

# ISA 401: Business Intelligence & Data Visualization

## 04: Scraping Webpages in

Fadel M. Megahed, PhD

Raymond E. Gloss Professor in Business  
Farmer School of Business  
Miami University

 @FadelMegahed



 fmegahed

 fmegahed@miamioh.edu


 Automated Scheduler for Office Hours

Fall 2025

# Quick Refresher from Last Class

- ✓ Subset data in .
- ✓ Read text-files, binary files (e.g., Excel, SAS, SPSS, Stata, etc), json files, etc.
- ✓ Export data from .

# Learning Objectives for Today's Class




- Understand when can we scrape data (i.e., `robots.txt`)
- Scrape a webpage using .

# Web Technology

# World Wide Web (WWW)

WWW (or the **Web**) is the information system where documents (web pages) are identified by Uniform Resource Locators (**URLs**)

A web page consists of:

-  **HTML** provides the basic structure of the web page
-  **CSS** controls the look of the web page (optional)
-  **JS** is a programming language that can modify the behavior of elements of the web page (optional)

# Hypertext Markup Language (HTML)

- with the extension `.html`.
- rendered using a web browser via an URL.
- text files that follows a special syntax that alerts web browsers how to render it.

## via a web browser

← → ↻ 🏠 ⚠ Not secure | plane crash info.com/2021/2021.htm

Apps Teaching Research Misc MU Mail MU Calendar Overleaf Canvas

2021

Date	Location / Operator	Aircraft Type / Registration	Fatalities
<a href="#">09 Jan 2021</a>	Near Jakarta, Indonesia Sriwijaya Air	Boeing 737-524 PK-CLC	62/62(0)
<a href="#">02 Mar 2021</a>	Pieri, Sudan South Sudan Supreme Airlines	Let L-410UVP-E HK-4274	10/10(0)
<a href="#">28 Mar 2021</a>	Near Butte, Alaska Soloy Helicopters	Eurocopter AS350B3 Ecureuil N351SH	5/6(0)
<a href="#">21 May 2021</a>	Near Kaduna, Nigeria Military - Nigerian Air Force	Beechcraft B300 King Air 350i NAF203	11/11(0)
<a href="#">10 Jun 2021</a>	Near Pailin, Myanmar Military - Myanmar Air Force	Beechcraft 1900D 4610	12/14(0)
<a href="#">04 Jul 2021</a>	Patikul, Sulu, Philippines Military - Philippine Air Force	Lockheed C-130H Hercules 5125	50/96(3)
<a href="#">06 Jul 2021</a>	Palana, Russia Kamchatka Aviation Enterprise	Antonov An 26B-100 RA-26085	28/28(0)
<a href="#">12 Sep 2021</a>	Kazachinskoye, Russia Aeroservice/SILA	Let L-410UVP-E20 RA-67042	4/16(0)
<a href="#">27 Dec 2021</a>	El Cajon, California Med Jet	Learjet 35A N880Z	4/4(0)

[Return to Home Page](#)

