SI206 W23 Discussion 9

HTML and BeautifulSoup

HTML

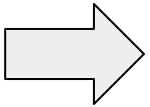
- HTML is the standard markup language for Web pages.
- HTML elements are delineated by tags, written using angle brackets.

Example

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```



This is a Heading

This is a paragraph.

Requests

- Make a request

```
>>> import requests
>>> url= "https://www.google.com/"
>>> r = requests.get(url)
>>> print(r)
<Response [200]>
```

- Response content

```
>>> r.text
'<!doctype html><html itemscope=""
itemtype="http://schema.org/WebPage" lang="en"><head><meta
content="Search the world\'s information, ...
>>> r.content
b'<!doctype html><html itemscope=""
itemtype="http://schema.org/WebPage" lang="en"><head><meta
content="Search the world\'s information, ...
```

BeautifulSoup for scraping

To use the BeautifulSoup module for scraping, you need to create the BeautifulSoup object.

There are 3 steps to it:

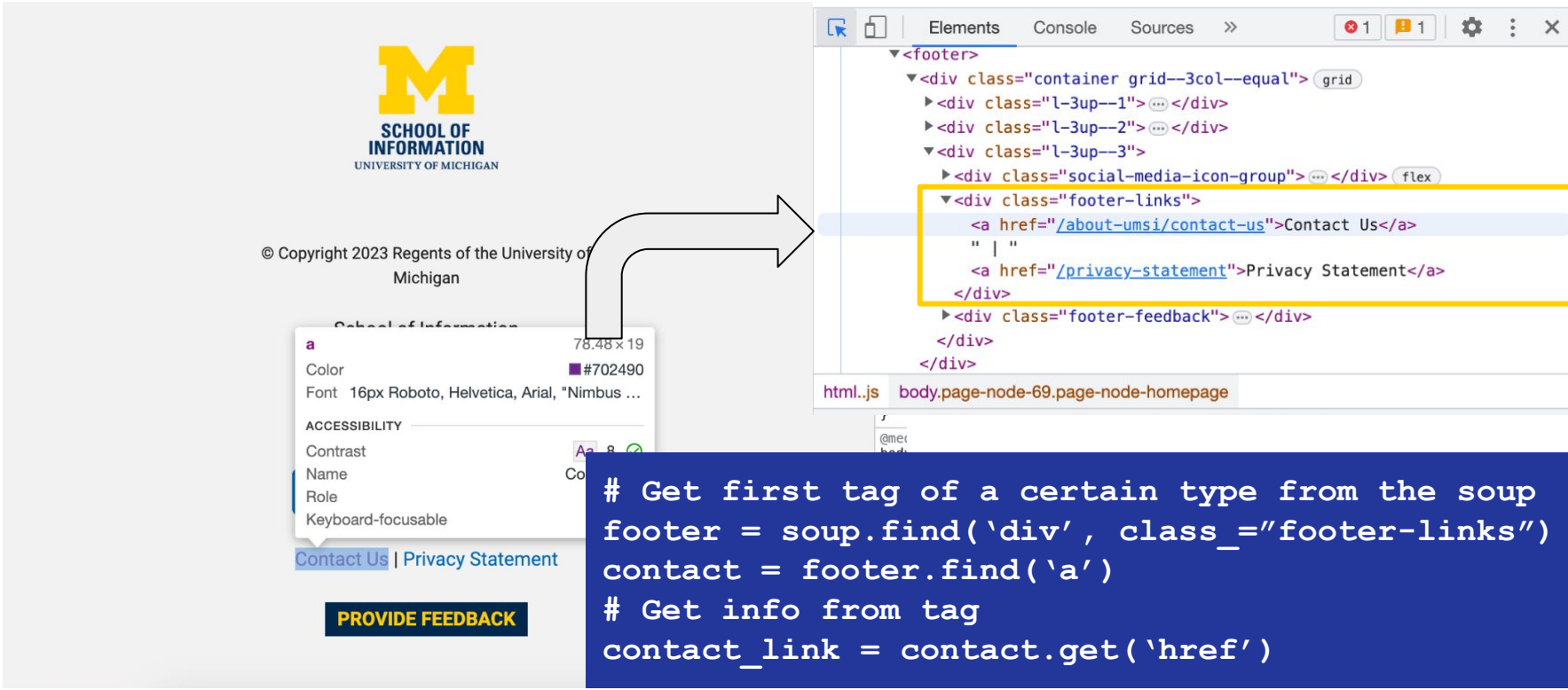
1. Create a variable that stores the url of website
2. Get the data from the url i.e. `r = requests.get(url)`
3. Create a soup object using the data, i.e.

```
soup = BeautifulSoup(r.text, 'html.parser')
```

