

# Data Input

## Quick reminder - Getting files from downloads

This course will involve moving files around on your computer and downloading files.

If you are new to this - check out the videos on the resource page of the website.

# R Projects

# R Projects

R Projects are a super helpful feature of RStudio. They help you:

- **Stay organized.** R Projects help in organizing your work into self-contained directories (folders), where all related scripts, data, and outputs are stored together. This organization simplifies file management and makes it easier to locate and manage files associated with your analysis or project.
- **Find the right files.** When you open an R Project, RStudio automatically sets the working directory to the project's directory. This is where RStudio “looks” for files. Because it's always the Project folder, it can help avoid common issues with file paths.
- **Be more reproducible.** You can share the entire project directory with others, and they can replicate your environment and analysis without much hassle.

## Why projects?

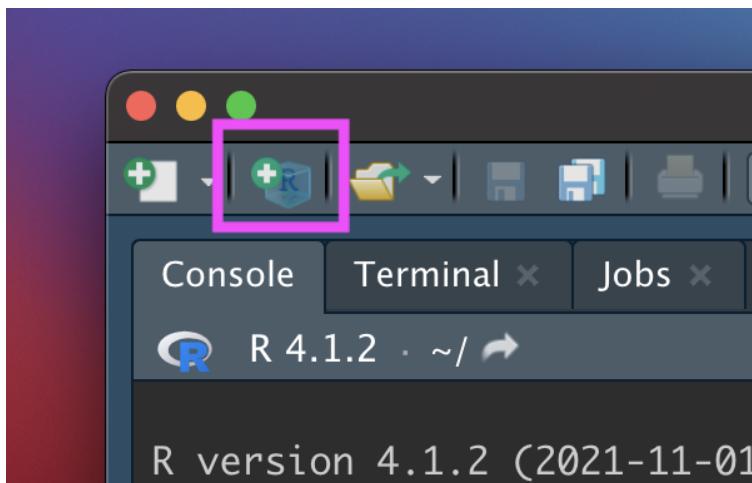
“The chance of the `setwd()` command having the desired effect – making the file paths work – for anyone besides its author is 0%. It’s also unlikely to work for the author one or two years or computers from now. The project is not self-contained and portable.”

- [Jenny Bryan](#)

