R Noobz: Class 1
Intro to R

YouthFirst Lab

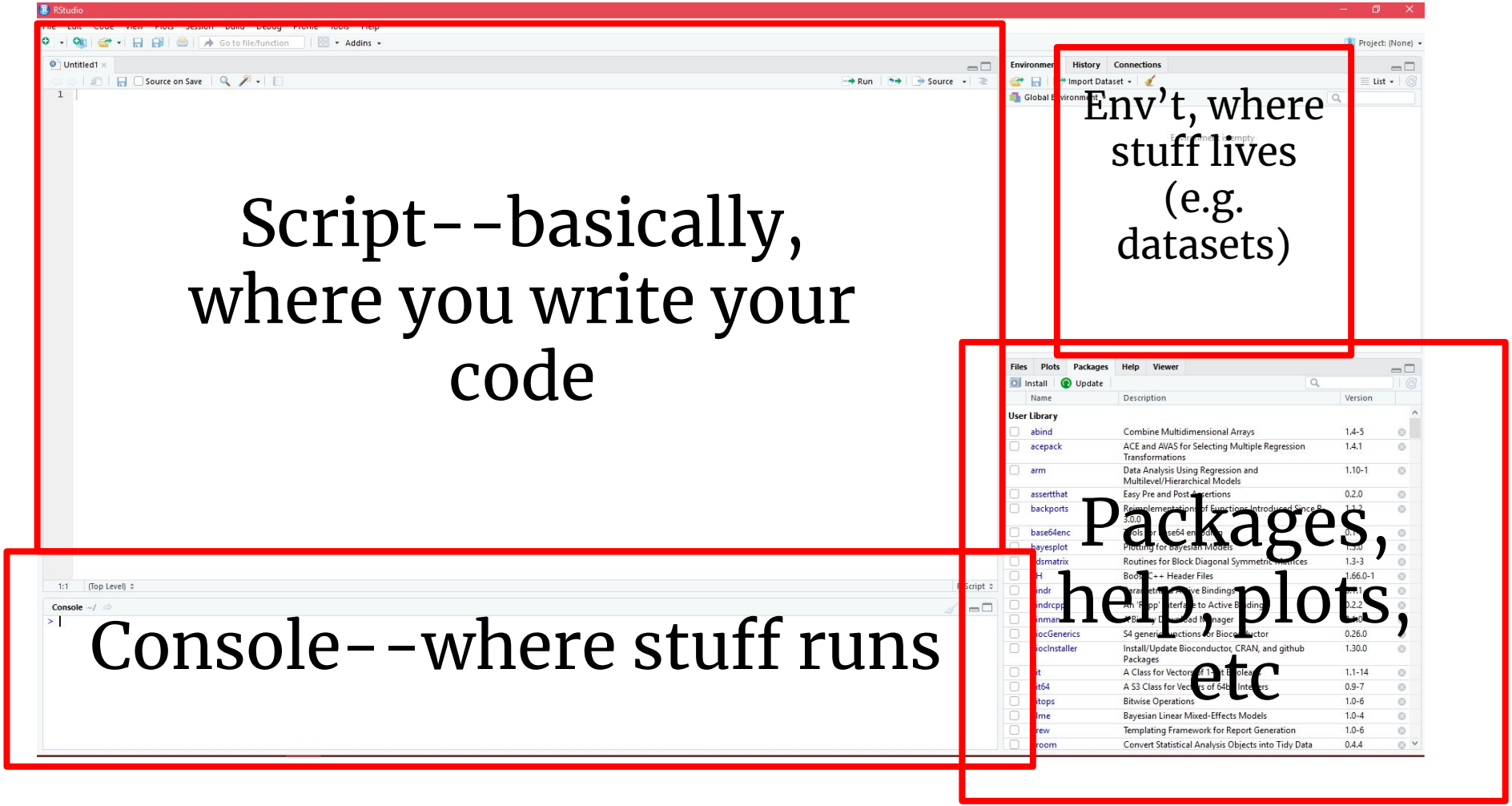
Class Goals

1. Install R on all our computers
2. Open R and not have a panic attack
3. Do some small/basic stuff in R
4. Load a data set into R

Ambitious Goal:

1. Look at data types in R

What is R?


RStudio
Untitled1
Source on Save
Run
Source
Script--basically, where you write your code
Console
Console--where stuff runs
Environment
History
Connections
Import Dataset
Global Environment
Env't, where stuff lives (e.g. datasets)
Files
Plots
Packages
Help
Viewer
Install
Update
Name
Description
Version
User Library
abind
Combine Multidimensional Arrays
1.4-5
acepack
ACE and AVAS for Selecting Multiple Regression Transformations
1.4.1
arm
Data Analysis Using Regression and Multilevel/Hierarchical Models
1.10-1
Packages, help, plots, etc

R: Why is it cool?

