…Today the biggest book fair of the world starts again in Frankfurt, Germany. I thought this might be a good opportunity to do you some good!

Springer is one of the most renowned scientific publishing companies in the world. Normally, their books are quite expensive but also in the publishing business *Open Access* is a megatrend.

If you want to use R in a little fun project to find the latest additions of open access books to their program read on!

The idea is to create an R script which you can run from time to time to see whether there are new titles available. So, we need some place to store the retrieved data in a persistent manner: a *database*! For our purposes here most database systems would be total overkill but there is one great solution available: the amazing `RSQLite` package (on CRAN).

This package brings its own lightweight database with it, no need to install any additional software! And it is fully *SQL* compatible (for *Structured Query Language*, the industry standard of *relational database management systems*) like any decent database software.

So, you only have to install the `RSQLite` package and then load the `DBI` package (for *database interface*). To render the output table in an appealing form we will use the `htmlTable` package (on CRAN).

Have a look at the following fully documented code which should (hopefully) be quite clear:

```
library(DBI)
library(htmlTable)
# inital search for English books from 2019
springer_initial <- read.csv("https://link.springer.com/search/csv?facet-content-type=%22Book%22&previous-end-year=2019&date-facet-mode=in&facet-language=%22En%22&showAll=false&query=&facet-end-year=2019&previous-start-year=2019&facet-start-year=2019",
encoding = "UTF-8")
# current search for English books from 2020 - has to be updated in the following years!
springer_search <- read.csv("https://link.springer.com/search/csv?previous-end-year=2020&facet-content-type=%22Book%22&date-facet-mode=in&previous-start-year=2020&facet-language=%22En%22&showAll=false&query=&facet-start-year=2020&facet-end-year=2020", encoding = "UTF-8")

# open database connection
springer_db <- dbConnect(RSQLite::SQLite(), "my-db.sqlite")

# initialize database
if (!dbExistsTable(springer_db, "search")) {
  dbWriteTable(springer_db, "search", springer_initial)
}

# read current search table, replace it with new search and compare both
springer_search_old <- dbReadTable(springer_db, "search")
dbRemoveTable(springer_db, "search")
dbWriteTable(springer_db, "search", springer_search)
new_books <- setdiff(springer_search_old, dbReadTable(springer_db, "search"))
if (nrow(new_books) > 0) htmlTable(new_books[c("Item.Title", "Authors", "URL")])
```

[showing only a subset of the more than 200 (!) free titles in 2019]

	Item.Title	Authors	URL
47	Disrupting Finance	Theo LynnProf. John G. MooneyDr. Pierangelo RosatiProf. Mark	http://link.springer.com/book/10.1007/978-3-030-02330-0

	Item.Title	Authors	URL
		Cummins	
84	Understanding Statistics and Experimental Design	Prof. Dr. Michael H. HerzogProf. Dr. Gregory FrancisPh.D. Aaron Clarke	http://link.springer.com/book/10.1007/978-3-030-03499-3
85	InformationConsciousnessReality	Dr. James B. Glattfelder	http://link.springer.com/book/10.1007/978-3-030-03633-1
133	Modelling our Changing World	Dr. Jennifer L. CastleProf. Dr. David F. Hendry	http://link.springer.com/book/10.1007/978-3-030-21432-6
147	Fundamentals of Clinical Data Science	Dr. Pieter KubbenMichel DumontierProf. Dr. Andre Dekker	http://link.springer.com/book/10.1007/978-3-319-99713-1
169	Reality Lost	Vincent F. HendricksMads Vestergaard	http://link.springer.com/book/10.1007/978-3-030-00813-0
172	The Brownian Motion	Prof. Dr. Andreas LfflerProf. Dr. Lutz Kruschwitz	http://link.springer.com/book/10.1007/978-3-030-20103-6
186	Automated Machine Learning	Prof. Dr. Frank HutterLars KotthoffPh.D. Joaquin Vanschoren	http://link.springer.com/book/10.1007/978-3-030-05318-5
209	Lithium-Ion Batteries	Beta Writer	http://link.springer.com/book/10.1007/978-3-030-16800-1

```
# close database connection
dbDisconnect(springer_db)
```

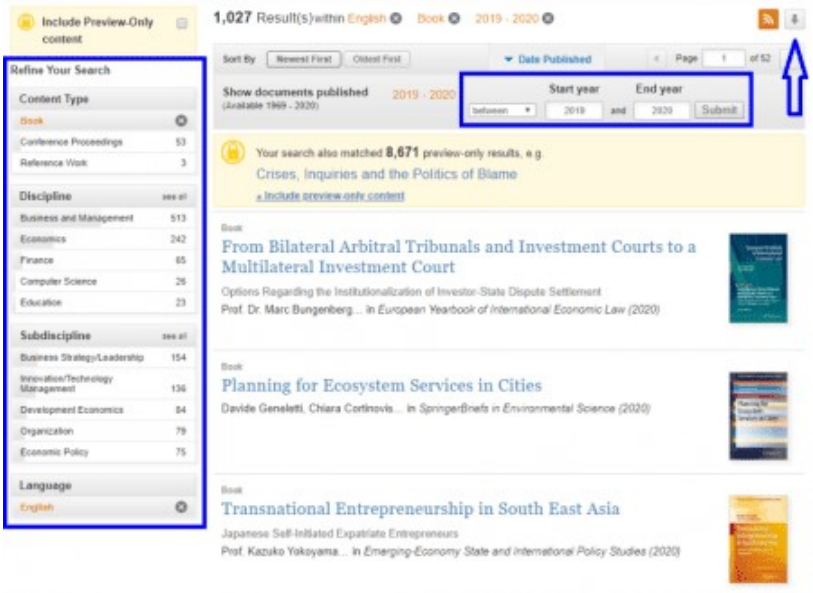
And you thought Christmas was yet to come, right!

As an aside, the last entry is an especially interesting case: it is the first machine-generated research book! The “author” *Beta Writer* was developed in a joint effort and in collaboration between Springer and researchers from Goethe University, Frankfurt. The book is a cross-corpora *auto-summarization* of current texts from SpringerLink, organized by means of a *similarity-based clustering* routine in coherent chapters and sections. It automatically condenses a large set of papers into a reasonably short book. More technical details of this fascinating endeavor, with the potential to revolutionize scientific publishing, can be found in the preface of the book.

By clicking on the link you will directly be directed to the respective book page, where you can download the *pdf* and in most cases also an *epub* file (bonus tip: in most cases you can also download a free version of the book for your kindle on amazon.com). To get clickable links you need to render *an HTML markdown* document. Otherwise, if you run it in RStudio directly you will have to copy and paste the links into your browser.

You just have to run the script from time to time to see what is new!

If you want to customize the data retrieved from link.springer.com have a look at their search interface:



You can customize your search by changing the values in the blue boxes. To get the URL which you can paste in the `read.csv` function above just right click on the button with the down arrow at the upper right corner (marked by the blue arrow) and choose “Copy link address” in the context menu.

In case you want to completely reset the database you can use the following function (with care):

```
# function for resetting the springer database
reset_springer_db <- function() {
  springer_db <- dbConnect(RSQLite::SQLite(), "my-db.sqlite")
  dbRemoveTable(springer_db, "search")
  dbDisconnect(springer_db)
}
```

One small thing: although I tried my very best there still seems to be an issue with the encoding… some special characters, like the German umlauts äöüÄÖÜ, are just not rendered. If you have a solution for me please leave it in the comments and I will add it to the post (or perhaps even write a post on the issues of encoding in R, RStudio and Windows).