pages by  jbpost2 via pages-build-deployment #1

Click on that and you should see a nicely rendered website! Copy the link to that site and that is what you'll turn in for this assignment!