- It is free to everyone, and anyone can create programs for it, so cool stuff exists like:
  - *Recoder*
    - *A package with a sole purpose of recoding variables*
  - *GGplot*
    - *Literally, the most amazing graphs*
  - *Psych*
    - *Stats stuff for psychological, psychometric, and personality research*
  - *PRoc*
    - *Literally, a whole package just for ROC curves (Jason!)*

R: Why is it cool?

- You can do a lot of stats, like:

ANALYTICS

- Basic Mathematics
- Basic Statistics
- Probability Distributions
- Big Data Analytics *
- Machine Learning
- Optimization and Mathematical Programming
- Signal Processing
- Simulation and Random Number Generation
- Statistical Modeling
- Statistical Tests

GRAPHICS AND VISUALIZATION

- Static Graphics
- Dynamic Graphics
- Devices and Formats

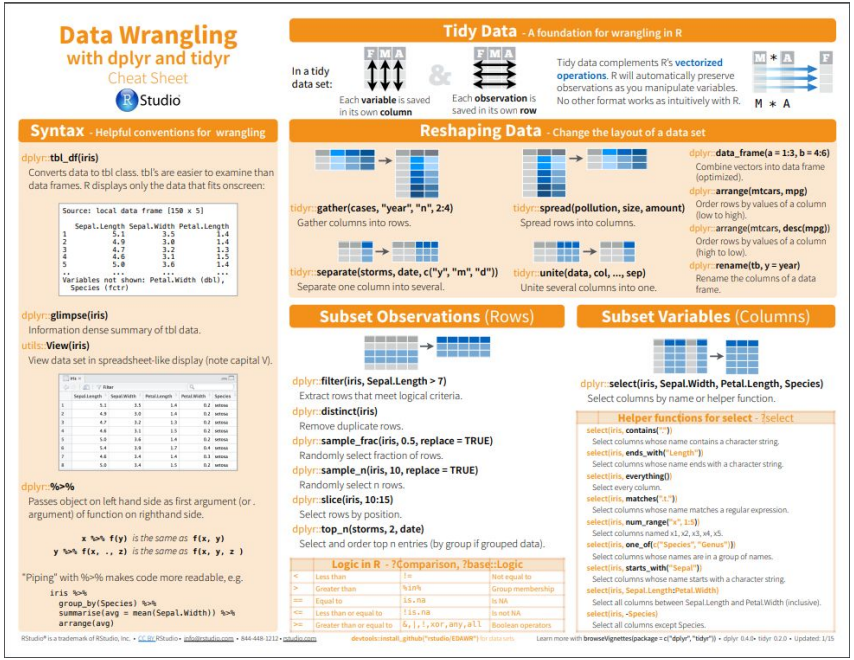
R APPLICATIONS and EXTENSIONS***

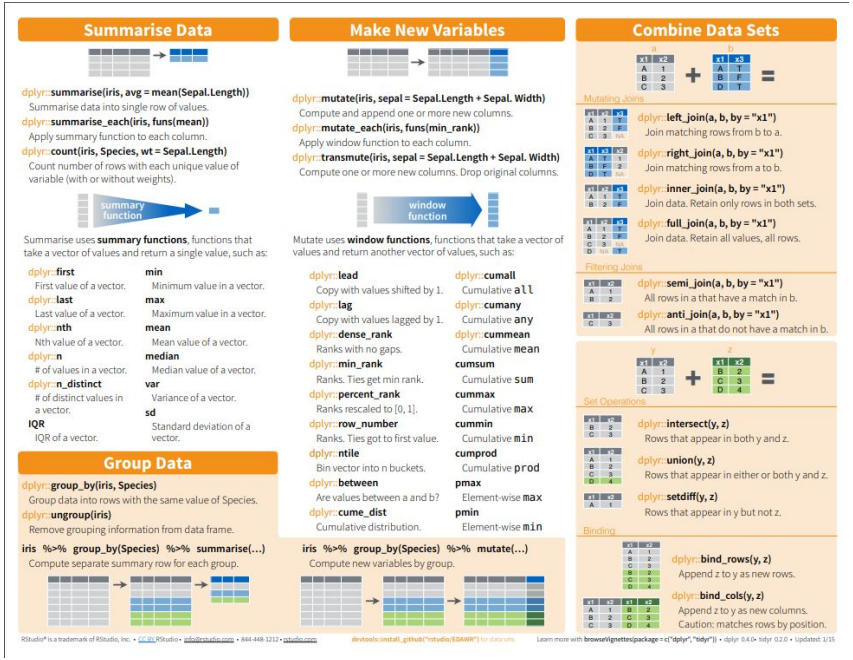
- Applications
- Data Mining and Machine Learning
- Statistical Methodology
- Other Distributions Available in Third-Party Packages ***

- PROGRAMMING LANGUAGE FEATURES
  - Input / Output
  - Object-oriented programming
  - Distributed Computing
  - Included R Packages

R: Why is it cool?

