Web Scraping

Hey, web scraping!

You may want to…

- download all videos from a website;
- download all news articles from a media platform;
- download all academic papers from a journal;
- download all tweets/weibo of a specific person.

You may need to spend days and nights downloading these data manually, and you can easily make a lot of mistakes.

Hey, web scraping!

In today’s class, we are going to learn about webscraping.
In Chinese, it is called 网络爬虫.

What is webscraping?
Using tools to gather data you can see on a webpage.
Almost anything you see on a website can be scraped.

It can be done with python, R,… We are doing it on R.

https://www.youtube.com/embed/Ct8Gxo8StBU?enablejsapi=1

Learning about HTML

HTML stands for “HyperText Markup Language.”

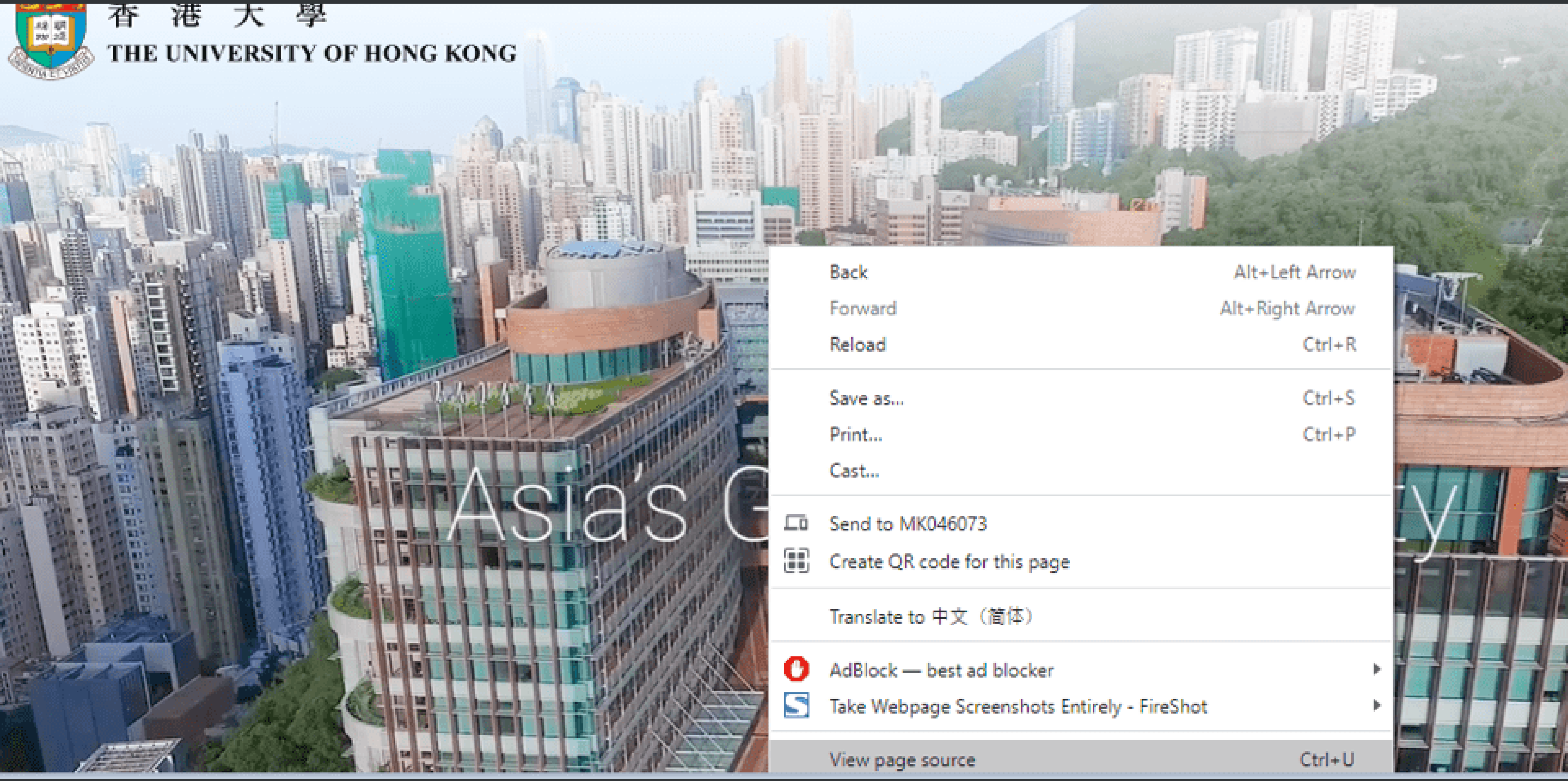
Websites are written on the HTML language, and web scraping is based on reading and interpreting the HTML of a webpage.

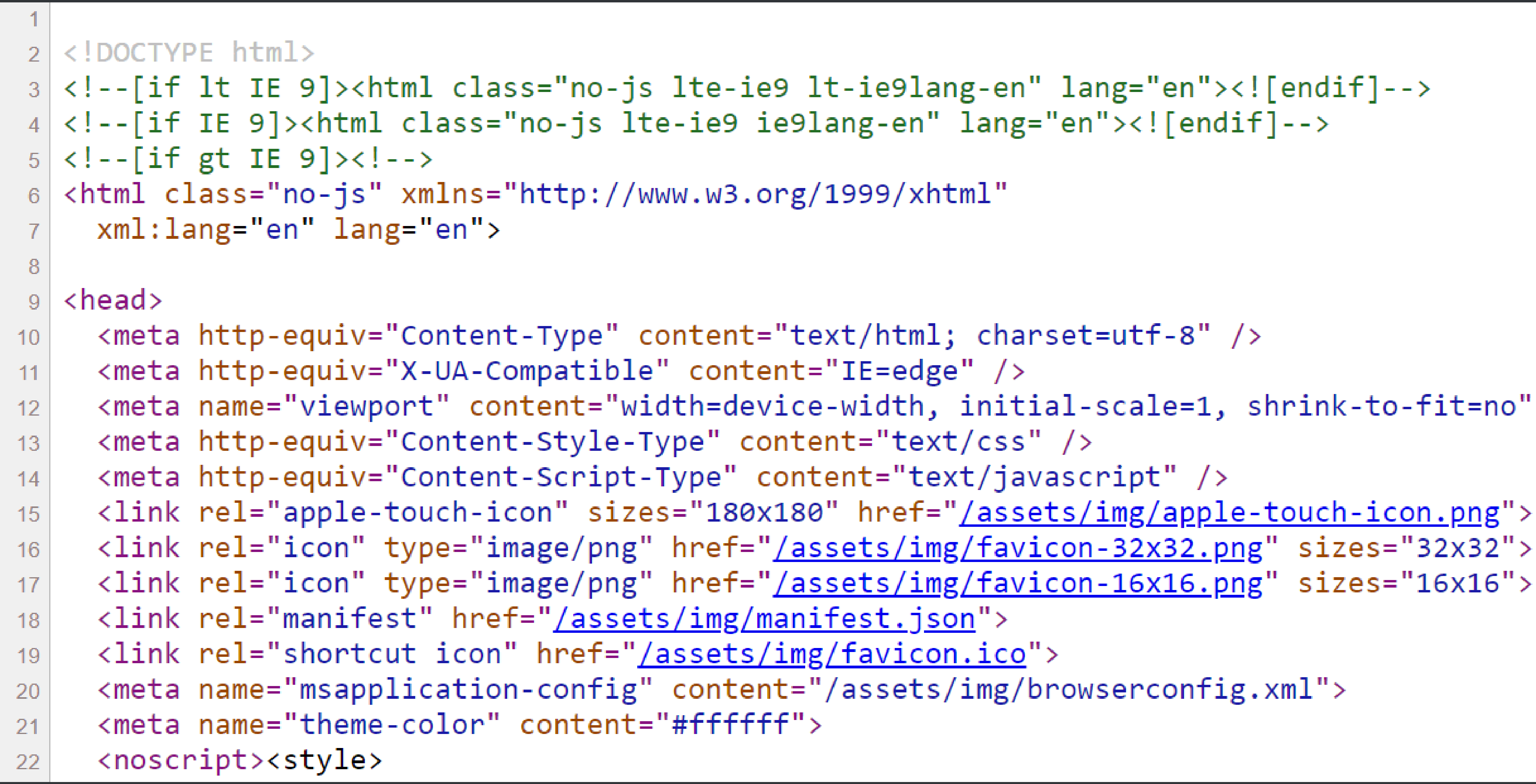
But how to find the HTML of a webpage?

Learning about HTML

Please use Google Chrome as your browser.

If you are not a user of Google Chrome, download and install one on your laptop. Google Chrome is particularly helpful for analyzing webpages for scraping.


香 港 大 學
THE UNIVERSITY OF HONG KONG
Asia's G
y
Back Alt+Left Arrow
Forward Alt+Right Arrow
Reload Ctrl+R
Save as... Ctrl+S
Print... Ctrl+P
Cast...
Send to MK046073
Create QR code for this page
Translate to 中文（简体）
AdBlock — best ad blocker
Take Webpage Screenshots Entirely - FireShot
View page source Ctrl+U



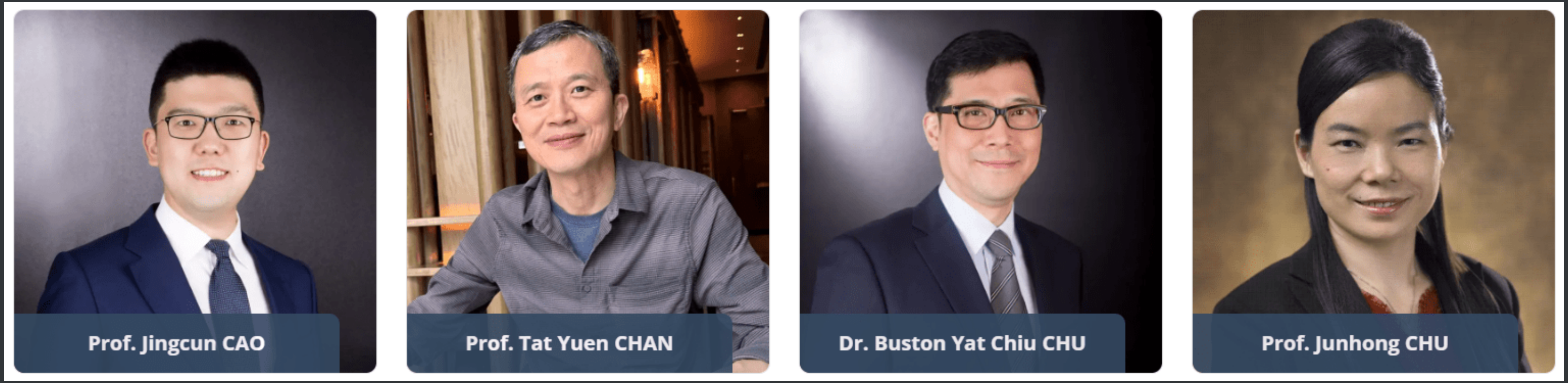
```html

<!DOCTYPE html>
<!--[if lt IE 9]><html class="no-js lte-ie9 lt-ie9lang-en" lang="en"><![endif]-->
<!--[if IE 9]><html class="no-js lte-ie9 ie9lang-en" lang="en"><![endif]-->
<!--[if gt IE 9]><!-->
<html class="no-js" xmlns="http://www.w3.org/1999/xhtml"
  xml:lang="en" lang="en">

<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no"
  <meta http-equiv="Content-Style-Type" content="text/css" />
  <meta http-equiv="Content-Script-Type" content="text/javascript" />
  <link rel="apple-touch-icon" sizes="180x180" href="/assets/img/apple-touch-icon.png">
  <link rel="icon" type="image/png" href="/assets/img/favicon-32x32.png" sizes="32x32">
  <link rel="icon" type="image/png" href="/assets/img/favicon-16x16.png" sizes="16x16">
  <link rel="manifest" href="/assets/img/manifest.json">
  <link rel="shortcut icon" href="/assets/img/favicon.ico">
  <meta name="msapplication-config" content="/assets/img/browserconfig.xml">
  <meta name="theme-color" content="#ffffff">
  <noscript><style>
```

