# Introduction to Web Scraping

Good afternoon, I'm Diarmuid Mcdonnell, I'm a lecturer in Social Sciences at the University of the West of Scotland. And in this learning resource, I'm going to show you how you can collect data from the web. And this is a technique more commonly known as web scraping.

In this first video, we're going to look at the fundamentals of this social research method. And in particular, we're gonna look at the value. The key question, why should we be interested as social scientists in collecting data from the web? and I'm going to outline the logic,  so the kind of core steps that every time you're looking to scrape data from the web, you will follow these steps. And then we will implement the logic as well and look how we can achieve it ourselves.

The first thing I'll do and before going into what it is, is a technique is actually demonstrate that I know what I am talking about. So here I have a programming script that looks to collect data from the web. So currently, I'm working on a project that is interested in the impact of the pandemic on charitable organisations. So one of the countries I'm interested in is Australia, which has quite a well developed, charity sector in place. And what I'm interested in is collecting information on the numbers of charities that are formed in a given month. So these are new charities that come into being. And I'm also interested in the number of charities that no longer exists. So these are charities that maybe have completely become insolvent, do not have the financial capacity to operate in, they merged with a different organisation. And maybe they've continued to operate, but they've renounced their charitable status, and they're operating as companies or private businesses and from now on.

Let's get the first piece of information, the number of new charities registered, I can do that in a very familiar way. Now I can download a data file from the charity regulator, and I can extract the the registration field and the date field, and I can do the analysis that I need. What's difficult is to get information on charities that no longer exists. That's not contained in the data file, I can't simply download it manually by going to the website and or any other usual means of doing so. But what I can do is I can view this information on the regulator's website. So using their search tool, I can see a list of different statuses. So I can see all the charities that continue to operate as organisations, maybe the're social enterprises or businesses now, and what they no longer want to be charities. So I can use the regulator's website, and I can find out how many of these charities there are.

So here we have the list. So again, this is all publicly available information and I can get the name of the organisation, and its charity status, and a link to its webpage on the regulator's website. And it's on that web page where I get the information that I need. And as you can see, this is its cancellation date. So this is the date this organisation continued operating but no longer existed as a charity. So what I needed was a programming script that went to the regulator's website, collected that tabular information, and output it into a file that I can use in my research. I go into the detail of how this

programming script is constructed in a later video. For now, I'm just going to set it running, continue with the presentation and hopefully in five or 10 minutes time, show you that it has actually worked.

So let's return to the fundamentals of web scraping. So how can we define it? Well, the first thing to say is that it is a computational technique for capturing or storing or extracting data that's stored on a web page. Computational is the obvious the key word here, you can manually visit a web page, just like I showed you right there. We can use our mouse or we can use our keyboard and we can select the information that we want, we can right click, we can copy and paste. And you know, we can right click on an image and say save images, etc. So we can manually download or extract data from webpages. And oftentimes, it can be simpler. If you only need a little bit of information infrequently, you can simply do manual extraction. But of course, it's very obvious downsides when it comes to accuracy and the kind of labour resource needed and to collect larger and more frequently updated social science datasets.

How do we do it? Well, we generally implemented using a programming script just like what I showed you, which is written in the Python language. But there are software solutions that you can use also. So Google Chrome has a means of scraping web page. So you can use your browser to do it. Microsoft Excel, I think has some functionality that allows you to scrape web pages. And there are specific software and packages that you can use. Also, I'm generally in favour of writing programming scripts, I think you get additional flexibility and kind of specificity of what you want to do. I think there are intellectual benefits that accrue also from learning a programming language as a social scientist. And there are some other reasons why I think it's, it's worthwhile, also. But the ultimate aim, were social scientists, it's to collect social research data. And whether you  you do it using a programming scripts? Whetheryou do it manually, etc. That's, that's, of course up to you. What I would say is the programmatic approach is actually relatively simple. It may seem daunting at the outset, but actually, it's a very mature methods, web scraping. There's lots of documentation, lots of examples, and lots of help out there that will get you up and running very quickly.