**Let's go over how to create and use an R Project!**

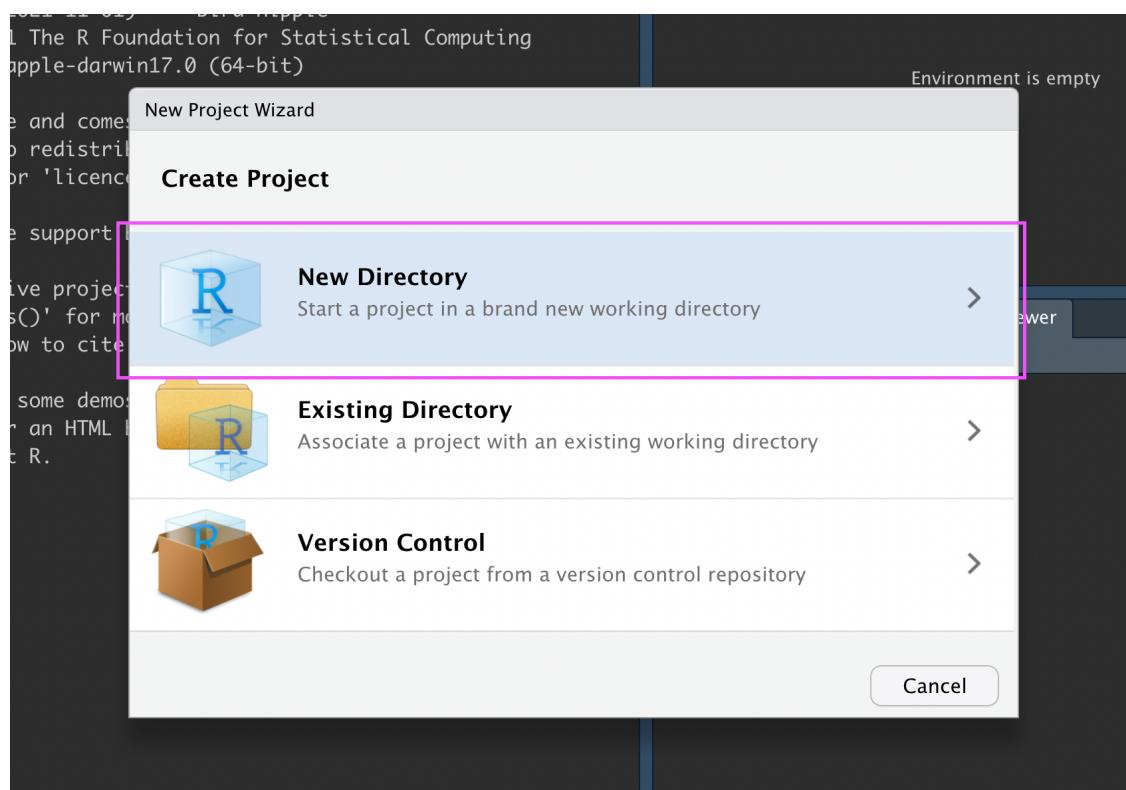
# New R Project

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:



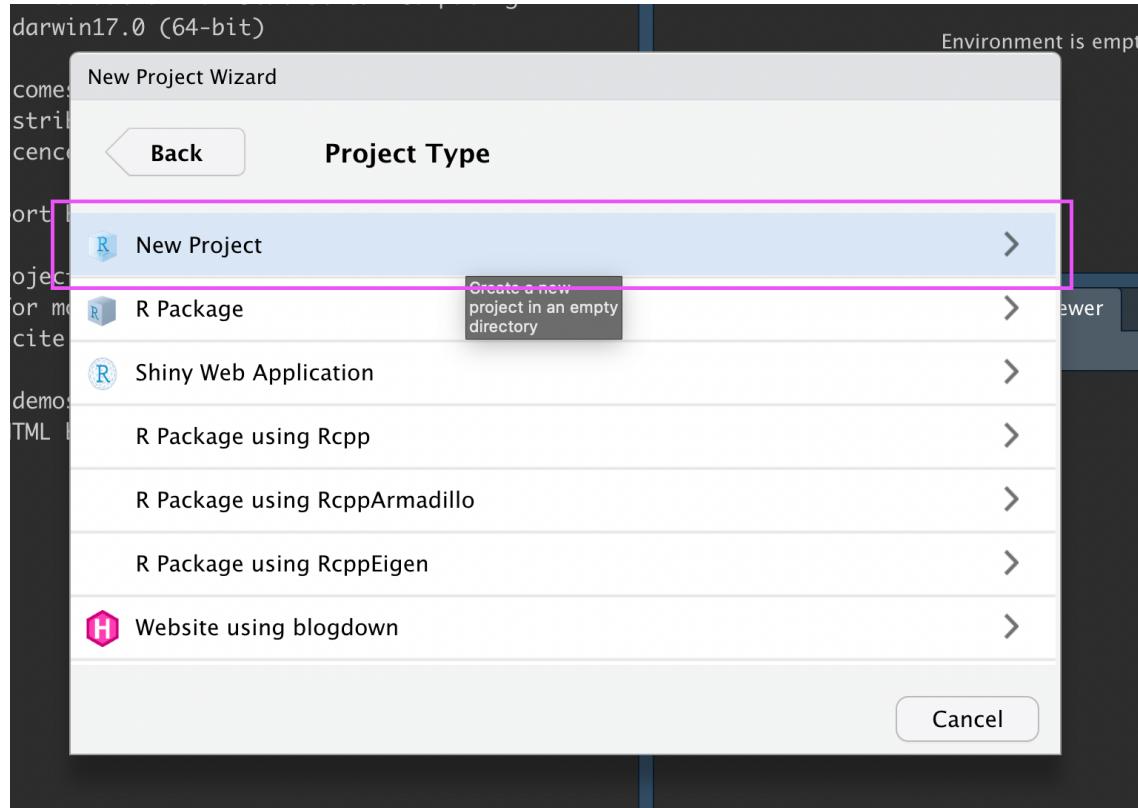
# New R Project

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



# New R Project

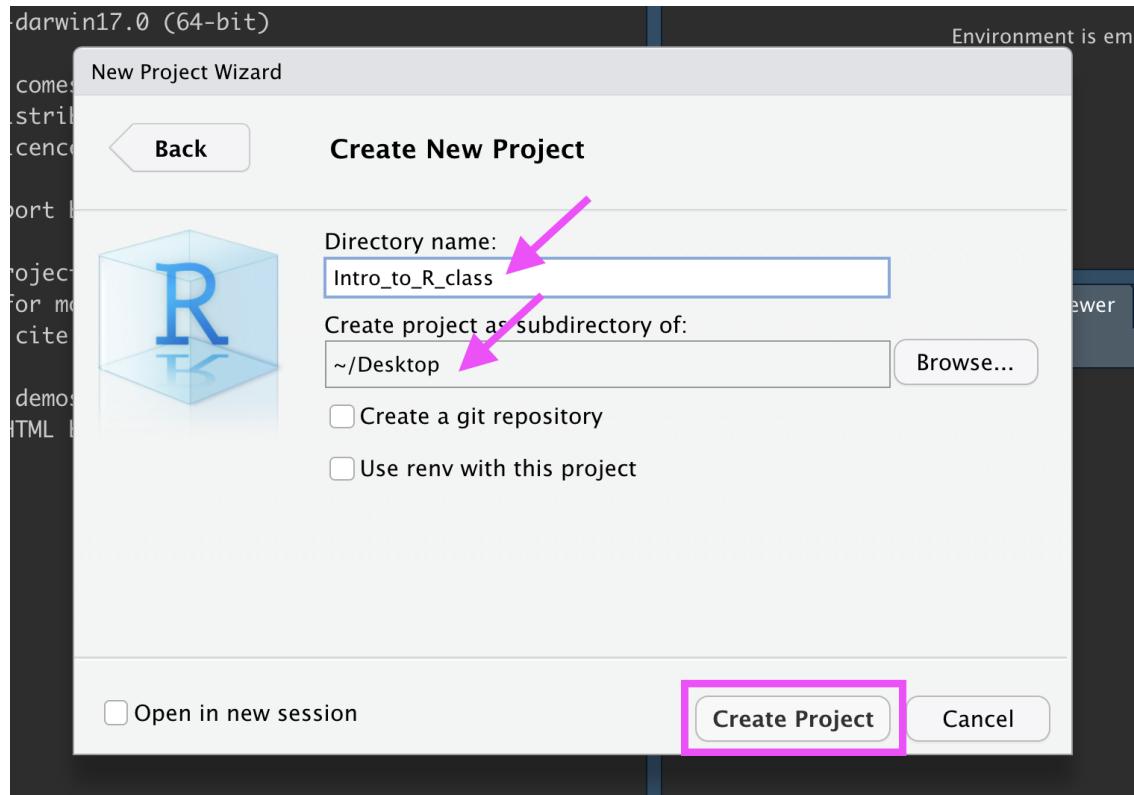
Click “New Project”:



# New R Project

Type in a name for your new folder.

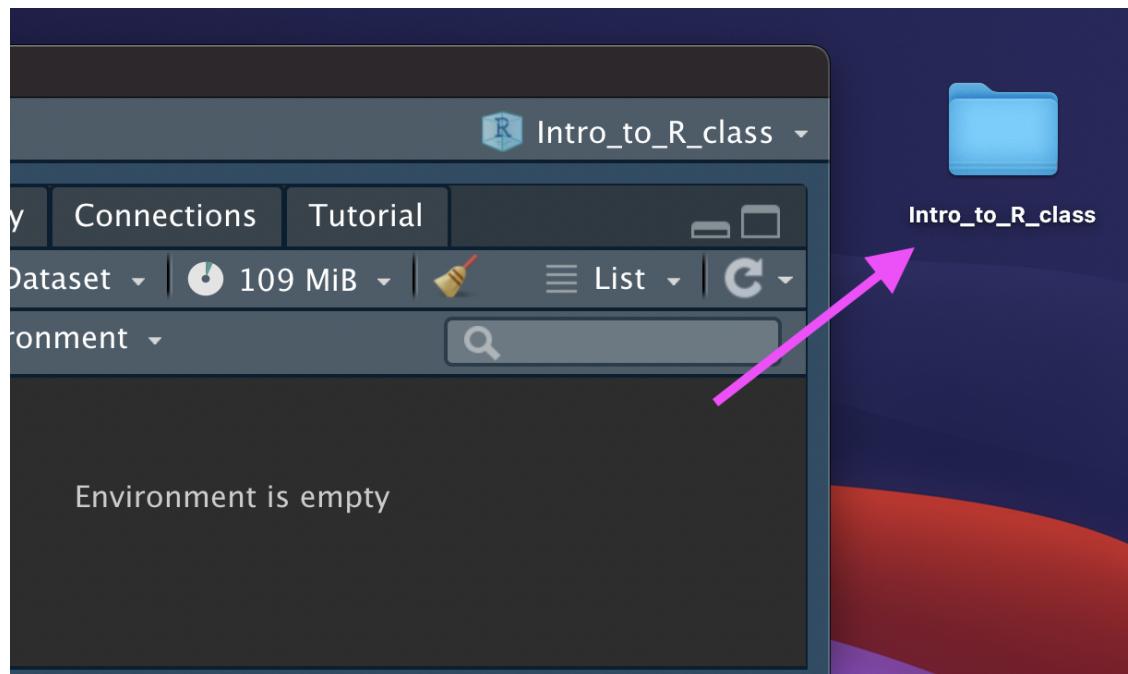
Store it somewhere easy to find, such as your Desktop:



# New R Project

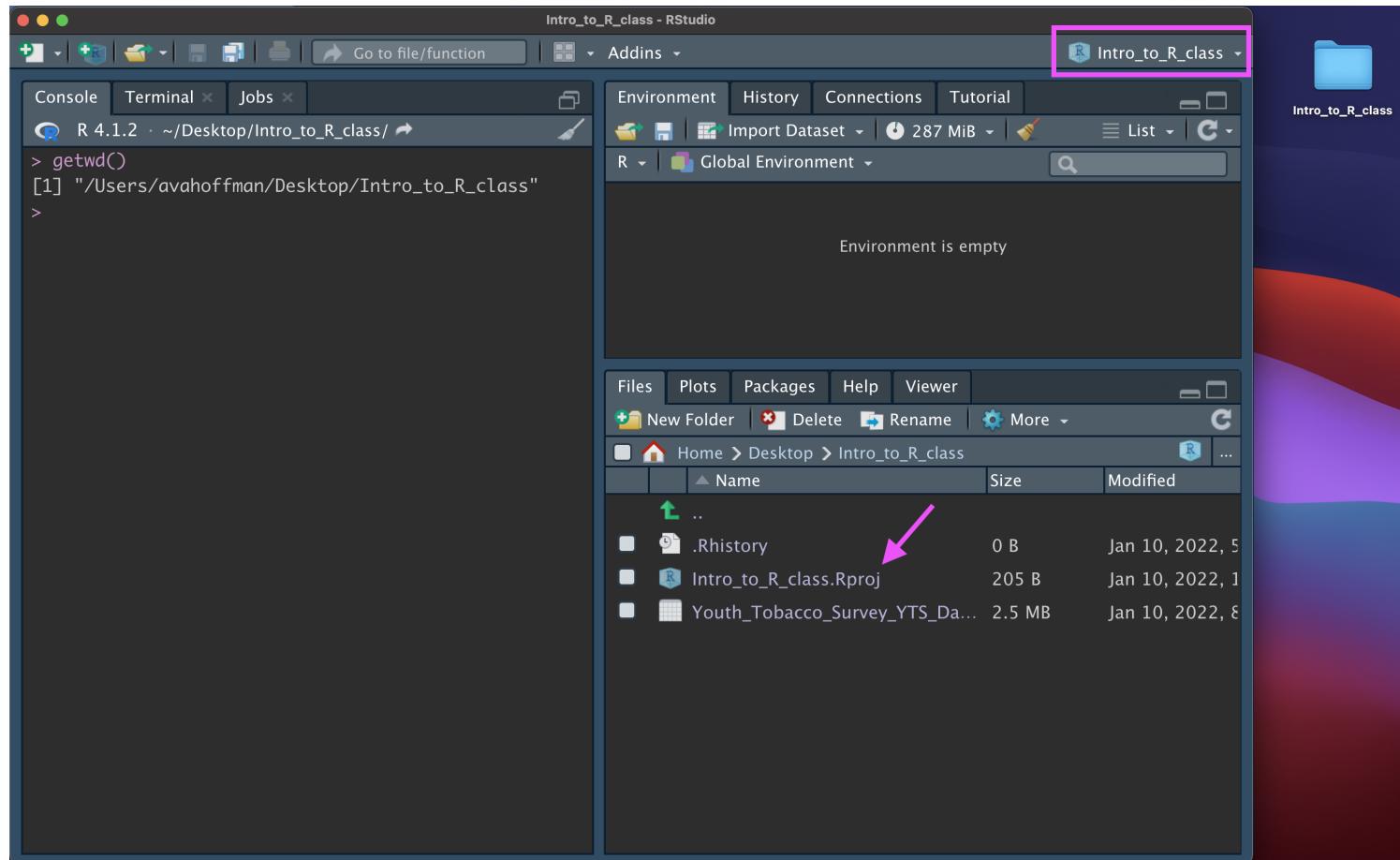
You now have a new R Project folder on your Desktop!

