# Webscraping with RSelenium

(see homework assignment at the bottom of this document)

* Webscraping JavaScript based websites requires techniques beyond what we learned in the Data Management course. One tool that allows you to scrape such websites is Selenium. Selenium is a very popular product, but it is not an “R” product. RSelenium is an R package that allows you to access the Selenium software via an R program.
* Watch this video:  
  Introduction to RSelenium using R (Jan 26, 2022) - <https://www.youtube.com/watch?v=U1BrIPmhx10>  
  I tried to take some notes for you below. They are not complete. I also added some comments of my own – so please read them.  
  + Install Java 8 (or later version) on your computer (The following explains a little about Java version numbers - https://www.stackchief.com/blog/Which%20Version%20of%20Java%20Should%20You%20Use%3F)
    - Java is a Programming language that is necessary to run some aspects of RSelenium software. You need to install it on your computer before running RSelenium. Instructions to do so are below. Note that there are many different products that all start with “Java …”. You need to install “Java SE” (the SE stands for “Standard Edition”).   
      NOTE – Java and Javascript are two TOTALLY DIFFERENT languages – the similarity between the names is a quirk of history. https://fossbytes.com/javascript-name-why-java/  
        
      Most software products (and Java is no exception) have newly released versions on a semi-regular schedule. This creates a nightmare for the software companies who have to support the existing versions. Therefore many software products, including Java, have designated LTS (long term support) versions. These versions will be supported for an extended period of time, while other versions may not be supported much after the next version comes out.
    - Windows 10 - <https://www.youtube.com/watch?v=IJ-PJbvJBGs&t=0s> (Mar 15, 2020) (10 min 37 sec)
    - Mac -
  + Install these RLibraries
    - install.packages(tidyverse)
    - install.packages(RSelenium)
    - install.packages(netstat) # for the free\_port() function
      * What is a “Network Port”? - (I will explain soon)
  + What do the following terms mean?
    - Client software – software that gets information from Server software (e.g. a browser is a client that get info from a web server program)
    - Server software – software that “serves” information to a “client” program (e.g. the software that runs a website is called a web server. This software serves info the the browsers (ie. The clients)
    - Driver software – a “driver” is typically a piece of software that “plugs into” or interfaces with some larger software. The purpose of the driver software is to make the larger software product able to work with various different other technologies.   
        
      For example, all modern computers have software that allows the operating system (e.g. Windows, Mac, etc) to connect with a printer. However, different types of printers have different features and functions. Therefore the exact software used to interface with a particular printer will be a little different than the software used to interface with a printer that has slightly different features. Therefore each particular printer requires its own particular “driver” software that was created specially for that printer to interact with the printer software for particular operating system (Windows, Mac, etc).   
        
      Another example is in RSelenium. The RSelnium software works with many different browsers. However, each browser has slightly different features and internal workings. Therefore RSelenium requires a different “driver software” for different types of web-browsers.
    - The R variable “remDr” stands for “remote driver”
    - Some methods (i.e. functions) of the Driver object
      * remDr$maxWindowSize()
      * remDr$navigage ( SOME URL )
      * remDr$findElement ( using = SEE DOCUMENTATION … )
      * SOME\_ELEMENT\_OBJECT$clickElement()
      * remDr$goBack()
      * SOME\_ELEMENT$sendKeysToElement( list (‘some text’, key=’enter’ ) )
      * driver.executeScript( SOME JAVASCRIPT CODE )
    - xpath – eg - https://riptutorial.com/xpath/example/6209/find-all-elements-with-certain-texta
* Go through the following tutorial. It goes through a complete scraping project. It uses RSelenium for part of it. There is also a video that goes through much of the material for the tutorial
  + <http://joshuamccrain.com/tutorials/web_scraping_R_selenium.html>
  + Video presentation of this tutorial - <https://www.youtube.com/watch?v=meapZAH7Jdc> (March 22, 2022 – 1:04:35)
    - 0:00 – 4:29 – you can skip this … presentation really starts at (4:30)
    - 4:30 – 44:44 - A review of basic webscraping using rvest
    - 44:45 – end – Introduction to RSelenium. Using RSelenium to scrape this site: <https://www.fcc.gov/media/engineering/dtvmaps>
  + Other related tutorials by the same author (not necessary for now, but good to know about):  
    http://joshuamccrain.com/tutorials

# PROJECT

Find a website to scrape that needs RSelenium techniques. Submit code that will successfully scrape the website. Choose a website that contains data you are interested in. This shouldn’t be too hard to find. YUTorah is a good example. For example you can scrape all information for shiurim from a particular speaker. Alternatively you can scrape all information for shiurim on a particular topic.