Q. Any thoughts on limitations and complexities of wayback machine capture?

A. This is not a site that I have used before, but from what I can see (using www.bbc.co.uk as an example), a requested URL might look like http://web.archive.org/web/20191101004429/https://www.bbc.co.uk/

The number in the middle is a datetime stamp (YYYYMMDDHHMMSS). You can truncate this (say to YYYYMMDD) and you will be served the nearest archive file for the site based on the information you provide. So you can use the iteration techniques demonstrated in the Webinar to collect all of the files for a given date/time period for the given site.

Q. You haven't mentioned Mac yet - only windows and Linux. Are these techniques transferable?

A. Yes. All of the software used is available on a Mac, including Excel. All of the Browsers and Python will behave in the same way as they do on a Windows machine. A Mac specific spreadsheet program, may render the copy and paste of a table differently, but as we saw, there are already many different renderings available just on a Windows machine.

Q. For what specific data from the Internet one would have to obtain ethics approval before collecting them? Thank you

A. Pretty much the same as for any other data. If it is personal data then there may be ethics requirements there is also the possibility of copyright infringement. But it can be a grey area; one argument would be that if the data appears on a publicly available web page, then it is in the public domain.

Q. Will the loop stop if it can't find a file? If so how can you force it to continue?

A. If a file is not found, then you will typically get a '404 file not found' message returned. After each call you can check the 'status_code' to ensure that it is '200' – meaning the file was found and returned. The checking python code could be a simple 'if' statement or a 'try'…'except' statement both of which would allow you to decide in code what you want to do next.

Q. I have been looking for a more efficient way of downloading data from Stats Wales. For example, this particular table https://statswales.gov.wales/Catalogue/Health-and-Social-Care/Social-Services/Childrens-Services/children-receiving-care-and-support/parentalfactorsofchildrenreceivingcareandsupport-by-measure-year.

Has drop downs for each local authority (x22 in Wales) as well as for child status (All, LAC, CP and Other). As I'm new to this, I have been downloading the individual tables but really I want all the permutations (92 if you include Wales). There is a link to Open Data but I don't understand this ...

A. The link to Open Data is a bit of a red-herring as it is only referring to the Metadata components, so not the actual data you want. I noticed that the https://statswales.gov.wales web site does have some tables available by local authority – obviously I don't know if this data would meet your requirements, but you would only have the tables for the 3 years to download. I found that simple copy/paste (keep source formatting)worked quite well.

Q. Are there any security concerns to be aware of when web scraping? E.g. Does it leave a footprint? Can it expose IP addresses?

A. Every time you access a website you leave a footprint. Typically this will include the IP address of your machine (although this may not be the real address of your machine if you are attached to a private network in an institution or even at home going through your ISP) , the operating system in use and normally what type of browser you were using. What is used for this when we use Python code, I am not sure. But the fact that it is not a browser can be detected and some sites may block you access for this reason. In general, the privacy issues are really any different than for normal browsing.

Q. How does Excel deal with missing or changed column names when combining files from a folder?

A. Excel uses the column headings from the first file in the folder. If you have files where the column heading may be different or additions and deletion of column headings, you can do the following:

Create a dummy file in the folder and name it so that it appears at the top (e.g. by starting the name with '!'). In the file place all of the possible column heading from all of the other files in the top row and add a second row below of all '0' (integer not string). When Excel combines them it will correctly align all of the column data and leave blanks or Nulls for columns with no data in a given file. You then delete the added line (which will of course will appear as the first data line. You could write a small piece of python code to create the dummy file. But if you are not a programmer, with a bit more trouble you could use Excel. In the example I gave I used csv files, but this will also work with Excel files as well.