Make sure you add any scripts or data files to this folder as you go through your Intro to R lessons, or work on a new project. This will make sure R is able to "find" your files.



# See and change projects

You can see what project you have open in the top right corner.



# Getting data into R (manual/point and click)

## 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')
  - tab delimited (e.g. '.txt')

**delimiters** are symbols that separate cells in a simple-text file.

## 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>

## Import Dataset (URL)

- > File
- > Import Dataset
- > From Text (readr)
- > paste the url  
[http://jhudatascience.org/intro\\_to\\_r/data/Youth\\_Tobacco\\_Survey\\_YTS\\_Data.csv](http://jhudatascience.org/intro_to_r/data/Youth_Tobacco_Survey_YTS_Data.csv)
- > click “Update” and “Import”

*Saves data in memory, not to hard drive*

# What Just Happened?

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

The screenshot shows the RStudio interface. On the left, a data preview pane is highlighted with a pink border, displaying a table titled "Youth\_Tobacco\_Survey\_YTS\_Data". The table has columns: YEAR, LocationAbbr, LocationDesc, TopicType, TopicDesc, and MeasureDesc. The data shows rows from 1 to 22, all corresponding to the year 2015, location Arizona, topic type Tobacco Use - Survey Data, and measure descriptions like Cessation (Youth), Percent of Current, and Quit Attempt in. The preview pane also indicates there are 9,794 entries and 31 total columns. Below the preview pane is the R console, which shows the R version information and the standard GNU General Public License notice. The right side of the interface includes the Global Environment pane, a file browser, and a file viewer.

YEAR	LocationAbbr	LocationDesc	TopicType	TopicDesc	MeasureDesc
1	2015 AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Percent of Current
2	2015 AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Percent of Current
3	2015 AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Percent of Current
4	2015 AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Quit Attempt in
5	2015 AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Quit Attempt in
6	2015 AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Quit Attempt in
7	2015 AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking Status
8	2015 AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking Status
9	2015 AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking Status
10	2015 AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking Status
11	2015 AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking Status
12	2015 AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking Status
13	2015 AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking Status
14	2015 AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking Status
15	2015 AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking Status
16	2015 AZ	Arizona	Tobacco Use - Survey Data	Smokeless Tobacco Use (Youth)	User Status
17	2015 AZ	Arizona	Tobacco Use - Survey Data	Smokeless Tobacco Use (Youth)	User Status
18	2015 AZ	Arizona	Tobacco Use - Survey Data	Smokeless Tobacco Use (Youth)	User Status
19	2015 AZ	Arizona	Tobacco Use - Survey Data	Smokeless Tobacco Use (Youth)	User Status
20	2015 AZ	Arizona	Tobacco Use - Survey Data	Smokeless Tobacco Use (Youth)	User Status
21	2015 AZ	Arizona	Tobacco Use - Survey Data	Smokeless Tobacco Use (Youth)	User Status
22	2015 AZ	Arizona	Tobacco Use - Survey Data	Smokeless Tobacco Use (Youth)	User Status

Showing 1 to 22 of 9,794 entries, 31 total columns

R 4.2.2 ~/ →

```
R version 4.2.2 (2022-10-31) -- "Innocent and Trusting"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: aarch64-apple-darwin20 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
```

# 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.

The screenshot shows the RStudio interface. On the left, the `Console` pane displays the R startup message:

```
R version 4.2.2 (2022-10-31) -- "Innocent and Trusting"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: aarch64-apple-darwin20 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
```

On the right, the `Environment` pane lists the object `Youth_Tobacco_Survey_YTS_Data` with the following details:

Data  
Youth\_Tobacco\_Survey\_YTS\_Data 9794 obs. of 31 variables

A pink arrow points from the text "The table button opens the data for you to view." to the table icon in the `Environment` pane. A pink circle highlights the table icon in the `Environment` pane.