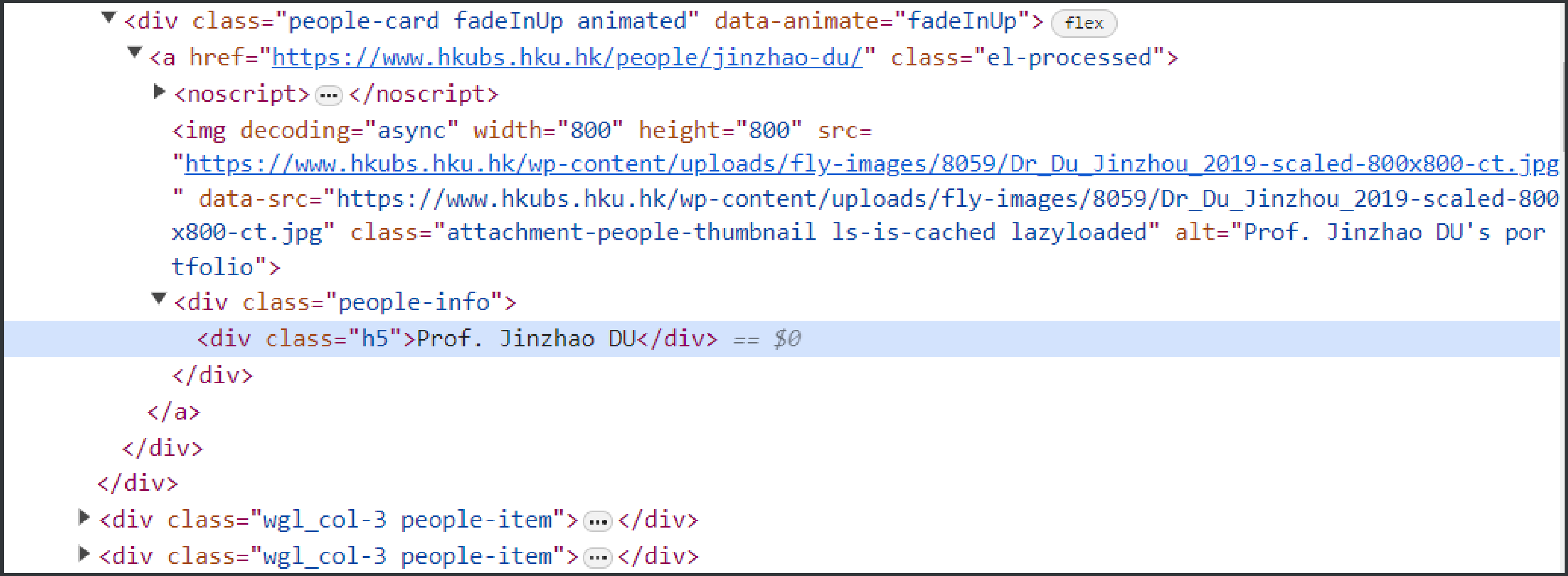
Learning about HTML

The data you want to scrape appears in certain place of the HTML. For example, suppose that you want to scrape data from the HKU marketing faculty webpage:



Learning about HTML

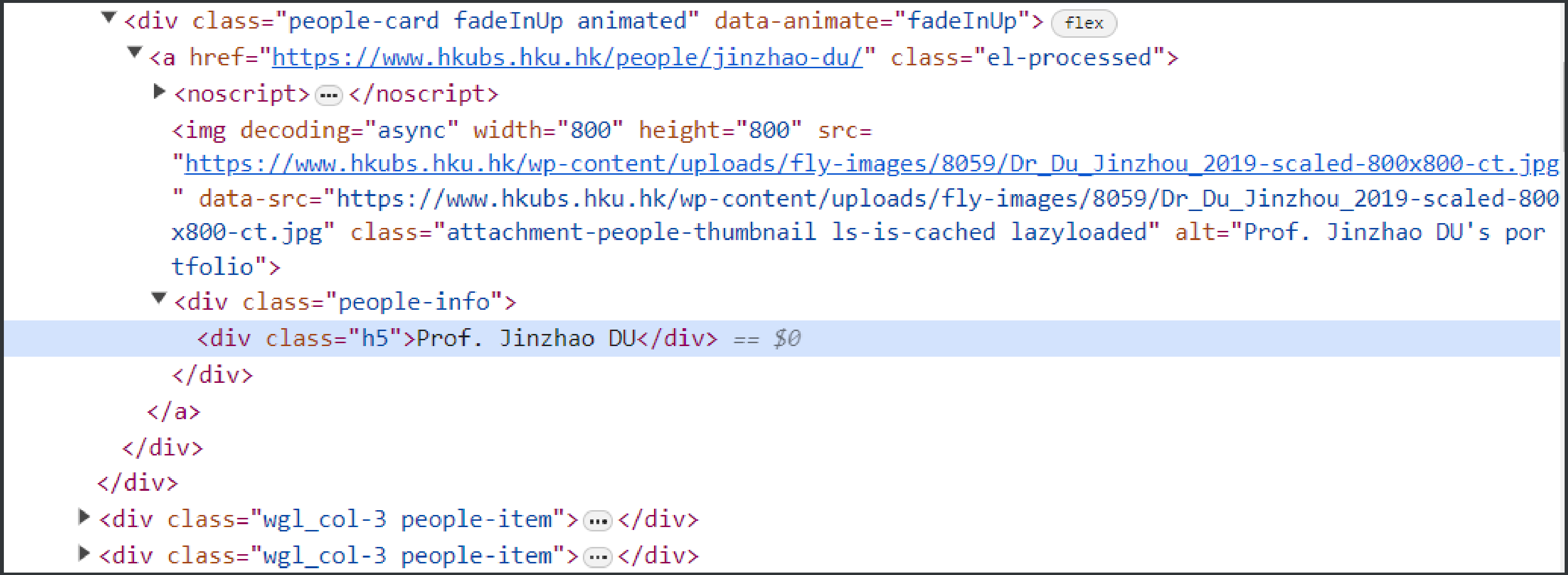
You can find the name and images of the professors from the HTML file:



```
    ▼<div class="people-card fadeInUp animated" data-animate="fadeInUp"> flex
      ▼<a href="https://www.hkubs.hku.hk/people/jinzhao-du/" class="el-processed">
        ▶<noscript>…</noscript>
         <img decoding="async" width="800" height="800" src=
         "https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Jinzhou_2019-scaled-800x800-ct.jpg
         " data-src="https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Jinzhou_2019-scaled-800
         x800-ct.jpg" class="attachment-people-thumbnail ls-is-cached lazyloaded" alt="Prof. Jinzhao DU's por
         tfolio">
        ▼<div class="people-info">
           <div class="h5">Prof. Jinzhao DU</div> == $0
         </div>
       </a>
     </div>
   </div>
  ▶<div class="wgl_col-3 people-item">…</div>
  ▶<div class="wgl_col-3 people-item">…</div>
```

Learning about HTML

And even the link to their photos (see this link for example).



```
▼<div class="people-card fadeInUp animated" data-animate="fadeInUp"> flex
  ▼<a href="https://www.hkubs.hku.hk/people/jinzhao-du/" class="el-processed">
    ▶<noscript> ⋯ </noscript>
     <img decoding="async" width="800" height="800" src=
     "https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Jinzhou_2019-scaled-800x800-ct.jpg
     " data-src="https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Jinzhou_2019-scaled-800
     x800-ct.jpg" class="attachment-people-thumbnail ls-is-cached lazyloaded" alt="Prof. Jinzhao DU's por
     tfolio">
    ▼<div class="people-info">
       <div class="h5">Prof. Jinzhao DU</div> == $0
     </div>
   </a>
 </div>
</div>
▶<div class="wgl_col-3 people-item"> ⋯ </div>
▶<div class="wgl_col-3 people-item"> ⋯ </div>
```

Webscraping

Suppose that you want to download the names of each individual marketing faculty, what should you do?

First, you need to get the HTML for the webpage.

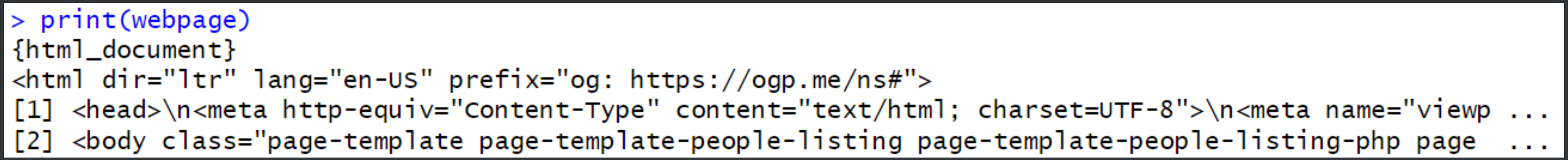
Second, you need to analyze the HTML to get the desired information --- this is much more difficult.

