FurstPerson.

# Master Tutorial Introduction to Reproducible Research using R, RStudio, and R Markdown

Rob Stilson

April 16, 2016

FurstPerson.

# Introduction

# Overview

FurstPerson

- What is Reproducible Research and why is it important?
- **Learning Objective #1:** Using R, RStudio, and R Markdown to setup a document for Reproducible Research
- **Learning Objective #2:** Publishing your Reproducible Research Document in HTML, Word, or PDF
- **Learning Objective #3:** Updating the data used in your document
- **Learning Objective #4:** Publishing your findings on the web via RPubs

# What is Reproducible Research?

FurstPerson

- Fully documenting the steps that were taken to conduct a research study
- This enables:
  - Replication of the study's findings
  - Understanding exactly what analyses were conducted
  - Updating an analysis efficiently.
- Full transparency of the data and the process

# Areas RR is applicable

FurstPerson.

- For Academics
  - Studies
  - Theses and Dissertations
- For Practitioners
  - Tech Reports
  - White Papers
  - Tech Manuals
  - Assessment Reports

# Things to Remember

FurstPerson

- This Master Tutorial is NOT about learning R
  - If you are new to R, this tutorial will give you an incentive to learn it
  - A lot of code (don't worry, all materials available at the end)
- This Master Tutorial IS about Reproducible Research Documents
  - Will be demonstrated in R
  - Markdown can also be used with
    - Python
    - JavaScript
    - Many others

FurstPerson

# Learning Objective #1: Using R, RStudio, and R Markdown to setup a document for Reproducible Research

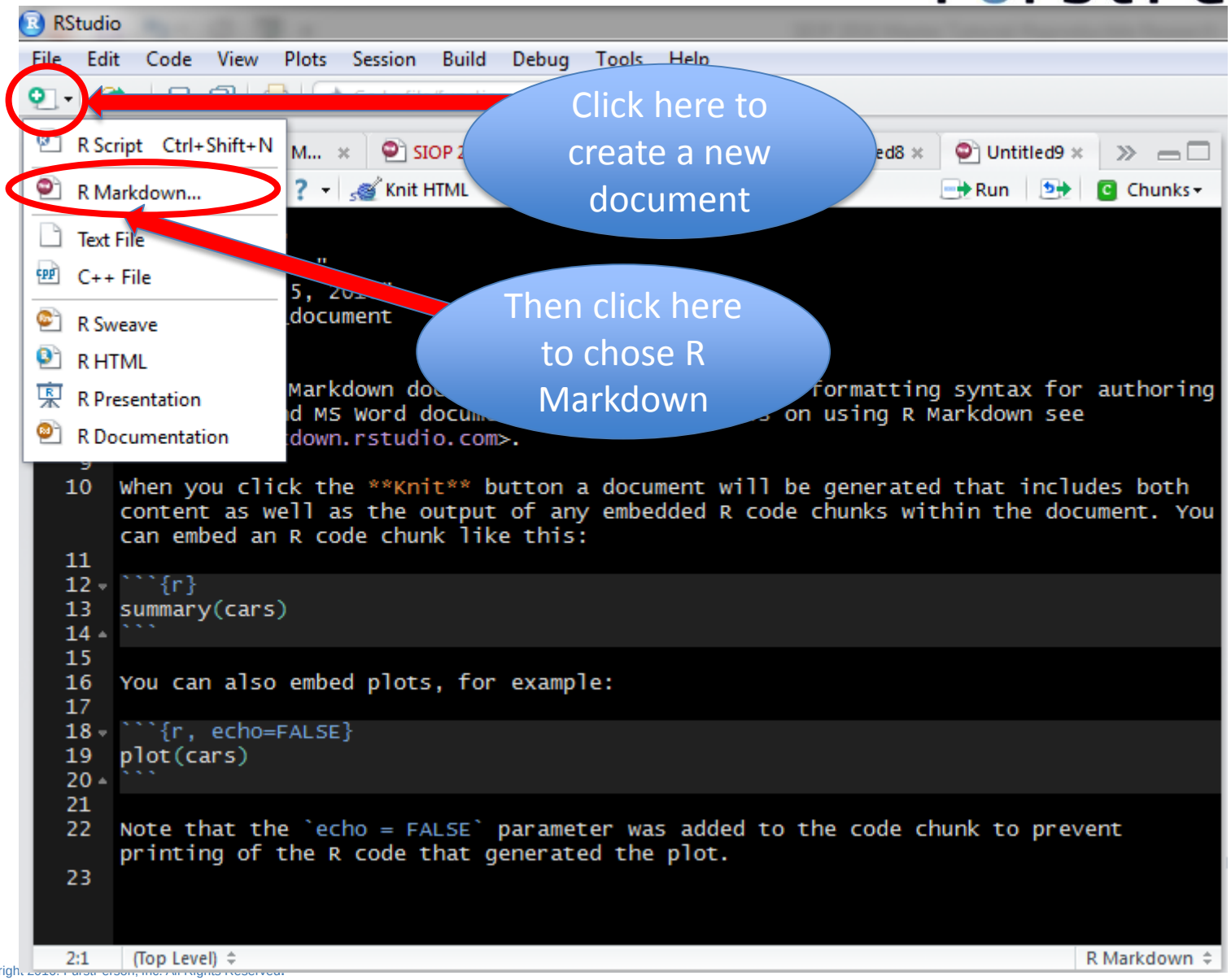
# R Studio

FurstPerson.

- Download from https://www.rstudio.com/products/RStudio/#Desktop
- R Studio is an Integrated Development Environment (IDE) for R
- Includes integration with R Markdown
- Additionally, it facilitates
  - Importing data
  - Installing packages
  - Updating packages
  - Syntax highlighting, code completion, and smart indentation
  - Quickly jumping to function definitions
  - Many other areas R

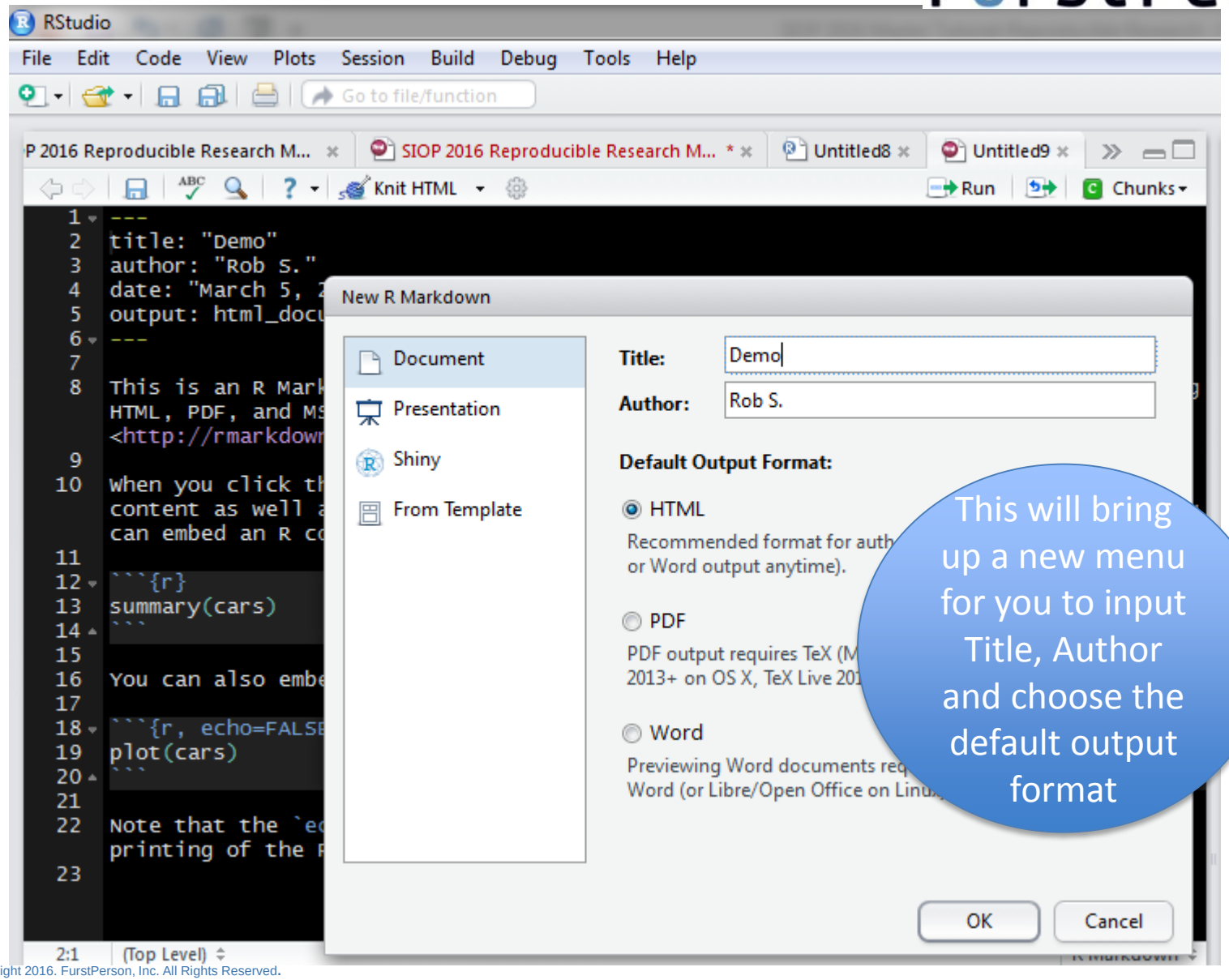
# R Studio

FurstPerson.



