Data Analytics
CS390
Chap 2, Intro to R

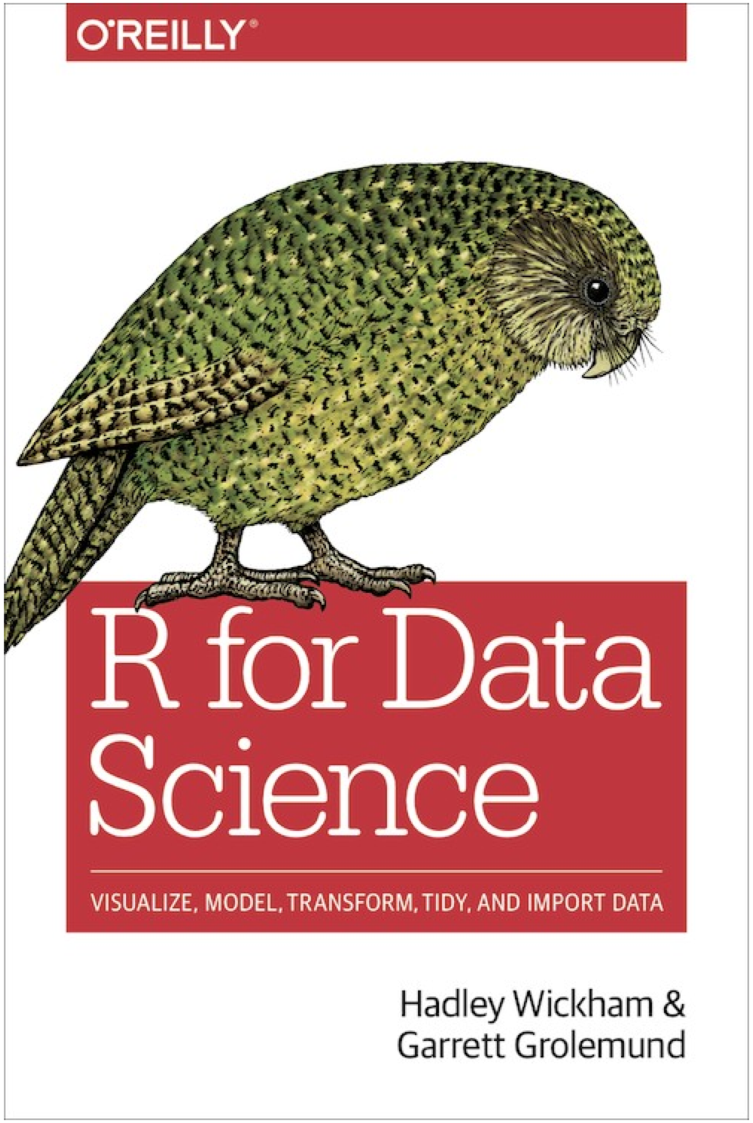
Week 3
Fall 2018
Oliver Bonham-Carter

Where To Now?

- Google Analytics is a tool allowing for convenient analysis of web sites
- The code was written by developers for this purpose.
- What if you need tools and there are no current developers to create them?

Develop Your Own Tools!!

