Web Scraping with BeautifulSoup

Bety E. Rodriguez-Milla

Web Scraping

Waterloo Data Science and Data Engineering MeetUp February 21, 2019

Motivation

A not-for-profit organization is trying to reach the 194 Community Foundations of Canada (CFC) across the nation.

They want to mail them some materials. They need a *spreadsheet* with the name of the contact, title, mailing address, etc. (11 fields), for each of the CFCs.

Copy-paste each field?   Web Scraping!

What is Web Scraping?

It is the practice of gathering data, through any means other than API.

For example, by writing an automated program that:

- queries a web server,
- requests and retrieves data,
- parses that data to extract information, and
- stores it.

Web Scraping with Python by Ryan Mitchell (O'Reilly)

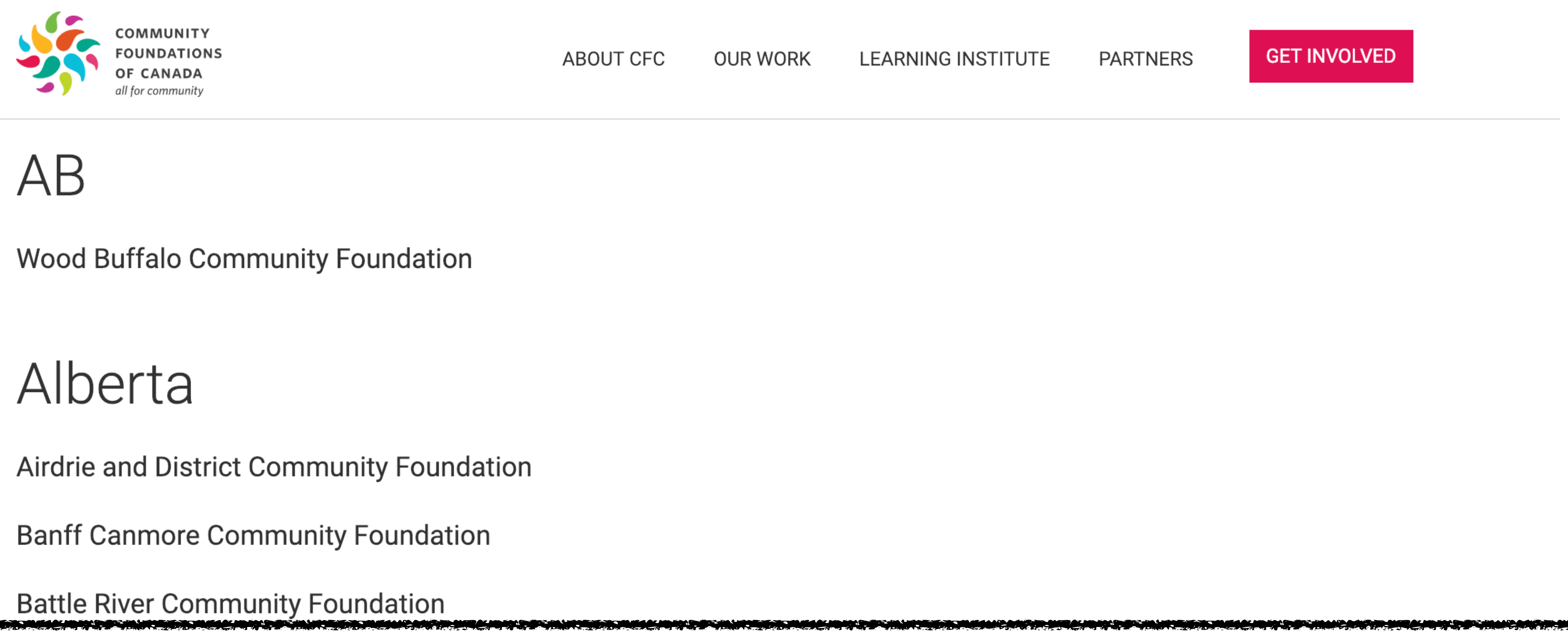
Why Web Scraping?

- Web scrapers are excellent at gathering and processing large amounts of data, thousands of pages at once, from a collection of sites.
- Not all websites have an API - or an API that suits your needs.

If you can *see* it in your browser, you can *access* it via a Python script.

