

Data Input

Outline

- Part 0: A little bit of set up!
- Part 1: reading in manually (point and click)
- Part 2: reading in directly & working directories
- Part 3: checking data & multiple file formats

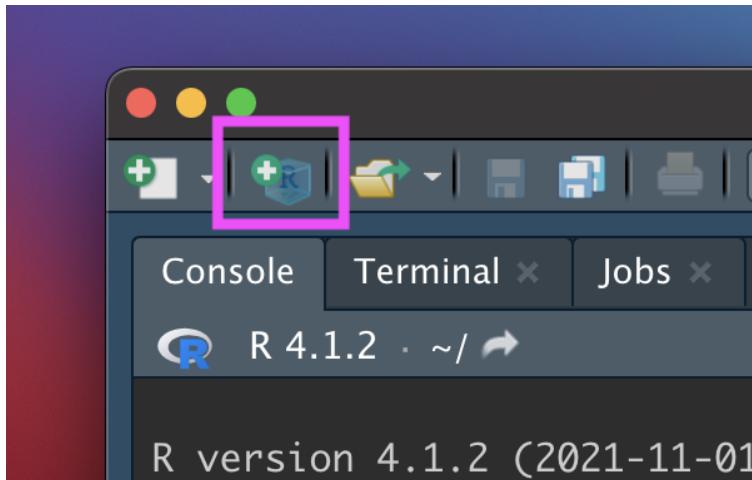
We will cover Output a bit later!

Part 0: Setup - R Project

New R Project

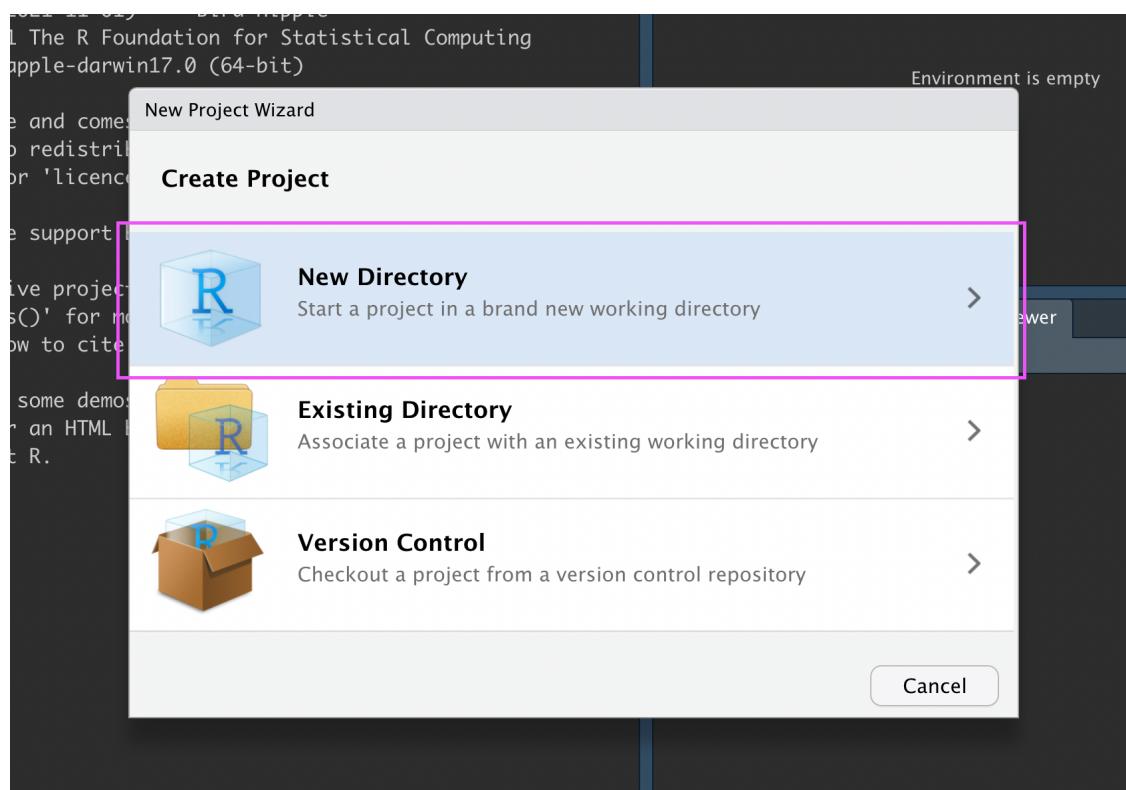
Let's make an R Project so we can stay organized in the next steps.

