Intro to Programming for Public Policy Week 7
The Web and Scraping

Eric Potash

May 7, 2018

The Internet

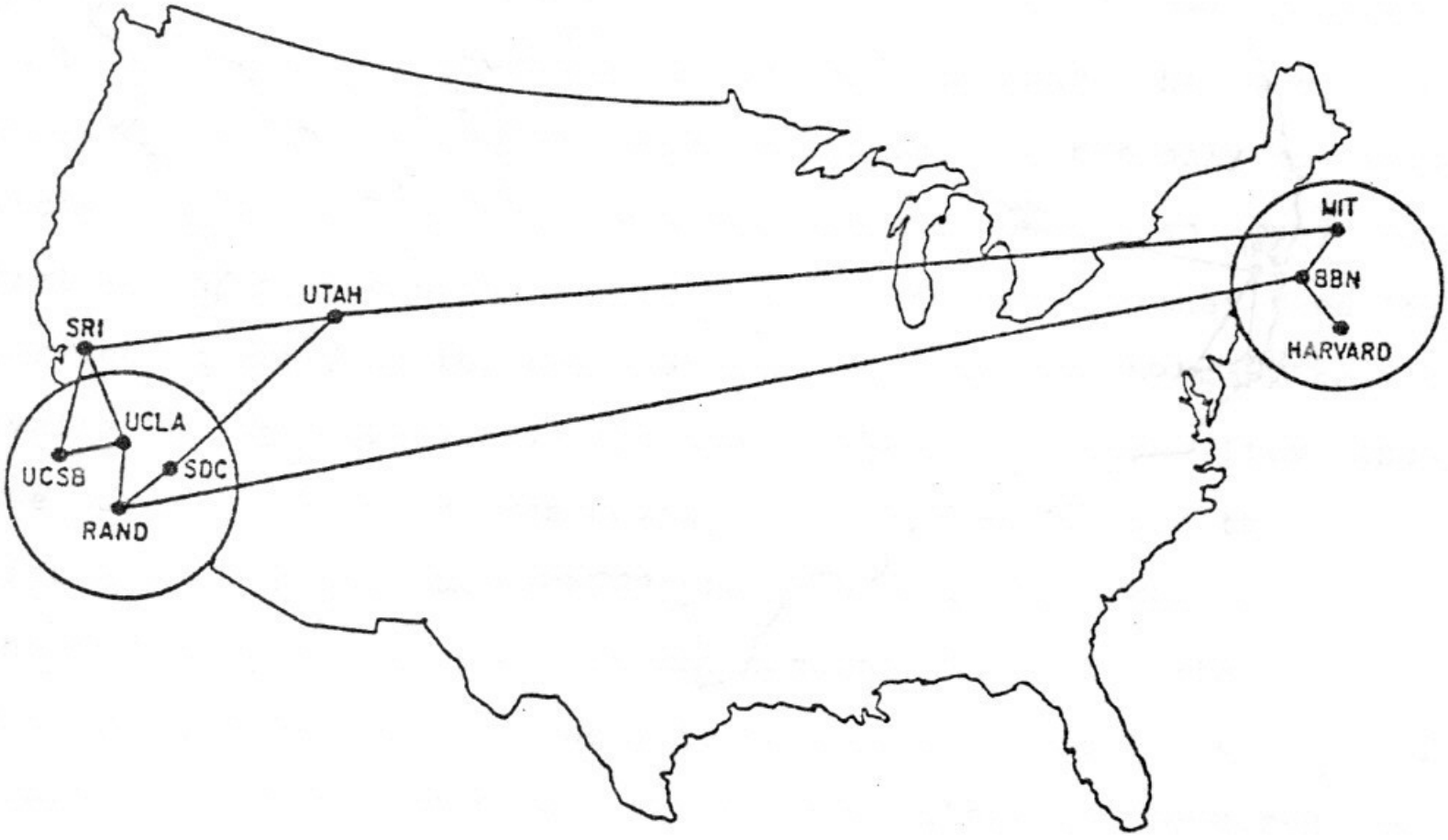
The Internet and World Wide Web

- The **Internet** is a *physical network* of cables and routers, and a set of protocols for moving information across that network.
- The **World Wide Web** (WWW) is an *information space* on the Internet. It combines several concepts:
  - Uniform Resource Locator (url)
  - Hypertext Transfer Protocol (http)
  - Hypertext Markup Language (html)