And, if you can *access* it, you can *store* it in a database to *retrieve* and *analyze*.

Inspect the Page



AB

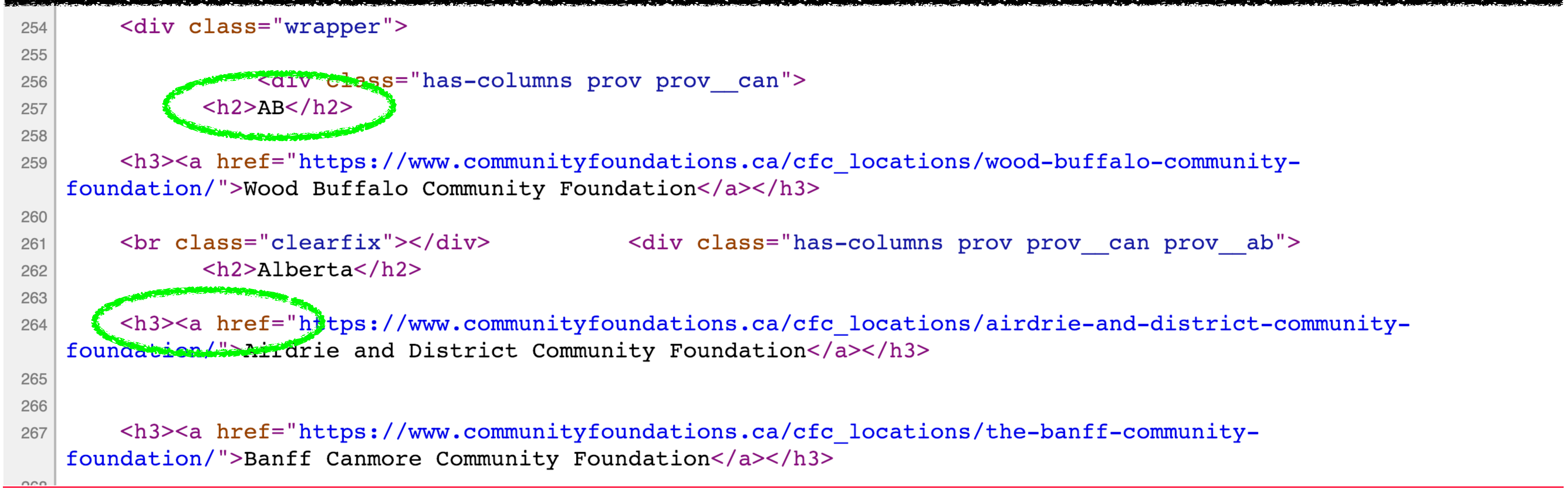
Wood Buffalo Community Foundation

Alberta

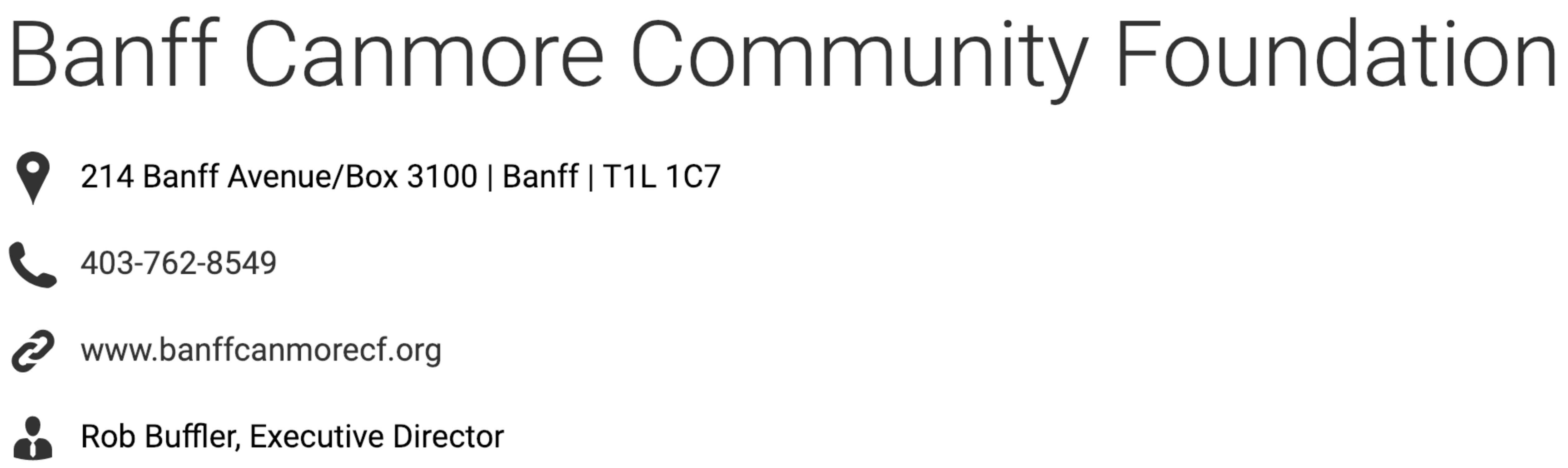
Airdrie and District Community Foundation

