Data607-wk3

Joe Rovalino

12/8/2019

R Markdown

R Markdown This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
raw.data <- "555-1239Moe Szyslak(636) 555-0113Burns, C. Montgomery555 -6542Rev. Timothy Lovejoy555 8904
raw.data
```

```
## [1] "555-1239Moe Szyslak(636) 555-0113Burns, C. Montgomery555 -6542Rev. Timothy Lovejoy555 8904Ned Fl
```

##Library

```
library(stringr)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v forcats 0.4.0
## v readr   1.3.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Problem #3 Copy the introductory example. The vector name stores the extracted names. R> name [1] “Moe Szyslak” [4] “Ned Flanders” “Burns, C. Montgomery” “Rev. Timothy Lovejoy” “Simpson, Homer” “Dr. Julius Hibbert”

a) Use the tools of this chapter to rearrange the vector so that all elements conform to the standard first_name last_name .

b) Construct a logical vector indicating whether a character has a title (i.e., Rev. and Dr.).

c) Construct a logical vector indicating whether a character has a second name.

Problem 3

```
data <-"555-1239Moe Szyslak(636) 555-0113Burns, C. Montgomery555-6542Rev. Timothy Lovejoy555 8904Ned Fla

#remove phone numbers from raw data page 206 of textbook
#names https://stackoverflow.com/questions/33826650/last-name-first-name-to-first-name-last-name

clnnames <- unlist(str_extract_all(raw.data, "[[:alpha:]., ]{2,}"))
#clnnames
splitname <- str_split(clnnames, ", ", simplify = TRUE)
#splitname
```

```
firstlast <- str_c(splitname[,2]," ",splitname[,1])
firstlast
```

```
## [1] " Moe Szyslak"          "C. Montgomery Burns"   " Rev. Timothy Lovejoy"
## [4] " Ned Flanders"         "Homer Simpson"         " Dr. Julius Hibbert"
```

```
frntname<- str_detect(firstlast, "[[:alpha:]]{2,}\\.")
frntname
```

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

```
middle <- str_detect(firstlast, "[A-Z]{1}\\.")
middle
```

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

Problem 4

4. Describe the types of strings that conform to the following regular expressions and construct an example that is matched by the regular expression

(a) [0-9]+ \ $ An expression that containt a number followed by a $

```
exp1 <- c("goog1e$", "google1$")
ans4a <- str_detect(exp1, "[0-9]+\\$")
ans4a
```

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

(b) \ b[a-z]{1,4} \ b an expression with an empty lowercase string then followed by alphas 1 through 4 characters in length and followed by expty string

```
exp2 <- c(" tomz ", " Tomz ","John", "john","mik","drinkmilk", "Josi " )
ans4b <- str_detect(exp2, "\\b[a-z]{1,4}\\b")
ans4b
```

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

(c) .*? \ .txt$ an epxression that ends in .txt

```
exp3 <- c("test .txt", "test.txt", "test.txta","test.xml", "test.json" )
ans4c <- str_detect(exp3, ".*?\\.txt$")
ans4c
```

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

(d) \ d{2}/ \ d{2}/ \ d{4} An expresion that is date with 2 digit day and 2 digit month and 4 digit year.

```
exp4 <- c("31/12/2019", "12/31/2019","1/01/2019" )
ans4d <- str_detect(exp4, "\\d{2}/\\d{2}/\\d{4}")
ans4d
```

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

(e) <(.+?)>.+?</ \ 1> This epxresssion is fro HTML tagging

```
exp5 <- c("<br>HTML is fun</br>", "<br> Happy Holidays!<br>" )
ans4e <- str_detect(exp5, "<(.+?)>.+?</\\1>")
ans4e
```

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