

Project 21

Name:Elizabeth McGuckin

This project is supposed to be an easy modification of the project example, since it is almost time for Spring Break!

Note: Do not worry if some of the results have extra spaces. We can deal with that later!

1. Modify the NPS example to extract the city location for every National Park.

```
library(RCurl)

## Loading required package: bitops

library(XML)
myparks <- xpathSApply(htmlParse(getURL("https://www.nps.gov/findapark/index.htm")), "//*[div/select/op
mygoodparks <- myparks[(myparks != "cbpo") & (myparks != "foca")]
mycityextractor <- function(x) {xpathSApply(htmlParse(getURL( paste0("https://www.nps.gov/", x, "/index
                                "//*[span[@itemprop='addressLocality']", xmlValue))}
myresults <- sapply( mygoodparks, mycityextractor )
head(myresults)

## $abli
## [1] "Hodgenville"
##
## $acad
## [1] "Bar Harbor"
##
## $adam
## [1] "Quincy"
##
## $afam
## [1] "Washington"
##
## $afbg
## [1] " New York"
##
## $agfo
## [1] "Harrison"
```

Explain solution: To find the city location for every national park, I first had to find all the national parks. Then I found all the cities. To do this I used mycityextractor.

2. Same question, for the state location for every National Park.

```
mystateextractor <- function(x) {xpathSApply(htmlParse(getURL( paste0("https://www.nps.gov/", x, "/index
                                "//*[span[@itemprop='addressRegion']", xmlValue))}
myresults <- sapply( mygoodparks, mystateextractor )
head(myresults)

## $abli
## [1] "KY"
##
## $acad
## [1] "ME"
##
```

```
## $adam
## [1] "MA"
##
## $afam
## [1] "DC"
##
## $afbg
## [1] "NY"
##
## $agfo
## [1] "NE"
```

Explain solution: To find the state location, I found all of the parks and then all their corresponding states. I used mystateextractor to do this.

3. Same question, for the zip code for every National Park.

```
myzipcodeextractor <- function(x) {xpathSApply(htmlParse(getURL( paste0("https://www.nps.gov/", x, "/index.html"),
                                                                    "/*/span[@itemprop='postalCode']", xmlValue))}
myresults <- sapply( mygoodparks, myzipcodeextractor )
head(myresults)
```

```
## $abli
## [1] "42748"
##
## $acad
## [1] "04609"
##
## $adam
## [1] "02169"
##
## $afam
## [1] "20024"
##
## $afbg
## [1] "10005"
##
## $agfo
## [1] "69346"
```

Explain solution: To find the zip code for all of the parks I first found the parks and then used the code 'myzipcodeextractor' to find the zip codes for each individual park.