Copyright © Richard Kebabjian / www.plane crash info.com

## via a text editor

```
1 <html>
2
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
5   <meta name="GENERATOR" content="Microsoft FrontPage 4.0">
6   <meta name="description" content="Aviation accidents">
7   <meta name="keywords" content="aircraft accident, plane crash, aviation disaster, safety, aviation safety, aviation accident,
8   aircraft, plane, statistics, airline statistics, airline, airlines, hijack, pilot, probable cause, crash, boeing, cockpit,
9   <meta name="ProgId" content="FrontPage.Editor.Document">
10  <meta name="Title" content="Aviations accidents 2021">
11  <title>2021</title>
12 </head>
13
14 <body>
15   <p align="center"><b><font face="Arial" color="#B086FF" size="5">2021</font></b></p>
16   <div align="center">
17     <center>
18       <table border="1" cellpadding="4" cellspacing="0" width="700">
19         <tr>
20           <td width="75" bgcolor="#B086FF" align="left"><b><font face="Arial" size="2">Date</font></b></td>
21           <td bgcolor="#B086FF" align="left"><b><font face="Arial" size="2">Location / Operator</font></b></td>
22           <td bgcolor="#B086FF" align="left"><b><font face="Arial" size="2">Aircraft Type / Registration</font></b></td>
23           <td align="right"><b><font face="Arial" size="2">Fatalities</font></b></td>
24         </tr>
25         <tr>
26           <td colspan="4"><hr>
27         </tr>
28         <tr>
29           <td align="left" valign="top"><font face="Arial" size="2"><a href="2021-1.htm">09 Jan 2021</a></td>
30           <td align="left" valign="top"><font face="Arial" size="2">Near Jakarta, Indonesia<br>Sriwijaya Air
31         </td>
32           <td align="left" valign="top"><font face="Arial" size="2">Boeing 737-524<br>PK-CLC</td>
33           &td align="right" valign="top"><font face="Arial" size="2">62/62(0)</td>
34         </tr>
35         <tr>
36           &td colspan="4"><hr>
37         </tr>
38         &td align="left" valign="top"><font face="Arial" size="2"><a href="2021-2.htm">02 Mar 2021</a></td>
39         &td align="left" valign="top"><font face="Arial" size="2">Pieri, Sudan<br>South Sudan Supreme Airlines
40         </td>
41         &td align="left" valign="top"><font face="Arial" size="2">Let L-410UVP-E<br>HK-4274 </td>
42         &td align="right" valign="top"><font face="Arial" size="2">10/10(0)</td>
43         </tr>
44       </table>
```

# HTML Structure

```
<!DOCTYPE html>

<html>
  <!--This is a comment and ignored by web client.-->
  <head>
    <!--This section contains web page metadata.-->
    <title>ISA 401: Business Intelligence and Data Viz</title>
    <meta name="author" content="Fadel Megahed">
    <link rel="stylesheet" href="css/styles.css">
  </head>

  <body>
    <!--This section contains what you want to display on your web page.-->
    <h1>I'm a first level header</h1>
    <p>This is a <b>paragraph</b>.</p>
  </body>
</html>
```

# HTML Syntax

`<span style="color:blue;">Author content</span>` Author content

---

start tag: `<span style="color:blue;">Author content</span>`

end tag: `<span style="color:blue;">Author content</span>`

content: `<span style="color:blue;">Author content</span>`

element name: `<span style="color:blue;">Author content</span>`

attribute: `<span style="color:blue;">Author content</span>`

attribute name: `<span style="color:blue;">Author content</span>`

attribute value: `<span style="color:blue;">Author content</span>`

---

**Not all HTML tags have an end tag**, for example:

`` → 



# HTML Elements

block element:	<code>&lt;div&gt;content&lt;/div&gt;</code>
inline element:	<code>&lt;span&gt;content&lt;/span&gt;</code>
paragraph:	<code>&lt;p&gt;content&lt;/p&gt;</code>
header level 1:	<code>&lt;h1&gt;content&lt;/h1&gt;</code>
header level 2:	<code>&lt;h2&gt;content&lt;/h2&gt;</code>
italic:	<code>&lt;i&gt;content&lt;/i&gt;</code>
emphasised text:	<code>&lt;em&gt;content&lt;/em&gt;</code>
strong importance:	<code>&lt;strong&gt;content&lt;/strong&gt;</code>
link:	<code>&lt;a href="https://github.com/fmegahed/isa401"&gt;content&lt;/a&gt;</code>
unordered list:	<code>&lt;ul&gt; &lt;li&gt;item 1&lt;/li&gt; &lt;li&gt;item 2&lt;/li&gt; &lt;/ul&gt;</code>

# Cascading Style Sheet (CSS)

- with the extension `.css`
- 3 ways to style elements in HTML:
  - **inline** by using the `style` attribute inside HTML start tag:

```
<h1 style="color:blue;">Blue Header</h1>
```
  - **externally** by using the `<link>` element:

```
<link rel="stylesheet" href="styles.css">
```
  - **internally** by defining within `<style>` element:

```
<style type="text/css">
h1 { color: blue; }
</style>
```

By convention, the `<style>` and `<link>` elements tend to go into the `<head>` section of the HTML document.

