2110446 - Data Science and Data Engineering

Data Extraction

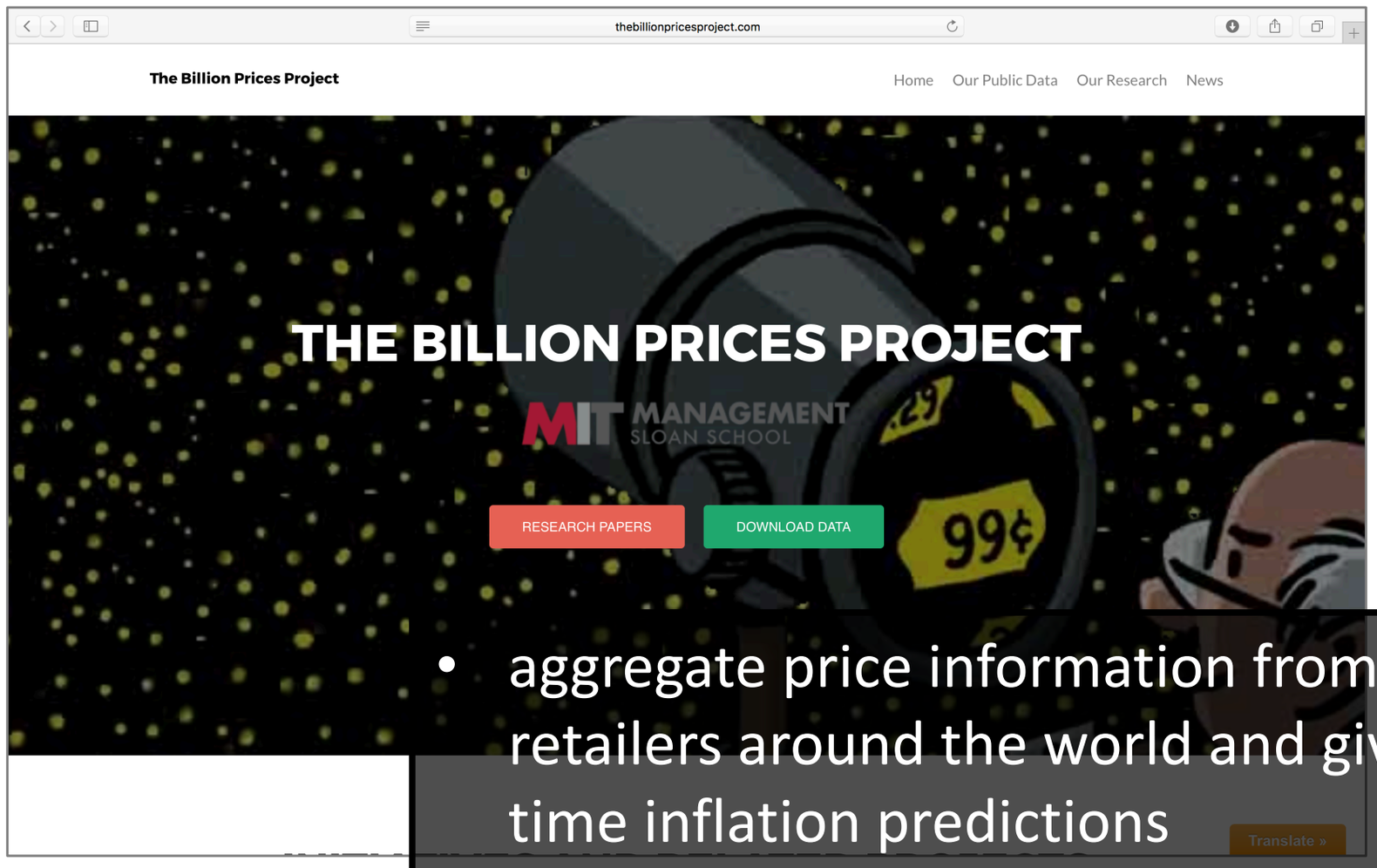
Asst.Prof. Natawut Nupairoj, Ph.D.
Department of Computer Engineering
Chulalongkorn University
natawut.n@chula.ac.th

CHULA ENGINEERING
Foundation toward Innovation

Internet Data Sources

- Internet provides lots of data sources
  - Financial
  - Weather
  - Sports
  - News
- Google creates business by utilizing Internet information
- Most data sources are human-readable, it is also possible to extract information systematically with program

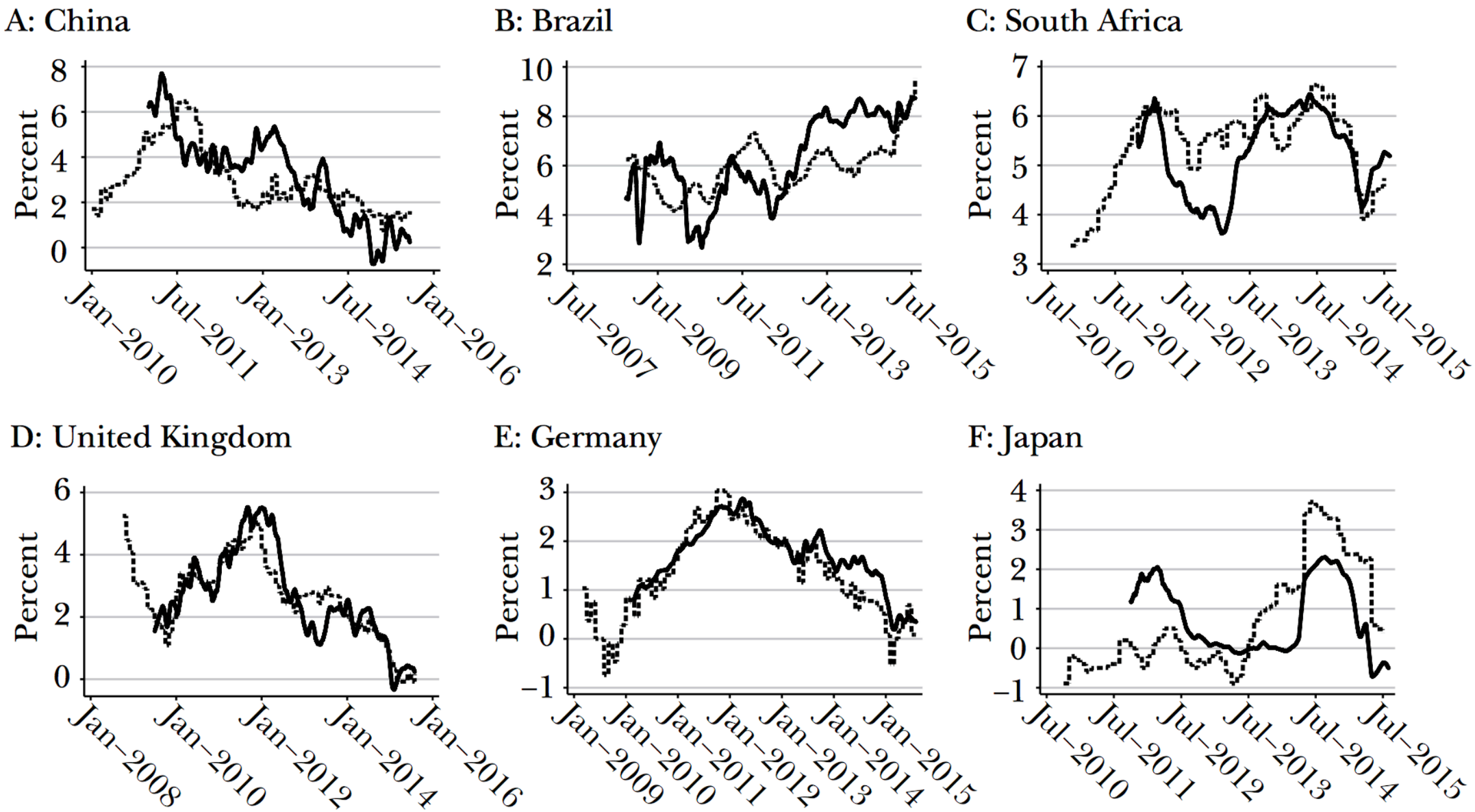
MIT's Billion Prices Project



- aggregate price information from multitude of online retailers around the world and gives real time inflation predictions
- monitor daily price fluctuations of ~5 million items sold by ~300 online retailers in more than 70 countries

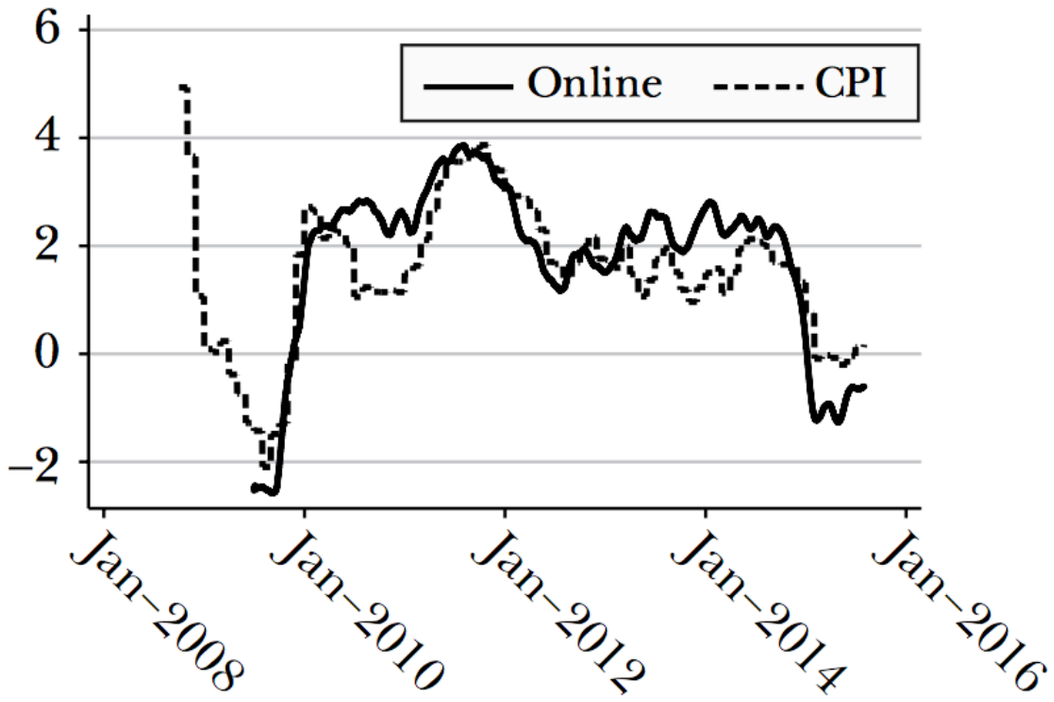
Cavallo, A., & Rigobon, R. (2016). The billion prices project: Using online prices for measurement and research. *The Journal of Economic Perspectives*, *30*(2), 151-178.

