Favorite Books

John Cruz

2023-03-08

Introduction

This project focuses on working with different types of files for analysis. I will be manually creating an HTML, XML and JSON formats that stores three of my favorite books related to data science and programming. Each file will store the title, author(s), publisher, published date, ISBN-13, and a best sellers rank from Amazon's website. The rankings were collected March 8, 2023.

HTML

HTML files is minimally comprised up of HTML elements and attributes. The HTML file storing my book data contains a web page title inside *<head>*, heading *<h1>* and *<table>*. The *<table>* is built using table headers *<th>* and table data *<td>*, similar to an Excel spreadsheet.

XML

XML forms a parent/child tree that stores information based on their relationship to each other. My XML file uses the "trunk" of the tree called *<fav_books>*. It then branches out into separate branches called *<book>*. Each *<book>* then contains the "leaves", unique information based on said book such as *<title>* and *<authors>*.

JSON

JSON can be compared similarly to Python's dictionaries. It contains a key:value pair that identifies the objects. To store my books, it is in a nested dictionary with key:value pairs such as fav_books:book:title, where the title is stored in book and book is store in fav_books.

Import Libraries

```
library(tidyverse)
library(rvest)
library(xml2)
library(jsonlite)
```

Import HTML File into Data Frame

To import the HTML file stored within the GitHub repository, I will use the *read_html()* function within the *rvest* library. This will then allow me to bring in the table using *read_table()* function to transform the html table into a data frame.

```
url <- 'https://raw.githubusercontent.com/hellojohncruz/favorite_books/main/books.html'

html <-
  read_html(url) |>
  html_table()

df_html <-
  as.data.frame(html) |>
  janitor::clean_names()

knitr::kable(df_html)
```

title	author_s	publishe	published	isbn_13	best_sellers_rank
Starting Out with C++ from Control Structures to Objects	Tony Gaddis	Pearson	February 13, 2017	978-0134498379	95,825
An Introduction to Statistical Learning: with Applications in R	Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani	Springer	July 30, 2021	978-1071614174	29,107
Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems	Aurélien Géron	O'Reilly Media	October 15, 2019	978-1492032649	18,543

Import XML File into Data Frame

To import the XML file stored within the GitHub repository, I will use the *read_xml()* function within the *xml2* library. We then can see the structure of the XML file using *xml_structure()*

```
url<- 'https://raw.githubusercontent.com/hellojohncruz/favorite_books/main/books.xml'

read_xml(url) |>
  xml_structure()
```

```
## <fav_books>
##   <book>
##     <title>
##       {text}
##     <authors>
##       {text}
##     <publisher>
##       {text}
```

```
##     <published>
##       {text}
##     <isbn_13>
##       {text}
##     <best_sellers_rank>
##       {text}
##   <book>
##     <title>
##       {text}
##     <authors>
##       {text}
##     <publisher>
##       {text}
##     <published>
##       {text}
##     <isbn_13>
##       {text}
##     <best_sellers_rank>
##       {text}
##   <book>
##     <title>
##       {text}
##     <authors>
##       {text}
##     <publisher>
##       {text}
##     <published>
##       {text}
##     <isbn_13>
##       {text}
##     <best_sellers_rank>
##       {text}
```

To transform the portions of the XML file, I stored into vectors each "leaf" data, then created a tibble combining them all into one.

```
xml_title <-
  read_xml(url) |>
  xml_find_all(xpath = "//title") |>
  xml_text()

xml_authors <-
  read_xml(url) |>
  xml_find_all(xpath = "//authors") |>
  xml_text()

xml_publisher <-
  read_xml(url) |>
  xml_find_all(xpath = "//publisher") |>
  xml_text()

xml_published <-
  read_xml(url) |>
  xml_find_all(xpath = "//published") |>
```

```
  xml_text()

xml_isbn_13 <-
  read_xml(url) |>
  xml_find_all(xpath = "//isbn_13") |>
  xml_text()

xml_rank <-
  read_xml(url) |>
  xml_find_all(xpath = "//best_sellers_rank") |>
  xml_text()

df_xml <-
  tibble(title = xml_title, author_s = xml_authors, publisher = xml_publisher, published = xml_publishe
             isbn_13 = xml_isbn_13, best_sellers_rank = xml_rank)

knitr::kable(df_xml)
```

title	author_s	publishe	published	isbn_13	best_sellers_rank
Starting Out with C++ from Control Structures to Objects	Tony Gaddis	Pearson	February 13, 2017	978-0134498379	95,825
An Introduction to Statistical Learning: with Applications in R	Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani	Springer	July 30, 2021	978-1071614174	29,107
Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems	Aurélien Géron	O'Reilly Media	October 15, 2019	978-1492032649	18,543

Import JSON File into Data Frame

To import the JSON file stored within the GitHub repository, I will use the *read_json()* function within the *jsonlite* library. We use the *simplifyVector = TRUE* parameter to create vectors that can be read into a data frame. Finally, using *clean_names()*, this will clean up the column names in the data frame.

What we do notice about the column names is that it creates a snake case format of how each key to value was reached. For example, for *title*, it required to go through fav_books > book > title to reach the data.

```
url <- 'https://raw.githubusercontent.com/hellojohncruz/favorite_books/main/books.json'

df_json <-
  as.data.frame(read_json(url, simplifyVector = TRUE)) |>
  janitor::clean_names()


knitr::kable(df_json)
```

fav_books_book_title	fav_books_book_author	fav_books_publisher	fav_books_publication_date	fav_books_isbn13	fav_books_best_sel
Starting Out with C++ from Control Structures to Objects	Tony Gaddis	Pearson	February 13, 2017	978-0134498379	95,825
An Introduction to Statistical Learning: with Applications in R	Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani	Springer	July 30, 2021	978-1071614174	29,107
Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems	Aurélien Géron	O'Reilly Media	October 15, 2019	978-1492032649	18,543

Conclusion

To import the data, we can utilize multiple formats to obtain the information inside a data frame. However, some column and data formatting may be required to get them all into a standardized form.