

# R Markdown & Import

Carolina A. de Lima Salge  
Assistant Professor  
Terry College of Business  
University of Georgia

*Business Intelligence*  
Spring 2021



Terry College of Business  
UNIVERSITY OF GEORGIA

The screenshot shows the RStudio interface. The left pane displays the code in 'test.Rmd':

```
1 ---  
2 title: "Diamond sizes"  
3 date: 2016-08-25  
4 output: html_document  
5 ---  
6  
7 `r setup, include = FALSE}  
8 library(ggplot2)  
9 library(dplyr)  
10  
11 smaller <- diamonds %>%  
12 filter(carat <= 2.5)  
13  
14  
15 We have data about `r nrow(diamonds)` diamonds. Only  
16 `r nrow(diamonds) - nrow(smaller)` are larger than  
17 2.5 carats. The distribution of the remainder is shown  
18 below:  
19  
20 `r, echo = FALSE}  
21 smaller %>%  
22 ggplot(aes(carat)) +  
23 geom_freqpoly(binwidth = 0.01)  
24`
```

The right pane shows the Environment and Global Environment panes, both indicating that the environment is empty. The Packages pane lists the User Library:

Name	Description	Version
BiocInstaller	Install/Update Bioconductor, CRAN, and github Packages	1.28.0
cellranger	Translate Spreadsheet Cell Ranges to Rows and Columns	1.1.0
curl	A Modern and Flexible Web Client for R	3.2
digest	Create Compact Hash Digests of R Objects	0.6.15
doParallel	Foreach Parallel Adaptor for the 'parallel' Package	1.0.11
googleAuthR	Authenticate and Create Google APIs	0.6.2.9000
googleComput...	R Interface with Google Compute Engine	0.2.0.9000
httr	Tools for Working with URLs and HTTP	1.3.1
lme4	Linear Mixed-Effects Models using 'Eigen'	1.1-17



# R Markdown

Provides a unified authoring framework for data science, combining your code, its results, and your comments.

- Reproducible
- Shareable
- Many output formats (PDF, Word, Slides)

R Markdown



# R Markdown

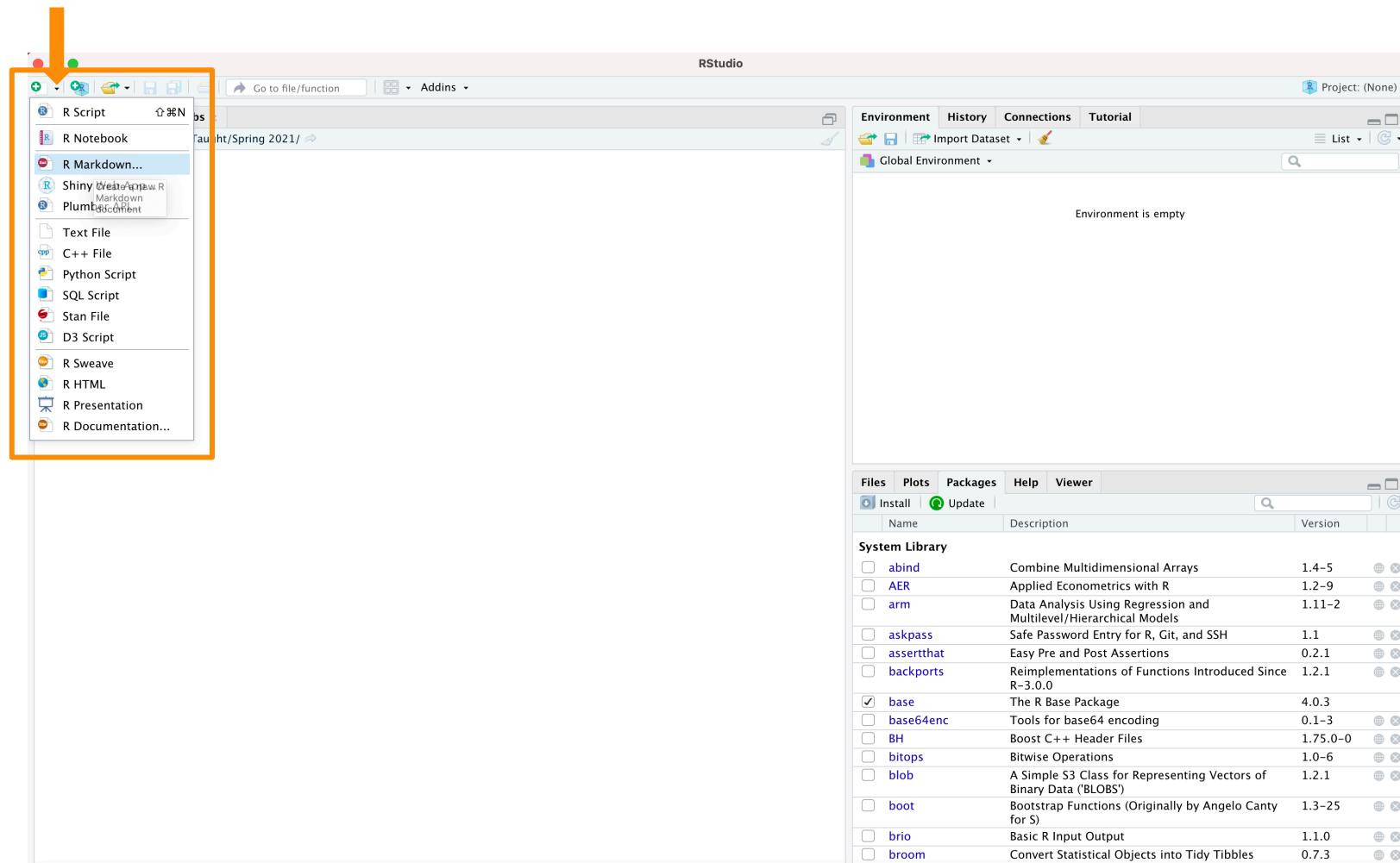


# Why R Markdown?

1. For communicating to decision makers, who want to focus on the conclusions, not the code behind the analysis.
2. For collaborating with other data scientists (including future you!), who are interested in both your conclusions, and how you reached them (i.e. the code).
3. As an environment in which to *do* data science, as a modern day lab notebook where you can capture not only what you did, but also what you were thinking.



# R Markdown in RStudio



# R Markdown in Rstudio (Rmd file)

The screenshot shows the RStudio interface with an Rmd file named "test.Rmd".

