R Markdown & Course Project

LEARNING OBJECTIVES

At the end of this group project, you should be able to:

- Produce a rich, fully-documented reproducible analysis using R Markdown.
- Put to work the knowledge and tools gained throughout this course.
- Experience using R tools on real life data sets.
- Self-direct your learning and interests to find unique and creative ways to explore data.

R MARKDOWN

- R Markdown introduces an easy way to produce fully-documented reproducible analysis.
- It allows the user to share a single file that contains all of the code and text needed to produce a nicely formatted HTML or PDF file.
- This is done without having knowledge of HTML or LaTeX and not having to fuss much about formatting.

R MARKDOWN

- One R Markdown file can generate a variety of different formats and all of this is done in a single text file with a few bits of formatting.
- Creating documents with R Markdown starts with an .Rmd file that contains a combination of text and R codes.
- The .Rmd file is fed to knitr, which executes all of the R code and creates a new markdown (.md) document with the output.

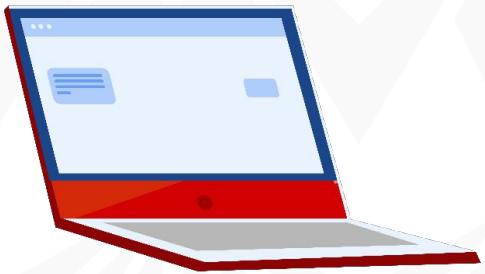
CREATING AN R MARKDOWN FILE

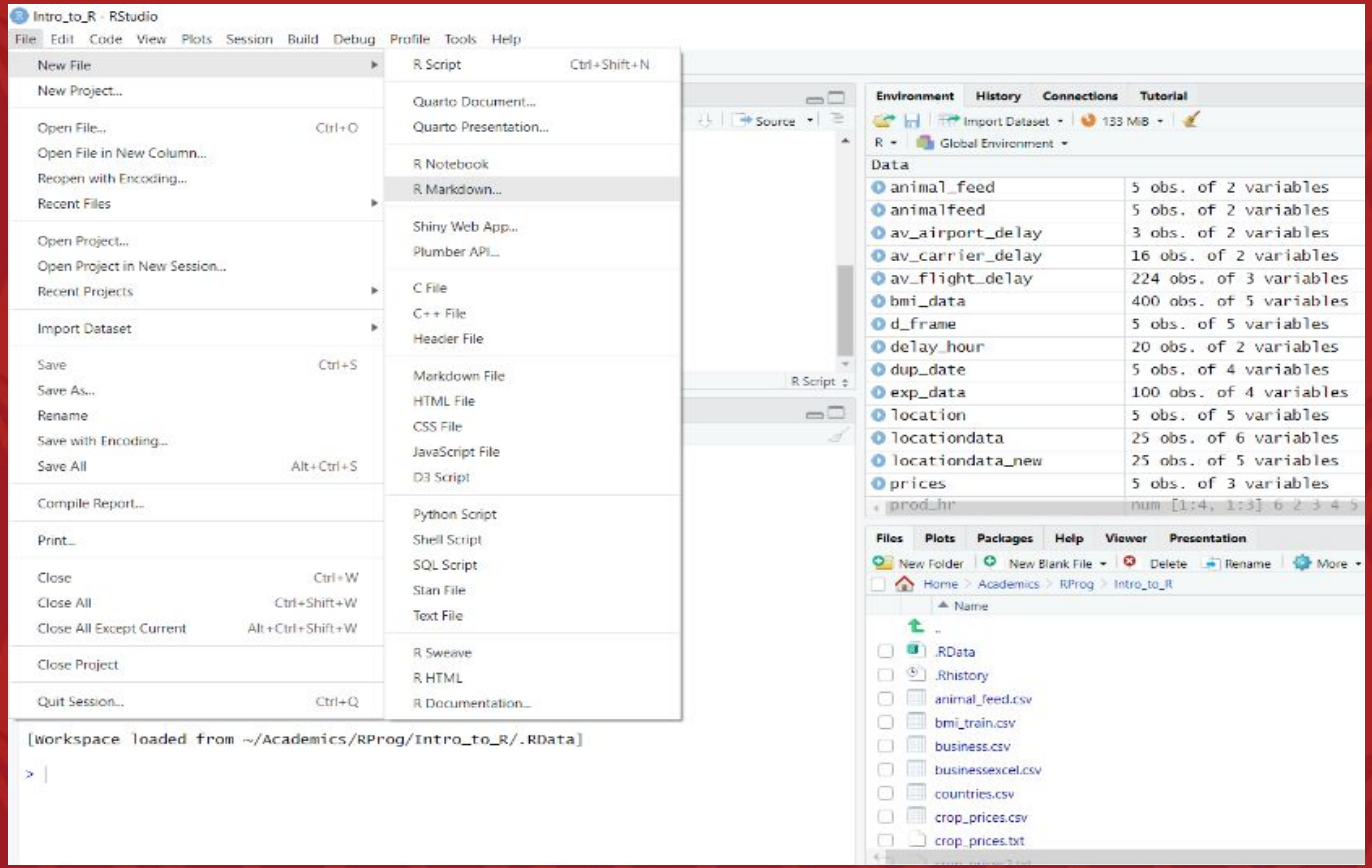
- Pandoc then renders the .md file to create a finished report in the form of a HTML, PDF, Word document, slide show, etc.
- This looks like lots of work and confusion, fortunately we don't have to this conversions as this takes place happens behind the scenes.
- You primarily need to worry only about the syntax required in the .Rmd file. You then press a button and out comes the report.

