# Intro to R

## Data Input

# Day 1 Review

- the RStudio Editor (top) is for static code like scripts or R Markdown documents
- The console is for testing code (bottom) - best to save your code though!
- R code goes within what is called a chunk (the gray box with a green play button)
- **Objects** (like nouns) are data or variables.

# Day 1 Review

- R functions as a calculator
- Use `<-` to save (assign) values to objects
- **Functions** (like verbs) perform specific tasks in R and are found within packages
- Use `c()` to **combine** vectors
- `length()`, `class()`, and `str()` tell you information about an object
- Install packages with `install.packages()`
- Load packages with `library()`
- Get help with `?` or help pane

# Day 1 Review

- Make sure we have installed and loaded the `tidyverse` package!

# Outline

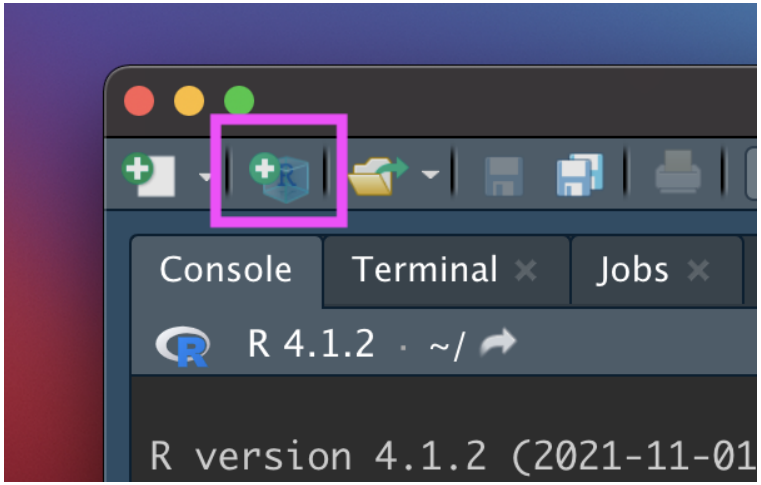
- Part 0: A little bit of set up!
- Part 1: reading in manually (point and click) (.csv)
- Part 2: checking data & multiple file formats (.xlsx)

# Part 0: Setup - R Project

# New R Project

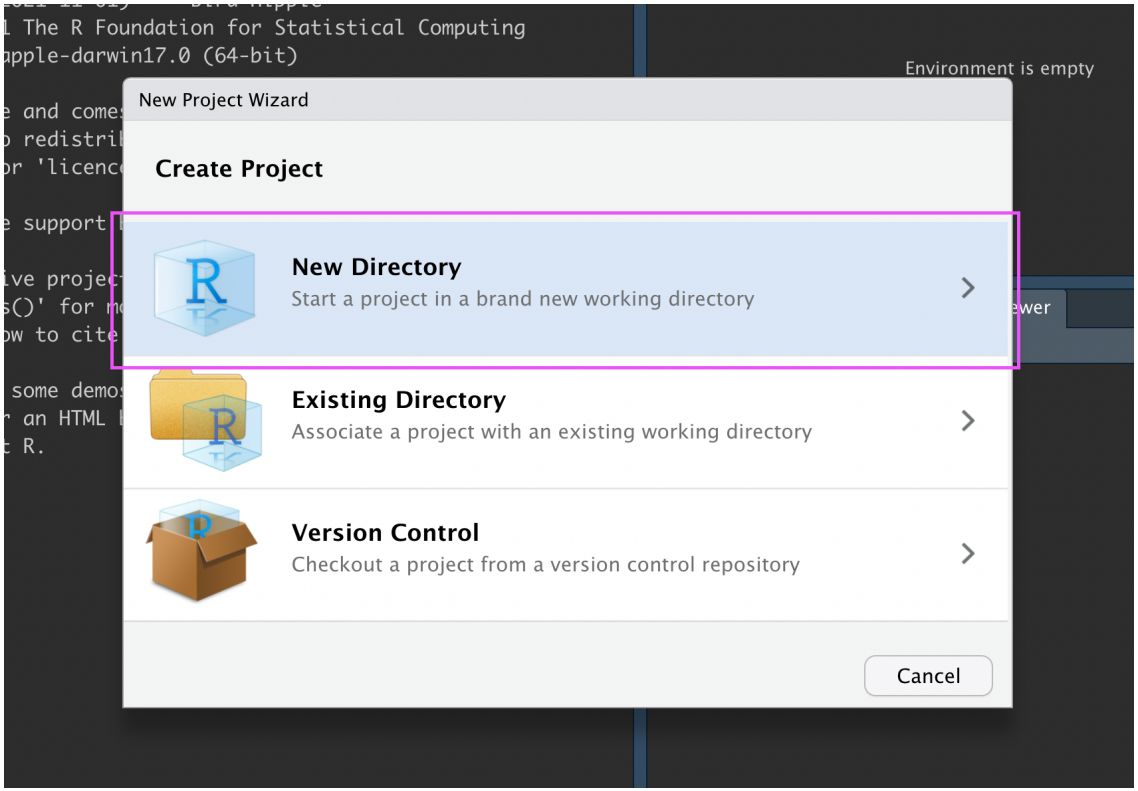
Let's make an R Project so we can stay organized in the next steps.