We will be using the Book

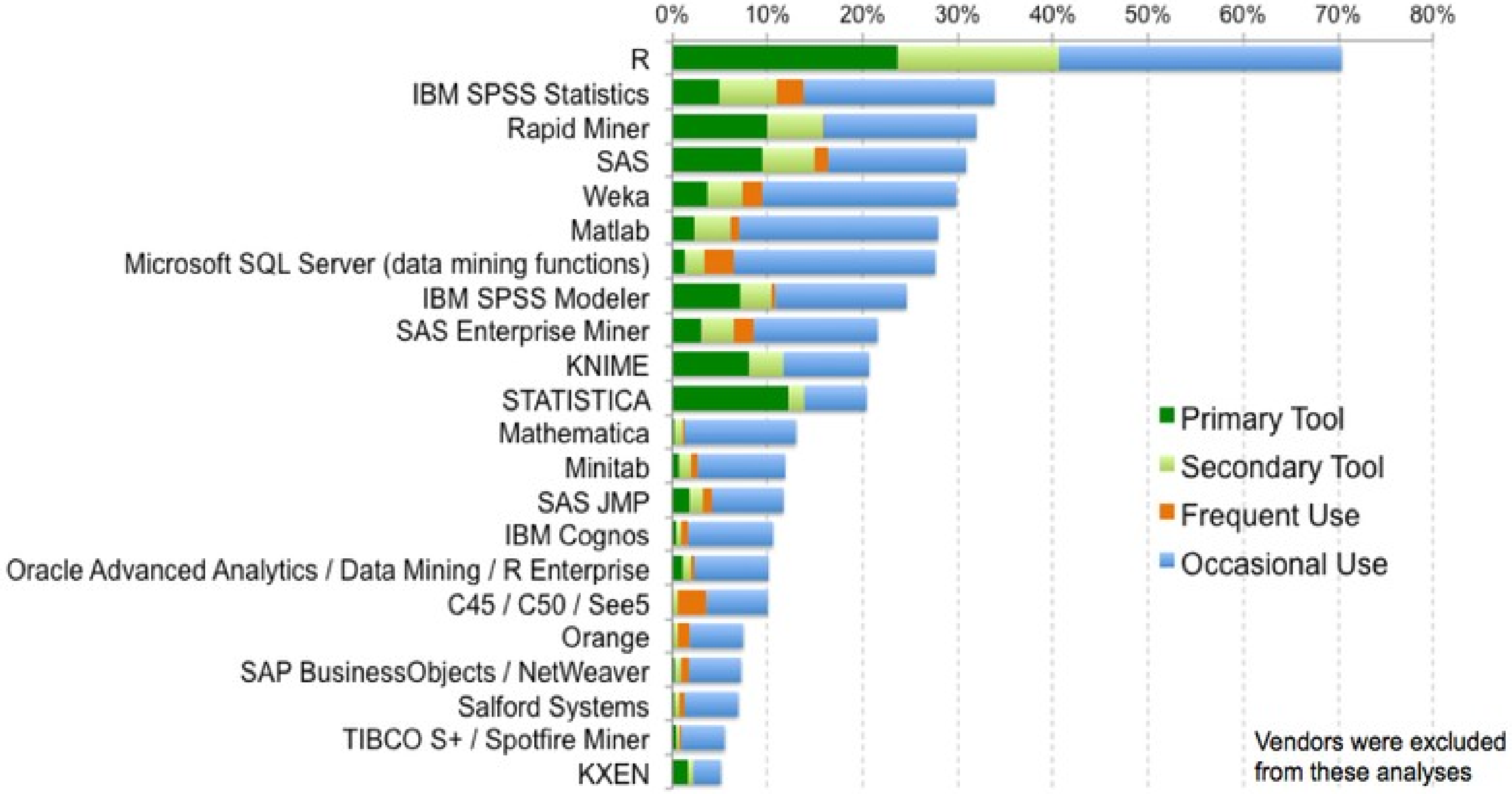


- Note the chapters between the book and the website are not numbered identically!
- Book:
  - Chap 1: Data Visualization with ggplot
  - **Chap 2: Workflow; Basics**
- On the web site:
  - http://r4ds.had.co.nz/
  - Chap 3: Data Visualization
  - **Chap 4: Workflow; Basics**

The R Programming Language

- https://www.r-project.org/
- What is the R language?
  - An open source, well-developed programming platform for work in statistics, mathematics and data analytics
  - Built-in libraries to simplify programming
  - Language includes conditionals, loops, user-defined recursive functions and input and output facilities.
- Community Blogs:
  - https://www.r-bloggers.com/
  - https://twitter.com/rstudiotips

