The purpose of this section is to showcase the ability to scrape and process web data using R. This tutorial draws heavily from a post on a great blog by Pascal Mickelson and Scott Chamberlain, two biologists and experienced R users. Suppose we want to find the number of available environmental economics journals. There are too many. Definitely. But suppose we want to find out just how many. To do this, we can visit crossref.org, which is a citation-linking network with a list of all journals and their Digital Object Identifiers (DOIs). We will query the list from within R and then parse the returned content to list journals with certain attributes. For this, we'll need to load the following libraries:

```
library(XML)
library(RCurl)
library(stringr)
```

The next step is to repeatedly query crossref.org for journal titles. Try to copy and paste the base URL address (`baseurl`) into your browser: `http://oai.crossref.org/OAIHandler?verb=ListSets`. The result is a long XML form. The function `getURL` in the following code pulls this response into R as a string, and the outer functions `xmlParse` and `xmlToList` convert the output into an R data structure. There are too many entries to fit into a single query, so the `while` loop continues to query until there are no more results. The final results are stored in `nameslist`.

```
token <- "characters"
nameslist <- list()
options(show.error.messages = FALSE)
while (is.character(token) == TRUE) {
    baseurl <- "http://oai.crossref.org/OAIHandler?verb=ListSets"
    if (token == "characters") {
      tok2 <- NULL
    } else {
      tok2 <- paste("&resumptionToken=", token, sep = "")
    }
    query <- paste(baseurl, tok2, sep = "")
    crsets <- xmlToList(xmlParse(getURL(query)))
    names <- as.character(sapply(crsets[[4]], function(x) x[["setName"]]))
    nameslist[[token]] <- names
    if (class(try(crsets[[2]]$.attrs[["resumptionToken"]])) == "try-error") {
        stop("no more data")
    }
    else {
      token <- crsets[[2]]$.attrs[["resumptionToken"]]
    }
}
```

How many journal titles are collected by this query? We first concatenate the results into a single list, and then find the total length:

```
allnames <- do.call(c, nameslist)
length(allnames)
```

```
 Error in do.call(c, nameslist) : object 'nameslist' not found
 Error: object 'allnames' not found
```

Now, suppose that we are looking for just those journals with *economic* in the title. We rely on regular expressions, a common way to parse strings, from within R. The following code snippet detects strings with some variant of *economic*, both lower- and upper-case, and selects those elements from within the `allnames` list.

```
econtitles <- as.character(allnames[str_detect(allnames, "^[Ee]conomic|\\s[Ee]conomic")])
length(econtitles)
```

```
Error: object 'allnames' not found
Error: object 'econtitles' not found
```

What in the hell? So many! I suppose that this is a good thing: at least one of the 440 journals should accept my crappy papers. If I blindly throw a dart in a bar, it may not hit the dartboard, but it will almost certainly hit one of the 440 patrons. Here is a random sample of ten journals:

```
sample(econtitles, 10)
```

```
Error in sample(econtitles, 10) : object 'econtitles' not found
```