- Data cleaning is super awesome on R





R: Why is it cool?

- So many other cool ways to share your analyses, like:
  - *MAPs website!*
  - *RPubs*
    - *Good option for public stuff*
  - *Github*
    - *Good option to keep private (e.g. de-identified data)*
    - *Super R noobz nerds can get their own github account---private ones are free for people who can prove they're students!!*

Alright fine, you convinced me...

:D Let's install R!

- Step 1: go *here* to download R
- Step 2: choose the r download type based on your computer

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- Download R for Linux
- Download R for (Mac) OS X
- Download R for Windows

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

:D Let's install R!

- Step 3: click "install R for the first time"

Subdirectories:

base	Binaries for base distribution. This is what you want to **install R for the first time**.
contrib	Binaries of contributed CRAN packages (for R >= 2.13.x; managed by Uwe Ligges). There is also information on third party software available for CRAN Windows services and corresponding environment and make variables.
old contrib	Binaries of contributed CRAN packages for outdated versions of R (for R < 2.13.x; managed by Uwe Ligges).
Rtools	Tools to build R and R packages. This is what you want to build your own packages on Windows, or to build R itself.

Please do not submit binaries to CRAN. Package developers might want to contact Uwe Ligges directly in case of questions / suggestions related to Windows binaries.

You may also want to read the R FAQ and R for Windows FAQ.

Note: CRAN does some checks on these binaries for viruses, but cannot give guarantees. Use the normal precautions with downloaded executables.

:D Let's install R!

- Step 4: Click "Download R 3.5.1 for Windows"
- Step 5: Follow all the installation instructions

R-3.5.1 for Windows (32/64 bit)

Download R 3.5.1 for Windows (62 megabytes, 32/64 bit)

Installation and other instructions

New features in this version

If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the md5sum of the .exe to the fingerprint on the master server. You will need a version of md5sum for windows: both graphical and command line versions are available.

Frequently asked questions

- Does R run under my version of Windows?
- How do I update packages in my previous version of R?
- Should I run 32-bit or 64-bit R?

Please see the R FAQ for general information about R and the R Windows FAQ for Windows-specific information.

Other builds

- Patches to this release are incorporated in the r-patched snapshot build.
- A build of the development version (which will eventually become the next major release of R) is available in the r-devel snapshot build.
- Previous releases

Note to webmasters: A stable link which will redirect to the current Windows binary release is <CRAN MIRROR>/bin/windows/base/release.htm.

Last change: 2018-07-02

:D Let's install R!

- Step 6: Go *here* to install R studio, choosing the leftmost option and clicking the "Download" button, choosing the right option for your computer OS when it directs you to the page shown on the pic to the right. Follow download instructions!

Choose Your Version of RStudio

RStudio is a set of integrated tools designed to help you be more productive with R. It includes a console, syntax-highlighting editor that supports direct code execution, and a variety of robust tools for plotting, viewing history, debugging and managing your workspace. Learn More about RStudio features.

	RStudio Desktop Open Source License	RStudio Desktop Commercial License	RStudio Server Open Source License	RStudio Server Pro Commercial License	RStudio Server Pro + RStudio Connect Commercial License
	FREE	$995 per year	FREE	$9,995 per year	$29,995 per year
	DOWNLOAD Learn More	BUY Learn More	DOWNLOAD Learn More	DOWNLOAD Learn More	TALK Learn More
Integrated Tools for R	●	●	●	●	●
Priority Support		●		●	●
Access via Web Browser			●	●	●
Enterprise Security				●	●
Project Sharing				●	●

