Assignment 2: Coding Basics

Elsie Liu

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on coding basics.

Basics Day 1

```
#1. Generate a sequence of numbers from 1 to 100, increasing by fours. Name this sequence as "seq_1_4_1
seq_1_4_100 <- seq(1,100,4)

#2. Calculate the mean and median of the sequence.
mean(seq_1_4_100)
```

```
## [1] 49
```

```
median(seq_1_4_100)
```

```
## [1] 49
```

```
#3. determine whether the mean is greater than the median
mean(seq_1_4_100) > median(seq_1_4_100)
```

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

Basics Day 2

```
#5. Create a series of vectors, each with four components, consisting of
#(a) names of students,
#(b) test scores out of a total 100 points
#(c) whether or not they have passed the test (TRUE or FALSE) with a passing grade of 50.
stdname <- c("Tom", "Todd", "May", "Merry")
scr <- c(48, 96, 96, 48)
ispass <- scr>50

#6. Label each vector with a comment on what type of vector it is.
class(stdname)
```

```
## [1] "character"
```

```
class(scr)
```

```
## [1] "numeric"
```

```
class(ispass)
```

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

```
#7. Combine each of the vectors into a data frame. Assign the data frame an informative name.
test.result <- data.frame(stdname,scr,ispass)
```

```r
#8. Label the columns of your data frame with informative titles.
colnames(test.result) <- c("Name", "Score", "Pass or Not")
```

9. QUESTION: How is this data frame different from a matrix?

   Answer: The elements in a matrix are in the same format (e.g. numeric), while one data frame could hold varoius types of data.

```r
#10. Create a function with an if/else statement. Your function should determine whether a test score i
is.pass.50 <- function(x) {
  for (i in 1:length(x)){
  if (x[i]>50) {
    print("TRUE")
  }
  else {
    print("FALUSE")
  }}
}

is.pass.50.ifelse <-function(x) {
  ifelse(x>50,"TURE","FALSE")
}

#11. Apply your function to the vector with test scores that you created in number 5.
is.pass.50(scr)
```

```
## [1] "FALUSE"
## [1] "TRUE"
## [1] "TRUE"
## [1] "FALUSE"
```

```r
is.pass.50.ifelse(scr)
```

```
## [1] "FALSE" "TURE"  "TURE"  "FALSE"
```

12. QUESTION: Which option of `if` and `else` vs. `ifelse` worked?

    Answer: Both of the function worked. "ifelse" is more convenient as it could handle the whole vector instead of only one element.