CREATING AN R MARKDOWN FILE

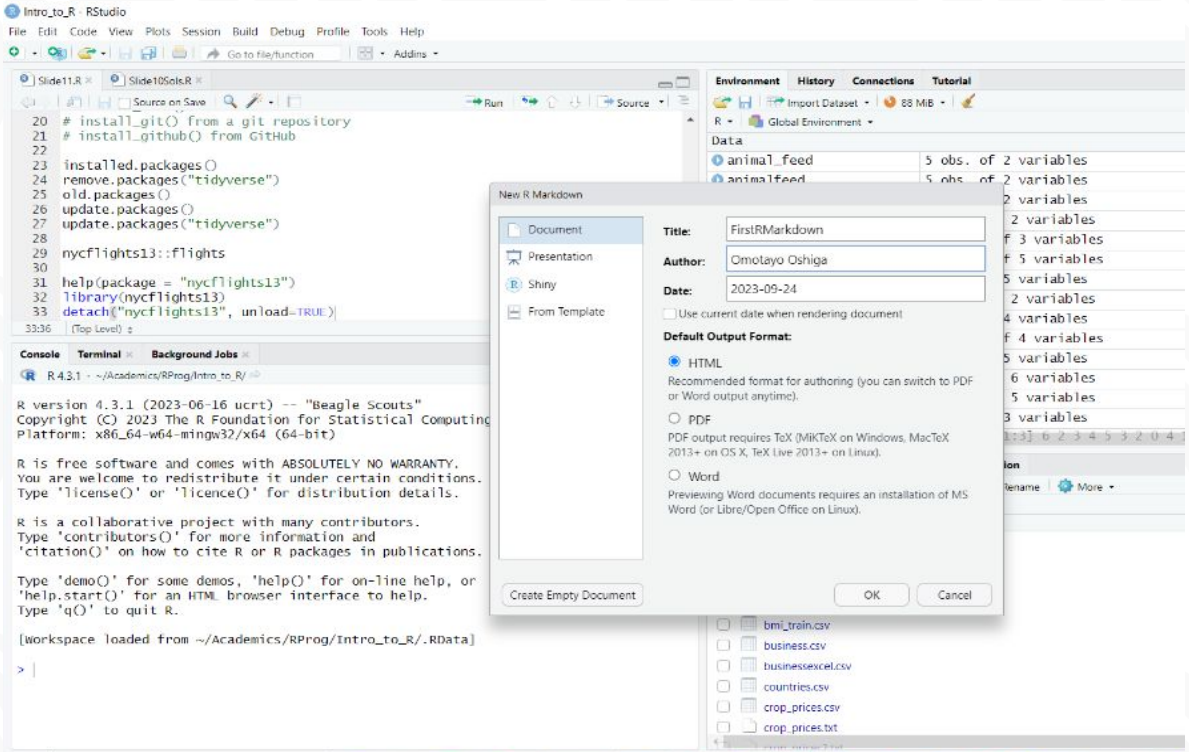
- To create an R Markdown file, select File → New File → R Markdown.