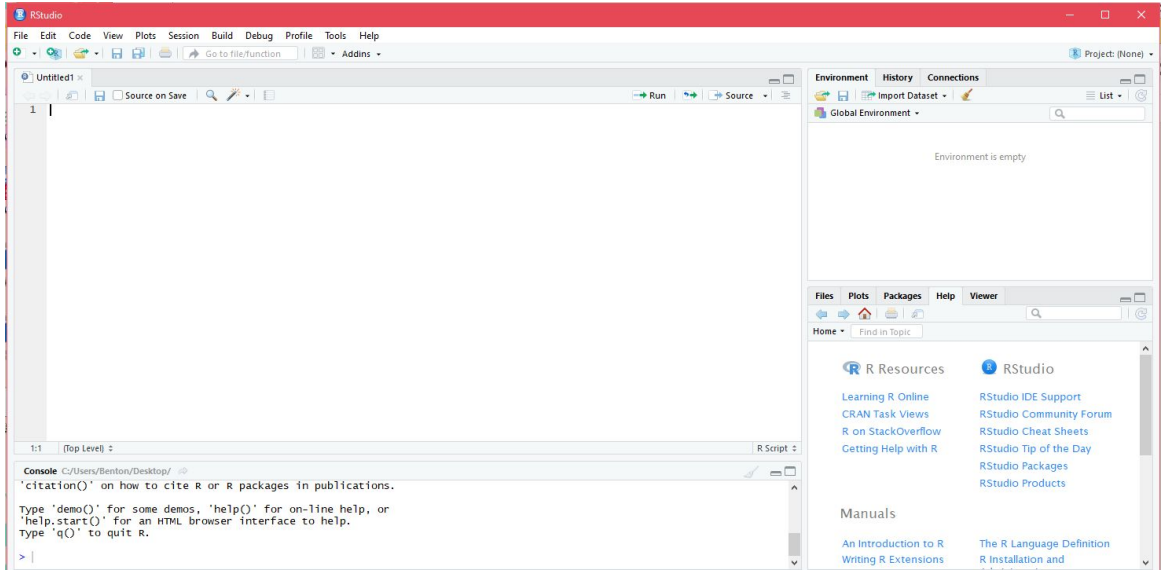
Installers for Supported Platforms

Installers	Size	Date	MD5
RStudio 1.1.453 - Windows Vista/7/8/10	85.8 MB	2018-05-16	bf287e385aef53829204023087e98735
RStudio 1.1.453 - Mac OS X 10.6+ (64-bit)	74.5 MB	2018-05-16	00a0088424ed06ac434f7a966f602b9c
RStudio 1.1.453 - Ubuntu 12.04-15.10/Debian 8 (32-bit)	89.3 MB	2018-05-16	6cfd86770c7b6dbc13e66f4f59c299ce
RStudio 1.1.453 - Ubuntu 12.04-15.10/Debian 8 (64-bit)	97.4 MB	2018-05-16	63e36e8138e369d19f9aaf4b0e995bbc
RStudio 1.1.453 - Ubuntu 16.04+/Debian 9+ (64-bit)	64.4 MB	2018-05-16	85b3e76c9fad4613bc9cf0de1f34b183
RStudio 1.1.453 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)	88.1 MB	2018-05-16	37cade7e162eab62483e6556e39dedee
RStudio 1.1.453 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)	90.6 MB	2018-05-16	44cddd285bc31c41e4eaec1d74b8eebb