The Internet

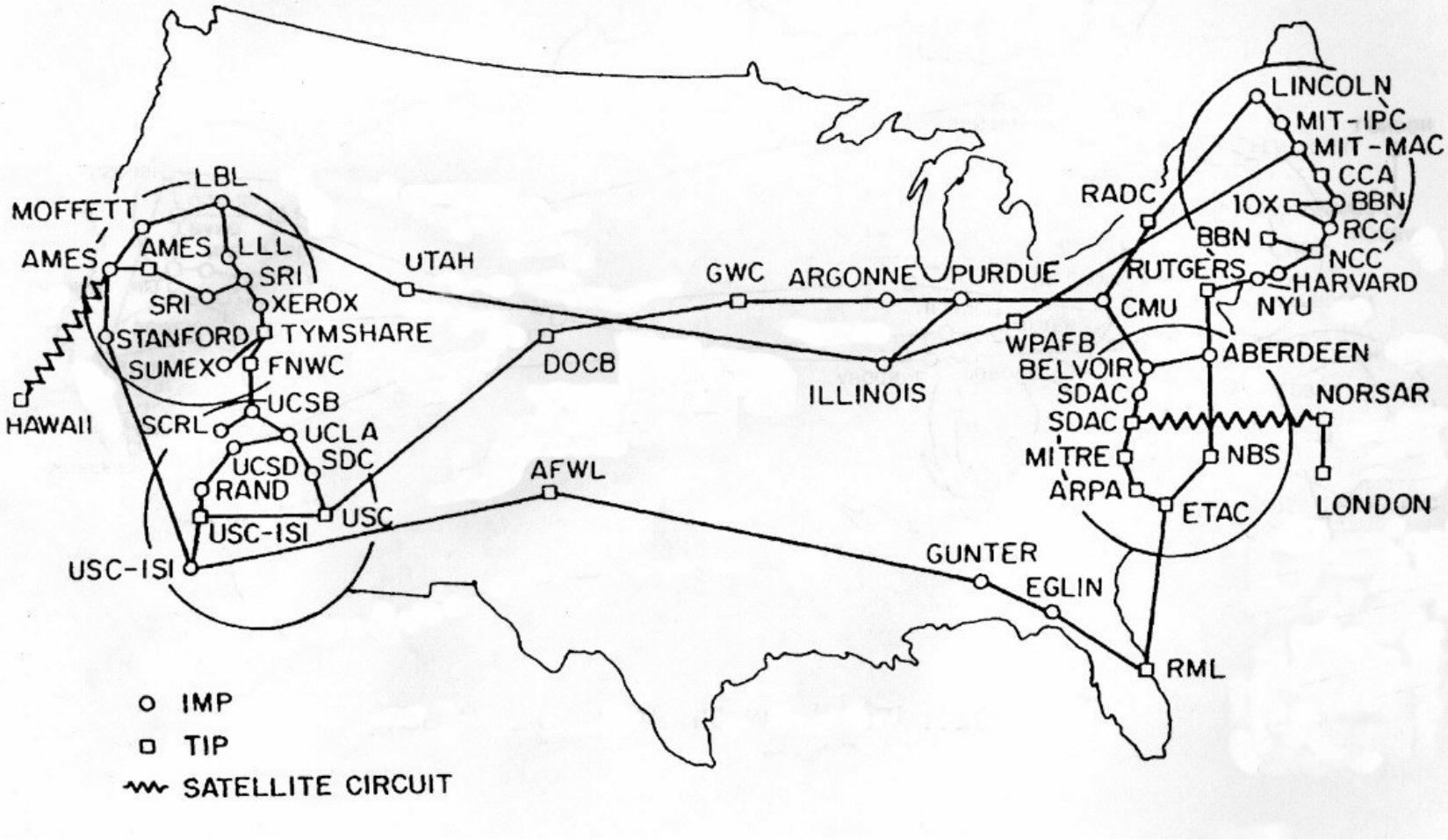
- Started as a Defense Advanced Research Projects Agency (DARPA) ARPANET project to interconnect computers
- First transmission between nodes at UCLA in 1969; by 1970 reached across the US to Boston.
- Transmission Protocol/Internet Protocol (TCP/IP) developed during the 1970s
- ARPANET declared "operational" in 1975 and transferred to military
- Ethernet (transmission on wires) standard written 1981
- During 1980s, shift to National Science Foundation
  - NSFNET provided the backbone of Internet from 1985 to 1995
- Internet backbone privatized under Clinton in 1994