- The option to create an HTML, PDF, or Word document is given, however, R Markdown allows change seamlessly between these options after a document is created.




Intro_to_R - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
New File
New Project...
Open File... Ctrl+O
Open File in New Column...
Reopen with Encoding...
Recent Files
Open Project...
Open Project in New Session...
Recent Projects
Import Dataset
Save Ctrl+S
Save As...
Rename
Save with Encoding...
Save All Alt+Ctrl+S
Compile Report...
Print...
Close Ctrl+W
Close All Ctrl+Shift+W
Close All Except Current Alt+Ctrl+Shift+W
Close Project
Quit Session... Ctrl+Q
R Script Ctrl+Shift+N
Quarto Document...
Quarto Presentation...
R Notebook
R Markdown...
Shiny Web App...
Plumber API...
C File
C++ File
Header File
Markdown File
HTML File
CSS File
JavaScript File
D3 Script
Python Script
Shell Script
SQL Script
Stan File
Text File
R Sweave
R HTML
R Documentation...
[Workspace loaded from ~/Academics/RProg/Intro_to_R/.RData]
Environment History Connections Tutorial
Import Dataset 133 MiB
Global Environment
Data
animal_feed 5 obs. of 2 variables
animalfeed 5 obs. of 2 variables
av_airport_delay 3 obs. of 2 variables
av_carrier_delay 16 obs. of 2 variables
av_flight_delay 224 obs. of 3 variables
bmi_data 400 obs. of 5 variables
d_frame 5 obs. of 5 variables
delay_hour 20 obs. of 2 variables
dup_date 5 obs. of 4 variables
exp_data 100 obs. of 4 variables
location 5 obs. of 5 variables
locationdata 25 obs. of 6 variables
locationdata_new 25 obs. of 5 variables
prices 5 obs. of 3 variables
prod_hr num [1:4, 1:5] 6 2 3 4 5
Files Plots Packages Help Viewer Presentation
New Folder New Blank File Delete Rename More
Home > Academics > RProg > Intro_to_R
Name
.RData
.Rhistory
animal_feed.csv
bmi_train.csv
business.csv
businessexcel.csv
countries.csv
crop_prices.csv
crop_prices.txt