# What Just Happened?

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

The screenshot shows the RStudio interface with the following components:

- Data View:** Displays a table titled "Youth\_Tobacco\_Survey\_YTS\_Data" with 18 rows and 31 columns. The columns include YEAR, LocationAbbr, LocationDesc, TopicType, TopicDesc, MeasureDesc, and various smoking-related metrics like Cessation (Youth), Percent of Current, Quit Attempt in, Smoking Status, etc.
- Global Environment:** Shows the dataset "Youth\_Tobacco\_Survey\_Y... 9794 obs. of 31 variables" is loaded.
- Console:** The bottom-left pane shows the R session history:

```
R 4.2.2 ~ ↵
> library(readr)
> Youth_Tobacco_Survey_YTS_Data <- read_csv("http://jhubdatascience.org/intro_to_r/data/Youth_Tobacco_Survey_YTS_Data.csv")
Rows: 9794 Columns: 31
--- Column specification ---
Delimiter: ","
chr (24): LocationAbbr, LocationDesc, TopicType, TopicDesc, MeasureDesc, DataSource, Respo...
dbl (7): YEAR, Data_Value, Data_Value_Std_Err, Low_Confidence_Limit, High_Confidence_Limi...
# Use `spec()` to retrieve the full column specification for this data.
# Specify the column types or set `show.col_types = FALSE` to quiet this message.
> View(Youth_Tobacco_Survey_YTS_Data)
> |
```
- File Explorer:** Shows the file structure: Home > Desktop.

# Import Dataset

The screenshot shows the RStudio interface with the following components:

- Top Bar:** Includes the RStudio logo, menu items (File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Window, Help), and system status (Mon Jan 10 9:30 PM).
- Console Tab:** Shows "R 4.1.2 ~/" and a prompt ">".
- Global Environment:** Displays the message "Environment is empty".
- Help Viewer:** Shows the documentation for the `read_delim` function from the `readr` package.

  - Header:** Files, Plots, Packages, Help, Viewer.
  - Search Bar:** Refresh Help Topic, Find in Topic.
  - Text:** R: Read a delimited file (including csv & tsv) into a tibble.
  - Function Name:** `read_delim {readr}`
  - Description:**

`read_csv()` and `read_tsv()` are special cases of the general `read_delim()`. They're useful for reading the most common types of flat file data, comma separated values and tab separated values, respectively. `read_csv2()` uses ; for the field separator and , for the decimal point. This is common in some European countries.

## Import Dataset (file)

- > *Download the data*
- > *Put data in the project folder*
- > File, Import Dataset, From Text (readr)
- > browse for the file
- > click “Update” and “Import”

## GUT CHECK!

How can we get data into R?

- A. From a URL
- B. From a file we downloaded
- C. Both of these!

## Lab - Part 1

- [Class Website](#)
- [Data Input Lab](#)

## Manual Import: Pros and Cons

Pros: easy!!

Cons: obscures some of what's happening, others will have difficulty running your code

# Getting data into R (directly)

# Data Input: Read in Directly

The tidyverse contains a package `readr` that is handy for importing data.

```
library(tidyverse)
dat <- read_csv(
  file = "http://jhudatascience.org/intro_to_r/data/Youth_Tobacco_Survey_YTS_Data.csv"
)

# `head` displays first few rows of a data frame. `tail()` works the same way.
head(dat, n = 5)

# A tibble: 5 × 31
  YEAR LocationAbbr LocationDesc TopicType      TopicDesc MeasureDesc DataSource
  <dbl> <chr>       <chr>        <chr>        <chr>       <chr>       <chr>
1 2015 AZ          Arizona     Tobacco Use ... Cessatio... Percent of... YTS
2 2015 AZ          Arizona     Tobacco Use ... Cessatio... Percent of... YTS
3 2015 AZ          Arizona     Tobacco Use ... Cessatio... Percent of... YTS
4 2015 AZ          Arizona     Tobacco Use ... Cessatio... Quit Attem... YTS
5 2015 AZ          Arizona     Tobacco Use ... Cessatio... Quit Attem... YTS
# ℹ 24 more variables: Response <chr>, Data_Value_Unit <chr>,
#   Data_Value_Type <chr>, Data_Value <dbl>, Data_Value_Footnote_Symbol <chr>,
#   Data_Value_Footnote <chr>, Data_Value_Std_Err <dbl>,
#   Low_Confidence_Limit <dbl>, High_Confidence_Limit <dbl>, Sample_Size <dbl>,
#   Gender <chr>, Race <chr>, Age <chr>, Education <chr>, GeoLocation <chr>,
#   TopicTypeId <chr>, TopicId <chr>, MeasureId <chr>, StratificationID1 <chr>,
#   StratificationID2 <chr>, StratificationID3 <chr>, ...
```