The ARPANET in December 1969

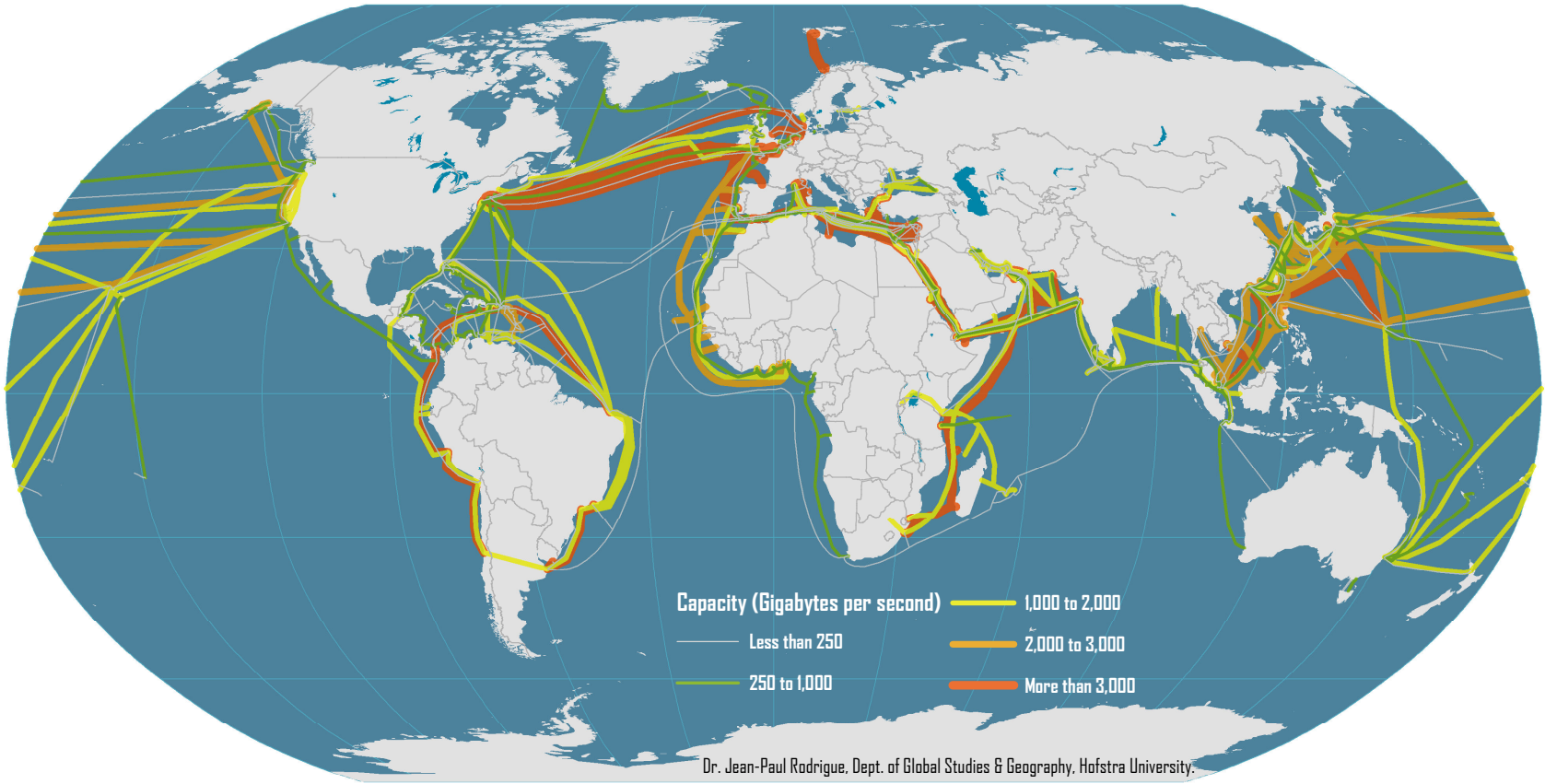

MIT
BBN
HARVARD
UTAH
SRI
UCLA
UCSB
SDC
RAND

ARPA NETWORK, GEOGRAPHIC MAP

JUNE 1975



Global Submarine Cable Network



Source: Dataset encoded by Greg Mahlknecht, http://www.cablemap.info

Uniform Resource Locator

URLs are a system of globally unique identifiers for resources on the Web and elsewhere.

`scheme://host[:port]/path[?query][#fragment]`

- `scheme` is a protcol such as `http`, `https`, etc.
- `host` is something like `google.com` or `localhost`
- `:port` is optional, allows a single host to have separate websites
- `path` is the path to a particular resource like `index.html`
- More on queries later

Hypertext Transfer Protocol (HTTP)

HTTP Overview

- HTTP is the core communications protocol for retrieving data
- Consists of messages– requests and responses– sent between a client and a server

