

my_book_again

Kevin Williams

2025-09-22

Table of contents

Preface	4
1 Introduction	5
1.1 more	5
1.1.1 more a	5
1.1.2 more b	5
1.1.3 more c	5
1.2 less	5
1.3 true	5
1.3.1 ture z	5
1.3.2 ffg	5
1.4 qwgwe=====	5
 I Part 1	 6
2 Meat	7
2.1 this is the meat of things	7
2.1.1 this is not the meat of things	7
3 why_me	8
3.1 getting harder b bb	8
 II Projects	 9
4 Projects - Creating Projects2	10
4.0.1 Creating a New Project	10
4.0.2 Setting Up Version Control with Git	10
4.0.3 add starting files	10
4.0.4 add data folder	11
4.0.5 fixing and updating README.md file with github info	11
5 Projects - Hex2	12

6 Projects - OpenAI	13
6.1 use ellmer package	13
6.2 get prompt working with openai	13
7 RRL.com Stuff	14
7.0.1 As of 20251010	14
 III Part 2	 15
8 Concepts	16
8.1 data.table vs DT	16
8.2 Link to kaggle datasets to downloaded data directly into R using the kaggle API	16
8.3 Leaflet	16
9 Summary	18
References	19
Errors	20
Github	20

Preface

This is a Quarto book.

To learn more about Quarto books visit <https://quarto.org/docs/books>.

1 + 1

[1] 2

1 Introduction

This is a book created from markdown and executable code.

See Knuth (1984) for additional discussion of literate programming.

```
1 + 1
```

```
[1] 2
```

1.1 more

1.1.1 more a

1.1.2 more b

1.1.3 more c

1.2 less

1.3 true

1.3.1 ture z

true za

1.3.2 ffg

1.4 qwgwe=====

Part I

Part 1

2 Meat

2.1 this is the meat of things

2.1.1 this is not the meat of things

3 why_me

3.1 getting harder b bb

Part II

Projects

4 Projects - Creating Projects2

4.0.1 Creating a New Project

To create a new project, follow these steps: Make this your working directory: `~/Documents/r-studio-and-git`

In console: `library(usethis)`

`usethis::create_project("~/Documents/r-studio-and-git/YOUR-PROJECT-NAME")` This will create a new directory for your project and open it in RStudio. It will also create a default R project file (`.Rproj`) for you. It will also create a readme file for you. It will also create a `.gitignore` file for you.

4.0.2 Setting Up Version Control with Git

To set up version control with Git for your new project, follow these steps: In console: `usethis::use_git()` This will initialize a Git repository in your project directory. It will also create a `.git` directory to track changes. It will also make an initial commit with the existing files in your project. **### Connecting to GitHub** To connect your local Git repository to a remote repository on GitHub, follow these steps: First, create a new repository on GitHub without a README, `.gitignore`, or license. Then, in console: `usethis::use_github()` This will create a new repository on GitHub and link it to your local Git repository. It will also push your initial commit to the remote repository on GitHub. **### Summary** You have now created a new R project, set up version control with Git, and connected it to a remote repository on GitHub. You can now start working on your project and use Git to track changes and collaborate with others.

4.0.3 add starting files

`usethis::use_r("import")` # for R scripts under "R" folder `usethis::use_r("tidy")` # for R scripts under "R" folder

4.0.4 add data folder

```
usethis::use_data_raw() # for raw data under "data-raw" usethis::use_directory("DATA") #  
for processed data under "data"
```

4.0.5 fixing and updating README.md file with github info

When README.md file was first created, it does not render properly on GitHub.

To fix this, follow these steps:

1. Open README.md file in RStudio.

2. Add the following YAML header at the top of the file:

```
— title: "Project Title" #optional author: "Your Name" # optional date: "2024-06-10" #op-  
tional output: github_document — 3. Save the file. 4. In console, run the following command  
to render the README.md file: rmarkdown::render("README.md") 5. Commit and push  
the changes to GitHub. This will update the README.md file to render properly on GitHub  
with the specified title, author, and date.
```

You will need to render the README.md file each time you make changes to it.

