# Let R browse the web for you: An introduction to web-scraping with RSelenium

Nicole Schwitter | Warwick R UserGroup

Nicole.Schwitter.1@warwick.ac.uk

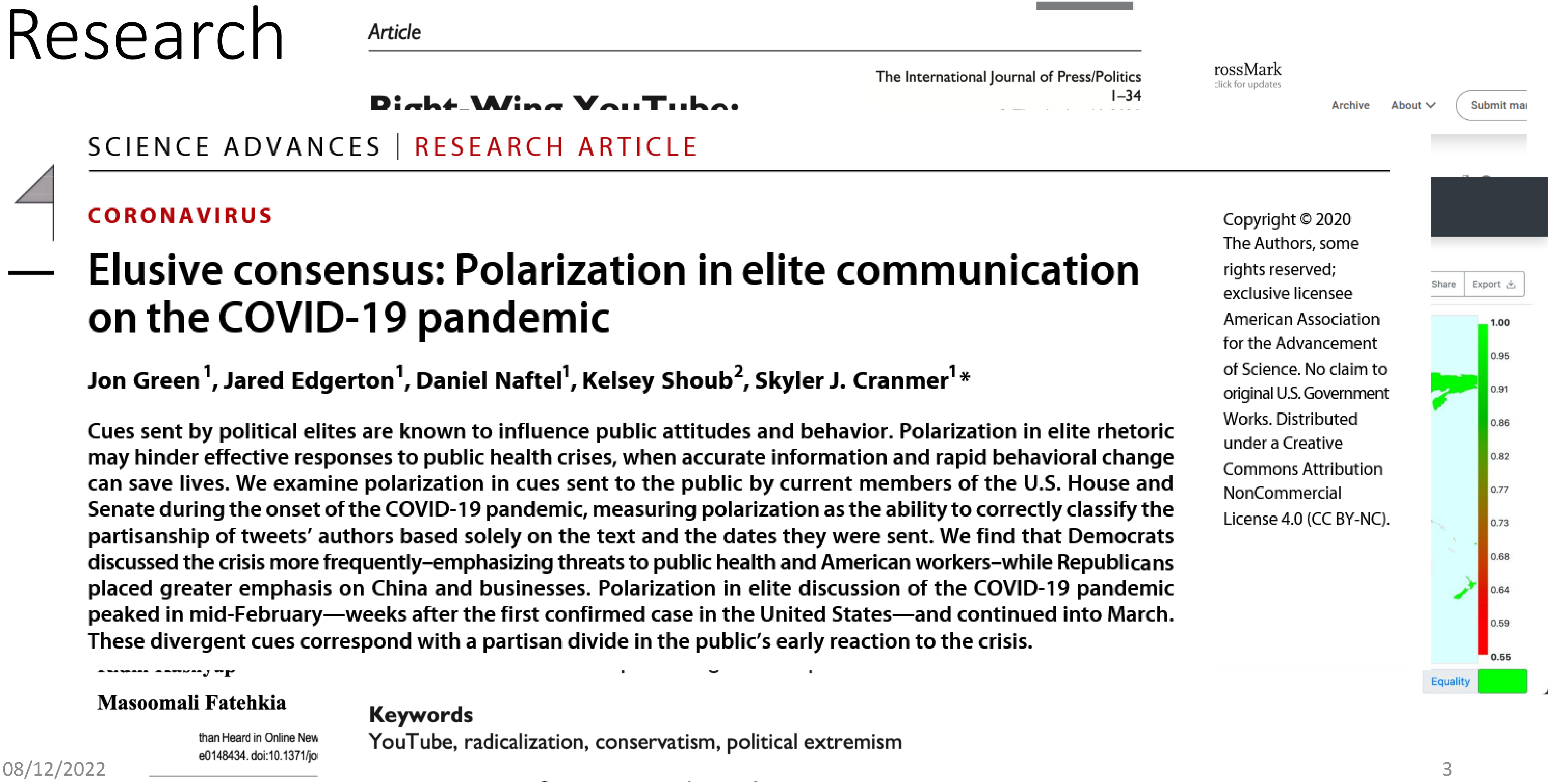
# Introduction

- Me
  - PhD student (in limbo) in the Department of Sociology
  - Seven years experience with web data collection