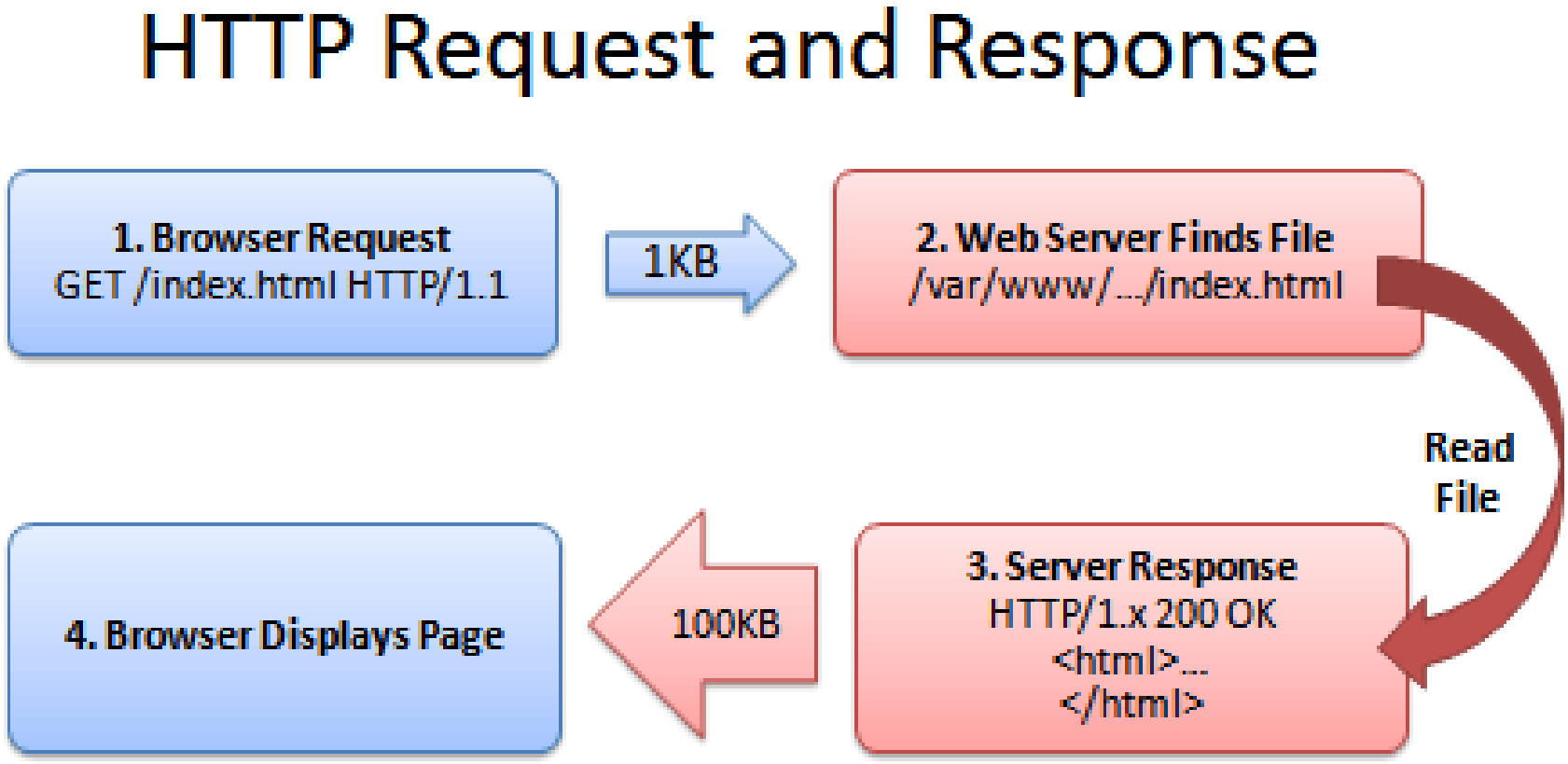


Figure 1: Source: opensourcevaristy.com

HTTP Request

```
GET /index.html HTTP/1.1
User-Agent: Mozilla/5.0
...
```

- First line contains:
  - HTTP method, here `GET`
  - Requested URL
  - HTTP version
- Rest of request may contain:
  - User-Agent: description of the client
    - Used e.g. to determine whether to serve mobile website version

HTTP Methods

- `GET`
  - Most common method, used to get data
- `POST`