Click the new R Project button at the top left of RStudio:



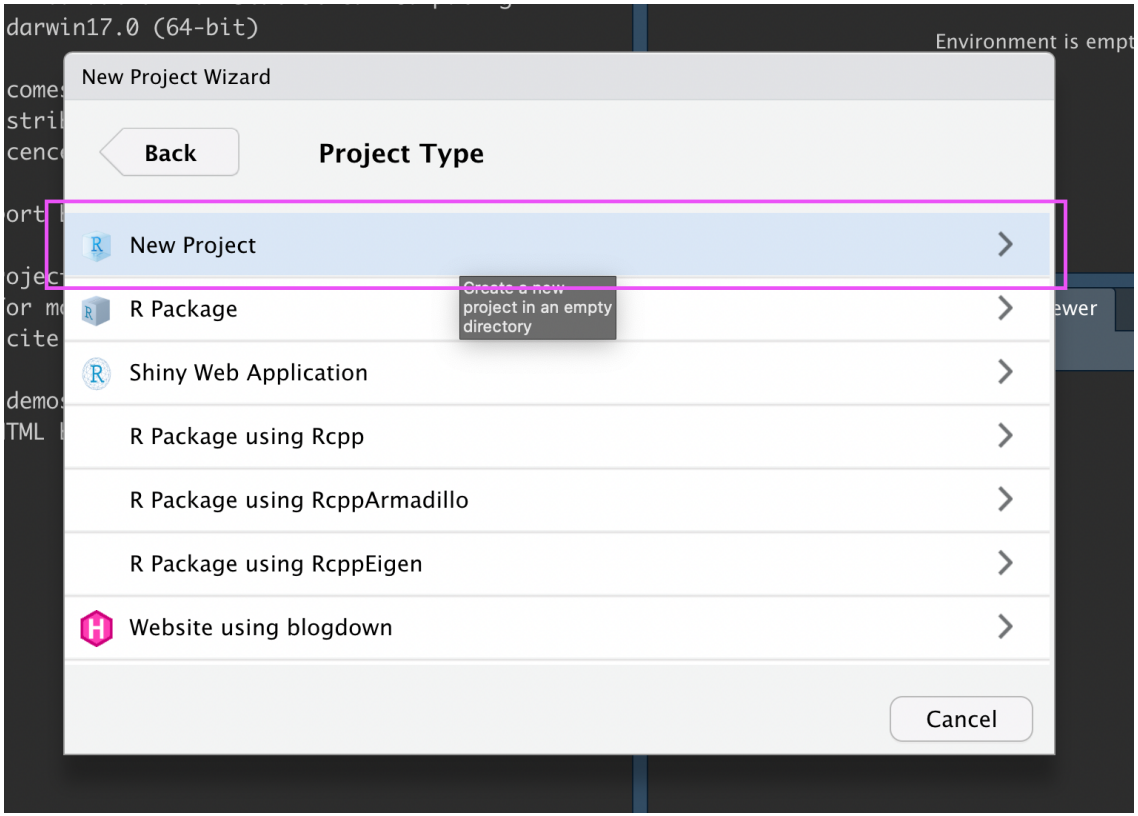
# New R Project

In the New Project Wizard, click “New Directory”:



# New R Project

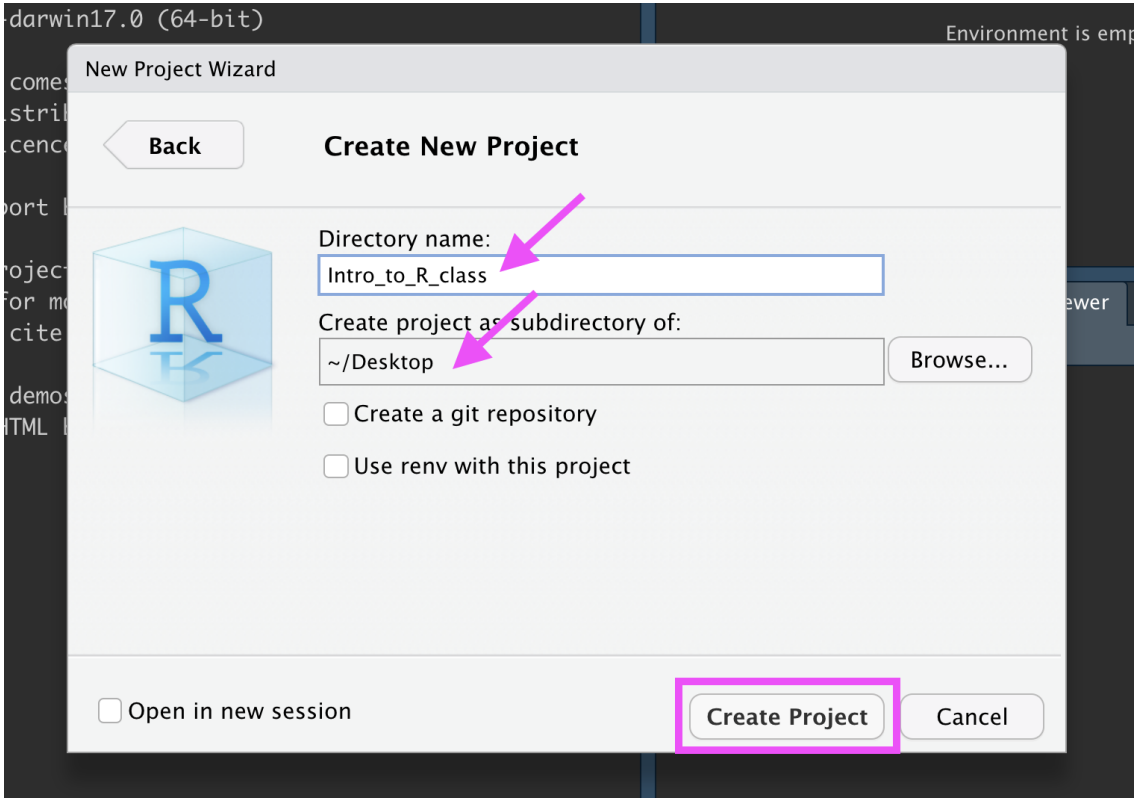
Click “New Project”:



# New R Project

Type in a name for your new folder.