- Web data
  - Data published on the internet
  - Increasing volume: social media posts, digitised archives, press releases, online data bases, etc.
  - Accessible via HTTP requests

- Slides and code: [https://github.com/nschwitter/RSelenium-warwick](https://github.com/nschwitter/RSelenium-warwick)

# Current Examples: Web Data in Social Science Research



*Article*

The International Journal of Press/Politics
1–34

Right-Wing YouTube:

rossMark
:lick for updates

SCIENCE ADVANCES | RESEARCH ARTICLE

CORONAVIRUS

## Elusive consensus: Polarization in elite communication on the COVID-19 pandemic

Jon Green[1], Jared Edgerton[1], Daniel Naftel[1], Kelsey Shoub[2], Skyler J. Cranmer[1]*

Cues sent by political elites are known to influence public attitudes and behavior. Polarization in elite rhetoric may hinder effective responses to public health crises, when accurate information and rapid behavioral change can save lives. We examine polarization in cues sent to the public by current members of the U.S. House and Senate during the onset of the COVID-19 pandemic, measuring polarization as the ability to correctly classify the partisanship of tweets' authors based solely on the text and the dates they were sent. We find that Democrats discussed the crisis more frequently–emphasizing threats to public health and American workers–while Republicans placed greater emphasis on China and businesses. Polarization in elite discussion of the COVID-19 pandemic peaked in mid-February—weeks after the first confirmed case in the United States—and continued into March. These divergent cues correspond with a partisan divide in the public's early reaction to the crisis.