Annual Inflation Rate: Online vs. CPI

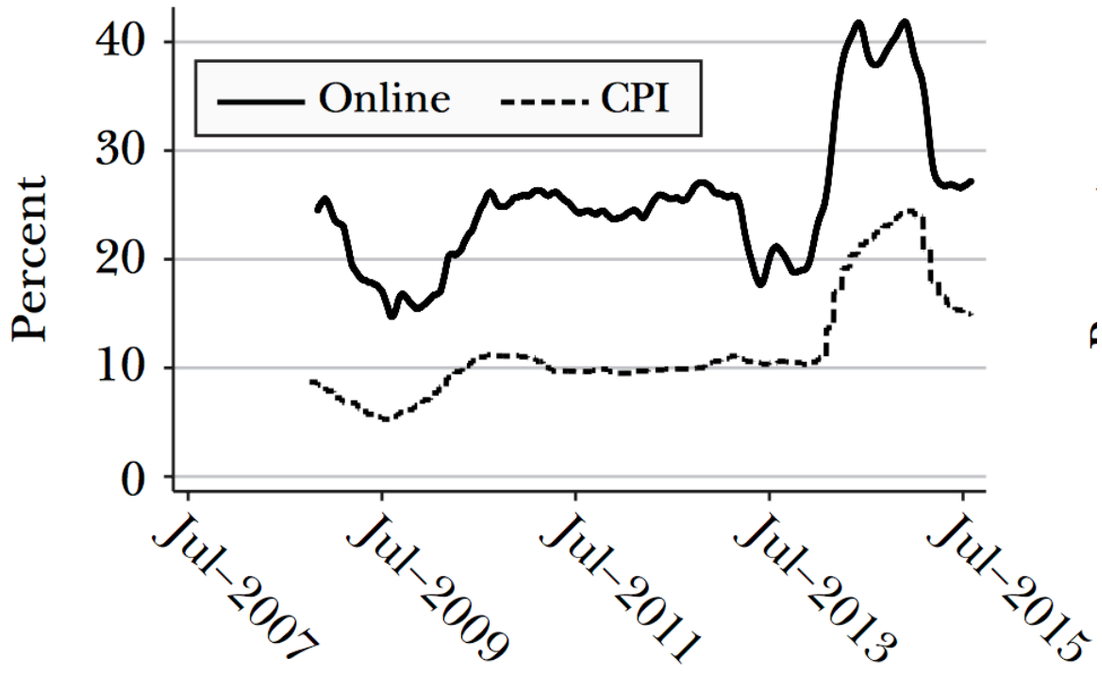


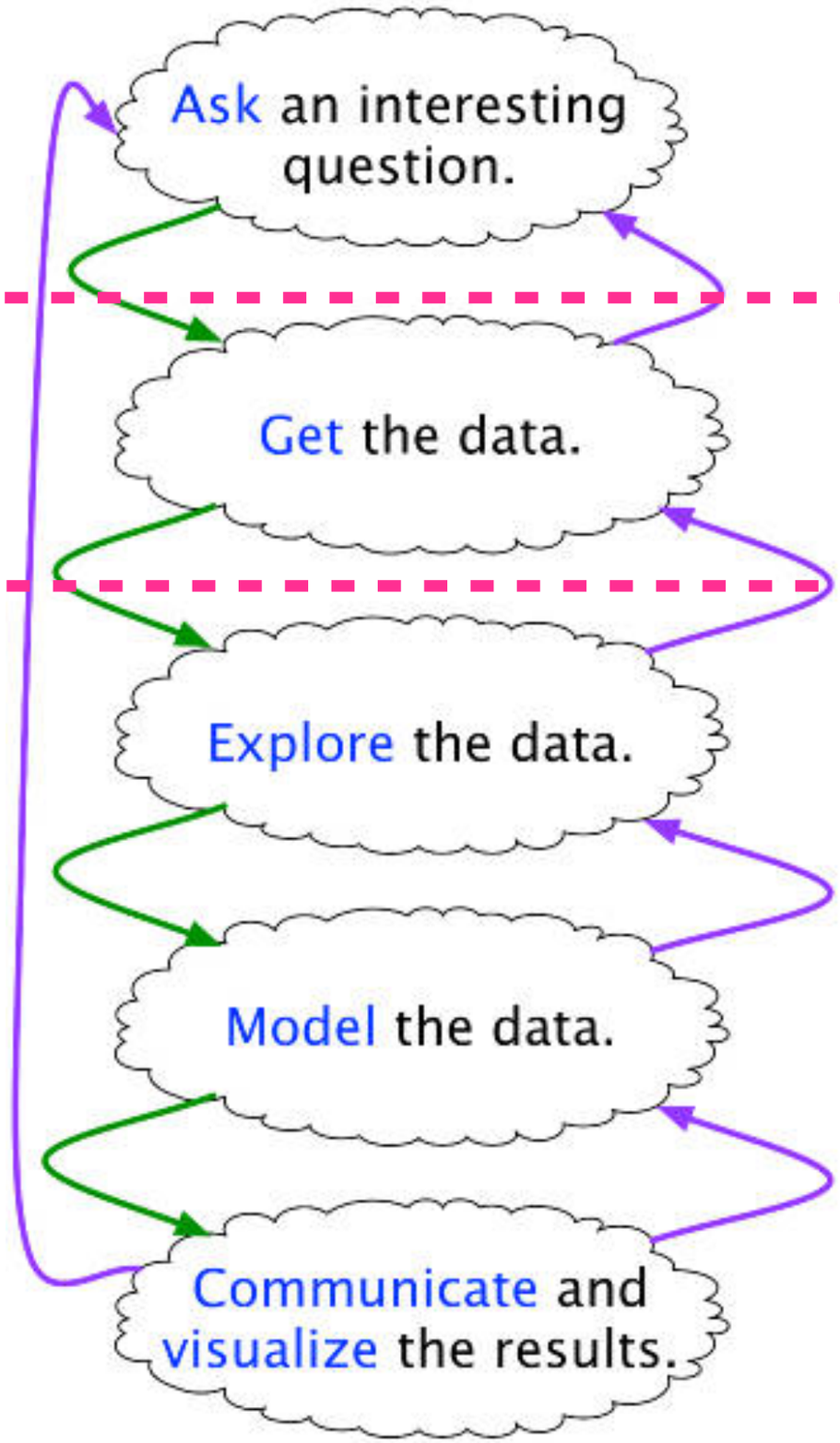
Annual Inflation Rate: Online vs. CPI

USA



Argentina





Scrapy
Requests
BeautifulSoup
Selenium

pandas
statsmodels
NLTK
matplotlib
seaborn
folium
Spark

scikit-learn
Theano
Tensorflow
SparkML

Seaborn
Bokeh
Basemap
NetworkX

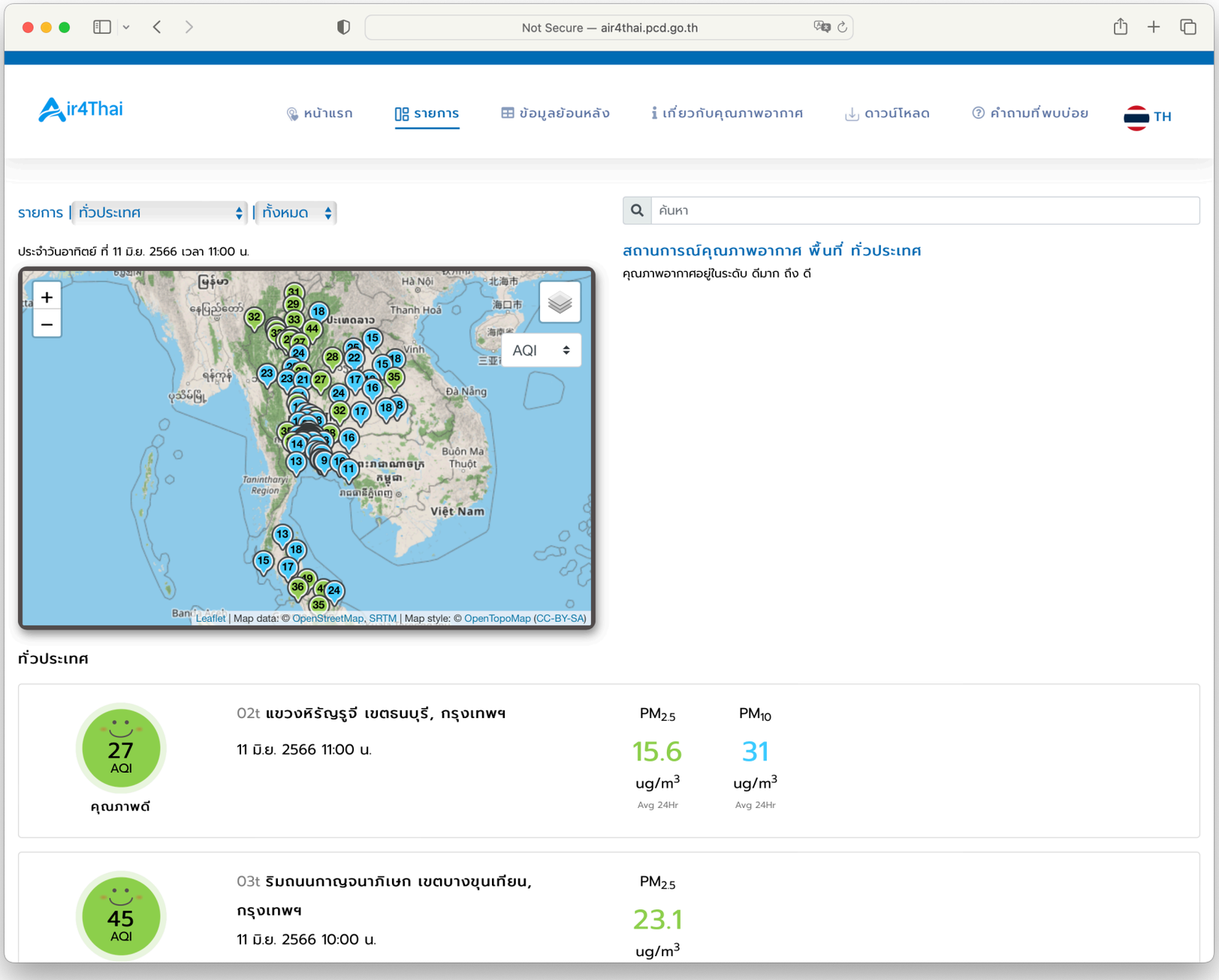
Data Extraction

- Data extraction is the process of getting data from a source for further data processing, storage or analysis elsewhere
- Many data sources are in unstructured or human-readable format
  - Gather organization structure from company's website
  - Extract accounting information from PDF financial reports
  - Collect relationship information from social media platform

CHULA ENGINEERING
Foundation toward Innovation

Example Data Source: Air4Thai

http://air4thai.pcd.go.th/webV3/#/Report



Data Extraction Steps

- Data Accessing
  - Reading a file
  - Issue a HTTP request
  - Call REST API using Request
  - Call Third-Party API using compatible library
  - Collect complicated web page with Selenium

Data Extraction Steps

- Data Parsing
  - Parsing HTML page using BeautifulSoup
  - Parsing HTML page using GenAI (Simple but Expensive)
  - Parsing text using regular expression
  - Parsing docx using python-docx
  - Parsing excel using Pandas
  - Parsing JSON string using JSON

Data Extraction Steps

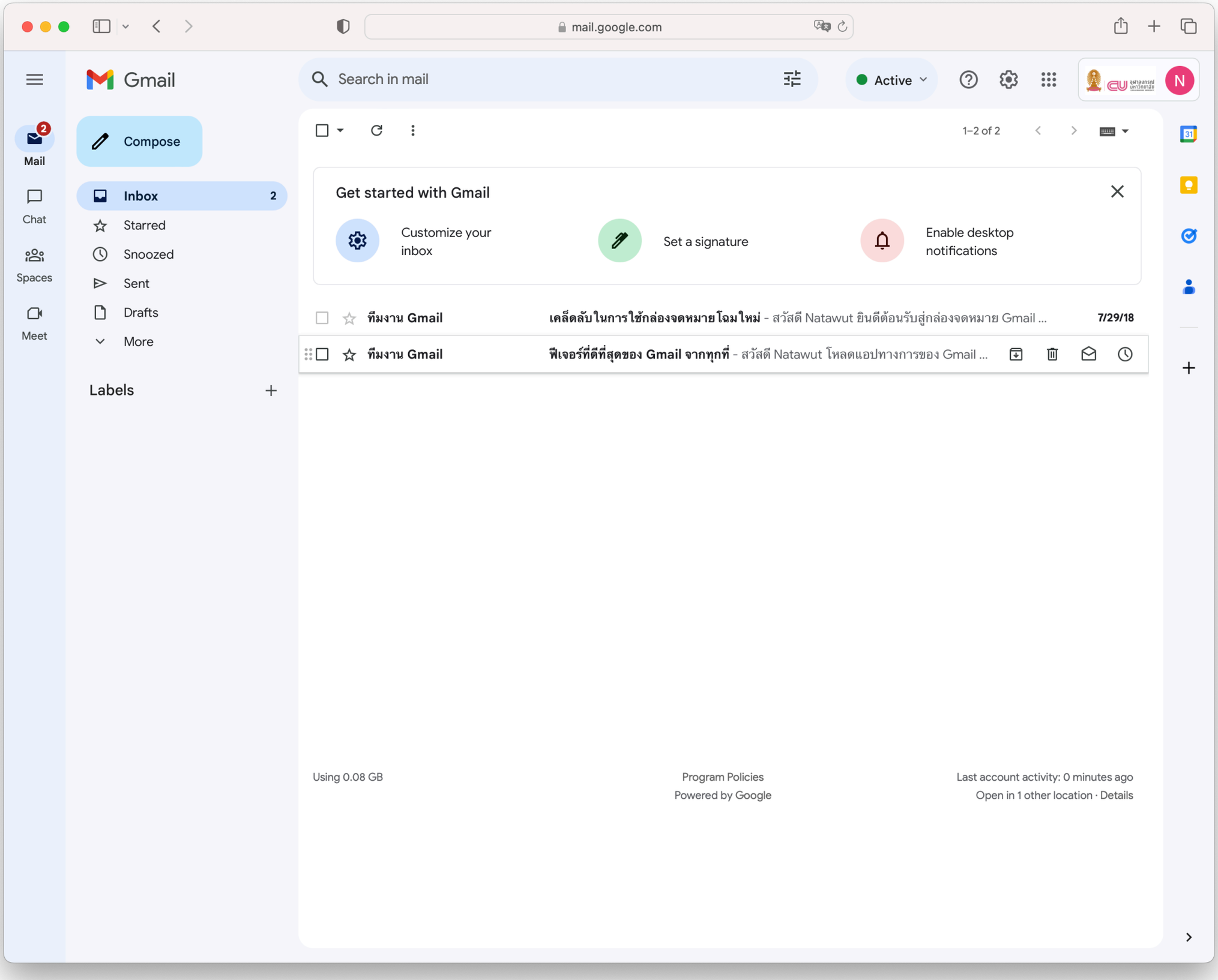
- Data Loading
  - Save to database
  - Save to data file
  - Save to dataframe