CREATING AN R MARKDOWN FILE

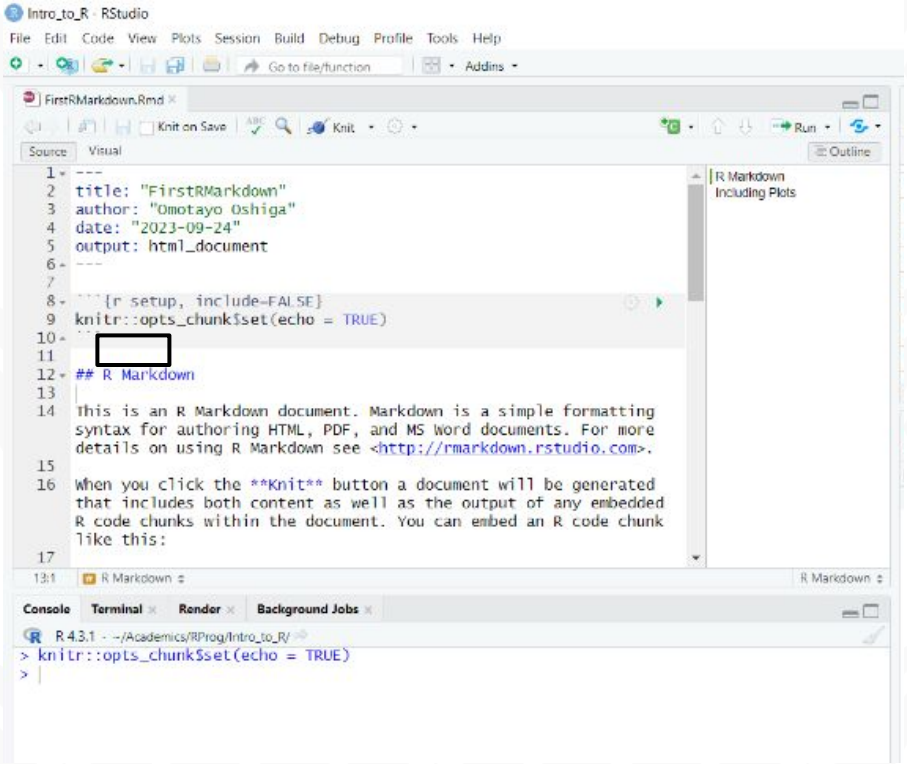


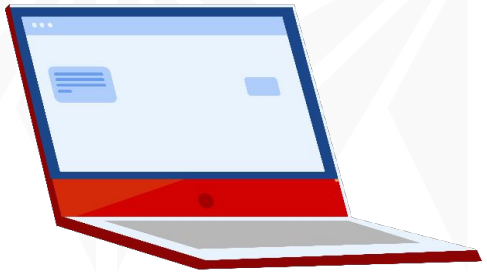
Additional options such as Presentations (HTML or PDF), Shiny documents, or other template documents can also be produced.

Knitting the R Markdown File

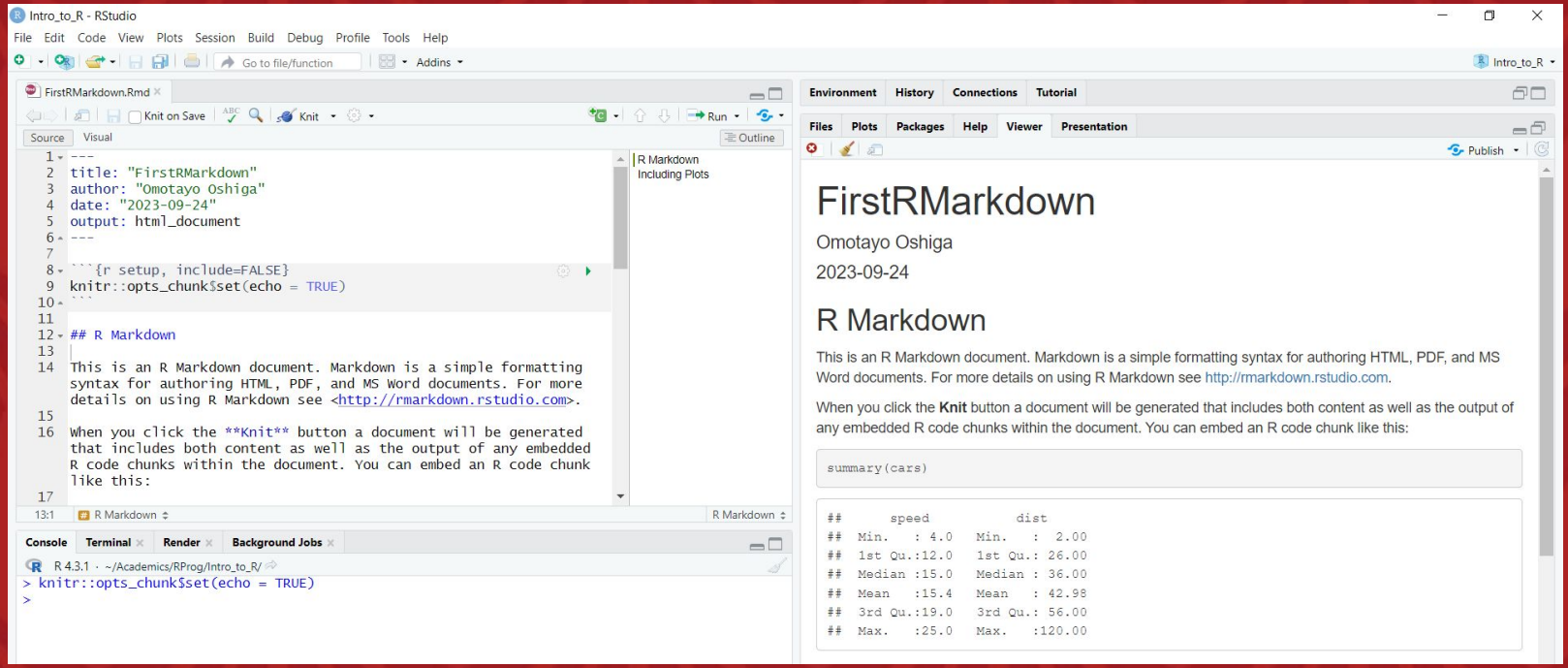
- Call the function in the console: render("document_name.Rmd", output_format = "html_document") or alternatively, the .rmd file can be converted by clicking on the drop down arrow next to the knit button highlighted below, select the document format (HTML, PDF, Word) and the report is then displayed on the viewer tab.