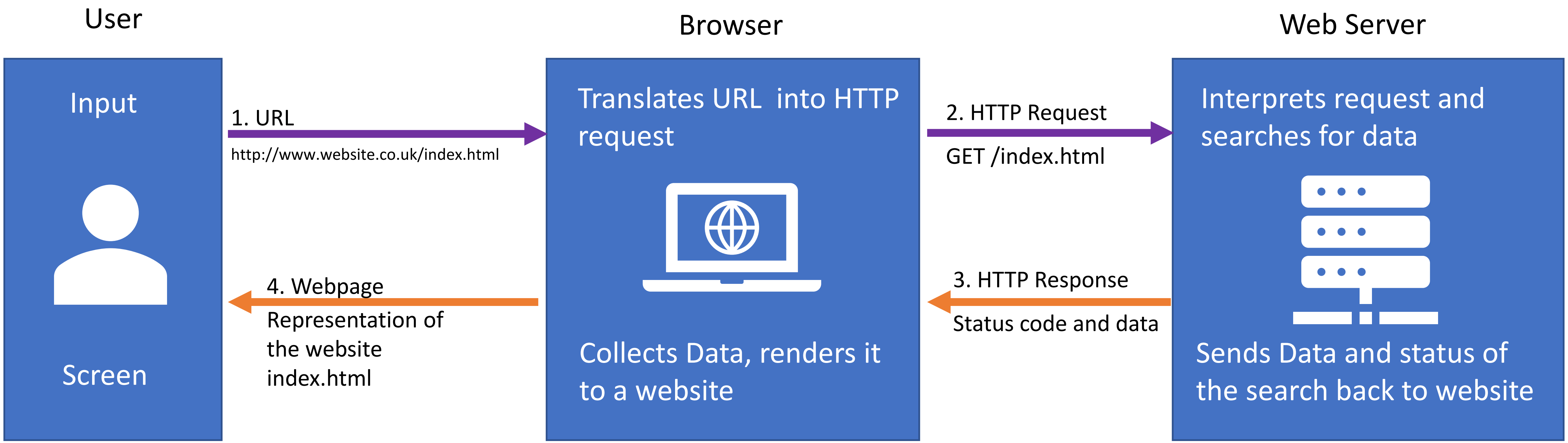
Masoomali Fatehkia

than Heard in Online New
e0148434. doi:10.1371/jo

**Keywords**
YouTube, radicalization, conservatism, political extremism

# How do we get the data?

# Understanding the communication process (HTTP)



User

Input

Screen

1. URL

http://www.website.co.uk/index.html

4. Webpage

Representation of the website index.html

Browser

Translates URL into HTTP request

Collects Data, renders it to a website

2. HTTP Request

GET /index.html

3. HTTP Response

Status code and data

Web Server

Interprets request and searches for data

Sends Data and status of the search back to website

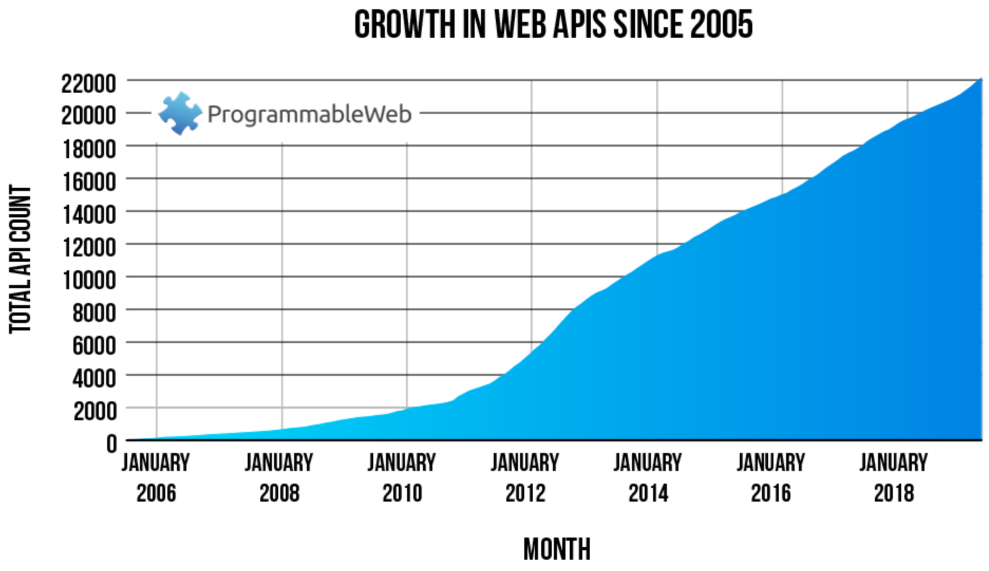
# Getting the data

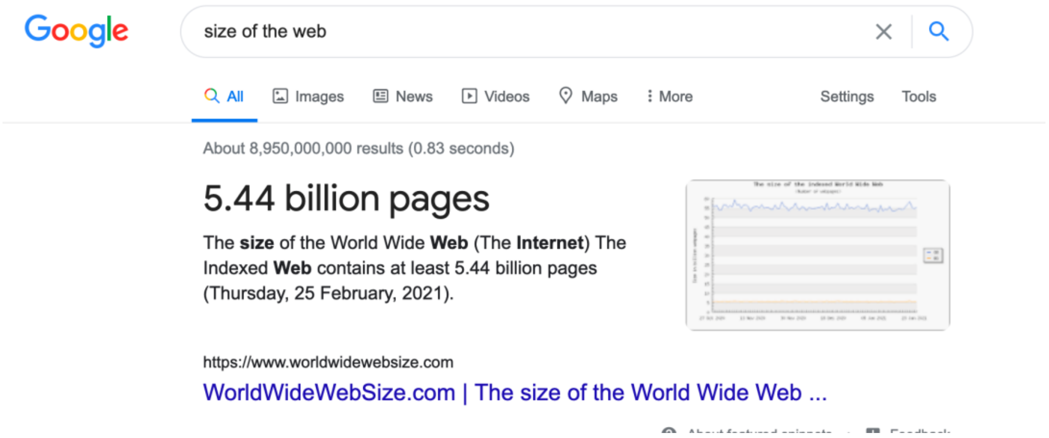
- Ctrl + c, Ctrl + v from displayed website
  - Tedious, error-prone, slow
  - Unstructured data: Sometimes, it might be your best option!
- Screen scraping
  - Automated collection of content hosted on webpage
    - Selecting contents of the webpage as accessed by the URL
    - We retrieve the HTML instead of displaying it in a browser.
  - Early origins: “Web crawling”
- Application programming interfaces (APIs)
  - Sending your own data requests to the server (if they let you)
  - Structured data

# Scraping vs API

- APIs
  - Extract data from public/non-public and visible/non-visible webpage content.
  - Data comes pre-packaged according to specified query.
  - Potential APIs to use: >22k indexed on: [https://www.programmableweb.com/api](https://www.programmableweb.com/api)

- Scraping
  - Extracts data from public/visible webpage content.
  - Needs to be reformatted to usable format.
  - Potential data sources: universe of webpages in existence: >5bn.