# R Studio-R Markdown

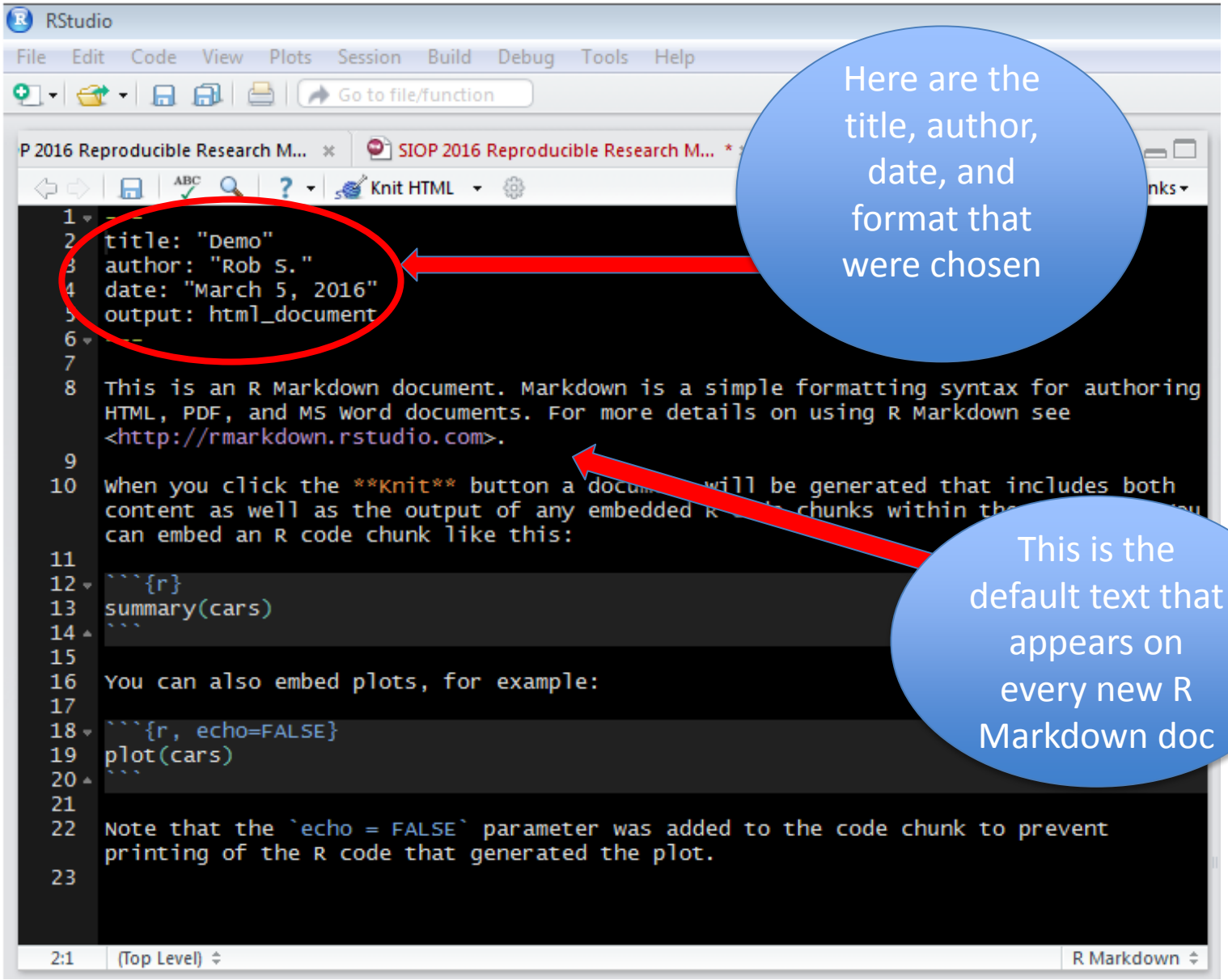
FurstPerson.




This will bring up a new menu for you to input Title, Author and choose the default output format

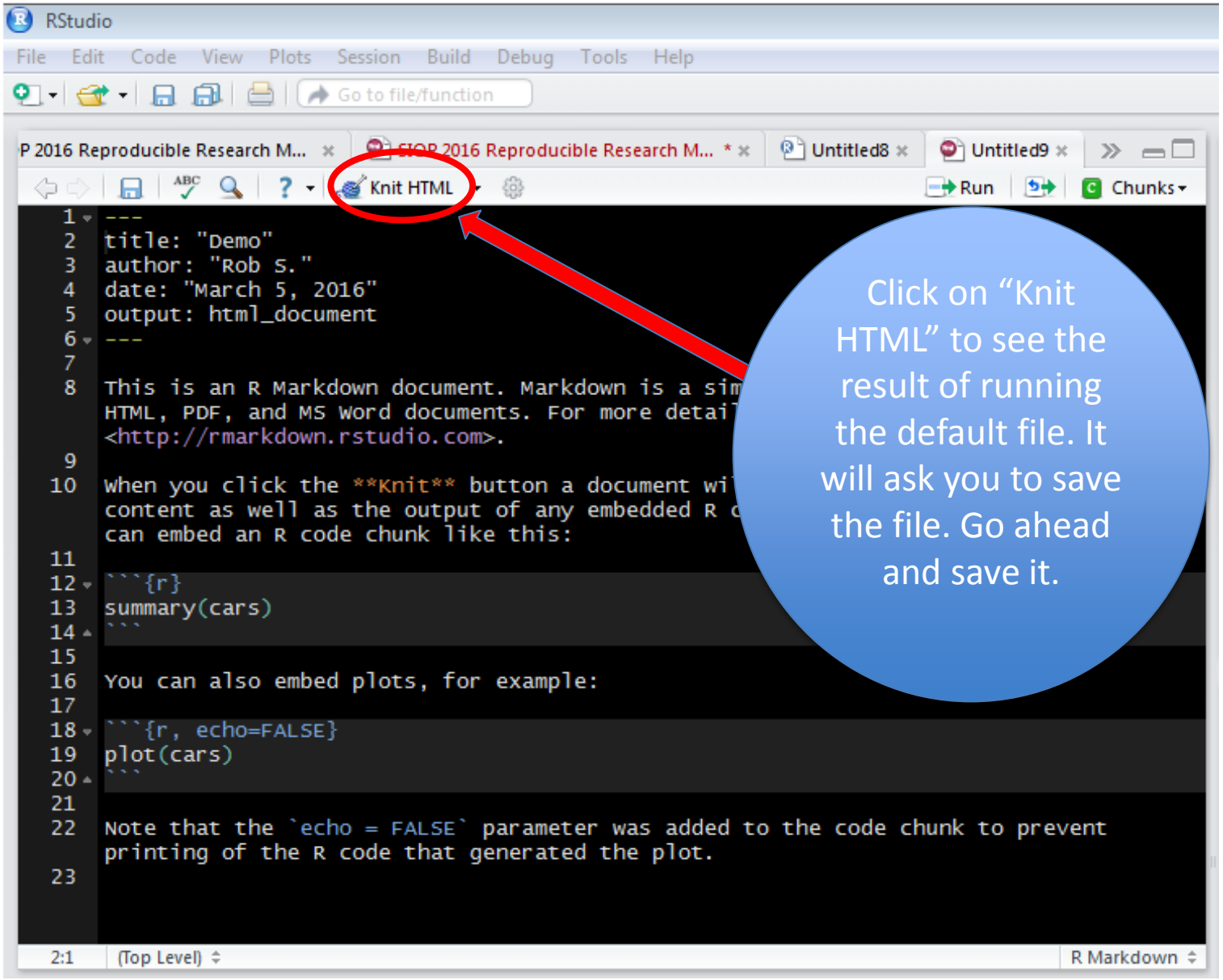
# R Studio-R Markdown

FurstPerson



# R Studio-R Markdown

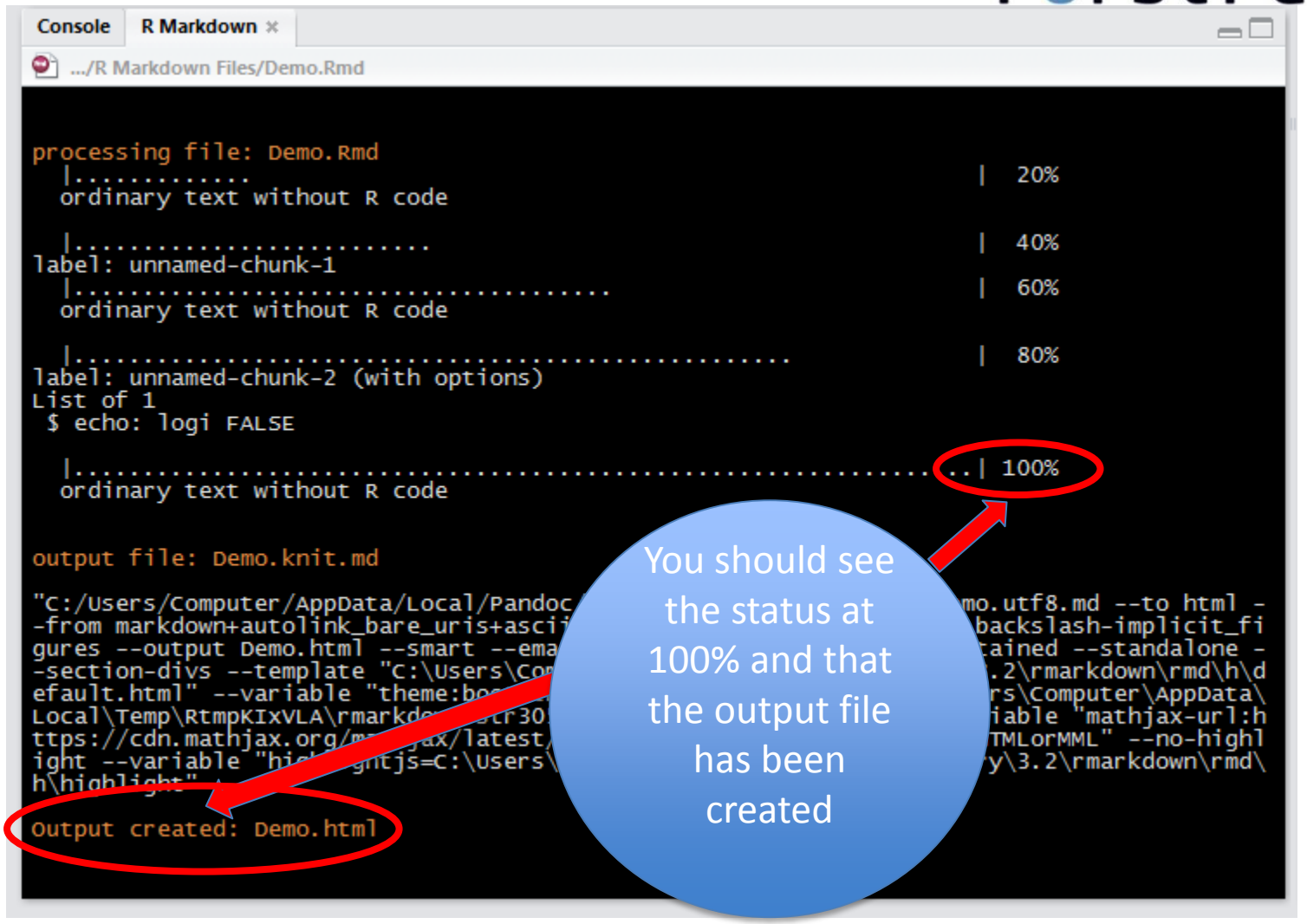
FurstPerson




Click on "Knit HTML" to see the result of running the default file. It will ask you to save the file. Go ahead and save it.

# R Studio-R Markdown

FurstPerson.



# R Studio-HTML Output

FurstPerson.