Webscraping

```r
install.packages("rvest")
library(rvest)
url = "https://www.hkubs.hku.hk/people/faculty?pg=1&staff_type=faculty&subject_area=marketing&track=all"
webpage = read_html(url, encoding = "UTF-8")
print(webpage)
```

Webscraping

Now, you get the HTML source file here. The next thing you need to do it to understand the HTML file, which is very challenging.



```
> print(webpage)
{html_document}
<html dir="ltr" lang="en-US" prefix="og: https://ogp.me/ns#">
[1] <head>\n<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">\n<meta name="viewp ...
[2] <body class="page-template page-template-people-listing page-template-people-listing-php page  ...
```

Webscraping

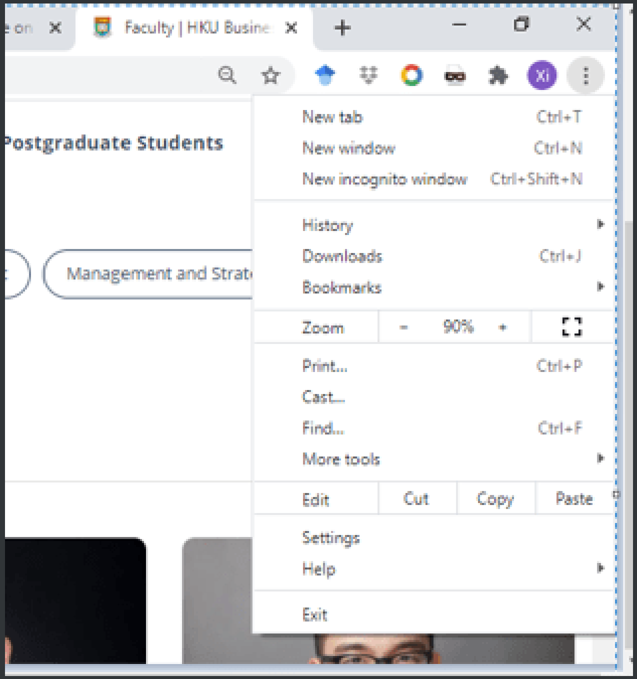
To better understand the HTML code, you are strongly recommended to use Chrome as your browser.

Chrome allows you to check the HTML code in a much more convenient matter.

Check HTML with Chrome

Open the webpage in your Chrome browser.

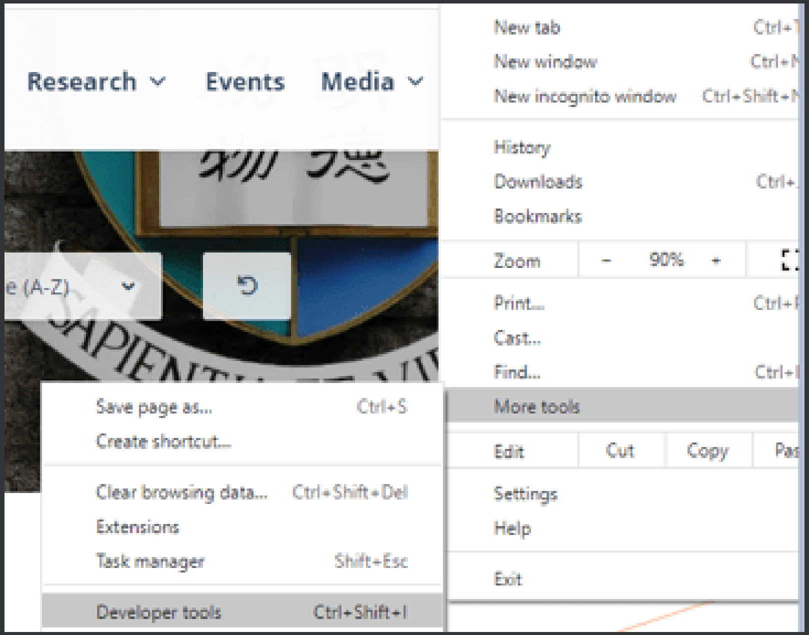
Click the upper right Chome setting button of your browser and you will be directed here.



Check HTML with Chrome

Choose “More tools”…

Choose “Developer tools”…

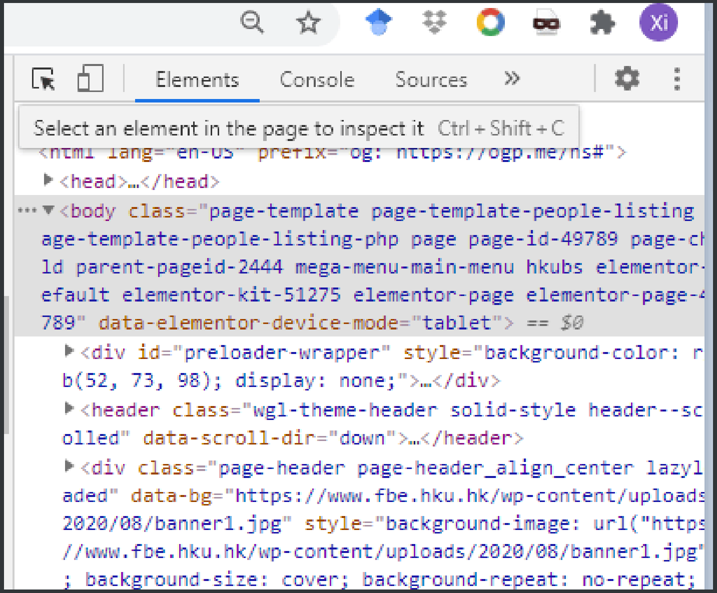


Check HTML with Chrome

Click the button and you will get to “select an element in the page to inspect it”.

Alternatively, use “Ctrl + Shift + C.”

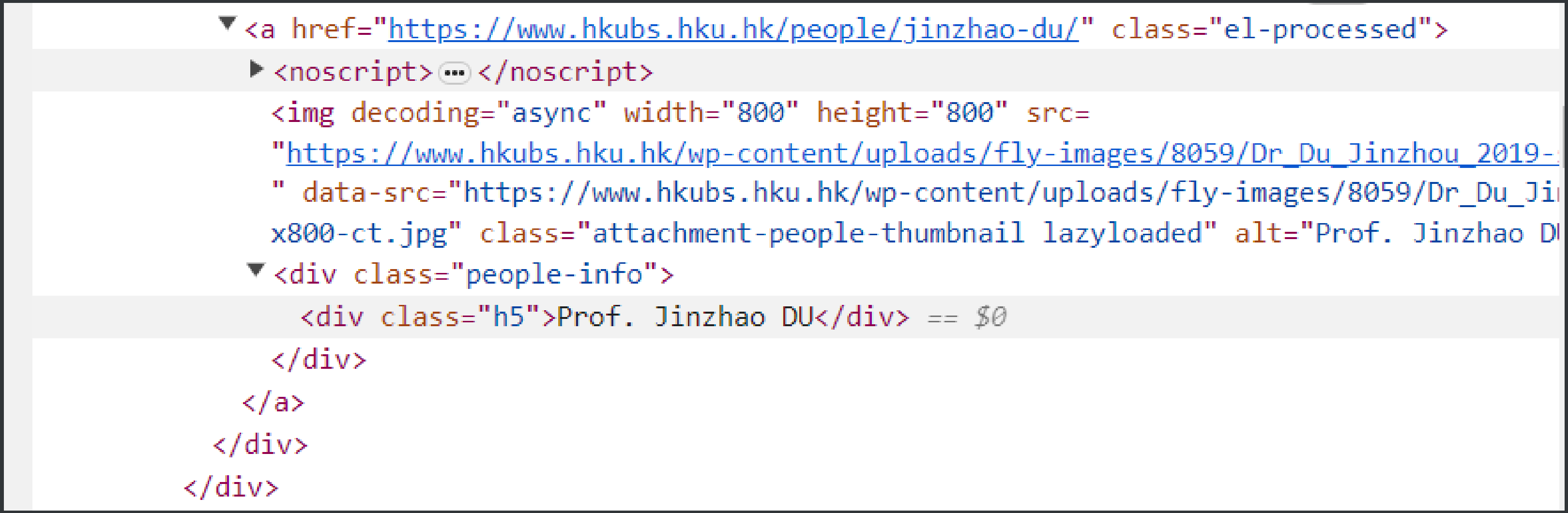
If needed, also click the button to switch between desktop and mobile version.


Xi
Elements
Console
Sources
»
Select an element in the page to inspect it Ctrl + Shift + C

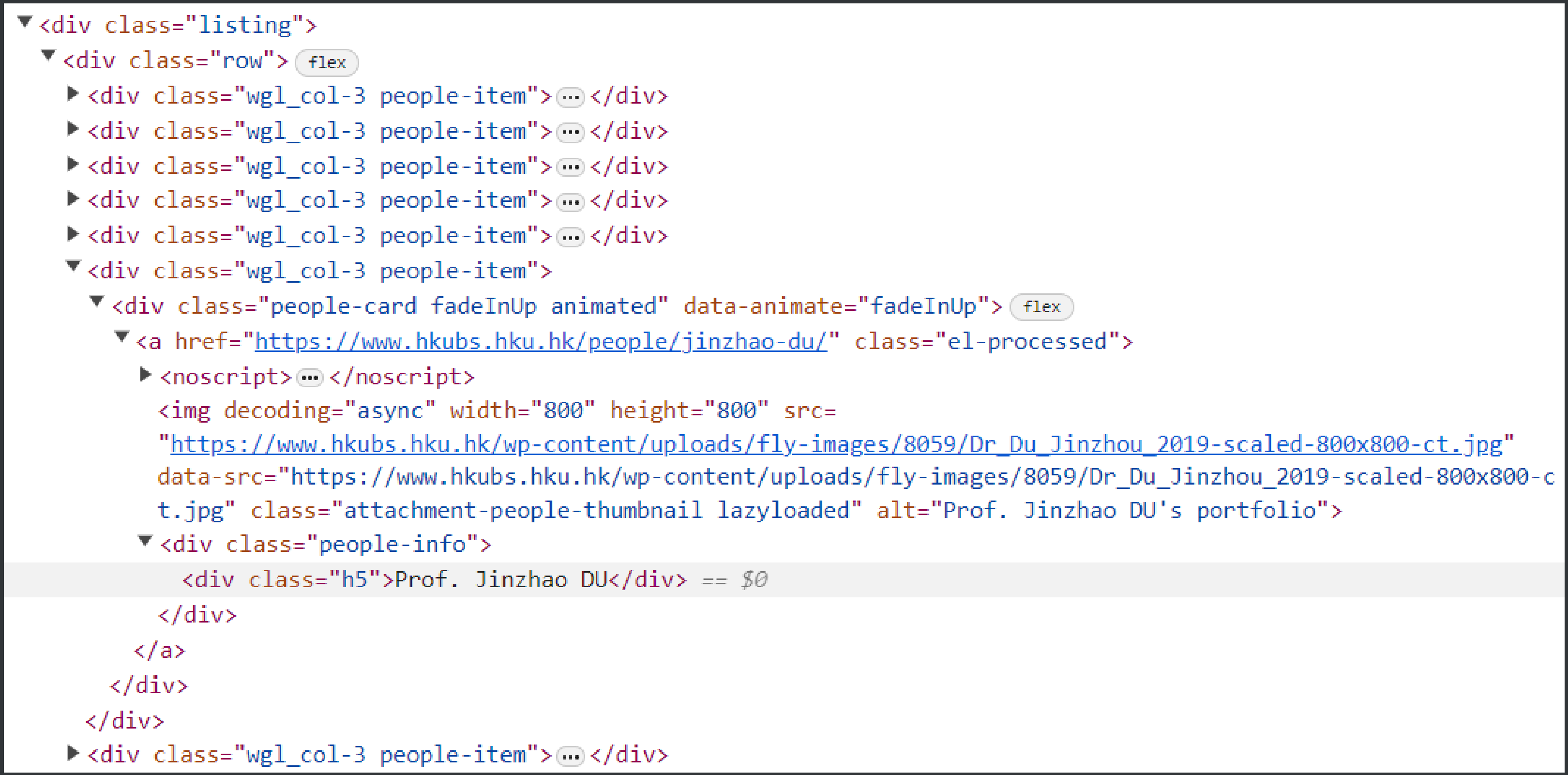
```
<html lang="en-US" prefix="og: https://ogp.me/ns#">
 ▶<head>…</head>
…▼<body class="page-template page-template-people-listing
  age-template-people-listing-php page page-id-49789 page-ch
  ld parent-pageid-2444 mega-menu-main-menu hkubs elementor-
  efault elementor-kit-51275 elementor-page elementor-page-4
  789" data-elementor-device-mode="tablet"> == $0
   ▶<div id="preloader-wrapper" style="background-color: r
   b(52, 73, 98); display: none;">…</div>
   ▶<header class="wgl-theme-header solid-style header--sc
   olled" data-scroll-dir="down">…</header>
   ▶<div class="page-header page-header_align_center lazyl
   aded" data-bg="https://www.fbe.hku.hk/wp-content/uploads
   2020/08/banner1.jpg" style="background-image: url("https
   //www.fbe.hku.hk/wp-content/uploads/2020/08/banner1.jpg"
   ; background-size: cover; background-repeat: no-repeat;
```

Check HTML with Chrome

Take Prof. Du's information as an example. You can see his name appears here in the HTML code. But what does this mean?

