Web scraping and Covid-19: html

Welcome back. This is the second lesson in our web scraping to build social research data training series. In the first lesson, we looked at the fundamentals of web scraping. So what it is, how we would do it in practice, the reasons why you might engage in it as a social scientist, and some of the limitations and ethical implications of using it as a data collection research method. And this second lesson, which is short and sweet, we're going to look at a particular area of knowledge you need to have in order to engage in web scraping. Specifically, you need to understand how web pages are structured. And recall that web pages contain the content we want to scrape via text or images or videos, tables, or lists, etc. All of that content is stored on a web page, that web page has a certain structure and if we know the structure, we're able to navigate the page and extract the information that we need. So a very quick lesson will just cover the fundamentals of web pages, and the language that structures.

So first and foremost, what is a web page? So it's a document, which can be displayed in a web browser. So if we're manually viewing web pages, we tend to use browsers such as Firefox, Google Chrome, Safari, and quite a number of other ones. So seems a bit basic, but that really is all it is. Web pages are files, they're written in a certain language. And then those files can be accessed and displayed through a web browser. And then a website is simply a collection of web pages that are connected or related in some kind of obvious way. So it's not a random assortment of web pages that constitutes a website. The BBC news website only has webpages relating to news content produced by the BBC, for example. So how then do we get access to those web pages or websites? Well, web pages are stored on something called a web server. A web server or a server more generally, is simply a computer, it's a physical computer, it's stored somewhere in the world, those web pages as files are stored on that computer. And then that computer is connected to the internet. And that's how we can use our browser to request those files, so we can view the contents of the webpage.

So how then are webpages structured so there are files that follow a given a given logic and a given structure. So webpages are written in something called a Hypertext Markup Language or HTML for short. You can think of HTML as a very simple programming language, you know, that gives some structure to a webpage. So it describes the structure of a web page and in turn that allows us to navigate the web page and to extract the information that we need. So how does it structure the web page? Well, the HTML based basically uses a series of elements. So these elements are what we're familiar with when we're writing documents in general, there are headings, there are paragraphs, there are images, there are tables, there are lists, there are videos there, site, sidebars and top bars, etc.

So those elements then are distinguished by tags. So if there's a table element in a web page, very helpfully, the table element is identified by the table tag, again, very simple to work with. If there's a list, and the list is ordered, you know, from one to three, and that'll be identified by the all l tag, ordered list tag, if it's an unordered list elements, so it's a list of your bullet points, then that will be identified by the

IU l time, it seems a bit, you know, abstract. So what we'll do is very quickly look at some examples of very simple web pages. So we can see how the structure works in practice.

So let's look at an example here. So on the left, we can see how a web page is actually written and structured. So at the top, we have a little tag here that tells you know, the browser that this is a HTML file as distinguished from a more general text file, or an image file or a video file, etc. Let me have, I mentioned where these tags and again, tags represent elements of a webpage. So the head tag basically represents the metadata associated with a web page. You don't usually see the metadata when you request the web page in your browser. Really, the metadata is aimed at search tools. You know, so you can search students can see what the title of the webpage is and a kind of brief description of what it is also. Already interested in as social scientists and people interested in collecting data is everything contained really in these body tags. So body is the content of the web page. It's, it's the information and it's the contents we see when we view the web page through our browser. And then this very simple web page, we have simply two elements, we have a top heading, which is identified by the h1 tag. And then we have a paragraph which is identified by the p tag. And if we run that, as I said, markup language, which is essentially code, and it produces the output over here. So this is akin to using your web browser to request a web page, and here it is the content is returned.

So again, very simple, which is good, we can look at maybe a slightly more, slightly more relevant and real an example of how a web page looks. So if we think of the ncrm website, that's a collection, again, of web pages with information on research, methods, training, guidance, advice, etc. So I've used my web browser here, I'm using Google Chrome, I've requested the web page. And here I can see all of the content that's on the homepage. So this is the first web page you encounter. When you request the ncrm. website. There's a bit more going on than the simple example, we looked at a neat trick that you may come across before, but maybe you haven't is in Google Chrome. Anyway, if you right click and view page source as the option here, you can actually look at the underlying HTML of any webpage. Now you can't make any changes, you know, I can, I can click anywhere here, I can type away, I can try and delete things, you won't make any difference to the actual web page itself. But it's really good. You can view the underlying HTML and a code. So again, we see the core elements that are part of any web page, we've got the little tag at the top saying, this is a web page type of file. Here, we got some some metadata. So again, we've got a title associated with this web page.

So that's what the search tools will see, when they request this web page. There's lots of little kind of code, and lots of little scripts that run when you request the web page. Again, this isn't content, this is metadata. So for example, here's kind of a little script that produces the little pop up that says, do you accept or reject cookies, for example. But as people interested in collecting data, we're interested in what's contained in the body cloud tag. So anything within those two body tags is the content of the webpage is what we see. And that's what we want to potentially scrape. So again, yeah, we can see different types of elements, we've got an unordered list here, as you can see, with the UL tag, and then we can see yet so we've got some header twos. So these are, you know, the smaller headings. So this one called training and events. And if we go back to the web page itself, here it is. So here's the code. The browser interprets this code as HTML. It says, Okay, this is a web page. And here's how I display the content. So we've got headings, and lists, etc.

So it's a very brief and quick introduction to HTML, and just let it wash over you just now in the next lesson, we're going to actually dig into requesting a web page, where again, we have to implement the step, which is, you know, locating the information on the webpage so that we can extract it. In order to locate it. We'll just do what I've done here, which is a manual inspection of the web page and the tags underpinning the web page, and then building our web scraper, telling it to extract the information contained in those tags. So I'll see you for the third lesson shortly.