  - Used to send data to server, e.g. form entries, search queries

HTTP Responses

```
HTTP/1.1 200 OK
...
```

- First line contains:
  - HTTP version
  - HTTP response code
- Rest of response contains:
  - Additional headers: `Server`, `Content-Type`, etc.
  - Requested Content

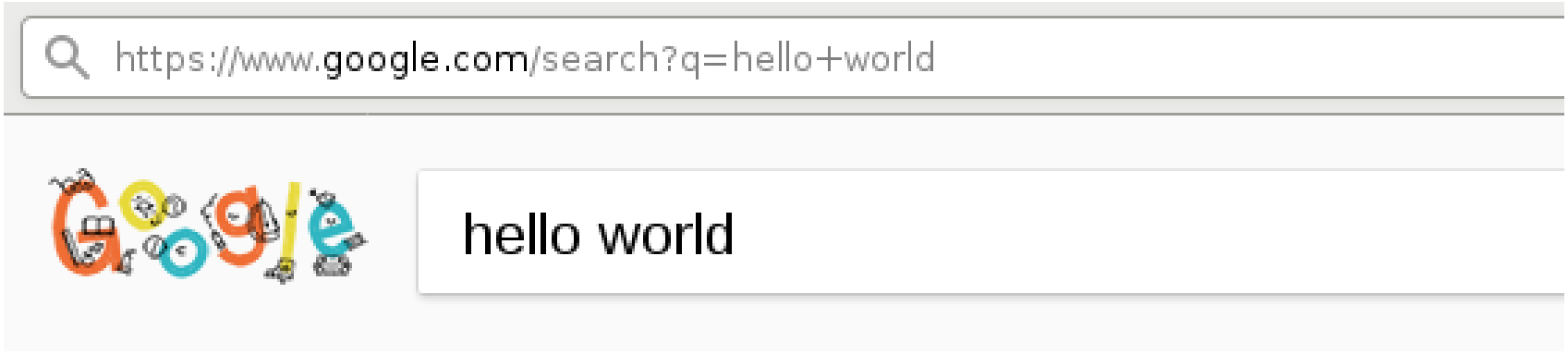
HTTP Response Codes

- 1xx: Informational
- 2xx: Success
  - 200: OK
- 3xx: Redirection
  - 301: Redirect
- 4xx: Errors
  - 404: File not found
  - 403: Forbidden