# CSS Syntax

```
<style type="text/css">
h1 { color: blue; }
</style>
<h1>This is a header</h1>
```

This is a header

---

selector:	<code>h1 { color: blue; }</code>
property:	<code>h1 { color: blue; }</code>
property name:	<code>h1 { color: blue; }</code>
property value:	<code>h1 { color: blue; }</code>

---

You may have multiple properties for a single selector. ➡

```
h1 {
  color: blue;
  font-size: 16pt;
}
```

# CSS Properties

```
<div>Sample text</div>
```

background color:	<code>div { background-color: yellow; }</code>	Sample text
text color:	<code>div { color: purple; }</code>	Sample text
border:	<code>div { border: 1px dashed brown; }</code>	Sample text
left border only:	<code>div { border-left: 10px solid pink; }</code>	Sample text
text size:	<code>div { font-size: 10pt; }</code>	Sample text
padding:	<code>div { background-color: yellow; padding: 10px; }</code>	Sample text
margin:	<code>div { background-color: yellow; margin: 10px; }</code>	Sample text

# CSS Selector

<code>.classname</code>	selects all elements with the attribute <code>class="classname"</code> .
<code>.c1.c2</code>	selects all elements with <i>both</i> <code>c1</code> and <code>c2</code> within its class attribute.
<code>.c1 .c2</code>	selects all elements with class <code>c2</code> that is a descendant of an element with class <code>c1</code> .
<code>#p1</code>	selects all elements with the attribute <code>id="p1"</code> .

```
<h1>This is a sample html</h1>
<blockquote>
  <p>Maybe stories are just
  <footer>—Brene Brown</footer>
</blockquote>
```

Unlike `class`, you can only have one `id` value and must be unique in the whole HTML document.

```
<div id="p1" class="parent">
  Hmm
  <p>Hi!</p>
  How are you?
  <div class="child nice">
    <p>Hello!</p>
  </div>
</div>
```

```
<p>Household 1</p>
```

```
<div class="parent">
  <p>Hi!</p>
  <blockquote class="child rebel">
    <p>Don't talk to me!</p>
  </blockquote>
</div>
```

```
<span class="child">
  <span class="parent child rebel">
    <p>Clean your room!</p>
  </span>
</span>
```

```
<p>End of households</p>
```

# JavaScript (JS)\*

- JS is a programming language and enable interactive components in HTML documents.
- 2 ways to insert JS into a HTML document:
  - **internally** by defining within `<script>` element:

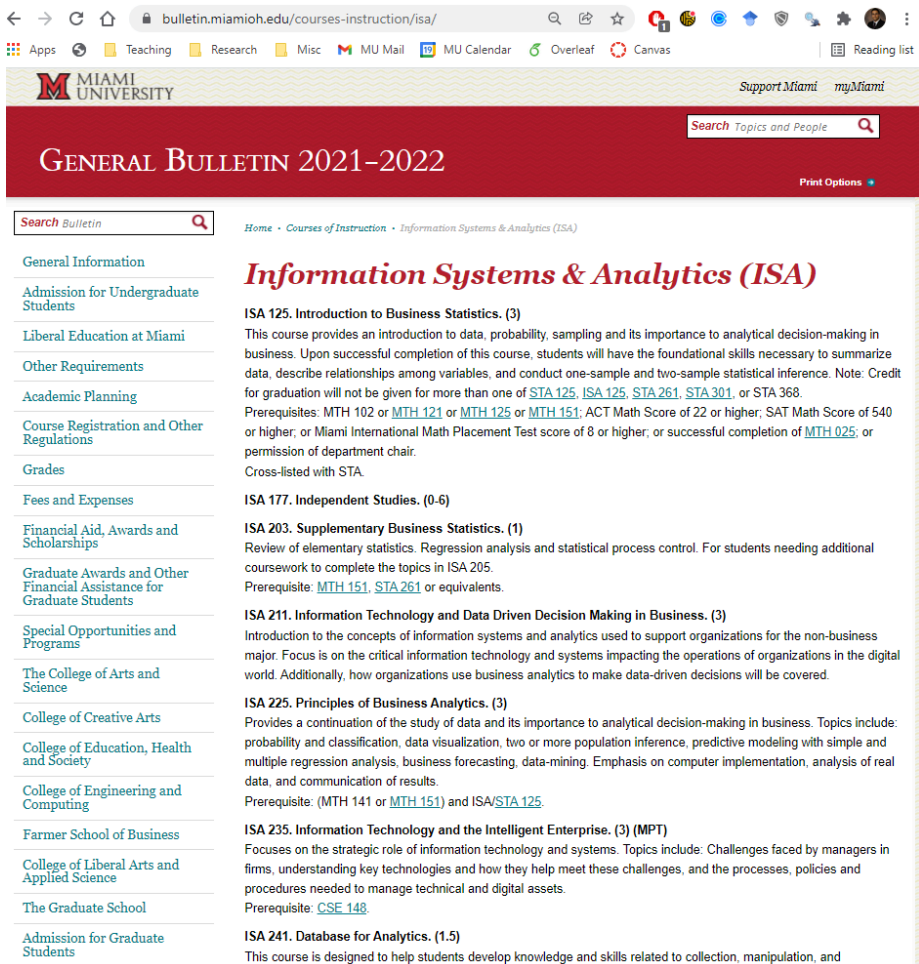
```
<script>  
document.getElementById("p1").innerHTML = "content";  
</script>
```

