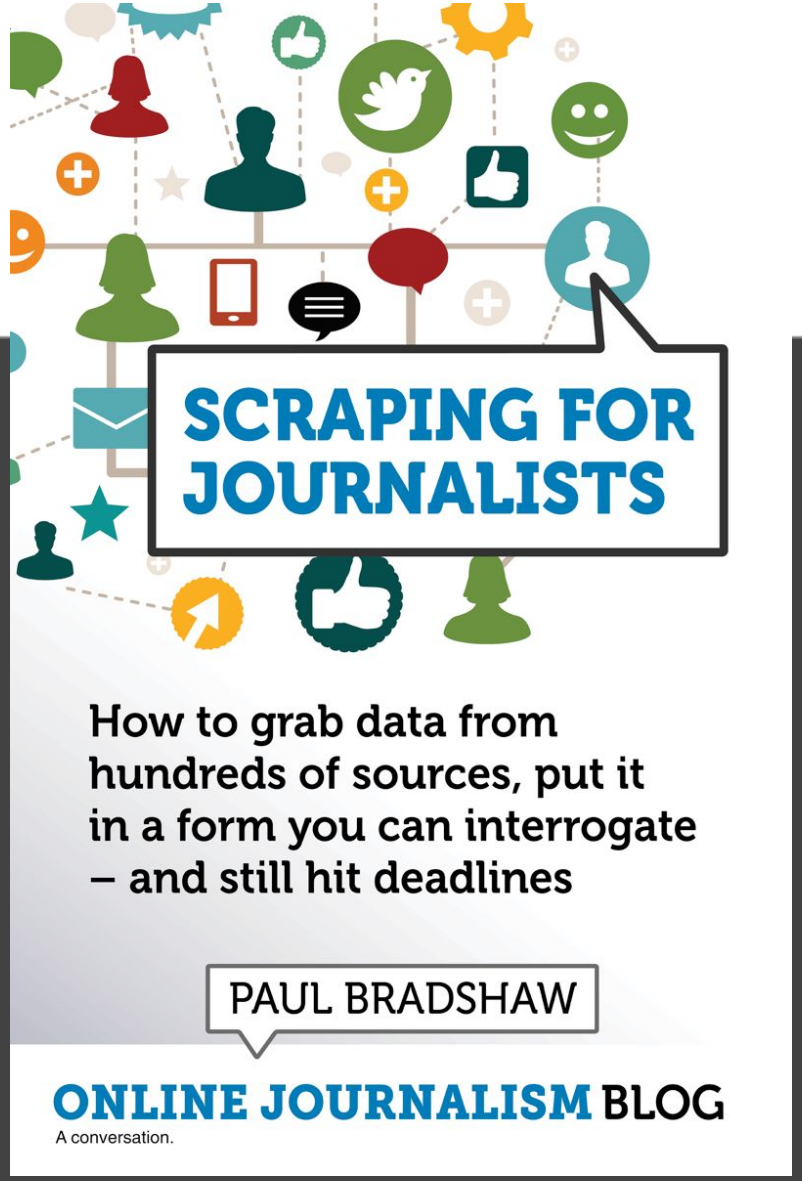
Coding in Google Colab: lists



Paul Bradshaw

Leanpub.com/scrapingforjournalists

What we'll cover

- What are **libraries** in Python - and why you need to know
- How to **import** libraries in a Python notebook in Google Colab

Libraries

- A library is a **collection of recipes (functions)** and other stuff that someone has created for a particular type of problem
- Make it possible to ‘stand on the shoulders of giants’ & use code created by others
- E.g. the **scraperwiki** library is a collection of tools for solving scraping problems
- And **lxml.html** is a library for converting to XML
- **Pandas** is a library for data analysis
- **Matplotlib** is a library for visualisation

```
[ ] #install the libraries
    #scraperwiki is a library for scraping webpages
    !pip install scraperwiki
    import scraperwiki
    #lxml.html is used to convert it into xml (more structured)
    import lxml.html
    #cssselect is used to drill down into that and find data in tags
    !pip install cssselect
    import cssselect
    #the pandas library which is used to work with data - we call it 'pd'
    import pandas as pd
```

Libraries… in Colab

- (Some) libraries need **installing** first
- (All) libraries need **importing**

(How do you know?)

Trial and error...

```
import scraperwiki
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
<ipython-input-2-71791e80ea22> in <module>()
----> 1 import scraperwiki

ModuleNotFoundError: No module named 'scraperwiki'

---------------------------------------------------------------------------
NOTE: If your import is failing due to a missing package, you can
manually install dependencies using either !pip or !apt.

To view examples of installing some common dependencies, click the
"Open Examples" button below.
---------------------------------------------------------------------------
```

OPEN EXAMPLES SEARCH STACK OVERFLOW

```
!pip install scraperwiki
import scraperwiki
import lxml.html
!pip install cssselect
import cssselect
```

pandas as pd?

- A library can be **renamed** at the same time as it is imported (typically with shorter names for convenience)
- ...because when you use a function from a library you need to name the library

```
import pandas as pd
```

Using a library

- When you use a **function** from a library you name the library and the function, with a period joining them:
- **scraperwiki.**scrape(fullurl)
- **lxml.html.**fromstring(html)
- **pandas.**DataFrame(columns=["title"])

  ...or if renamed when imported:
  pd. DataFrame(columns=["title"])

Hold on — functions?

Functions = recipes

- A **function** is a name for a recipe. Used in Excel, e.g. SUM, AVERAGE, VLOOKUP
- A function is always followed by parentheses to ‘pass’ any ingredients, e.g. =SUM(A1:A10)
- scraperwiki.**scrape(fullurl)**
- lxml.html.**fromstring(html)**
- pd.**DataFrame(columns=["title"])**

Recap

- A library is (pre-)installed, and imported:

```
!pip install scraperwiki
import scraperwiki
```

- Functions (recipes) from that library are joined by a period and followed by parentheses:

```
html = scraperwiki.scrape("http://blah.com")
```

Try it now:

- Create a notebook and import the libraries we will need:
  - scraperwiki
  - lxml.html
  - cssselect
  - pandas
- Use the scrape() function from scraperwiki to scrape a webpage, then print it
  - html = scraperwiki.scrape(“INSERT URL”)
  - print(html)

Next:

- We have scraped the page into ‘html’
- But we need to drill down into that to grab specific items of information… with **cssselect**