Banff Canmore Community Foundation

Battle River Community Foundation



```
    <div class="wrapper">

          <div class="has-columns prov prov__can">
        <h2>AB</h2>

    <h3><a href="https://www.communityfoundations.ca/cfc_locations/wood-buffalo-community-foundation/">Wood Buffalo Community Foundation</a></h3>

    <br class="clearfix"></div>            <div class="has-columns prov prov__can prov__ab">
        <h2>Alberta</h2>

    <h3><a href="https://www.communityfoundations.ca/cfc_locations/airdrie-and-district-community-foundation/">Airdrie and District Community Foundation</a></h3>


    <h3><a href="https://www.communityfoundations.ca/cfc_locations/the-banff-community-foundation/">Banff Canmore Community Foundation</a></h3>
```



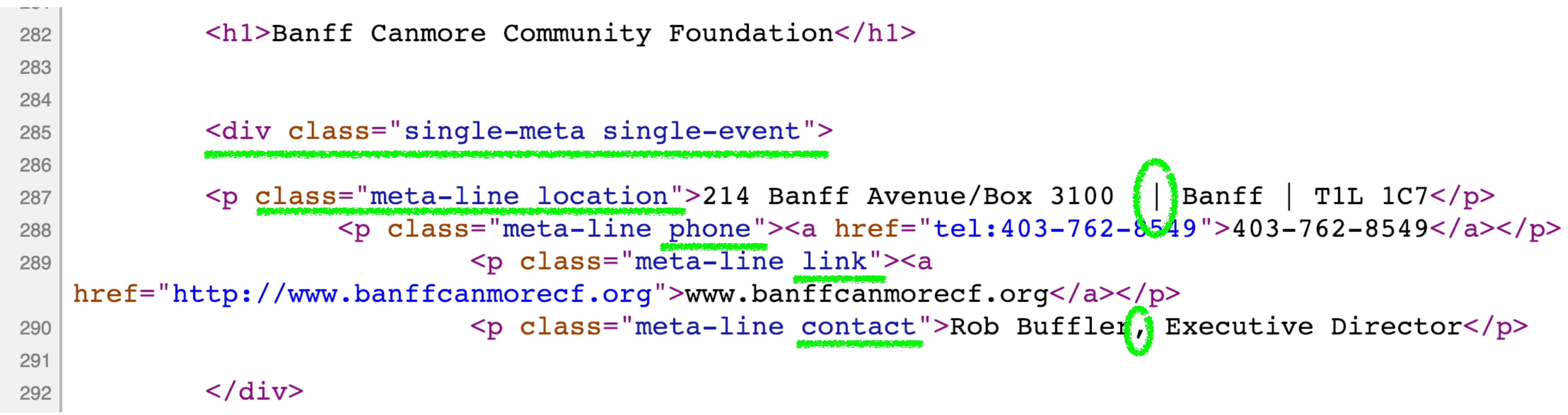
Banff Canmore Community Foundation

214 Banff Avenue/Box 3100 | Banff | T1L 1C7

403-762-8549