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



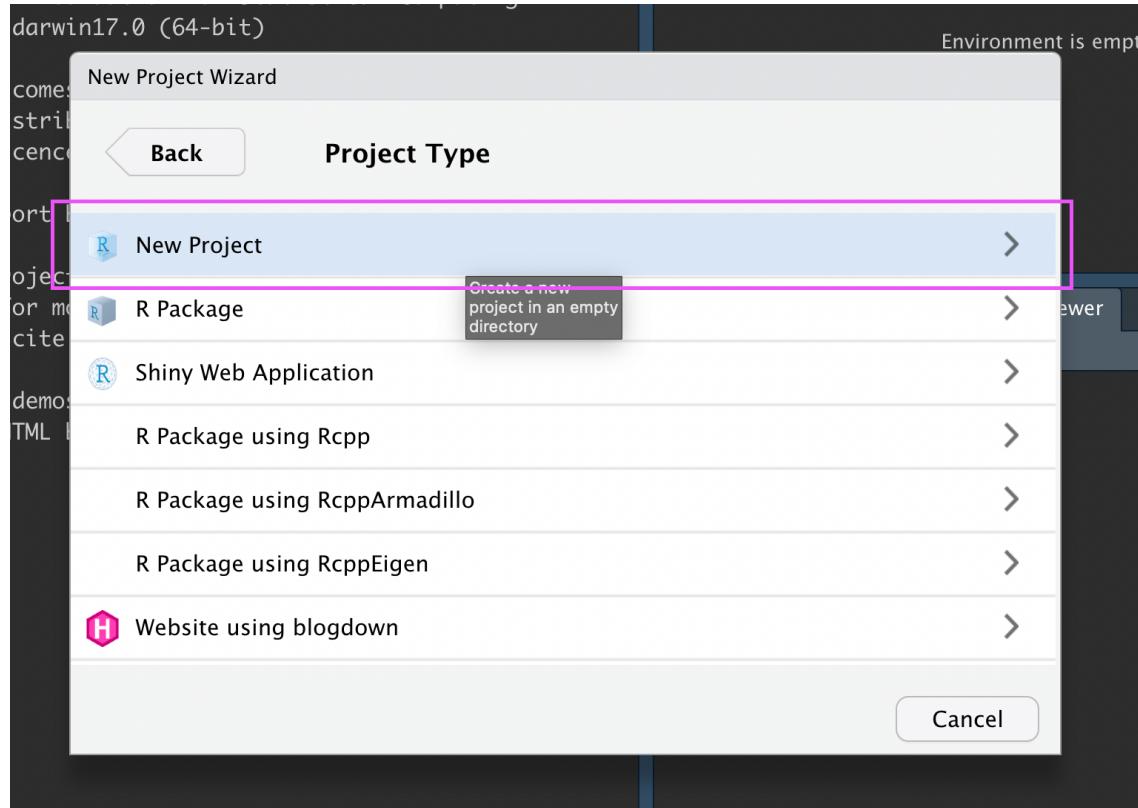
New R Project

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



New R Project

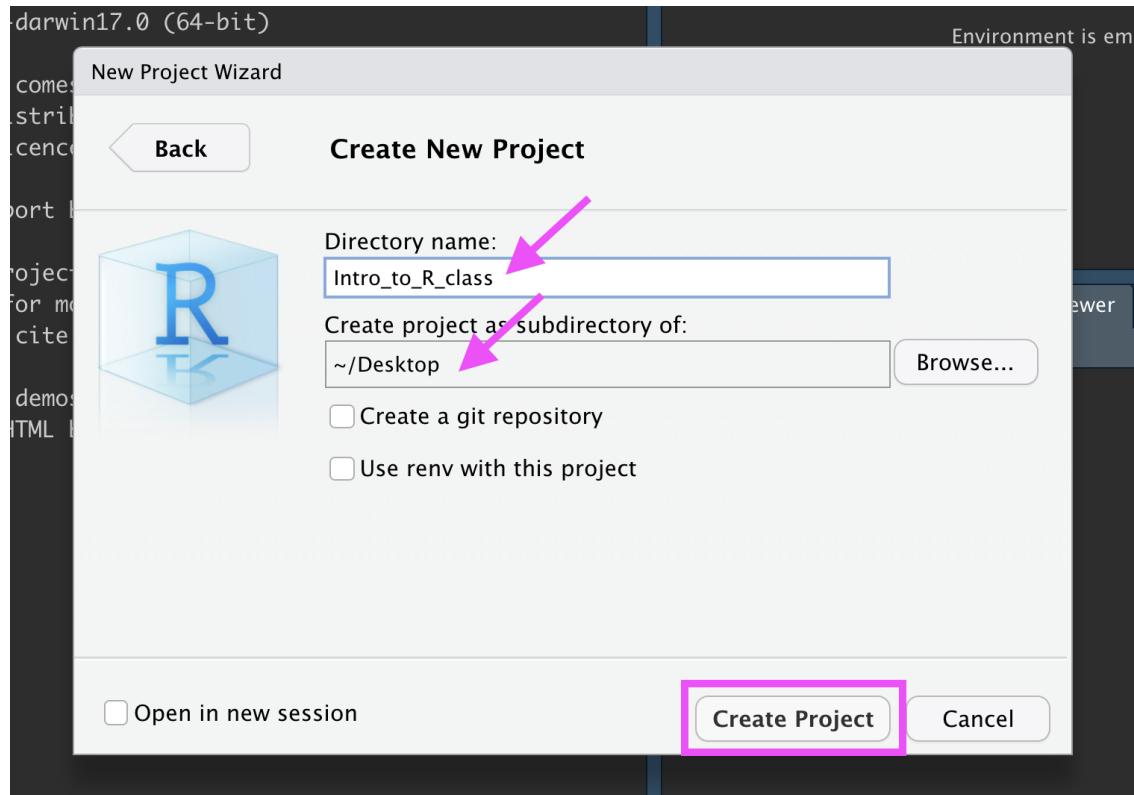
Click “New Project”:



New R Project

Type in a name for your new folder.