Suh, Bro

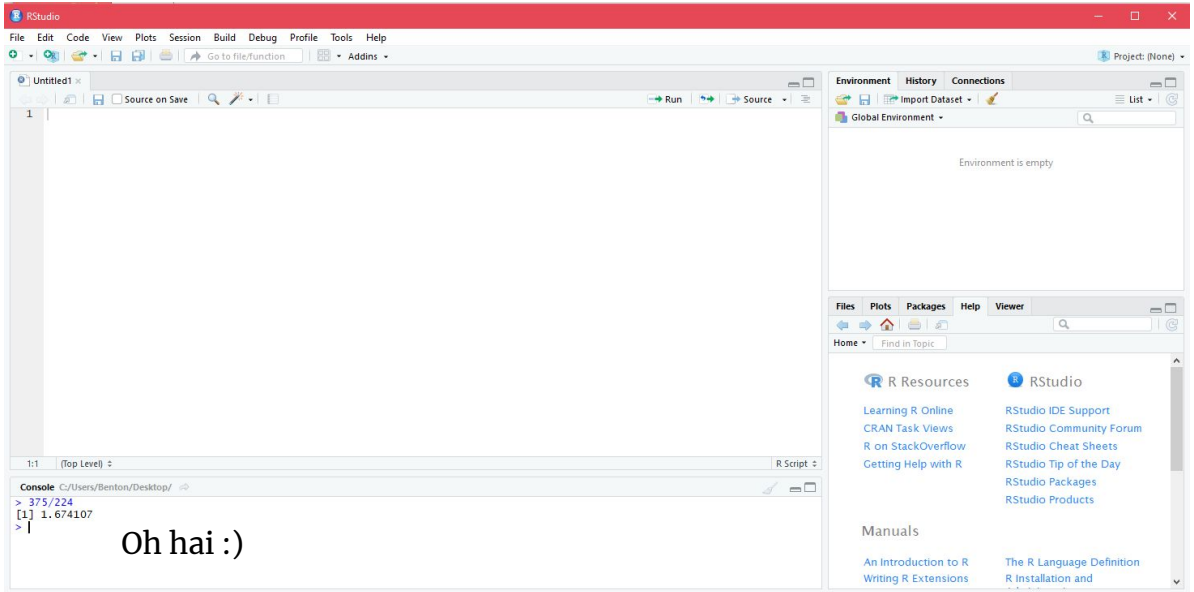
Open R

- You've just downloaded R! Congrats! Open the R Studio link--do you see what I showed you earlier?



If so, let's do something cool

The console of R is a calculator! Type in an equation into it, and see what happens



Wut is an object?

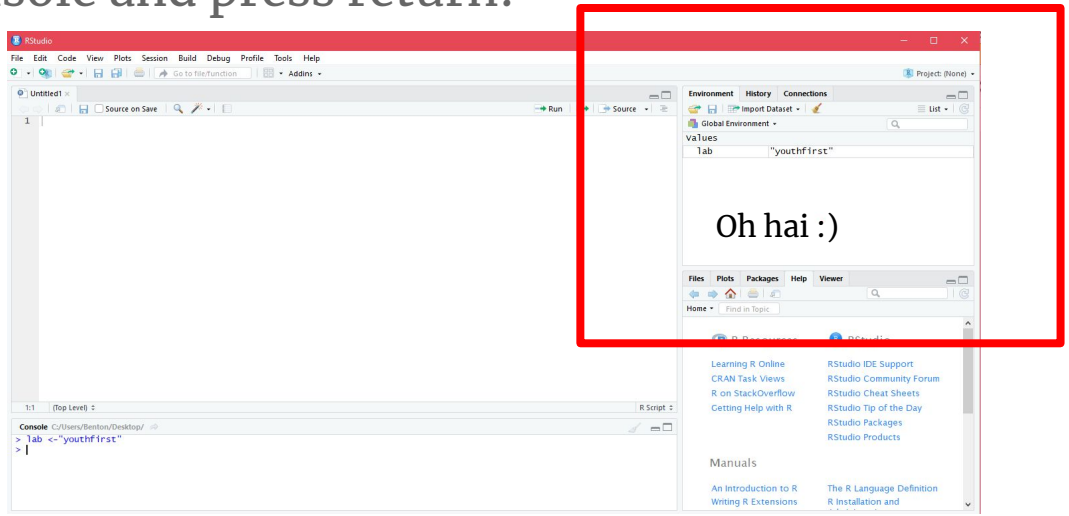
- One of the most basic things you use in R are objects. Objects are defined as:
  - *An object is a data structure having some attributes and methods which act on its attributes.*
  - *...wut?*
  - *Basically, an object is:*
    - *a thing in R*
    - *that is some kind of way*
      - *e.g. a value! A number! A word! A dataset! Whatever! Who am I to set parameters on how you define yourself, object?*
    - *that we can do things to*

Make an object!

- Let's make our first object! Type in the phrase:

  lab <- "YouthFirst"

  into your console and press return!



Make an object!

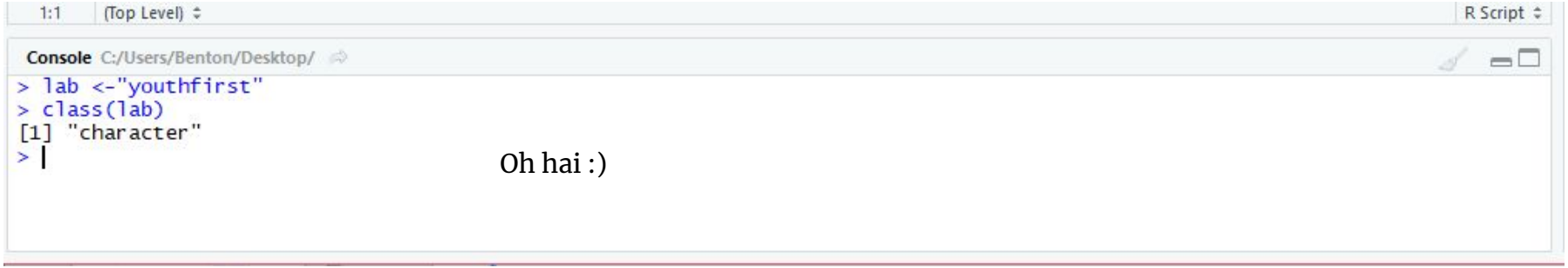
- You just successfully made an object! Congrats! You're basically Peter at this point! 🤓
- Some important R notes:
  - *"<- " is like "=" , but we don't use = in R*
    - *...IDK? It just is.*
  - *R is case sensitive. That means, unlike spss, if you name your object "lab" it will not think that "Lab" is in any way the same thing as "lab"...just go with it.*
    - *Hint...if you can't remember what you named your variable, try looking at the dropdown list that pops up above what you're typing---its like phone a friend for R*
- You can figure out what type of object "lab" is by typing in"

```
class(lab)
```

into your console

Make an object!