CHULA ENGINEERING
Foundation toward Innovation

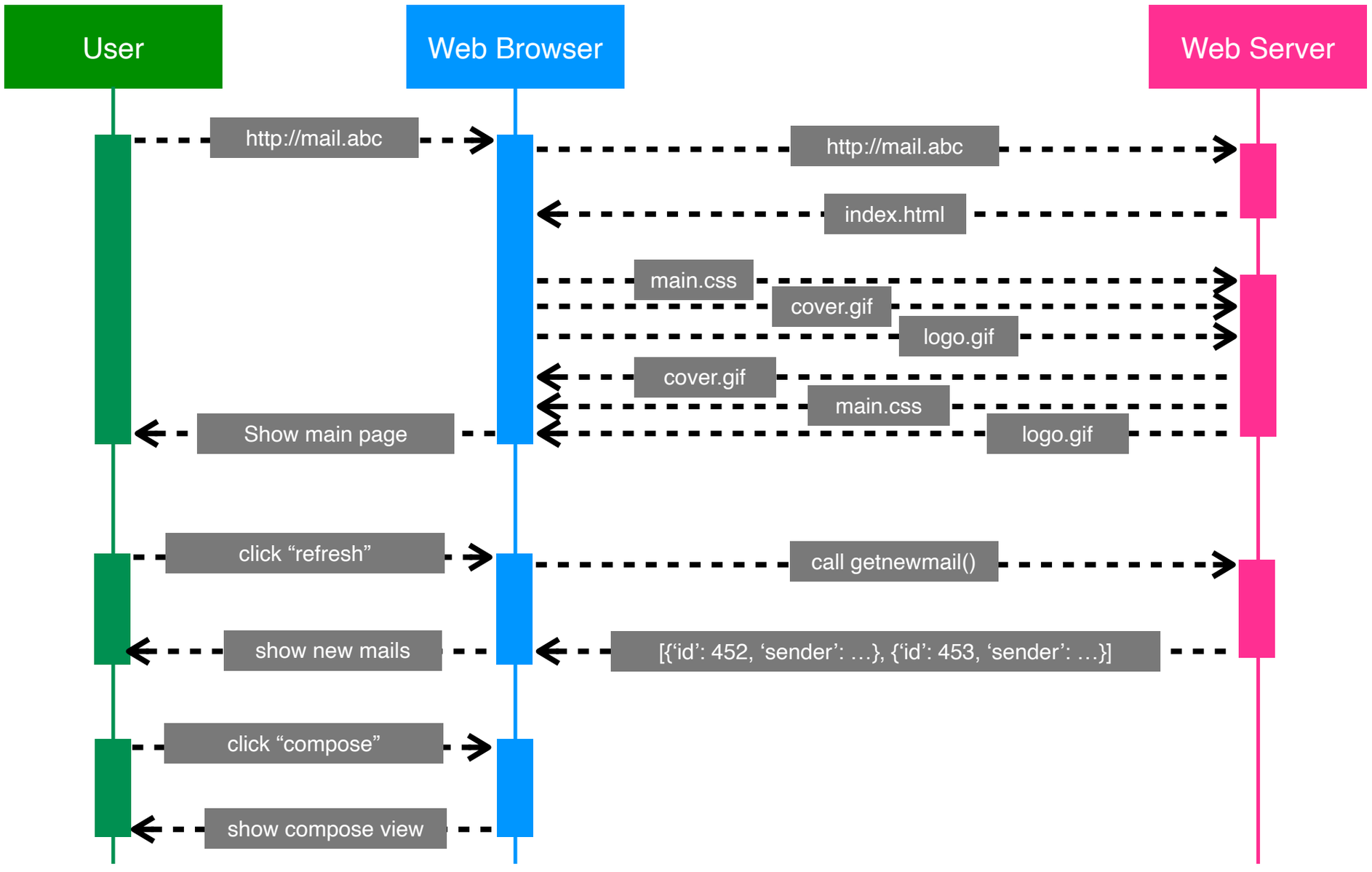
Scraping a Web Page

Understand HTML and how HTTP works

CHULA ENGINEERING
Foundation toward Innovation



Understand Web Page Mechanism



HTML

- HTML is a markup language that defines the structure of a web page using series of elements to wrap different parts of the content to make it appear a certain way, or act a certain way
- For example, <p> is a tag to indicate a paragraph

```
<p>Hello, my name is Natawut Nupairoj.  Nice to meet you!</p>
```

- Furthermore, HTML element can be nested

```
<p>Hello, my name is <b>Natawut Nupairoj</b>.  Nice to meet you!</p>
```

- A web browser will show this snippet as followed

Hello, my name is **Natawut Nupairoj**. Nice to meet you!

More on HTML Element

- HTML element can have an attribute to provide extra information regarding to this element

```
<p class="greeting">Hello, my name is <b>Natawut Nupairoj</b>.  Nice to meet you!</p>
```

Note that "class" is an attribute that can be used in conjunction with CSS

- Some elements have no content and are called void elements

```
<img src="images/firefox-icon.png" alt="My test image" />
```

Anatomy of HTML Page (my_page.html)

```
<!DOCTYPE html>
<html lang="en-US">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width" />
    <title>My test page</title>
  </head>
  <body>
    <p class="greeting">Hello, my name is <b>Natawut Nupairoj</b>.  Nice to meet you!
</p>
    <img src="http://mis.cp.eng.chula.ac.th/view.php?q=instructor/
picture&v=badge&key=00041200" alt="My test image" />
  </body>
</html>
```

Greeting

Hello, my name is **Natawut Nupairoj**. Nice to meet you!

CSS

- CSS (Cascading Style Sheets) is a declarative language that controls how webpages look in the browser
  - Font, size, color, background color, spacing
  - Alignment, number of columns
- The browser applies CSS style declarations to selected elements to display them properly
- A style declaration contains the properties and their values, which determine how a webpage looks

CSS Rules and Selectors

- A CSS rule is a set of properties associated with a selector
- Here is an example that makes every HTML paragraph yellow against a black background:

```
/* The selector "p" indicates that all paragraphs in the document will be
affected by that rule */
p {
  /* The "color" property defines the text color, in this case yellow. */
  color: yellow;

  /* The "background-color" property defines the background color, in this case
black. */
  background-color: black;
}
```

- "Cascading" refers to the rules that govern how selectors are prioritized to change a page's appearance

my_page_with_css.html

```
<!DOCTYPE html>
<html lang="en-US">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width" />
    <style>
      p {
        color: yellow;
        background-color: black;
      }
    </style>
    <title>My test page</title>
  </head>
  <body>
    <p class="greeting">Hello, my name is <b>Natawut Nupairoj</b>.  Nice to meet you!</p>
    <img src="http://mis.cp.eng.chula.ac.th/view.php?q=instructor/
picture&v=badge&key=00041200" alt="My test image" />
  </body>
</html>
```

Note that CSS can be embedded in an HTML page or in a separated file

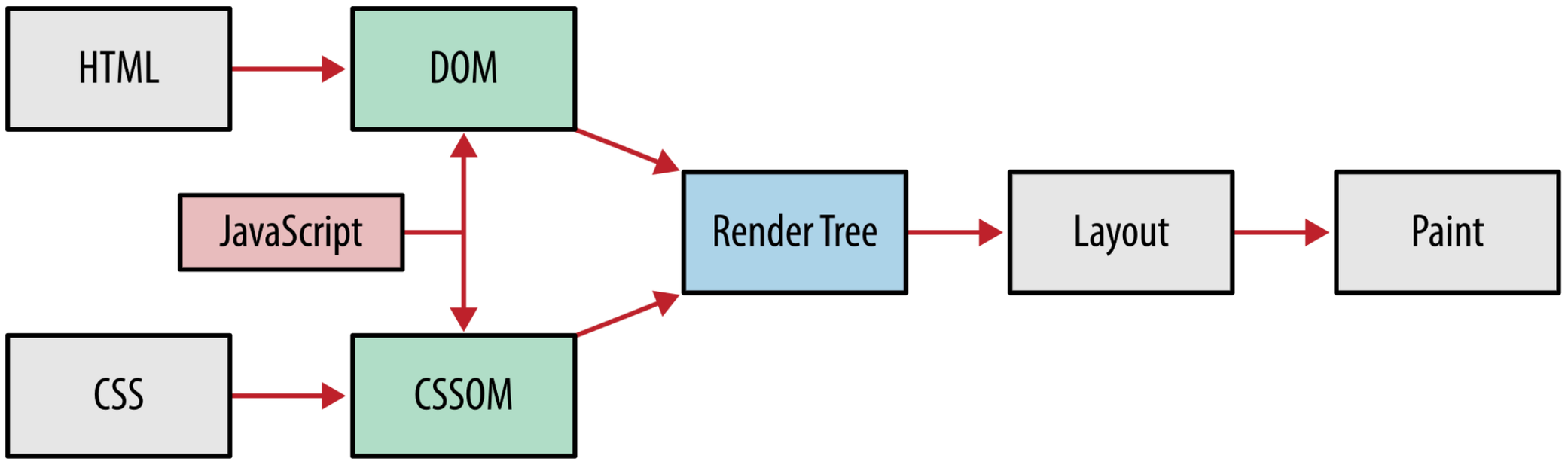
Greeting

Hello, my name is **Natawut Nupairoj**. Nice to meet you!

CSS Selector

- CSS Selector is a pattern of elements and other terms that tell the browser which HTML elements should be selected to have the CSS property values inside the rule applied to them
  - Elements with a specific tag e.g. all H1 elements
  - Elements with a specific class / id / attribute e.g. all elements with class = “special”
  - Combination of selectors e.g.
    - all H1 elements with class = “special”
    - all H1 elements or elements with class = “special”
- A selector may select multiple elements at once

Understand Web Page Mechanism



Source: https://hpbn.co/primer-on-web-performance/

Run ›

Result Size: 426 x 495

Get your own website

```
<!DOCTYPE html>
<html>
<body>

<p>Click the button to display the domain name of the server
that loaded this document.</p>

<button onclick="myFunction()">Try it</button>

<p id="demo"></p>

<script>
function myFunction() {
  document.getElementById("demo").innerHTML =
document.domain;
}
</script>

</body>
</html>
```

Click the button to display the domain name of the server that loaded this document.

Try it

www.w3schools.com

https://www.w3schools.com/js/tryit.asp?filename=tryjs_doc_domain

CHULA ENGINEERING
Foundation toward Innovation

Parsing A Simple Webpage

- Applicable for a simple page only
  - No content rendered by javascript
  - Parse HTML with BeautifulSoup
- BeautifulSoup allows
  - Parsing HTML
  - Accessing and Navigating the DOM tree
  - Getting information from nodes

mentation »

Beautiful Soup Documentation

Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. It commonly saves programmers hours or days of work.

These instructions illustrate all major features of Beautiful Soup 4, with examples. I show you what the library is good for, how it works, how to use it, how to make it do what you want, and what to do when it violates your expectations.

This document covers Beautiful Soup version 4.9.1. The examples in this documentation should work the same way in Python 2.7 and Python 3.2.

You might be looking for the documentation for Beautiful Soup 3. If so, you should know that Beautiful Soup 3 is no longer being developed and that support for it will December 31, 2020. If you want to learn about the differences between Beautiful Soup 3 see Porting code to BS4.

This documentation has been translated into other languages by Beautiful Soup users:

- 这篇文档当然还有中文版.
- このページは日本語で利用できます(外部リンク)
- 이 문서는 한국어 번역도 가능합니다.
- Este documento também está disponível em Português do Brasil.
- Эта документация доступна на русском языке.

Getting help

If you have questions about Beautiful Soup, or run into problems, send mail to the discussi
lem involves parsing an HTML document, be sure to mention what the diagnose() fun
document.

Quick Start

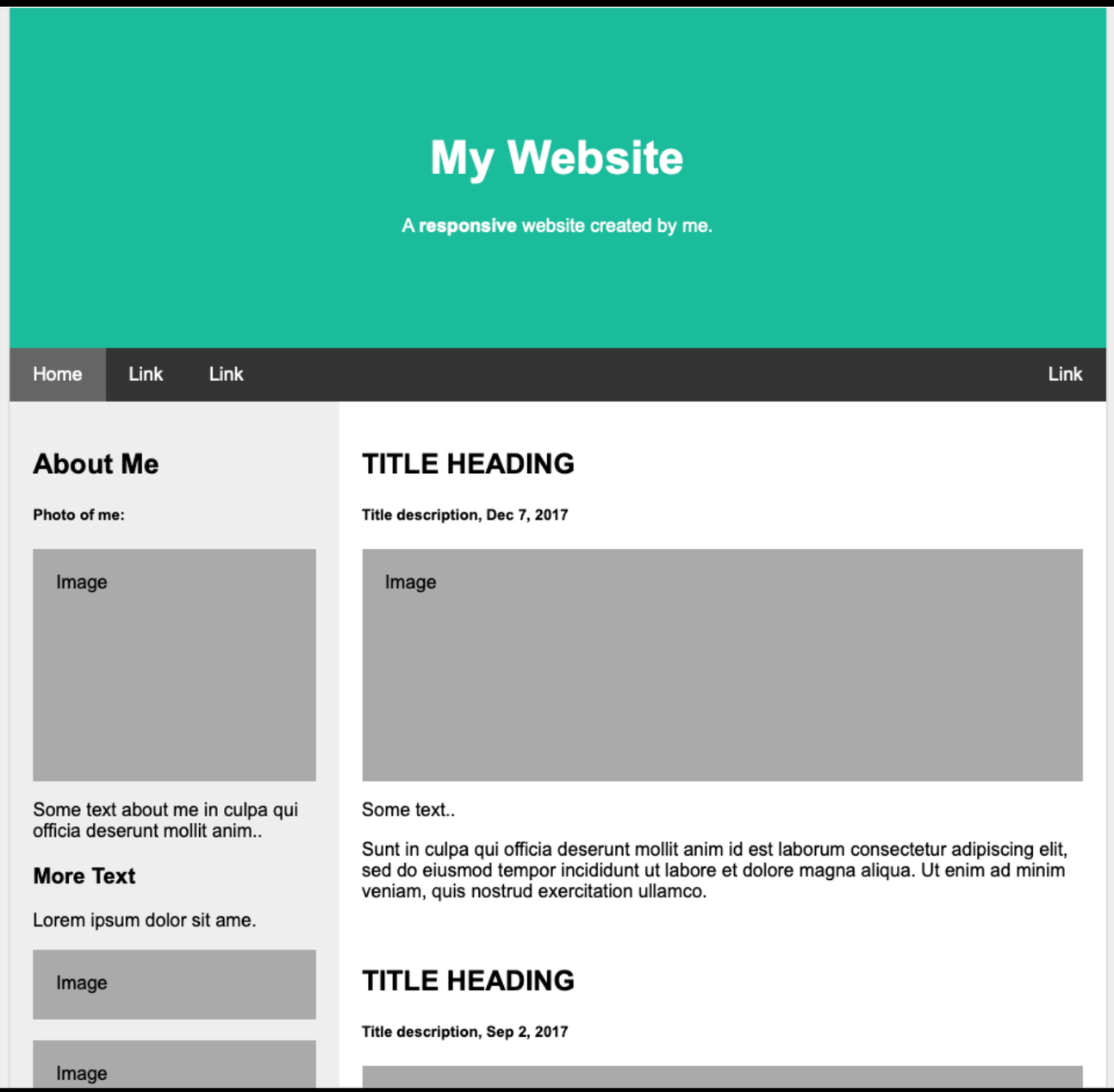
Here's an HTML document I'll be using as an example throughout this document. It's part
Wonderland:

```
html_doc = """
<html><head><title>The Dormouse's story</title></head>
<body>
<p class="title"><b>The Dormouse's story</b></p>

<p class="story">Once upon a time there were three little sisters; and their names
<a href="http://example.com/elsie" class="sister" id="link1">Elsie</a>,
<a href="http://example.com/lacie" class="sister" id="link2">Lacie</a> and
<a href="http://example.com/tillie" class="sister" id="link3">Tillie</a>;
and they lived at the bottom of a well.</p>

<p class="story">...</p>
"""
```