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

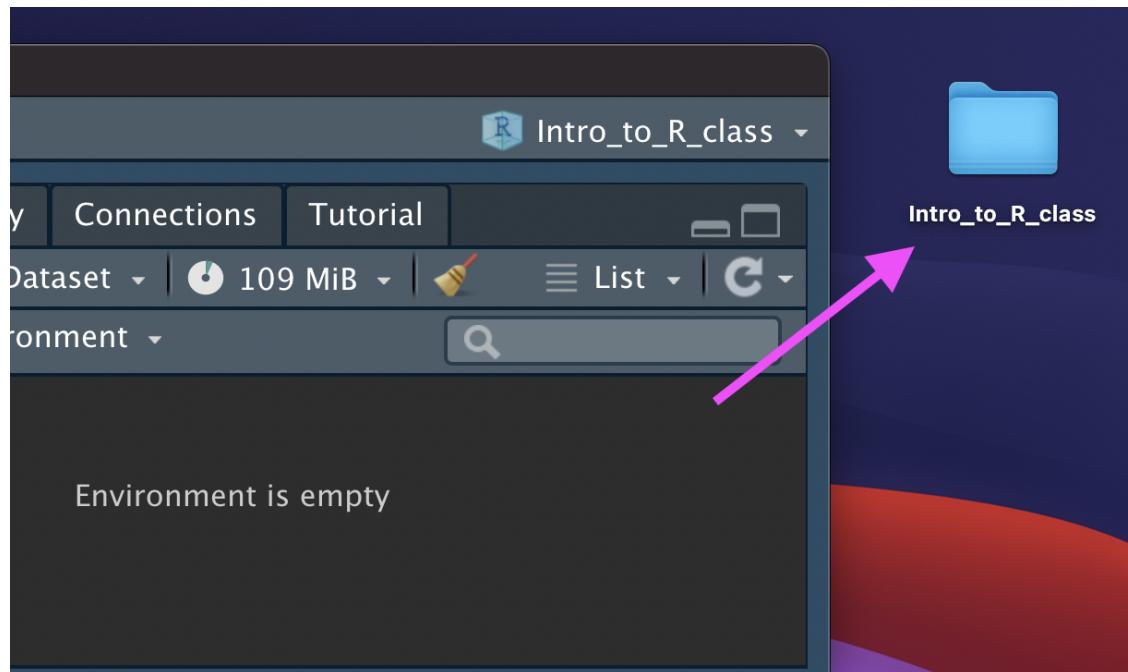


New R Project

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.

We will review this in lab.



Part 1: 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')
 - Microsoft Excel (e.g. '.xlsx')

Note: data for demonstration

- We have added functionality to load some datasets directly in the `dasehr` package

Data Input

CalEnviroScreen Dataset:

CalEnviroScreen is a project that ranks census tracts in California based on potential exposures to pollutants, adverse environmental conditions, socioeconomic factors and the prevalence of certain health conditions. Data used in the CalEnviroScreen model come from national and state sources.

- Check out the data at: <https://calenviroscreen-oehha.hub.arcgis.com/#Data>

Import Dataset

- > File
- > Import Dataset
- > From Text (`readr`)
- > paste the url (https://daseh.org/data/CalEnviroScreen_data.csv)
- > click “Update” and “Import”

What Just Happened?

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

The screenshot shows the RStudio interface. The top-left pane displays a data preview for the dataset "Youth_Tobacco_Survey_YTS_Data". The preview table includes columns: YEAR, LocationAbbr, LocationDesc, TopicType, TopicDesc, and MeasureDesc. The data shows 22 rows of survey results for Arizona in 2015, categorized by topic like Tobacco Use - Survey Data and measures like Cessation (Youth) or Percent of Current. The bottom-left pane is the R Console, showing the R version information and the start of the R environment setup. The top-right pane shows the Global Environment and the Data browser, which lists the dataset "Youth_Tobacco_Survey_Y... 9794 obs. of 31 variables".

