Scraping Nuclear Reactors

In this project,[2] you're going to look at data about nuclear reactors. Let's use Japan as an example. Often, when you are doing a quick project, sources like Wikipedia are useful.

[2] Devised initially by Prof. Nicholas Horton, Amherst College

Go to the page `http:://en.wikipedia.org/wiki/List_of_nuclear_reactors"`. Find the reactor list for Japan. Figure A.24 shows part of the list[3] as a cut-and-paste image from a web browser.

[3] on March 23, 2015

Name	Reactor	Reactor		Status	Capacity in MW		Construction Start Date	Commercial Operation Date	Closure
		Type	Model		Net	Gross			
Fukushima Daiichi	1	BWR	BWR-3	Shutdown	439	460	25 Jul, 1967	26 Mar, 1971	19 May 2011
Fukushima Daiichi	2	BWR	BWR-4	Shutdown	760	784	09 Jun, 1969	18 Jul, 1974	19 May 2011
Fukushima Daiichi	3	BWR	BWR-4	Shutdown	760	784	28 Dec, 1970	27 Mar, 1976	19 May 2011
Fukushima Daiichi	4	BWR	BWR-4	Shutdown	760	784	12 Feb, 1973	12 Oct, 1978	19 May 2011

Figure A.24: Part of the Wikipedia table describing nuclear reactors in Japan.

Unfortunately, it is not a matter of cut-and-paste to get the tables in Wikipedia into the form of a data table in R. The tables often have a complex, non-tidy form. In addition, the tables are written using HTML tags, which can have be confusing. For instance, here a bit of the HTML behind the table of reactors in Japan.

```
<table class="wikitable sortable">
<tr>
<th rowspan="2" style="background:#FFDEAD;">Name</th>
<th rowspan="2" style="background:#FFDEAD;">Reactor</th>
<th colspan="2" style="background:#FFDEAD;">Reactor</th>
<th colspan="2" style="background:#FFDEAD;">Capacity in MW</th>
...
</tr>
<tr>
<td>Fukushima Daiichi</td>
<td>1</td><td>BWR</td><td>BWR-3</td><td>Shutdown</td>
```

```
<td>439</td><td>460</td><td>25 Jul, 1967</td>
<td>26 Mar, 1971</td><td>19 May 2011</td>
<tr>
```

Compare the human-readable version of the table with the HTML markup. You'll see that the data is there, but there is a lot of extraneous material and the arrangement is set not by position in a spreadsheet layout but by *HTML tags* like `<td>` and `<tr>`.

HTML TAG: A markup indicator, analogous to `*` or `###` or `[text](line)` in Markdown.

```
library(rvest)
library(lubridate)
page <- "http://en.wikipedia.org/wiki/List_of_nuclear_reactors"
xpath <- '//*[@id="mw-content-text"]/table'
table_list <- page %>%
  read_html() %>%
  html_nodes(xpath = xpath) %>%
  html_table(fill = TRUE)
```

The `result` object is not a data table; it is a *list* of data tables. Here are some of the operations you can apply to lists:

Description	Syntax	Example
How many elements in the list	`length(`*table*`)`	`length(tableList)`
Grab a single element	*table*`[[`*element number*`]]`	`tableList[[20]]`

1) Find the table element

Start with `head(tableList[[5]])` and go down the list until you find the table for Japan. The tables are listed by number in the same order that they appear on the page. As of the time of this writing,[4] `tableList[[5]]` is for Austria, so you'll have to go a good distance down the table to get to Japan.

[4] Wikipedia articles are works in progress. Over a period of even a few days they may have been modified substantially.

2) The table will look like this:

Your turn: In what ways is the table tidy? How is it not tidy? What's different about it from a tidy table?

Once you've answered the above questions ... and only then ... continue reading.