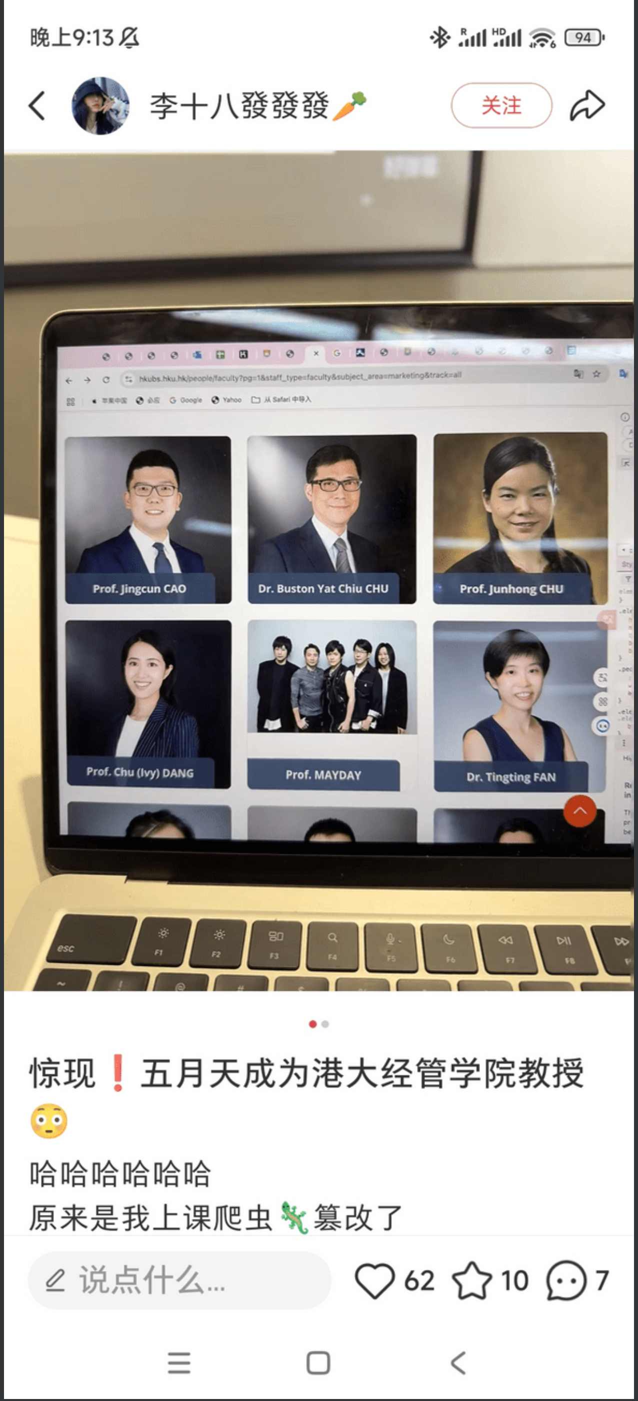


```
      ▼<a href="https://www.hkubs.hku.hk/people/jinzhao-du/" class="el-processed">
        ▶<noscript>…</noscript>
         <img decoding="async" width="800" height="800" src=
         "https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Jinzhou_2019-
         " data-src="https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Ji
         x800-ct.jpg" class="attachment-people-thumbnail lazyloaded" alt="Prof. Jinzhao D
        ▼<div class="people-info">
           <div class="h5">Prof. Jinzhao DU</div> == $0
         </div>
       </a>
     </div>
   </div>
```



```
▼<div class="listing">
  ▼<div class="row"> flex
    ▶<div class="wgl_col-3 people-item"> ⋯ </div>
    ▶<div class="wgl_col-3 people-item"> ⋯ </div>
    ▶<div class="wgl_col-3 people-item"> ⋯ </div>
    ▶<div class="wgl_col-3 people-item"> ⋯ </div>
    ▶<div class="wgl_col-3 people-item"> ⋯ </div>
    ▼<div class="wgl_col-3 people-item">
      ▼<div class="people-card fadeInUp animated" data-animate="fadeInUp"> flex
        ▼<a href="https://www.hkubs.hku.hk/people/jinzhao-du/" class="el-processed">
          ▶<noscript> ⋯ </noscript>
           <img decoding="async" width="800" height="800" src=
           "https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Jinzhou_2019-scaled-800x800-ct.jpg"
           data-src="https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Jinzhou_2019-scaled-800x800-c
           t.jpg" class="attachment-people-thumbnail lazyloaded" alt="Prof. Jinzhao DU's portfolio">
          ▼<div class="people-info">
             <div class="h5">Prof. Jinzhao DU</div> == $0
           </div>
         </a>
       </div>
     </div>
    ▶<div class="wgl_col-3 people-item"> ⋯ </div>
```

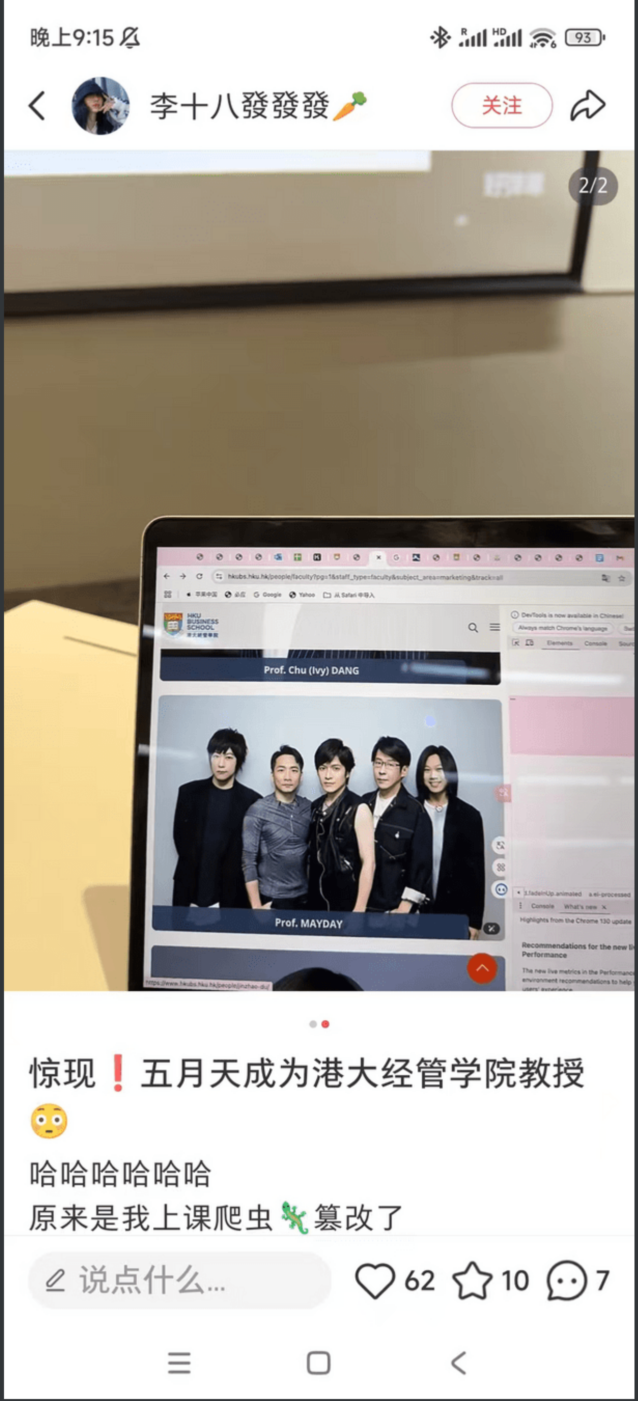
Secret: Changing the webpage



惊现❗五月天成为港大经管学院教授😳

哈哈哈哈哈哈

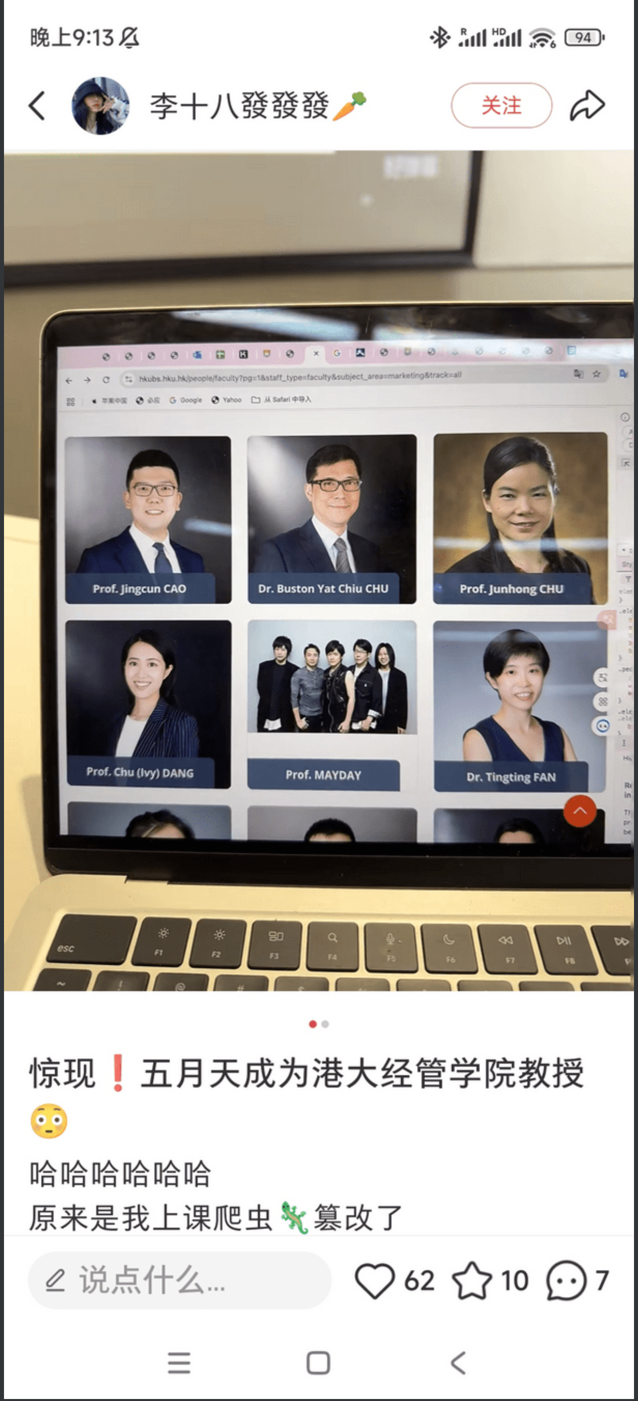
原来是我上课爬虫🦎篡改了



惊现❗五月天成为港大经管学院教授😳

哈哈哈哈哈哈

原来是我上课爬虫🦎篡改了



惊现❗五月天成为港大经管学院教授😳

哈哈哈哈哈哈

原来是我上课爬虫🦎篡改了

According to Xiaohongshu, this student joined McDonald's.

https://www.youtube.com/embed/u0OeZfIfBRI?enablejsapi=1

https://www.youtube.com/emb
ed/u0OeZfIfBRI?enablejsapi=1

Understanding HTML

Here, the name information is within a “div” node.