www.banffcanmorecf.org

Rob Buffler, Executive Director



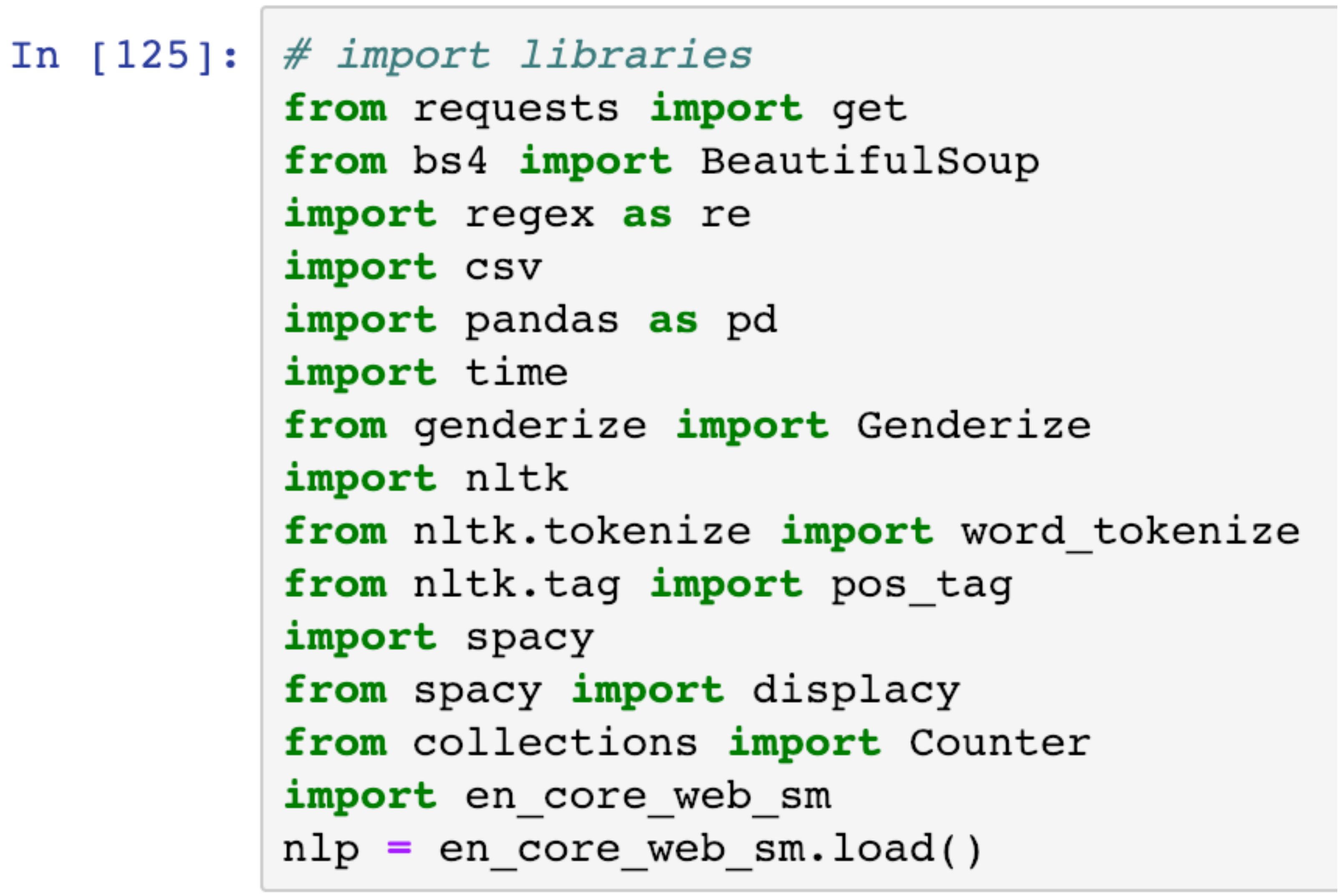
```
        <h1>Banff Canmore Community Foundation</h1>


        <div class="single-meta single-event">

        <p class="meta-line location">214 Banff Avenue/Box 3100 | Banff | T1L 1C7</p>
                <p class="meta-line phone"><a href="tel:403-762-8549">403-762-8549</a></p>
                        <p class="meta-line link"><a
href="http://www.banffcanmorecf.org">www.banffcanmorecf.org</a></p>
                        <p class="meta-line contact">Rob Buffler, Executive Director</p>

        </div>
```

BeautifulSoup

- Python library for parsing HTML and XML documents - even for pages with malformed markup or poorly designed.
- Provides simple methods to navigate, search, and modify parse trees.
- Automatically converts incoming documents to Unicode and outgoing to UTF-8.

