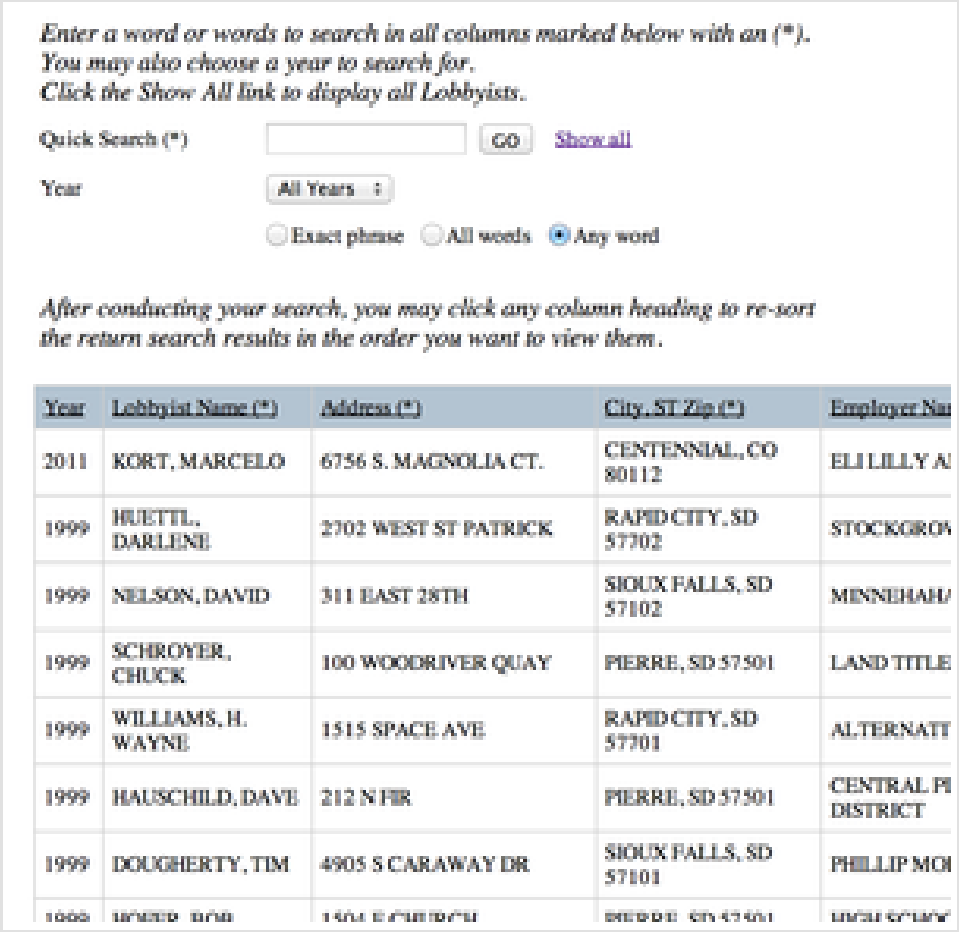
ire.org

How to use Helium Scraper to grab data online

by Liz Lucas, Ire & Nicar | 08.21.2014 • Aug. 21, 2014 • 4 min read • original

By Liz Lucas, IRE & NICAR | 08.21.2014



Enter a word or words to search in all columns marked below with an (*).
You may also choose a year to search for.
Click the Show All link to display all Lobbyists.

Quick Search (*) GO Show all

Year All Years

Exact phrase All words Any word

After conducting your search, you may click any column heading to re-sort the return search results in the order you want to view them.

Year	Lobbyist Name (*)	Address (*)	City, ST Zip (*)	Employer Na
2011	KORT, MARCELO	6756 S. MAGNOLIA CT.	CENTENNIAL, CO 80112	ELI LILLY A
1999	HUETTL, DARLENE	2702 WEST ST PATRICK	RAPID CITY, SD 57702	STOCKGRO
1999	NELSON, DAVID	311 EAST 28TH	SIOUX FALLS, SD 57102	MINNEHAH
1999	SCHROYER, CHUCK	100 WOODRIVER QUAY	PIERRE, SD 57501	LAND TITLE
1999	WILLIAMS, H. WAYNE	1515 SPACE AVE	RAPID CITY, SD 57701	ALTERNATI
1999	HAUSCHILD, DAVE	212 N FIR	PIERRE, SD 57501	CENTRAL PI DISTRICT
1999	DOUGHERTY, TIM	4905 S CARAWAY DR	SIOUX FALLS, SD 57101	PHILLIP MO

South Dakota lobbyist database

We've all visited websites that contain valuable information we want to analyze, such as inspection reports, lobbying expenditures, or a list of jail inmates. We're glad those agencies have made this information available, but as data journalists it doesn't do us much good locked in HTML: We can't sort, filter, sum or join it to other data in this format.