- YAML header:** The first few lines of the file are a YAML header:

```
---  
title: "Diamond sizes"  
date: 2016-08-25  
output: html_document  
---
```

A blue box highlights this area.
- Chunks of R code:** Several lines of R code are enclosed in triple backticks:

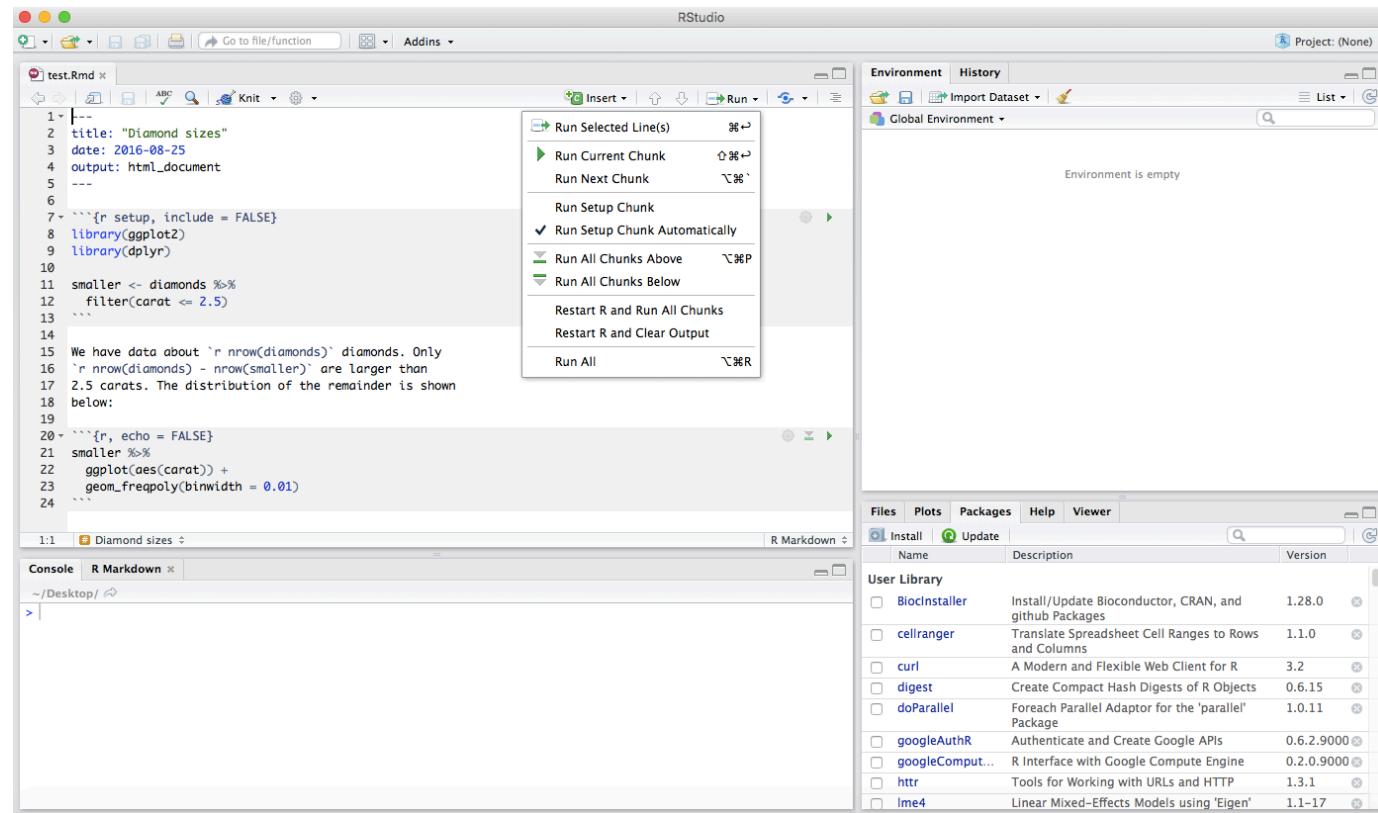
```
```{r setup, include = FALSE}  
library(ggplot2)  
library(dplyr)  
  
smaller <- diamonds %>%  
  filter(carat <= 2.5)  
...  
  
We have data about `r nrow(diamonds)` diamonds. Only  
`r nrow(diamonds) - nrow(smaller)` are larger than  
2.5 carats. The distribution of the remainder is shown  
below:  
  
```{r, echo = FALSE}  
smaller %>%  
  ggplot(aes(carat)) +  
  geom_freqpoly(binwidth = 0.01)  
...``
```

A blue box highlights the first chunk, and an orange box highlights the explanatory text and the second chunk.
- Text mixed with R code:** The explanatory text between the two code chunks is highlighted with an orange box.
- Environment:** The Environment tab shows an empty global environment.
- Packages:** The Packages tab shows the following user library packages:

Name	Description	Version
BioInstaller	Install/Update Bioconductor, CRAN, and github Packages	1.28.0
cellranger	Translate Spreadsheet Cell Ranges to Rows and Columns	1.1.0
curl	A Modern and Flexible Web Client for R	3.2
digest	Create Compact Hash Digests of R Objects	0.6.15
doParallel	Foreach Parallel Adaptor for the 'parallel' Package	1.0.11
googleAuthR	Authenticate and Create Google APIs	0.6.2.9000
googleComput...	R Interface with Google Compute Engine	0.2.0.9000
httr	Tools for Working with URLs and HTTP	1.3.1
lme4	Linear Mixed-Effects Models using 'Eigen'	1.1-17



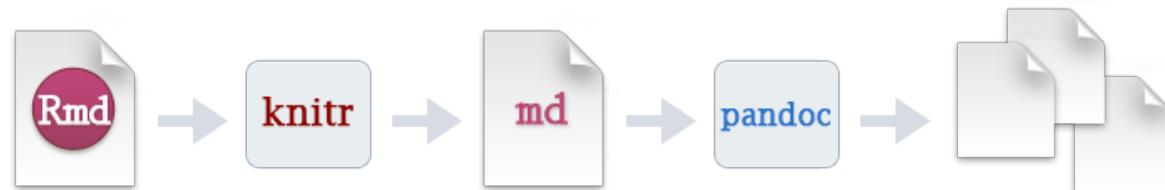
# Run Code



# Knit

To create a report, click “Knit” or press Cmd/Ctrl + Shift + K

- Rmd file is sent to **knitr**, which executes the code and creates a new markdown (.md) document
- The markdown file is then processed by **pandoc**, which creates the finished file



# HTML

```
1 ---  
2 title: "R Markdown"  
3 author: "Carolina Alves de Lima Salge"  
4 date: "1/25/2021"  
5 output: html_document  
6 ---  
7  
8 ```{r setup, include=FALSE}  
9 knitr::opts_chunk$set(echo = TRUE)  
10 ---  
11  
12 ## R Markdown  
13  
14 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>.  
15  
16 When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:  
17  
18 ```{r cars}  
19 summary(cars)  
20 ---  
21  
22 ## Including Plots  
23  
24 You can also embed plots, for example:  
25  
26 ```{r pressure, echo=FALSE}  
27 plot(pressure)  
28 ---  
29  
30 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.  
31
```

Untitled.html | Open in Browser | Find

~/Desktop/Untitled.html

R Markdown

Carolina Alves de Lima Salge

1/25/2021

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>.

