DATA607_Assignment3

Harpreet Shoker

Environment Setup

Extracting data from example from book

```
raw.data <- "555-1239Moe Szyslak(636) 555-0113Burns, C. Montgomery555
-6542Rev. Timothy Lovejoy555 8904Ned Flanders636-555-3226Simpson,
Homer5553642Dr. Julius Hibbert"
name <- unlist(str_extract_all(raw.data, "[[:alpha:]., ]{2,}"))
name
```

```
## [1] "Moe Szyslak"          "Burns, C. Montgomery" "Rev. Timothy Lovejoy"
## [4] "Ned Flanders"         "Simpson,"             "Homer"
## [7] "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

```
names_new <- str_replace(name, pattern = "Rev. |Dr. |Mr. |Mrs. |Ms ", replacement = ""); #replacing the
firstname = sapply(strsplit(names_new, ' '), function(x) x[1])
lastname = sapply(strsplit(names_new, ' '), function(x) x[length(x)])
full_names <- cbind(firstname,lastname)
full_names
```

```
##      firstname  lastname
## [1,] "Moe"      "Szyslak"
## [2,] "Burns,"   "Montgomery"
## [3,] "Timothy"  "Lovejoy"
## [4,] "Ned"      "Flanders"
## [5,] "Simpson," "Simpson,"
## [6,] "Homer"    "Homer"
## [7,] "Julius"   "Hibbert"
```

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

```
title <- str_detect(name, "^.{3}\\.|^.{2}\\.")
check_title <- cbind(name, title = title)
check_title
```

```
##      name                   title
## [1,] "Moe Szyslak"          "FALSE"
## [2,] "Burns, C. Montgomery" "FALSE"
## [3,] "Rev. Timothy Lovejoy" "TRUE"
## [4,] "Ned Flanders"         "FALSE"
## [5,] "Simpson,"             "FALSE"
## [6,] "Homer"                "FALSE"
## [7,] "Dr. Julius Hibbert"   "TRUE"
```

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

```
second_name <- ifelse(str_detect(name,"[A-Z]\\.") == TRUE, "YES", "NO")
mid_name <- cbind(name,second_name)
mid_name
```

```
##      name                   second_name
## [1,] "Moe Szyslak"          "NO"
## [2,] "Burns, C. Montgomery" "YES"
## [3,] "Rev. Timothy Lovejoy" "NO"
## [4,] "Ned Flanders"         "NO"
## [5,] "Simpson,"             "NO"
## [6,] "Homer"                "NO"
## [7,] "Dr. Julius Hibbert"   "NO"
```

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]+\$

This expression matches any digit or digits between 0 to 9 ending with $ sign here is the example

```
data1 <- c("Example for the above is expression is 0$","46758$","65472999")
unlist(str_extract_all(data1, "[0-9]+\\$"))
```

```
## [1] "0$"     "46758$"
```

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

This expression matches 1 to 4 character lower case words in input string

```
data2 <- c("Example for the above is expression is one four","1","Two","one")
unlist(str_extract_all(data2, "\\b[a-z]{1,4}\\b"))
```

```
## [1] "for"  "the"  "is"   "is"   "one"  "four" "one"
```

(c) .*?\.txt$

This expression matches any string ending with .txt including the string that equals .txt. Can be used to find text files in a file listing.

```
data2 <- c("Example for the above is expression is one four.txt","1","Two","one.txt")
unlist(str_extract_all(data2, ".*?\\.txt$"))
```

```
## [1] "Example for the above is expression is one four.txt"
## [2] "one.txt"
```

(d) \d{2}/\d{2}/\d{4}

This expression matches 2 digits followed by a / followed by another 2 digits followed by a / and finally followed by four digits We can say date format

```
data2 <- c("18/07/1984","1","Two","one.txt")
unlist(str_extract_all(data2, "\\d{2}/\\d{2}/\\d{4}"))
```

```
## [1] "18/07/1984"
```

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

This expression matches a pair of openning and closing tags - one or more characters enclosed with < > brackets /tags .

```
data2 <- c("18/07/1984","1","Two","one.txt","<a>Harpreet<a>test</a></a>")
unlist(str_extract_all(data2, "<(.+?)>.+?</\\1>"))
```

```
## [1] "<a>Harpreet<a>test</a>"
```

Extra credit Question — 9. The following code hides a secret message. Crack it with R and regular expressions.Hint: Some of the characters are more revealing than others! The code snippet is alsoavailable in the materials at www.r-datacollection.com.clcopCow1zmstc0d87wnkig7OvdicpNuggvhryn92

```
secret_message <- c("clcopCow1zmstc0d87wnkig7OvdicpNuggvhryn92Gjuwczi8hqrfpRxs5Aj5dwpn0TanwoUwisdij7Lj8
paste(unlist(str_extract_all(secret_message, "[:upper:]|[:punct:]")))
```

```
##  [1] "C" "O" "N" "G" "R" "A" "T" "U" "L" "A" "T" "I" "O" "N" "S" "." "Y"
## [18] "O" "U" "." "A" "R" "E" "." "A" "." "S" "U" "P" "E" "R" "N" "E" "R"
## [35] "D" "!"
```