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 `ca1EnviroScreen_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 `ca1EnviroScreen_data` object is displayed as a table titled `Youth_Tobacco_Survey_YTS_Data`. The table has columns: YEAR, LocationAbbr, LocationDesc, TopicType, TopicDesc, and MeasureDesc. The data shows 22 rows of survey results from 2015, primarily from Arizona, regarding tobacco use and cessation. On the right, the Global Environment pane shows the `Youth_Tobacco_Survey_YTS_Data` object with the description "9794 obs. of 31 variables". Below the environment pane is the file browser, which shows a folder named "Desktop" containing files like "New Blank File.RData" and "New Folder". At the bottom, the R console shows the R version information and the `license()` command being typed.

```
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?

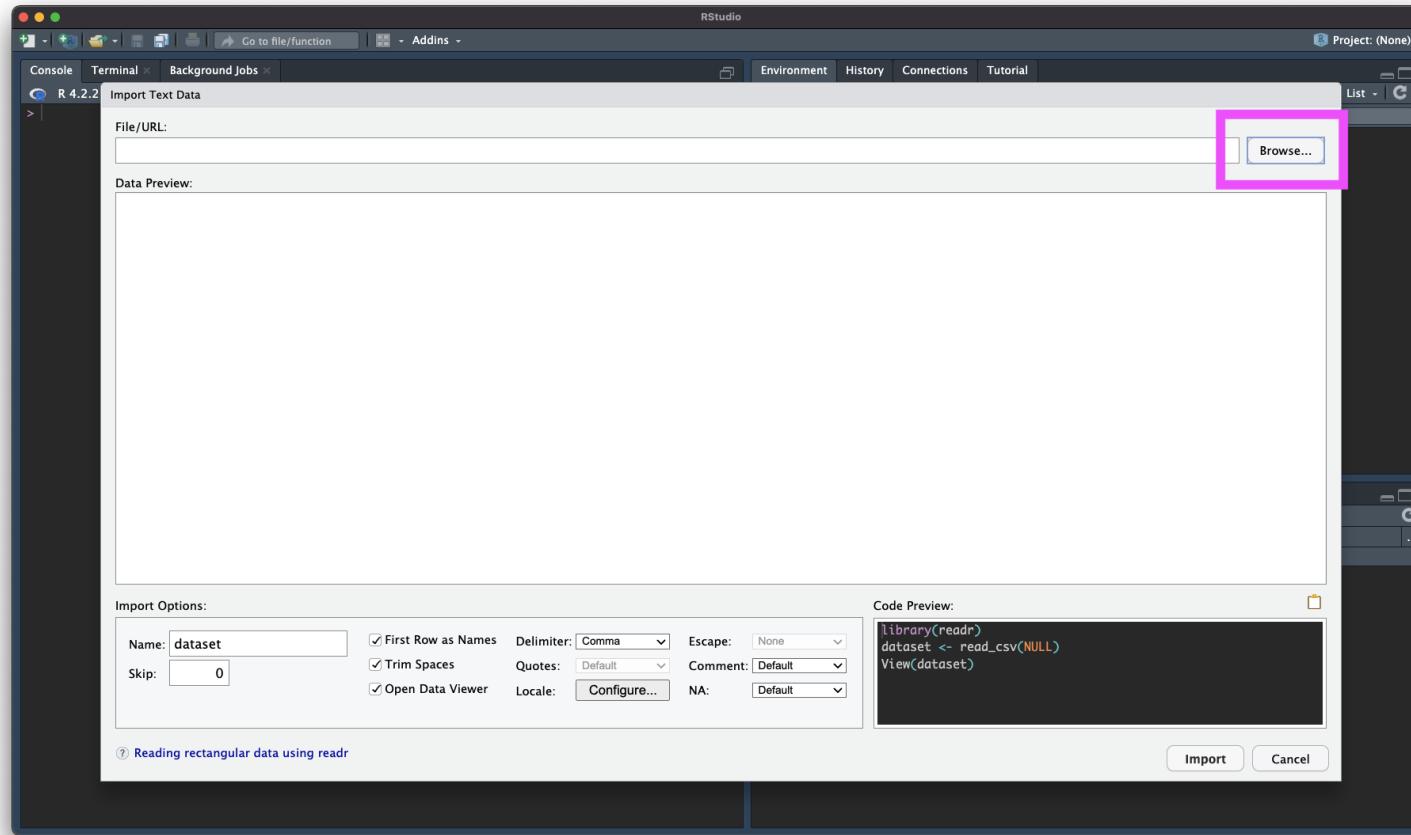
R ran some code in the console (bottom left).

The screenshot shows the RStudio interface with the following components:

- Data View:** Displays the "Youth_Tobacco_Survey_YTS_Data" dataset as a table 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.
- Environment View:** Shows the global environment with one object: "Youth_Tobacco_Survey_YTS_Data" of type "data.frame" with 9794 observations and 31 variables.
- Console View:** Contains the R session history with the following code:

```
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 View:** Shows the current working directory as "Desktop".

Browsing for Data on Your Machine



Import Dataset

The screenshot shows the RStudio interface with the following details:

- Top Bar:** RStudio, File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Window, Help, Gif, Project: (None), Mon Jan 10 9:30 PM.
- Console Tab:** Shows "R 4.1.2 ~/" and a prompt ">".
- Environment Tab:** Shows "Import Dataset" selected, 549 MiB, and "Global Environment". It displays the message "Environment is empty".
- Help Tab:** Shows the documentation for `read_delim` from the `readr` package. The title is "Read a delimited file (including csv & tsv) into a tibble".
 - 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.

Manual Import: Pros and Cons

Pros: easy!!

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

Summary & Lab Part 1

R Projects will make it easier to find files later.

Importing data:

- File > Import Dataset > From Text (readr)
- Paste the url (https://daseh.org/data/CalEnviroScreen_data.csv)
- Click “Update” and “Import”

Review the process: <https://youtu.be/LEkNfJgpunQ>

□ [Class Website](#)

□ [Data Input Lab](#)

Part 2: Getting data into R (directly)

Data Input: Read in Directly

```
# load library `readr` that contains function `read_csv`  
library(readr)  
dat <- read_csv(  
  file = "https://daseh.org/data/CalEnviroScreen_data.csv"  
)  
  
# `head` displays first few rows of a data frame. `tail()` works the same way.  
head(dat, n = 5)  
  
# A tibble: 0 × 1  
#   1 variable: https://daseh.org/data/CalEnviroScreen_data.csv <chr>
```

Data Input: Declaring Arguments

```
dat <- read_csv(  
  file = "https://daseh.org/data/CalEnviroScreen_data.csv"  
)  
# EQUIVALENT TO  
dat <- read_csv(  
  "https://daseh.org/data/CalEnviroScreen_data.csv"  
)
```

Data Input: Read in Directly

`read_csv()` needs an argument `file =`.