So why should we do it? This is the fundamental question as social scientists, maybe it's interesting, maybe you get a kick out of it? I certainly do. But why do we do it for research purposes? The first thing to say is maybe an obvious point but it should be stated, webpages are an incredibly rich and important source of publicly available information on a whole variety of social phenomena of interest. And my particular area, as I've just said, is charitable organisations, the data files and the kind of data resources that are available are increasing in quality and in terms of volume, there's still quite a lot of information. And as we've seen, in my example, just there, that can only be accessed through web scraping. Now I can do it manually, I can take an incredible amount of time, but the project I'm doing needs monthly snapshots of the number of new charities and the number of disappeared charities and web scraping provides very accurate and reliable way of doing.

So we think of the variety of information that web pages can hold. I mean, there's a whole host of different data types, we've got files, we've got text, we've got photos, videos, we've got lists, like I've just showed you, you know, tables of information that we might be interested in. Web scraping can be used to collect all of those different data types. So it's not just for cutting and pasting, you know, in an automatic way from a web page, we can use it to request files and download them to our machines. It's

a very flexible method in that sense. Once collected the data that we think is of interest and is really rich and varied, we can kind of reshape it into the format's we need for analysis. Typically, you know, i'm a quantitative researcher, so I use stata. And as we've seen with the example I've just been showing you, it's going to spit out a nice CSV file that I can import nice and easy into stata for analysis.

So what's the logic? So how do we actually go about implementing this research method? Well, basically, there's two kind of stages, there's, there's things we need to know and there's things we need to do. So the first thing we need to know is the location where the web page can be accessed. This is maybe more commonly known as the URL or the web address or the link. Different interchangeable terms can be used. But all of those things represent the location on the web where that web page is actually stored. So for example, the BBC homepage can be found at the following URL, HTTPS, colon forward slash forward slash, BBC co.uk. And that's the location of the BBC website, on the World Wide Web. So once we know the location of the actual web page, we need, then we need to find the location of the information on that web page. Now, webpages, you'll be very glad to hear follow a very standardised and interpretable structure. So it's not just you know, a wash of text and programming, you know, cold web pages are hierarchical, they use tags to identify different types of information on the web page. And we can use that to our advantage to locate the information we need, and then to actually extract it.

So once we know those two critical pieces of information, and we can move on to the fun bit, the doing the actual scraping of data. So the first thing we do is we need to actually request that web page. We do that manually by opening a browser from a Google Chrome, you type in a the address or the link or the URL, and you press enter or you press the button, and your browser requests the web page and returns the contents to view so you can view it. So we can also mimic that process so we can conduct that process using a programming language. Once we've requested the web page, we then need to you know parse so we need to understand the structure of the web page. So then we can start picking up the bits of information that we need. In essence, this is just basically telling, in my case, Python that what we've requested is a web page that's structured like a web page and take that into account as we move forward. So once we understand the structure of the webpage, then we need to start picking out the bits of information that we need, or that we're interested in and then we can write that information to a file, so we can use it for analysis later. So six kind of key bits, or six elements of logic to work scraping, two relate to knowledge that we need, and four relate to practices that we need to engage in.

So what would be the value of all of that kind of process. So really, as I said before, it's a very mature method got lots of established packages,  what I've been using, what we will use going forward in this learning resource is the requests module, and the beautifulsoup module in the Python programming language. There's lots of examples, lots of help available, and you're not starting from scratch, there's plenty of available help. If we use computational rather than a manual approach, and we get the ability to schedule or automate our data collection activities. As I said, previously, what I do is I set this programming script running once a month, on the 28th of every month, I have a server, so I have my own computer in the cloud, it runs on Matt, all the data gets downloaded there. And then once a month, I download to my machine, I imported this data, and I produce the analysis that I need. And again, the richness of the information and its variety as well that you can find on webpages, I think it's a point worth repeating again, and again, there's an enormous amount of digital information relating to social

phenomena, that if only we could you know, access in a very reliable and accurate way,  we can do so much more with as social scientists. So we can be a very accurate and reliable data collection method, and a very useful part of your toolkit as a social science researcher.