- Turns out it is a character. Huh, cool.
- R has a few types of objects:
  - *Character*
  - *Integer*
  - *Numeric*
  - *Logical*
  - *Matrix*
  - *Data Frame*



Buckle your seatbelt

Let's see if we can load in a dataset...

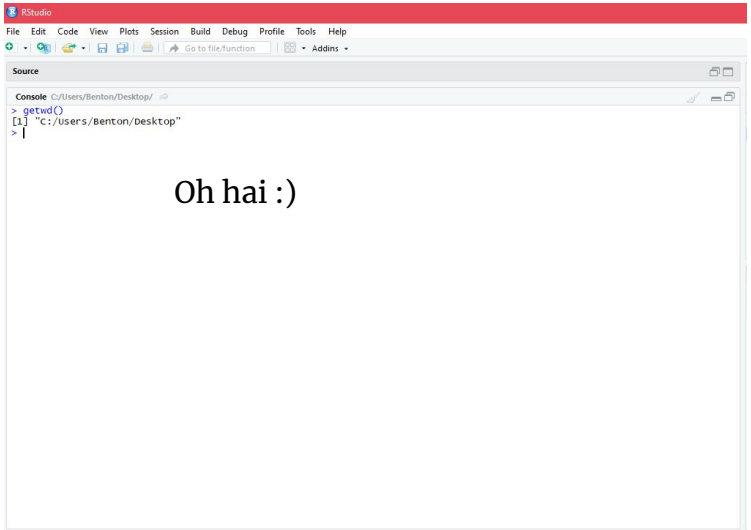
I emailed you some files. Let's see if we can load them into r, and play around with the data by seeing the class of each variable!

Go into your email, download the 2 files I sent you, and put them on your desktop.

Let's see if we can load in a dataset...

Working directory is basically where R is getting its schtuff from.

You can see what your current working directory is by typing into the console the command: "**getwd()**"



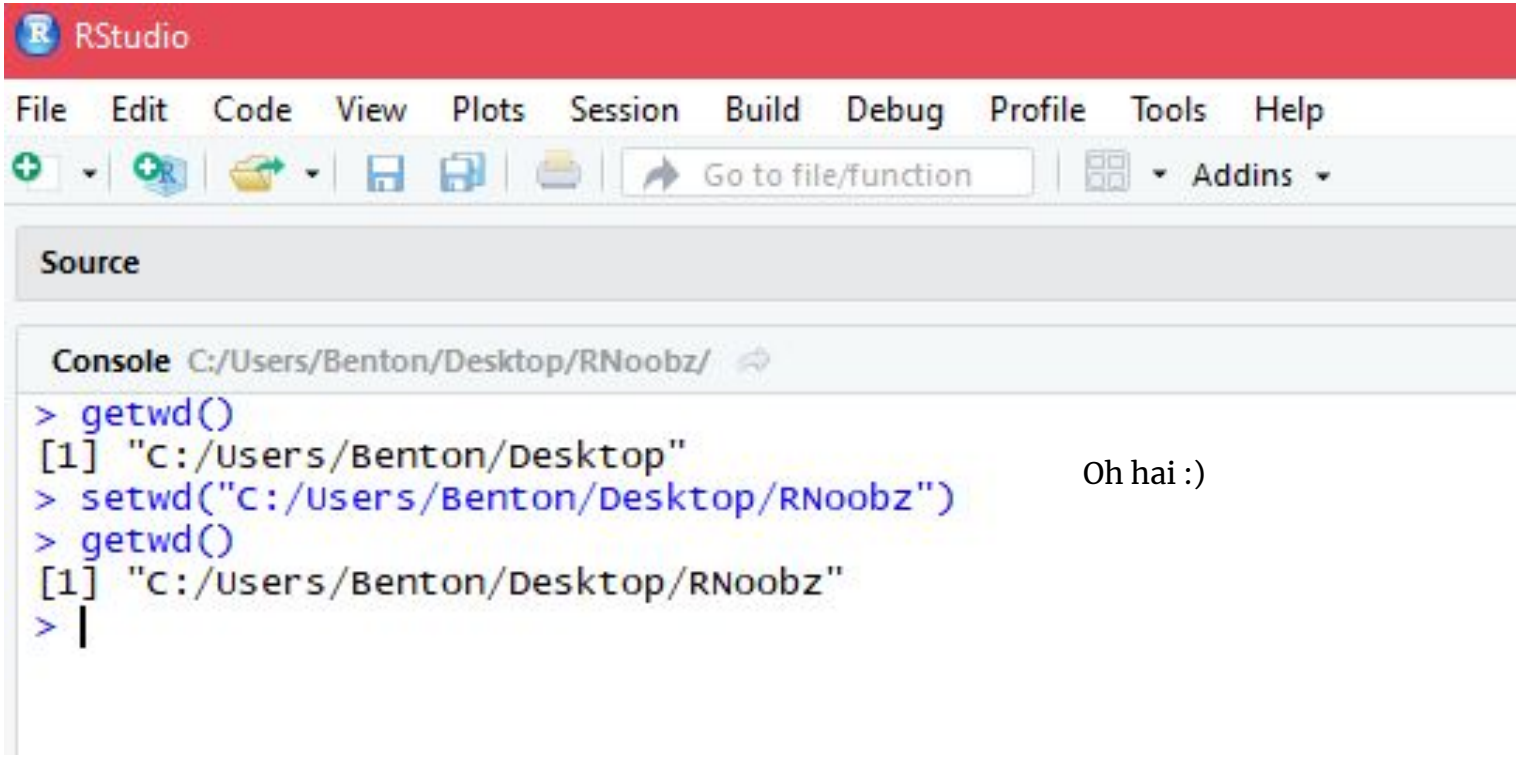
Oh hai :)