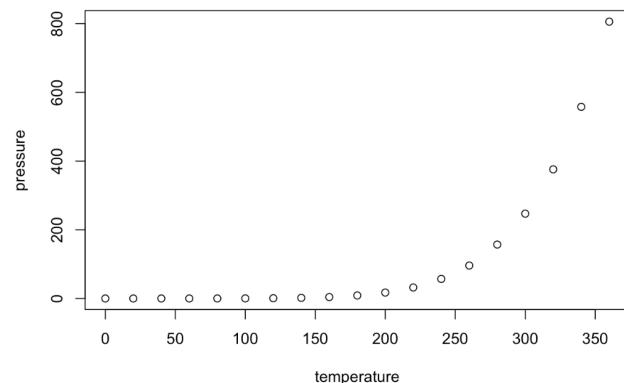
When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed         dist  
##  Min.   : 4.0   Min.   : 2.00  
##  1st Qu.:12.0  1st Qu.: 26.00  
##  Median :15.0  Median : 36.00  
##  Mean   :15.4  Mean   : 42.98  
##  3rd Qu.:19.0  3rd Qu.: 56.00  
##  Max.   :25.0  Max.   :120.00
```

## Including Plots

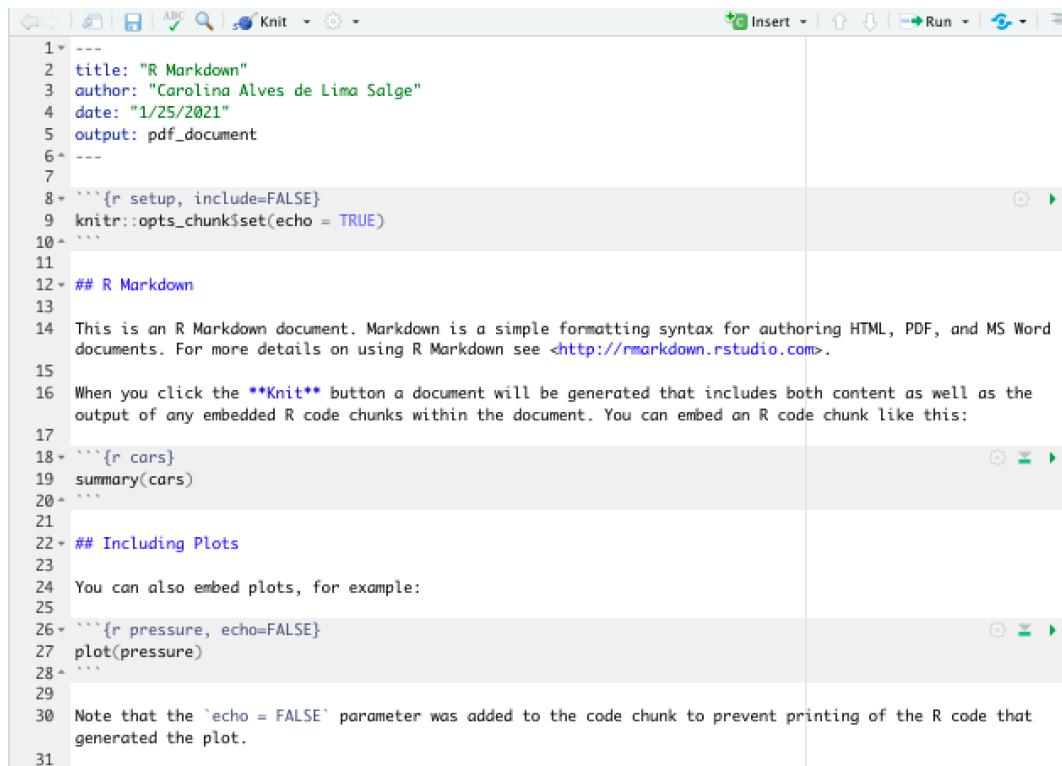
You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.



# PDF



The screenshot shows the RStudio interface with an R Markdown document open. The code editor contains the following content:

```
1 * ---  
2 title: "R Markdown"  
3 author: "Carolina Alves de Lima Salge"  
4 date: "1/25/2021"  
5 output: pdf_document  
6 ---  
7  
8 ```{r setup, include=FALSE}  
9 knitr::opts_chunk$set(echo = TRUE)  
10```  
11  
12 ## R Markdown  
13  
14 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>.  
15  
16 When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:  
17  
18 ```{r cars}  
19 summary(cars)  
20```  
21  
22 ## Including Plots  
23  
24 You can also embed plots, for example:  
25  
26 ```{r pressure, echo=FALSE}  
27 plot(pressure)  
28```  
29  
30 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.  
31
```

## R Markdown

Carolina Alves de Lima Salge

1/25/2021

### R Markdown

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>.

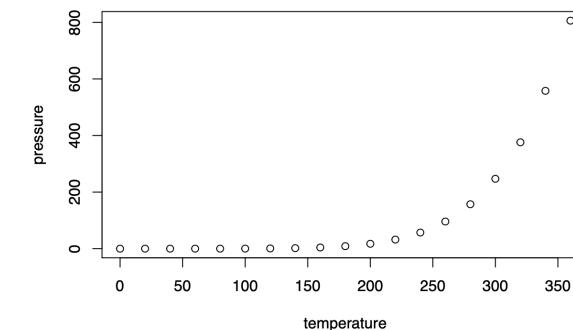
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed         dist  
##  Min.   :4.0   Min.   : 2.00  
##  1st Qu.:12.0  1st Qu.:26.00  
##  Median :15.0  Median :36.00  
##  Mean   :15.4  Mean   :42.98  
##  3rd Qu.:19.0  3rd Qu.:56.00  
##  Max.   :25.0  Max.   :120.00
```

