

# Lab #1: Introduction to R

Jacob Jameson

We expect you to watch the Module 1 material, [here](#) prior to lab.

## General Guidelines:

---

You will encounter a few functions we did not cover in the lecture video. This will give you some practice on how to use a new function for the first time. You can try following steps:

1. Start by typing `?new_function` in your Console to open up the help page
2. Read the help page of this new\_function. The description might be too technical for now. That's OK. Pay attention to the Usage and Arguments, especially the argument `x` or `x,y` (when two arguments are required)
3. At the bottom of the help page, there are a few examples. Run the first few lines to see how it works
4. Apply it in your lab questions

**It is highly likely that you will encounter error messages while doing this lab**  
**Here are a few steps that might help get you through it.**

1. Locate which line is causing this error first
2. Check if you may have a typo in the code. Sometimes another person can spot a typo faster than you.
3. If you enter the code without any typo, try googling the error message
4. Scroll through the top few links see if any of them helps
5. Try working on the next few questions while waiting for answers by TAs

## Warm-up

---

1. The most important warm up question: do you have R and RStudio installed?
2. Which of these allow you to pull up the documentation for a command in R?
  - a. `*`
  - b. `?`
  - c. `help()`
  - d. `documentation()`

In the code block below, run code that will pull up documentation for the function `paste0()`.

```
?paste0()
```

What does this function do?

3. What are the two ways that you can assign a value to a variable?

## Guess the Output: Algebra

---

4. Guess the output of the following code:

```
a <- 3  
b <- a^2 + 1
```

b

Now, run the code block to check your answer.

5. Guess the output of the following code:

```
a <- 10  
b <- 3 %% a  
  
b + 5
```

*Hint:* If you are not sure what `%%` does you can try running `?%%` to better understand.

5. Guess the output of the following code:

```
a <- c(1, 2, 3)  
b <- a^2 + 1  
  
b
```

## Guess the Output: Boolean

---

8. Guess the output of the following code:

```
25 >= 14
```

9. Guess the output of the following code:

```
10 != 100
```

10. Guess the output of the following code:

```
7%%5 == 2
```

11. Guess the output of the following code:

```
(5 > 7) & (7 * 7 == 49)
```

12. Ok, let's try some logic! Try to figure out each one before running the code!

a.

```
TRUE & FALSE
```

b.

```
FALSE & FALSE
```

c.

```
TRUE | (FALSE & TRUE)
```

d.

```
FALSE | (TRUE | FALSE)
```

e.

```
(TRUE & (TRUE | FALSE)) | FALSE
```

## Data Types

---

13. Run these lines to create these variables in your environment.

```
item_1 <- "Hi, my name is item 1!"  
item_2 <- 7
```

```
item_3 <- FALSE
```

What are the type (or mode) of each of these items?

*Hint:* If you are not sure, you could apply the `mode()` function to each item and check the output. If you are unsure about how to apply the `mode()` function, you can always run `?mode()`.

14. Guess the output of the following code:

```
(item_2 + 19 <= 25) == item_3
```

15. Do you remember earlier when you ran `?paste0()`? We are now going to try to use this function. In the code block below, initialize two variables that are of mode "character". The output when you apply `paste0()` to these variables should be "Hello, world!".

```
#v1 <-  
#v2 <-
```

Well done! You've learned how to work with R to perform simple variable assignment and operations!

**Want to improve this tutorial?** Report any suggestions/bugs/improvements on [here](#)! We're interested in learning from you how we can make this tutorial better.