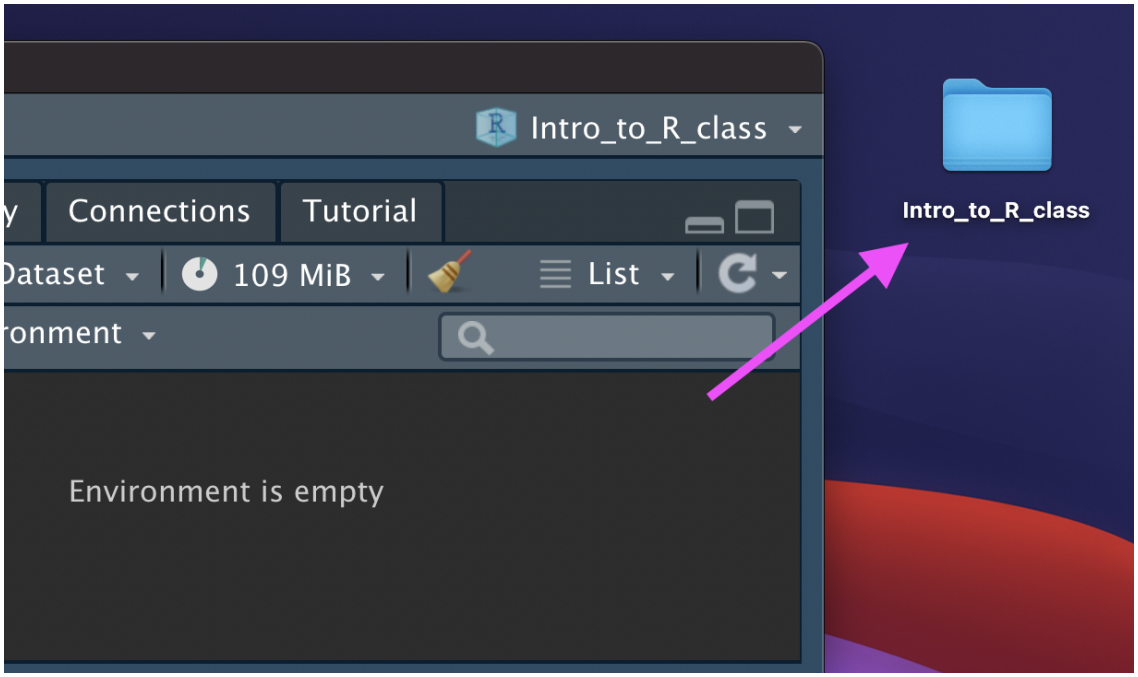
Store it somewhere easy to find, such as your Desktop:



# New R Project

You now have a new R Project folder on your Desktop!

Make sure you add any scripts or data files to this folder as we go through today's lesson. This will make sure R is able to "find" your files.



# Part 1: Getting data into R (manual/point and click, .csv)

# Data Input

- 'Reading in' data is the first step of any real project/analysis
- R can read almost any file format, especially via add-on packages
- We are going to focus on simple delimited files first
    - comma separated (e.g. '.csv')

# Data Input

Youth Tobacco Survey (YTS) dataset:

"The YTS was developed to provide states with comprehensive data on both middle school and high school students regarding tobacco use, exposure to environmental tobacco smoke, smoking cessation, school curriculum, minors' ability to purchase or otherwise obtain tobacco products, knowledge and attitudes about tobacco, and familiarity with pro-tobacco and anti-tobacco media messages."

- Check out the data at: https://catalog.data.gov/dataset/youth-tobacco-survey-yts-data