### Including Plots

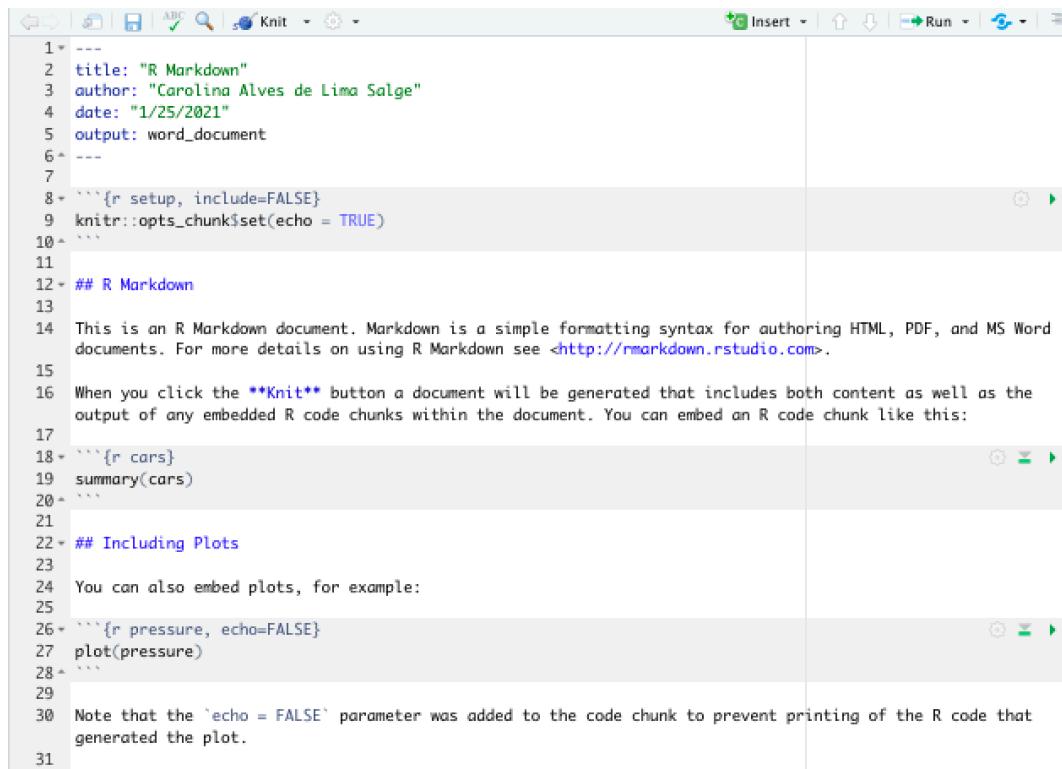
You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.



# Word



A screenshot of the RStudio interface showing an R Markdown document. The code editor contains the following content:

```
1 * ---  
2 title: "R Markdown"  
3 author: "Carolina Alves de Lima Salge"  
4 date: "1/25/2021"  
5 output: word_document  
6 ---  
7  
8 ```{r setup, include=FALSE}  
9 knitr::opts_chunk$set(echo = TRUE)  
10```  
11  
12 ## R Markdown  
13  
14 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>.  
15  
16 When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:  
17  
18 ```{r cars}  
19 summary(cars)  
20```  
21  
22 ## Including Plots  
23  
24 You can also embed plots, for example:  
25  
26 ```{r pressure, echo=FALSE}  
27 plot(pressure)  
28```  
29  
30 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.  
31
```

## R Markdown

Carolina Alves de Lima Salge

1/25/2021

### R Markdown

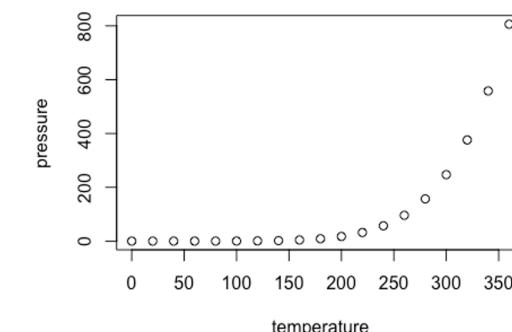
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)  
##      speed          dist  
##  Min.   : 4.0   Min.   : 2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

### Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.



# Data Import

Load flat files (e.g., csv) in R with the **readr** package, which is part of the core tidyverse

```
install.packages("tidyverse") # install package  
  
library(tidyverse) # load already installed package
```



# Getting Started

Turn flat files into tables (e.g., data frames) by using `read_csv()`

```
read_csv() # reads comma delimited files

read_csv2() # reads semicolon separated files
# common in countries where , is used as a decimal place

read_tsv() # reads tab delimited files

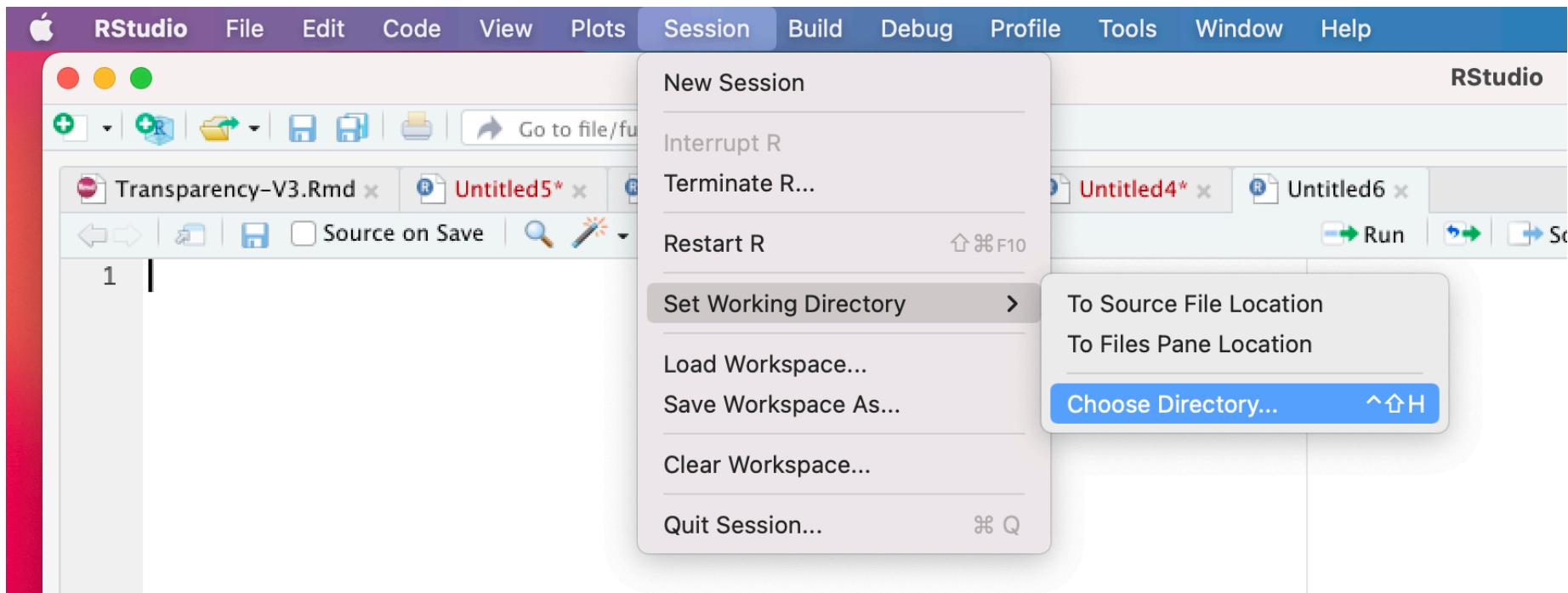
read_delim() # reads in files with any delimiter

# see http://r4ds.had.co.nz/data-import.html for more functions
```



