

# Quarto and GitHub

## Setting up Quarto's Integration with RStudio

Begin by going to the Quarto webpage: <https://quarto.org> . From there, download the version for your operating system.

Once downloaded, go into your RStudio, then to preferences, then to the RMarkdown tab. Once there you can enable Quarto.

Once Quarto is enabled, you can create new Quarto documents by going to File, New File, and then Quarto Document.

## Using Quarto

Once installed, Quarto can be used similarly to RMarkdown.

## Getting Started with Git

Begin by ensuring you have XCode command line tools enabled.

Go to the Terminal and enter

```
git --version
```

If it is installed already, it will say so. If not, accept the offer to install.

```
## install if needed (do this exactly once):  
## install.packages("usethis")  
  
library(usethis)  
use_git_config(user.name = "Jordan Dewar", user.email = "jordanldewar@gmail.com")
```

## Getting Started with Github

Begin by making a Github account if you don't already have one:

[https://github.com/signup?ref\\_cta=Sign+up&ref\\_loc=header+logged+out&ref\\_page=%2F&source=header-home](https://github.com/signup?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F&source=header-home)

In R, enter:

```
#This is to create a Github token
usethis::create_github_token()

#Then
gitcreds::gitcreds_set()

#When prompted, enter your token
```

Begin by making a repo on Github or follow along using mine.

Make the repo public, go to the section labeled “code”, and copy the HTTPS URL at the top. This is the URL for mine: <https://github.com/J-Dewar/729B.git>

Enter RStudio, select Terminal, and enter

```
git clone “your repository”
```

or

```
git clone https://github.com/J-Dewar/729B.git
```

My repository for this is 729B, which will be the example I use moving forward.

I can change my working directory to my Github repo using the command: “cd 729B” in the terminal.

## Pushing Changes

Let's start by making a simple change. We'll add the line “This is the repository for the GVPT class 729B” to the README file.

In the terminal, I'll type: `echo “This is the repository for the GVPT class 729B” » README.md`

To check to see if the change worked, I'll type “git status” in the terminal.

It will then show:

```
jordandewar@wireless-10-105-58-3 729B % echo "This is the repository for the GVPT class 729B" » README.md
jordandewar@wireless-10-105-58-3 729B % git status
On branch master
```

No commits yet

Untracked files: (use "git add ..." to include in what will be committed) README.md

nothing added to commit but untracked files present (use "git add" to track)

To add the change, I use the command "git add README.md" followed by "git commit -m" A commit from my local computer" The comment allows me to see who added the change and from where. Finally, I enter "git push" to push the change.

This is what it then looks like on my Github page:

