



Statistical Genomics

Lab 1: Intro to R, RStudio, and R Markdown

Meijing Liang

Washington State University



What will be Covered today?

1. What is R?
2. What is RStudio?
3. What is R Markdown?
4. Basics of R Syntax
5. Hands-On Activities





What is R ?

Definition: R is a programming language and environment for *statistical* computing and graphics.

Origin: Developed in the 1990s by Ross Ihaka and Robert Gentleman at the University of Auckland.

Purpose: Open-source alternative to the S programming language.



Ross Ihaka



Robert Gentleman



Why use R ?

Key Features:

- Free and open-source.
- Extensive library of statistical and data visualization tools.

*The CRAN package repository features **21,862** available packages.*

- Highly extensible and customizable.
- Active community and support.

Applications:

- Data analysis, visualization, machine learning, and reproducible research.



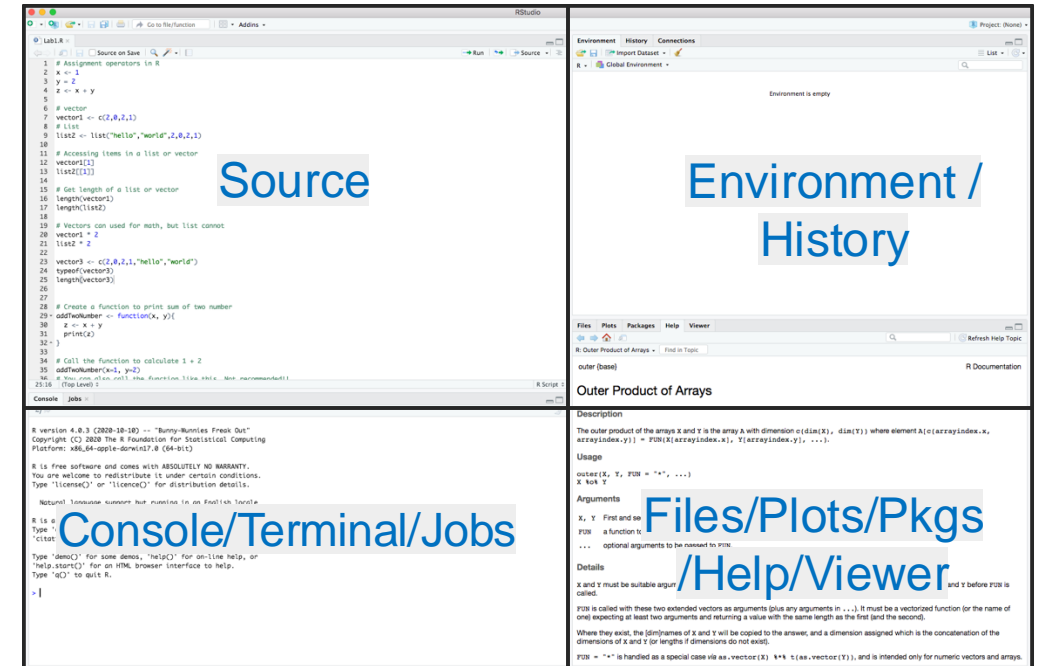
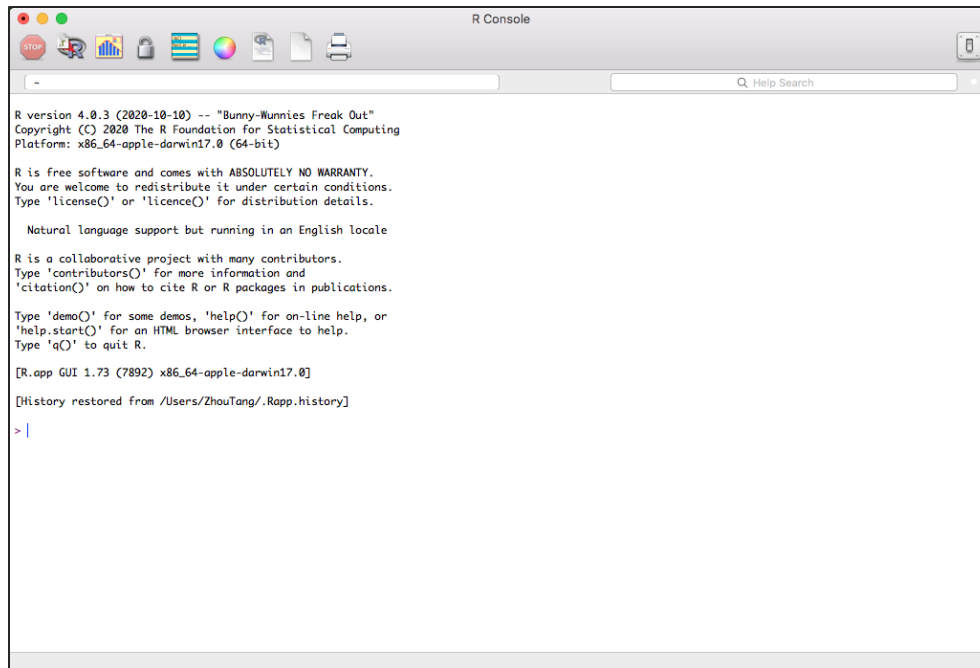


- Download and install 

<https://cran.r-project.org/>

- Download and install  studio

<https://rstudio.com/products/rstudio/download/#download>





projects.R x datavis.Rmd x

Knit Insert Run

```

400 {r, eval=FALSE}
401 # generate distance/delay scatter aesthetic
402 ggplot(delay, aes(x = dist, y = delay))
403   geom_point(aes(size = count), alpha = 0.5)
404   scale_size_area(max_size = 3)
405

```

aesthetic
aseptic
ascetic
Ignore word
Add to user dictionary

delay

count

100
200
300

401:40 Chunk 22 R Markdown

Console Terminal Find in Files Jobs

Results for "context" in ~/git/rsrecovr

Replace with: session Replace All

~/git/rsrecovr/R/all.R

```

37: results <- rbind(results, cbind(project = NA, contextsession = NA,

```

~/git/rsrecovr/R/projects.R

```

10: # list all the contextsession IDs
11: contextsessions <- list.files(state_folder, pattern = "[a-zA-Z0-9]
13: # recover the sources from each contextsession
14: results <- lapply(contextsessions, function(contextsession_id) {
16:   recovered <- recovr_sessions(file.path(state_folder, contextsession_
21:   cbind(data.frame(contextsession = contextsession_id), recovered)

```

Environment History Connections Build Git Tutorial

Data basics

learnr: ex-data-basics

Learn about the base data types in R. Explore R's data frames, and learn how to interact with data frames and their columns.

Start Tutorial ▶

Filter observations

learnr: ex-data-filter

Learn how to filter observations in a data frame. Use `filter()` to extract observations from a data frame, and use `&`, `|`, and `!` to write logical tests.

Start Tutorial ▶

Files Plots Packages Help Viewer

Zoom Export Publish

Scatter chart with size and color

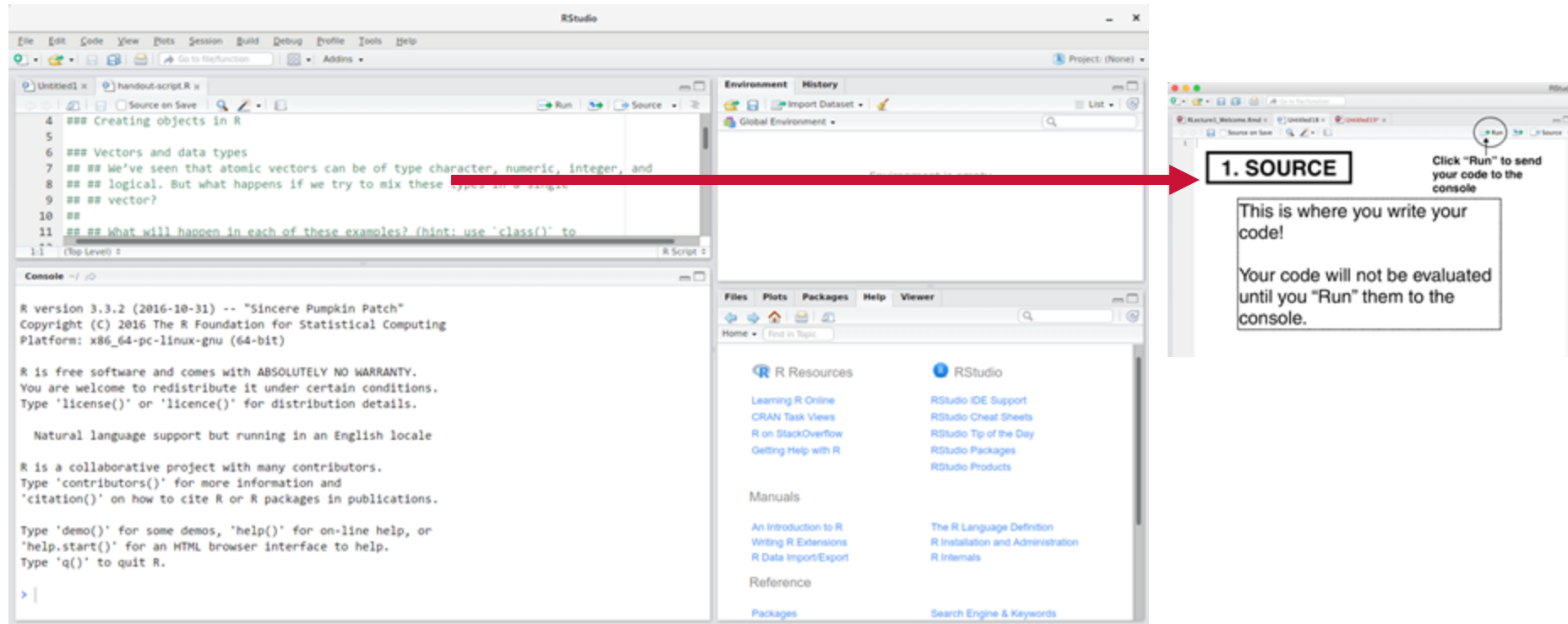


4 Key Components:

- Source pane (code editor)
- Console pane (execute R commands)
- Environment pane (view variables and objects)
- Plots pane (view charts and graphs)



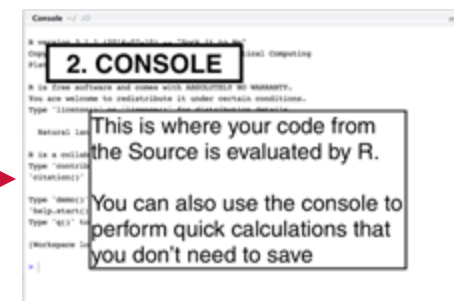
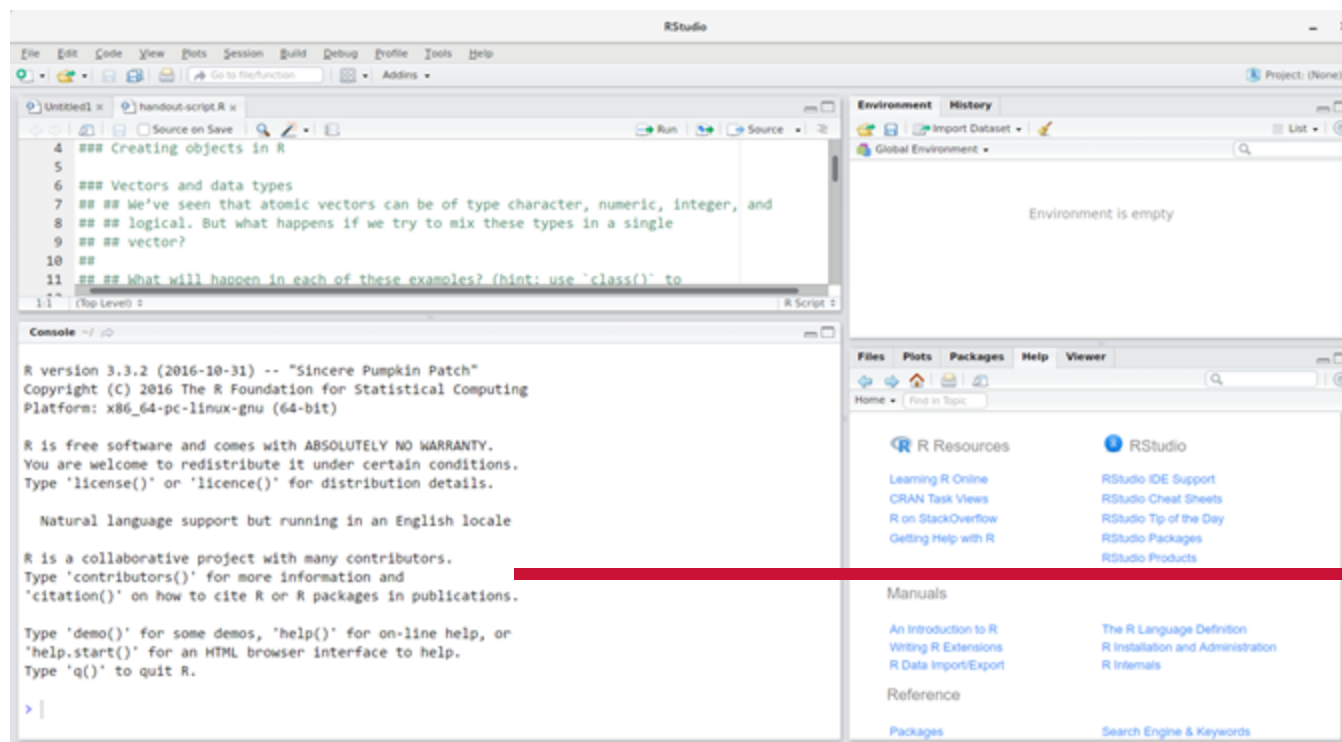
R Studio® Interface



Reference: <https://datacarpentry.org/R-ecology-lesson/00-before-we-start.html>



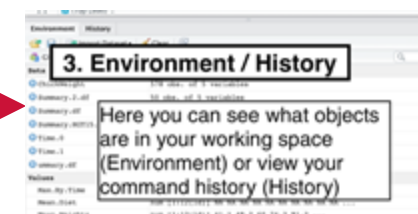
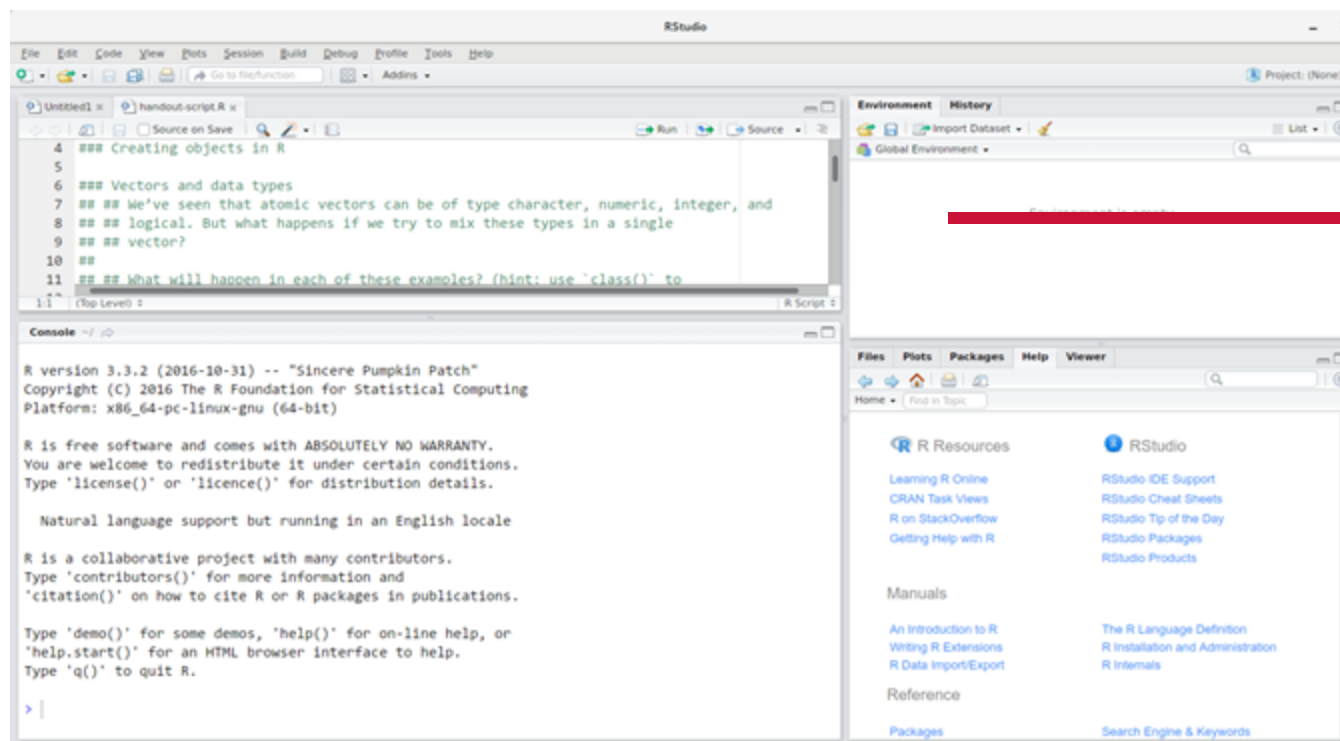
R Studio® Interface



Reference: <https://datacarpentry.org/R-ecology-lesson/00-before-we-start.html>



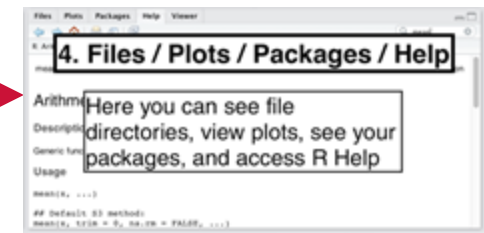
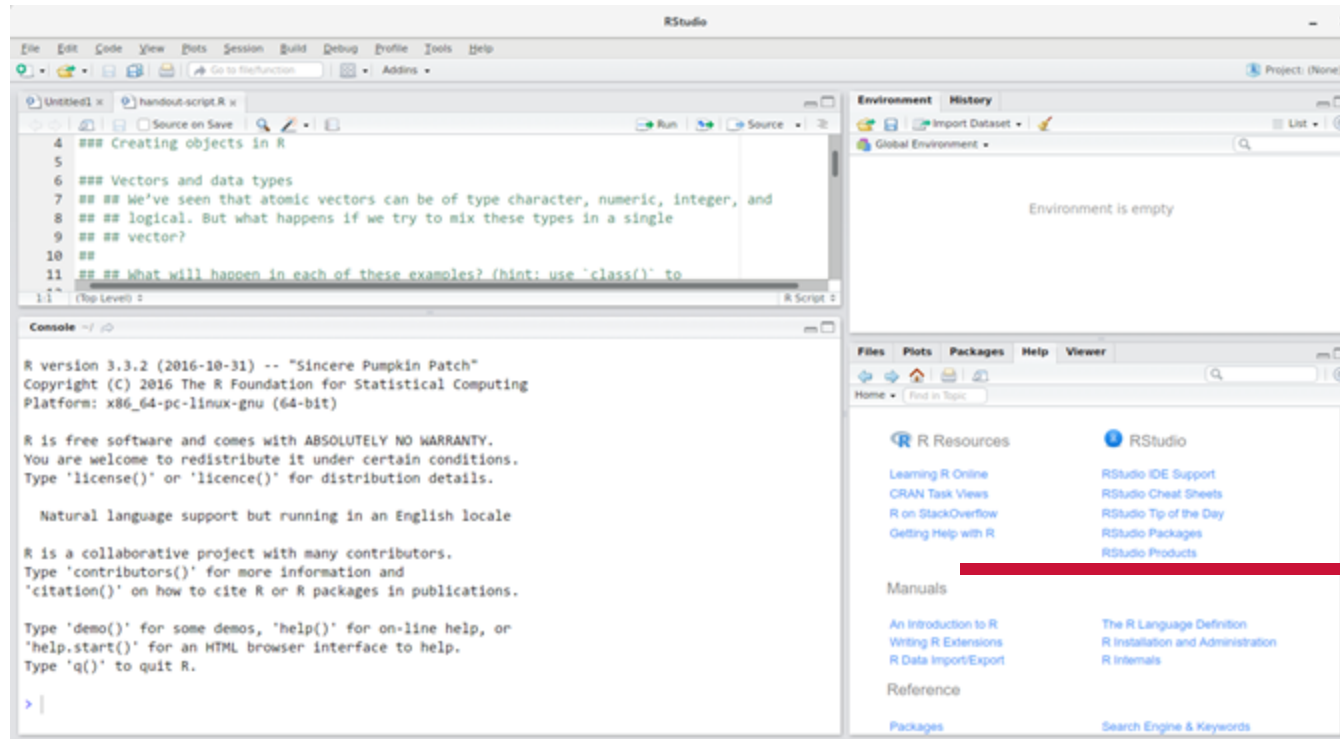
R Studio® Interface



Reference: <https://datacarpentry.org/R-ecology-lesson/00-before-we-start.html>



R Studio® Interface



Reference: <https://datacarpentry.org/R-ecology-lesson/00-before-we-start.html>





What is R Markdown?

Definition: R Markdown is a tool for creating dynamic and reproducible documents using R.

Key Features:

- Combine code, text, and output in one document.
- Output formats: HTML, PDF, Word, slides, etc.
- Reproducibility: Share your analysis with others.





Basic R Syntax

- Assignment: `x <- 10`
- Arithmetic: `x + 5`
- Functions: `mean(c(1, 2, 3))`
- Data structures: `vector`, `matrix`, `data.frame`

R Shortcuts	
Execute Code	Ctrl/Cmd + Enter
Clear Console	Ctrl/Cmd + L
Assignment Operator <-	Alt + -
Pipe %>%	Ctrl/Cmd + Shift + M
Insert Code Chunk in Markdown	Ctrl/Cmd + Alt + I

Essential R Commands

- `help()`, `?function_name` (get help) `?mean`
- `install.packages()`, `library()` (install/load packages)
- `summary()`, `str()` (explore data)
 - `install.packages('ggplot2')`
 - `library(ggplot2)`





Key Data Structures in R

- **Vectors:**

- One-dimensional data structure.
- Example: `v <- c(1, 2, 3)`.

- **Matrices:**

- Two-dimensional data structure.
- Example: `m <- matrix(1:9, nrow=3, ncol=3)`.

- **Data Frames:**

- Two-dimensional, like a spreadsheet.
- Example: `df <- data.frame(Name=c('Alice', 'Bob'), Age=c(25, 30))`.

What does `m[2,]` return?

- A) The second column
- B) The second row
- C) An error
- D) The value 2

Logical Operations and Conditions

- **Logical Operators:**

- `>`, `<`, `==`, `!=`, `&`, `|`.
- Example: `x <- 10; x > 5`.

- **Conditional Statements:**

- Example:

```
if (x > 5) {  
  print("x is greater than 5")  
} else {  
  print("x is not greater than 5")  
}
```





Loop Structures

- **For Loops:**

- Example:

```
for (i in 1:5) {  
  print(i)  
}
```

- **While Loops:**

- Example:

```
i <- 1  
while (i <= 5) {  
  print(i)  
  i <- i + 1  
}
```

Loops are great for processing data or running simulations!



Creating Functions

- **Syntax:**

- Example:

```
my_function <- function(a, b) {  
  return(a + b)  
}  
result <- my_function(3, 5)  
print(result)
```

- **Usage:** Encourages reusability and modularity.

Functions are a powerful way to organize your code and reduce redundancy!

Activity 1 - Break the Ice with R

Start RStudio and try running this simple command:

```
cat("Welcome to the world of R!")
```

Discussion Questions:

- How is `cat()` different from `print()`?
- What happens if you add a `\n` (newline character) inside the string?



Activity 2 - String Shenanigans

If you run the following code:

```
str <- "Graduate Students Rock!"  
cat(toupper(substring(str, 10, 18)))
```

What will it print?

- A) "STUDENTS"
- B) "GRADUATE"
- C) "ROCK!"
- D) "GRADUATE STUDENTS ROCK!"





Activity 3 – More Pop Quiz!

What will the following code print?

```
x <- TRUE  
y <- FALSE  
x & y
```

```
x <- factor(c("high", "low", "medium", "high", "medium"))  
levels(x)
```

```
x <- c(1, 2, 3, 4)  
x[5]
```

Activity 4 - Create Your First R Markdown

Prompt: Start a new R Markdown file and generate a simple report.

Steps:

1. Go to **File > New File > R Markdown**.
2. Add a title and author, then select the default template.
3. Add this code to the code chunk:

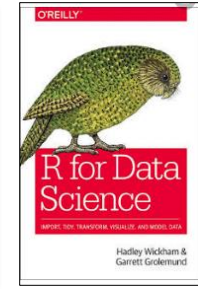
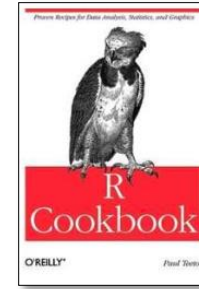
```
summary(cars)
```

4. Click **Knit** to generate an HTML document.





R Resources



RStudio Online Learning
R-bloggers How to learn R
Quick-R
R Swirl Package

- * <https://www.rstudio.com/resources/training/>
- * <https://www.r-bloggers.com/how-to-learn-r-2/>
- * <https://www.statmethods.net/r-tutorial/index.html>
- * <https://swirlstats.com/students.html>

Beginner

R-bloggers
R-documentation
The R-help Archives
R graph gallery
The R Graph Gallery
From Data to Viz

- * <https://www.r-bloggers.com/>
- * <https://www.rdocumentation.org/>
- * <https://stat.ethz.ch/mailman/listinfo/r-help>
- * <http://rgraphgallery.blogspot.com/>
- * <https://r-graph-gallery.com/>
- * <https://www.data-to-viz.com/>

Advanced

WSU R Working Group

- * <https://cereo.wsu.edu/r-group-resources/>