# Getting Started

Set a working directory, and dump your files in there



# Import CSV files

Then use `read_csv()` to import the file of interest

```
CoffeeChain <- read_csv("CoffeeChain.csv")
```

— Column specification

---

---

```
cols(  
  Profit = col_double(),  
  Margin = col_double(),  
  Sales = col_double(),  
  COGS = col_double(),  
  TotExp = col_double(),  
  Marketing = col_double(),  
  Inventory = col_double(),  
  BudgProf = col_double(),  
  BudgMarg = col_double(),  
  BudgSales = col_double(),
```



# Save CSV files

Use `write_csv()` to save a file of interest. This function increases the chances of the output file being read back in correctly by:

- Always encoding strings in UTF-8
- Saving dates and date-times in ISO8601 format

```
write_csv(CoffeeChain, "CoffeeChain.csv") # save csv file
```

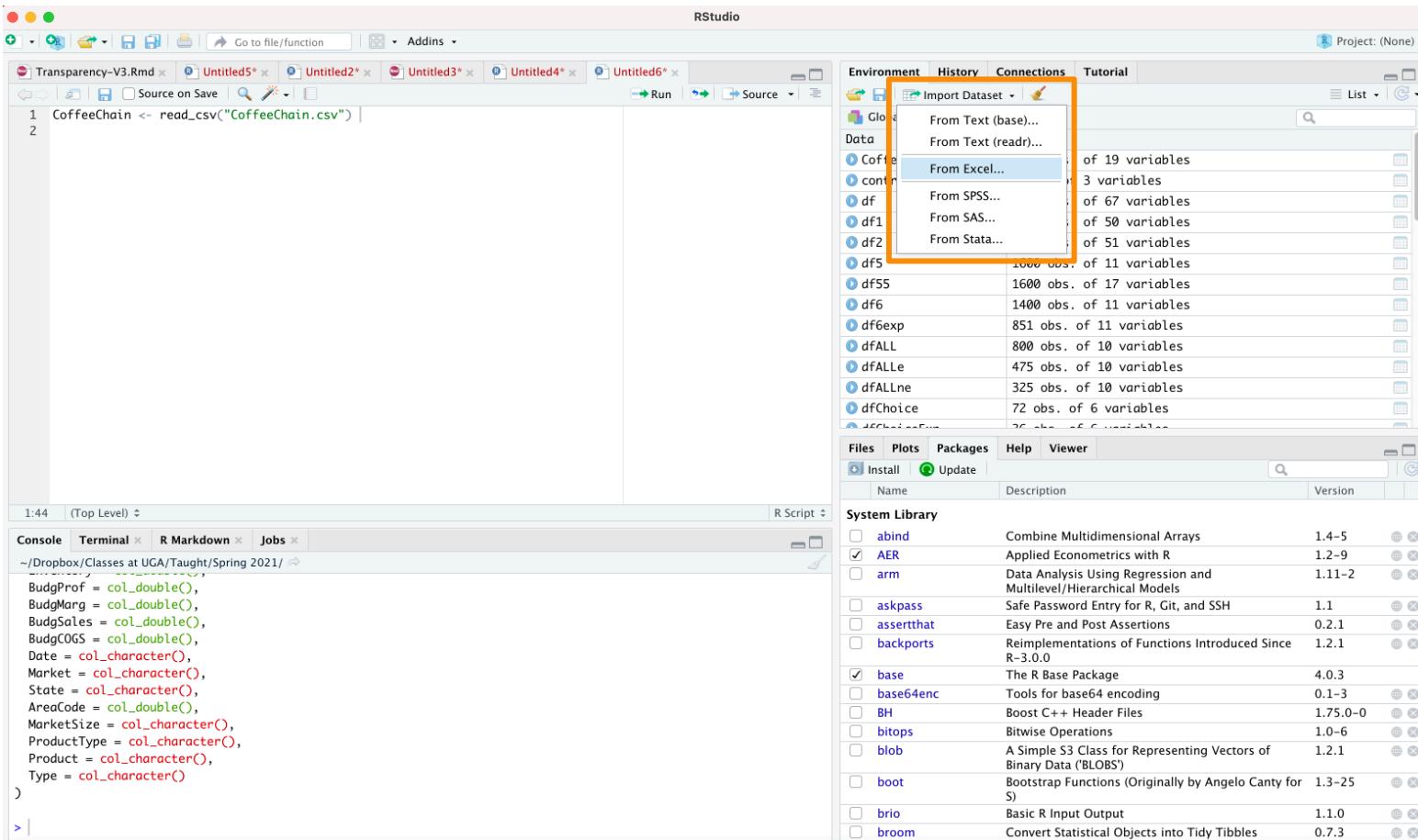


**“This is all you need to know to read ~75% of CSV files”**

Wickham & Grolemund in *R for data science*  
<http://r4ds.had.co.nz/data-import.html>



# Import Excel files



The screenshot shows the RStudio interface. In the top-left corner, there's a red decorative element. The main window displays an R script in the left pane and the Environment tab of the global environment in the right pane. A red box highlights the 'Import Dataset' option under the 'Data' section of the dropdown menu.

```
1 CoffeeChain <- read_csv("CoffeeChain.csv")  
2
```

Environment tab (highlighted by a red box):

- From Text (base)...
- From Text (readr)...
- From Excel...**
- From SPSS...
- From SAS...
- From Stata...