# Data Input: Dataset Location

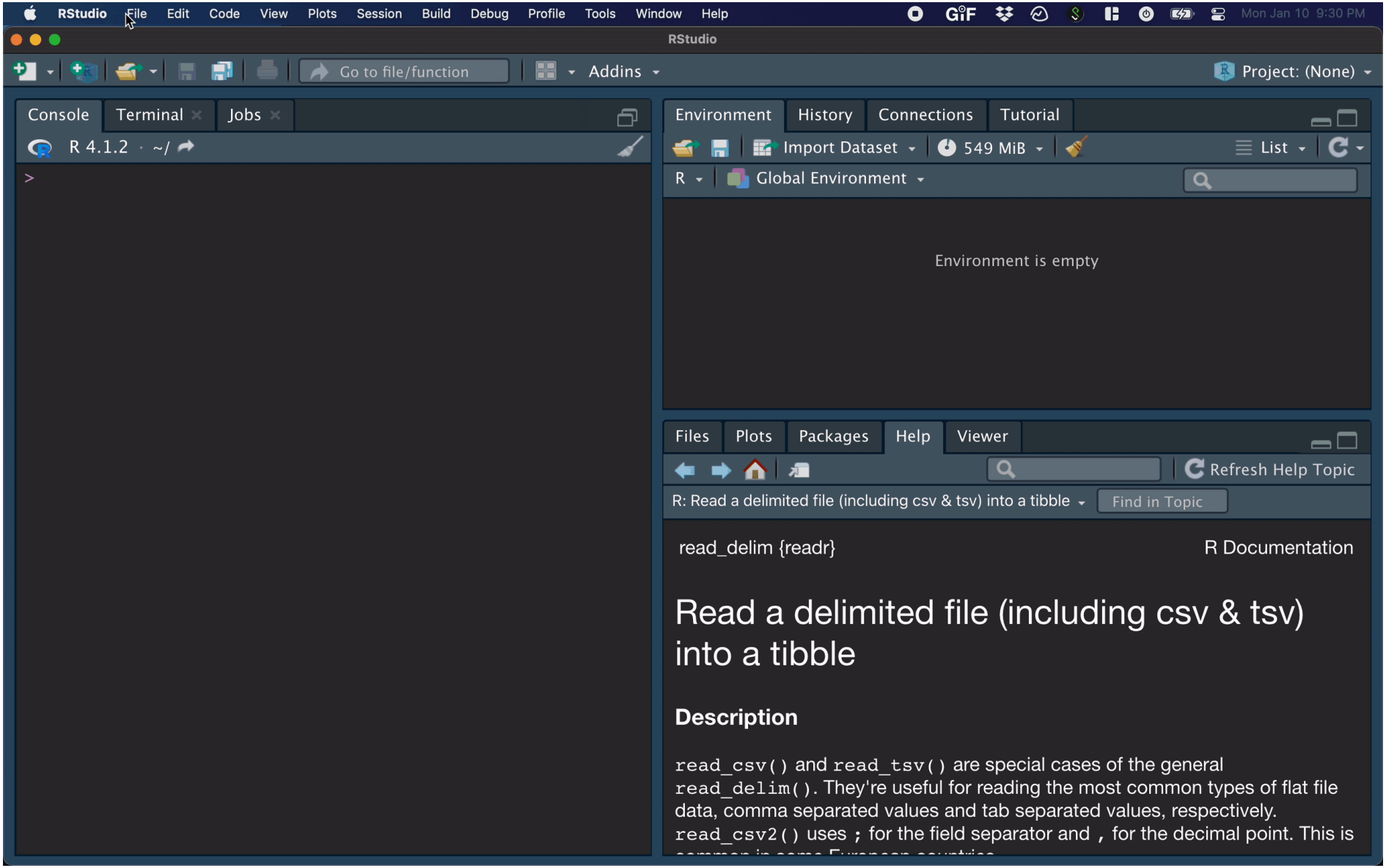
Dataset is located at
http://jhudatascience.org/intro_to_r/data/Youth_Tobacco_Survey_YTS_Data.csv

- Download data by clicking the above link
    - Safari - if a file loads in your browser, choose File –> Save As, select, Format “Page Source” and save

# Import Dataset

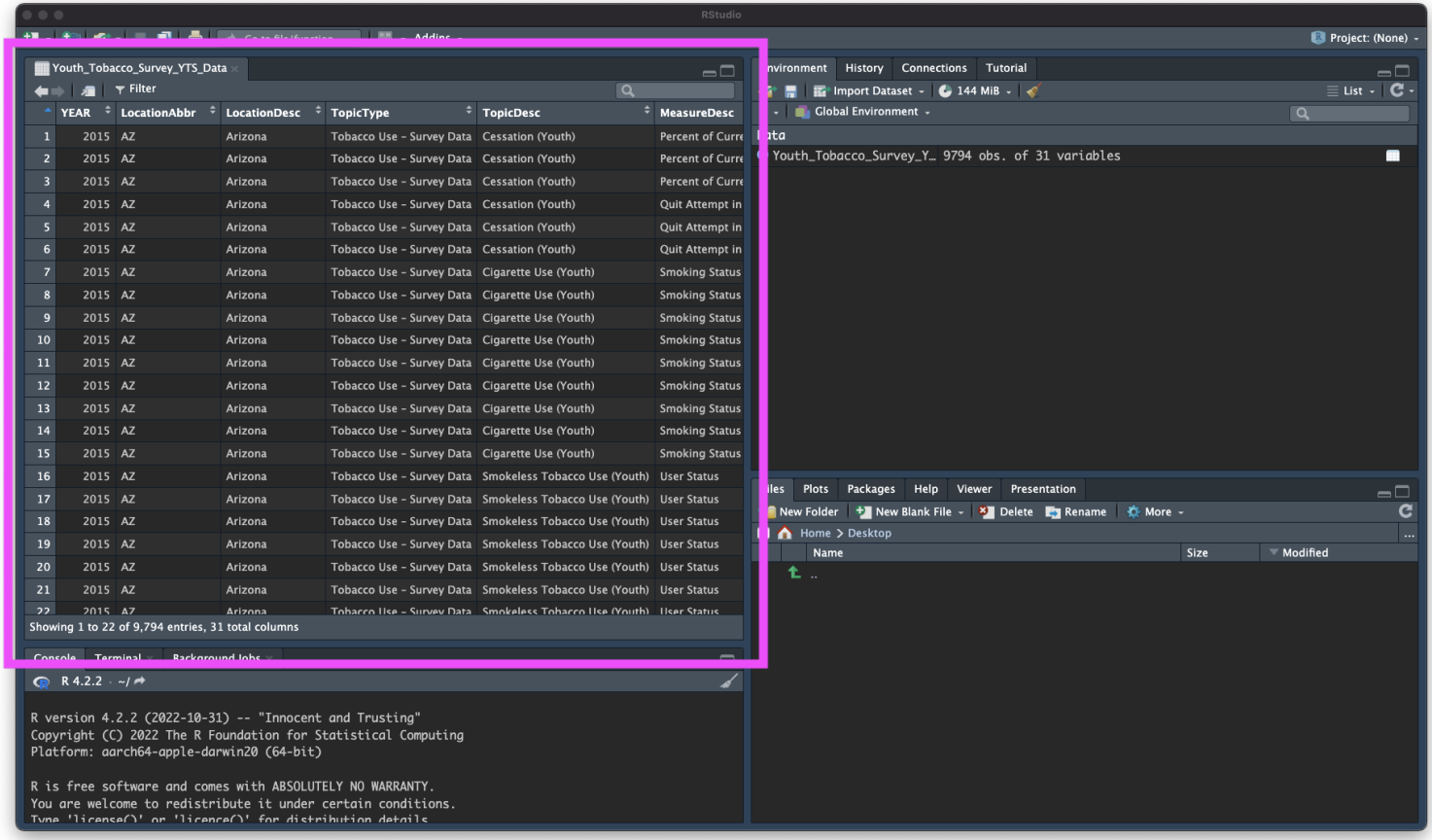
- > File
- > Import Dataset
- > From Text (`readr`)
- > paste the url (http://jhudatascience.org/intro_to_r/data/Youth_Tobacco_Survey_YTS_Data.csv)
- > click “Update” and “Import”