E:/R Stuff/R Markdown Files/Demo.html

Demo.html | Open in Browser | Find | Publish

## Demo

*Rob S.*

*March 5, 2016*

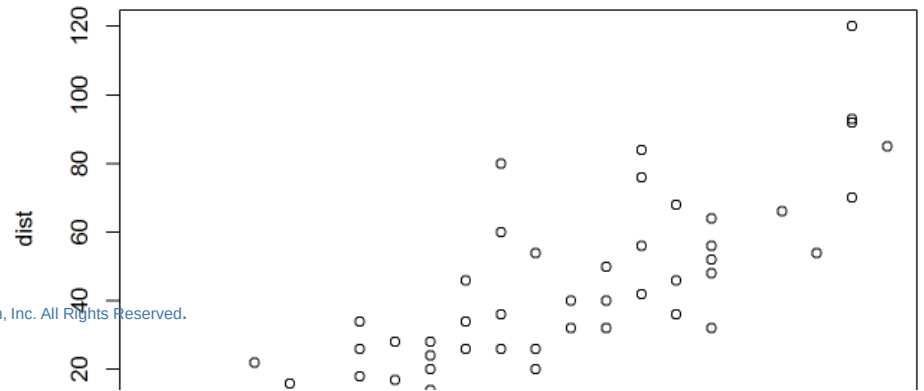
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

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
```

You can also embed plots, for example:




This is the HTML file that was created

# Syntax Color Scheme

FurstPerson

- Any syntax in the presentation will be color coded
- ```r{}
- summary(cars)
- ```

GREEN means code chunk R syntax or R Script

- `r nrow (mydata)`

BLUE means inline code R syntax

- Click here

Purple means hyperlink R syntax

# R Markdown Coding

FurstPerson

- In R Markdown, there are two type of code
  - Inline Code
  - Code Chunks
- Inline code is where you embed R code directly into your text
  - Uses include:
    - Keeping sample size fluid (e.g., `` `r nrow (mydata)` ``)
    - Letting reader know text specifically references something in R (e.g., the `` `psych` `` package)
- Code chunks separate out R code from your text and stand alone
  - Uses include:
    - The bulk of the analysis of a project
    - Keeping comments hidden

# R Markdown-Code Chunk

FurstPerson

- ```{r}``` designates a code chunk.
- This is where your R code goes
- The short cut to put in a code chunk is CTRL+ALT+ i (remember this)
- Line 13, summary(cars) is R code for run summary statistics on the data set "cars"

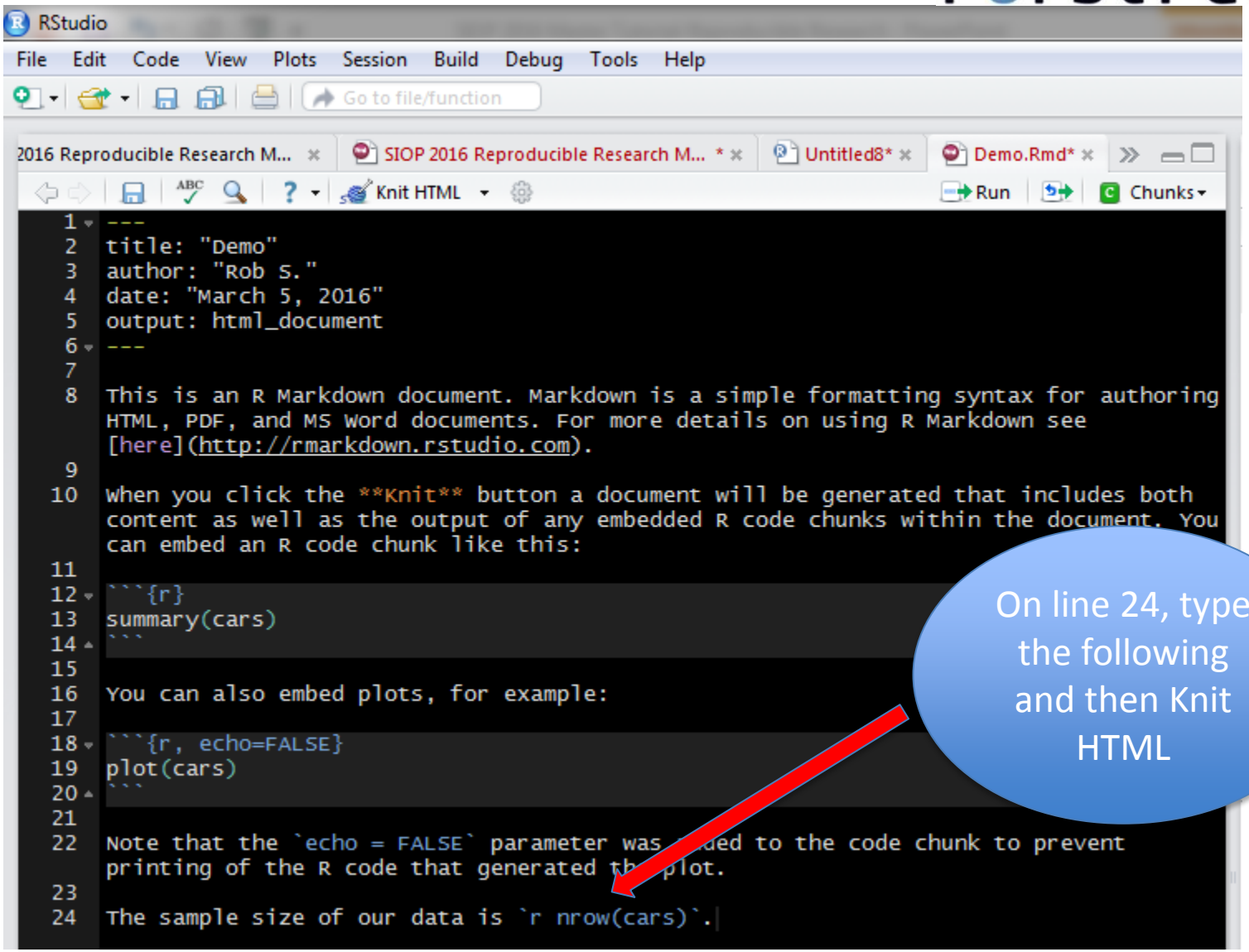
# R Markdown-Inline Code

FurstPerson

- Go back to you R Markdown Demo File in R Studio
- On line 8, notice the <> around the URL
- These <> symbols tell R Markdown to make it a hyperlink
- If you wanted it to say “here” with the hyperlink embedded in the word, it would look like the following with square brackets around the word in which the URL is embedded:
    - For more details on using R Markdown see [here](http://rmarkdown.rstudio.com).
- On line 10 of your script, notice the ** around **Knit**. This tells R Markdown to make the work “Knit” **bold** like CTRL+B in MS Word
- Let’s do some examples

# R Markdown-Inline Code

FurstPerson

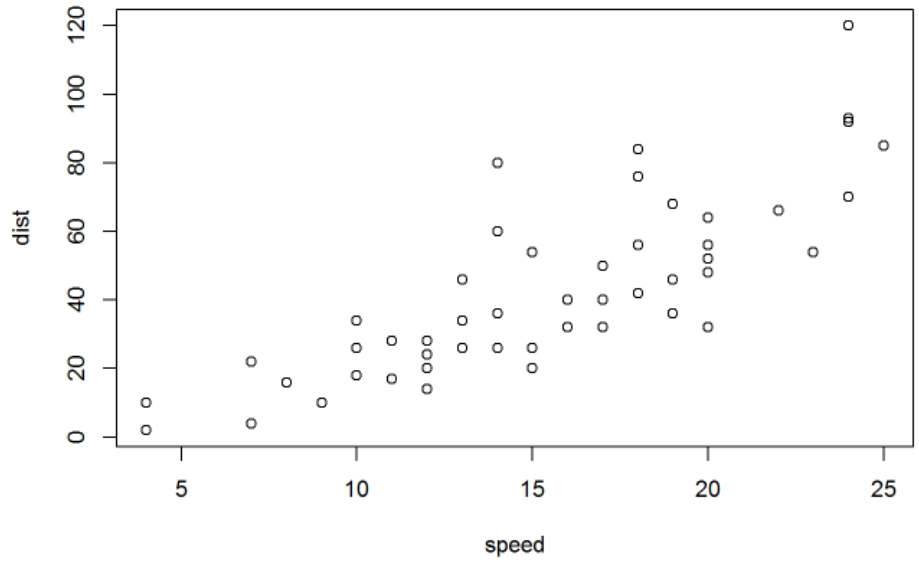


# R Studio-HTML Output

FurstPerson

```
##      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
```

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

The sample size of our data is 50.

Instead of printing `r nrow(cars)` we get the actual number of rows, 50.

# R Markdown-Inline Code

FurstPerson

- Now, you try.
- On line 26, type in the following:
- The mean of speed is `r mean(cars$speed)` and the mean of distance is `r mean(cars$dist)`.
- Now click Knit HTML
- Notice how the document now includes the means of your variables
- If you received more “cars” data, you could simply amend the current cars file with the data, rerun the document, and the sample size and means would instantly update. (more on this later)

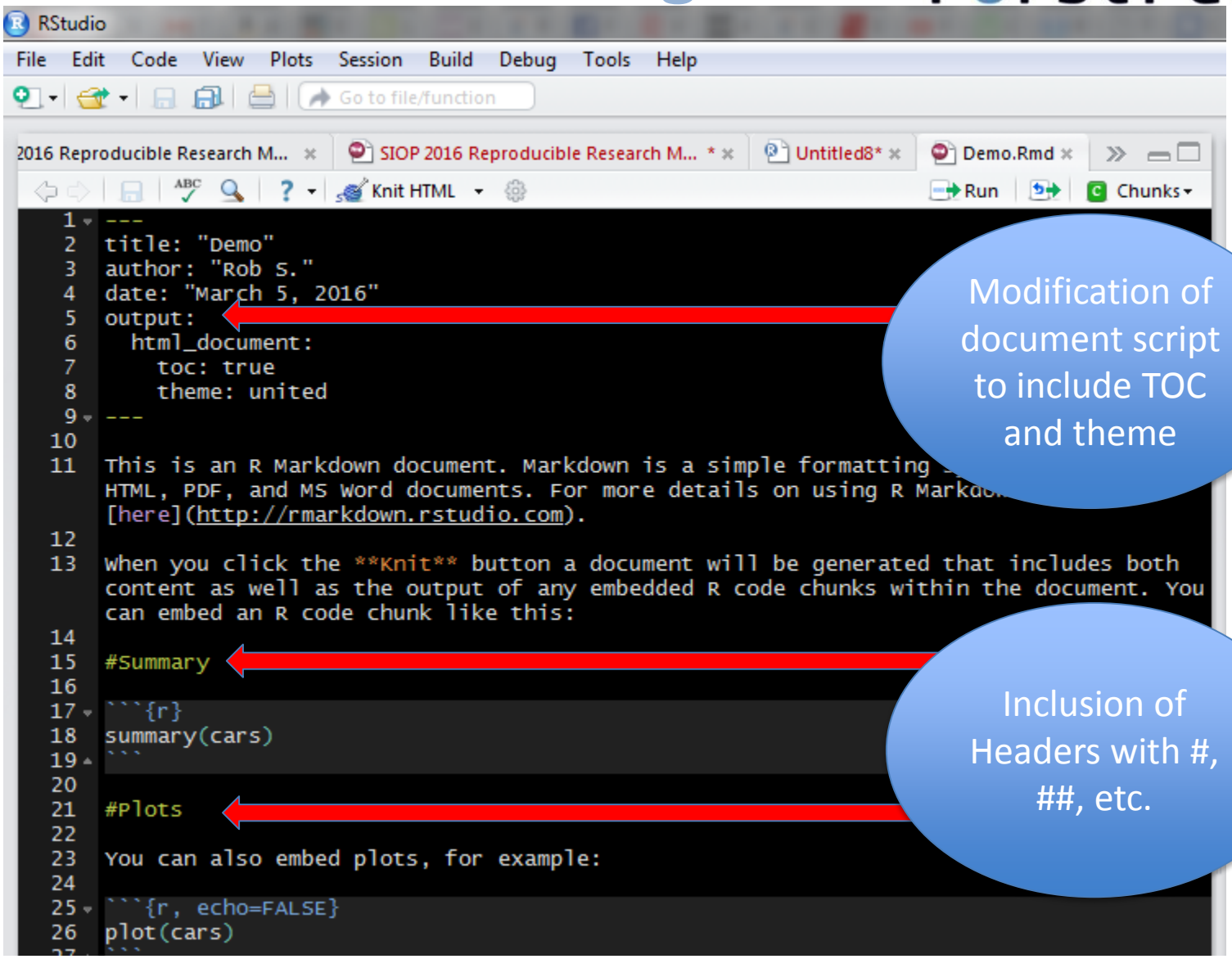
# R Markdown-Formatting

FurstPerson

- Within R Markdown documents you can add:
  - Table of Contents
  - Headers
  - Italics and Bold
  - Superscripts and Subscripts
  - Strikethrough
  - Links
  - Inline equations
  - Images
  - Ordered and unordered list
  - Tables
- Here is a link to the cheat sheet