And this node belongs to a “div” node.

This “div” node further belongs to another “a” node.

And so on….

We call this is “path”: …div/div/div/a/div/div

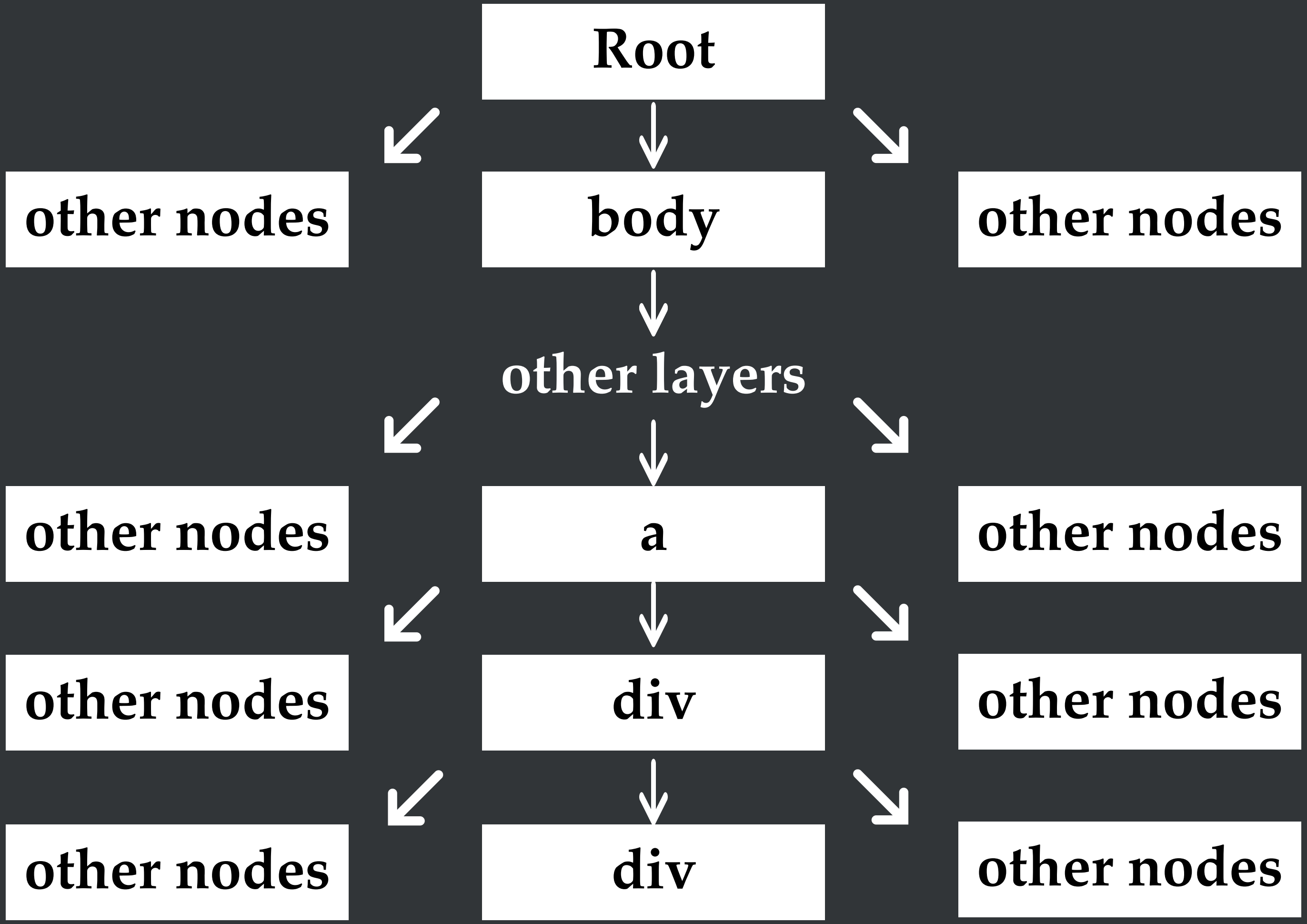
Understanding HTML

You can see that we have various types of nodes, including “div”, “a”, and “img”. You may wonder, “what do these types mean?”

Here, these types are called “tag”. For example, an “img” tag is used to mark up an image in the HTML language.

For detailed information, check here.

Understanding HTML



Understanding HTML

This is something like your home address:

We have something like…
Country/Province/City/District/Street/Building/Floor/Room

The path helps us locate nodes and find the content of the nodes.

Understanding HTML

However, unlike your home address, here each node does not have its name.

For example, we know it is a “div” node (not an “a” node) but there may be multiple “div” nodes.

My building is in a street (not an avenue or road) but there may be multiple streets here.

Understanding HTML

Let’s get all “div” nodes. This can be done by running this:

```r
nodes <- html_nodes(webpage,xpath = '//div')
```

You can see that in total we have 253 “div” nodes.

```r
print(length(nodes))
```

Understanding HTML

We want to make the path more accurate to pin down to the "div" nodes that we are interested in. That is, we want to remove other unrelated "div" nodes.

We can do this by putting more restrictions on the path.

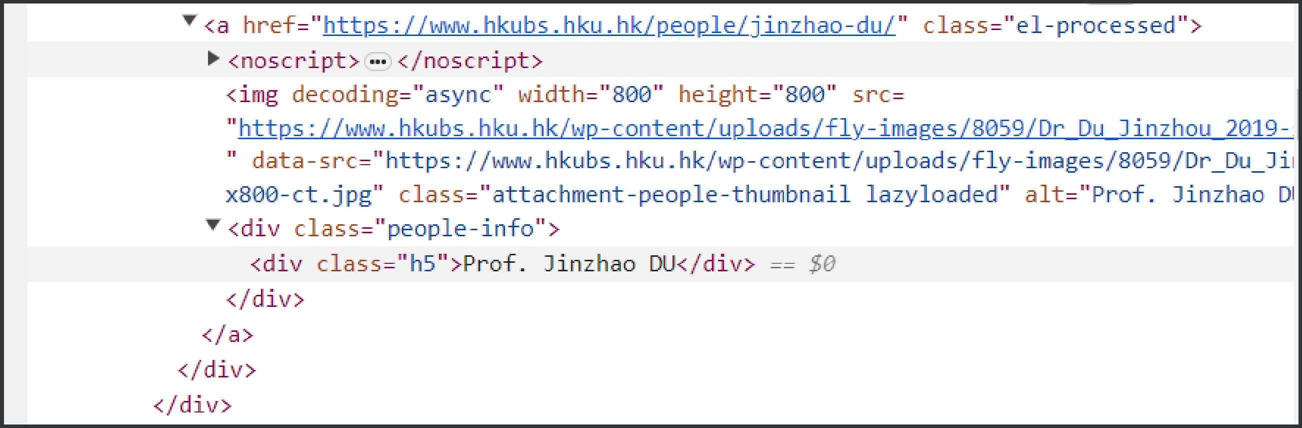
https://www.youtube.com/embed/vNOyRZIkC7o?enablejsapi=1

Understanding HTML

Consider the following code:

```r
nodes <- html_nodes(webpage,xpath = '//div/div')
print(length(nodes))
```

Here we restrict the parent of the "div" node must also be a "div" node. Now, we have 199 nodes --- still too many.



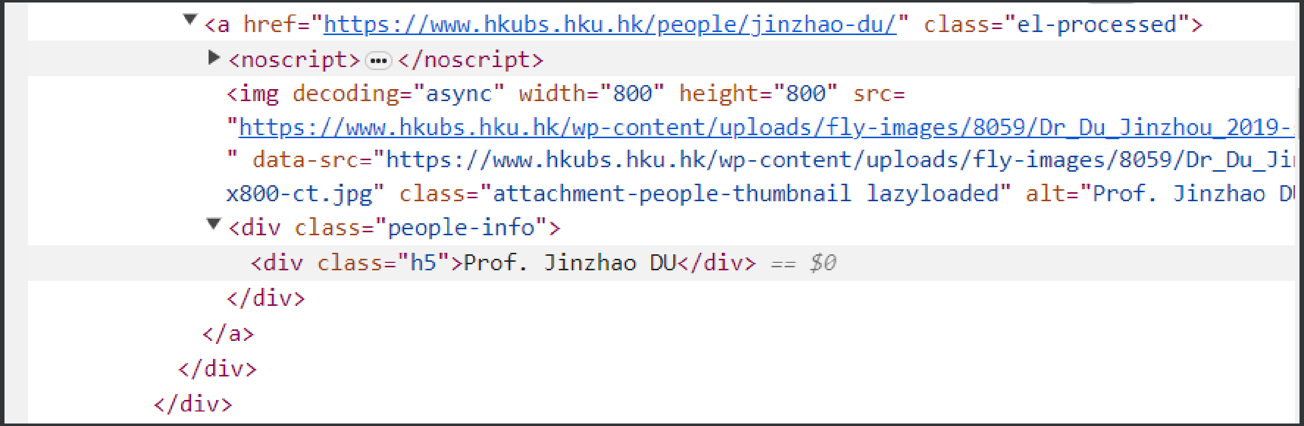
```html
▼<a href="https://www.hkubs.hku.hk/people/jinzhao-du/" class="el-processed">
  ▶<noscript>…</noscript>
   <img decoding="async" width="800" height="800" src=
   "https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Jinzhou_2019-
   " data-src="https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Ji
   x800-ct.jpg" class="attachment-people-thumbnail lazyloaded" alt="Prof. Jinzhao D
  ▼<div class="people-info">
     <div class="h5">Prof. Jinzhao DU</div> == $0
   </div>
  </a>
 </div>
</div>
```

Understanding HTML

Consider the following code:

```r
nodes <- html_nodes(webpage,xpath = '//div[@class="people-info"]/div')
print(length(nodes))
```

Here we restrict the parent of the "div" node must also be a "div" node. Moreover, the its parent node must have a class attribute will is called "people-info."



```
▼<a href="https://www.hkubs.hku.hk/people/jinzhao-du/" class="el-processed">
  ▶<noscript>⋯</noscript>
   <img decoding="async" width="800" height="800" src=
   "https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Jinzhou_2019-
   " data-src="https://www.hkubs.hku.hk/wp-content/uploads/fly-images/8059/Dr_Du_Ji
   x800-ct.jpg" class="attachment-people-thumbnail lazyloaded" alt="Prof. Jinzhao D
  ▼<div class="people-info">
     <div class="h5">Prof. Jinzhao DU</div> == $0
   </div>
  </a>
 </div>
</div>
```

Understanding HTML

Now, we only have 16 div nodes selected. These are actually all HKU marketing faculties. Let us print their names:

```r
nodes <- html_nodes(webpage,xpath = '//div[@class="people-info"]/div')
for (node in nodes)
  print(html_text(node))
```

Understanding HTML

You can also use other refinement to select the nodes that you are looking for. For example, the following codes work as well:

```r
nodes <- html_nodes(webpage,xpath = '//div[@class="h5"]')
for (node in nodes)
  print(html_text(node))
```

The complete code is here.