# Import Dataset



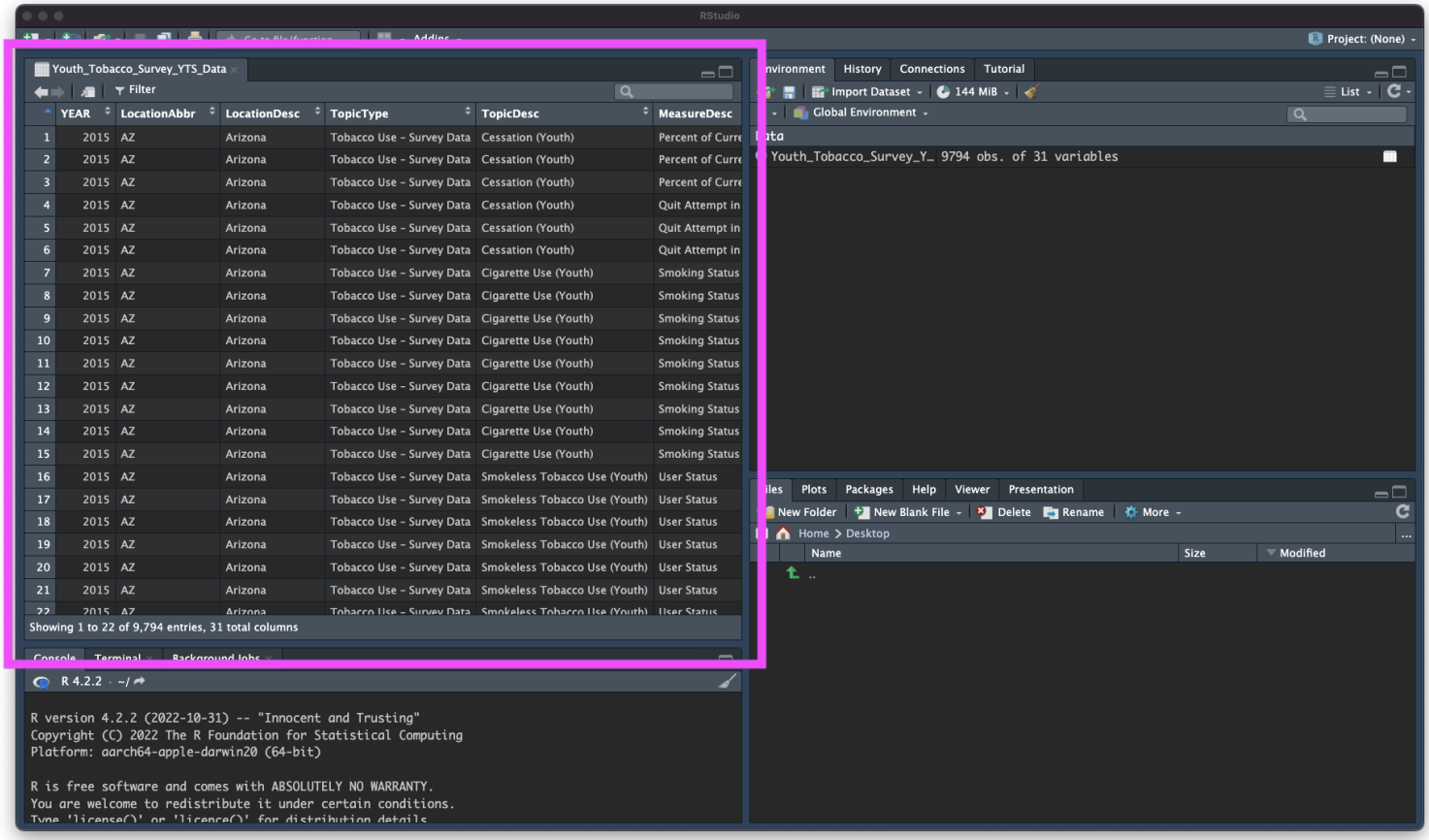
# What Just Happened?

You see a preview of the data on the top left pane.