KNITTING THE R MARKDOWN FILE





Knitting the R Markdown File



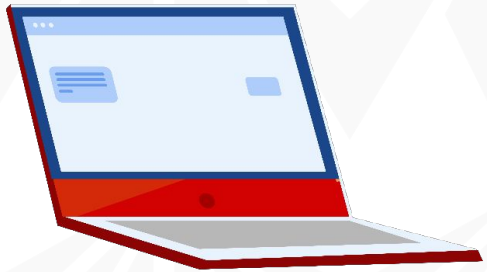
Components of an R Markdown File

YAML Header:

- The first few lines you see in the R Markdown report are the YAML.

```
---
title: "FirstRMarkdown"
author: "Omotayo Oshiga"
date: "2023-09-24"
output: html_document
---
```

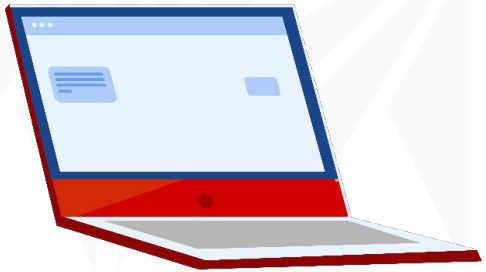
```
---
title: "FirstRMarkdown"
author: "[Omotayo Oshiga](www.omotayo.com)"
date: "`r Sys.Date()`"
output:
  html_document:
    toc: true
    keep_md: true
---
```



Components of an R Markdown File

- These lines will generate a generic heading at the top of the final report. Several YAML options can be used to enhance reports:
  A. hyperlinks can be used around the title or author name.
  B. The current date can be coded so that anytime the report is run the current date will print off.
  C. The date, author or title can be excluded by using NULL or delete the line.
  D. The report does not include a table of contents (TOC) by default.
  E. Beneficial to always keep a copy of the markdown file after rendering.

COMPONENTS OF AN R MARKDOWN FILE



Text Formatting:

- R Markdown has the ability to easily combine text and code. For the text component, it is similar to typing in a Word document; however, to perform basic text formatting, basic markdown code are used:

```
This is an R Markdown document. Markdown is a
simple formatting syntax for authoring HTML, PDF,
and MS Word documents. For more details on using R
Markdown see <http://rmarkdown.rstudio.com>.

Examples of basic markdown syntax

*italics*

**bold**

~~strike through~~

superscript\^2

[R Markdown link](http://rmarkdown.rstudio.com)
```