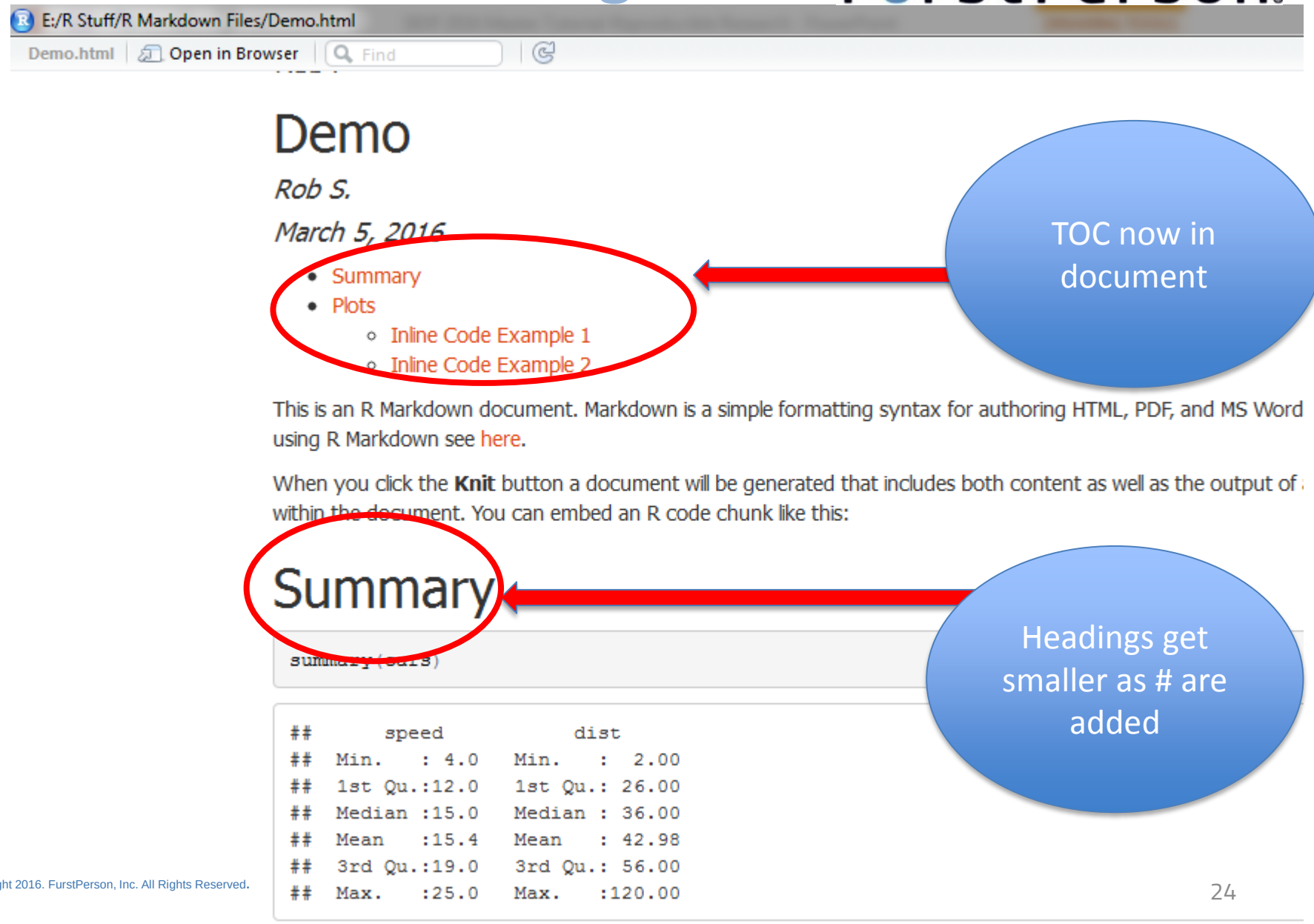
# R Markdown-Formatting

FurstPerson.



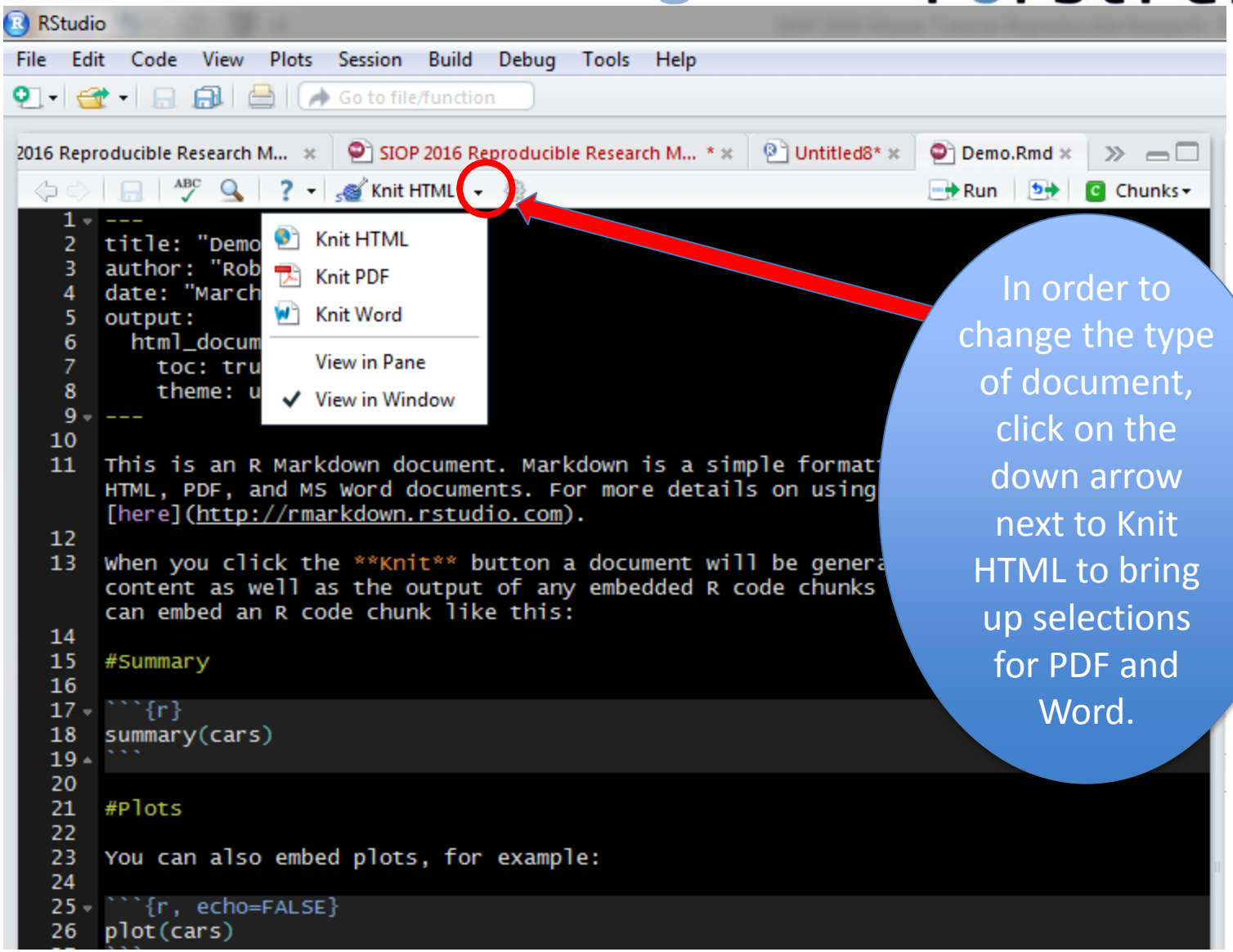
# R Markdown-Formatting

FurstPerson



# R Markdown-Formatting

FurstPerson.




In order to change the type of document, click on the down arrow next to Knit HTML to bring up selections for PDF and Word.

# R Markdown-Formatting

FurstPerson

- Much of the formatting is interchangeable between document types
- Tables sometimes behave differently between HTML and Word and PDF

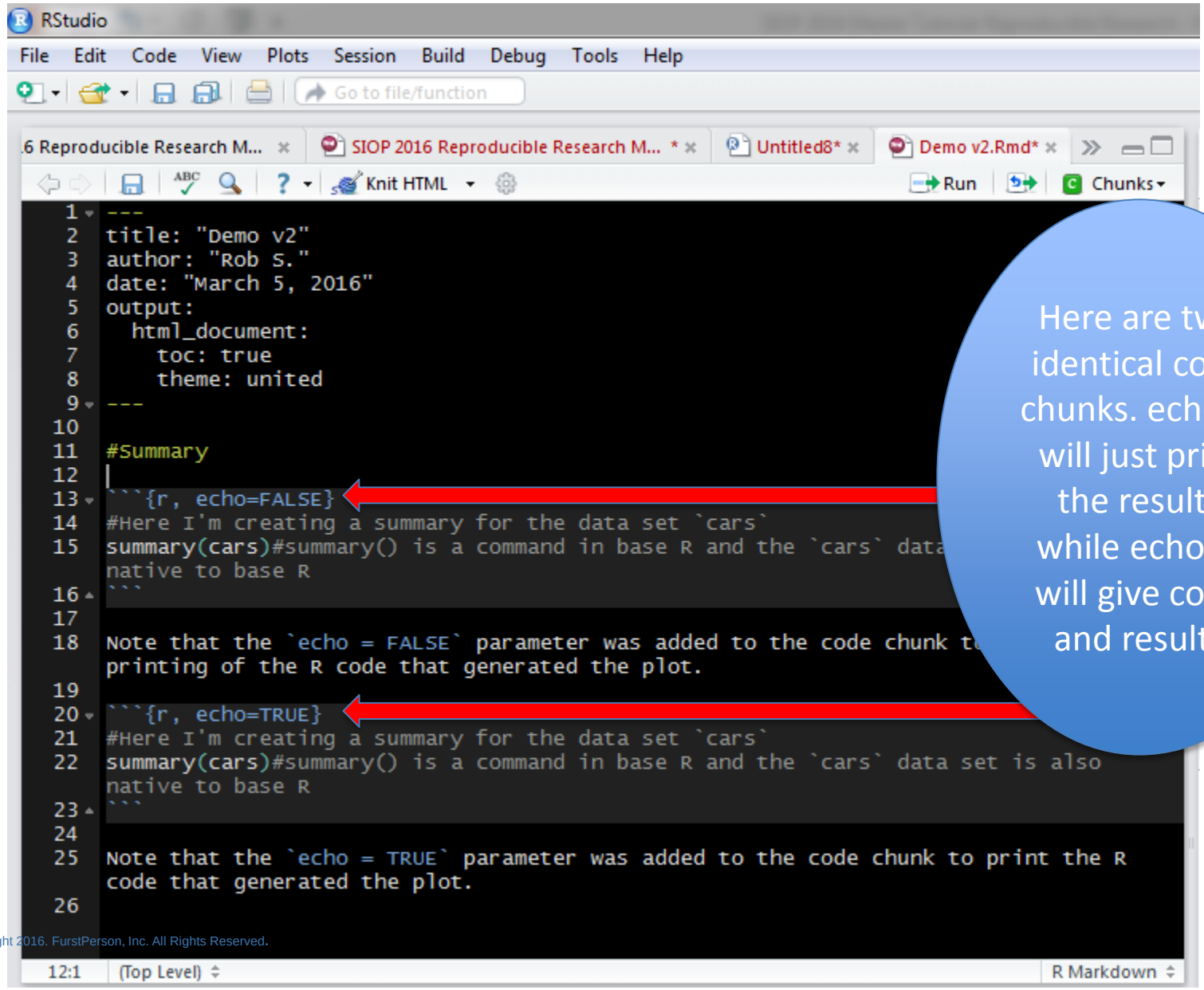
# Formatting Code Chunks

FurstPerson.