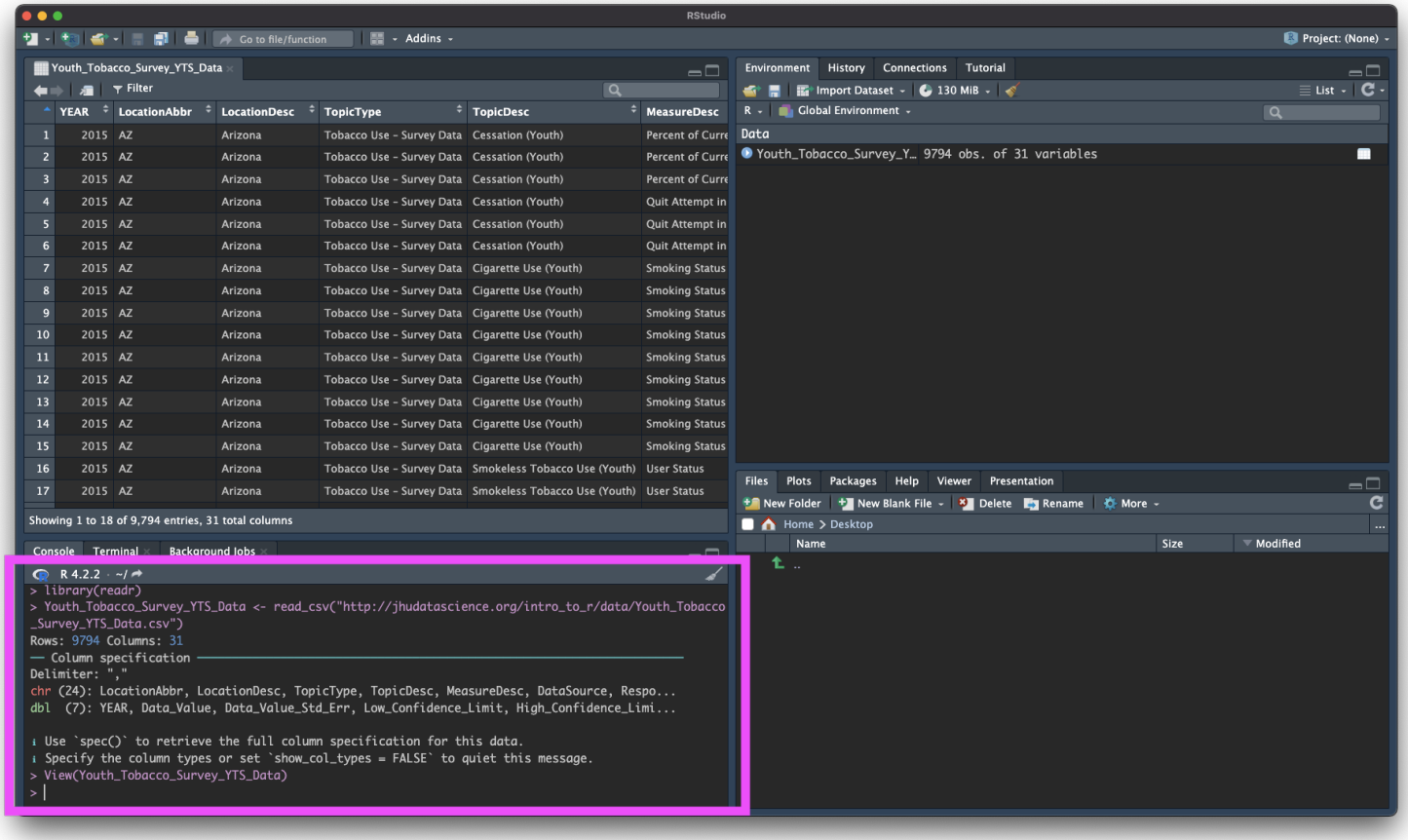
# What Just Happened?

You see a new object called `Youth_Tobacco_Survey_YTS_Data` in your environment pane (top right). The table button opens the data for you to view.