- `file` is the path to your 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

Examples

```
dat <- read_csv(file = "www.someurl.com/table1.csv")
```

```
dat <- read_csv(file = "/Users/avahoffman/Downloads/CalEnviroScreen_data.csv")
```

```
dat <- read_csv(file = "CalEnviroScreen_data.csv")
```

Data Input: File paths

Reading from your computer.. What is my “path”?

PC: *autosaves file*

Me: Cool, so where did the
file save?

PC:

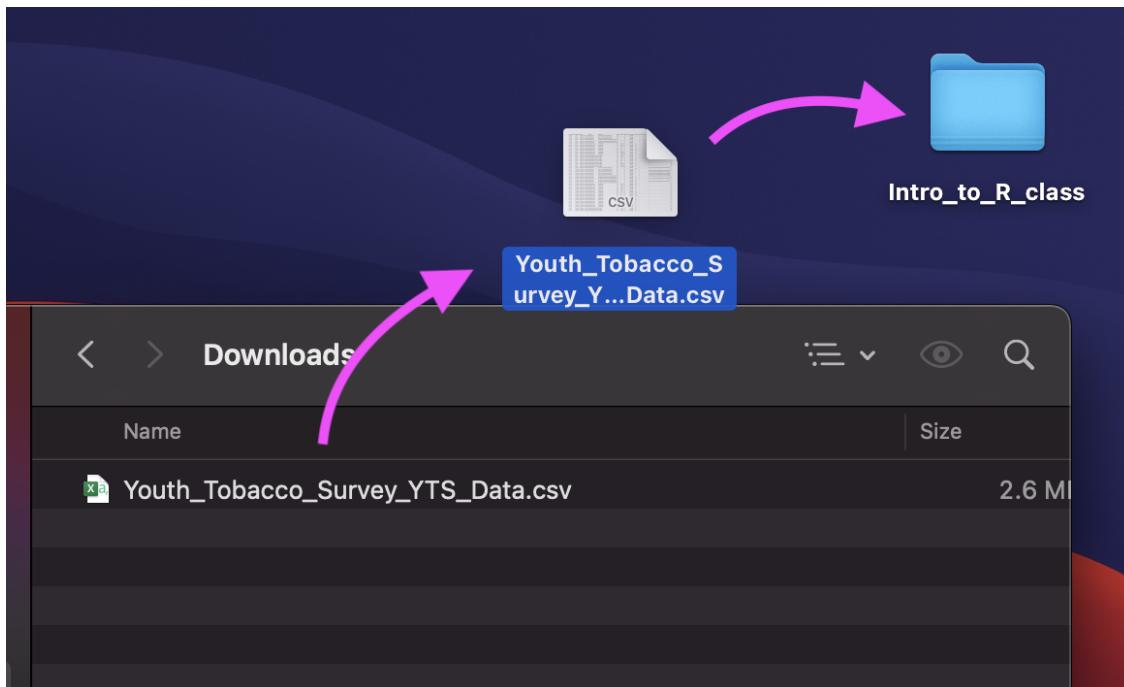


Data Input: File paths

When you set up an R Project, R looks for files in that folder.

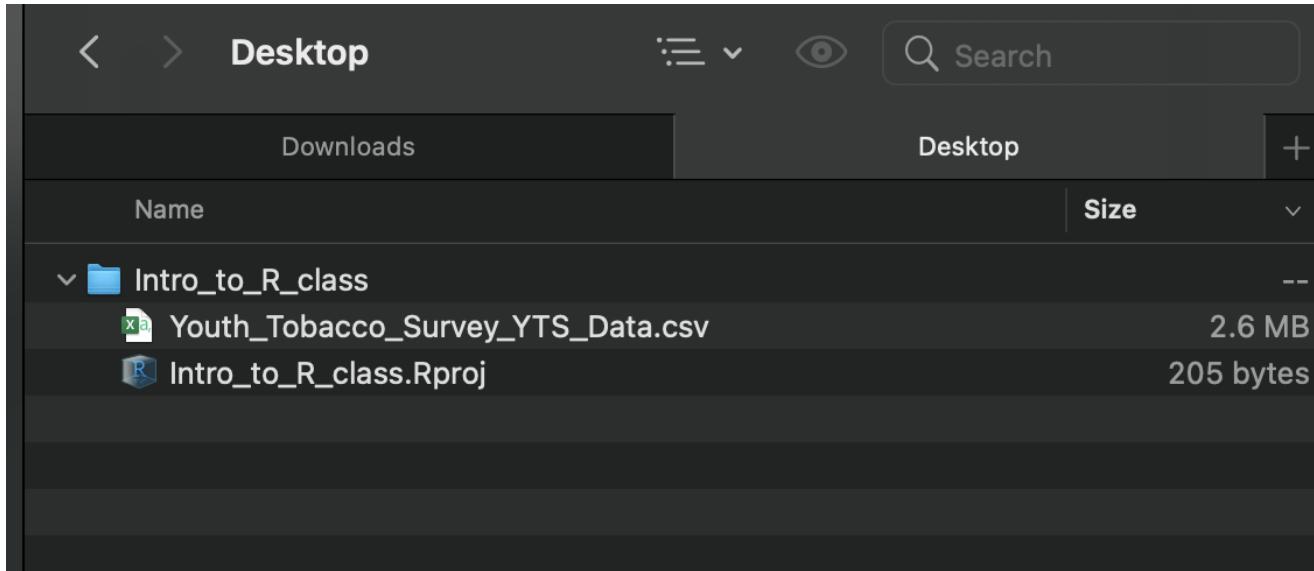
Luckily, we already set up an R Project!