Global Environment (right pane):

- CoffeeChain (19 variables)
- cont (3 variables)
- df (67 variables)
- df1 (50 variables)
- df2 (51 variables)
- df5 (1000 obs., 11 variables)
- df55 (1600 obs., 17 variables)
- df6 (1400 obs., 11 variables)
- df6exp (851 obs., 11 variables)
- dfALL (800 obs., 10 variables)
- dfALLE (475 obs., 10 variables)
- dfALLne (325 obs., 10 variables)
- dfChoice (72 obs., 6 variables)
- dfCountry (26 obs., 6 variables)

System Library (bottom right):

Name	Description	Version
abind	Combine Multidimensional Arrays	1.4-5
<input checked="" type="checkbox"/> AER	Applied Econometrics with R	1.2-9
arm	Data Analysis Using Regression and Multilevel/Hierarchical Models	1.11-2
askpass	Safe Password Entry for R, Git, and SSH	1.1
assertthat	Easy Pre and Post Assertions	0.2.1
backports	Reimplementations of Functions Introduced Since R-3.0.0	1.2.1
<input checked="" type="checkbox"/> base	The R Base Package	4.0.3
base64enc	Tools for base64 encoding	0.1-3
BH	Boost C++ Header Files	1.75.0-0
bitops	Bitwise Operations	1.0-6
blob	A Simple S3 Class for Representing Vectors of Binary Data ('BLOBS')	1.2.1
boot	Bootstrap Functions (Originally by Angelo Canty for S)	1.3-25
brio	Basic R Input Output	1.1.0
broom	Convert Statistical Objects into Tidy Tibbles	0.7.3



# Import Excel files

The screenshot shows the RStudio interface with the 'Import Excel Data' dialog box open. The dialog box has the following sections:

- Data Preview:** Shows a preview of the first 50 entries from the 'CoffeeChain.xlsx' file. The columns include Profit, Margin, Sales, COGS, TotExp, Marketing, Inventory, BudgProf, BudgMarg, BudgSales, BudgCOGS, and Date.
- Import Options:** Includes fields for Name (set to 'CoffeeChain'), Max Rows, First Row as Names (checked), Sheet (set to 'Default'), Skip (set to 0), Open Data Viewer (checked), Range (set to 'A1:D10'), and NA.
- Code Preview:** Displays the R code used to import the data:

```
library(readxl)  
CoffeeChain <- read_excel("CoffeeChain.xlsx")  
View(CoffeeChain)
```
- Buttons:** 'Import' and 'Cancel' buttons at the bottom right.



# Import Excel files

Then use `read_excel()` to import the file of interest

```
library(readxl)
CoffeeChain <- read_excel("CoffeeChain.xlsx")
View(CoffeeChain)

> library(readxl)
> CoffeeChain <- read_excel("CoffeeChain.xlsx")

> View(CoffeeChain)
```



# Import from Database

Use **DBI**, along with a database specific package  
(e.g. **RMySQL**, **RSQLite**, **RPostgreSQL**)

```
library(RMySQL)
library(DBI)
# Connect to Classic Models database
conn1 <- dbConnect(MySQL(),
  host= "",
  dbname= "",
  user= "",
  password= "")

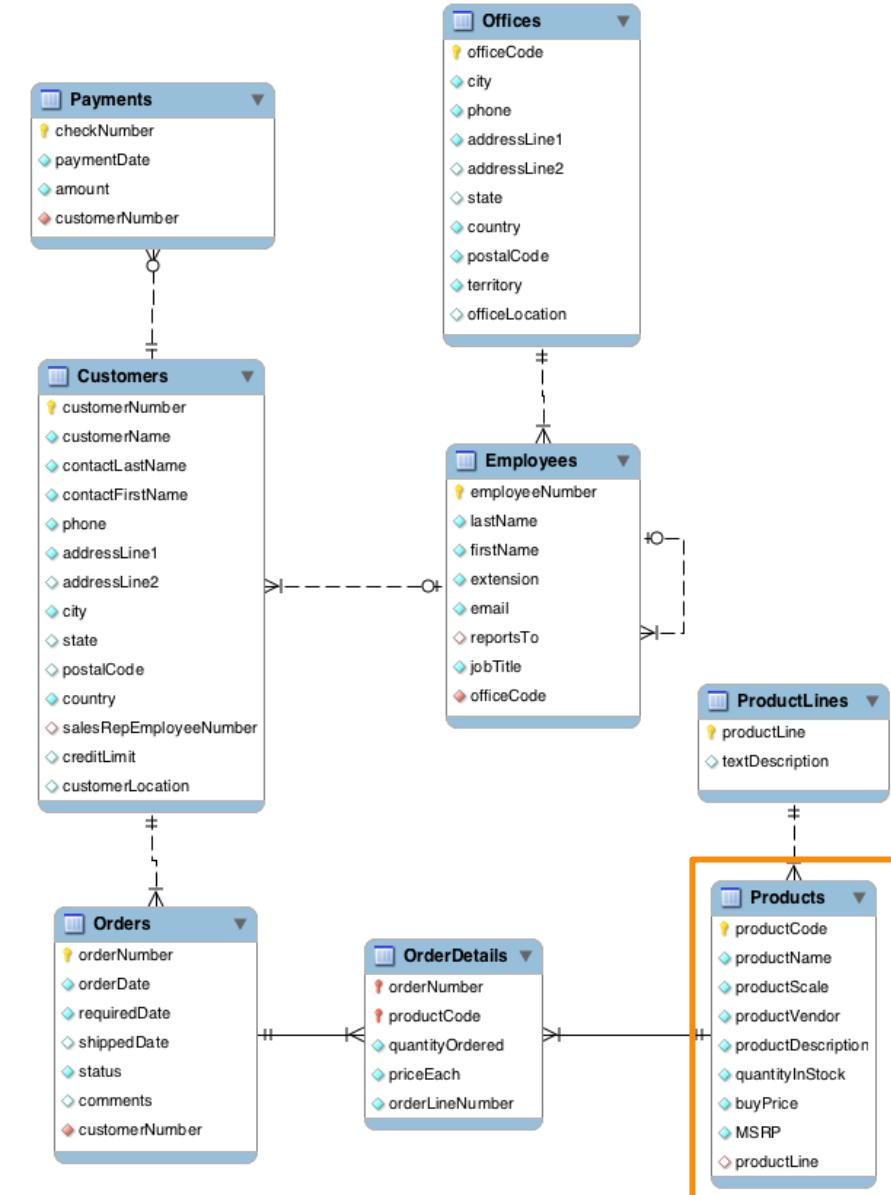
# pull data from the database
products <- dbGetQuery(conn1, "select * from Products;")
```



# Import from Database

```
library(RMySQL)
library(DBI)
# Connect to Classic Models database
conn1 <- dbConnect(MySQL(),
  host= "",
  dbname= "",
  user= "",
  password= "")

# Pull data from the database with SQL
products <- dbGetQuery(conn1, "select *
from Products;")
```



# Other Types of Data

Start with the tidyverse list of packages below:

- **haven** reads SPSS, Stata, and SAS files
- **jsonlite** for json and **xml2** for XML



# Exercise

Open RStudio and do the following:

- Install the **readxl**, **RMySQL**, and **DBI** packages
- Create a R Markdown file (see slide 5)
- Set a working directory (see slide 14)
- Download the CoffeeChain files from eLc and save them in your working directory folder



# Exercise (cont.)

Open RStudio and do the following:

- Copy and paste the following code into one or many chunks of R code in R Markdown

```
library(tidyverse)
CoffeeChain <- read_csv("CoffeeChain.csv")
CoffeeChain # Print first 10 rows of the data

library(readxl)
CoffeeChain2 <- read_excel("CoffeeChain.xlsx")
CoffeeChain2

library(RMySQL)
library(DBI)
# Connect to Classic Models database
conn1 <- dbConnect(MySQL(),
                  host= "",
```



## Exercise (cont.)

Open RStudio and do the following:

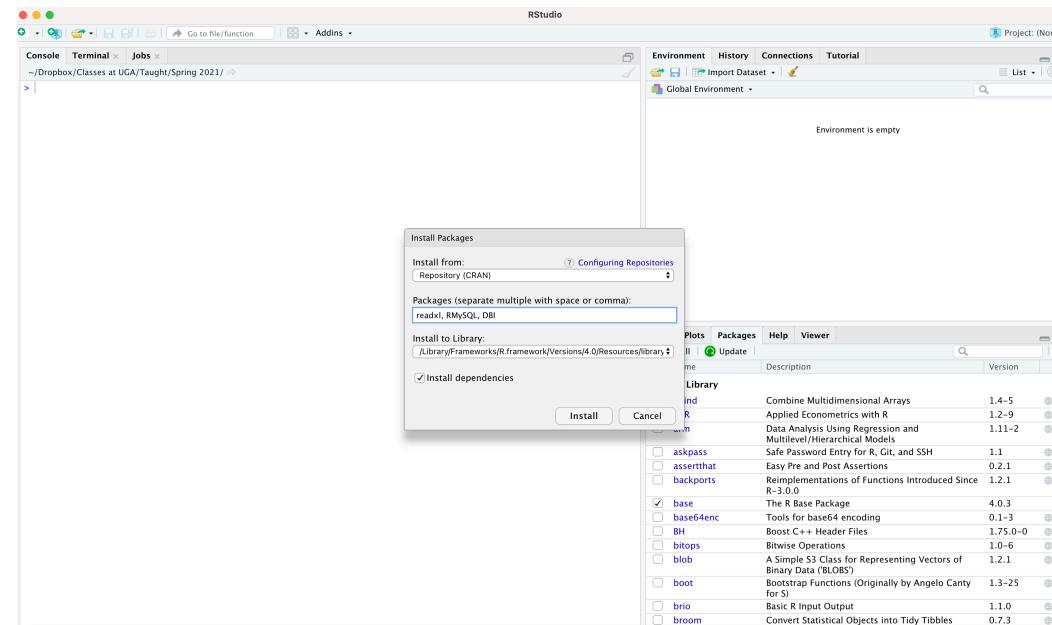
- Change `{r setup, include=FALSE}` to `{r setup, include=TRUE}` as well as Note that the ``echo = FALSE`` parameter was added to the code chunk to prevent printing of the R code that generated the plot. to Note that the ``echo = TRUE`` parameter was added to the code chunk to print the R code that generated the results.
- Knit the file in your saved directory to PDF or Word



# Exercise (visuals)

Open RStudio and do the following:

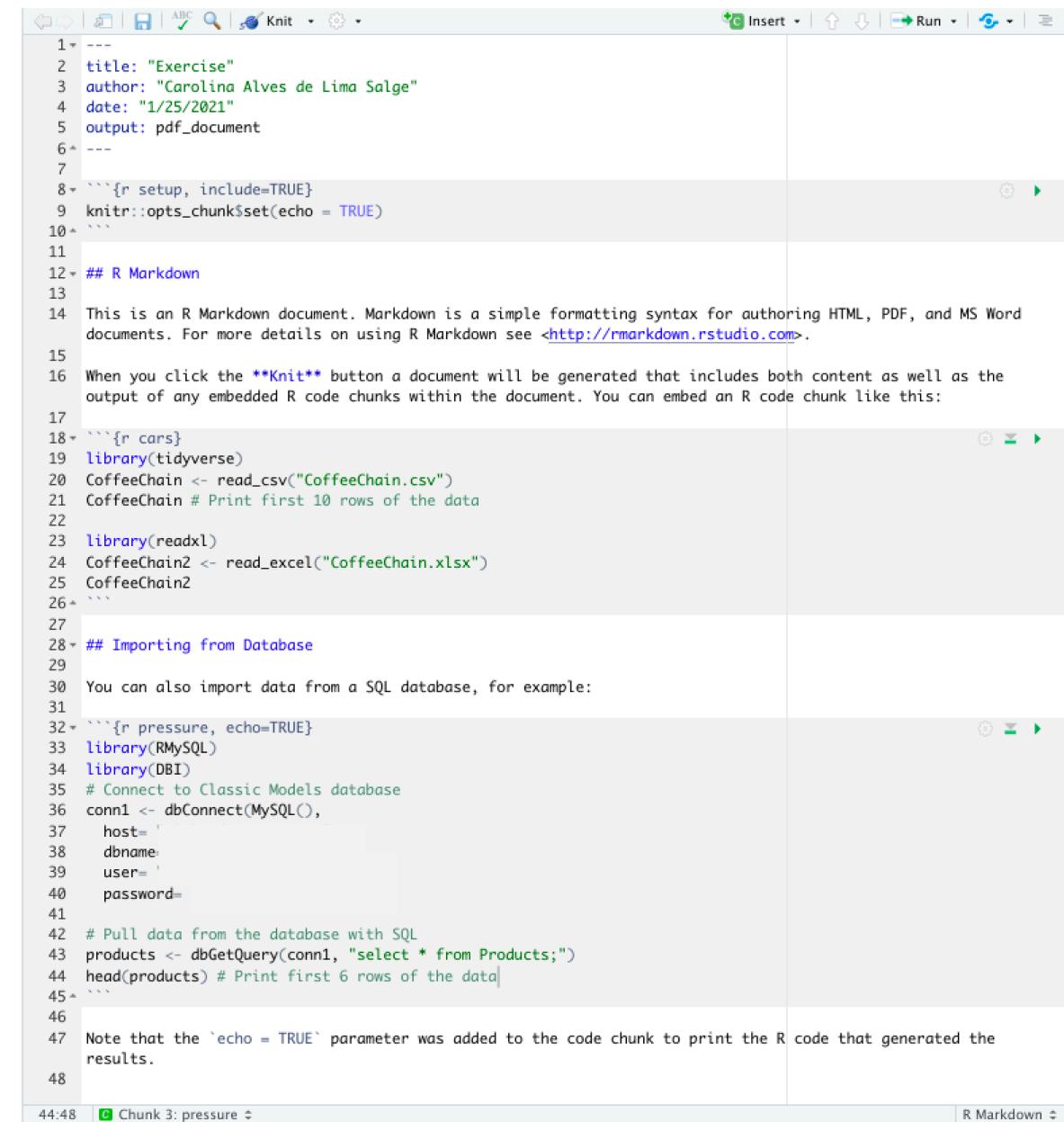
- Install the **readxl**, **RMySQL**, and **DBI** packages