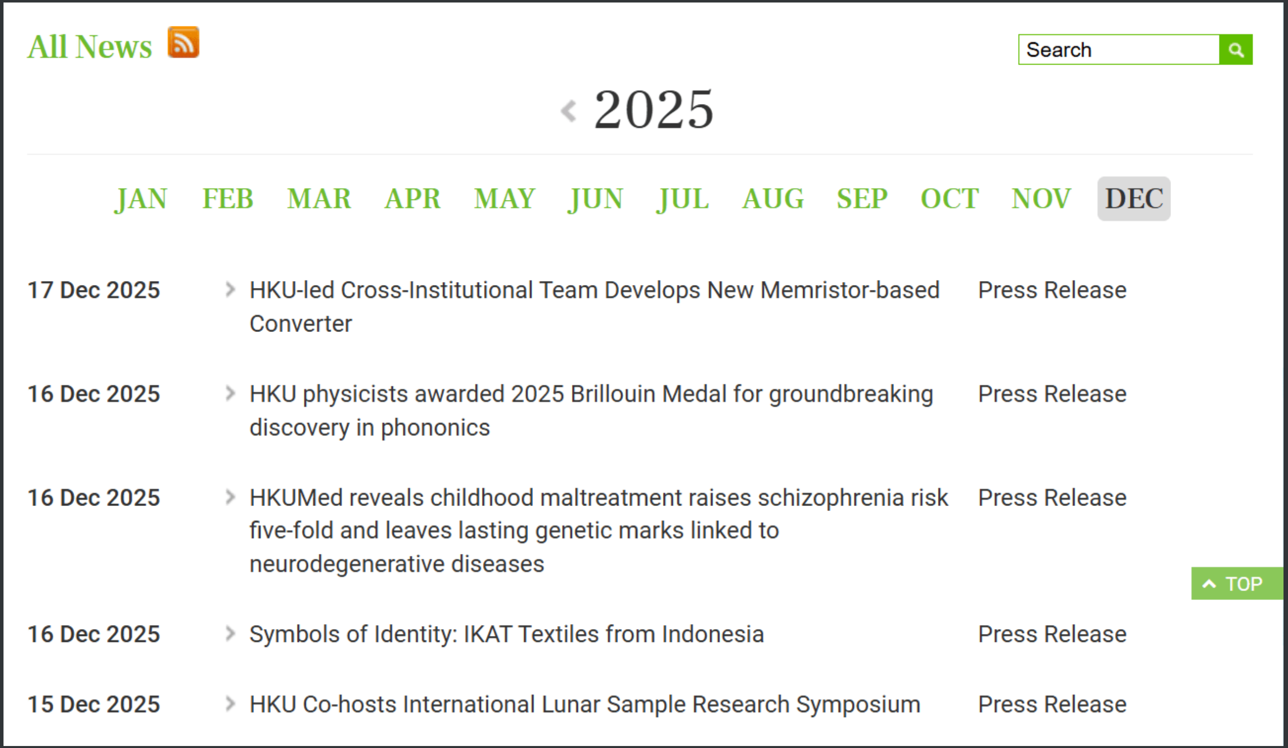
```r
library(rvest)
url = "https://www.hkubs.hku.hk/people/faculty?pg=1&staff_type=faculty&subject_area=marketing&track=all"
webpage = read_html(url, encoding = "UTF-8")
nodes <- html_nodes(webpage,xpath = '//div[@class="h5"]')
for (node in nodes)
  print(html_text(node))
```

Exercise

Great! You know have a sense of how to scrape data from the web. It is very preliminary, and you will need a lot more exercises. Let us try the following exercise.

Exercise

HKU makes press announcements on its official news webpage: https://hku.hk/press/all/



All News

Search

2025

JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC

17 Dec 2025	HKU-led Cross-Institutional Team Develops New Memristor-based Converter	Press Release
16 Dec 2025	HKU physicists awarded 2025 Brillouin Medal for groundbreaking discovery in phononics	Press Release
16 Dec 2025	HKUMed reveals childhood maltreatment raises schizophrenia risk five-fold and leaves lasting genetic marks linked to neurodegenerative diseases	Press Release
16 Dec 2025	Symbols of Identity: IKAT Textiles from Indonesia	Press Release
15 Dec 2025	HKU Co-hosts International Lunar Sample Research Symposium	Press Release

TOP

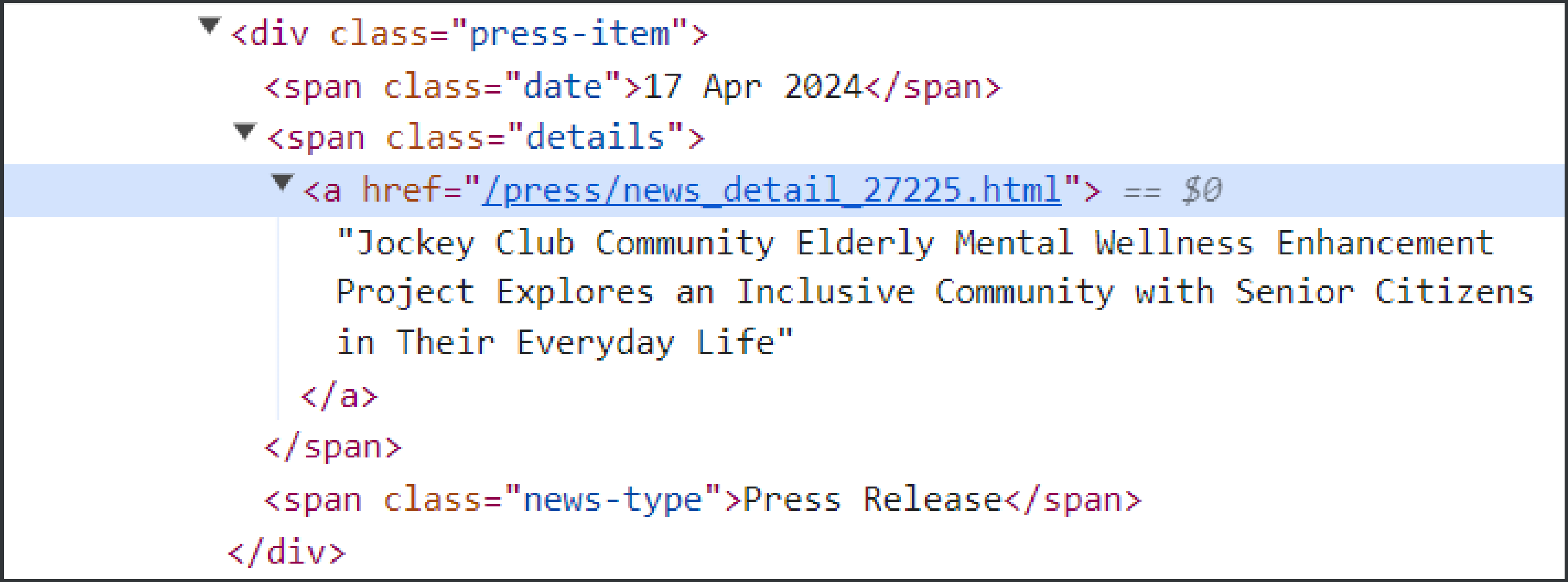
Exercise

Try to download the titles of these press articles!

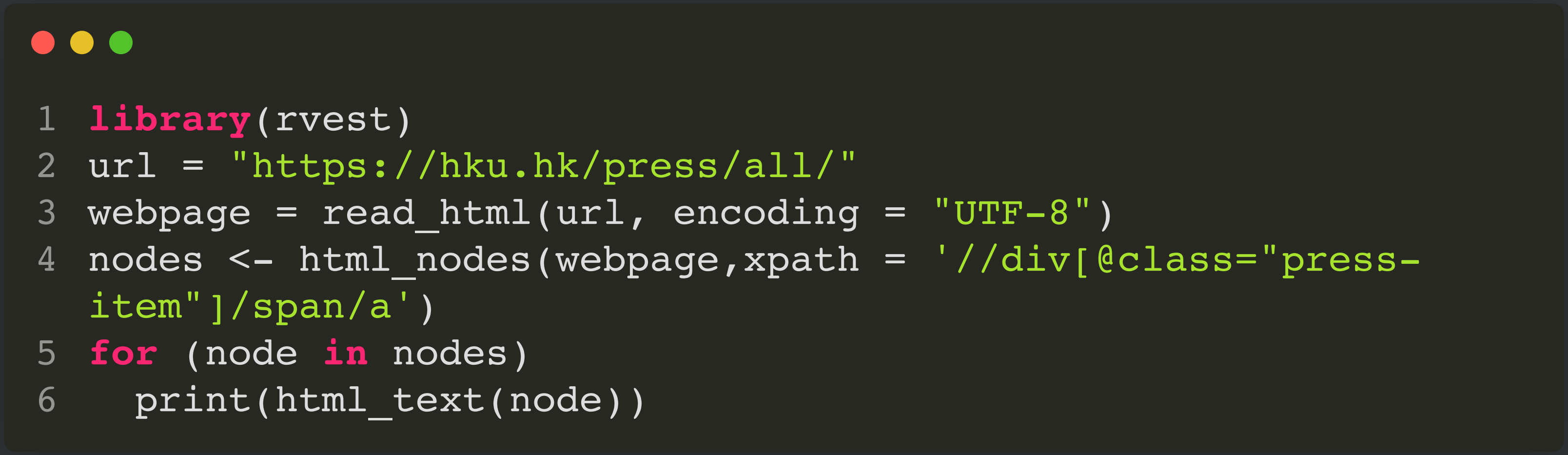
URL: https://hku.hk/press/all/

Exercise

First, let us scrape the titles. We must understand the corresponding HTML code to scrape the data.



```
▼<div class="press-item">
   <span class="date">17 Apr 2024</span>
 ▼<span class="details">
   ▼<a href="/press/news_detail_27225.html"> == $0
       "Jockey Club Community Elderly Mental Wellness Enhancement
       Project Explores an Inclusive Community with Senior Citizens
       in Their Everyday Life"
     </a>
   </span>
   <span class="news-type">Press Release</span>
 </div>
```



```r
library(rvest)
url = "https://hku.hk/press/all/"
webpage = read_html(url, encoding = "UTF-8")
nodes <- html_nodes(webpage,xpath = '//div[@class="press-item"]/span/a')
for (node in nodes)
  print(html_text(node))
```

Challenge:

Can anyone use AI to produce the code?

