R workshop activity md files from RStudio.

1. Create the workshop markdown (.md) file within R Studio.
   1. Open R Studio to a clean workspace
   2. Create a new .md file:
      1. File > New file > R Markdown

A screenshot of a computer

Description automatically generated

1. Set document metadata:
   1. Set the title as the Page title (the level 0 page header)
   2. Enter DSC as the Author.

A screenshot of a computer

Description automatically generated

1. Save the document
   1. File > Save As
   2. The filename will be the segment of the url that specifies the specific activity within the workshop website.
   3. Save to your uviclibraries/rstudio github repository.

A screenshot of a computer

Description automatically generated

NOTE : You will work in the Code editor for the bulk of the development.

* Work in the console will be only for testing purposes and importing libraries

1. Your file will appear pre-populated with metadata and ```{r setup}``` which is the code needed for knitting the file.  
   A screenshot of a computer

   Description automatically generated
2. Clear lines 12-end
3. In the Console, install the knitr package.
   1. ` install.packages(“knitr”) `
      1. Approve any dependencies to be installed when prompted
      2. More information: <https://community.rstudio.com/t/run-install-packages-knitr/78554>
4. To knit to a github-compatable md file, replace “output” with `github\_document` [https://rmarkdown.rstudio.com/github\_document\_format.htmlA close-up of a computer code

   Description automatically generated](https://rmarkdown.rstudio.com/github_document_format.html)
5. Save your file (cmd+S on Mac, ctrl+S on Windows)
6. Enter some test text on Line 12.
7. Under “Knit”, you will now see “Knit to github\_document.”  
   A screenshot of a computer

   Description automatically generated  
     
   11. Knit your file by clicking on “Knit to github\_document.” You will see a popup with a preview version of the document.  
     
   A screenshot of a computer

   Description automatically generated
8. You now can begin to build your file.

Commands:

* To enter code blocks, wrap the code in ```{r} code code code ```
  + The code block and output will appear in the markdown file (after being knit)
* To hide the code block, write `include=False`

```{r, include=FALSE} …

To hide the output, write `eval=FALSE`

```{r, eval=FALSE} …

To hide the code block but show the output, write `echo=False`

```{r, echo=FALSE} …

* Headers are notated by hash symbol (#) followed by a space and then the header name
  + # level 1
  + ## level 2
  + ### level 3
* In your setup code block (which begins with ` ```{r setup, include=FALSE}`), include any library() load commands you would typically run in the console.
  + Each time you knit the file, the library needs to be loaded. By including the library() load command in this block, the library can be used in the remainder of the knit.
  + Make sure you’ve already installed the library using install.packages() in your console. You’ll only need to do this once on your own device.
    - The install lives on your computer, not the file.
  + When you might use this: in a workshop teaching the ‘tidyverse’ library. You would include `library(tidyverse)` in the setup r code block.
* Hyperlinks are formatted as [display text] (url) {target open}
  + [Viewer Page](https://rstudio.github.io/rstudio- extensions/rstudio\_viewer.html){:target="\\_blank"}
  + The user will see “Viewer Page”, and when they click on it, [https://rstudio.github.io/rstudio- extensions/rstudio\_viewer.html](https://rstudio.github.io/rstudio-%20extensions/rstudio_viewer.html) will open in a new tab.
* To create unordered lists (bullet points), begin the line of text with `-`. Ensure there is a full space before the beginning of the list.

I am going to create a bullet list

* This is an item
* This is another item
* To create a block quote, begin the line of text with `>`
  + Multi-line quotes: begin each line with `>` and end each line with <br>

Code blocks will contain the commands you would typically enter into your standard code editor block or the console.

If your file uses an imported dataset, that will be done through a code block in your .Rmd file. You’ll need to know the correct file path. Ideally it will be in your Github repository. If you want to show the user the command for importing a dataset, you can omit “echo=FALSE”. To hide contents of the dataset, omit , “results='hide' “

```{r load-data, echo=FALSE, results='hide'}

income <- read\_excel("/Users/yourUserName/Documents/GitHub/dsc-workshop/income.xlsx")

```

After you have knit the file, you will see both the .Rmd file you created in RStudio as well as a .md file. The .md file will be the one you use in your Github page.  
A close up of black text

Description automatically generated

When you’re ready, make a commit on your Github repo, push it, and refresh your webpage page via Github.