- Code Chunks can be formatted to display many different layers
- Within the ```r{}```, set the following to either TRUE or FALSE
  - eval=run the code or not
  - echo=display the code or not
  - error=display error messages or not
  - message=display messages or not
  - warning=display warnings or not
- results-set to 'hide' if you want to hide the results
- For example, you may set echo=TRUE to display the code for a tutorial but echo=FALSE for a report to a client
- You may set eval=FALSE within a tutorial if you are giving many examples but don't want to crowd the document with output

# R Markdown-Formatting

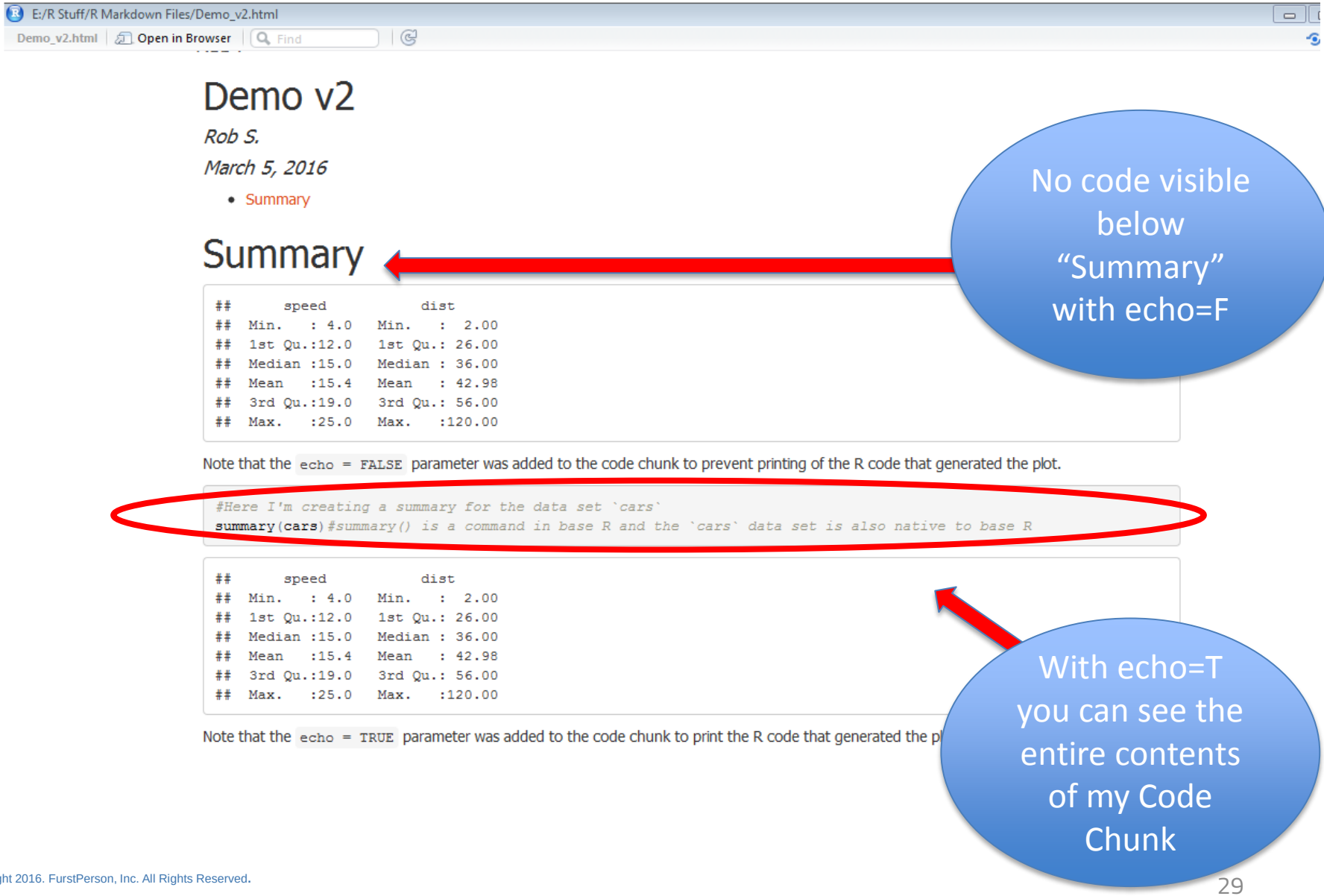
FurstPerson.




Here are two identical code chunks. echo=F will just print the results while echo=T will give code and results

# R Markdown-Formatting

FurstPerson



FurstPerson

# Questions?

FurstPerson

# Learning Objective #2: Publishing your Reproducible Research Document in HTML, Word, or PDF

# Locally Publishing R Markdown

FurstPerson

- So far we've seen how to generate a HTML, Word, or PDF file from R Markdown
- This section will focus how to go from initial analysis via R Script to R Markdown document to distribute to others
- We will answer the following questions
  - How do I add in bibliography information?
  - How do I go from the very beginning of a project to having something I can revisit later and remember what I did?
  - How do I make tables that look nice and are easy to read?
  - When sharing my document, how can I have R automatically load all of the necessary packages for the analysis?
  - What is are the R packages I probably don't know about but are essential to nice looking documents?

# Mock Study

FurstPerson

- This study will use a subset of the data from:
- Johnson, J. A. (2014). Measuring thirty facets of the five factor model with a 120-item public domain inventory: Development of the IPIP-NEO-120. J. of Research in Personality, 51, 78-89.
- Original study data can be found here
  - https://osf.io/wxvth/
- Data for the Mock Study can be found here
  - https://github.com/RobStilson/SIOP_2016_Master_Tutorial

# Syntax Color Scheme

FurstPerson

- Any syntax in the presentation will be color coded
- ```r{}
- summary(cars)
- ```

GREEN means code chunk R syntax or R Script

- `r nrow (mydata)`

BLUE means inline code R syntax

- Click here

Purple means hyperlink R syntax

# Mock Study

FurstPerson

- Load into R with the following code:
- #tryCatch code to load all necessary packages (Thanks Mike G.!)
- tryCatch(require(RCurl),finally=utils:::install.packages(pkgs='RCurl',repos='http://cran.r-project.org'));
- require(RCurl)
- library(RCurl)
- x <- getURL("https://raw.githubusercontent.com/RobStilson/SIOP_2016_Master_Tutorial/master/mydata_10k.csv")
- mydata_10k <- read.csv(text = x)
- Now run (not Knit) in your R script

# Mock Study

FurstPerson

- We will take a random sample of 1000
- Use the `set.seed()` command to make sure it takes the same random sample each time
- set.seed(1)
- mydata <- mydata_10k[ sample( which(mydata_10k$country=='USA'), 1000, replace=FALSE),]
- #By setting the seed to a specified number it makes the sampling reproducible
- #Below we will take another sample of 1000 and save to a different data set. The set seed will be the same so it will take the exact same
- set.seed(1)
- mydata2 <- data[ sample( which(data$country=='USA'), 1000, replace=FALSE),]
- rm(mydata2) #This removes the data set since we no longer need it

# Mock Study

FurstPerson

- The rest of the R code will be available on Github
- It will not be reproduced here, but will be demonstrated
- Overview of mock study
  - Take subset of only items (no ID info)
  - Rename items
  - Create keys
  - Score items and get alpha reliability
  - Create overall scale scores
  - Present results

# Mock Study-Bibliography

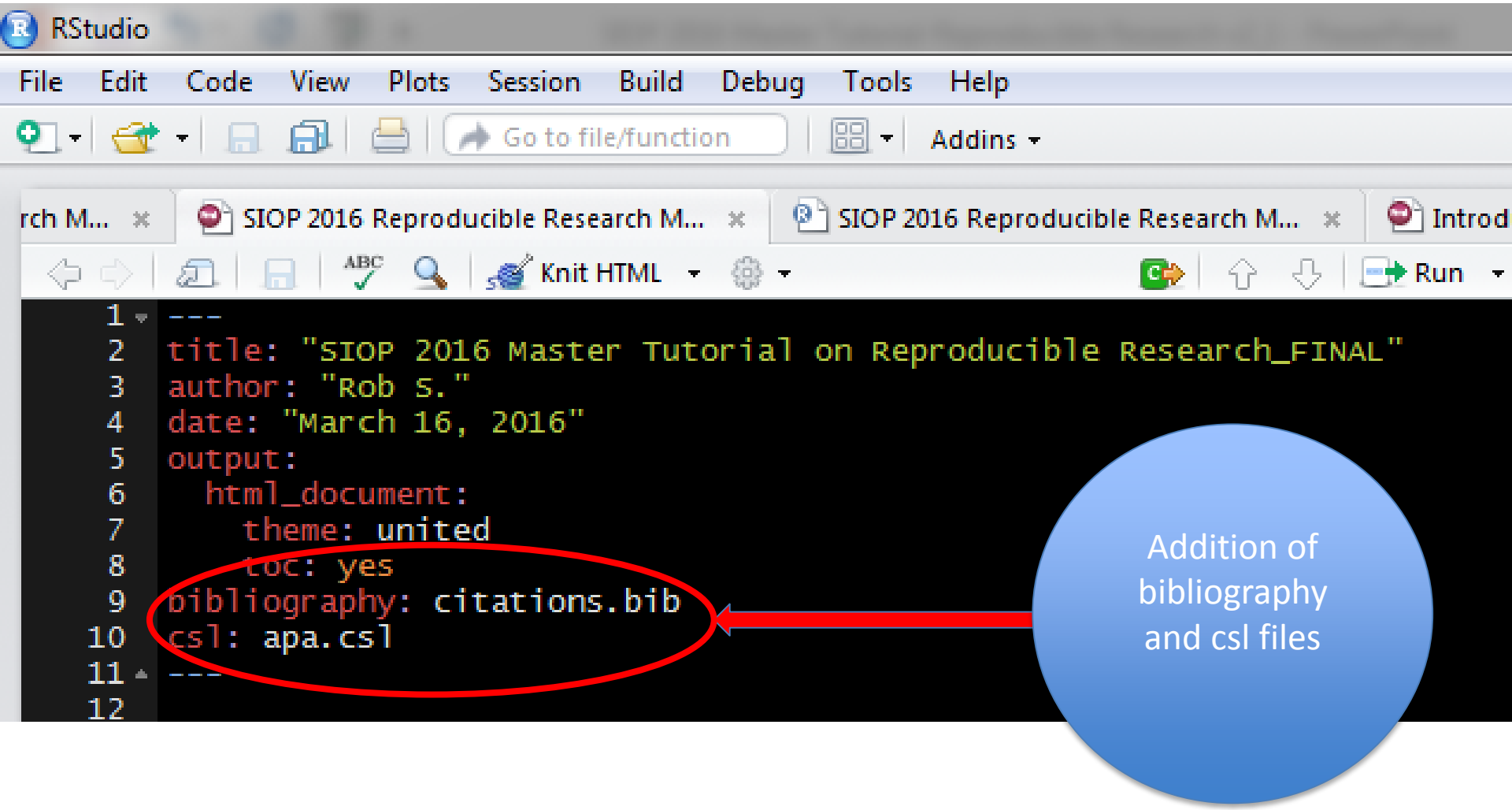
FurstPerson

- Now we need to cite where we got data
- Need additional files for bibliography
- Add them into Title portion of R Markdown document
- Additional files are
  - .bib-bibliography file
  - .csl-citation style file
- Overview of process on R Studio site here
  - http://rmarkdown.rstudio.com/authoring_bibliographies_and_citations.html
- Tutorial using APA format is here
- Make sure to save .bib and .csl files into same folder as R Markdown document