HTTP GET Request Parameters

`/index.php?name1=value1&name2=value2`

- Query string with parameters sent in the URL of a GET request
- Parameter names and values are like a python dictionary
- Shouldn't use with sensitive data



Hypertext Markup Language (HTML)

Web browser

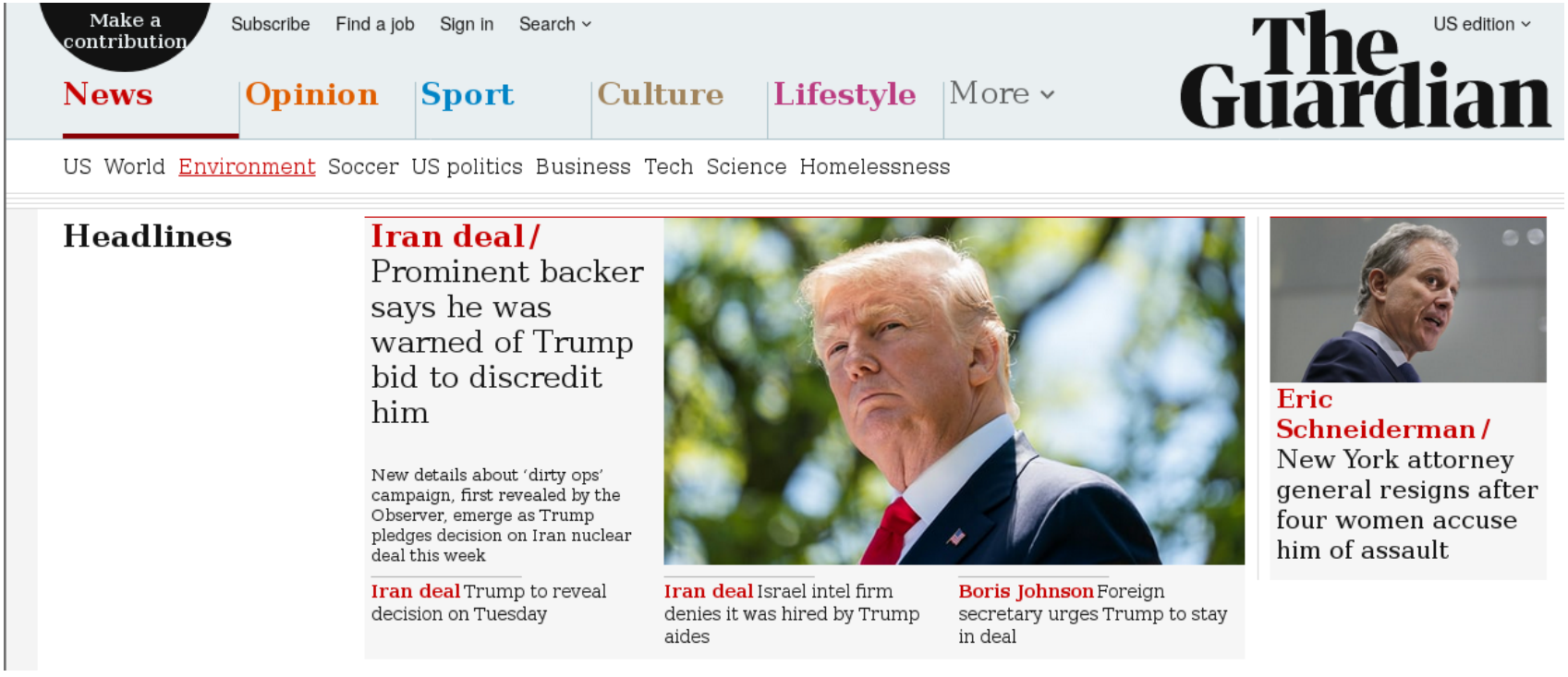


Figure 2: guardian.com rendered in a web browser

HTML sourcecode

```

<!DOCTYPE html>
<html id="js-context" class="js-off is-not-modern id--signed-out" lang="en" data-page-path="/international">
<head>
<!--

     __        __                   _     _      _
     \ \      / /__    __ _ _ __ ___  | |__ (_)_ __(_)_ __   __ _
      \ \ /\ / / _ \  / _` | '__/ _ \ | '_ \| | '__| | '_ \ / _` |
       \ V  V /  __/ | (_| | | |  __/ | | | | | |  | | | | | (_| |
        \_/\_/ \___|  \__,_|_|  \___| |_| |_|_|_|  |_|_| |_|\__, |
                                                            |___/
    Ever thought about joining us?
    https://workforus.theguardian.com/careers/digital-development/
     --->
<title>News, sport and opinion from the Guardian's global edition | The Guardian</title>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=Edge"/>
<meta name="format-detection" content="telephone=no"/>