Components of an R Markdown File

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

Examples of basic markdown syntax

italics

bold

~~strike through~~

superscript^2

R Markdown link

TEXT FORMATTING

```
# Header 1

## Header 1

### Header 1

inline equation: $\mu = \sum{\dfrac{X}{N}}$

image: ![](xyplot.png)

-   unordered list

1.  ordered list
2.  item 2
    -   sub-list 1

    -   sub-list 2
```

- unordered list

1. ordered list
2. item 2
    - sub-list 1
    - sub-list 2

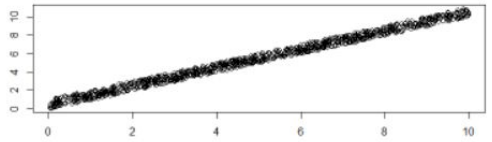
Header 1

Header 1

Header 1

inline equation: $\mu = \sum \frac{X}{N}$

image:



- There are many additional formatting options which can be viewed here.

Components of an R Markdown File

Code Chunks:

- R code chunks can be used to render R output into documents or display code for illustration.
- Code chunks start with: ```{r chunk_name} and end with ```. They are easily inserted into R Markdown with the shortcut Ctrl + Alt + I.
- Naming a chunk is optional, but recommended. Each chunk name must be unique, and only contain alphanumeric characters and _:

COMPONENTS OF AN R MARKDOWN FILE

- Let's load the tidyverse package and Titanic.csv file, we will insert a chunk at the start and call it '**setup**'. Since we don't want this code or its output to show in the HTML document, we add an include = FALSE option after the chunk name ({r setup, include = FALSE}).

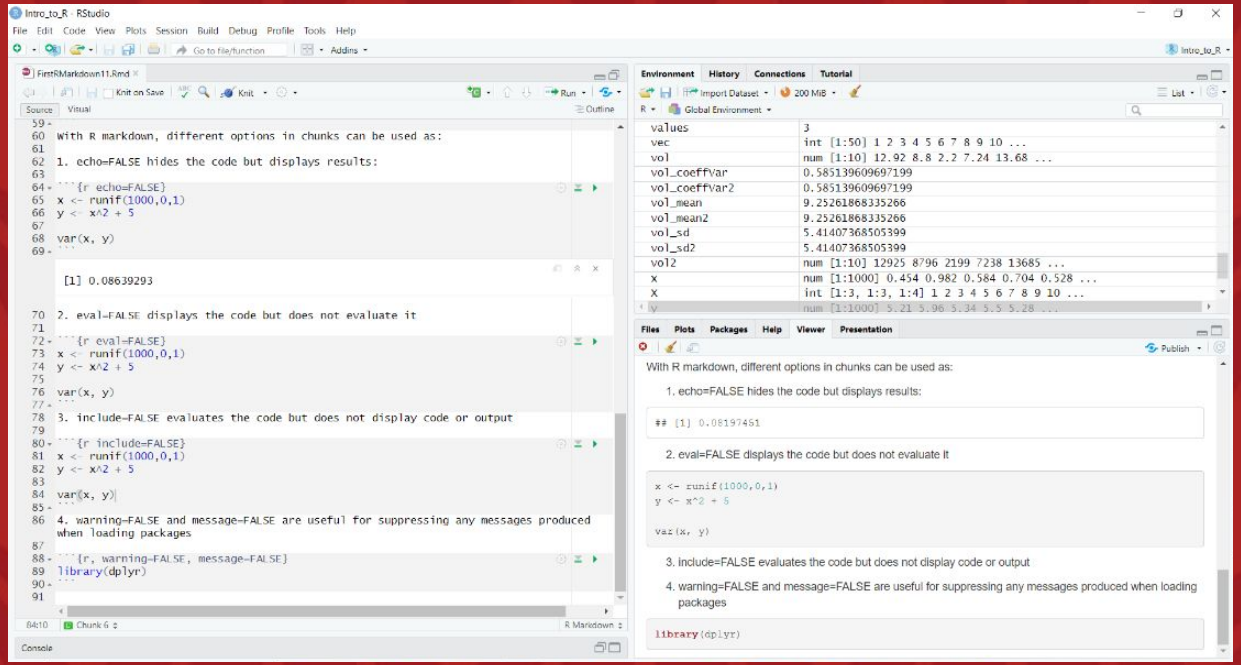
```
```{r setup, include=FALSE}
library(tidyverse)
Titanicdata <- read.csv("Titanic.csv")
knitr::opts_chunk$set(echo = TRUE)
```
```