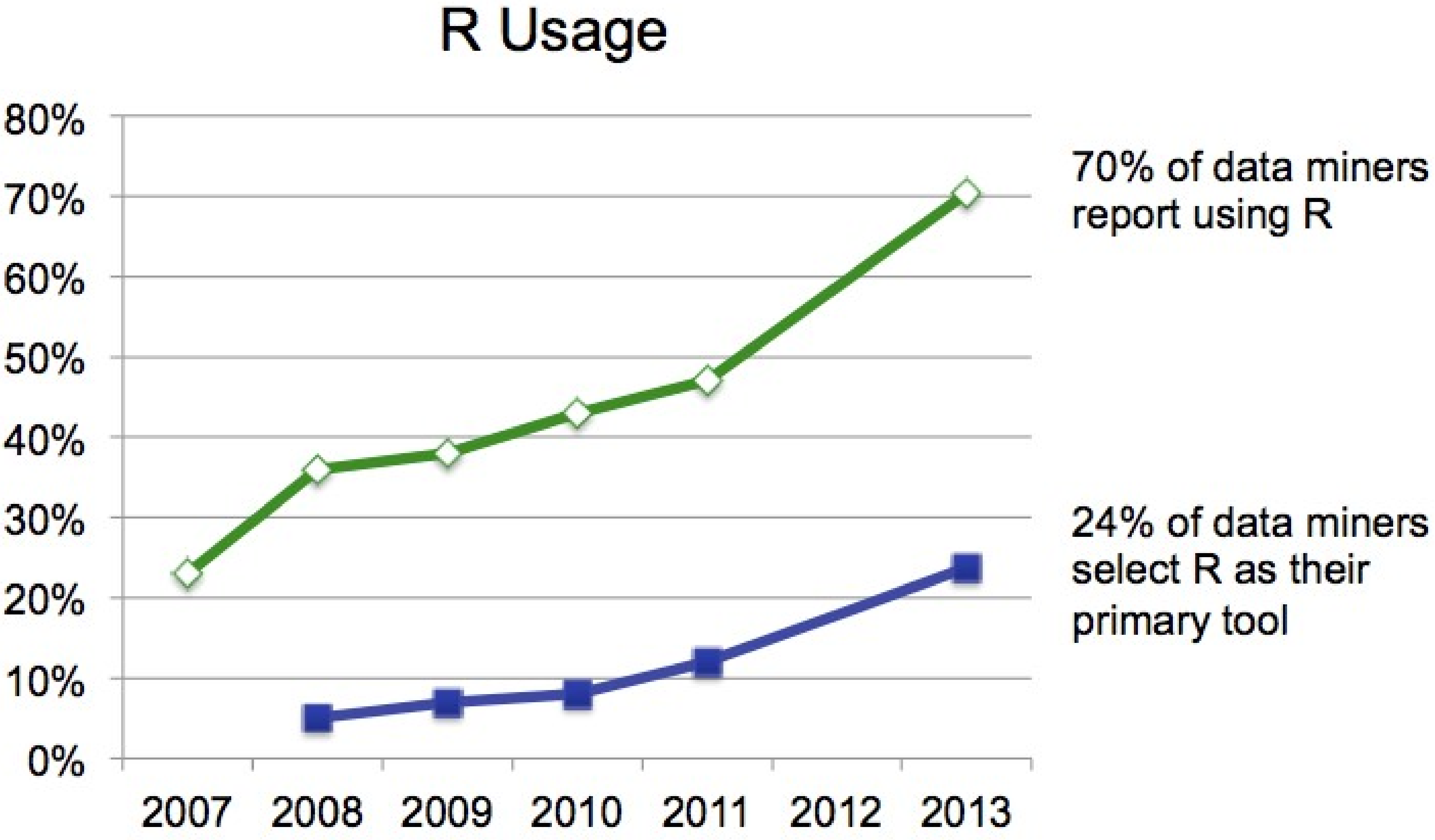
R: The Most Popular Data Mining Tool



Vendors were excluded from these analyses

http://blog.revolutionanalytics.com/2013/10/r-usage-skyrocketing-rexer-poll.html

R is Exploding in Growth



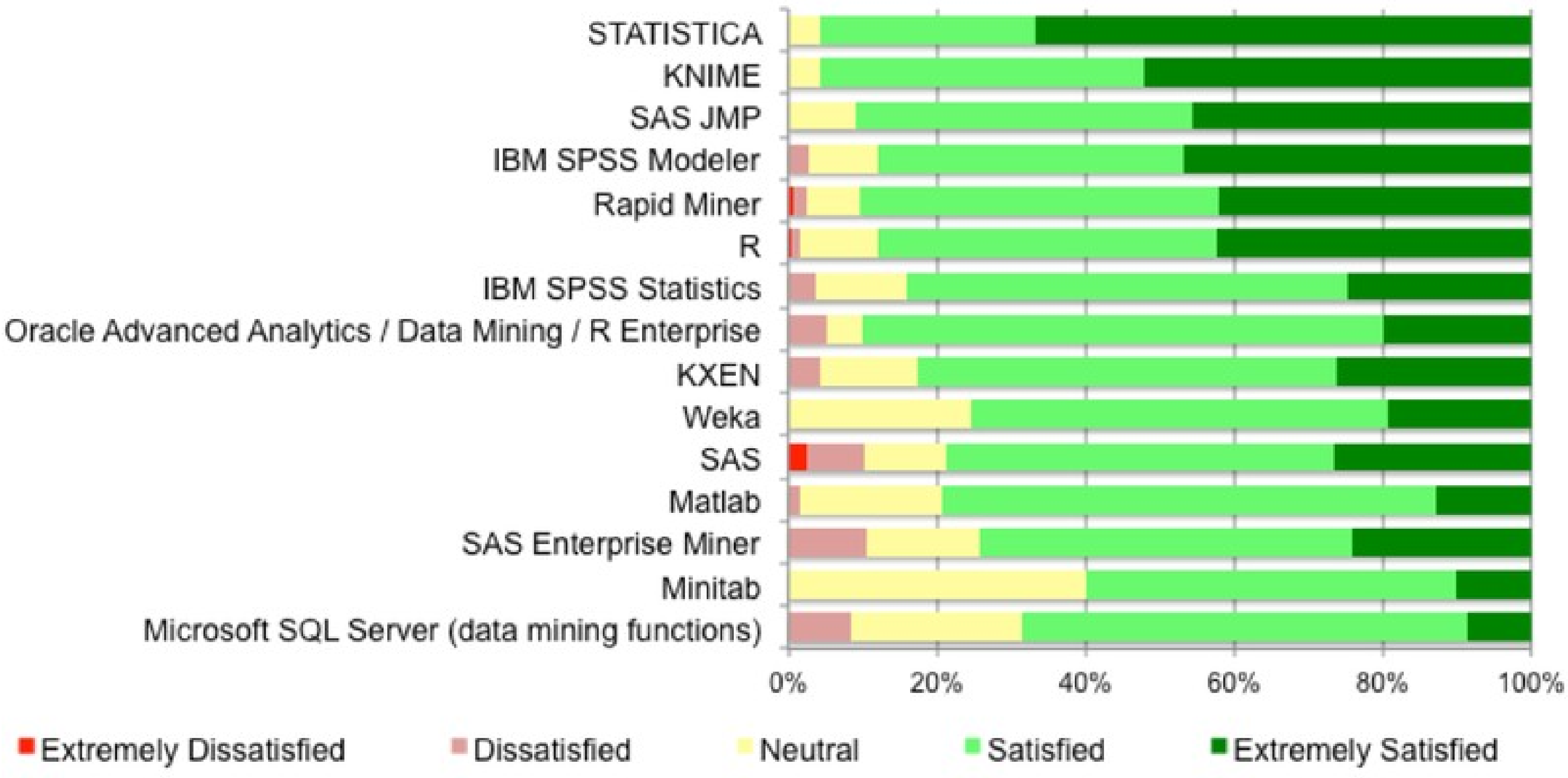
R Usage

70% of data miners report using R

24% of data miners select R as their primary tool

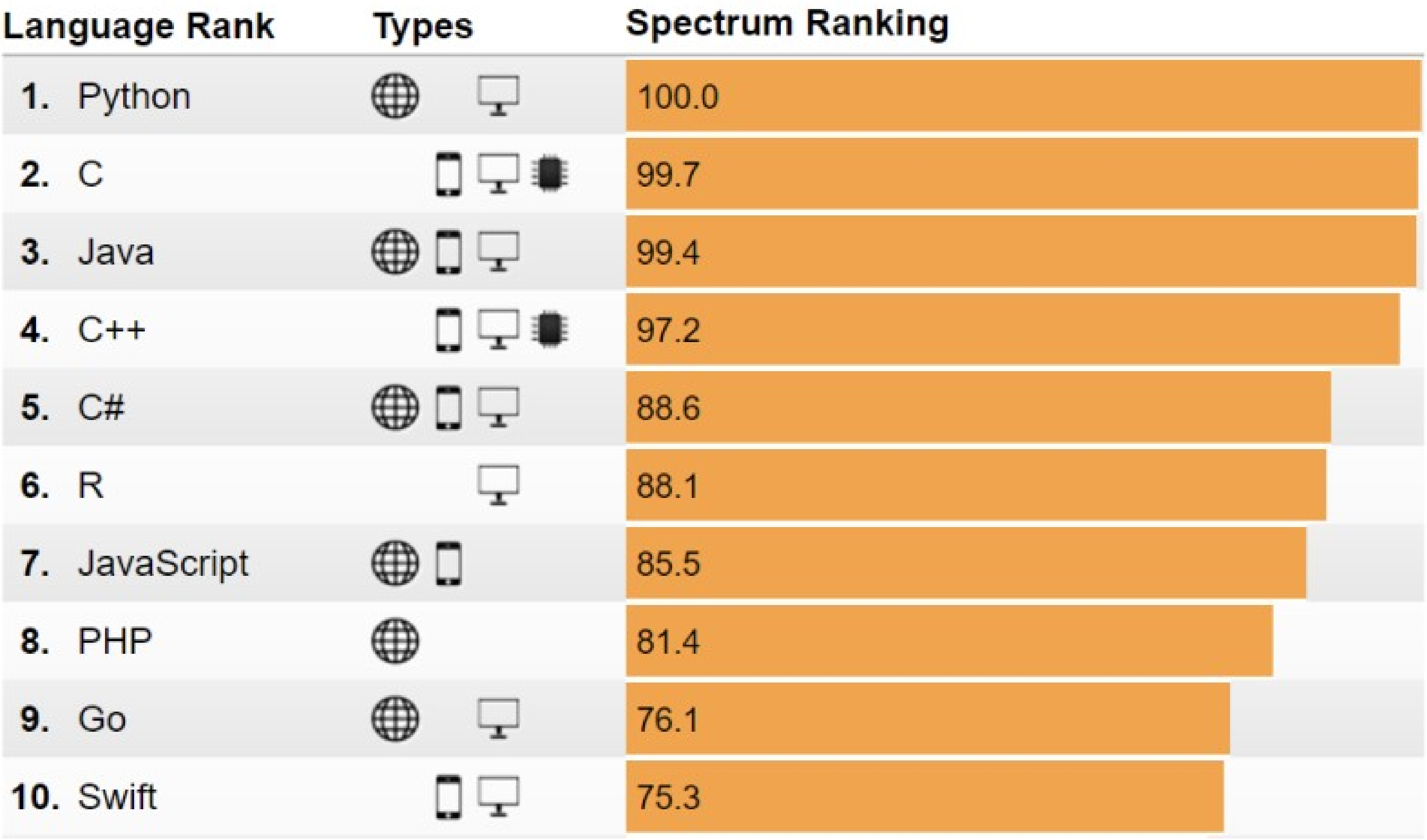
http://blog.revolutionanalytics.com/2013/10/r-usage-skyrocketing-rexer-poll.html

Most users are satisfied with R



http://blog.revolutionanalytics.com/2013/10/r-usage-skyrocketing-rexer-poll.html

Ranking To Others: IEEE 2017



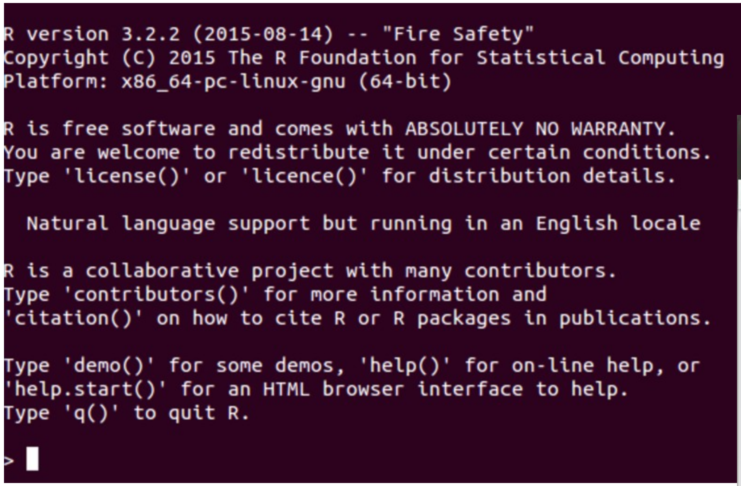
Language Rank	Types	Spectrum Ranking
1. Python	🌐 🖥	100.0
2. C	📱 🖥 ▣	99.7
3. Java	🌐 📱 🖥	99.4
4. C++	📱 🖥 ▣	97.2
5. C#	🌐 📱 🖥	88.6
6. R	🖥	88.1
7. JavaScript	🌐 📱	85.5
8. PHP	🌐	81.4
9. Go	🌐 🖥	76.1
10. Swift	📱 🖥	75.3

Find more amazing studies about R:

http://blog.revolutionanalytics.com/2018/06/pypl-programming-language-trends.html

Let’s Try It Out!

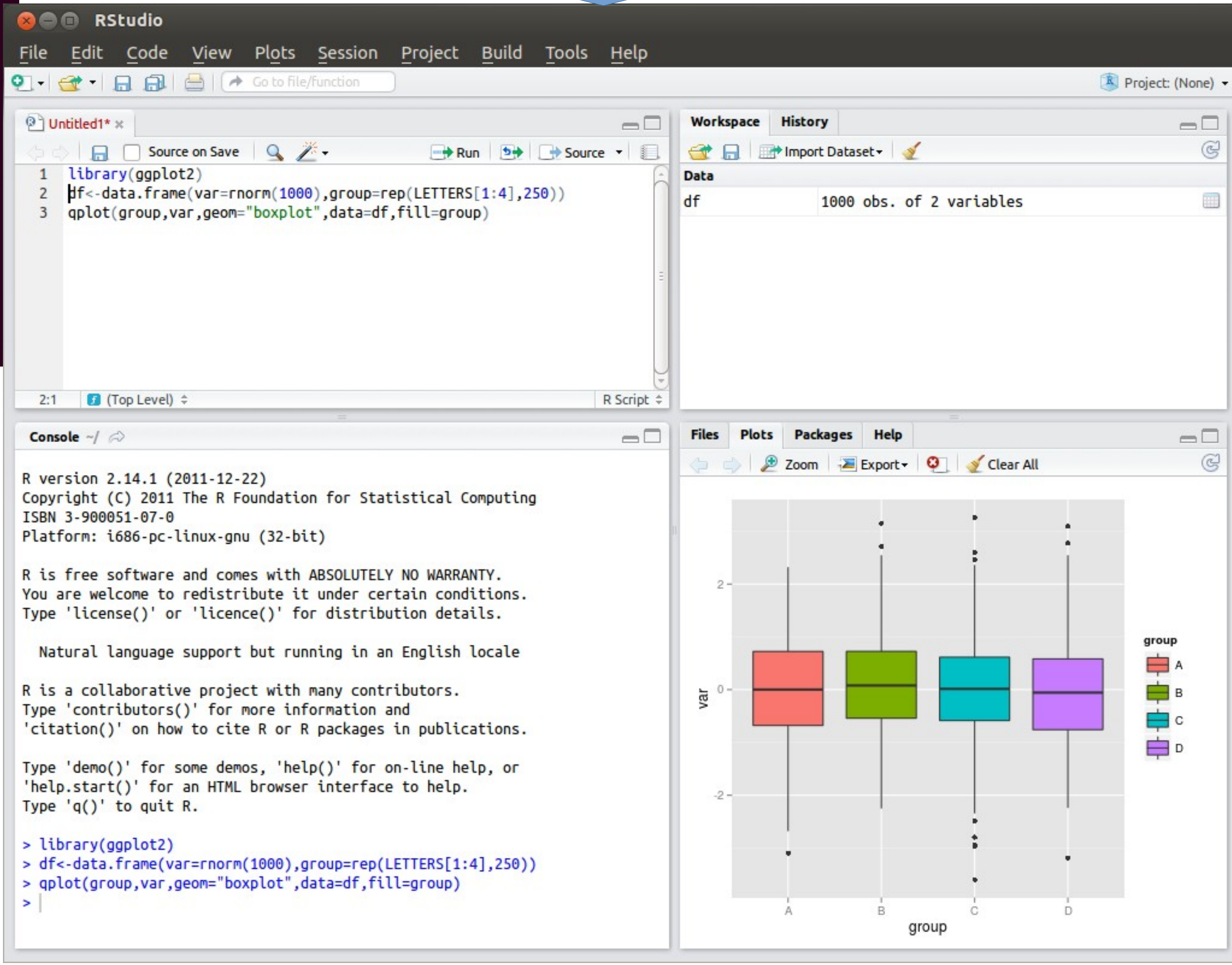
- Wait! R or Rstudio?



```
R version 3.2.2 (2015-08-14) -- "Fire Safety"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (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.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> 
```

To run:
Type “*R*” at terminal

To run:
Find its icon or type *rstudio at terminal*



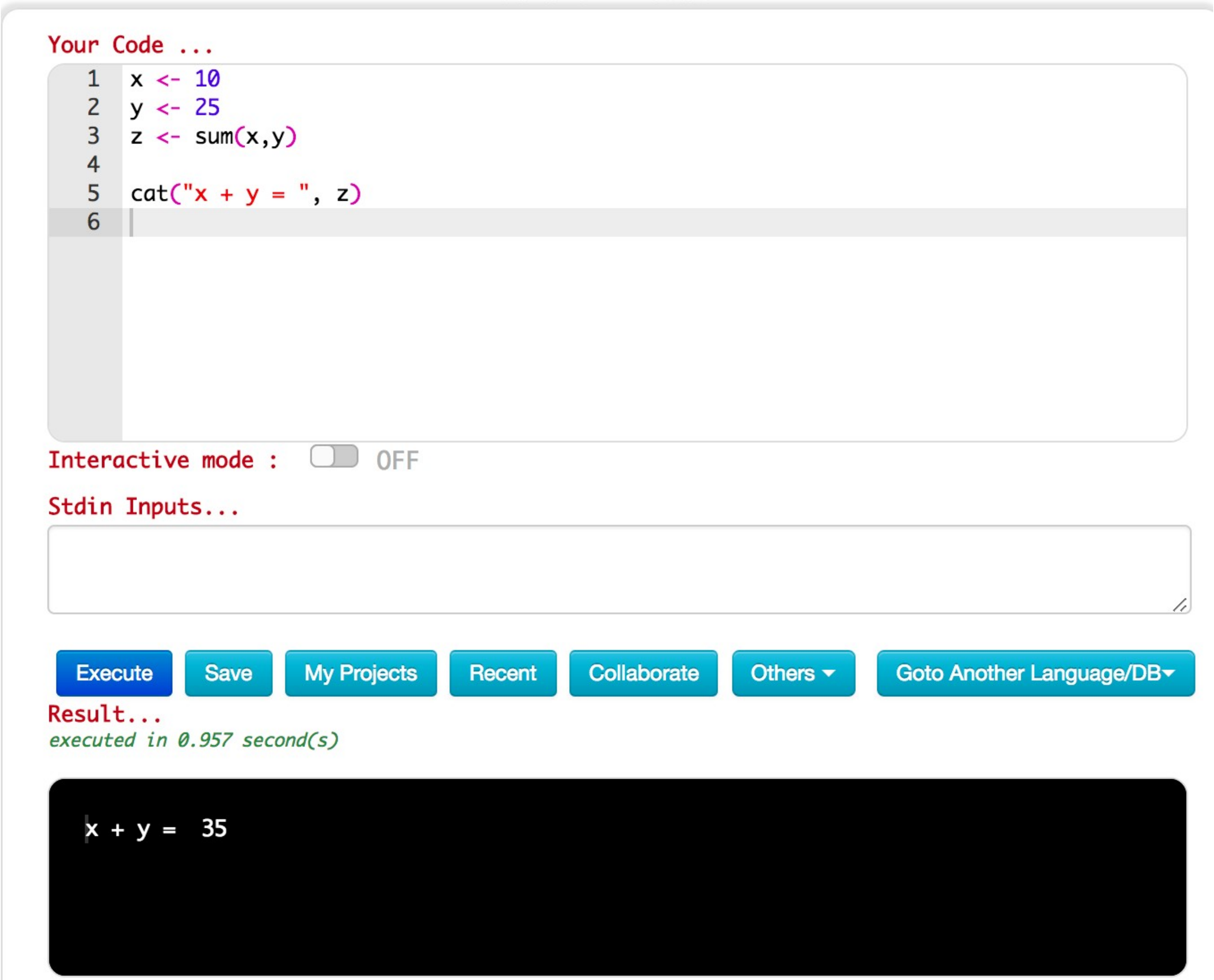
```
library(ggplot2)
df<-data.frame(var=rnorm(1000),group=rep(LETTERS[1:4],250))
qplot(group,var,geom="boxplot",data=df,fill=group)
```