Running the "three sisters" document through Beautiful Soup gives us a `BeautifulSoup` ob
the document as a nested data structure:

https://www.w3schools.com/howto/tryit.asp?filename=tryhow_css_example_website

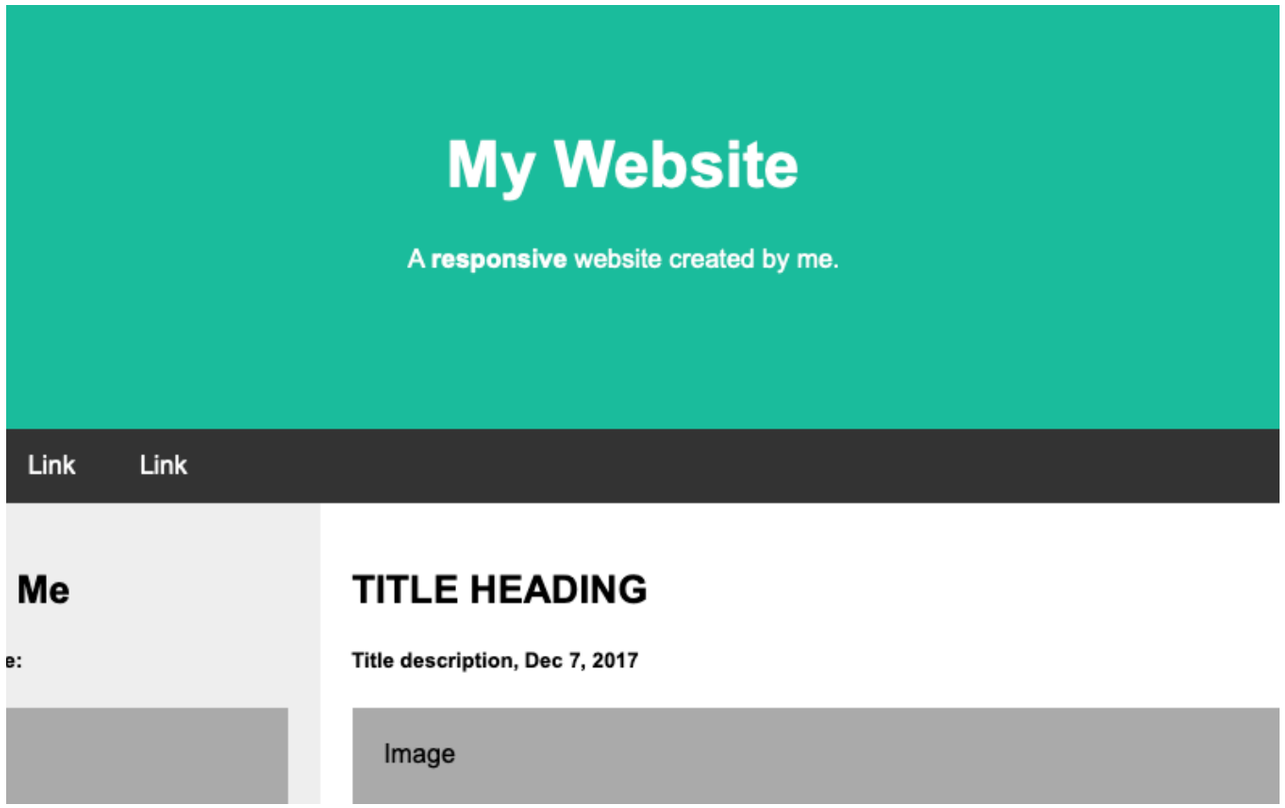
```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Page Title</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
* {
  box-sizing: border-box;
}

/* Style the body */
body {
  font-family: Arial, Helvetica, sans-serif;
  margin: 0;
}

/* Header/logo Title */
.header {
  padding: 80px;
  text-align: center;
  background: #1abc9c;
  color: white;
}

/* Increase the font size of the heading */
.header h1 {
  font-size: 40px;
}

/* Sticky navbar - toggles between relative and fixed, depending on the scroll position. It is
positioned relative until a given offset position is met in the viewport - then it "sticks" in place
(like position:fixed). The sticky value is not supported in IE or Edge 15 and earlier versions.
However, for these versions the navbar will inherit default position */
.navbar {
  overflow: hidden;
  background-color: #333;
  position: sticky;
  position: -webkit-sticky;
  top: 0;
}

/* Style the navigation bar links */
.navbar a {
  float: left;
  display: block;
  color: white;
  text-align: center;
  padding: 14px 20px;
  text-decoration: none;
}
```



simple_page.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Page Title</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
* {
  box-sizing: border-box;
}

/* Style the body */
body {
  font-family: Arial, Helvetica, sans-serif;
  margin: 0;
}

/* Header/logo Title */
.header {
  padding: 80px;
  text-align: center;
  background: #1abc9c;
  color: white;
}

/* Increase the font size of the heading */
.header h1 {
  font-size: 40px;
}

/* Sticky navbar - toggles between relative and fixed, depending on the scroll position. It is positioned relative until
a given offset position is met in the viewport - then it "sticks" in place (like position:fixed). The sticky value is
not supported in IE or Edge 15 and earlier versions. However, for these versions the navbar will inherit default
position */
.navbar {
  overflow: hidden;
  background-color: #333;
  position: sticky;
  position: -webkit-sticky;
  top: 0;
}
```

```
<body>

<div class="header">
  <h1>My Website</h1>
  <p>A <b>responsive</b> website created by me.</p>
</div>

<div class="navbar">
  <a href="#" class="active">Home</a>
  <a href="#">Link</a>
  <a href="#">Link</a>
  <a href="#" class="right">Link</a>
</div>

<div class="row">
  <div class="side">
    <h2>About Me</h2>
    <h5>Photo of me:</h5>
    <div class="fakeimg" style="height:200px;">Image</div>
    <p>Some text about me in culpa qui officia deserunt mollit anim..</p>
    <h3>More Text</h3>
    <p>Lorem ipsum dolor sit ame.</p>
    <div class="fakeimg" style="height:60px;">Image</div><br>
    <div class="fakeimg" style="height:60px;">Image</div>
  </div>
  <div class="main">
    <h2>TITLE HEADING</h2>
    <h5>Title description, Dec 7, 2017</h5>
    <div class="fakeimg" style="height:200px;">Image</div>
    <p>Some text..</p>
    <p>Sunt in culpa qui officia deserunt mollit anim id est laborum consectetur adipiscing elit, sed
do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
exercitation ullamco.</p>
    <br>
    <h2>TITLE HEADING</h2>
    <h5>Title description, Sep 2, 2017</h5>
    <div class="fakeimg" style="height:200px;">Image</div>
    <p>Some text..</p>
    <p>Sunt in culpa qui officia deserunt mollit anim id est laborum consectetur adipiscing elit, sed
do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
exercitation ullamco.</p>
  </div>
</div>

<div class="footer">
  <h2>Footer</h2>
</div>

</body>
```

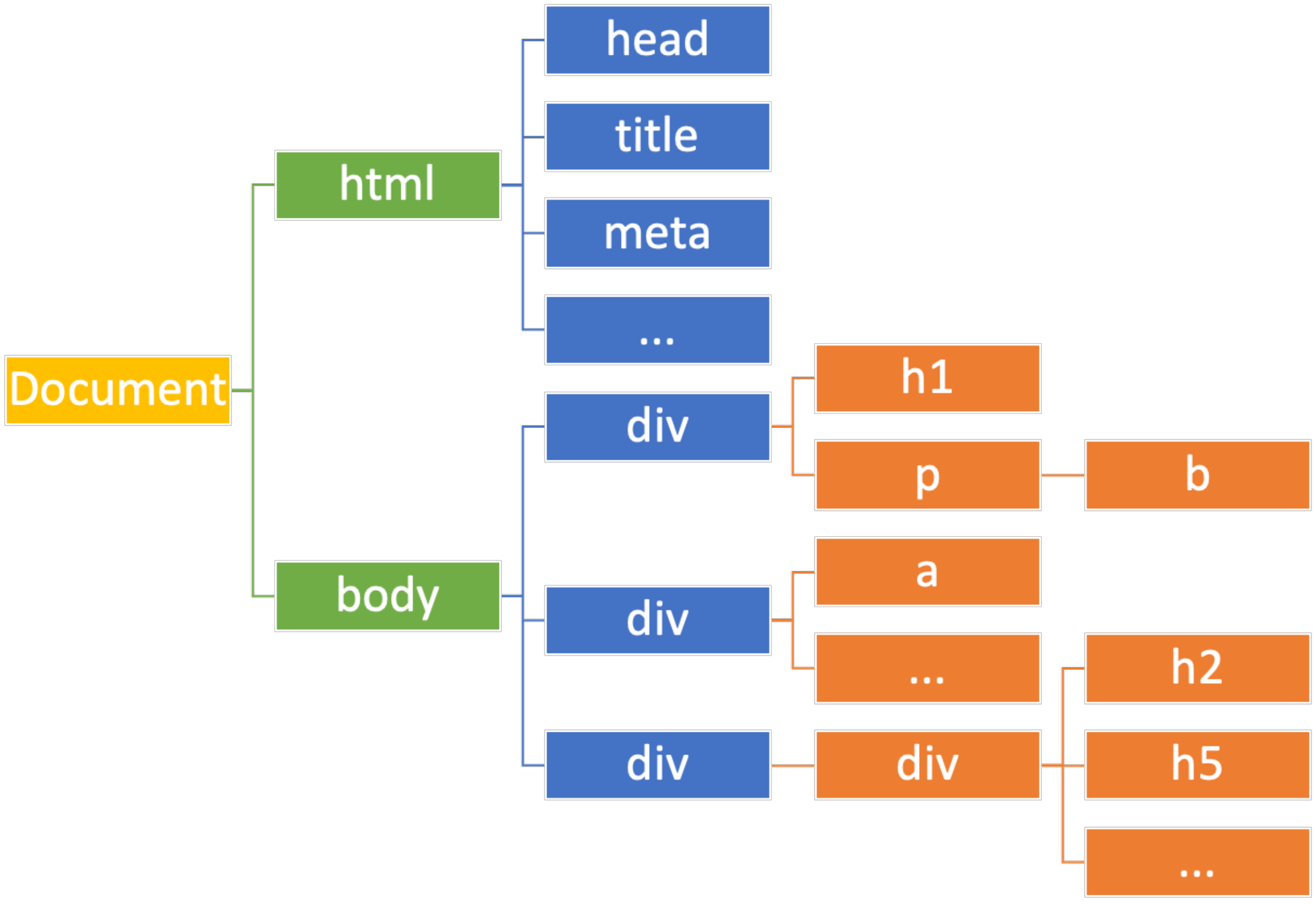


CSS

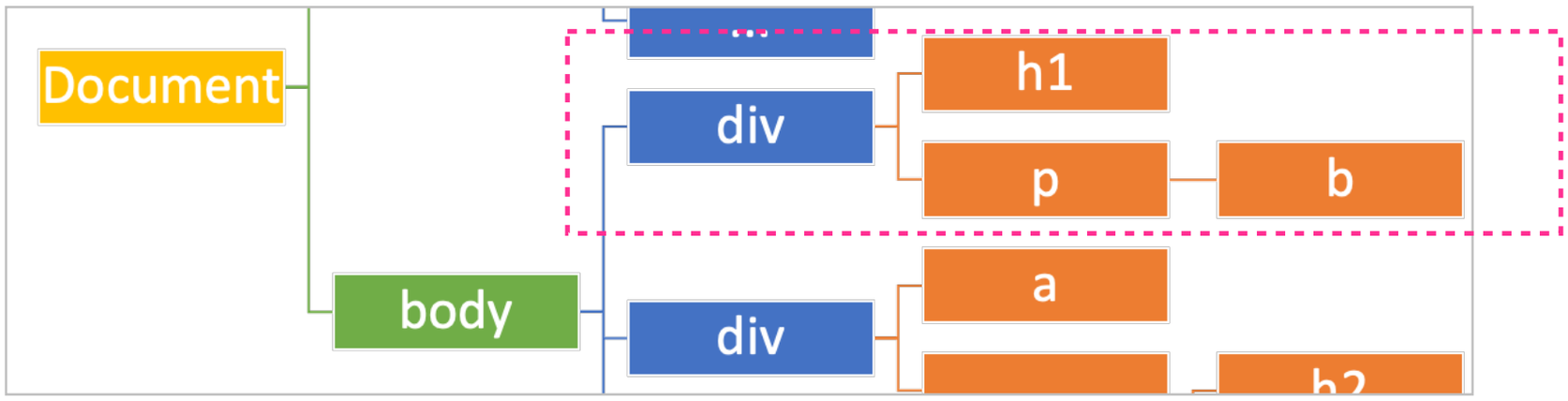
```css
/* Header/logo Title */
.header {
  padding: 80px;
  text-align: center;
  background: #1abc9c;
  color: white;
}

/* Increase the font size of the heading */
.header h1 {
  font-size: 40px;
}
```

body

```html
<div class="header">
  <h1>My Website</h1>
  <p>A <b>responsive</b> website created by me.</p>
</div>
```


Document
html
head
title
meta
...
body
div
h1
p
b
div
a
...
div
div
h2
h5
...

DOM Node Example