- **externally** by using the `src` attribute to refer to the external file:

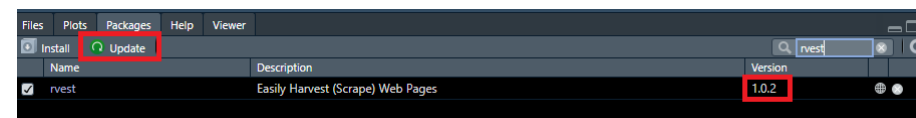
```
<script src="js/myjs.js"></script>
```

# Web Scrapping

# Invest: Step 1 - Reading Static HTML Pages



Use `{rvest}`  $\geq$  v1.0.2 (if not, update)



```
if(require(pacman)==FALSE) install.packages("pacman")
pacman::p_load(rvest)
isa_courses = read_html("http://bulletin.baylor.edu/undergraduate/academic/undergraduate-catalog/undergraduate-catalog-2019-2020/undergraduate-catalog-2019-2020.html")
isa_courses
```

```
## {html_document}
## <html xml:lang="en" lang="en" dir="ltr">
## [1] <head>\n<title>Information System
## [2] <body>\n\n\n\n\n\n\n\n<!-- Google Tag
```



# rvest: Step 2 - Selecting HTML Elements

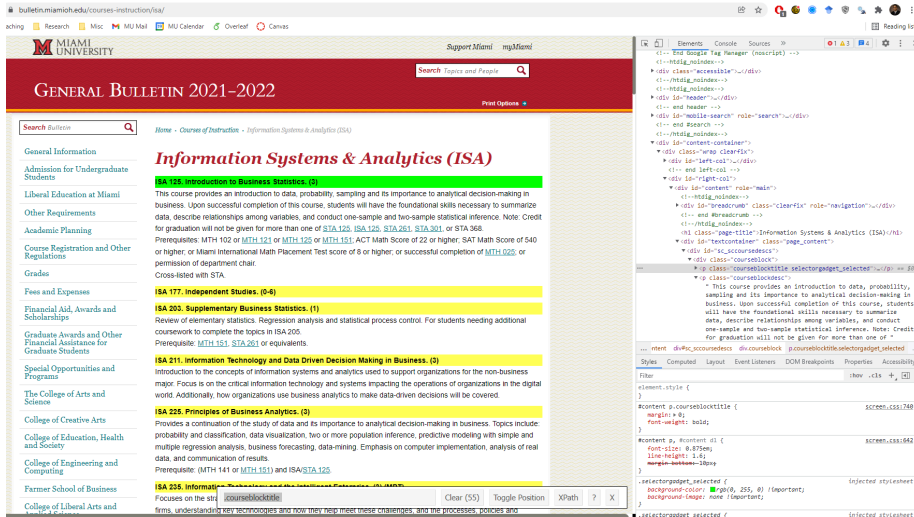
## Inspector

The screenshot shows the Chrome DevTools Inspector. The 'Elements' panel on the right displays the DOM tree. The 'Information Systems & Analytics (ISA)' section is selected, showing its HTML structure. The 'Styles' panel on the left shows the default styles for the selected element.

## Selector Gadget

The screenshot shows the Chrome DevTools Selector Gadget. The 'Elements' panel on the right displays the DOM tree. The 'Information Systems & Analytics (ISA)' section is selected. The 'Selector Gadget' is visible on the left, showing the selected element's path and the corresponding CSS selector.