Let's see if we can load in a dataset...

You can change your working directory by typing in the command "**setwd()**" and putting a path to the place you'd like to set your directory in between the parentheses. Be sure you put quotes around it.

For example, say I wanted to make my working directory a folder on my desktop, I'd do the following:

Let's see if we can load in a dataset...

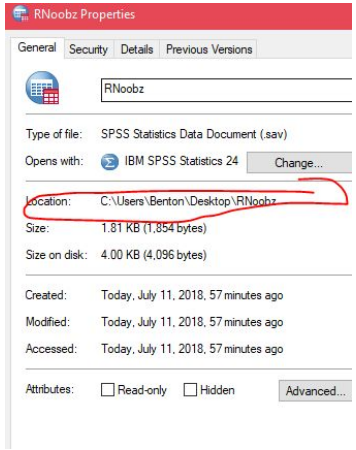


Oh hai :)

Let's see if we can load in a dataset...

- Small note here:
  - *Paths are different in Mac vs. Windows. I still haven't memorized the difference (the slashes are different--it's weird).*

    If you really want to know how to do the right paths to your director, files, etc, you can always find an example from getting info from a file in the working directory you're using/wanting to set...



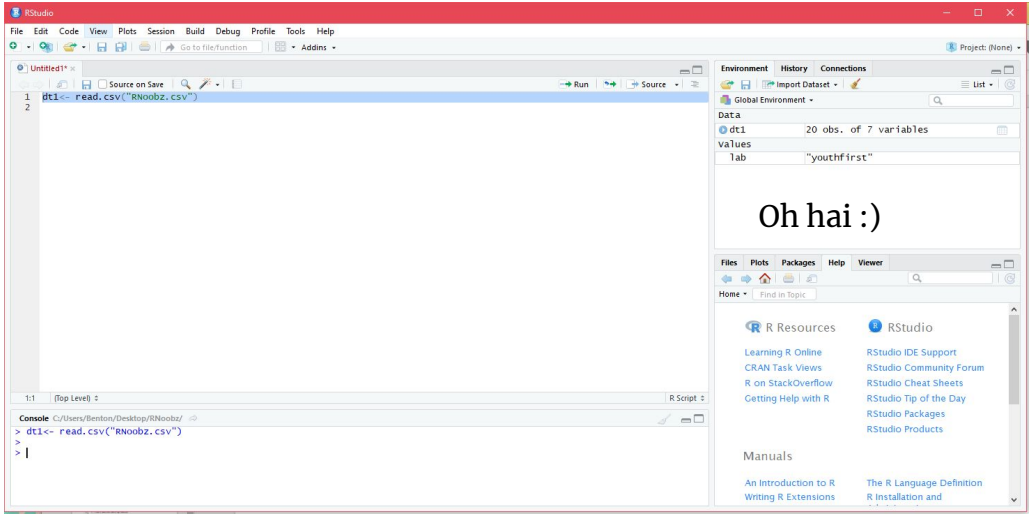
Let's see if we can load in a dataset...

