Week 7

Rachael Valdez

10/4/2017

test1 = "The current year is 2016"  
#exercise 1  
my\_pattern = "[xqb]"  
my\_pattern = "[0-9]"  
grepl(pattern = my\_pattern, test1)

## [1] TRUE

#exercise 2  
gregexpr(my\_pattern, test1)

## [[1]]  
## [1] 21 22 23 24  
## attr(,"match.length")  
## [1] 1 1 1 1  
## attr(,"useBytes")  
## [1] TRUE

#output = gregexpr(my\_pattern, test1) #output puts value on top right dashboard into a list  
temp = gregexpr(my\_pattern, test1)  
string\_position = temp[[1]]  
  
gregexpr("[[:blank:]]", test1)

## [[1]]  
## [1] 4 12 17 20  
## attr(,"match.length")  
## [1] 1 1 1 1  
## attr(,"useBytes")  
## [1] TRUE

#exercise3  
grepl(pattern = "[[:upper:][:digit:]]", test1)

## [1] TRUE

#exercie4  
temp = regexpr(pattern = "[[:blank:]]", test1)  
first\_space = temp[[1]][1] #first slot and first entry  
first\_space

## [1] 4

#exercise5  
my\_pattern = "[[:lower:]].[[:digit:]]"  
#my\_pattern = "[[:lower:].[:digit:]]" #does this do the same as line above? NO  
grepl(my\_pattern, test1)

## [1] TRUE

#exercise6  
loc = gregexpr(my\_pattern, test1)[[1]][1] #this accomplishes same as below two lines of code, shortcut  
#temp = gregexpr(my\_pattern, test1)  
#loc = temp[[1]] [1]  
  
#exercise7  
my\_pattern = "\\s[[:lower:]][[:lower:]]\\s"  
loc = gregexpr(my\_pattern, test1)[[1]][1]  
#exercise8  
text2 = sub(my\_pattern, replacement = " is not ", test1)  
  
my\_pattern = "\\s[[:lower:]]{2}\\s" #same as above  
gregexpr(my\_pattern, test1)[[1]][1]

## [1] 17

#exercise9  
my\_pattern = "\\d{4}"  
temp = gregexpr(my\_pattern, text2)  
string\_pos4 = temp[[1]][1]  
  
#exercise 10  
  
substr(text2, start = string\_pos4, stop = string\_pos4 + 1)

## [1] "20"