# Mock Study-Bibliography

FurstPerson

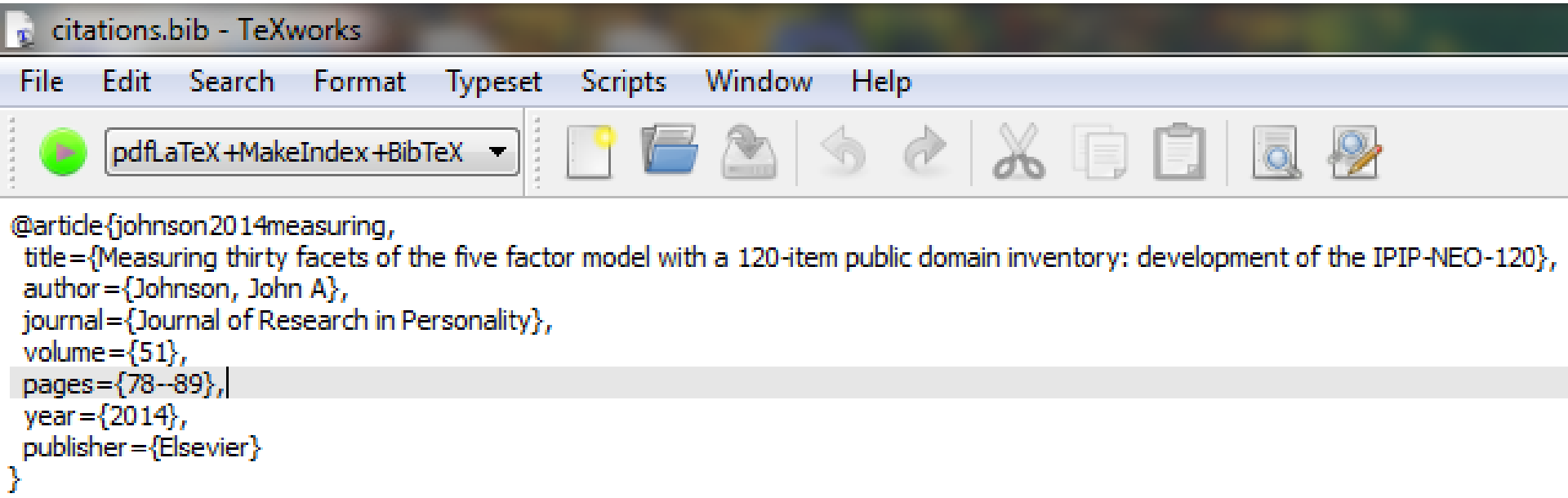
- In the current file, it will now look like this:



# Mock Study-.bib file

FurstPerson

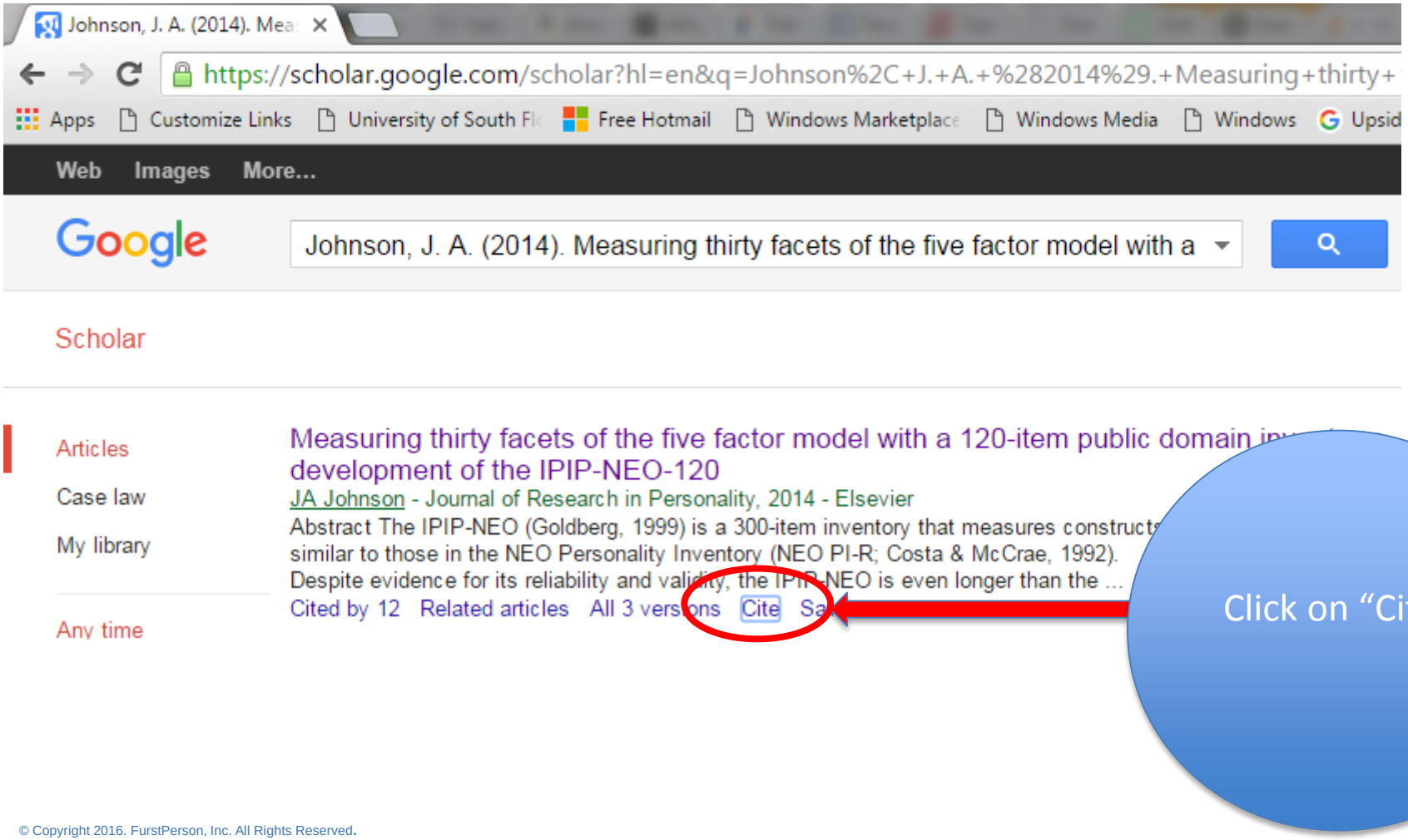
- Specific format needed for .bib file
- Scholar.google.com makes this easy



# Mock Study-Citation for Mock Study

FurstPerson.

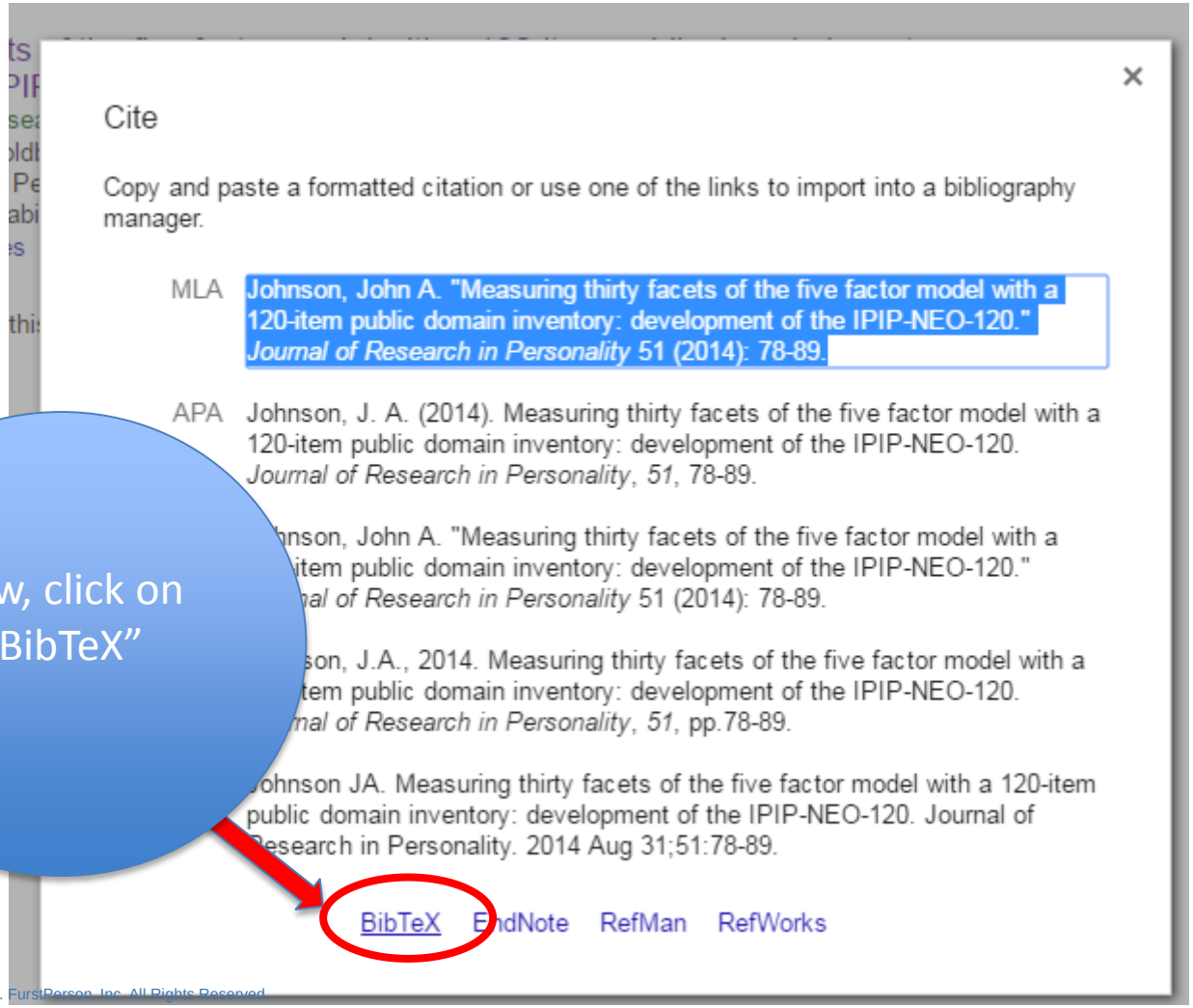
- Look up study on Google Scholar



# Mock Study-Citation for Mock Study

FurstPerson.