<meta name="HandheldFriendly" content="True"/>
<meta name="viewport" content="width=device-width,minimum-scale=1,initial-scale=1">
<link rel="dns-prefetch" href="https://assets.guim.co.uk/"/>
<link rel="dns-prefetch" href="https://i.guim.co.uk"/>
<link rel="dns-prefetch" href="https://api.nextgen.guardianapps.co.uk"/>
<link rel="dns-prefetch" href="https://hits-secure.theguardian.com"/>
<link rel="dns-prefetch" href="//j.ophan.co.uk"/>
<link rel="dns-prefetch" href="//ophan.theguardian.com"/>
<link rel="dns-prefetch" href="//beacon.gu-web.net"/>
<link rel="dns-prefetch" href="//www.google-analytics.com"/>
<link rel="dns-prefetch" href="//sb.scorecardresearch.com"/>
```

Figure 3: guardian.com HTML sourcecode

HTML overview

- Language that webpages are written in
- Consists of *tags*
- Most tags come in pairs (opening and closing):
  - `<html></html>`
  - `<head></head>`
  - `<body></body>`
  - `<a></a>`
- Some do not:
  - `<img>`
  - `<br>`
- Whitespace doesn't matter (unlike Python)

Basic HTML webpage

```
<html>

<head>
    <title>My Title</title>
</head>

<body>
<h1>My Site</h1>
<p>
More information <a href="main.html">here</a>.
</p>

</body>
</html>
```