GROWTH IN WEB APIS SINCE 2005



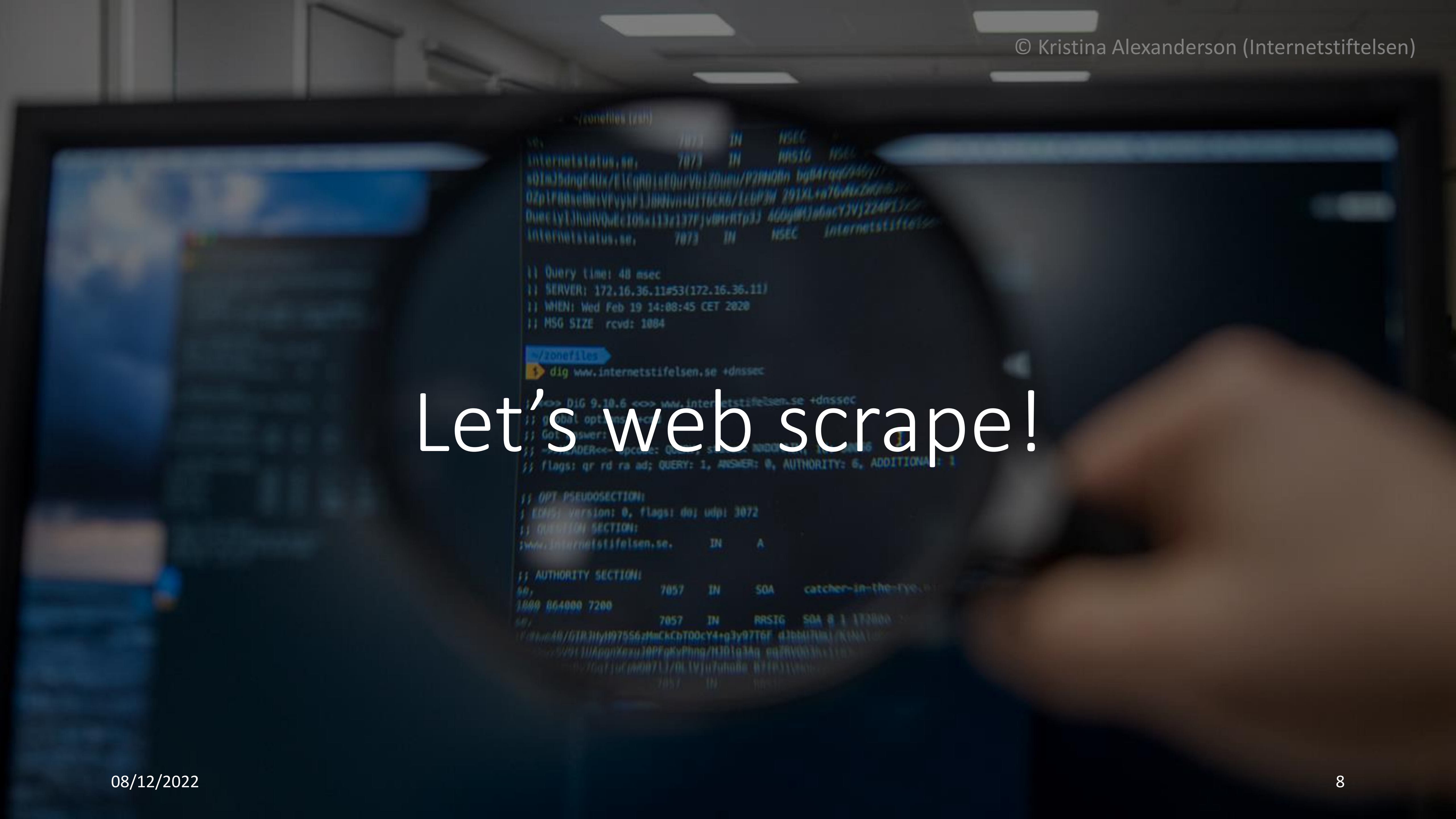
About 8,950,000,000 results (0.83 seconds)

5.44 billion pages

The **size** of the World Wide **Web** (The **Internet**) The Indexed **Web** contains at least 5.44 billion pages (Thursday, 25 February, 2021).

https://www.worldwidewebsize.com

WorldWideWebSize.com | The size of the World Wide Web ...