So I've been evangelising about this method so far, but of course, it has a couple of limitations that needs to be borne in mind. The first is the information that you request, it tends to be updated quite frequently. Now there are some web pages that you know, never really experienced updates or changes. Again, go back to maybe the BBC website that changes every couple of minutes, nevermind every hour or every day. So the information on that and changes very frequently. If the information changes, then maybe the structure of the webpage changes, which means your programming script can't find the the tag or the element on the webpage it's looking for, then your script breaks. Now there's ways of improving your script so that it doesn't break when it encounters changes. But it does mean that there's quite a bit of manual checking on your part of the web page you're scraping every so often to check that it looks the way you expect it to look. And it should be bear in mind, if you share your code as well that you know, you have a responsibility to maintain it so that it works with with future versions of web pages. Some websites are advanced enough that they can block or kind of slow down or throttle the scraping that you're conducting. And, for example, the Amazon website, it's not possible to scrape you know price and product information. And it's clever enough it knows to block automatic attempts to download information from the amazon.co.uk. website. That's the big example. But there are plenty, it's not a very, I say advanced but it's not actually a very uncommon technique by websites, you may be surprised which ones can actually stop you or slow you down in your web scraping activities.

Web scraping and of course, any kind of computation work that you do is dependent on your computing setup. Where it's working is very possible to do on standard desktops or laptops. You don't need lots of memory for example, it's not too difficult to set up Python on your laptop. That's something we can, we can demonstrate. But it still should be borne in mind. So Python itself gets updated every so often. The packages that you use to scrape websites, that gets updated every so often and this all needs to be taken into consideration and updates need to be made to your laptop or desktop or whatever that you're using.

And of course there are some ethical and legal implications of web scraping. We'll cover those in the final video. You don't need to be too scared the legal kind of ramifications, but you do need to bear in mind that it's not a free for all that just because the information can be viewed publicly does not mean you have a right to automatically or computationally scrape that data or to use it for whatever purpose you deem necessary.

So why have I been collecting that Australian data? returning to my example that I showed you earlier/ Well, this is what I want to get at. So this is the analytical output I'm interested in. So I've called these charity removals. So that's an organisation that for whatever reason, no longer operates as a charity. I'm interested in these seven jurisdictions globally, the three UK jurisdictions and the USA and Canada and New Zealand and Australia. So the reason I want to collect data on charities in Australia that no longer exists, because I want to calculate well, is 2020 significantly different than previous years when it comes to disappearing or removed charities. So you can see in Scotland, 2020 was an untypical year,

it's, you know, nearly two standard deviations below the average. So there's an average amount of deregistrations that we would expect and 2020 was was considerably less than that. It's not quite the case for Australia, and in fact, yeah, there are fewer charities registered, or sorry, deregistered in 2020 from what we would expect, but it's not so low, you know, it's not even one standard deviation below what we would expect. But that's the use to which I am putting the data that I've been colleittle output window. So you know, I'm asking the programming script to print some relevant information to the screen. It's telling me it's, you know, it's requesting different URLs, different page numbers, we'll get into what all that means in future videos.

But here's the output. So here's today's date. And today's time, just to show you that, you know, the programming script has created this file, it's not something I've done in the background. We can open it up. Let's see if it's actually scraped the information that we need. Follow up? Yes. How's work? Excellent. So the table we were looking at previously, this table here, I have now basically scraped it and converted it into a data file that I can know important to stata and use for my analysis. So here we go. Yeah, the Astronomical Society of Tasmania, which I think is on a different list. And that charity, yep has its charity status voluntarily revoked, it's no longer entitled to be a charity. What I do with the broader programming scripts, because that was just a snippet is I go to every charities webpage, and I scraped what date it was registered and what age it was deregistered as well, and that's how we conduct a web scraping piece of data collection for social science purposes.

In the next video, we're going to get into a more in depth practical demonstration and it's something you can use as well, on your own machine. There's no downloading software on your machine, everything we can do through the cloud. So I'd encourage you to join in with me as well. Excellent. So see you in the next