```
<div class="header">
  <h1>My Website</h1>
  <p>A <b>responsive</b> website created by me.</p>
</div>
```

- Node ‘div’ has 2 children (h1 and p) and 1 attribute (key=‘class’, value=‘header’)

localhost

jupyter 1 - Basic Webpage Scraping (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

Basic Webpage Scraping

Webpage scraping consists of two steps: crawling and parsing. In this tutorial, we focus on parsing HTML data. Beautifulsoup is a powerful tool to process static HTML. More details can be found at https://www.crummy.com/software/BeautifulSoup/bs4/doc/

To simplify our learning, we will use a simple example from W3Schools: https://www.w3schools.com/howto/tryit.asp?filename=tryhow_css_example_website

In [ ]:
```
import sys
IN_COLAB = 'google.colab' in sys.modules
if IN_COLAB:
    !wget https://www.dropbox.com/s/w5khgpro1ym3icg/simple_page.html?dl=
```

In [ ]:
```
with open('simple_page.html') as f:
    html = f.read()
```

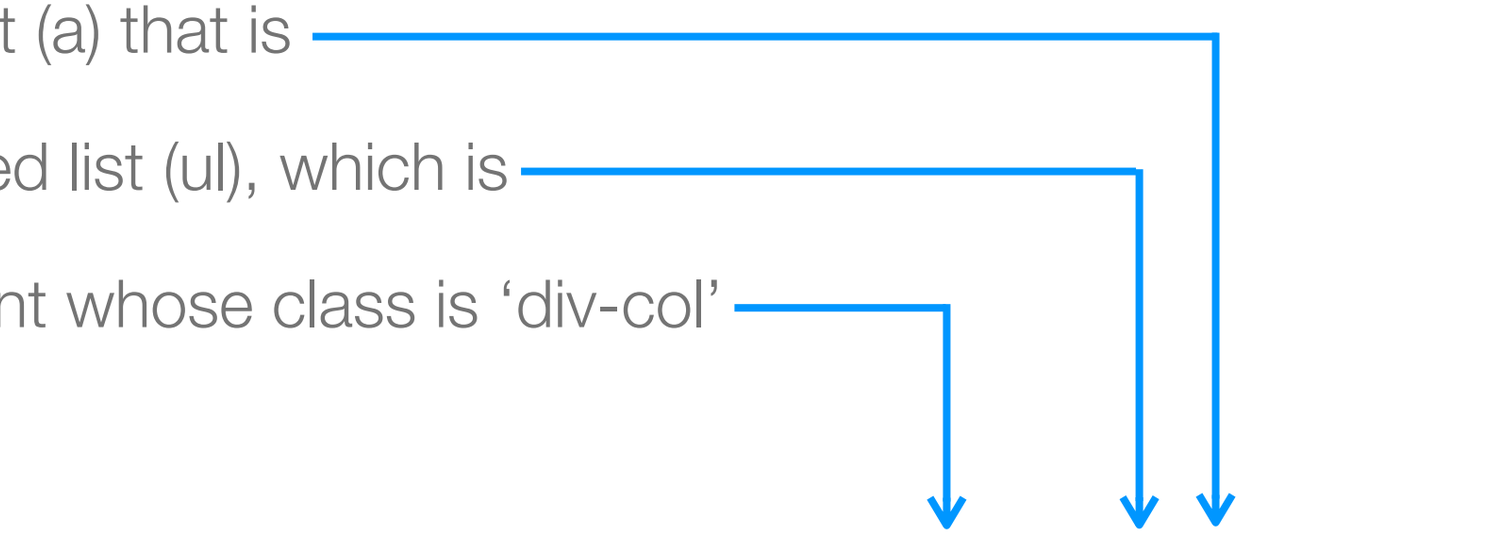
In [ ]:
```
html
```

In [ ]:
```
from bs4 import BeautifulSoup
from bs4.element import Tag
from IPython.core.display import HTML
```

In [ ]:
```
soup = BeautifulSoup(html, "lxml")
print(soup.prettify())
```

Understand CSS Selector Syntax

- CSS Selector is a powerful way to navigate DOM tree and locate DOM node(s)
- Find node(s) by tag, id, class, attributes, states (active, focus, hover, etc.), and much more
- Support hierarchical structure e.g. descendant, sibling, order of children, etc.
- In the example, we want to select
  - an anchor element (a) that is
  - inside an unordered list (ul), which is
  - inside a div element whose class is ‘div-col’