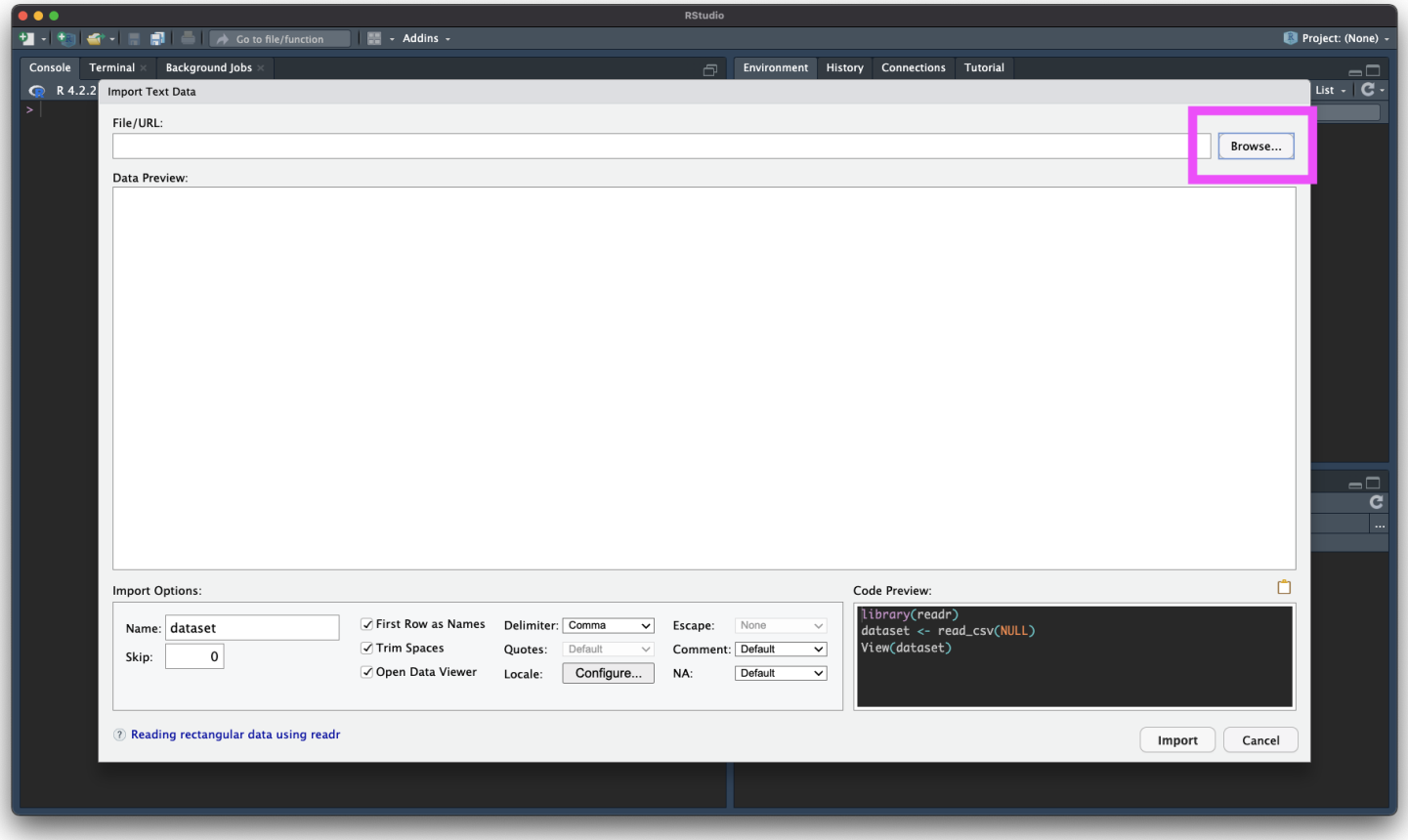


# What Just Happened?

R ran some code in the console (bottom left).



# Browsing for Data on Your Machine



## Summary

Review the process: `https://youtu.be/LEkNfJgpunQ`

- > File
- > Import Dataset
- > From Text (`readr`)
- > paste the url (http://jhudatascience.org/intro_to_r/data/Youth_Tobacco_Survey_YTS_Data.csv)
- > click “Update” and “Import”

# Let’s practice!

# Importing "states" data

- Try downloading the dataset located here: https://hutchdatascience.org/SeattleStatSummer_R/data/states.csv
- Use the File > Import Dataset > from Text (`readr`)
- Browse for the downloaded file on your machine
- Inspect the code that was run. Copy this code into your R Markdown document for later!

# Looking at the code

```
library(readr)
states <- read_csv("~/Downloads/states.csv")
View(states)
```

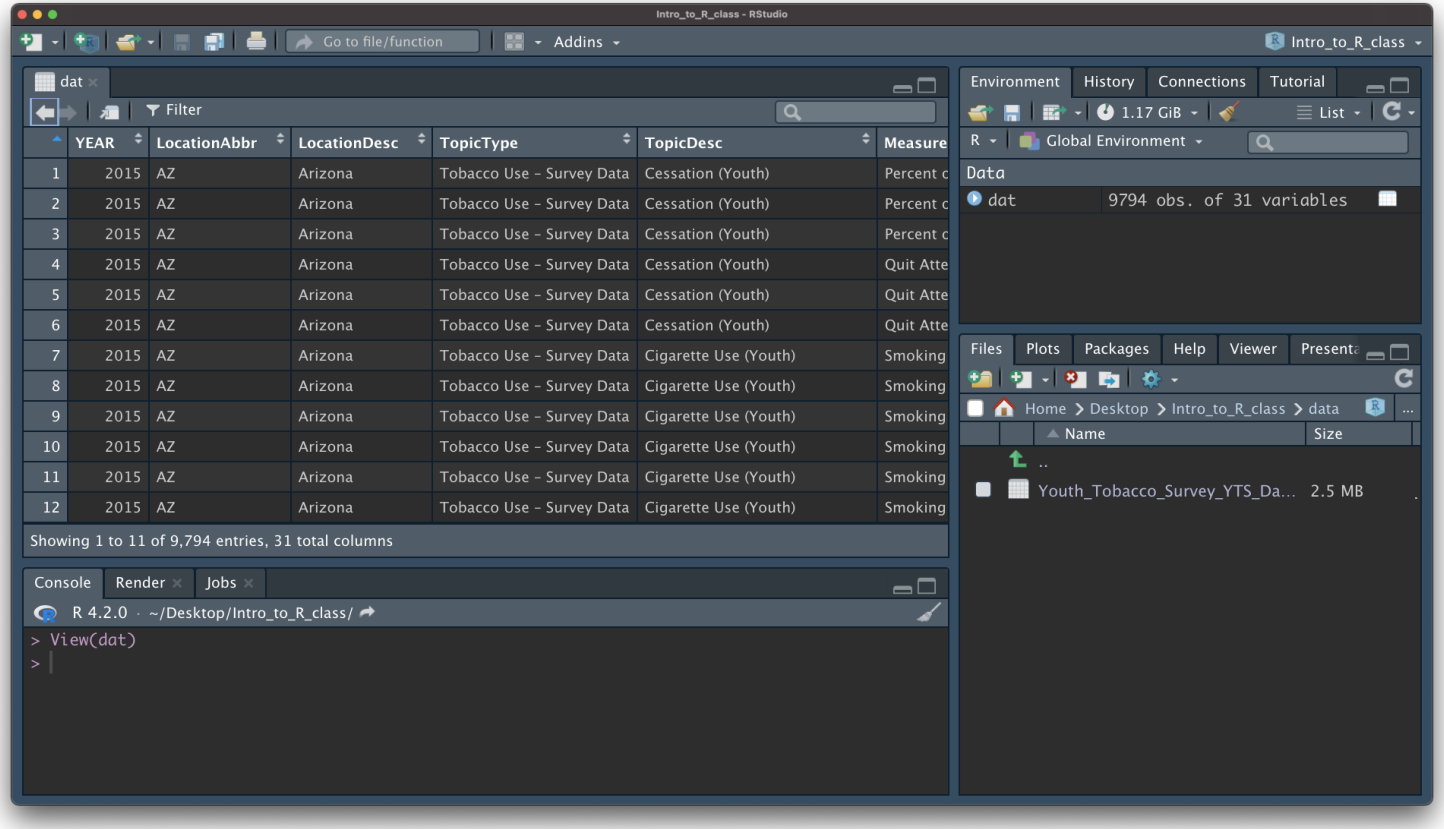
Notice that the part in quotes is a location on my computer. If I move the file, I won't be able to use the same code again. Better to move that file to my project folder (instead of my downloads for example) for longer-term storage.