OK so, let's load in the CSV file, and name it dt1

Type in the command:

dt1<-
read.csv("RNoobz.csv"**)**

See dt1 in your data section of the global environment?



Let's see if we can load in a dataset...

Let's see what kind of class of object dt1 is...

class(dt1**)**

```
Console C:/Users/Benton/Desktop/RNoobz/
> dt1<- read.csv("RNoobz.csv")
>
> class(dt1)
[1] "data.frame"
>
```

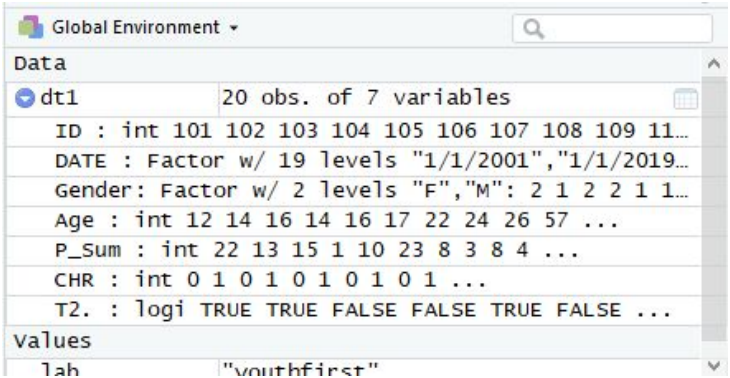
Oh hai :)

OK, neat! It is a data frame! Cool beans!

Let's see if we can load in a dataset...

If you click the little blue button on the left, that can show you what kind of variables you've got going on, and their respective classes...

Let's take a look at these bad boys!!



Oh hai :)

Let's see if we can load in a dataset...

If you type in the name of your dataframe, then $, then the variable name, that is a way we can let R know we are talking about a specific variable...like, for gender...

```
Console C:/Users/Benton/Desktop/RNoobz/
> dt1$Gender
 [1] M F M M F F F M F M M F M F M F M F F M
Levels: F M
>
```

```
> class(dt1$Gender)
[1] "factor"
>
```

```
> summary(dt1$Gender)
 F  M
10 10
>
```

Let's see if we can load in a dataset...

OK cool, let's see if we can do this for the variables in our dataset...

Let's see if we can load in a dataset...

OK we are bored...fine.

Let's install a package!!

Type in:

install.packages("foreign"**)**

Then:

library(foreign**)**

Into your console

```
> install.packages("foriegn")
Installing package into 'C:/Users/Benton/Documents/R/win-library/3.5'
(as 'lib' is unspecified)
Warning in install.packages :
  package 'foriegn' is not available (for R version 3.5.1)
> library(foreign)
>
```

Let's see if we can load in a dataset...

This package lets us import weirdo types of files into R, like SPSS files.

It is super clutch.

Try typing in the code pictured here...

Do you see a new object called dt2 in your global environment?

```
> dt2 <- read.spss("RNoobz.sav")
re-encoding from UTF-8
>
```

So...did it work?

This package is nice, because it can import all kinds of neat-o labels and values from SPSS data files!

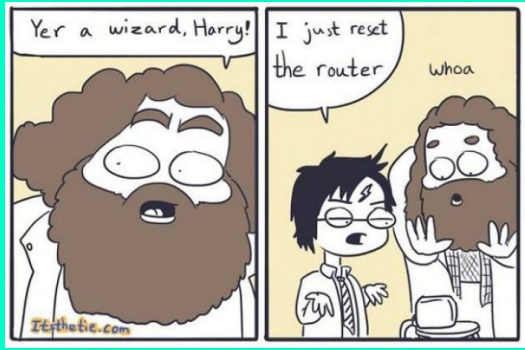
```
Console C:/Users/Benton/Desktop/RNoobz/
> dt2 <- read.spss("RNoobz.sav")
re-encoding from UTF-8
> dt2$Gender
 [1] Male   Female Male   Male   Female Female Female Male   Female Male   Male   Female Male   Female Male   Female
[17] Male   Female Female Male
Levels: Female Male
>
```

Are we feeling stressed or ambitious?

R Snooping

In Class homework assignment!!

See if you can figure out the class of every variable in the dataset I gave you, and let me know how variables may be different in the CSV file vs. the SPSS file!



CONGRATS, R NOOBZ! YOU ARE LITERALLY A DATA WIZARD NOW!

Next Class:
Data cleaning in R!

Homework, if you feel like it:

Load in the all available data dataset into R, and look at the variables!

Questions?
Comments?
Thanks!!!
Go team!