```
a_list = soup.select('div.div-col ul a')
```

Selector	Example	Example description
.class	.intro	Selects all elements with class="intro"
.class1.class2	.name1.name2	Selects all elements with both *name1* and *name2* set within its class attribute
.class1 .class2	.name1 .name2	Selects all elements with *name2* that is a descendant of an element with *name1*
#id	#firstname	Selects the element with id="firstname"
*	*	Selects all elements
element	p	Selects all <p> elements
element.class	p.intro	Selects all <p> elements with class="intro"
element,element	div, p	Selects all elements and all <p> elements
element element	div p	Selects all <p> elements inside elements
element>element	div > p	Selects all <p> elements where the parent is a element
element+element	div + p	Selects the first <p> element that is placed immediately after elements
element1~element2	p ~ ul	Selects every element that is preceded by a <p> element
[*attribute*]	[target]	Selects all elements with a target attribute
[*attribute=value*]	[target=_blank]	Selects all elements with target="_blank"
[*attribute~=value*]	[title~=flower]	Selects all elements with a title attribute containing the word "flower"
[*attribute|=value*]	[lang|=en]	Selects all elements with a lang attribute value equal to "en" or starting with "en-"
[*attribute^=value*]	a[href^="https"]	Selects every <a> element whose href attribute value begins with "https"

https://www.w3schools.com/cssref/css_selectors.asp

Example: Extracting references from a wiki page

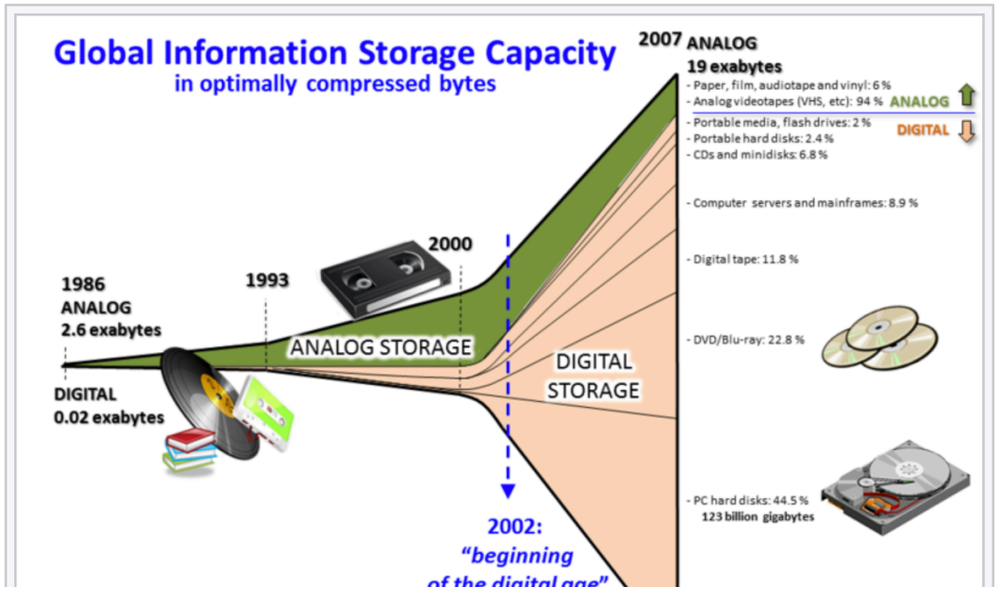
Big data

From Wikipedia, the free encyclopedia

This article is about large collections of data. For the band, see Big Data (band). For buying and selling of personal and consumer data, see Surveillance capitalism.

This article **may contain an excessive number of citations**. Please consider removing references to unnecessary or disreputable sources, merging citations where possible, or, if necessary, flagging the content for deletion. *In particular many references are "spammed" here for promotional purposes. These need to be removed.*. *(November 2019) (Learn how and when to remove this template message)*

Big data is a field that treats ways to analyze, systematically extract information from, or otherwise deal with data sets that are too large or complex to be dealt with by traditional data-processing application software. Data with many cases (rows) offer greater statistical power, while data with higher complexity (more attributes or columns) may lead to a higher false discovery rate.[2] Big data challenges include capturing data, data storage, data analysis, search, sharing, transfer, visualization, querying, updating, information privacy and data source. Big data was



- Encouraging members of society to abandon interactions with institutions that would create a digital trace, thus creating obstacles to social inclusion

If these potential problems are not corrected or regulated, the effects of big data policing may continue to shape societal hierarchies. Conscientious usage of big data policing could prevent individual level biases from becoming institutional biases, Brayne also notes.

See also [edit]

For a list of companies, and tools, see also: Category:Big data

- Big data ethics – Ethics of mass data analytics
- Big data maturity model – Aspect of computer science
- Big memory – A large amount of random-access memory
- Data curation – work performed to ensure meaningful and enduring access to data
- Data defined storage – Marketing term for managing data by combining application, information and storage tiers
- Data engineering – Software engineering approach to designing and developing information systems
- Data lineage – Origins and events of data
- Data philanthropy – Aspect of culture
- Data science – Interdisciplinary field of study on deriving knowledge and insights from data
- Datafication – Technological trend
- Document-oriented database – Type of computer program
- List of big data companies
- Very large database – type of database containing a very large amount of data, so much that it can require specialized architectural, management, processing and maintenance methodologies
- XLDB – annual conference series on databases, data management and analytics

References [edit]

1. ^ Hilbert, Martin; López, Priscila (2011). "The World's Technological Capacity to Store, Communicate, and Compute Information". *Science*. **332** (6025): 60–65. Bibcode:2011Sci...332...60H. doi:10.1126/science.1200970. PMID 21310967. S2CID 206531385. Archived from the original on 14 April 2016. Retrieved 13 April 2016.
2. ^ Breur, Tom (July 2016). "Statistical Power Analysis and the contemporary "crisis" in social sciences". *Journal of Marketing Analytics*. London, England: Palgrave Macmillan. **4** (2–3): 61–65. doi:10.1057/s41270-016-0001-3. ISSN 2050-3318.

104. ^ Josh Rogin (2 August 2018). "Ethnic cleansing makes a comeback – in China". No. Washington Post. Archived from the original on 31 March 2019. Retrieved 4 August 2018. "Add to that the unprecedented security and surveillance state in Xinjiang, which includes all-encompassing monitoring based on identity cards, checkpoints, facial recognition and the collection of DNA from millions of individuals. The authorities feed all this data into an artificial-intelligence machine that rates people's loyalty to the Communist Party in order to control every aspect of their lives."

Brayne also notes.

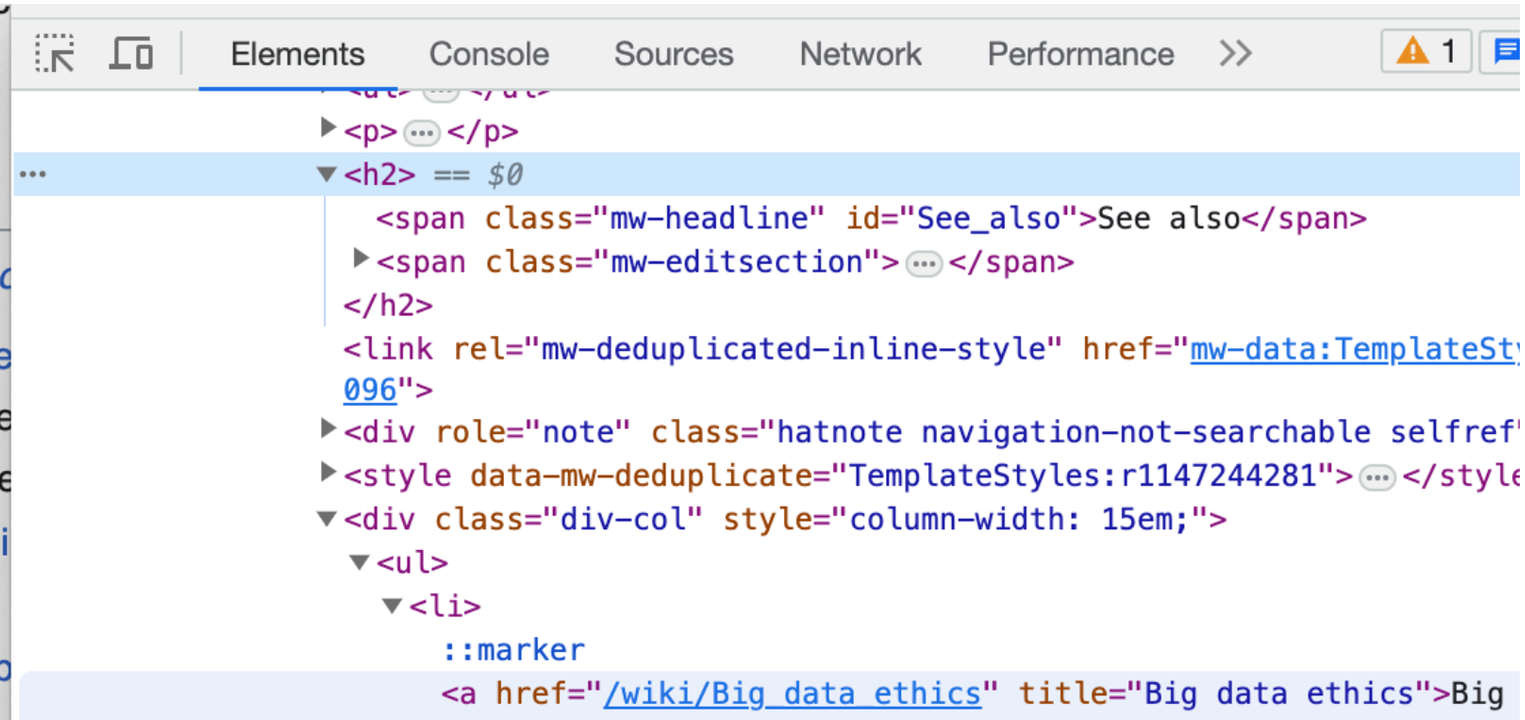
See also [edit]

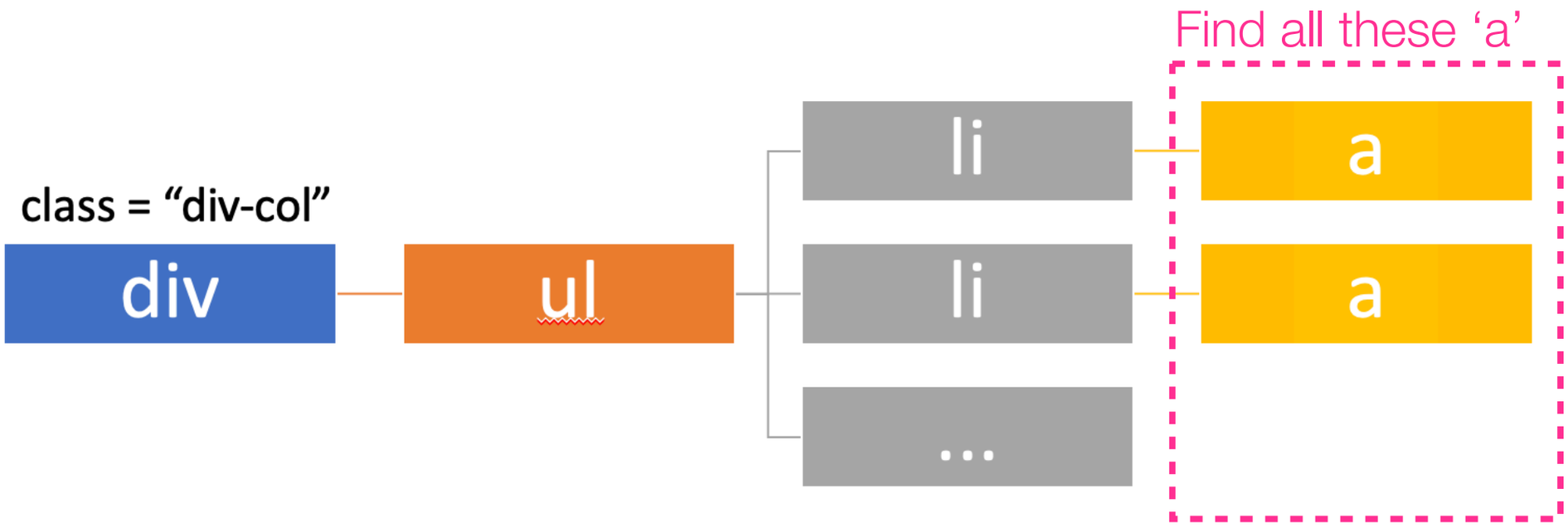
a 91.84 × 16

panies, and tools, see also: Catego

- Big data ethics – Ethics of mass data analytics
- Big data maturity model – Aspect of computer science
- Big memory – A large amount of random-access memory
- Data curation – work performed to ensure meaningful and enduring access to data
- Data defined storage – Marketing term

- Data e engine and de
- Data li data
- Data p
- Data s study insight
- Datafic





localhost

jupyter 2 - Wikipedia page data extraction (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Markdown

Wikipedia page data extraction

In this tutorial, we will learn how to extract a static page and convert it into useful information.

We first get a wikipeidia page using requests.

In [ ]:
```
import requests
import re
from bs4 import BeautifulSoup
```

In [ ]:
```
bigdata = requests.get('https://en.wikipedia.org/wiki/Big_data')
```

In [ ]:
```
len(bigdata.text)
```

In [ ]:
```
bigdata.text
```

Parsing a wikipedia page

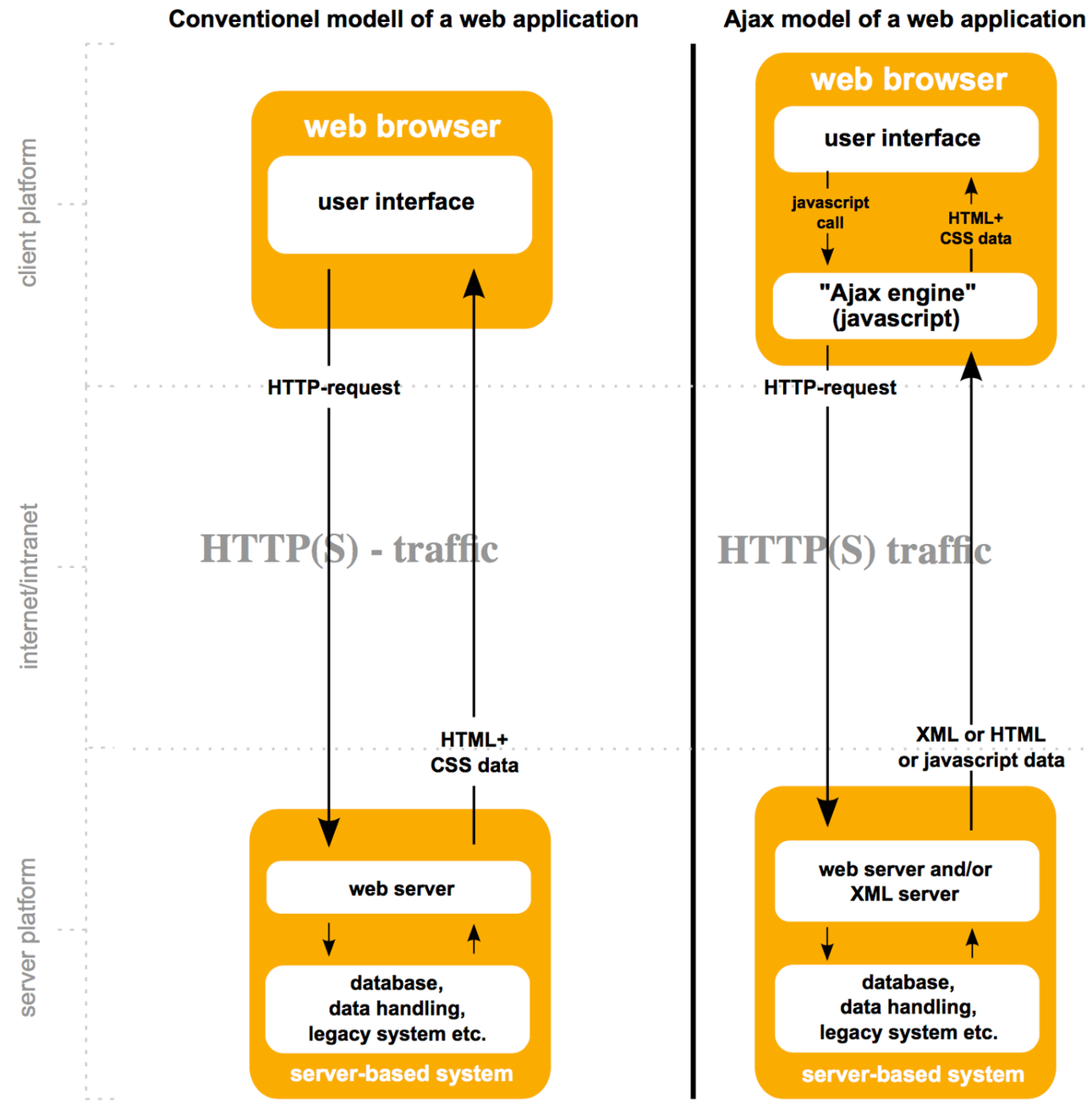
In [ ]:
```
soup = BeautifulSoup(bigdata.text, "lxml")
print(soup.prettify())
```

In [ ]:
```
soup.title.string
```

CHULA ENGINEERING
Foundation toward Innovation

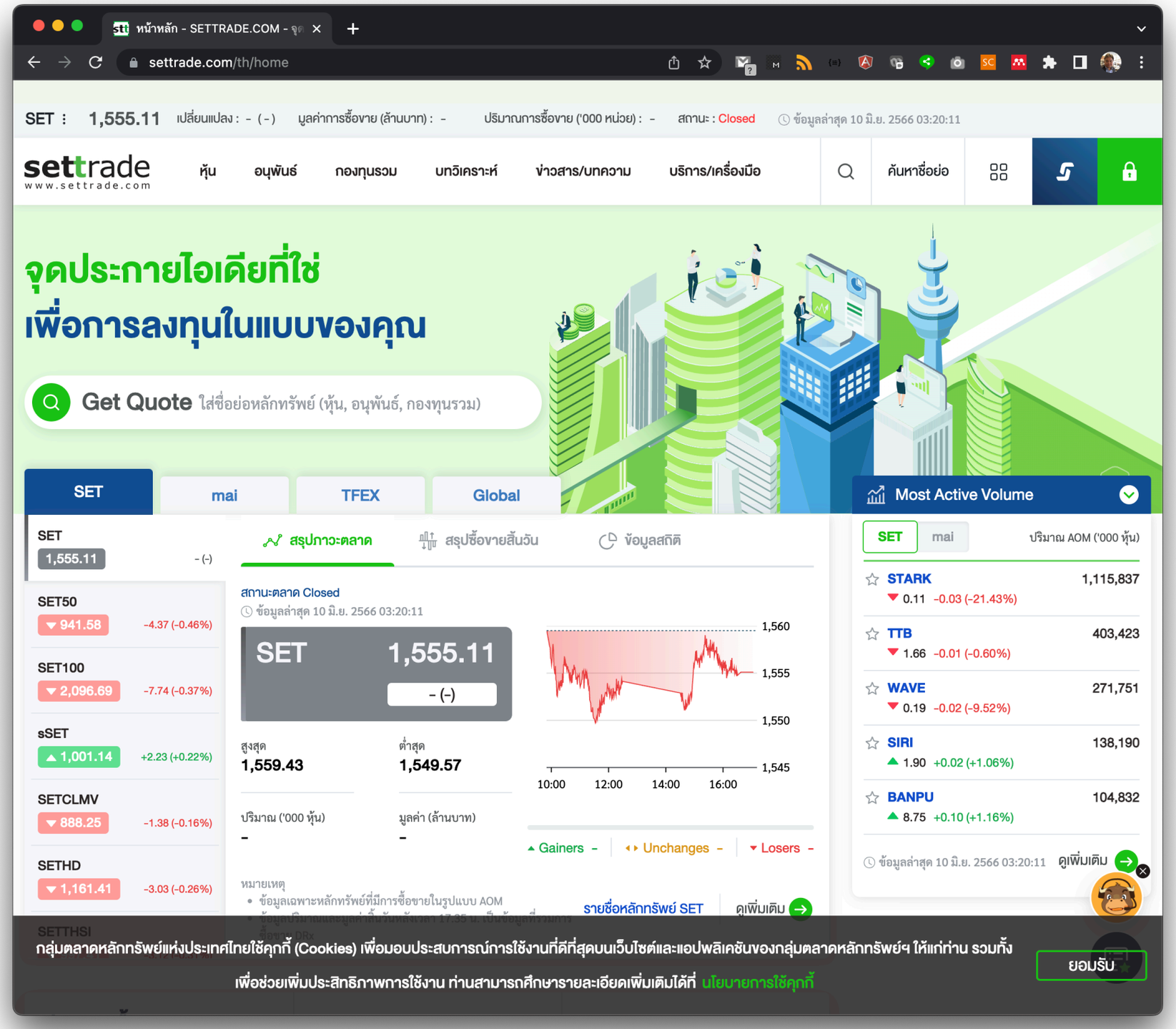
Extracting AJAX-based Webpage

- Modern web applications are AJAX-based e.g. SPA
- Typical web request approach does not work



Source: https://en.wikipedia.org/wiki/Ajax_(programming)

https://www.settrade.com/settrade/home



DevTools - www.settrade.com/th/home

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights AngularJS

8 1 6

Preserve log Disable cache No throttling

Filter Invert Hide data URLs All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other Has blocked cookies Blocked Requests 3rd-party requests

5000 ms 10000 ms 15000 ms 20000 ms 25000 ms 30000 ms 35000 ms 40000 ms 45000 ms 50000 ms 55000 ms 60000 ms 65000 ms 70000 ms 75000 ms 80000 ms 85000 ms 90000 ms 95000 ms 100000 ms 105000 ms 110000 m

Name	Status	Type	Initiator	Size	Time	Waterfall
Pug?vcode=bz0yJnR5cGU9MSZqcz0xJmNvZGU9MjkzNiZ0bD00…9C0E44D643DA…	200	document	um.simpli.fi/pm_match?https://simage2.p…	53 B	38 ms	
Pug?vcode=bz0yJnR5cGU9MSZjb2RlPTMwNzQmdGw9MTI5NjAw…ckCookie=uid:i…	200	document	pm.w55c.net/ping_match.gif?ei=PUBMAT…	301 B	41 ms	
cs?pid=45&rndcb=7018091133	302	document / Redirect	sync.1rx.io/usersync2/pubmatic&gdpr=0&…	435 B	535 ms	
4298321646126704324?dspret=1&gdpr=0&gdpr_consent=&us_privacy=	302	document / Redirect	ad.turn.com/r/cs?pid=45&rndcb=7018091…	664 B	68 ms	
Pug?vcode&piggybackCookie={viewer_token}	200	document	csync.loopme.me/?pubid=11331&redirect…	74 B	39 ms	
cksync.php?cs=1&ovsid=3d8264dc-3139-4be6-8991-58dedda99953&type=loop&gd…	200	gif	csync.loopme.me/	473 B	97 ms	
RX-a9f8e578-bdd5-43f2-8a3a-4b62a5cdec0b-004?redir=…kie%3DRX-a9f8e578-bd…	302	document / Redirect	sync.1rx.io/usersync/turn/4298321646126…	527 B	206 ms	
Pug?vcode=bz0yJnR5cGU9MSZjb2RlPTMyMDMmdGw9NDMyMDA=…ookie=RX-a9…	200	document	sync.targeting.unrulymedia.com/csync/RX…	336 B	41 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	18 ms	
SPug?partnerID=158497&gdpr=0&gdpr_consent=&us_privacy=	200	script	user_sync.html?kdntuid=1&p=158497:1	128 B	40 ms	
list?type=INDEX	304	xhr	xhr.js:177	123 B	17 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	23 ms	
dc_oe=ChMIsNHvorq7_wIV4pdmAh2XCwV0EAAYACCox6dGQhMI…1;&timestamp=…	200	gif	express_html_inpage_rendering_lib_200_2…	108 B	97 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	23 ms	
PugMaster?sec=1&async=1&kdntuid=1&rnd=42620359&p=1…0&spug=1&coppa=0&…	200	script	user_sync.html?p=156011&s=165626&pr…	168 B	42 ms	
list?type=INDEX	304	xhr	xhr.js:177	125 B	19 ms	
ecm3?ex=pubmatic.com&id=PM_UID2271923A-F270-436E-BF37-15B167A758A2	200	document		479 B	272 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	38 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	26 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	18 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	21 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	16 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	17 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	22 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	22 ms	
?ai=ChC2P1d2FZPrdMt-NssUPpuGv0A2S_af_cNSby8n5EcLlk…w_focus&gqid&qqid…	200	text/html	window_focus_fy2021.js:6	0 B	50 ms	
?ai=CGKJv1t2FZPTUE4yXoQPZ7quQD7CK7qJvrqf81ZUNZBABI…w_focus&gqid&q…	200	text/html	window_focus_fy2021.js:6	0 B	48 ms	
list?type=INDEX	304	xhr	xhr.js:177	126 B	16 ms	
list?type=INDEX	304	xhr	xhr.js:177	125 B	34 ms	
list?type=INDEX	304	xhr	xhr.js:177	125 B	18 ms	
dc_oe=ChMIsNHvorq7_wIV4pdmAh2XCwV0EAAYACCox6dGQhMI…1;&timestamp=…	200	gif	express_html_inpage_rendering_lib_200_2…	63 B	108 ms	
list?type=INDEX	304	xhr	xhr.js:177	125 B	23 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	15 ms	
?ai=ChC2P1d2FZPrdMt-NssUPpuGv0A2S_af_cNSby8n5EcLlk…w_focus&gqid&qqid…	200	text/html	window_focus_fy2021.js:6	0 B	53 ms	
?ai=CGKJv1t2FZPTUE4yXoQPZ7quQD7CK7qJvrqf81ZUNZBABI…w_focus&gqid&q…	200	text/html	window_focus_fy2021.js:6	0 B	49 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	21 ms	
list?type=INDEX	304	xhr	xhr.js:177	123 B	19 ms	
?ai=ChC2P1d2FZPrdMt-NssUPpuGv0A2S_af_cNSby8n5EcLlk…w_focus&gqid&qqid…	200	text/html	window_focus_fy2021.js:6	0 B	49 ms	
?ai=CGKJv1t2FZPTUE4yXoQPZ7quQD7CK7qJvrqf81ZUNZBABI…w_focus&gqid&q…	200	text/html	window_focus_fy2021.js:6	0 B	49 ms	

48 requests | 18.6 kB transferred | 77.7 kB resources

CHULA ENGINEERING
Foundation toward Innovation

DevTools - www.settrade.com/th/home

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights AngularJS

8 1 6

Preserve log Disable cache No throttling

Filter Invert Hide data URLs All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other Has blocked cookies Blocked Requests 3rd-party requests

20000 ms 40000 ms 60000 ms 80000 ms 100000 ms 120000 ms 140000 ms 160000 ms 180000 ms 200000 ms 220000 ms 240000 ms 260000 ms

Name	Status	Type	Initiator	Size	Time	Waterfall
list?type=INDEX	304	xhr	xhr.js:177	124 B	38 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	26 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	18 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	21 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	16 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	17 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	22 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	22 ms	
list?type=INDEX	304	xhr	xhr.js:177	126 B	16 ms	
list?type=INDEX	304	xhr	xhr.js:177	125 B	34 ms	
list?type=INDEX	304	xhr	xhr.js:177	125 B	18 ms	
list?type=INDEX	304	xhr	xhr.js:177	125 B	23 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	15 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	21 ms	
list?type=INDEX	304	xhr	xhr.js:177	123 B	19 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	23 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	19 ms	
list?type=INDEX	304	xhr	xhr.js:177	123 B	20 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	27 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	22 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	25 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	25 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	26 ms	
list?type=INDEX	304	xhr	xhr.js:177	125 B	26 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	31 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	27 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	27 ms	
list?type=INDEX	304	xhr	xhr.js:177	123 B	20 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	24 ms	
list?type=INDEX	200	xhr	xhr.js:177	(disk cache)	0 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	20 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	25 ms	
list?type=INDEX	304	xhr	xhr.js:177	125 B	17 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	27 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	27 ms	
list?type=INDEX	304	xhr	xhr.js:177	124 B	31 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	30 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.0 kB	20 ms	
list?type=INDEX	200	xhr	xhr.js:177	1.1 kB	19 ms	

45 / 79 requests | 27.8 kB / 35.5 kB transferred | 164 kB / 165 kB resources

CHULA ENGINEERING
Foundation toward Innovation

DevTools - www.settrade.com/th/home

Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights AngularJS

Preserve log Disable cache No throttling

Filter Invert Hide data URLs All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other Has blocked cookies Blocked Requests 3rd-party requests

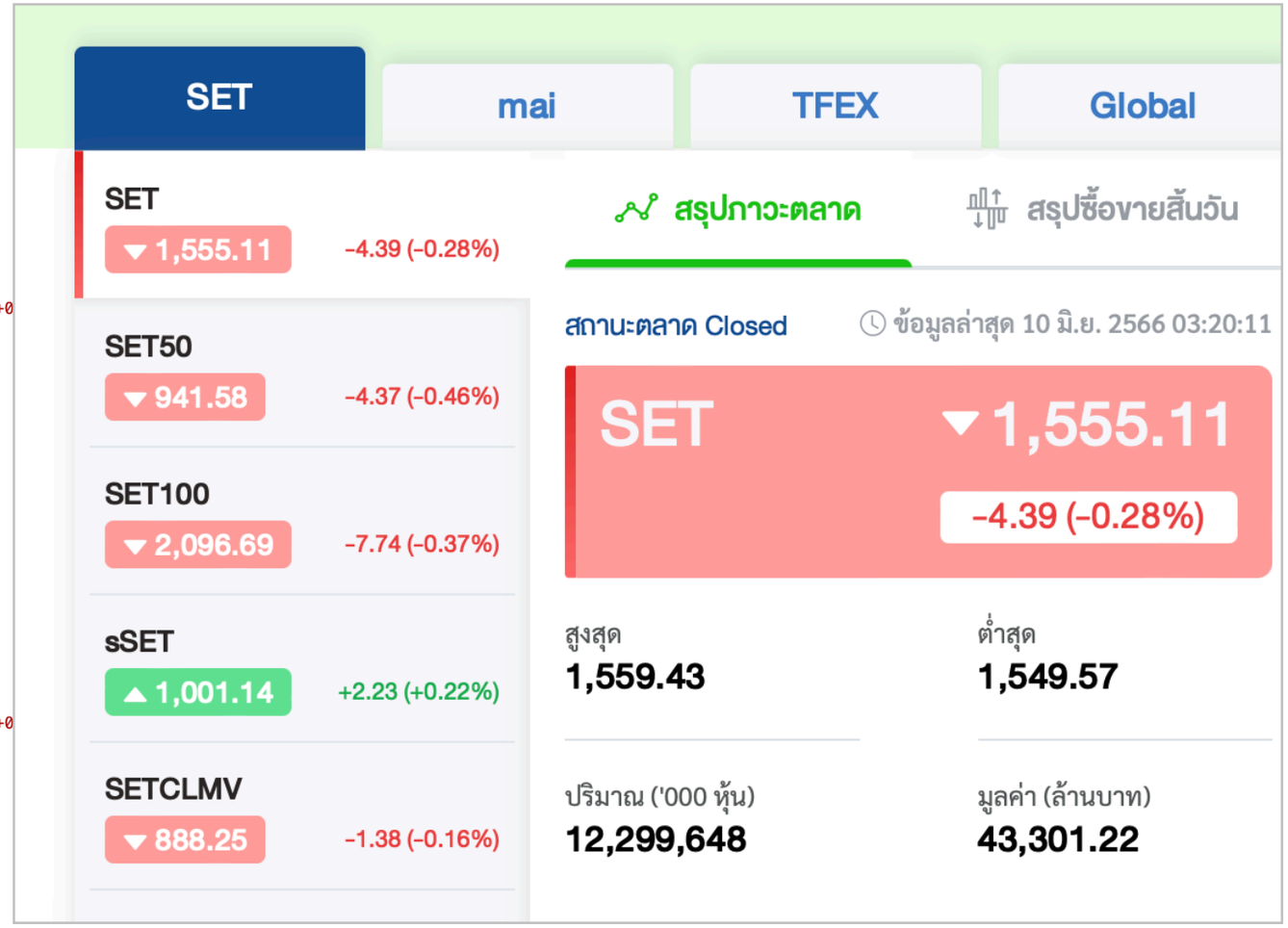
20000 ms 40000 ms 60000 ms 80000 ms 100000 ms 120000 ms 140000 ms 160000 ms 180000 ms 200000 ms 220000 ms 240000 ms 260000 ms 280000 ms 300000 ms 320000 ms

Name

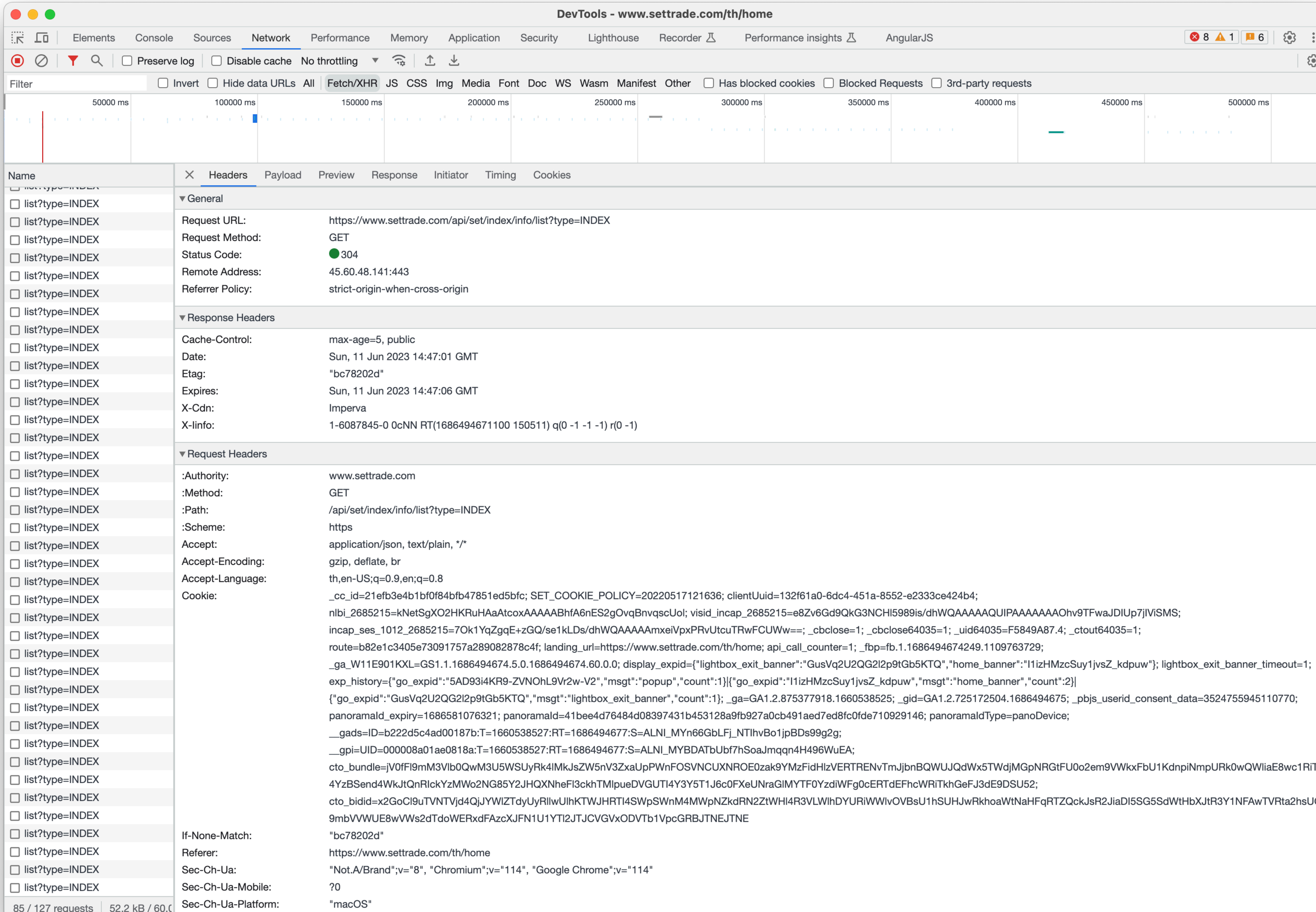
list?type=INDEX

Headers Payload Preview Response Initiator Timing Cookies