Move downloaded files into the R Project folder.



Data Input: File paths

Confirm the data is in the R Project folder.



Data Input: File paths

If we add the CalEnviroScreen_data.csv file to the R Project folder, we can use the file name for the `file` argument:

```
dat <- read_csv(file = "CalEnviroScreen_data.csv")
```

Why does this work?

When we create an R Project, we establish the **working directory**.

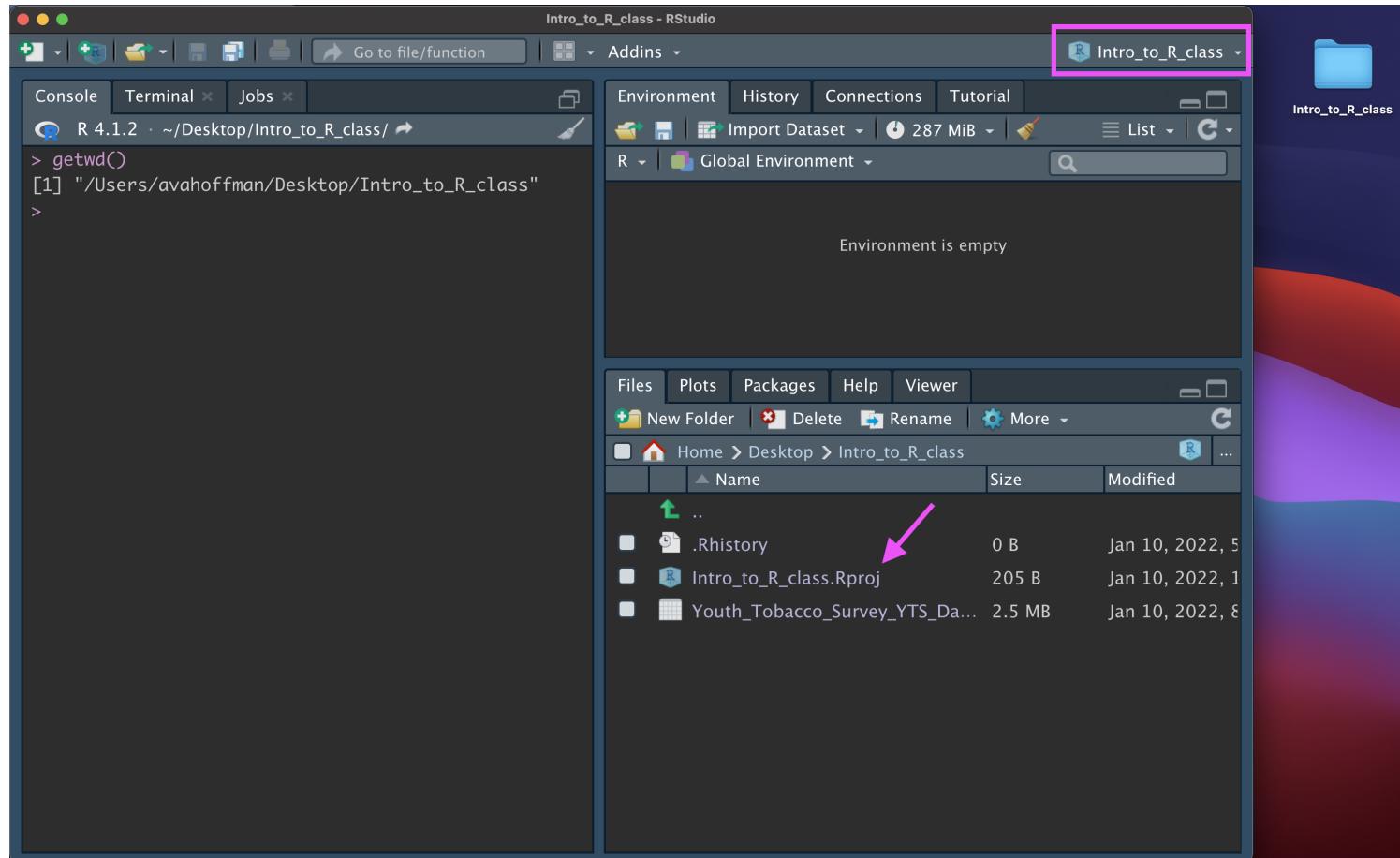
Working directory is a folder (directory) that RStudio assumes “you are working in”.

It's where R looks for files.



The Working Directory

The working directory is wherever the .Rproj file is.



Data Input: Getting Organized!

If you move a file into a nested folder, **you must update the path!**

```
# Notice "data/" has been added!
dat <- read_csv(file = "data/CalEnviroScreen_data.csv")
```

Always confirm you read in the data by checking the “Environment” pane (top right).

Part 3: 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)`

The screenshot shows the RStudio interface with the following components:

- Top Bar:** Intro_to_R_class - RStudio
- Left Sidebar:** Addins
- Environment Tab:** Shows 1.17 GiB of memory used.
- Data Tab:** Shows the dataset `dat` with 9794 observations and 31 variables.
- Files Tab:** Shows the file structure: Home > Desktop > Intro_to_R_class > data.
- Console Tab:** Displays the command `> View(dat)`.
- Plots Tab:** Not visible in the screenshot.
- Packages Tab:** Not visible in the screenshot.
- Help Tab:** Not visible in the screenshot.
- Viewer Tab:** Not visible in the screenshot.
- Present Tab:** Not visible in the screenshot.