Data	
df	1000 obs. of 2 variables

```
R version 2.14.1 (2011-12-22)
Copyright (C) 2011 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: i686-pc-linux-gnu (32-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.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(ggplot2)
> df<-data.frame(var=rnorm(1000),group=rep(LETTERS[1:4],250))
> qplot(group,var,geom="boxplot",data=df,fill=group)
> 
```

Failing that: R by Jdoodle

- https://www.jdoodle.com/execute-r-online



Your Code ...

```r
x <- 10
y <- 25
z <- sum(x,y)

cat("x + y = ", z)

```

Interactive mode : OFF

Stdin Inputs...

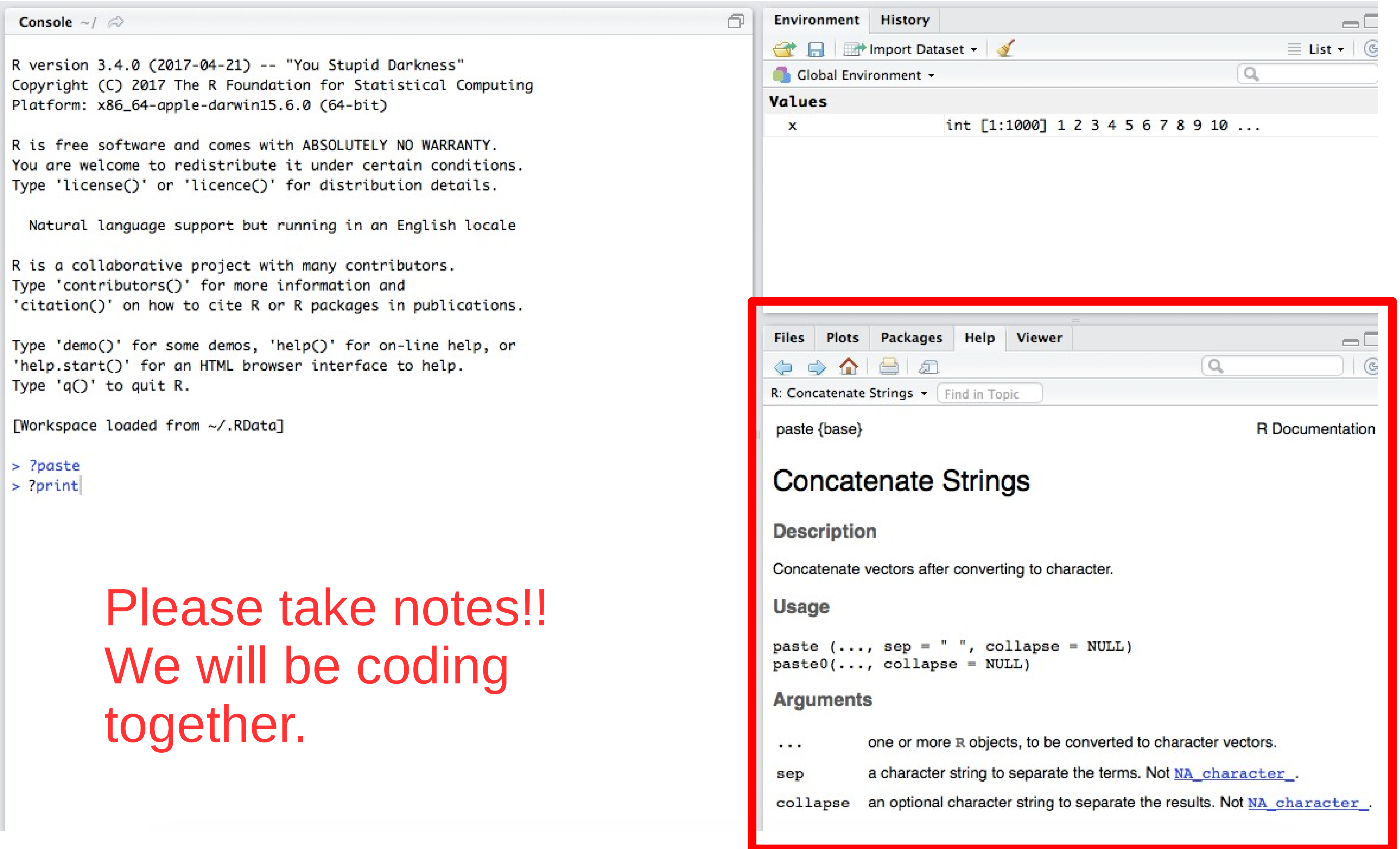
Result...

executed in 0.957 second(s)

```
x + y =  35
```

Getting Help in R

- Online help: place a "?" in front of a keyword
  - Ex: ?print



```
R version 3.4.0 (2017-04-21) -- "You Stupid Darkness"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (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.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/.RData]

> ?paste
> ?print
```

Values

```
x                int [1:1000] 1 2 3 4 5 6 7 8 9 10 ...
```

paste {base} R Documentation

Concatenate Strings

Description

Concatenate vectors after converting to character.