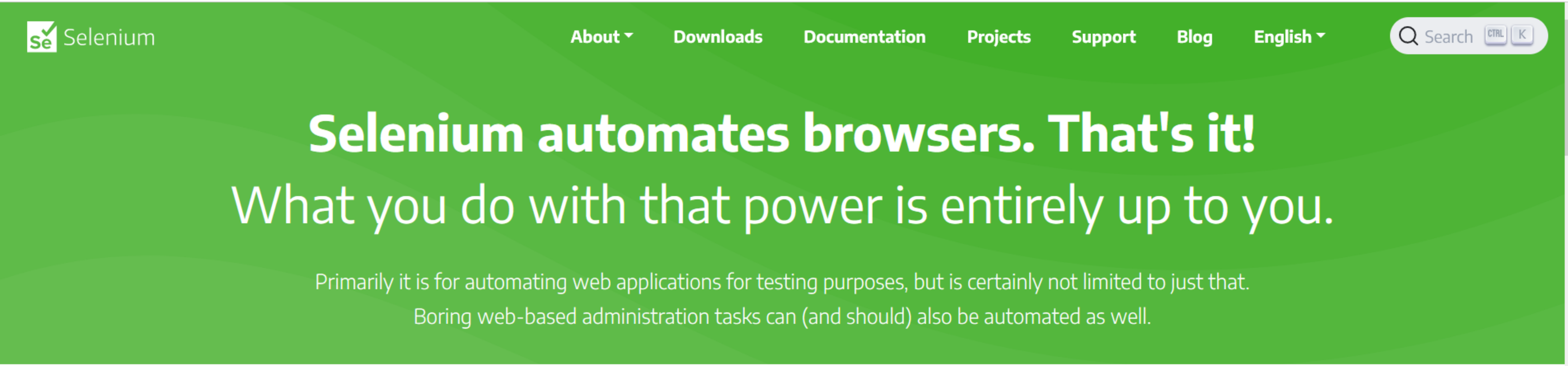
© Kristina Alexanderson (Internetstiftelsen)

# Let’s web scrape!

# Web scraping with R

- rvest: harvesting static HTML content
- https://rvest.tidyverse.org/
- Developer: Hadley Wickham

- RSelenium: driving a web browser natively
- https://www.selenium.dev/
- Developer: John Harrison