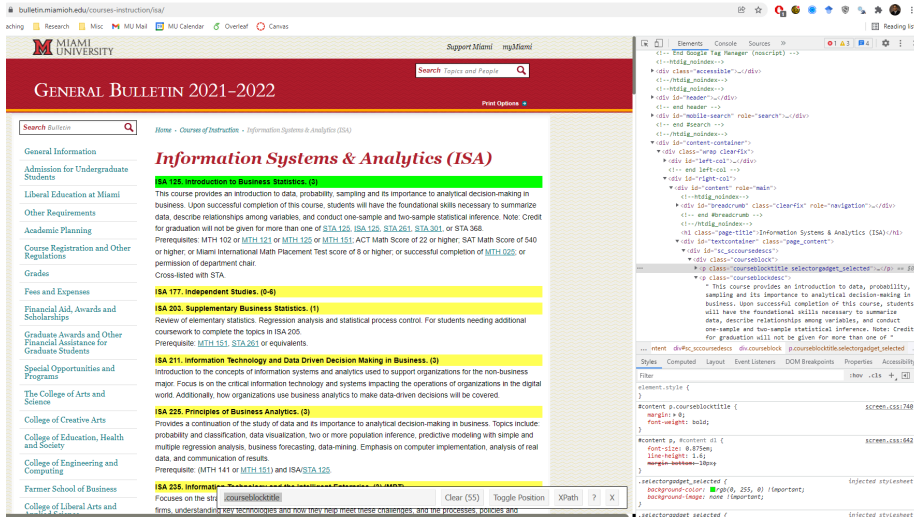
# rvest: Step 2 - Selecting HTML Elements



```
isa_course_titles = isa_courses |>
  html_elements(css = "p.courseblocktitle")
isa_course_titles
```

```
## {xml_nodeset (50)}
## [1] <p class="courseblocktitle"><str
## [2] <p class="courseblocktitle"><str
## [3] <p class="courseblocktitle"><str
## [4] <p class="courseblocktitle"><str
## [5] <p class="courseblocktitle"><str
## [6] <p class="courseblocktitle"><str
## [7] <p class="courseblocktitle"><str
## [8] <p class="courseblocktitle"><str
## [9] <p class="courseblocktitle"><str
## [10] <p class="courseblocktitle"><str
## [11] <p class="courseblocktitle"><str
## [12] <p class="courseblocktitle"><str
## [13] <p class="courseblocktitle"><str
## [14] <p class="courseblocktitle"><str
## [15] <p class="courseblocktitle"><str
## [16] <p class="courseblocktitle"><str
## [17] <p class="courseblocktitle"><str
## [18] <p class="courseblocktitle"><str
## [19] <p class="courseblocktitle"><str
## [20] <p class="courseblocktitle"><str
## [21] <p class="courseblocktitle"><str
## [22] <p class="courseblocktitle"><str
## [23] <p class="courseblocktitle"><str
## [24] <p class="courseblocktitle"><str
## [25] <p class="courseblocktitle"><str
## [26] <p class="courseblocktitle"><str
## [27] <p class="courseblocktitle"><str
## [28] <p class="courseblocktitle"><str
## [29] <p class="courseblocktitle"><str
## [30] <p class="courseblocktitle"><str
## [31] <p class="courseblocktitle"><str
## [32] <p class="courseblocktitle"><str
## [33] <p class="courseblocktitle"><str
## [34] <p class="courseblocktitle"><str
## [35] <p class="courseblocktitle"><str
## [36] <p class="courseblocktitle"><str
## [37] <p class="courseblocktitle"><str
## [38] <p class="courseblocktitle"><str
## [39] <p class="courseblocktitle"><str
## [40] <p class="courseblocktitle"><str
## [41] <p class="courseblocktitle"><str
## [42] <p class="courseblocktitle"><str
## [43] <p class="courseblocktitle"><str
## [44] <p class="courseblocktitle"><str
## [45] <p class="courseblocktitle"><str
## [46] <p class="courseblocktitle"><str
## [47] <p class="courseblocktitle"><str
## [48] <p class="courseblocktitle"><str
## [49] <p class="courseblocktitle"><str
## [50] <p class="courseblocktitle"><str
```

