STRING MANIPULATION WITH STRINGR

Introducing stringr

stringr

- Powerful but easy to learn
- Built on `stringi`
- Concise and consistent
  - All functions start with `str_`
  - All functions take a vector of strings as the first argument

str_c()

```
> my_toppings
[1] "green peppers" "olives"         "onions"
```

str_c()

```
> my_toppings
[1] "green peppers" "olives"         "onions"

> paste(c("", "", "and "), my_toppings, sep = "")
[1] "green peppers" "olives"         "and onions"
```

str_c()

```
> my_toppings
[1] "green peppers" "olives"        "onions"

> paste(c("", "", "and "), my_toppings, sep = "")
[1] "green peppers" "olives"        "and onions"


> library(stringr)

> str_c(c("", "", "and "), my_toppings)
[1] "green peppers" "olives"        "and onions"
```

str_c()

```
> my_toppings
[1] "green peppers" "olives"        "onions"

> paste(c("", "", "and "), my_toppings, sep = "")
[1] "green peppers" "olives"        "and onions"


> library(stringr)

> str_c(c("", "", "and "), my_toppings)
[1] "green peppers" "olives"        "and onions"
```

str_length(), str_sub()

Babynames

- USA from 1880 to 2014
- You'll use 2014 only

```
> library(babynames)
> head(babynames)
  year sex      name    n       prop
1 1880   F      Mary 7065 0.07238359
2 1880   F      Anna 2604 0.02667896
3 1880   F      Emma 2003 0.02052149
4 1880   F Elizabeth 1939 0.01986579
5 1880   F    Minnie 1746 0.01788843
6 1880   F  Margaret 1578 0.01616720
```

STRING MANIPULATION WITH STRINGR

Let’s practice!

STRING MANIPULATION WITH STRINGR

Hunting for matches

stringr functions that look for matches

- All take a `pattern` argument
  - `str_detect()`
  - `str_subset()`
  - `str_count()`

Finding matches

```
> pizzas <- c("cheese", "pepperoni", "sausage and green peppers")

> str_detect(string = pizzas,  pattern = "pepper")
[1] FALSE  TRUE  TRUE

> str_detect(string = pizzas, pattern = fixed("pepper"))
[1] FALSE  TRUE  TRUE

> str_subset(string = pizzas, pattern = fixed("pepper"))
[1] "pepperoni"                 "sausage and green peppers"

> str_count(string = pizzas, pattern = fixed("pepper"))
[1] 0 1 1
```

STRING MANIPULATION WITH STRINGR

Let’s practice!

STRING MANIPULATION WITH STRINGR

Splitting strings

str_split()

"Tom & Jerry" **& not of interest**

```
> str_split(string = "Tom & Jerry", pattern = " & ")
[[1]]
[1] "Tom"   "Jerry"

> str_split("Alvin & Simon & Theodore", pattern = " & ")
[[1]]
[1] "Alvin"    "Simon"    "Theodore"

> str_split("Alvin & Simon & Theodore", pattern = " & ", n = 2)
[[1]]
[1] "Alvin"            "Simon & Theodore"
```

str_split() returns a list

```
> chars <- c("Tom & Jerry",
             "Alvin & Simon & Theodore")

> str_split(chars, pattern = " & ")
[[1]]
[1] "Tom"   "Jerry"

[[2]]
[1] "Alvin"    "Simon"    "Theodore"
```

str_split() can return a matrix

```
> chars <- c("Tom & Jerry",
             "Alvin & Simon & Theodore")

> str_split(chars, pattern = " & ", simplify = TRUE)
     [,1]    [,2]    [,3]
[1,] "Tom"   "Jerry" ""
[2,] "Alvin" "Simon" "Theodore"
```

Combing with `lapply()`

```
> chars <- c("Tom & Jerry",
             "Alvin & Simon & Theodore")

> split_chars <- str_split(chars, pattern = " & ")
> split_chars
[[1]]
[1] "Tom"   "Jerry"

[[2]]
[1] "Alvin"    "Simon"    "Theodore"

> lapply(split_chars, length)
[[1]]
[1] 2

[[2]]
[1] 3
```

STRING MANIPULATION WITH STRINGR

Let’s practice!

STRING MANIPULATION WITH STRINGR

Replacing matches in strings

str_replace()

```
> str_replace("Tom & Jerry",
              pattern = "&",
              replacement = "and")
[1] "Tom and Jerry"

> str_replace("Alvin & Simon & Theodore",
              pattern = "&",
              replacement = "and")
[1] "Alvin and Simon & Theodore"

> str_replace_all("Alvin & Simon & Theodore",
              pattern = "&",
              replacement = "and")
[1] "Alvin and Simon and Theodore"
```

str_replace() with vectors

```
> chars <- c("Tom & Jerry",
             "Alvin & Simon & Theodore")

> str_replace_all(chars,
                  pattern = "&",
                  replacement = "and")
[1] "Tom and Jerry"
[2] "Alvin and Simon and Theodore"
```

STRING MANIPULATION WITH STRINGR

Let’s practice!