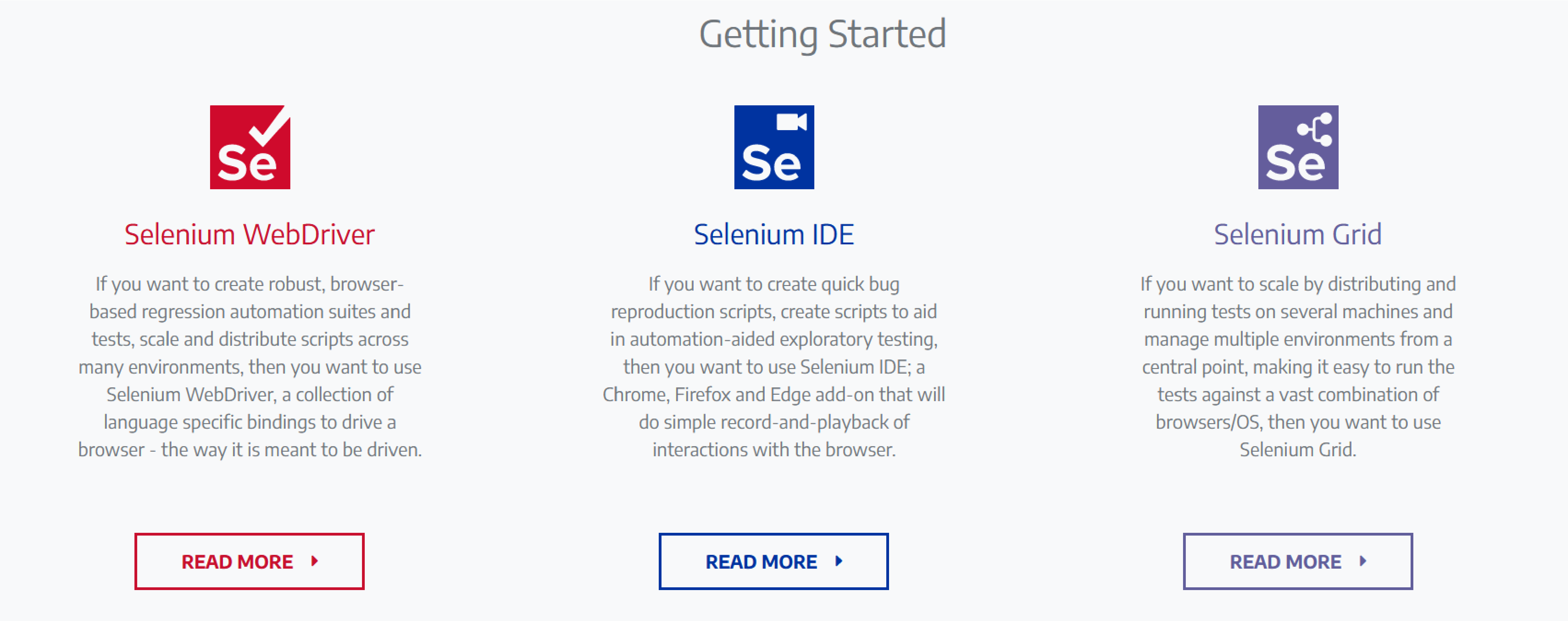


**Selenium automates browsers. That's it!**

What you do with that power is entirely up to you.

Primarily it is for automating web applications for testing purposes, but is certainly not limited to just that.
Boring web-based administration tasks can (and should) also be automated as well.

https://www.selenium.dev/



## Getting Started

### Selenium WebDriver

If you want to create robust, browser-based regression automation suites and tests, scale and distribute scripts across many environments, then you want to use Selenium WebDriver, a collection of language specific bindings to drive a browser - the way it is meant to be driven.

### Selenium IDE

If you want to create quick bug reproduction scripts, create scripts to aid in automation-aided exploratory testing, then you want to use Selenium IDE; a Chrome, Firefox and Edge add-on that will do simple record-and-playback of interactions with the browser.

### Selenium Grid

If you want to scale by distributing and running tests on several machines and manage multiple environments from a central point, making it easy to run the tests against a vast combination of browsers/OS, then you want to use Selenium Grid.

https://www.selenium.dev/

# Selenium

- Use cases
  - Web-scraping
  - Website testing / test automation
  - Any repetitive online tasks (filling in forms, etc.)
- How does it work?
  - Selenium WebDriver is an interface to write instructions.
  - Accepts commands which we write via the Client API, sends them to the browser.
  - This is implemented through a browser-specific browser-driver, which sends commands to a browser and retrieves the results.
  - It starts a browser instance and controls it.