# rvest: Step 3 - Getting HTML Text



```
isa_course_titles_en = isa_course_titles_html_text2()
```

```
isa_course_titles_en
```

```
## [1] "ISA 125. Introduction to Business Statistics. (3)"
## [2] "ISA 177. Independent Studies. (0-4)"
## [3] "ISA 211. Information Technology and Data Driven Decision Making in Business. (3)"
## [4] "ISA 225. Principles of Business Analytics. (3)"
## [5] "ISA 235. Information Technology and Data Driven Decision Making in Business. (3)"
## [6] "ISA 241. Database for Analytics. (3)"
## [7] "ISA 242. Programming for Analytics. (3)"
## [8] "ISA 250. Basic Math for Analytics. (3)"
## [9] "ISA 277. Independent Studies. (0-4)"
## [10] "ISA 301. Business Data Communications. (3)"
## [11] "ISA 303. Enterprise Systems. (3)"
## [12] "ISA 305. Information Technology and Data Driven Decision Making in Business. (3)"
## [13] "ISA 321. Optimization in Business. (3)"
## [14] "ISA 333. Nonparametric Statistical Inference. (3)"
## [15] "ISA 335. Blockchain and Business Analytics. (3)"
## [16] "ISA 340. Internship. (0-20)"
## [17] "ISA 345. Database for Analytics. (3)"
```

# Demo: Scraping the Course Descriptions

- We will build on the previous example and we will scrape the **course descriptions** associated with these courses.
- Then, we will create a **data frame** containing **both** the **course titles** and **descriptions**
- Then, we will **export the results to a CSV** so that we can analyze that in a separate program if we wanted to.

# Non-Graded Class Activity

---

Activity

Your Solution

My Solution

---

- Go to [this database on plane crashes](#)
- Scrape the HTML table. **Note the difference from text elements:**
  - The CSS selector for `html_elements()` will be different.
  - You will extract a table (in its **entirety**) and hence:
  - we will use `html_table()` instead of `html_text2()`
- Store the scraped data in an appropriate location on your computer (e.g., within the data folder for ISA 401)

# Legal and Ethical Issues with Web Scraping

# Robots.txt

When scraping/crawling the web you need to be aware of `robots.txt`.

*The robots exclusion standard, also known as the robots exclusion protocol or simply robots.txt, is a standard used by websites to communicate with web crawlers and other web robots. The standard specifies how to inform the web robot about which areas of the website should not be processed or scanned. --- [Wikipedia](#)*

Using the excellent `robotstxt` 📦 to check if scraping/crawling a specific directory is allowed.

```
if(require(robotstxt)==FALSE) install.packages("robotstxt")
robotstxt::paths_allowed(paths = "2025/", domain = "planecrashinfo.com", bot = "*")
```

```
## [1] TRUE
```

# Terms of Service

Most large companies have **terms of service** that supplement what is permitted and/or disallowed on their `robots.txt` file. Examples include:

- [Yelp's US Terms of Service](#)
- [LinkedIn Terms of Service](#)



# Ethical/Legal Considerations

- **Use of publicly available reviews as a part of your service:** Would you classify the [Yelp vs Google Feud](#) as such an example?

