607_01_Week3_Assignment_Regular_Expressions

Shyam BV

September 13, 2016

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

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

require(stringr)
```

```
## Loading required package: stringr
```

```
name <- unlist(str_extract_all(raw.data, "[[:alpha:]., ]{3,}"))
name
```

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

```
name_test <- name


#Store and show only the correct names in a vector

correct_names <- unlist(str_extract_all(name_test,"[A-Z][a-z]+\\.? [A-Z][a-z]+( [A-Z][a-z]+)?"))

#Store the incorrect names in a vector

lastname <- unlist(str_extract_all(name_test,"(.+), .+"))

#Store the correct firstnames in another vector

firstlastname_1 <-  unlist(str_extract_all(lastname,"(, .+)"))

#Store the correct lastnames in another vector

lastname_1 <- unlist(str_extract_all(lastname,"[A-Z](.+),"))

#Create a dataframe by combining firstname and lastname

final_fullname <- data.frame(firstlastname_1,lastname_1,fullname = paste0(firstlastname_1,' ' ,lastname

#Replace the Comma(,) and string pattern

final_fullname$fullname <- str_replace(final_fullname$fullname, pattern = "((, )|,)?,", replacement = "

#Replace the Comma(,) and string pattern

final_fullname$fullname <- str_replace(final_fullname$fullname, pattern = ",", replacement = "")

#Final vector with corrected string
```

```
correctednames <-   c(correct_names,final_fullname$fullname)

#Remove the unwanted vector

remove(firstlastname_1,lastname,lastname_1,name_test,final_fullname)

#Final vector with corrected names
correctednames
```

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

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

```
#1. [0-9]+\\$

str_extract_all("7340$","[0-9]+\\$")
```

```
## [[1]]
## [1] "7340$"
```

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

str_extract_all(" shya ","\\b[a-z]{1,4}\\b")
```

```
## [[1]]
## [1] "shya"
```

```
#3. .*?\\.txt$

str_extract_all("shyam.txt", ".*?\\.txt$")
```

```
## [[1]]
## [1] "shyam.txt"
```

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


str_extract_all("73/40/7340", "\\d{2}/\\d{2}/\\d{4}")
```

```
## [[1]]
## [1] "73/40/7340"
```

```
#5. <(.+?)>.+?</\\1>

str_extract_all("<HTML>h</HTML>", "<(.+?)>.+?</\\1>")
```

```
## [[1]]
## [1] "<HTML>h</HTML>"
```