Usage

```
paste (..., sep = " ", collapse = NULL)
paste0(..., collapse = NULL)
```

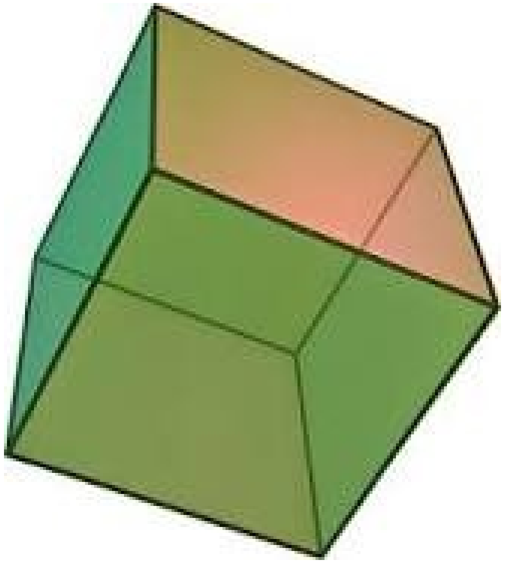
Arguments

`...`	one or more R objects, to be converted to character vectors.
`sep`	a character string to separate the terms. Not NA_character_.
`collapse`	an optional character string to separate the results. Not NA_character_.

Please take notes!!
We will be coding together.

Variable Names

- Variable Names:
  - Begin with a letter, and can only include letters, numbers, periods and hyphens.
  - Hyphens: “-”
  - Periods: “.”
- SnakeCase (recommended by book)
  - val_of_height,
  - val_of_length,
  - val_of_width



Variable Names

- CamalCase:
  - valOfHeight,
  - valOfLength,
  - valOfWidth
- Period.Case
  - Val.of.height,
  - Val.of.length,
  - Val.of.width

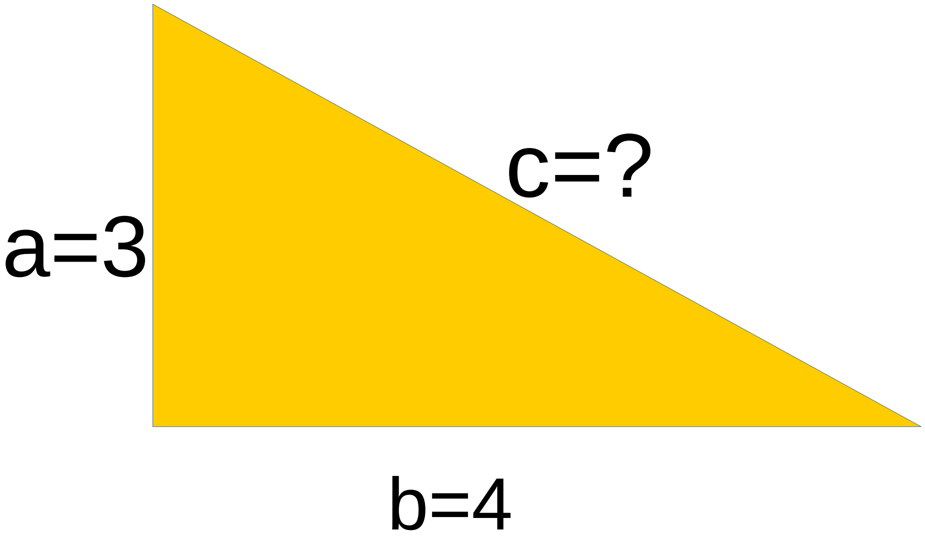
- What-EVER.Case
  - Val.ofHEIGHT,
  - Val.Of_Length,
  - Val.oF.Width

Basic Math

- Mathematics
  - Addition: 1+1
  - Subtraction: 1-1
  - Multiplication: 3*7
  - Division: 0.25
- More complicated math, var assignments:
  - 4*(7+3)/10+1   Note: watch the order of operations!
  - Parameter of circle (*C* = 2 * pi * *r*)
    - *R* <- 4, Note the "<-" means *equal* in R.
    - *C* <- 2 * pi * R = 2 * 3.1415 * 4
    - *C* is 25.13274

Variables and Assignments

- X <- 10.
- You could also use “X=10” but this is not traditional programming in R...
- Hypotenuse = c = sqrt(a^2 + b^2)
- A <- 3
- B <- 4
- C <- sqrt(3^2 + 4^2)
- C is ??



Logical Operations

- Booleans: Returning True or False:

3 > 4, 3 < 4,

2 + 4 == 6,

2 + 3 == 4 + 1

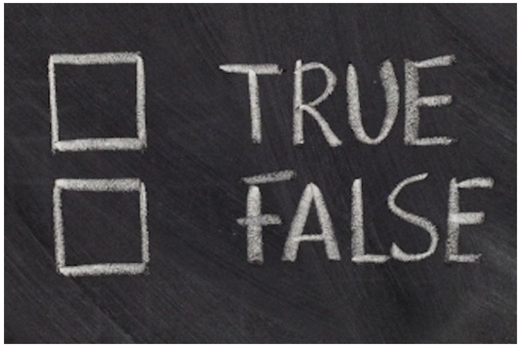
T == TRUE

F == FALSE

3 + 4 != 5

3 + 4 == 7

5 * 2 != 11

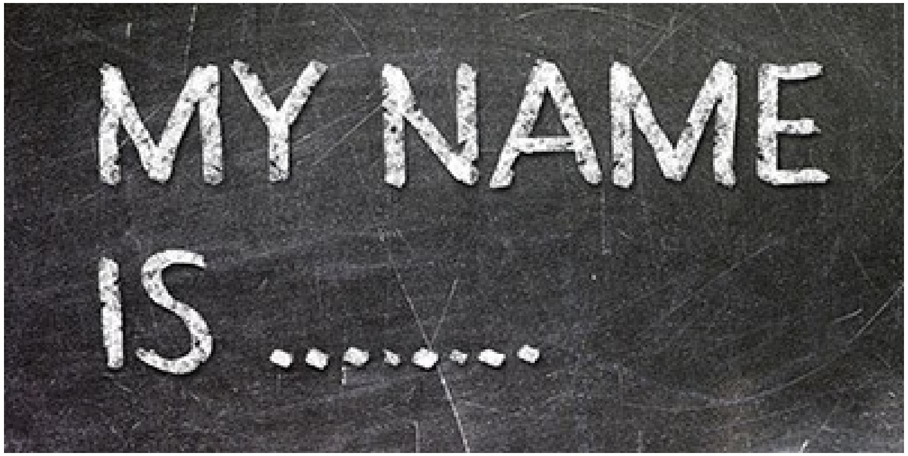


Try some of These in R!

- Logical **AND** (&&)
  - F && F: F
  - F && T: F
  - T && F: F
  - T && T: T
- Logical **OR** (||)
  - F || F: F
  - F || T: T
  - T || F: T
  - T || T: T
- Logical **NOT** (!)
  - !F: T
  - !T: F

Simple Steps

- Strings
  - “Hello World”
- Concatenation of strings
  - H <- “Hello”
  - W <- “world”
  - paste(H,W, sep = “ ”)
    - What is the result here??



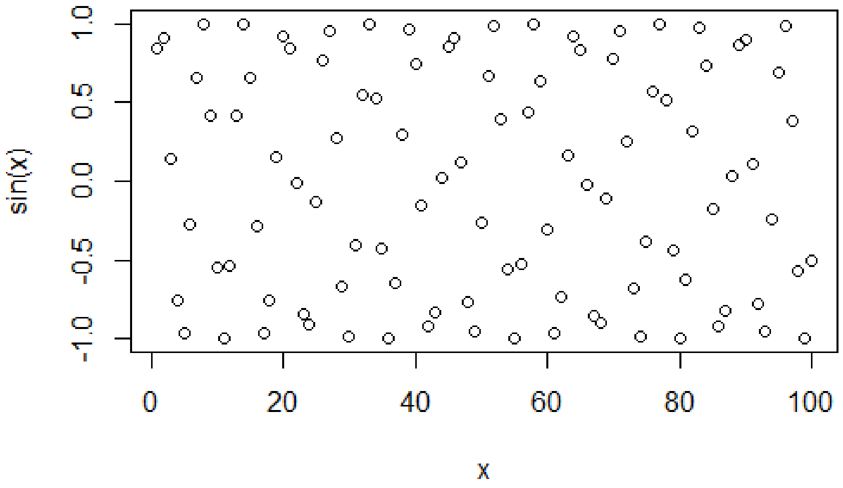
- You try: print your full name!
  - name <- first-*name*,
  - Lastname <- *last-name*

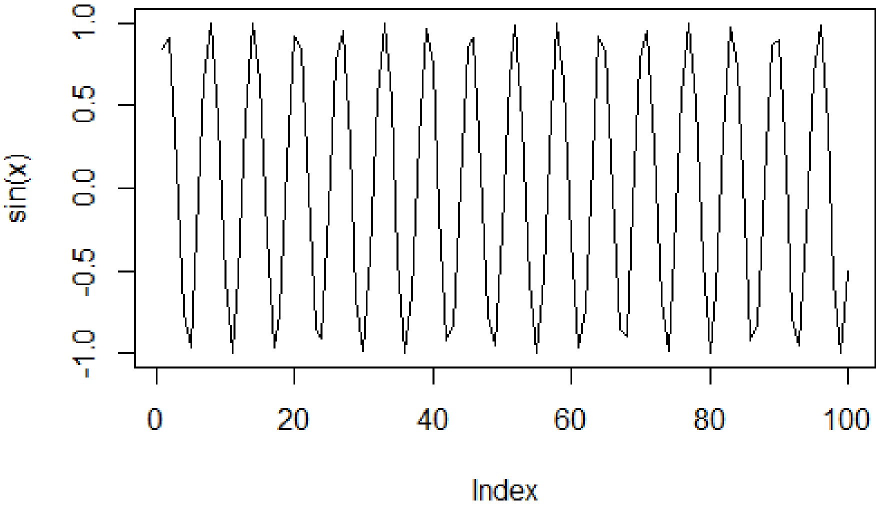
Built-in Functions

- R has a large collection of built-in functions:
  - function_name(arg1 = val1, arg2 = val2, …)
- Try calling this function:
  - Seq(0,10)
  - Gives a sequence, *S* ={0,…,10}
  - What happens when you press TAB after typing, “seq”?
- Use the sum() function to add two numbers.
- Sum() to add three numbers?
- Sum() to add a whole lot of numbers?

Simple Plots

- x<- seq(1,100) # assign *x* to the sequence 1 to 100
- plot(x) # plot this sequence
- plot(sin(x)) or plot(x,sin(x)) # left plot
- plot(sin(x)) or plot(x,sin(x), type = “l”) # right plot





Now, You Try

- Use R to write a command that...
  - Find the **sum** of all numbers, 0 through 100
  - Find the **sum** of all numbers, 0 through 10000

    (now, set a variable equal to the sequence first)
  - Use the plot function, **plot(x,y,type = “l”)** to plot a line of the function, f(*x*) = sin(*x*) for *x* in {0, …, 30}
  - Plots the function, f(*x*) = cos(x) for *x* in {0, …, 30}
  - Plots the function, f(*x*) = tan(x) for *x* in {0, …, 30}

Exiting R:
q()