- Brings up "Cite" window

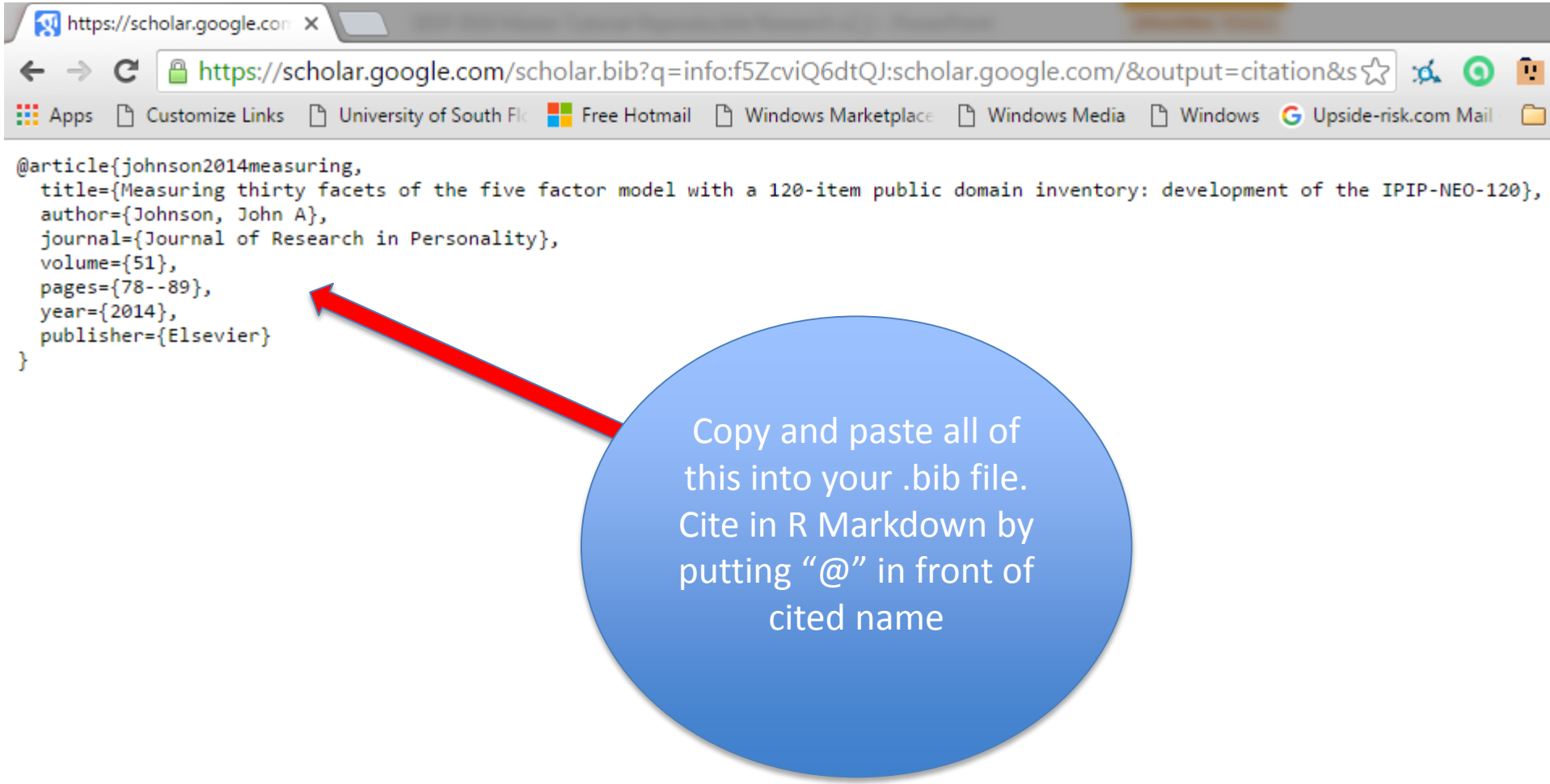



Now, click on "BibTeX"

# Mock Study-Citation for Mock Study

FurstPerson

- This should look familiar



# Mock Study-Reference Section

FurstPerson

- R Markdown automatically puts .bib output at end of document
- Simply end document with ##References and .bib output will be put below

## References


Costa, P. T., & McCrae, R. R. (1992). Neo pl-r professional manual.

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. Psychometrika, 16(3), 297–334.

Johnson, J. A. (2014). Measuring thirty facets of the five factor model with a 120-item public domain inventory: Development of the iPIP-nEO-120. Journal of Research in Personality, 51, 78–89.

Nunnally Jr, J. C. (1970). Introduction to psychological measurement.


Here is the citation among others

# Tables and Data Visualization

FurstPerson

- Base R can present tables and graphs in R Markdown
- However, not very visually appealing
- How to fix this?
  - sjPlot
  - pander
  - Stargazer
  - Xtable
  - Kable

# Base R vs. sjPlot

FurstPerson

- Will focus on sjPlot for HTML and pander for Word and PDF for the tutorial
- Base R can present tables in R Markdown
- Much more presentable with sjPlot (see below)
- Learn more about sjPlot here

```
## Call: scoreItems(keys = keys, items = items)
##
## (Unstandardized) Alpha:
##        AGR  CON  EXT  NEU  OPN
## alpha 0.59 0.46 0.63 0.69 0.12
##
```

Here is a more succint table in a better looking format.

Table 3: Alpha reliabilities

Alpha Reliability of Scales

| *Variable* | *AGR* | *CON* | *EXT* | *NEU* | *OPN* |
|---|---|---|---|---|---|
| alpha | 0.587 | 0.463 | 0.635 | 0.691 | 0.122 |

# sjPlot in R Markdown

FurstPerson

- Here is the code to create the sjPlot table
- Below, the data is rounded to 3 digits and transformed to a df
- scores_alpha <- round(as.data.frame(scores$alpha), 3)
- Below is the specific way to wrap the sjPlot (sjt.df) code to get it to present correctly in R Markdown
- scores_alpha_table <- sjt.df(scores_alpha, title = "Alpha Reliability of Scales", alternateRowColors=TRUE, useViewer=F, describe=FALSE,encoding = "UTF-8", no.output=TRUE, hideProgressBar = TRUE)$knitr
- The above code is in the ```r{}``` code chunk.
- We then use the following inline code to actually present the table
- `r scores_alpha_table`
- Adding "###Table 3: Alpha reliabilities" as a third level header

# sjPlot plotting in R Markdown

FurstPerson

- This method can be used for any of the many sjPlot commands like:
  - sjp.frq – plot frequencies of (count) variables
  - sjp.glm – plot odds ratios (forest plots) of generalized linear models
  - sjp.glmer – plot odds ratios (forest plots) of generalized linear mixed effects models
  - sjp.grpfrq – plot grouped or stacked frequencies
  - sjp.int – plot interaction effects of regression models
  - sjp.likert – plot Likert scales
  - sjp.lm – plot estimates of linear models
  - sjp.lmer – plot estimates (forest plots) of linear mixed effects models
  - sjp.pca – plot principal component analysis
  - sjp.scatter – plot (grouped) scatter plots
  - sjp.stackfrq – plot stacked proportional bars
  - sjp.xtab – plot contingency tables

# sjPlot tables in R Markdown

FurstPerson

- This method can be used for any of the many sjPlot commands like:
  - sjt.corr – correlations as HTML table
  - sjt.df – (description of) data frames as HTML table
  - sjt.frq – frequencies of (count) variables as HTML table
  - sjt.lm – linear models as HTML table
  - sjt.pca – principal component analysis as HTML table
  - sjt.stackfrq – stacked frequencies as HTML table
  - sjt.xtab – contingency tables as HTML table

# pander in R Markdown

FurstPerson

- There may be a way to render sjPlot in PDF and Word and still have it look nice
- I have not found it
- Enter pander!
- Pander makes tables and figures pretty in:
  - HTML
  - PDF
  - Word
- Not as customizable as sjPlot (that I know of)

FurstPerson

# Questions?

FurstPerson

# Learning Objective #3: Updating the data used in your document

# Updating data in R Markdown

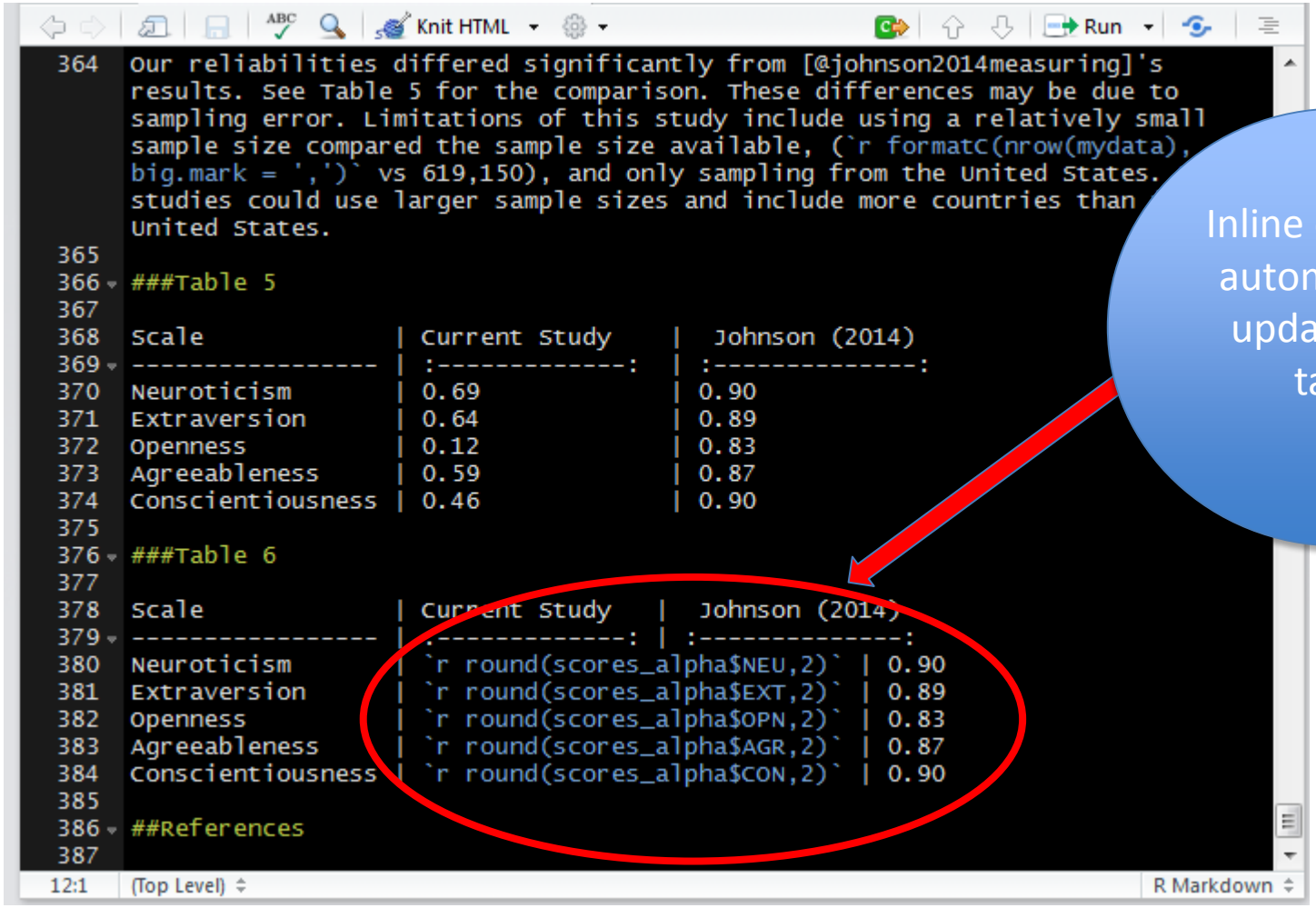
FurstPerson

- Now that we know how to create aesthetically pleasing tables what happens when new data comes in?
  - Simply add the data to your original file
  - Rename new data to original data file name
- Knit again and everything will be updated
  - Render any number in text with code if possible