The main workspace displays a data preview for the first 12 rows of the `dat` dataset:

	YEAR	LocationAbbr	LocationDesc	TopicType	TopicDesc	Measure
1	2015	AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Percent c
2	2015	AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Percent c
3	2015	AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Percent c
4	2015	AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Quit Atte
5	2015	AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Quit Atte
6	2015	AZ	Arizona	Tobacco Use - Survey Data	Cessation (Youth)	Quit Atte
7	2015	AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking
8	2015	AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking
9	2015	AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking
10	2015	AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking
11	2015	AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking
12	2015	AZ	Arizona	Tobacco Use - Survey Data	Cigarette Use (Youth)	Smoking

Below the preview, a message states: "Showing 1 to 11 of 9,794 entries, 31 total columns".

Data Input: Other delimiters with `read_delim()`

`read_csv()` is a special case of `read_delim()` – a general function to read a delimited file into a data frame

`read_delim()` needs path to your file and **file's delimiter**, will return a tibble

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

Examples

```
dat <- read_delim(file = "www.someurl.com/table1.tsv", delim = "\t")  
  
dat <- read_delim(file = "data.txt", delim = "|")
```

Data Input: Excel files

- You **cannot** read in an excel file from a URL.
- Need to load the `readxl` package with `library()`.
- The argument is `path` (not `file`).

```
# Programmatically download
download.file(
  url = "https://daseh.org/data/asthma.xlsx",
  destfile = "asthma.xlsx",
  overwrite = TRUE,
  mode = "wb"
)
```

Data Input: Excel files

- You **cannot** read in an excel file from a URL.
- Need to load the `readxl` package with `library()`.
- The argument is `path` (not `file`).

```
library(readxl)  
  
read_excel(path = "asthma.xlsx")
```

Data input: other file types

- `haven` package has functions to read SAS, SPSS, Stata formats

```
library(haven)

# SAS
read_sas(file = "mtcars.sas7bdat")

# SPSS
read_sav(file = "mtcars.sav")

# Stata
read_dta(file = "mtcars.dta")
```

- There are also resources for REDCap : [REDCapR](#)

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.

TROUBLESHOOTING: Common new user mistakes we have seen

1. Working directory problems: trying to read files that R “can’t find”

- Path misspecification
- more on this shortly!

2. Typos (R is **case sensitive**, x and X are different)

- RStudio helps with “tab completion”

3. Open ended quotes, parentheses, and brackets

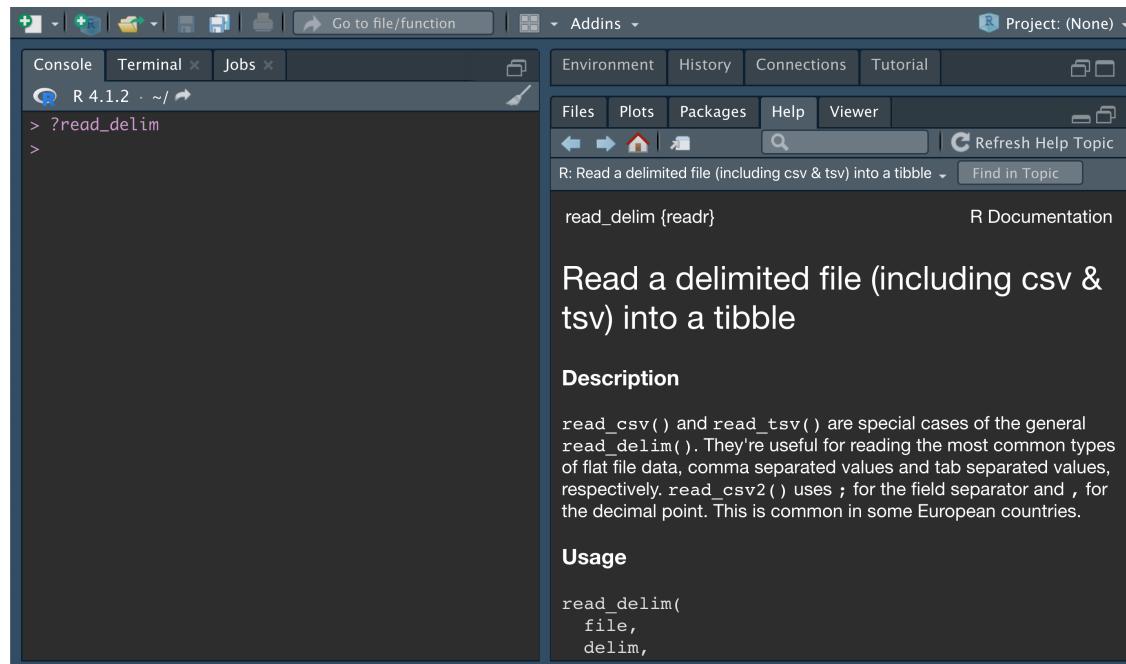
4. Different versions of software

5. Deleting part of the code chunk

TROUBLESHOOTING: Help

For any function, you can write `?FUNCTION_NAME`, or `help("FUNCTION_NAME")` to look at the help file:

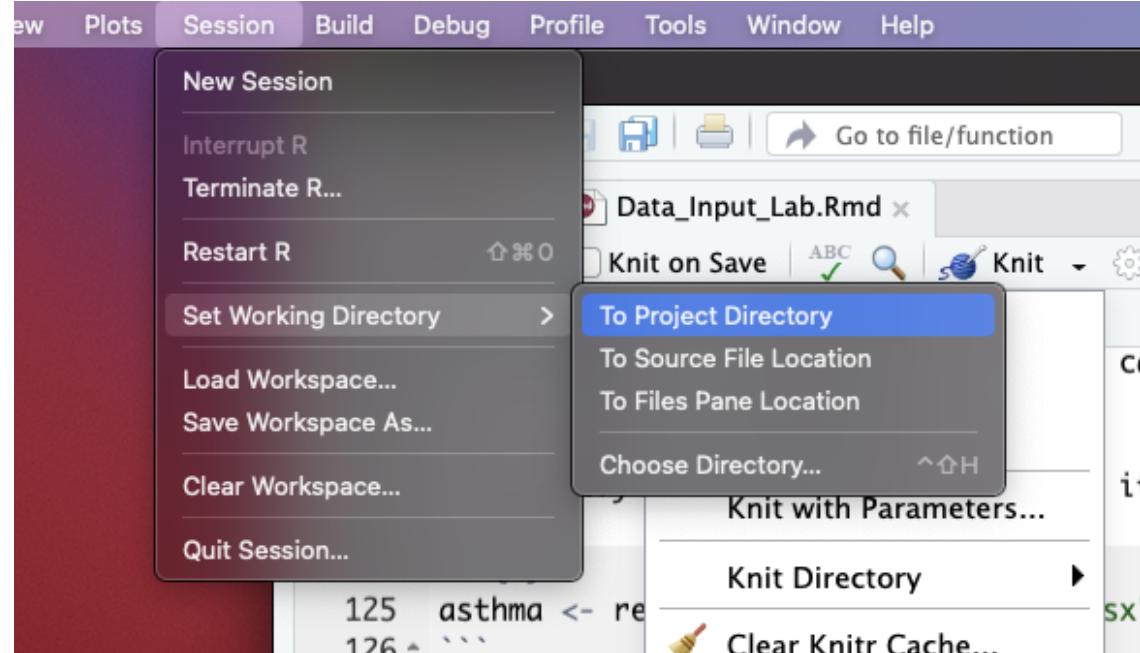
```
?read_delim  
help("read_delim")
```



TROUBLESHOOTING: Setting the working directory

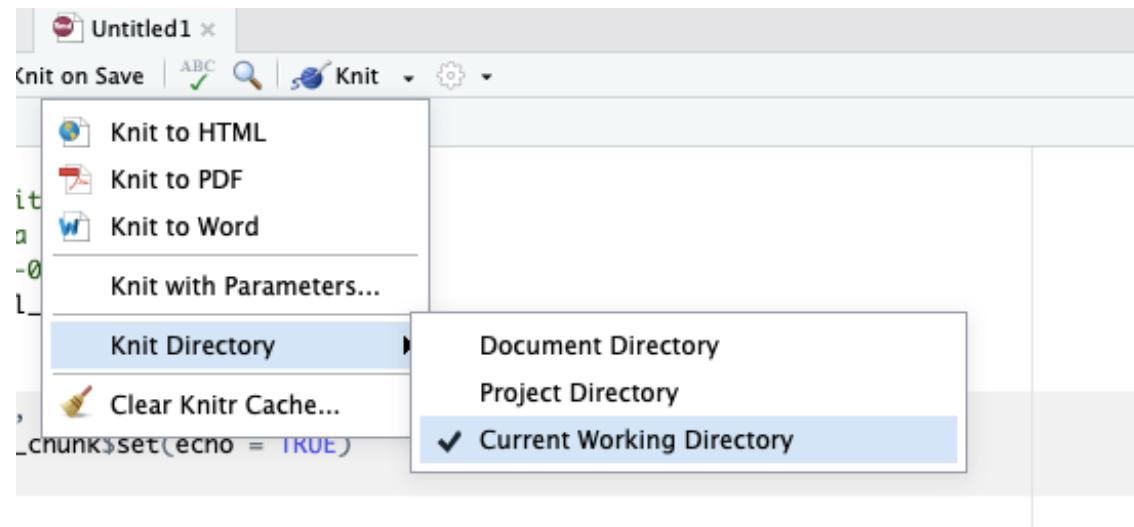
If your R project directory and working directory do not match:

- Session > Set Working Directory > To Project Directory



TROUBLESHOOTING: Setting the working directory

If you are trying to knit your work, it might help to set the knit directory to the “Current Working Directory”:



TROUBLESHOOTING: Setting the working directory

You can also run the `getwd()` function to determine your working directory.

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

You can also set the working directory manually with the `setwd()` function:

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

Other Useful Functions

- The `str()` function can tell you about data/objects (different variables and their classes - more on this later).
- We will also discuss the `glimpse()` function later, which does something very similar.
- `head()` shows first few rows
- `tail()` shows the last few rows
- `here` package

```
library(here)  
here()
```

Summary - Part 2

`read_csv()` function from `readr` package:

- comma delimited data
- needs a file path to be provided
- returns a tibble (data frame)

R Projects are a good way to keep your files organized and reduce headaches

- 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
- `readxl` package: `read_excel()` for Excel files

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

Lab Part 2

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



Image by [Gerd Altmann from Pixabay](#)

W