# Exercise (visuals cont.)

Open RStudio and do the following:

- Copy and paste the following code into one or many chunks of R code in R Markdown



The screenshot shows the RStudio interface with an R Markdown document open. The code editor contains the following R Markdown code:

```
1 ---  
2 title: "Exercise"  
3 author: "Carolina Alves de Lima Salge"  
4 date: "1/25/2021"  
5 output: pdf_document  
6 ---  
7  
8 ```{r setup, include=TRUE}  
9 knitr::opts_chunk$set(echo = TRUE)  
10 ```  
11  
12 ## R Markdown  
13  
14 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>.  
15  
16 When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:  
17  
18 ```{r cars}  
19 library(tidyverse)  
20 CoffeeChain <- read_csv("CoffeeChain.csv")  
21 CoffeeChain # Print first 10 rows of the data  
22  
23 library(readxl)  
24 CoffeeChain2 <- read_excel("CoffeeChain.xlsx")  
25 CoffeeChain2  
26  
27  
28 ## Importing from Database  
29  
30 You can also import data from a SQL database, for example:  
31  
32 ```{r pressure, echo=TRUE}  
33 library(RMySQL)  
34 library(DBI)  
35 # Connect to Classic Models database  
36 conn1 <- dbConnect(MySQL(),  
37 host=''  
38 dbname=''  
39 user=''  
40 password=''  
41  
42 # Pull data from the database with SQL  
43 products <- dbGetQuery(conn1, "select * from Products;")  
44 head(products) # Print first 6 rows of the data  
45  
46  
47 Note that the `echo = TRUE` parameter was added to the code chunk to print the R code that generated the results.  
48
```

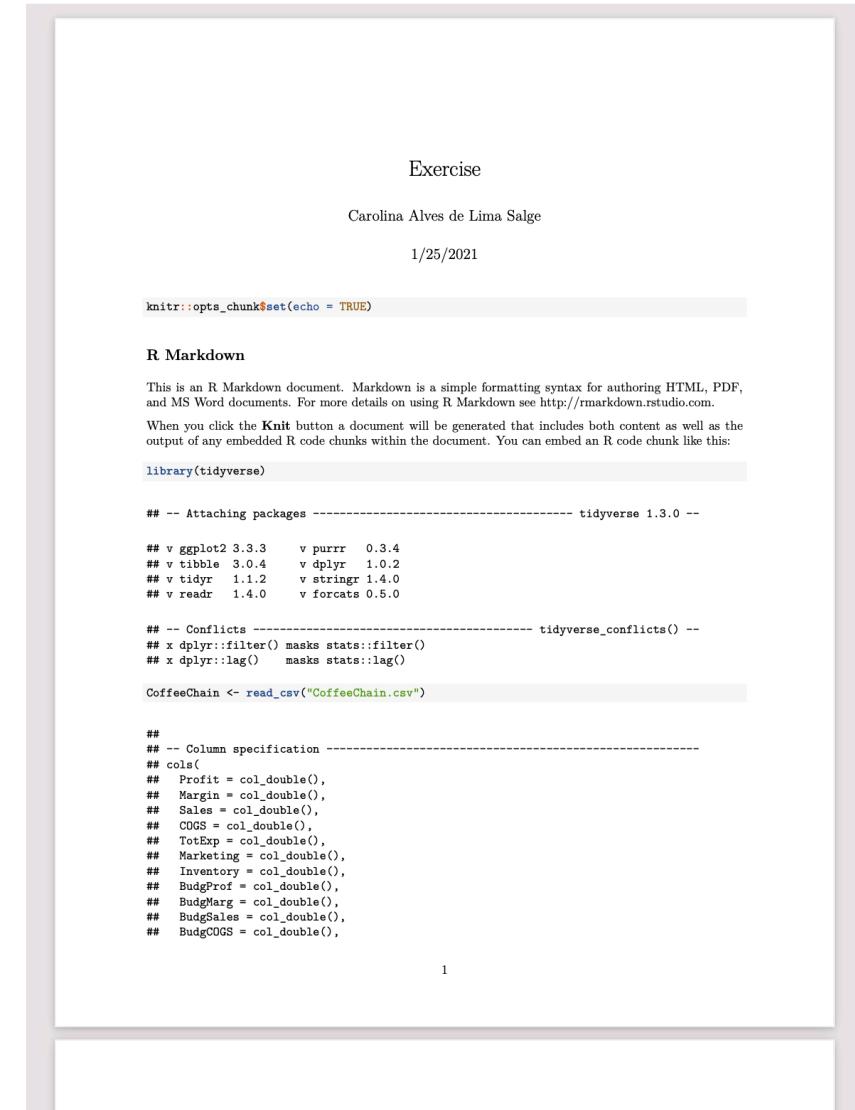
The status bar at the bottom indicates the time is 44:48 and the current chunk is Chunk 3: pressure.



# Exercise (visuals cont.)

Open RStudio and do the following:

- Knit the file in your saved directory to PDF or Word



The screenshot shows a PDF document titled "Exercise". The header includes the title, author ("Carolina Alves de Lima Salge"), and date ("1/25/2021"). Below the header is a code block starting with "knitr::opts\_chunk\$set(echo = TRUE)". The section "R Markdown" is described as a simple syntax for authoring HTML, PDF, and MS Word documents. It explains that clicking the "Knit" button generates a document including both content and R code chunks. The code block then continues with library("tidyverse") and other R code for reading a CSV file and specifying column types.

```
knitr::opts_chunk$set(echo = TRUE)

library(tidyverse)

## -- Attaching packages -----
## v ggplot2 3.3.3     v purrr  0.3.4
## v tibble   3.0.4     v dplyr   1.0.2
## v tidyverse 1.1.2    v stringr 1.4.0
## v readr   1.4.0     vforcats 0.5.0

## -- Conflicts -----
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()    masks stats::lag()

CoffeeChain <- read_csv("CoffeeChain.csv")

## -- Column specification -----
## cols(
##   Profit = col_double(),
##   Margin = col_double(),
##   Sales = col_double(),
##   COGS = col_double(),
##   TotExp = col_double(),
##   Marketing = col_double(),
##   Inventory = col_double(),
##   BudgProf = col_double(),
##   BudgMarg = col_double(),
##   BudgSales = col_double(),
##   BudgCOGS = col_double(),
```

# *Thank You!*



Terry College of Business  
UNIVERSITY OF GEORGIA