# Part 2: Checking data & Other formats

# Data Input: Checking the data

- the `View()` function shows your data in a new tab, in spreadsheet format
- be careful if your data is big!

```
View(states)
```



# Data Input: Checking the data

The `str()` function can tell you about data/objects(different variables and their classes - more on this later).

```
str(states)
```

```
spec_tbl_df [52 × 14] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ entity                    : chr [1:52] "Alabama" "Alaska" "Arizona" "Arkansas" ...
 $ state_abb                 : chr [1:52] "AL" "AK" "AZ" "AR" ...
 $ state_area_sq_miles       : num [1:52] 51609 589757 113909 53104 158693 ...
 $ state_division            : chr [1:52] "East South Central" "Pacific" "Mountain" "West South
 $ state_region              : chr [1:52] "South" "West" "West" "South" ...
 $ population                : num [1:52] 4903185 731545 7278717 3017804 39512223 ...
 $ births_in_2021            : num [1:52] 58054 9367 77916 35965 420608 ...
 $ fertility_rate_per_1000   : num [1:52] 59.5 64.9 55.5 61.7 52.8 52.5 52.1 56.5 54.9 55.9 ...
 $ cesarean_percent          : num [1:52] 35.1 24.2 28.7 34.3 30.8 27.3 35.4 31.9 35.8 35.1 ...
 $ life_expect               : num [1:52] 73.2 76.6 76.3 73.8 79 78.3 78.4 76.7 77.5 75.6 ...
 $ cancer_rate_per_100000    : num [1:52] 160 156 135 168 132 ...
 $ cancer_mortality          : num [1:52] 10429 1093 12813 6516 59503 ...
 $ Administered_Dose1_Pop_Pct: num [1:52] 64.8 72.8 77.1 69.6 84.3 83.3 95 87.7 82.1 68.1 ...
 $ Series_Complete_Pop_Pct   : num [1:52] 53 64.9 65.8 56.7 74.4 73.2 82.8 72.9 69.2 57.1 ...
 - attr(*, "spec")=
  .. cols(
  ..   entity = col_character(),
  ..   state_abb = col_character(),
  ..   state_area_sq_miles = col_double(),
  ..   state_division = col_character(),
  ..   state_region = col_character(),
  ..   population = col_double(),
  ..   births_in_2021 = col_double(),
  ..   fertility_rate_per_1000 = col_double()
```

# Data Input: Excel files

- Getting data from Excel is a bit more complicated. You have to download the file, either through R or manually.
- R does not know how to read excel files by default. We will use a package called `readxl` to do that.

## Let's bring the following into R:

https://hutchdatascience.org/SeattleStatSummer_R/data/asthma.xlsx

- > File
- > Import Dataset
- > From Excel …
- > paste the url (https://hutchdatascience.org/SeattleStatSummer_R/data/asthma.xlsx)
- > click "Update" and "Import"

# Looking at the code:

```
library(readxl)
url <- "https://hutchdatascience.org/SeattleStatSummer_R/data/asthma.xlsx"
destfile <- "asthma.xlsx"
curl::curl_download(url, destfile)
asthma <- read_excel(destfile)
View(asthma)
```

# Let’s practice!

# Importing tuberculosis data

- Try downloading the dataset located here: https://github.com/fhdsl/SeattleStatSummer_R/raw/main/data/tb_incidence.xlsx
- Use the File > Import Dataset > from Excel
- Browse for the downloaded file on your machine
- Inspect the code that was run. Copy this code into your R Markdown document for later!

# Looking at the code

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

# Modifying the code

You can name the dataset whatever you want, it's an object in your Environment now.

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

## Summary

- > File
- > Import Dataset
- > From Text (`readr`) **OR** From Excel
- > paste the url or Browse for the file
- > click “Update” and “Import”
- > save the code for later!

Workshop Website