# Let’s try to collect some data!

# A few introductory words

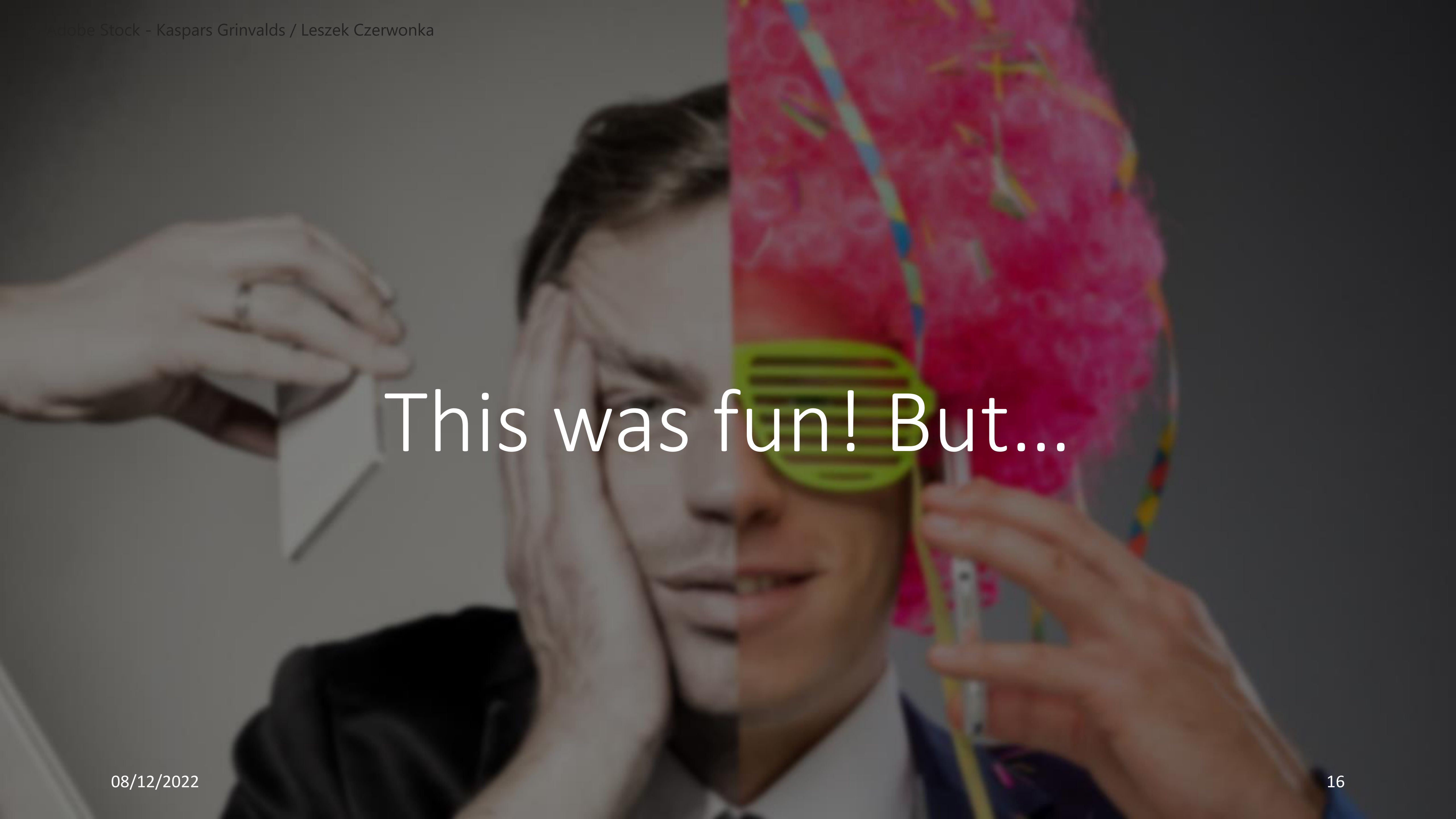
- I expect…
  - you have a basic knowledge of R.
  - you understand .Rmd files.
  - you have basic programming knowledge, e.g. know how loops work.

- I briefly cover internet technologies like HTML and CSS.
- I will provide sources and links to further readings and helpful tutorials.

- At any time: Feel free to interrupt and ask if you are lost somewhere!
- Getting Selenium to run can be a bit fiddly because of different platforms and browsers.

- RSelenium.Rmd

# This was fun! But…



Adobe Stock - Kaspars Grinvalds / Leszek Czerwonka

# Can we just collect everything and anything?

- No.

- Legal constraints placed by platforms (terms of services)
  - Be fair to the servers: limit number of requests and use timeouts.
- Ethical protection of users’ privacy  and contextual integrity
  - Protection of minorities and vulnerable groups
  - Users are not posting on social media to become research observations

If you consider this – happy scraping!

# Appendix: Running RSelenium

- As of now, the recommended way to run a Selenium Server is by running a Docker container (https://cran.r-project.org/web/packages/RSelenium/vignettes/basics.html)
- Docker is a free software for isolating applications using container virtualisation.
- To install Docker: https://www.docker.com/products/docker-desktop/
- We can start a Docker container from within RStudio (using the terminal).