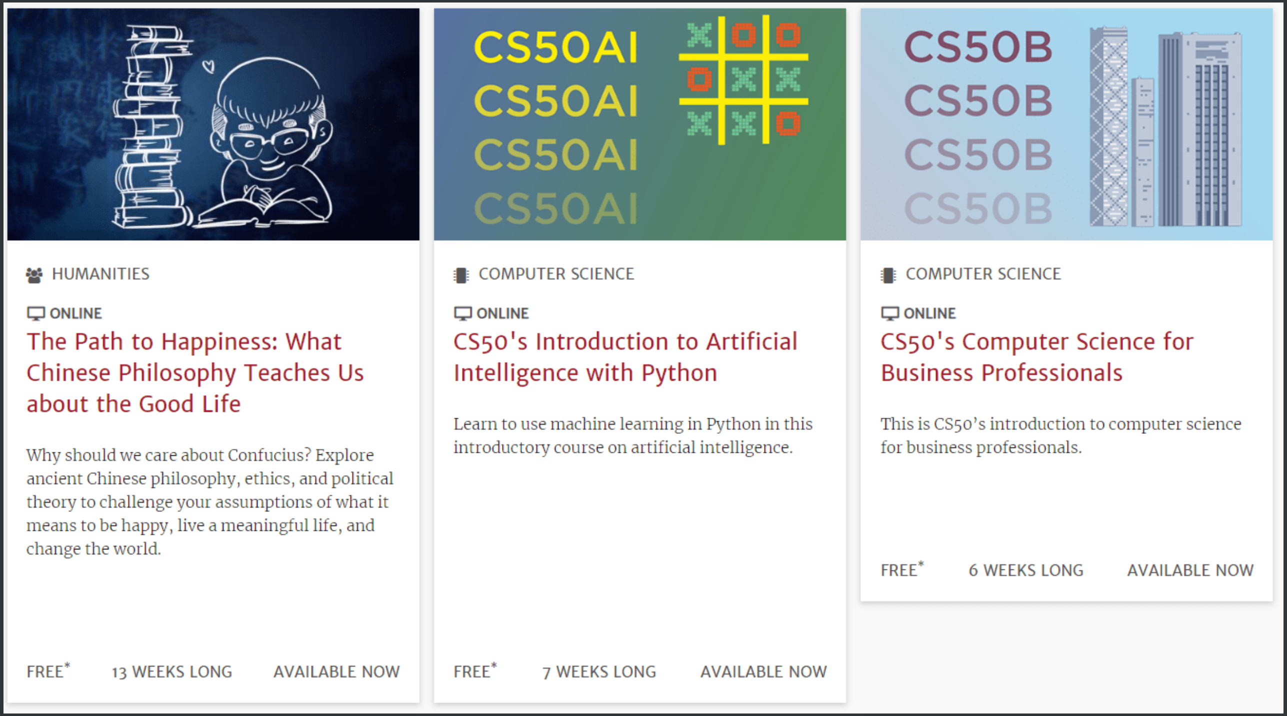
Using AI to help you find xpath:

I want to scrape all titles like HKU-led Cross-Institutional Team Develops New Memristor-based Converter. Show me the xpath to locate the titles. Here is the HTML: ....

```
//div[@class='press-item']/span[@class='details']/a/text()
```

Exercise

Now, let us visit the Harvard School of Professional Learning: https://pll.harvard.edu/trending



HUMANITIES
ONLINE
The Path to Happiness: What Chinese Philosophy Teaches Us about the Good Life

Why should we care about Confucius? Explore ancient Chinese philosophy, ethics, and political theory to challenge your assumptions of what it means to be happy, live a meaningful life, and change the world.

FREE* 13 WEEKS LONG AVAILABLE NOW

COMPUTER SCIENCE
ONLINE
CS50's Introduction to Artificial Intelligence with Python

Learn to use machine learning in Python in this introductory course on artificial intelligence.

FREE* 7 WEEKS LONG AVAILABLE NOW

COMPUTER SCIENCE
ONLINE
CS50's Computer Science for Business Professionals

This is CS50's introduction to computer science for business professionals.

FREE* 6 WEEKS LONG AVAILABLE NOW

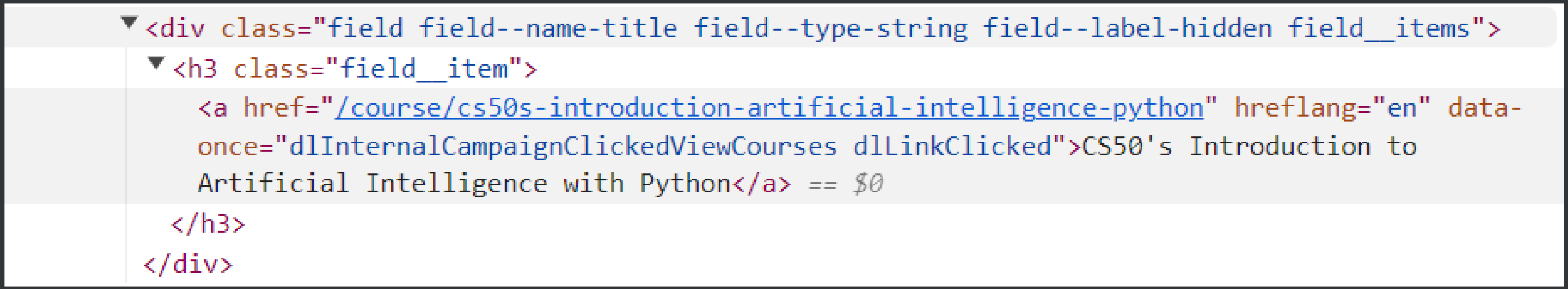
Exercise

In this exercise, we attempt to scrape the course titles, e.g., “CS50's Introduction to Artificial Intelligence with Python”

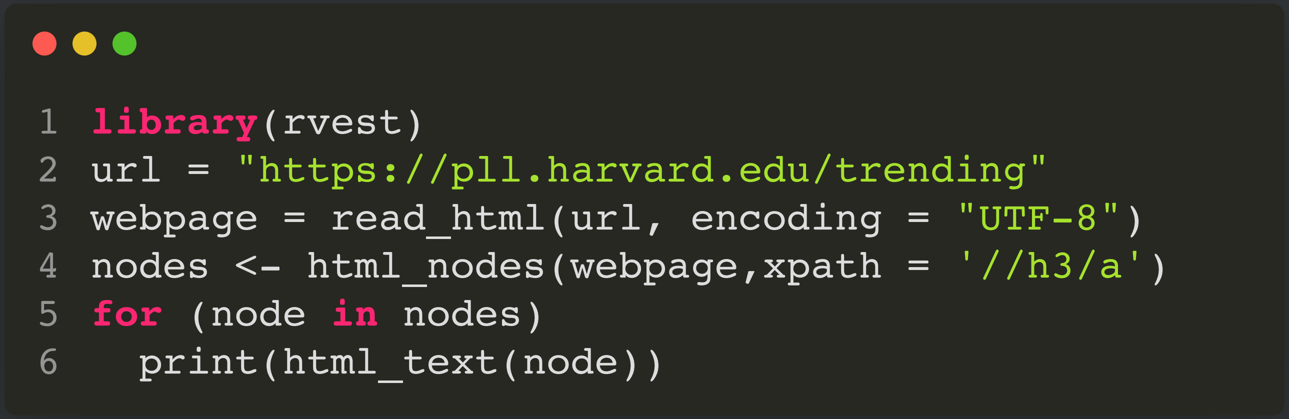
Try this exercise yourself!

Exercise

First, we identify the root of each individual course. We need to inspect the HTML code first.



```
▼<div class="field field--name-title field--type-string field--label-hidden field__items">
  ▼<h3 class="field__item">
     <a href="/course/cs50s-introduction-artificial-intelligence-python" hreflang="en" data-
     once="dlInternalCampaignClickedViewCourses dlLinkClicked">CS50's Introduction to
     Artificial Intelligence with Python</a> == $0
   </h3>
 </div>
```



```r
library(rvest)
url = "https://pll.harvard.edu/trending"
webpage = read_html(url, encoding = "UTF-8")
nodes <- html_nodes(webpage,xpath = '//h3/a')
for (node in nodes)
  print(html_text(node))
```

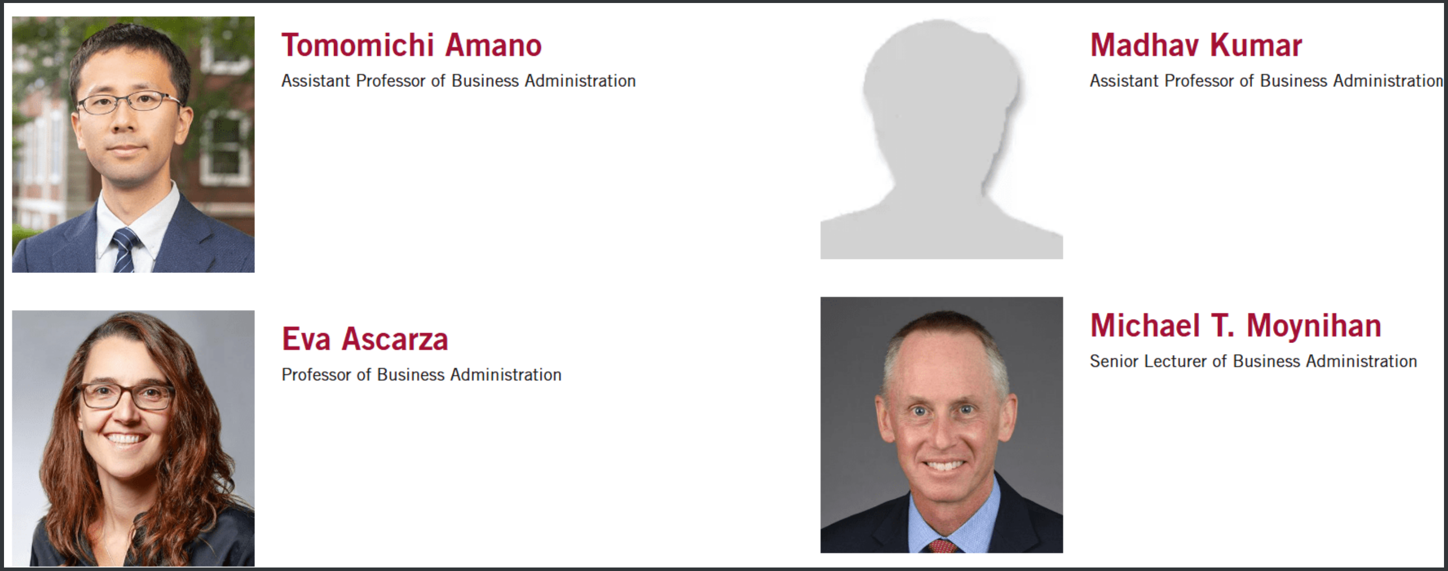
Scraping Images

Previously, we have discussed how to scrape text information from a website using a web scraper.