CUSTOMISING CHUNK OUTPUT

- We can use include = FALSE in a code chunk to prevent the code and output from printing in the knitted document. There are additional options available to customise how the code-chunks are presented. The options are entered in the code chunk after the chunk_name and separated by commas {r chunk_name, eval = FALSE, echo = TRUE, include = FALSE}:

CUSTOMISING CHUNK OUTPUT

Option	Options	Output
`eval`	`TRUE` or `FALSE`	Whether or not the code within the code chunk should be run.
`echo`	`TRUE` or `FALSE`	Choose if you want to show your code chunk in the output document. `echo = TRUE` will show the code chunk.
`include`	`TRUE` or `FALSE`	Choose if the output of a code chunk should be included in the document. `FALSE` means that your code will run, but will not show up in the document.
`warning`	`TRUE` or `FALSE`	Whether or not you want your output document to display potential warning messages produced by your code.
`message`	`TRUE` or `FALSE`	Whether or not you want your output document to display potential messages produced by your code.
`fig.align`	`default`, `left`, `right`, `center`	Where the figure from your R code chunk should be output on the page

The default settings for chunk options are all TRUE. The default settings can be set for all chunak in the document by entering: knitr::opts_chunk$set(echo = TRUE) - this will change the default value of echo to TRUE for every code.

MID-LESSON QUESTIONS

What happens if you use eval = FALSE and echo = FALSE? What is the difference between this and include = FALSE?

Create a chunk with {r eval = FALSE, echo = FALSE},

```
```{r eval = FALSE, echo = FALSE}
head(Titanicdata)
```
```

then create another chunk with {r include = FALSE} to compare.

```
```{r include = FALSE}
head(Titanicdata)
```
```

MID-LESSON QUESTIONS

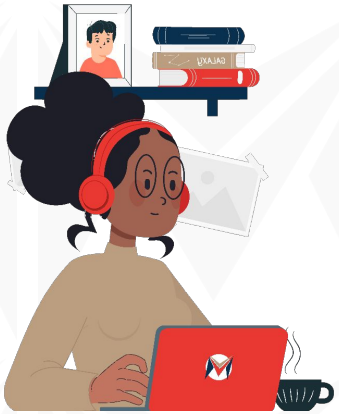
eval = FALSE and echo = FALSE will neither run the code in the chunk, nor show the code in the knitted document. The code chunk essentially doesn't exist in the knitted document as it was never run. Whereas include = FALSE will run the code and store the output for later use.

MID-LESSON QUESTIONS

Play around with the different options in the chunk with the code for the table, and re-Knit to see what each option does to the output.



IN-LINE R CODE



- Let look at some in-line R code in presenting descriptive statistics.
- We use a similar method as used for code chunk, the only difference is the number of backticks.
- In-line R code uses one backtick (`` `r` ``), whereas code chunks use three backticks (```` ```r``` ````).
- For example, today's date is `` `r Sys.Date()` ``, will be knitted as: today's date is 24-09-2023. Technically, the code will display the date the document was last knitted.
- The best way to use in-line R code, is to minimise the amount of code you need to produce the in-line output by preparing the output in code chunks previously.

COURSE PROJECT

- The aim of the course project is demonstrate proficiency in the techniques covered in this class and apply them to a novel dataset in a meaningful way.
- To achieve this, students would have to import a real life data set, assess, clean and tidy the data, and perform basic exploratory data analysis; all while using R Markdown to produce an HTML report that is fully reproducible.
- Students report are expected to tell a story with the data and provide a coherent explanation of their findings.
- The project is open ended and there is no limit on what tools or packages to use.