Things to keep in mind with BeautifulSoup

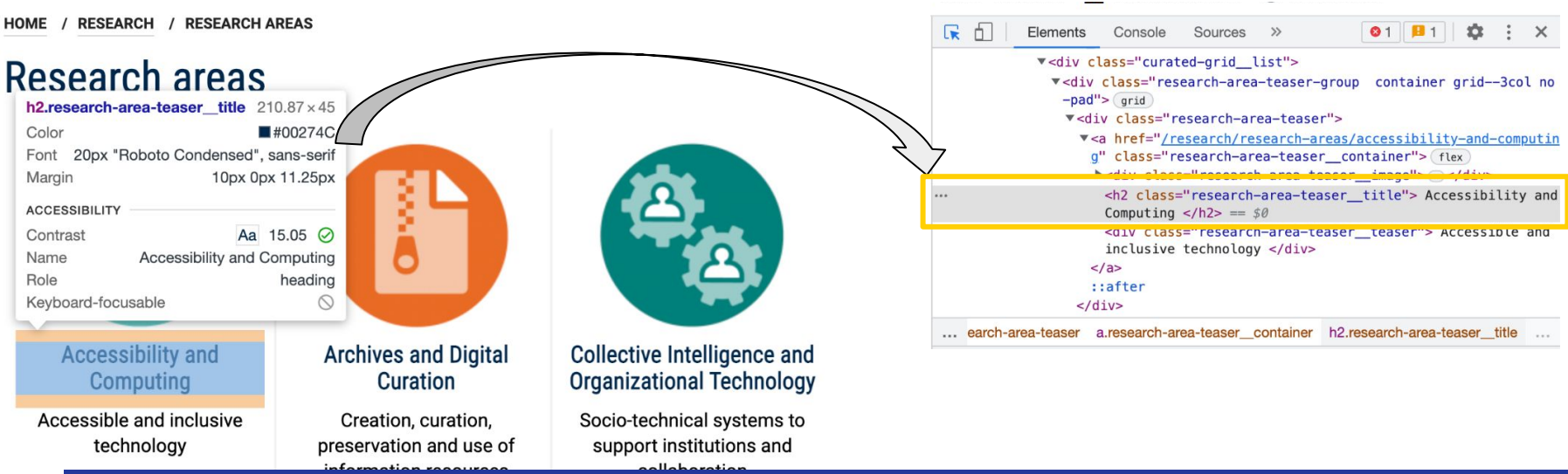
1. `soup.find('tag')` will return **the first tag** that matches
2. `soup.find_all('tag')` will return **a list of all the tags** that match
3. You can use find and find_all on the tag objects to find children tags!
4. Use the `tag_object.attrs` to obtain a dictionary of the attributes in a tag object
5. Use the `tag_object.get(attr_name)` to get a specific attribute

Getting info using dev tools - `find()`




```
# Get first tag of a certain type from the soup
footer = soup.find('div', class_="footer-links")
contact = footer.find('a')
# Get info from tag
contact_link = contact.get('href')
```

Getting info using dev tools - `find_all()`




```
# Get Get all tags of a certain type from the soup
research_areas = soup.find_all("h2", class_="research-area-teaser__title")
research_areas_list = []
for tag in research_areas:
    # Get info from tag
    research_areas_list.append(tag.text)
```

Assignment: scraping Wikipedia

We will use BeautifulSoup to get some data from

https://en.wikipedia.org/wiki/University_of_Michigan

Task 1: Create a BeautifulSoup object.

Task 2: Get the URL that links to list of *American universities with Olympic medal wins*. The clickable link can be found near the end of the introduction of the University of Michigan page.

Assignment: scraping Wikipedia

Task 3: Get the details of the table in the section titled "Organization and administration" in the University of Michigan Wikipedia page. Get all the *college/school names and the year they were founded* and organize that information into key-value pairs of a dictionary.

Organize the details into a dictionary as shown below:

```
{'Literature, Science, and the Arts': '1841',
'Medicine': '1850', ... 'Kinesiology': '1984'}
```

Tips

1. We can filter tags by their attributes by passing additional arguments to the **`find()`** or **`find_all()`** methods. For instance, if I only want to get a tags that link to Google, I could do:
   a. **`soup.find_all('a', href=' https://www.google.com')`**
2. Remember that you need to use `class_` instead of `class` in **`find()`** or **`find_all()`** because class is a reserved word in Python.
3. When trying to decide how you want to grab a particular tag, remember that in HTML a class is typically assigned to multiple tags while an id is unique.
   a. Sometimes a tag may have multiple classes separated by a space. Do not treat these all as one class.