

607 Week 3 Assignment

Femi Adesanya

February 17, 2018

```
setwd("C:\\Users\\26291\\Documents")
```

3. Copy the introductory example. The vector name stores the extracted names.

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

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

```
library(stringr)
name_data <- "555-1239Moe Szyslak(636) 555-0113Burns, C. Montgomery555-6542Rev. Timothy Lovejoy555 89041
name <- unlist(str_extract_all(name_data, "[[:alpha:]]+", )$2,))
name
```

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

3a.1 To convert list to dataframe and rename.

```
#convert list to dataframe
namesdf<-do.call(rbind, lapply(name, data.frame, stringsAsFactors=FALSE))
#rename column
namesdf$names<-namesdf$X..i.
```

3a.3 rearrange name

```
namesdf$stdFormatNames<-ifelse(grepl( " ",namesdf$names),paste(word(namesdf$names,-1),word(namesdf$names,
```

3a.4 commas and prefixes exclusion

```
namesdf$stdFormatNames<-gsub("Rev.|Dr.|,","", namesdf$stdFormatNames)
namesdf$stdFormatNames
```

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

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

```
namesdf$hasTitle<-str_detect(namesdf$names, "Rev.|Dr.")
namesdf$names
```

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

```
namesdf[,c("names", "hasTitle")]
```

```
##           names hasTitle
## 1      Moe Szyslak  FALSE
## 2 Burns, C. Montgomery  FALSE
```

```
## 3 Rev. Timothy Lovejoy      TRUE
## 4      Ned Flanders         FALSE
## 5      Simpson, Homer       FALSE
## 6      Dr. Julius Hibbert    TRUE
```

3.C Construct a logical vector indicating whether a character has a second name

```
grepl( " ",str_trim(namesdf$stdFormatNames))
```

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

4.

4.a [0-9]+\

```
library(stringr)
schema="[0-9]+\\"
eg=c("3533$", "xy12$ef", "2$")
str_detect(eg,schema)
```

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

4.b \b[a-z]{1,4}\b

```
schema="\b[a-z]{1,4}\b"
eg=c("a","cb","xyz","wxyz","456 abcd 12v")
str_detect(eg,schema)
```

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

4.c .*?\.txt\$

```
schema=".*?\.txt$"
eg=c(".txt", "xyz.txt", "123cde.txt", "c$d#2.txt")
str_detect(eg,schema)
```

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

4.d \d{2}/\d{2}/\d{4}

```
schema = "\\d{2}/\\d{2}/\\d{4}"
eg=c("05/14/3025", "02/01/1980 Son!", "!! 11/02/1979 !!")
str_detect(eg,schema)
```

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

4.e <(.*?)>.+?</\1>

```
schema="<(.*?)>.+?</\1>"
eg=c("<tag>Text</tag>", "<Font size=4,color=blue>Blue Text</Font size=4,color=blue>")
str_detect(eg,schema)
```

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

9. Extra Credit

```
cipher <- "clcopCowlzmstc0d87wnkig70vdicpNuggvhryn92Gjuwcz8hqrfrRxs5Aj5dwpn0Tanwo
Uwisdij7Lj8kpf03AT5Idr3coc0bt7yczjat0aootj55t3Nj3ne6c4Sfek.r1w1Ywwojig0
d6vrfUrbz2.2bkAnbhvgv4R9i05zEcrop.wAgnb.SqoU65fPa1otfb7wEm24k6t3sR9zqe5
fy89n6Nd5t9kc4fE905gmc4Rgxo5nhDk!gr"

decipher <- unlist(str_extract_all(cipher, "[[:upper:]]{1,}"))
decipher <- str_replace_all(paste(decipher, collapse = ''), "[.]", " ")
decipher

## [1] "CONGRATULATIONS YOU ARE A SUPERNERD"
```