PROJECT DATA

- The final project for this course will consist of the statistical analysis of any of the provided or downloaded datasets.
- All data sets contain key attributes that will demonstrate the capabilities learned in this course.
- Students will also learn new skills not taught to accomplish this projects.
- These include working with:

A. multiple data types (numerics, characters, dates, factors)
B. unclean data (missing values)
C. variables that need to be created.
D. data that needs to be filtered out
E. and much more!

PROJECT DATA

- You can choose from one of the following data sets:
  - A. Data on the population of forcibly displaced persons
    - https://www.unhcr.org/refugee-statistics/insights/explainers/refugees-r-package.html
  - B. Data on spam emails
    - https://archive.ics.uci.edu/dataset/94/spambase
  - C. Data on every world cup match from 1930 to 2018
    - https://www.kaggle.com/datasets/evangower/fifa-world-cup/code
  - D. Data on netflix shows
    - https://www.kaggle.com/datasets/evangower/fifa-world-cup/code

- Any manageable dataset from:
  - https://github.com/rfordatascience/tidytuesday

PROJECT REPORT

- Students will write an R Markdown HTML report based on the sections below:
  A. Introduction:
     - Provide an introduction that explains the problem statement you are addressing.

  B. Packages Required:
     - All packages used are loaded upfront so the reader knows which are required to replicate the analysis.
     - Explanation is provided regarding the purpose of each package.

PROJECT REPORT

A. Data Preparation:

- Original source where the data was obtained is cited and hyperlinked.
- Source data is thoroughly explained.
- Data importing and cleaning steps are explained in the text and follow a logical process.
- Once your data is clean, show what the final data set looks like.
- Provide summary information about the variables of concern in your cleaned data set.

PROJECT REPORT

- Exploratory Data Analysis:
  A. Uncover new information in the data that is not self-evident (slice and dice the data in different ways, create new variables for more information).
  B. Provide findings in the form of plots.
  C. Plots are carefully presented. One graph illustrates one primary point and is appropriately formatted (plot and axis titles, legend if necessary, scales are appropriate, appropriate geoms used, etc.).
  D. Insights obtained from the analysis are explained.

PROJECT REPORT

- Summary:
  A. Summarize the problem statement you addressed.
  B. Summarize how you addressed this problem statement (the data used and the methodology employed).
  C. Summarize the interesting insights that your analysis provided.

- Formatting:
  A. Tools and techniques from the course are applied competently.
  B. Rmd fully executes without any errors and matches the HTML report submitted by student.

PROJECT SUBMISSION

- Students are expected to complete this project as a a team.
- All group members are expected to contribute equally to the completion of this assignment and anyone judged to not have sufficient contributed to the final report will have their grade penalized.
- While different teams members may have different backgrounds and abilities, it is the responsibility of every group member to understand how and why all code and approaches in the project works.

PROJECT SUBMISSION

- Submissions should include:
  A. RMarkdown file (formated to clearly present all of your code and results)
  B. HTML file
  C. Dataset

- This project is due for submission at the end of the 12th week of the academic calendar for the semester.

SUMMARY

- Rich, fully-documented reproducible analysis can be produced easily using R Markdown.
- The course project allows students to put to work the knowledge and tools gained throughout this course.
- Unique and creative ways to explore data would influence student learning and interests.

FURTHER READING RESOURCES

1. R Core Team. (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/.
2. RStudio Team. (2020). Rstudio: Integrated development environment for R. RStudio, PBC. Boston, MA. http://www.rstudio.com/.
3. HTML reports: http://rmarkdown.rstudio.com/html_document_format.html
4. Latex reports: http://rmarkdown.rstudio.com/pdf_document_format.html

FURTHER READING RESOURCES

5. Word reports: http://rmarkdown.rstudio.com/word_document_format.html
6. Wickham, H. (2019). Stringr: simple, consistent wrappers for common string operations [R package version 1.4.0]. https://CRAN.R-project.org/package=stringr.
7. https://www.freshersnow.com/r-programming-mcqs-and-answers-with-explanation/
8. https://www.sanfoundry.com/r-programming-questions-answers-history/

Thank You