Among other things, two of the variables names are missing and others have multiple words separated by spaces. You can rename them using the data verb `rename()`, finding the names from the Wikipedia table. Another problem is that the first row is not data but a continuation of the variable names. So row number 1 should be dropped.

```
names(Japan)[c(4,7)] <- c("model", "grossMW")
Japan <-
  Japan %>%
  filter(row_number() > 1) %>%
  rename(name = Name, reactor = `Reactor No.`,
         type = Reactor,
         status = Status, netMW = `Capacity in MW`,
         construction = `Construction Start Date`,
         operation = `Commercial Operation Date`, closure = Closure)
```

This sort of variable-name cleaning is common. But it's not the only sort of reformatting that's needed here. Look at each of the variables and decide what the data type is: character, numerical, date, etc. Now use `str()` to see how the variable is typed in the data table itself.

You are going to need to `mutate()` the variables that are not in the right type. Some suggestions:

1. To convert a character string of digits into a number, use `as.numeric()` or `as.integer()`.

2. The `lubridate` package functions can be used to turn character string dates into a *POSIXct date object*. Identify what the format of the date is. The `lubridate` translation functions are `mdy()`, `mdyhms()`, `dmy()`, and so on.

POSIXCT DATE OBJECT: A type of R object representing points in time and allowing plotting, mathematical operations and extraction of components (such as the year or day of the week).

Your turn: Your cleaned data, make a plot of net generation capacity versus date of construction. Color the points by the *type* of reactor, e.g., BWR, PWR, or FBR.[5]In addition to your plot, give a sentence or two of interpretation; what patterns do you see?

[5] Boiling water reactor, pressurized water reactor, fast breeder reactor, respectively

Your turn: Carry out the same cleaning process for the China reactor table and append it with the Japan data. (Hint: You'll want one of the functions `cbind()` or `rbind()`.) You'll also want to add a variable to each table that has the name of the country.

Collating the data for all countries is a matter of repeating this process over and over. (You don't have to do this.) Inevitably, there are inconsistencies. For example, the US data is somewhat different in format than Japan or China.

Your turn: Read in to R the table on the Wikipedia page for US reactors. What is the physical meaning of a case in the US table? How does it compare to the meaning of a case for the Japan or China data.

Your turn: Make an informative graphic similar to Figure A.25 that shows how long it took between start of construction and commissioning for operation of each nuclear reactor in Japan (or another

country of your choice). One possibility: use reactor name vs date as the frame. For each reactor, set the glyph to be a line extending from start of construction to commissioning. You can do this with `geom_segment()` using name as the y coordinate and time as the x coordinate.

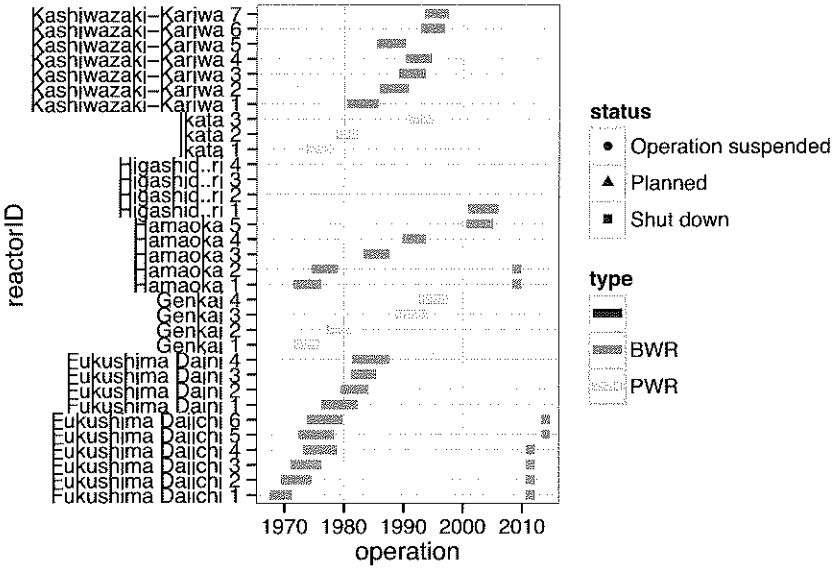

Figure A.25: Time interval from start of construction to operation.