What we want to see is the DOWNLOAD button, and while that's catching on (at both local and federal levels), most of the time downloading is not an option.

So how do we get it?

Imagine harvesting data from this site: https://apps.sd.gov/st12odrs/LobbyistViewlist.ascmd=resetall. The South Dakota Secretary of State allows anyone to search through a list of all lobbyists who are registered with the state. Names, addresses and employer information are included. Click "Show All" next to the search bar and you'll see a table of registered lobbyists appear.

Scroll to the bottom of the table and you'll notice that there are 406 pages of names. If you wanted to end up with a spreadsheet of all lobbyists, how would you do it?

If you're thinking, "I'd write a Python script to scrape each page," then move along! Nothing more to read here, go forth and write scripts. If you're thinking "I'd copy and paste each page into a spreadsheet," then I've got some good news. If you find yourself

contemplating five hours of doing the same rote task over and over again, it's time to think about whether a computer can do that work for you. That's where web scrapers come in.

A web scraper is simply a piece of software, a bit of code, or a browser plugin that does that work for us, preferably while we go have a beer. Helium Scraper is a $99 piece of software (for Windows only) that will make quick work of the lobbyist data, and will not require you to write any code. Helium offers a 10-day free trial.

A browser plugin, such as Chrome Scraper, does a pretty good job of grabbing data that is in a table-like format and exists on a single webpage. Scraping something with Chrome Scraper will take much less time than almost anything (save for just copying and pasting), but it has limitations. If you're tackling something like the lobbyist page where you need the scraper to click through pages, then Helium Scraper is a much better choice.

However, Helium Scraper is a piece of software that is a lot like a black box – we can't see exactly what's happening on the inside. There are some quirks you'll need to get used to. For example, when selecting items on a webpage to create a **kind**, Helium Scraper acts very differently if you click directly on text in a cell than if you click in the whitespace *next to* the text in the cell. You'll just have to get used to these little flukes if you use the software.

Writing a script in something like Python ultimately gives you the most control over your web scraping process, which is partly why a lot of data journalists prefer it. It's much easier to make small tweaks to an existing scraper in Python than it is in Helium Scraper. On the other hand, learning Python requires a significant investment of time that we can't all afford.

How Helium Scraper works

Helium helps you build a web scraper using a point-and-click interface that operates with two general concepts: kinds and actions. Kinds are the pieces of

the website you'll need the scraper to act on. Think of them as building blocks. Actions are what we create to arrange those building blocks in a particular order.

The result of the scrape is the data, which Helium Scraper stores in a database that you can query right inside the program. You can also export the data as a Microsoft Access database or a comma-separated values text file.

To scrape the lobbyist data, we need the scraper to click "Show All." First we have to tell the scraper how to recognize the "show all" link by creating a kind, and we tell it to actually click the button by creating an action.

Additionally, we want to create a **kind** for each column of data we want: year, name, address, etc. That way the scraper can recognize the pieces of data we want. The **action** we'll create is to take those **kinds** and extract them to a table.

These are the basic steps, but Helium Scraper has a lot of additional functionality. For example, you can make the program wait between each click, which is helpful

because certain web servers won't allow you to make too many requests too quickly. You can also send the results of your scrape to different tables if you want to create a relational database.

You can download a full tutorial for Helium Scraper here. It walks through all the steps to scrape the SD registered lobbyists.

Practical tips

If you're just starting out with Helium Scraper, start with a simple scrape. Even if you have a big scraping job that you need to do, first use Helium Scraper to tackle a small piece of it. This will help you get used to how the software works before you try to dive into something complicated.

Save multiple versions of your scraper as you go. You can save an entire Helium Scraper project as an .hsp file. If you save copies periodically, it will be easier to take a step backward should something go wrong.

Run the scraper over individual pieces as you build them. If you set everything up and then it doesn't work as planned, it can be hard to go back and figure out what went wrong. Testing pieces of the scraper and making sure they work before you move on to a new thing will help the whole process go more smoothly.

Liz Lucas is director of the IRE and NICAR database library. Contact her at liz@ire.org.

Original URL:
http://ire.org/blog/uplink/2014/08/21/free-data-helium-scraper/