5 Projects - Hex2

- Get Hexwall working
- Get Hexwall working with maps
- Make a RRL Hex with Shiny App
- Make a RRL Hex with hexSticker package
- Get a Shiny Hexwall working
- Get a Shiny Hexwall working with maps
- Get a Shiny Hexwall working with multiple map choices
- Get an updated Hexwall working with an input of a hex

6 Projects - OpenAI

6.1 use ellmer package

```
library(ellmer)
#chat <- chat_openai()
```

```
Sys.setenv(OPENAI_API_KEY = '<your key here')
```

```
#> Using model = "gpt-4.1". chat$chat(" What is the difference between a tibble and a data  
frame? Answer with a bulleted list ")
```

```
chat$chat("Who created R?", echo = FALSE)
```

6.2 get prompt working with openai

7 RRL.com Stuff

7.0.1 As of 20251010

www.wordpress.com account to use is: User: kwill1992rrl which links to kwill1992@hotmail.com There is another one (at least one other), which is User: kwill1992 and links to kevin.williams@rrldataanalytics.com The kwill1992 one does not have any websites linked to it. The kwill1992rrl account has both the old one @ www.rrldataanalytics.com and the one in development at the numbered IP address.

To get to the linked sites at www.wordpress.com, go to “managed blogs” button and this will come up: <https://wordpress.com/sites>

Only one is default site at any given time. You still need to login with user account to edit each website. Each website has a “user” and “kwill1992” user. Different passwords for each for each website.

Part III

Part 2

8 Concepts

8.1 data.table vs DT

The `data.table` package is a powerful R package for data manipulation and aggregation, known for its speed and efficiency with large datasets. It provides an enhanced version of data frames with additional features like fast grouping, indexing, and in-place updates.

On the other hand, `DT` is an R package that provides an interface to the JavaScript library `DataTables`. It is primarily used for creating interactive tables in R Markdown documents and Shiny applications. `DT` allows users to create sortable, searchable, and paginated tables with ease, enhancing the user experience when working with tabular data in web applications.

for a quarto document and output as a website, use `DT` for a data table.

8.2 Link to kaggle datasets to download data directly into R using the kaggle API

```
library(RKaggle) # for interacting with Kaggle API
```

using `RKaggle` to download the dataset from Kaggle

Example:

```
superstore_dataset <- get_dataset("vivek468/superstore-dataset-final")
```

github for `RKaggle`:

<https://github.com/benjaminndsmith/RKaggle>

8.3 Leaflet

The `leaflet` package in R is a powerful tool for creating interactive maps. It allows users to visualize spatial data with various layers, markers, and pop-ups. The package is built on top of the `Leaflet` JavaScript library, providing a user-friendly interface for R users to create dynamic maps that can be embedded in R Markdown documents, Shiny applications, or viewed in RStudio. With `leaflet`, users can easily add tiles, polygons, and other geographical features to their maps, making it a versatile choice for geospatial data visualization.

<https://rstudio.github.io/leaflet/index.html>
<https://r-charts.com/spatial/interactive-maps-leaflet/>
<https://r-graph-gallery.com/package/leaflet.html>
https://bookdown.org/nicohahn/making__maps__with__r5/docs/leaflet.html <https://leafletjs.com/reference.html>
<https://www.geeksforgeeks.org/r-language/leaflet-package-in-r/>
<https://www.jla-data.net/eng/leaflet-in-r-tips-and-tricks/>

9 Summary

In summary, this book has no content whatsoever.

1 + 1

[1] 2

References

Knuth, Donald E. 1984. “Literate Programming.” *Comput. J.* 27 (2): 97–111. <https://doi.org/10.1093/comjnl/27.2.97>.

Errors

Github

- to fix a git push commit holding up
 - `rm .git/index.lock` [in Terminal]