

Applied Economic Forecasting

Tutorial on R Studio II

Spring 2020

In class, you can download this set of code here

Whenever possible, use comments! Anything following the symbol `#` will not be run in R. Comments are notes we leave ourselves so we know:

- Exactly who wrote the code (important in companies where many people may work on a project)
- The purpose of the code!
- What our thought process was at a particular line of code.

I promise that this will become useful when you come back to your code after an extended time. I cannot tell you the number of times I have had a moment of pure genius while coding and I spend hours on a different day trying to understand why I coded it like that or what I actually did.

For example, below is the type of comments that I always include in my programs

```
# Project: 'Tutorial on R Studio II'
# Author:  Shamar Stewart
# This program illustrates some basic programming philosophy
# and R operations
```

You can also understand the following code without even knowing what exactly each line of command does because I tell you what they are!

```
# Set seed number so that all the results based on
# random samples are reproducible.
set.seed(12345)
# Then create a normally distributed random variable, x,
# with 500 observations.
x <- rnorm(500)
# Notice '<-' is the universal assignment operator in R
# (I prefer this to '=')
```

Task 1 (-cont.-)

At the top of the previous script, write the following comments:

- 1 The project
- 2 The author
- 3 The purpose of this program

R Basics

Arithmetic

```
1 + 1 #add numbers
```

```
## [1] 2
```

```
8 - 4 #subtract them
```

```
## [1] 4
```

```
13/2 #divide
```

```
## [1] 6.5
```

```
4*pi #multiply
```

```
## [1] 12.56637
```

```
2^10 #exponentiate
```

```
## [1] 1024
```

Task 1 (-cont.-)

In one step:

- 1 Compute the difference between 2019 and the year you started at Virginia Tech
- 2 Divide this by the difference between 2019 and the year you were born.
- 3 Multiply this with 100 to get the percentage of your life you have spent at this university. Use brackets if you need them.
- 4 Assign this to a variable of your choosing

Logical Comparison

```
3 < 4
```

```
## [1] TRUE
```

```
3 > 4
```

```
## [1] FALSE
```

```
# Try 3 = 4
```

```
3 == 4
```

```
## [1] FALSE
```

```
3 != 4
```

```
## [1] TRUE
```

```
10 - 6 == 4
```

```
## [1] TRUE
```

```
# Notice the difference between single and double equal
```


Strings (text)

```
#R delimits strings with EITHER double or single quotes.  
# There is only a very minimal difference
```

```
message1 <- 'Let us get to coding!'  
message2 <- "Please get to coding!"  
print(message1)
```

```
## [1] "Let us get to coding!"
```

```
print(message2)
```

```
## [1] "Please get to coding!"
```

Variables

- variable are used to store values and results. Assignment to a variable happens from right to left - the value on the right side gets assigned to the name on the left side. You can use nearly anything as a variable name in R. The only rules are:
 - ① “.” and “_” are OK, but no other symbols.
 - ② Your variable name must not start with a number or _ (2squared and _one are illegal).
- A note for those of you who have programming experience: while R supports object-oriented programming, periods “.” do not have a special meaning in the language. For historical reasons, R programmers often use periods in place of underscores in variable names, but either works. Just be consistent to keep your code readable.
- R is case sensitive. Capitalization of variable names matter.

```
x <- 42
```

```
x/2
```

```
# [1] 21
```

```
# redefine x
```

```
x <- x + 3
```

```
x
```

```
# [1] 45
```

```
# if we assign something else to x, the old value is deleted
```

```
x <- "Hokies!"
```

```
x
```

```
# [1] "Hokies!"
```

```
foo <- 3
```

```
bar <- 5
```

```
foo.bar <- foo + bar
```

```
foo.bar
```

```
# [1] 8
```

Clearing the memory

To remove all variables in memory:

```
#    ls() # List of all variables in memory  
rm(list=ls())
```

- I usually place this at the begining of my R script (just after the document details).