

# R Markdown and GitHub Tips

Code ▼

Instructor - Soumya Mukherjee

Content Credit- Dr. Matthew Beckman and Olivia Beck

Aug 1, 2023

## Overview

- Absolute vs. relative file paths
- Outputting data in RMarkdown
- Interactive functions in R: when to use them and when to not

## Absolute vs. Realative File Paths

The goal of a GitHub repo is to be entirely self contained. - **self contained**: everything needed for someone else to reproduce every single thing in your GitHub repo is inside your repo. Nothing in the repo references anything outside of the repo - We need to ensure all data is loaded in referencing only the file structure inside the repo, and nothing outside of it (i.e. do no reference where the repo is located on your computer)

When we open an RProject, R automatically sets our current working directory. We can see our current working directory with the `getwd()` function.

Hide

```
getwd()
```

```
[1] "C:/Users/SMukherjee/Documents/STAT-184-course-materials/Day 20 notes"
```

When we want to load a file into our R environment there are two ways to do it

1. Absolute paths - reference the exact file location on your computer
2. Relative paths - reference the file location with respect to your current working directory.

If I want to load in the Harry Potter Characters list (located in the Day 20 folder) I can ....

1. Reference the exact location on my computer

[Hide](#)

```
path.absolute <- "C:\\Users\\SMukherjee\\Documents\\STAT-184-course-materials\\Day 20 notes\\Characters.csv"
harry.potter <- read_delim(path.absolute,
  delim = ";", escape_double = FALSE, trim_ws = TRUE)
```

```
Rows: 140 Columns: 15— Column specification —————
Delimiter: ";"
chr (14): Name, Gender, Job, House, Wand, Patronus, Species, Blood status, Hair colour, Eye ...
dbl (1): Id
i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

2. Reference the file location with respect to my current working directory

[Hide](#)

```
path.relative <- "Characters.csv"
harry.potter <- read_delim(path.relative,
  delim = ";", escape_double = FALSE, trim_ws = TRUE)
```

- When you use absolute file paths, you will always reference the exact file you want, BUT when you push to GitHub, no else will be able to reference this path.
- When you use relative file paths, you need to ensure your working directory is correct, but when you push to GitHub, everyone will be able to reference the file path.
- Read more about file paths in R here: <https://ytakemon.github.io/2019-10-22-R-BCCRC/02-filedir/> (<https://ytakemon.github.io/2019-10-22-R-BCCRC/02-filedir/>)

## Outputting data in RMarkdown

When we output data in RMarkdown, we want to output tables that the user can acutally read

Let's look at the Harry Potter Data set.

[Hide](#)

```
harry.potter <- read_delim("Characters.csv",  
  delim = ";", escape_double = FALSE, trim_ws = TRUE)
```

```
harry.potter <- as.data.frame(harry.potter)
```

```
harry.potter
```

Id	Name	Gender	
<dbl>	<chr>	<chr>	
1	Harry James Potter	Male	
2	Ronald Bilius Weasley	Male	
3	Hermione Jean Granger	Female	
4	Albus Percival Wulfric Brian Dumbledore	Male	
5	Rubeus Hagrid	Male	
6	Neville Longbottom	Male	
7	Fred Weasley	Male	
8	George Weasley	Male	
9	Ginevra (Ginny) Molly Weasley	Female	
10	Dean Thomas	Male	
1-10 of 140 rows   1-3 of 15 columns			
Previous 1 2 3 4 5 6 ... 14 Next			

This is A LOT of data. You can technically read all every entry in this table. But no one is going to do that!

When introducing the reader to the data structure, we only want to show them enough for them to get an idea of what the data looks like. We don't need to show them the entire thing. Use `head`, `tail`, `sample_n`, etc.

Hide

```
harry.potter %>%
  sample_n(8)
```

<b>Id</b>	<b>Name</b>	<b>Gender</b>
<dbl>	<chr>	<chr>
49	Gregory Goyle	Male
61	Millicent Bulstrode	Female
27	Romilda Vane	Female
83	Newton Scamander	Male
32	Arthur Weasley	Male
57	Blaise Zabini	Male
116	Benjy Fenwick	Male
46	Severus Snape	Male

8 rows | 1-3 of 15 columns

This still isn't very readable. Let's only select a few interesting columns.

Hide

```
harry.potter %>%
  sample_n(8) %>%
  select(Name, Job, House )
```

<b>Name</b>	<b>Job</b>	<b>House</b>
<chr>	<chr>	<chr>
Benjy Fenwick	NA	NA
Barty Crouch Sr.	Head of the Department of International Magical Cooperation	NA

Name <chr>	Job <chr>	House <chr>
Parvati Patil	Student	Gryffindor
Regulus Arcturus Black	NA	Slytherin
Argus Filch	Caretaker of Hogwarts	NA
Cedric Diggory	Student	Hufflepuff
Hermione Jean Granger	Student	Gryffindor
Roger Davies	Student	Ravenclaw
8 rows		

This is a significant improvement! But we can still make it look nice, and not just code output using `kable`

Hide

```
library(knitr)
```

```
Warning: package 'knitr' was built under R version 4.2.3
```

Hide

```
#?kable
harry.potter %>%
  sample_n(8) %>%
  select(Name, Job, House ) %>%
  kable(caption = "Some Characters in Harry Potter")
```

```
Registered S3 methods overwritten by 'htmltools':
```

```
method          from
print.html       tools:rstudio
print.shiny.tag   tools:rstudio
print.shiny.tag.list tools:rstudio
```

## Some Characters in Harry Potter

Name	Job	House
Fred Weasley	Student	Gryffindor
Narcissa Malfoy	NA	Slytherin
Blaise Zabini	Student	Slytherin
George Weasley	Student	Gryffindor
Oliver Wood	Student	Gryffindor
Lily J. Potter	NA	Gryffindor
Albus Percival Wulfric Brian Dumbledore	Headmaster	Gryffindor
Dobby	Malfoy family's house-elf (? - 1993),	
Hogwarts kitchens worker (December 1994 - 1998)	NA	

Much nicer! We can use function in the package `kableExtra` to make fancy tables. This is a great resource: [https://cran.r-project.org/web/packages/kableExtra/vignettes/awesome\\_table\\_in\\_html.html](https://cran.r-project.org/web/packages/kableExtra/vignettes/awesome_table_in_html.html) ([https://cran.r-project.org/web/packages/kableExtra/vignettes/awesome\\_table\\_in\\_html.html](https://cran.r-project.org/web/packages/kableExtra/vignettes/awesome_table_in_html.html))

(You are not required to use `kable` for your final project.)

## Using interactive funtions

Functions like `view` and `?` don't actually run anything, but instead open a new window in RStudio.

- The point of an RMarkdown document is to make a report that someone can read without looking at your code in RStudio.
  - Everything included in the RMarkdown document needs to be rendered in the .html file.
  - **DO NOT USE `view` AND `?` in MARKDOWN DOCUMENTS**
    - These functions can not be seed in html files as they are only do things **inside** of Rstudio
    - Instead of `view` you can use `head`, `tail`, `str`, etc.
    - You should almost never be using `?` or `help` in a report. Help functions help you learn to code. They are not useful for your reader.