Basic webpage

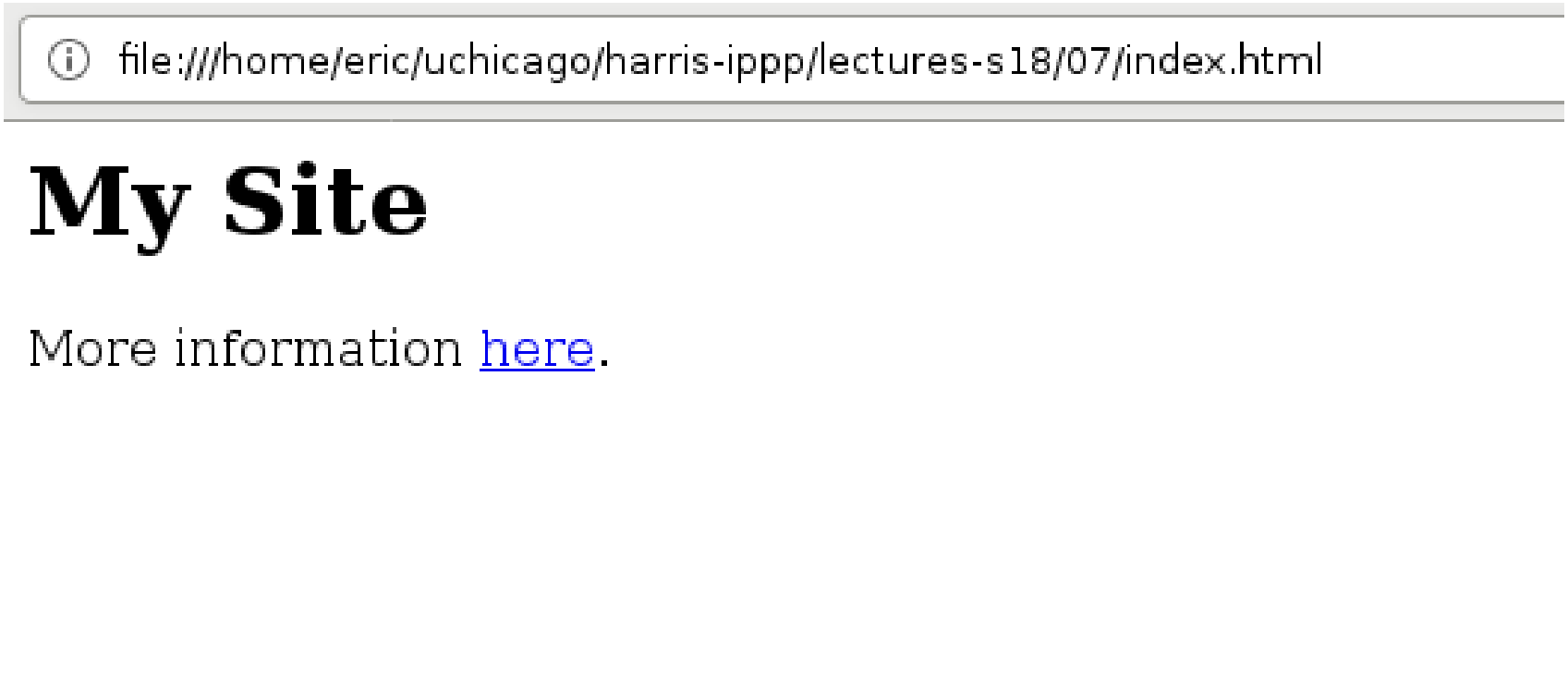


Figure 4: Webpage rendered in browser

HTML links

```
<a href="main.html">here</a>
```

- `a` is the tag
- `here` is the content
- `href` is an attribute whose value is `main.html`
  - attributes and their values form something like a dictionary

More HTML tags

- `<title>`: page title
- `<body>`: page body
- `<h1>`: largest header
- `<h6>`: smallest header
- `<p>`, `<br>`: paragraph, break
- `<em>`, `<b>`: emphasis (italics), bold
- `<ol>`, `<ul>`, `<li>`: (Un-)ordered list, list element
- `<a href="">`: Hyperlink
- `<table>`, `<tr>`, `<th>`, `<td>`: table row, header, and cell
- `<img src="">`: image
- `<div>`: block

HTML table

```
<table>
  <tr>
    <td>Header 1</td>
    <td>Header 2</td>
  </tr>
  <tr>
    <td>Data 1</td>
    <td>Data 2</td>
  </tr>
</table>
```

Other web technologies

- Javascript
  - `.js` files sourced or code directly embedded in `<script>...</script>` tags
  - Used to make content dynamic
- CSS
  - `.css` files or code embedded in `<style></style>` tags
- Server-side languages
  - Ruby, PHP, Python (Django), Java, etc.

Web Scraping

Web scraping overview

- Scraping is the process of programmatically extracting information from websites
- Anything that you can view in a web browser can *potentially* be scraped

Why scrape?

Some websites offer services (APIs) that allow you to get data directly. So why scrape?

- Not all websites provide an API
- Not all of a website's content is available through its API
- APIs often use tokens to limit the amount of data that can be requested
  - With scraping there is, in principle, no limit

When to scrape?

Scraping is good for:

- Downloading all `.mp3` files linked from a site
- Constructing a pandas DataFrame from a table on Wikipedia
- Parsing articles from a small news website

Scraping is not ideal for:

- Extracting information from your emails (python has POP/IMAP libraries)
- Analyzing tweets (Twitter provides an API)

Ethical considerations

- Credit all sources
  - Publishing scraped content can be a copyright violation
- Don't overload websites
  - Most sites will block you before you can do this
- Obey `robots.txt`
  - Most sites have a file describing which areas of the site are prohibited from being scraped
- You are not anonymous on the web
  - Unless you take explicit steps (VPN, Tor, etc.) to do so