Now, let us consider scraping images from the web.

Scraping Images

Let us study the Harvard marketing faculty webpage:



Tomomichi Amano
Assistant Professor of Business Administration

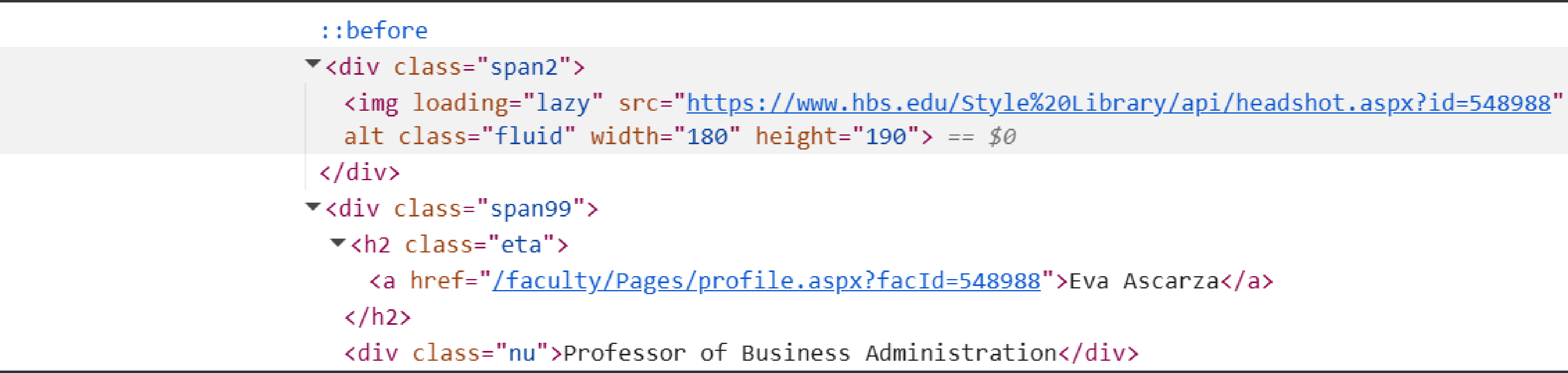
Madhav Kumar
Assistant Professor of Business Administration

Eva Ascarza
Professor of Business Administration

Michael T. Moynihan
Senior Lecturer of Business Administration

Scraping Images

You can find a link to each photo (in “src” attribute):



```
        ::before
      ▼<div class="span2">
         <img loading="lazy" src="https://www.hbs.edu/Style%20Library/api/headshot.aspx?id=548988"
         alt class="fluid" width="180" height="190"> == $0
        </div>
      ▼<div class="span99">
        ▼<h2 class="eta">
           <a href="/faculty/Pages/profile.aspx?facId=548988">Eva Ascarza</a>
         </h2>
         <div class="nu">Professor of Business Administration</div>
```

Scraping Images

Once you get the link, you will have access to the photo.

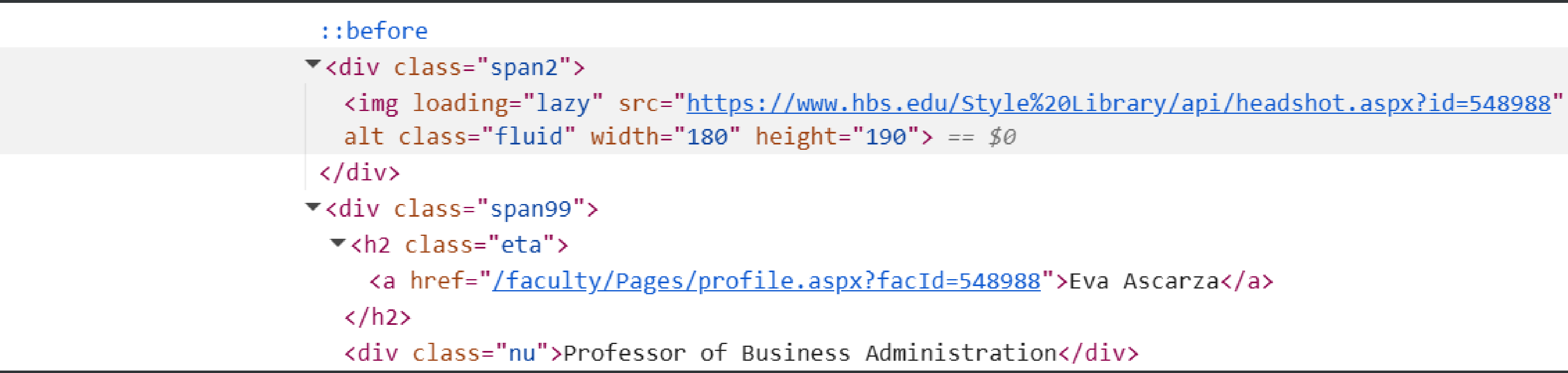
So, our first step to get the link information.

Scraping Images

```r
library(rvest)
url = "https://www.hbs.edu/faculty/units/marketing/Pages/faculty.aspx"
webpage = read_html(url, encoding = "UTF-8")
image_nodes <- html_nodes(webpage, xpath = '//img[@width="180"]')
print(length(image_nodes))
```

Scraping Images

But that’s not enough. We not only want to get the nodes, but also need the link to each of the nodes. The link appears in the “src” attribute.



```
::before
▼<div class="span2">
  <img loading="lazy" src="https://www.hbs.edu/Style%20Library/api/headshot.aspx?id=548988"
  alt class="fluid" width="180" height="190"> == $0
</div>
▼<div class="span99">
 ▼<h2 class="eta">
    <a href="/faculty/Pages/profile.aspx?facId=548988">Eva Ascarza</a>
  </h2>
  <div class="nu">Professor of Business Administration</div>
```

Scraping Images

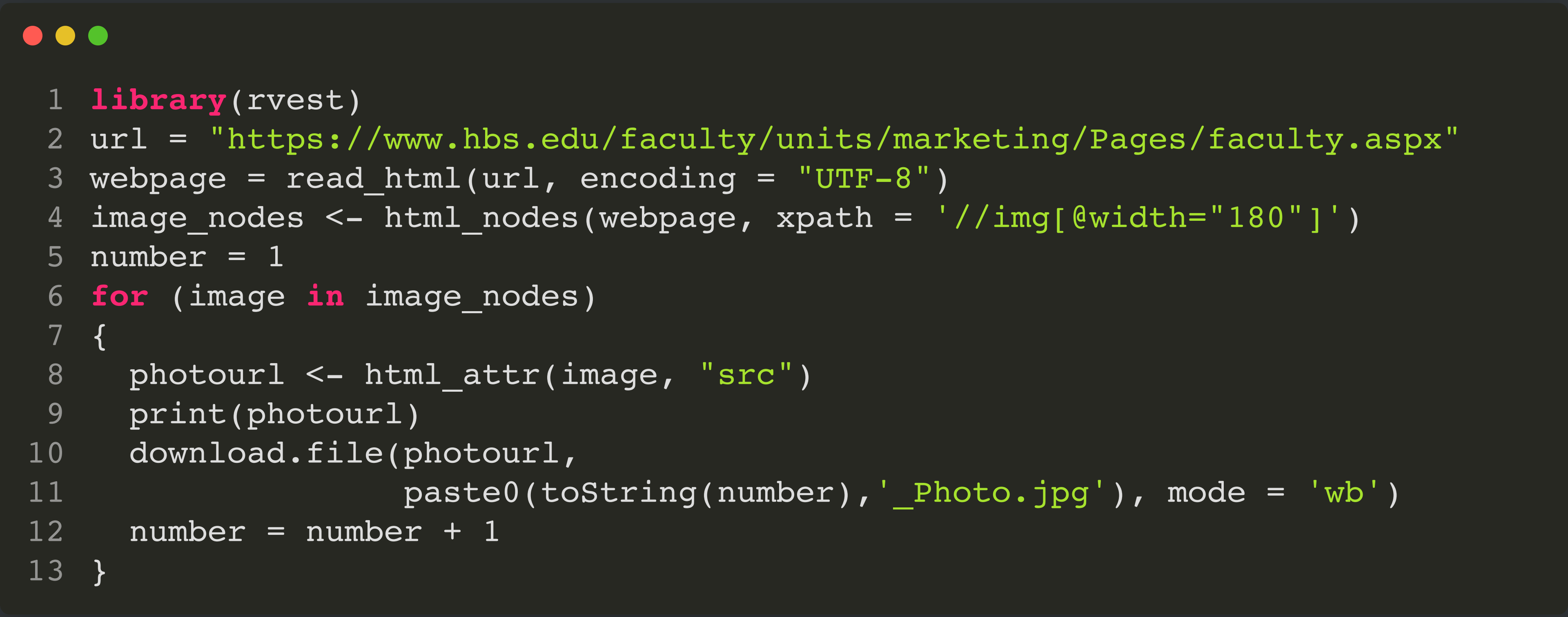
But that’s not enough. We not only want to get the nodes, but also need the link to each of the nodes. The link appears in the “src” attribute.

```r
image_nodes <- html_nodes(webpage, xpath = '//img[@width="180"]')
for (image in image_nodes)
{
  photourl <- html_attr(image, "src")
  print(photourl)
}
```

Scraping Images

```r
number = 1
for (image in image_nodes)
{
  photourl <- html_attr(image, "src")
  print(photourl)
  download.file(photourl,
                      paste0(toString(number),'_Photo.jpg'), mode = 'wb')
  number = number + 1
}
```

The compete code is here.



```r
library(rvest)
url = "https://www.hbs.edu/faculty/units/marketing/Pages/faculty.aspx"
webpage = read_html(url, encoding = "UTF-8")
image_nodes <- html_nodes(webpage, xpath = '//img[@width="180"]')
number = 1
for (image in image_nodes)
{
  photourl <- html_attr(image, "src")
  print(photourl)
  download.file(photourl,
                paste0(toString(number),'_Photo.jpg'), mode = 'wb')
  number = number + 1
}
```

Static vs. Dynamic Websites

https://www.youtube.com/embed/hlg6q6OFoxQ?enablejsapi=1

Dynamic Websites

What we learned in today’s class works well for static websites. But it does not work equally well on dynamic websites. If you want to scrape data from a dynamic website, you may need to use some more advanced tools.

Thank you!

Enjoy scraping!