```
{
    "indexIndustrySectors": [
        {
            "symbol": "SET",
            "nameEN": "SET",
            "nameTH": "SET",
            "prior": 1559.500000,
            "open": 1558.940000,
            "high": 1559.430000,
            "low": 1549.570000,
            "last": 1555.110000,
            "change": null,
            "percentChange": null,
            "volume": null,
            "value": null,
            "querySymbol": "SET",
            "marketStatus": "Closed",
            "marketDateTime": "2023-06-10T03:20:11.564264874+0
            "marketName": "SET",
            "industryName": "",
            "sectorName": "",
            "level": "INDEX"
        },
        {
            "symbol": "SET50",
            "nameEN": "SET50",
            "nameTH": "SET50",
            "prior": 945.950000,
            "open": 944.800000,
            "high": 945.280000,
            "low": 938.260000,
            "last": 941.580000,
            "change": -4.370000,
            "percentChange": -0.460000,
            "volume": 1240610800,
            "value": 29122205677.810000,
            "querySymbol": "SET50",
            "marketStatus": "Closed",
            "marketDateTime": "2023-06-10T03:20:11.564264874+0
            "marketName": "SET",
            "industryName": "",
            "sectorName": "",
            "level": "INDEX"
        },
        {
            "symbol": "SET100",
            "nameEN": "SET100",
            "nameTH": "SET100",
            "prior": 2104.430000,
            "open": 2102.360000,
            "high": 2103.880000,
            "low": 2088.550000,
            "last": 2096.690000,
            "change": -7.740000,
            "percentChange": -0.370000,
            "volume": 1822239600,
            "value": 34185571732.510000,
            "querySymbol": "SET100",
            "marketStatus": "Closed",
            "marketDateTime": "2023-06-10T03:20:11.564264874+07:00",
            "marketName": "SET",
            "industryName": "",
            "sectorName": ""
```

64 / 102 requests 41.3 kB / 49.0

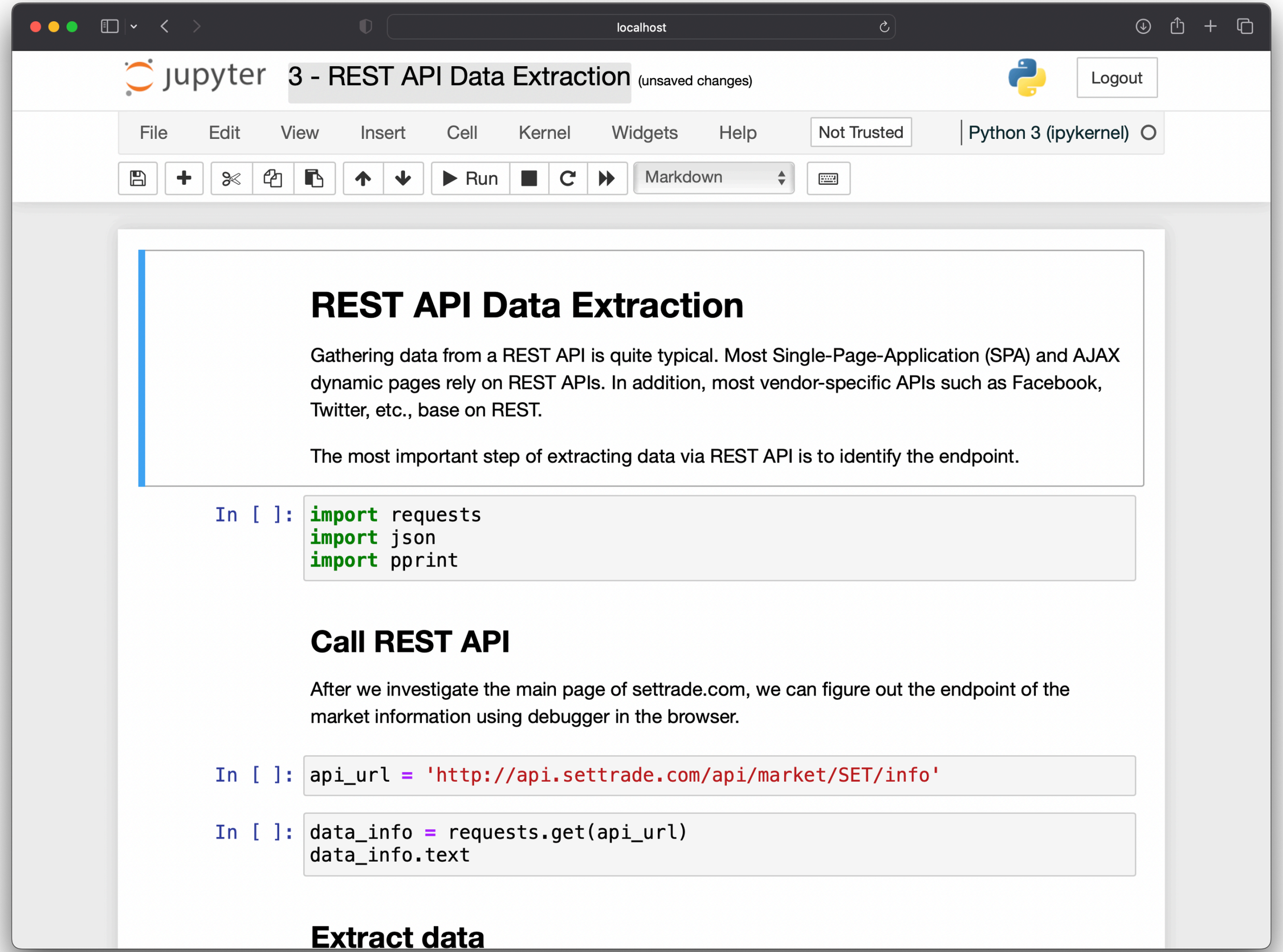

DevTools - www.settrade.com/th/home
Elements Console Sources Network Performance Memory Application Security Lighthouse Recorder Performance insights AngularJS
8 1 6
Preserve log Disable cache No throttling
Filter Invert Hide data URLs All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other Has blocked cookies Blocked Requests 3rd-party requests
50000 ms 100000 ms 150000 ms 200000 ms 250000 ms 300000 ms 350000 ms 400000 ms 450000 ms 500000 ms
Name
list?type=INDEX
85 / 127 requests 52.2 kB / 60.0
Headers Payload Preview Response Initiator Timing Cookies
General
Request URL: https://www.settrade.com/api/set/index/info/list?type=INDEX
Request Method: GET
Status Code: 304
Remote Address: 45.60.48.141:443
Referrer Policy: strict-origin-when-cross-origin
Response Headers
Cache-Control: max-age=5, public
Date: Sun, 11 Jun 2023 14:47:01 GMT
Etag: "bc78202d"
Expires: Sun, 11 Jun 2023 14:47:06 GMT
X-Cdn: Imperva
X-Iinfo: 1-6087845-0 0cNN RT(1686494671100 150511) q(0 -1 -1 -1) r(0 -1)
Request Headers
:Authority: www.settrade.com
:Method: GET
:Path: /api/set/index/info/list?type=INDEX
:Scheme: https
Accept: application/json, text/plain, */*
Accept-Encoding: gzip, deflate, br
Accept-Language: th,en-US;q=0.9,en;q=0.8
Cookie: _cc_id=21efb3e4b1bf0f84bfb47851ed5bfc; SET_COOKIE_POLICY=20220517121636; clientUuid=132f61a0-6dc4-451a-8552-e2333ce424b4;
nlbi_2685215=kNetSgXO2HKRuHAaAtcoxAAAAABhfA6nES2gOvqBnvqscUoI; visid_incap_2685215=e8Zv6Gd9QkG3NCHI5989is/dhWQAAAAAQUIPAAAAAAAOhv9TFwaJDIUp7jIViSMS;
incap_ses_1012_2685215=7Ok1YqZgqE+zGQ/se1kLDs/dhWQAAAAAmxeiVpxPRvUtcuTRwFCUWw==; _cbclose=1; _cbclose64035=1; _uid64035=F5849A87.4; _ctout64035=1;
route=b82e1c3405e73091757a289082878c4f; landing_url=https://www.settrade.com/th/home; api_call_counter=1; _fbp=fb.1.1686494674249.1109763729;
_ga_W11E901KXL=GS1.1.1686494674.5.0.1686494674.60.0.0; display_expid={"lightbox_exit_banner":"GusVq2U2QG2I2p9tGb5KTQ","home_banner":"I1izHMzcSuy1jvsZ_kdpuw"}; lightbox_exit_banner_timeout=1;
exp_history={"go_expid":"5AD93i4KR9-ZVNOhL9Vr2w-V2","msgt":"popup","count":1}|{"go_expid":"I1izHMzcSuy1jvsZ_kdpuw","msgt":"home_banner","count":2}|
{"go_expid":"GusVq2U2QG2I2p9tGb5KTQ","msgt":"lightbox_exit_banner","count":1}; _ga=GA1.2.875377918.1660538525; _gid=GA1.2.725172504.1686494675; _pbjs_userid_consent_data=3524755945110770;
panoramaId_expiry=1686581076321; panoramaId=41bee4d76484d08397431b453128a9fb927a0cb491aed7ed8fc0fde710929146; panoramaIdType=panoDevice;
__gads=ID=b222d5c4ad00187b:T=1660538527:RT=1686494677:S=ALNI_MYn66GbLFj_NTIhvBo1jpBDs99g2g;
__gpi=UID=000008a01ae0818a:T=1660538527:RT=1686494677:S=ALNI_MYBDATbUbf7hSoaJmqqn4H496WuEA;
cto_bundle=jV0fFl9mM3Vlb0QwM3U5WSUyRk4lMkJsZW5nV3ZxaUpPWnFOSVNCUXNROE0zak9YMzFidHlzVERTRENvTmJjbnBQWUJQdWx5TWdjMGpNRGtFU0o2em9VWkxFbU1KdnpiNmpURk0wQWliaE8wc1RiT0l
4YzBSend4WkJtQnRIckYzMWo2NG85Y2JHQXNheFl3ckhTMlpueDVGUTI4Y3Y5T1J6c0FXeUNraGlMYTF0YzdiWFg0cERTdEFhcWRiTkhGeFJ3dE9DSU52;
cto_bidid=x2GoCl9uTVNTVjd4QjJYWlZTdyUyRllwUlhKTWJHRTI4SWpSWnM4MWpNZkdRN2ZtWHl4R3VLWlhDYURiWWlvOVBsU1hSUHJwRkhoaWtNaHFqRTZQckJsR2JiaDI5SG5SdWtHbXJtR3Y1NFAwTVRta2hsUG
9mbVVWUE8wVWs2dTdoWERxdFAzcXJFN1U1YTl2JTJCVGVxODVTb1VpcGRBJTNEJTNE
If-None-Match: "bc78202d"
Referer: https://www.settrade.com/th/home
Sec-Ch-Ua: "Not.A/Brand";v="8", "Chromium";v="114", "Google Chrome";v="114"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "macOS"

CHULA ENGINEERING
Foundation toward Innovation

settrade.com/api/set/index/info

settrade.com/api/set/index/info/list?type=INDEX