# Updating data in R Markdown

FurstPerson

- You can use inline code within tables



FurstPerson

# Questions?

FurstPerson

# Learning Objective #4: Publishing your findings on the web via RPubs

# RPubs

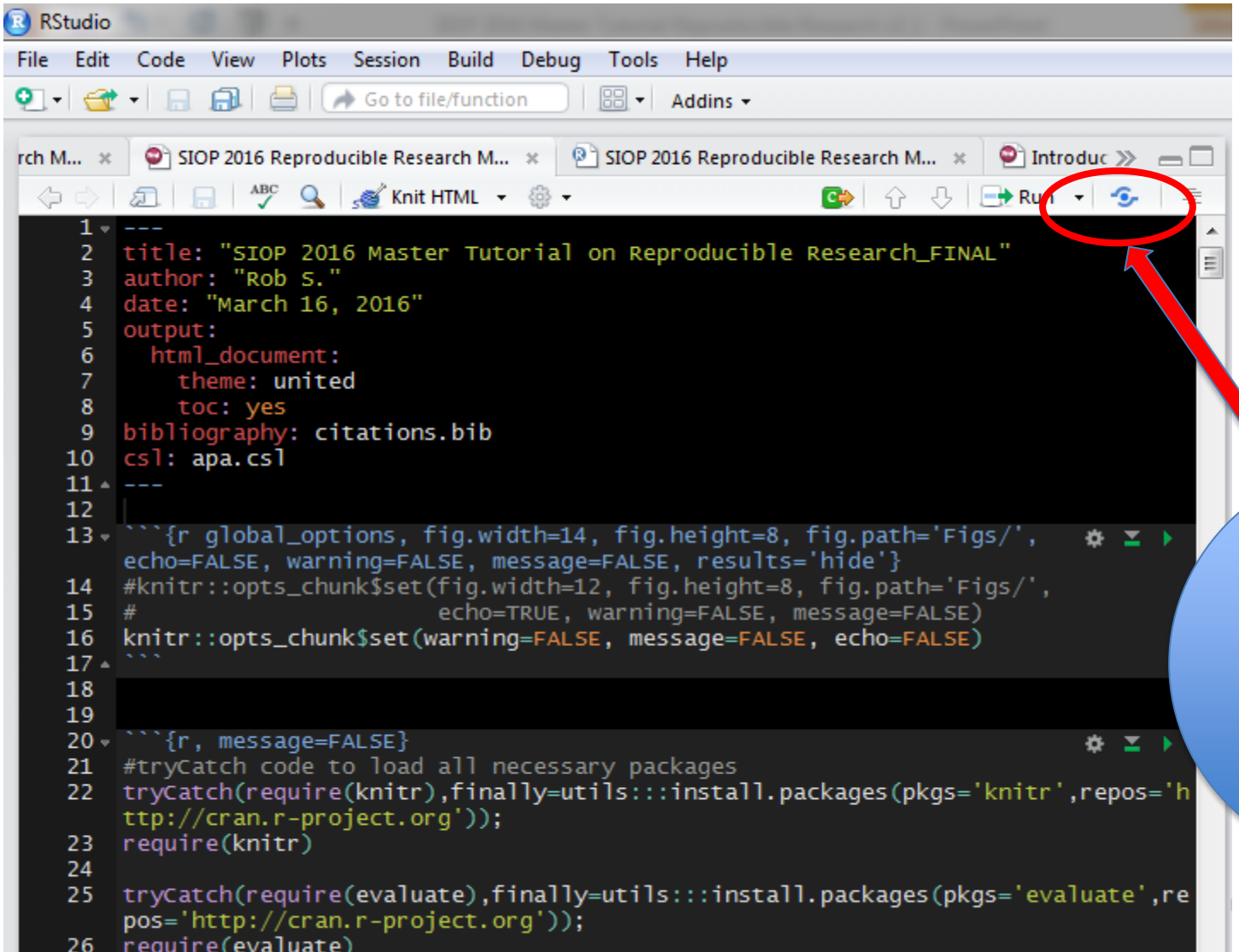
FurstPerson

- Share your documents on RPubs.com
- You will need to create a free account
- Remember that ANYBODY can see this work
- Good place to go to see potential of R Markdown documents

# RPubs

FurstPerson.

- Can publish directly from R Studio



FurstPerson

# Questions?

FurstPerson

# Summary and Wrap Up

# What did we learn

FurstPerson

- How to create R Markdown Documents in R Studio
  - Inline code
  - Formatting text
  - Code chunks
  - Formatting Code chunks
  - sjPlot and pander within R Markdown
  - Publishing in HTML, Word, and PDF
  - How to create a bibliography in R Markdown
  - How to update existing docs with fresh data
  - How to publish to Rpubs.com

# How you can get started

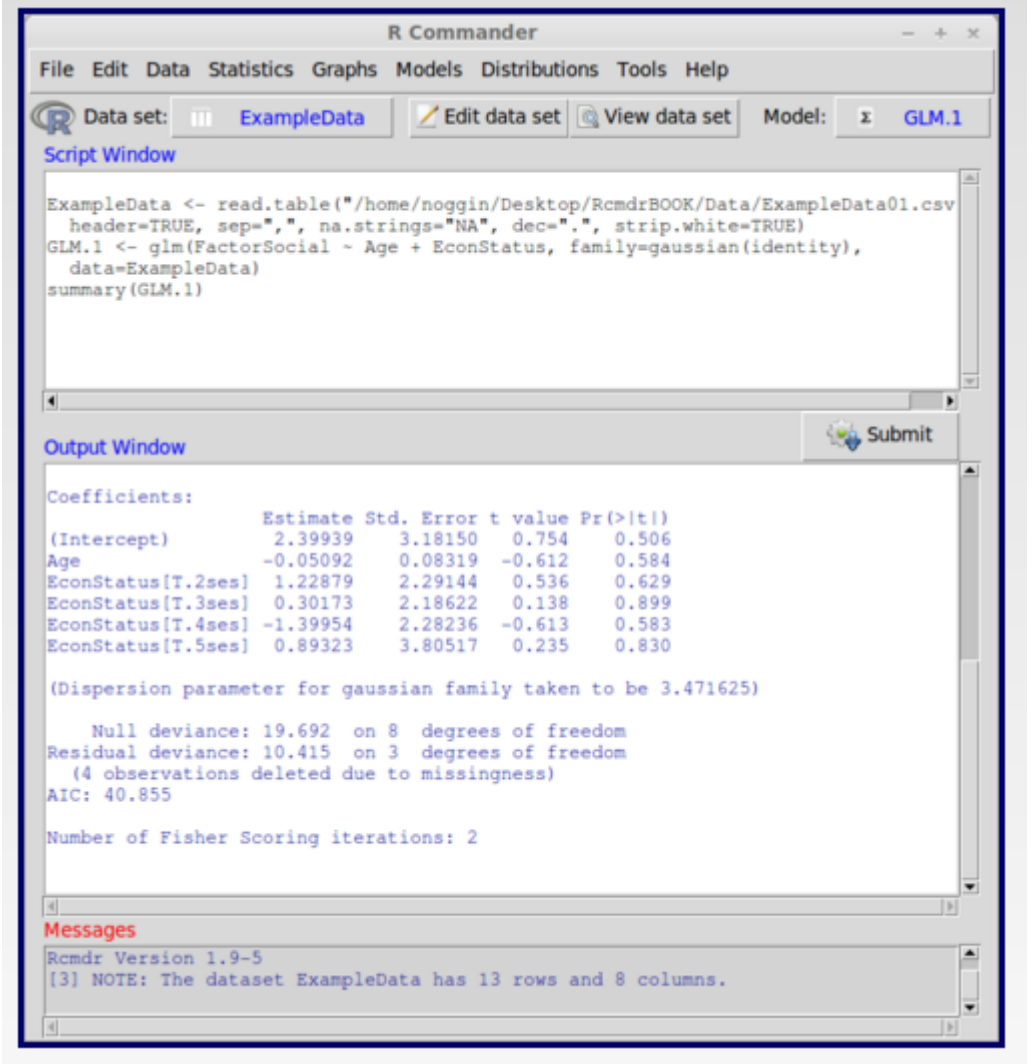
FurstPerson

- If you know nothing about R
- Data Camp
- Introduction to R Class
  - Walks you through R from the beginning
  - Provides helpful hints and exercise solutions
- Create R Markdown documents of the class
  - Add you own comments to the docs
  - Try to exactly recreate the format

# How you can get started

FurstPerson

- If you know nothing about R and you don't like syntax
- R Commander
- Menu driven like SPSS
- Creates R Markdown script as you do analysis
- Integrates with R Studio



# How you can get started

FurstPerson.

- Start your next project in R
- This is a great way to learn
- Get as far as you can before switching to a format you know better (e.g., loaded and cleaned data, but don't know how to do LM)
- Make notes of where you got stuck in R so you can figure out for next project
- Lather, rinse, repeat

# How you can get started

FurstPerson

- If you know some R or
- If no project springs to mind
- Go to www.twotorials.com
  - Put together by Anthony Damico
  - 90+ ~2 minute video tutorials on R
  - Examples include
    - 013 how to read spss, stata, and sas files into r
    - 014 how to read an excel file (dot xls and dot xlsx) into a data frame with r
    - 016 how to add comments, save a script file, and make your work reproducible in r
    - 022 how to generate basic descriptive statistics - means, medians, sums, quantiles - on data tables in r

# How you can get started

FurstPerson.

- Follow along with the videos
- Make a transcript of what is being done
- Copy the code being created
- Now you have a mini-tutorial
- Make these your own
  - Comments
  - Multiple iterations of code (e.g., with and without missing data, etc.)
  - Multiple iterations of graphs (e.g., different color schemes, legend or not, etc.)

FurstPerson

# Questions?
# https://github.com/RobStilson

FurstPerson

# Thank You!

Rob Stilson
Sr. Manager, R&D
Rob.Stilson@furstperson.com

FurstPerson

# Appendix

# Additional Information

FurstPerson

- Embedding RData files in R Markdown files for more reproducible analyses
- Packrat-packrat is a package dependency management program for R
- Making Reproducible Research Enjoyable