# Data Input: Declaring Arguments

```
dat <- read_csv(  
  file = "http://jhubdatascience.org/intro_to_r/data/Youth_Tobacco_Survey_YTS_Data.csv"  
)  
# EQUIVALENT TO  
dat <- read_csv(  
  "http://jhubdatascience.org/intro_to_r/data/Youth_Tobacco_Survey_YTS_Data.csv"  
)
```

## Data Input: Read in Directly

`read_csv()` needs an argument `file =` in quotation marks.

- can be path to a file on a website (URL)
- can be **path** in your local computer – absolute file path or relative file path

```
# URL  
dat <-  
  read_csv("http://jhubdatascience.org/intro_to_r/data/Youth_Tobacco_Survey_YTS_Data.csv")  
  
# In project folder  
dat <-  
  read_csv("Youth_Tobacco_Survey_YTS_Data.csv")
```

# The working directory

When we work in an R Project, our project folder is our **working directory**.

Working directory is a folder (directory) that RStudio will use to find files.



## Checking the working directory

Run the `getwd()` function to determine your working directory.

```
# Get the working directory  
getwd()
```

## Setting the working directory

You can set the working directory manually with the `setwd()` function. But it's easier to set up a project :)

```
# set the working directory  
setwd("/Users/avahoffman/Desktop")
```

Now what? 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(dat)`

Waiting for R to respond after you accidentally asked it to print a 2million row dataframe:



## Data Input: Other delimiters

`read_tsv()` can read tab delimited (separated) files.

`read_delim()` can be used to specify the delimiter.

- `file` is the path to your file, in quotes
- `delim` is what separates the fields within a record

```
## Examples
dat2 <- read_tsv(file = "table1.tsv", delim = "\t")

dat3 <- read_delim(file = "data.txt", delim = ":")
```

## Data input: other file types

- `readxl` package can read excel files
- `haven` package has functions to read SAS, SPSS, Stata formats
- There are also resources for REDCap : [REDCapR](#)

## **WARNING! `read.csv` is \*base R\***

There are also data importing functions provided in base R (rather than the `readr` package), like `read.delim()` and `read.csv()`.

These functions have slightly different syntax for reading in data (e.g. `header` argument).

However, while many online resources use the base R tools, the latest version of RStudio switched to use these new `readr` data import tools, so we will use them in the class for slides. They are also up to two times faster for reading in large datasets, and have a progress bar which is nice.

## Other Useful Functions

- The `str()` function can tell you about data/objects.
- We will also discuss the `glimpse()` function later, which does something very similar.
- `head()` shows first few rows
- `tail()` shows the last few rows

## Summary

**R Projects** can make it easier to find files.

Importing data manually:

- File > Import Dataset > From Text (`readr`)
- Paste the url / browse
- Click “Update” and “Import”
- Review the process: <https://youtu.be/LEkNfJgpunQ>

Importing data programmatically:

- `read_csv()` function from tidyverse (`readr`) package
- Use `getwd()` to check your working directory, where R looks for your data files

## Summary - Part 2

Look at your data!

- Check the environment for a data object
- `View()` gives you a preview of the data in a new tab

Other file types

- `readr` package: `read_delim()` for general delimited files
- other packages for more complicated files.

Don't forget to use `<-` to assign your data to an object!

## Lab - Part 2

- [Class Website](#)
- [Data Input Lab](#)
- [Posit's Data Import Cheatsheet](#)
- [Day 2 Cheatsheet](#)



Image by [Gerd Altmann from Pixabay](#)