```
// 20230611215914
// https://www.settrade.com/api/set/index/info/list?type=INDEX

{
  "indexIndustrySectors": [
    {
      "symbol": "SET",
      "nameEN": "SET",
      "nameTH": "SET",
      "prior": 1559.500000,
      "open": 1558.940000,
      "high": 1559.430000,
      "low": 1549.570000,
      "last": 1555.110000,
      "change": -4.390000,
      "percentChange": -0.280000,
      "volume": 12299648049,
      "value": 43301220472,
      "querySymbol": "SET",
      "marketStatus": "Closed",
      "marketDateTime": "2023-06-10T03:20:11.564264874+07:00",
      "marketName": "SET",
      "industryName": "",
      "sectorName": "",
      "level": "INDEX"
    },
    {
      "symbol": "SET50",
      "nameEN": "SET50",
      "nameTH": "SET50",
      "prior": 945.950000,
      "open": 944.800000,
      "high": 945.280000,
      "low": 938.260000,
      "last": 941.580000,
      "change": -4.370000,
      "percentChange": -0.460000,
      "volume": 1240610800,
      "value": 29122205677.810000,
      "querySymbol": "SET50",
      "marketStatus": "Closed"
```


localhost
jupyter
3 - REST API Data Extraction
(unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
REST API Data Extraction
Gathering data from a REST API is quite typical. Most Single-Page-Application (SPA) and AJAX dynamic pages rely on REST APIs. In addition, most vendor-specific APIs such as Facebook, Twitter, etc., base on REST.
The most important step of extracting data via REST API is to identify the endpoint.
In [ ]:
import requests
import json
import pprint
Call REST API
After we investigate the main page of settrade.com, we can figure out the endpoint of the market information using debugger in the browser.
In [ ]:
api_url = 'http://api.settrade.com/api/market/SET/info'
In [ ]:
data_info = requests.get(api_url)
data_info.text
Extract data

Extract Complicated Webpage with Selenium

- Modern webpages are Javascript-based, which can be very difficult or impossible to use our previous techniques
  - Normal web applications usually base on session of multiple web pages using cookies and HTTP headers
  - Single Page Application (SPA) augments existing elements when being interacted and generates new elements on-the-fly
- Selenium is an open source framework that allow us to scrape data from all these web applications

Selenium Architecture

Selenium Capabilities

- Process any website similar to normal web browser
  - Access a web page using normal URL
  - Grab HTML and other resource files in a particular web page
  - Find components based on criteria e.g. CSS selector, etc.
  - Interacting with components in a web page e.g. click, input keyboard
  - Navigate within a web page and across multiple pages
- During operation, Selenium can be normal mode or headless mode
- Checkout https://www.selenium.dev/documentation/ for more information

Example: Selenium on Google

- Visit https://www.google.com
- Get HTML of this web page
- Find a query box
- Enter ‘ประเทศไทย’ in the query box and press ‘enter’
- Search result links (‘a’ elements under an element with id=‘search’)
- Click the first link

Data Extraction with Selenium

In this tutorial, we discuss how to use Selenium to extract data from the web. Please see https://selenium-python.readthedocs.io for more details.

Installation

Before using selenium, we will have to install a webdriver of your choice. It can be Chrome or Firefox. Once installed, you will need to know the location of the drive as it will be used as a parameter to start a browser. To install the driver, just install python helper package chromedriver_autoinstaller.

```
pip install chromedriver_autoinstaller
```

We also have to install selenium package.

```
pip install selenium
```

In [ ]:
```
from selenium import webdriver
import chromedriver_autoinstaller
import time
import os
```

In [ ]:
```
chromedriver_autoinstaller.install()
```

Recommended Resources

- http://python.gotrained.com/python-json-api-tutorial/
- https://www.analyticsvidhya.com/blog/2015/10/beginner-guide-web-scraping-beautiful-soup-python/
- https://www.dataquest.io/blog/web-scraping-tutorial-python/
- https://www.datacamp.com/courses/importing-data-in-python-part-2
- https://code.tutsplus.com/tutorials/the-30-css-selectors-you-must-memorize--net-16048