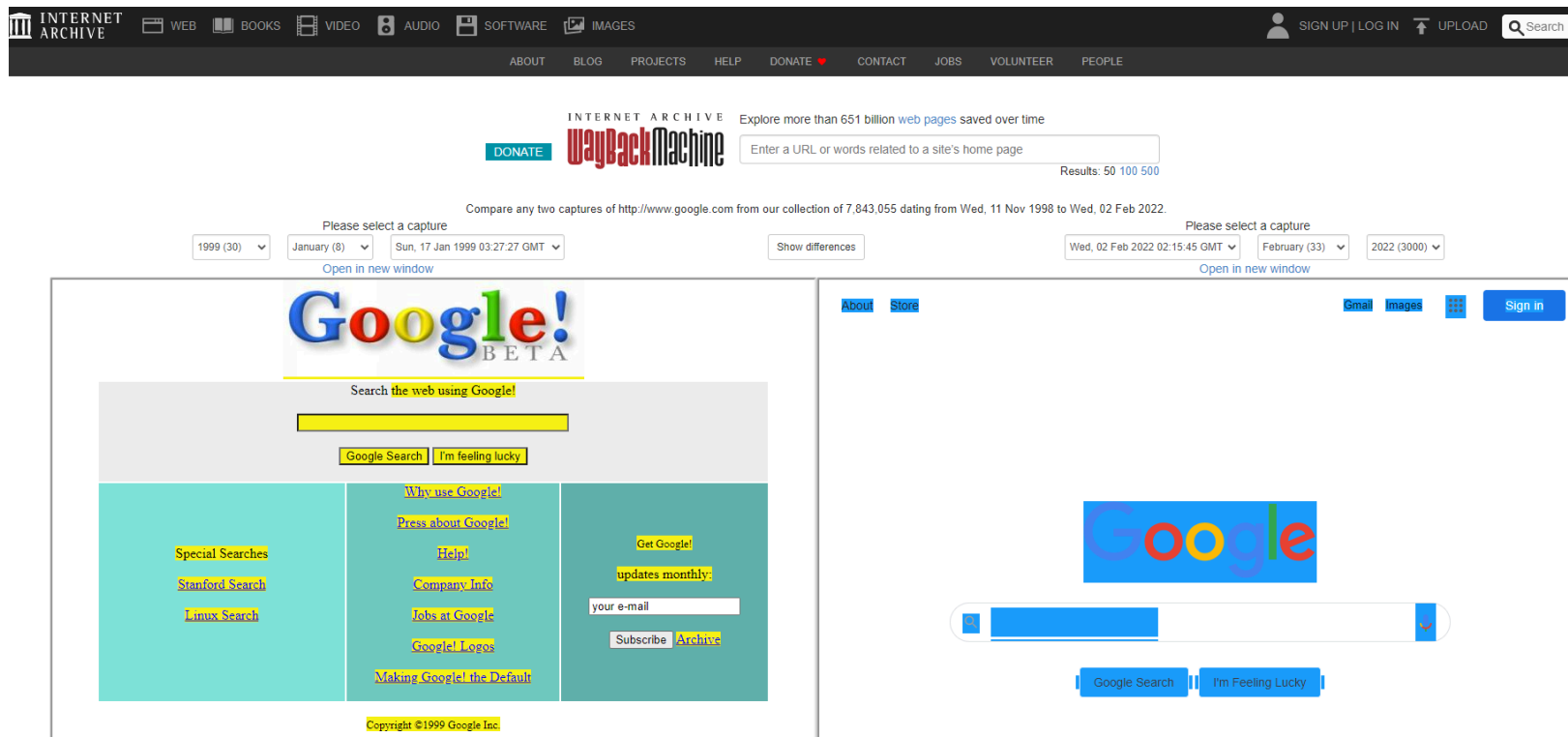


# Ethical/Legal Considerations

- **Use of publicly available profiles as a part of your service:**
  - LinkedIn vs Hiq Labs: Ninth Circuit Decision in 2019
  - Revival of Case in 2021 by Supreme Court

# Ethical/Legal Considerations

- What about scraping entire websites/webpages for the purpose of archiving the internet?




The evolution of the home page for Google per the Wayback Machine

# Recap

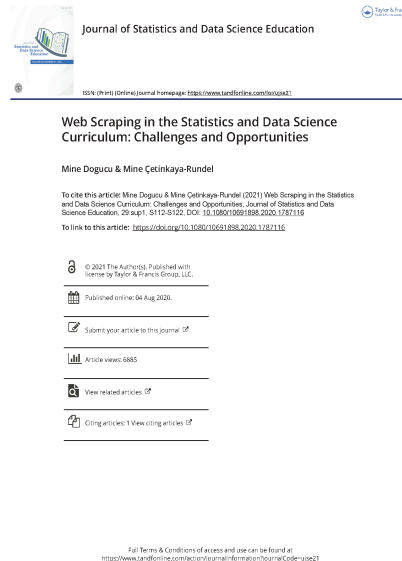
# Summary of Main Points

By now, you should be able to do the following:

- Understand when can we scrape data (i.e., `robots.txt`)
- Scrape a webpage using .

# Things to Do to Prepare for Next Class

- Go over your notes, read through the supplementary material (below) and complete **Assignment 04** on Canvas.



- PDF of Published Paper
- ePub of Published Paper

- Selector Gadget
- Getting Started with rvest