Coding Assignment 2

Objectives of this assignment

- Gain familiarity in working in R notebooks.
- Review introductory material about R data types and R data objects.
- Gain introduction to errors and use knowledge from class to diagnose problem.

Some notes about running code chunks

- To run a single chunk, you can click on the green triangle pointing right within the chunk.
- To run all chunks above a given chunk, you press the gray triangle pointing down with the green dash underneath.
- Notice that at the top of the pane, in the right corner, there is also a drop down menu for different run options. It also shows keyboard shortcuts. Play around with these different options.
- When you have tested and are finished with all text and all code chunks, click "Knit" at the top of the pane. This will run all chunks and "compile" a PDF that includes text, code chunks and output.

Coding

1. The following code chunk assigns a vector to the object `run.times1`. Note that `run.times1` holds data on how long a person ran each of seven days in a week.

```
run.times1 <- c(47.25,46.50,50.25,44.75,45.00,31.50,48.50)
```

- What is the "data type" of `run.times1`?

numeric

- What is the "data structure" of `run.times1`?

vector

- Below, create a new code chunk. Name the code chunk `hw2chunk2`. In it, write a line of code that will display (print) the the object `run.times1`?

```
run.times1
```

```
## [1] 47.25 46.50 50.25 44.75 45.00 31.50 48.50
```

```
# or print(run.times1)
```

2. The object `run.times.by.week` shows data collected on someone's run times over two weeks.

```
run.times2 <- c(46.50,50.25,0,30.75,27.00,48.25,56.00)
run.times.by.week <- cbind(run.times1,run.times2)
run.times.by.week
```

```
##      run.times1 run.times2
## [1,]      47.25      46.50
## [2,]      46.50      50.25
## [3,]      50.25       0.00
## [4,]      44.75      30.75
## [5,]      45.00      27.00
## [6,]      31.50      48.25
## [7,]      48.50      56.00
```

- What are two observations that tell a story about the data?

[ENTER YOUR ANSWER HERE, 2 POINTS FOR COMPLETION, NO WRONG ANSWERS]

- What are two analyses you do with this run time data when you know how to do it in R?

[ENTER YOUR ANSWER HERE, 2 POINTS FOR COMPLETION, NO WRONG ANSWERS]

- In the following code chunk, write a line of code to show the number of rows and columns in the matrix `run.times.by.week`. Also, add a comment to the code chunk, saying whatever you think makes sense. [2 POINTS]

```
dim(run.times.by.week)
```

```
## [1] 7 2
```

3. Note that when we use double equal signs (==) in R, we are evaluating whether or not the value to the left of == is equivalent to the value to the right. For example:

```
x <- 4
y <- x==4
```

- In the next code chunk, write 2 lines of code to (1) print `y`; and (2) show the data type of `y`. [2 POINTS]

```
y
```

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

```
class(y)
```

```
## [1] "logical"
```

- What is another word for the data type of `y`? That is, what is a word that is not returned by R but that describes the type of data held by `y`?

Boolean or dummy

4. Below, create a new code chunk and name it `hw2chunk7`. In the chunk, create two vectors (your choice of length). Then "column bind" the two vectors in order to create a matrix. Print the matrix and its dimensions.

```
a <- c(1,2,3)
b <- c(4,5,6)
my.matrix <- cbind(a,b)
my.matrix
```

```
##      a b
## [1,] 1 4
## [2,] 2 5
## [3,] 3 6
```

```
dim(my.matrix)
```

```
## [1] 3 2
```

5. When you run the next code chunk, you will get an error.

```
#my.object <- dog
my.object <- "dog"
```

- Explain the error and make a correction in the code chunk.

Character data requires quotation marks surrounding it.

6. When you run the next code chunk, you will get an error.

```
#my.var <- (1,2,3)
my.var <- c(1,2,3)
```

- Explain the error and make a correction in the code chunk.

Creating a vector requires c().

Extra Credit

1. Run the following code chunk.

```
my.vector <- c(4,3,"tree")
my.vector
```

```
## [1] "4"    "3"    "tree"
```

- What do you observe in the print of `my.vector`? Explain why you observe what you do.

All entries in a vector must have the same data type. Rather than returning an error when a vector was created with numeric and character data, R converted the numeric data to character.

2. In the code chunk below, create an object called `sunday.runs` and assign to the data that is in the first row of `run.times.by.week` (i.e. separate out the first row of the matrix in `sunday.runs`. [2 POINTS]

```
sunday.runs <- run.times.by.week[1,]
sunday.runs
```

```
## run.times1 run.times2
##      47.25      46.50
```

ASSIGNMENT IS WORTH 20 POINTS (24 POINTS POSSIBLE WITH EXTRA CREDIT)