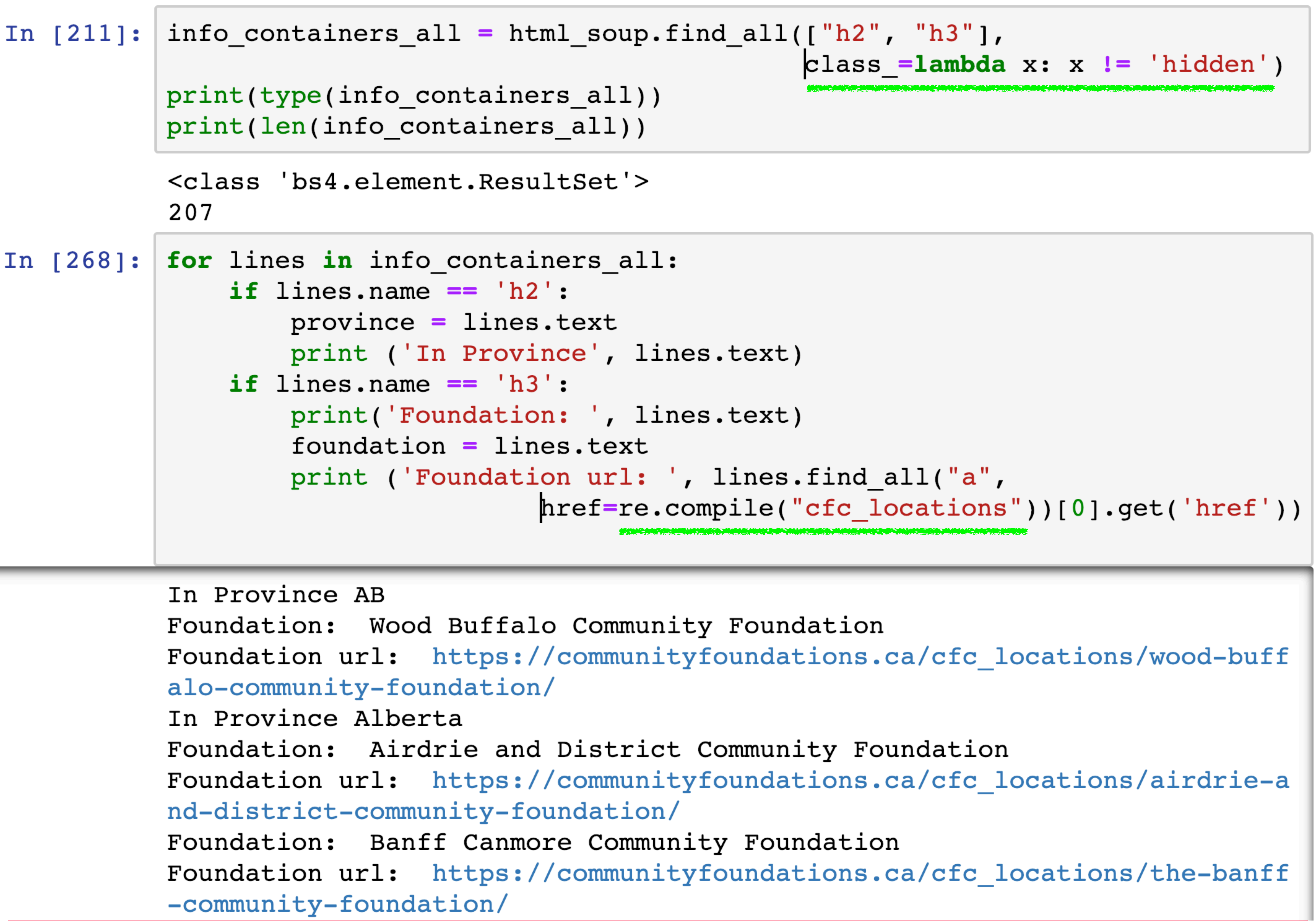
www.crummy.com/software/BeautifulSoup

Libraries Used

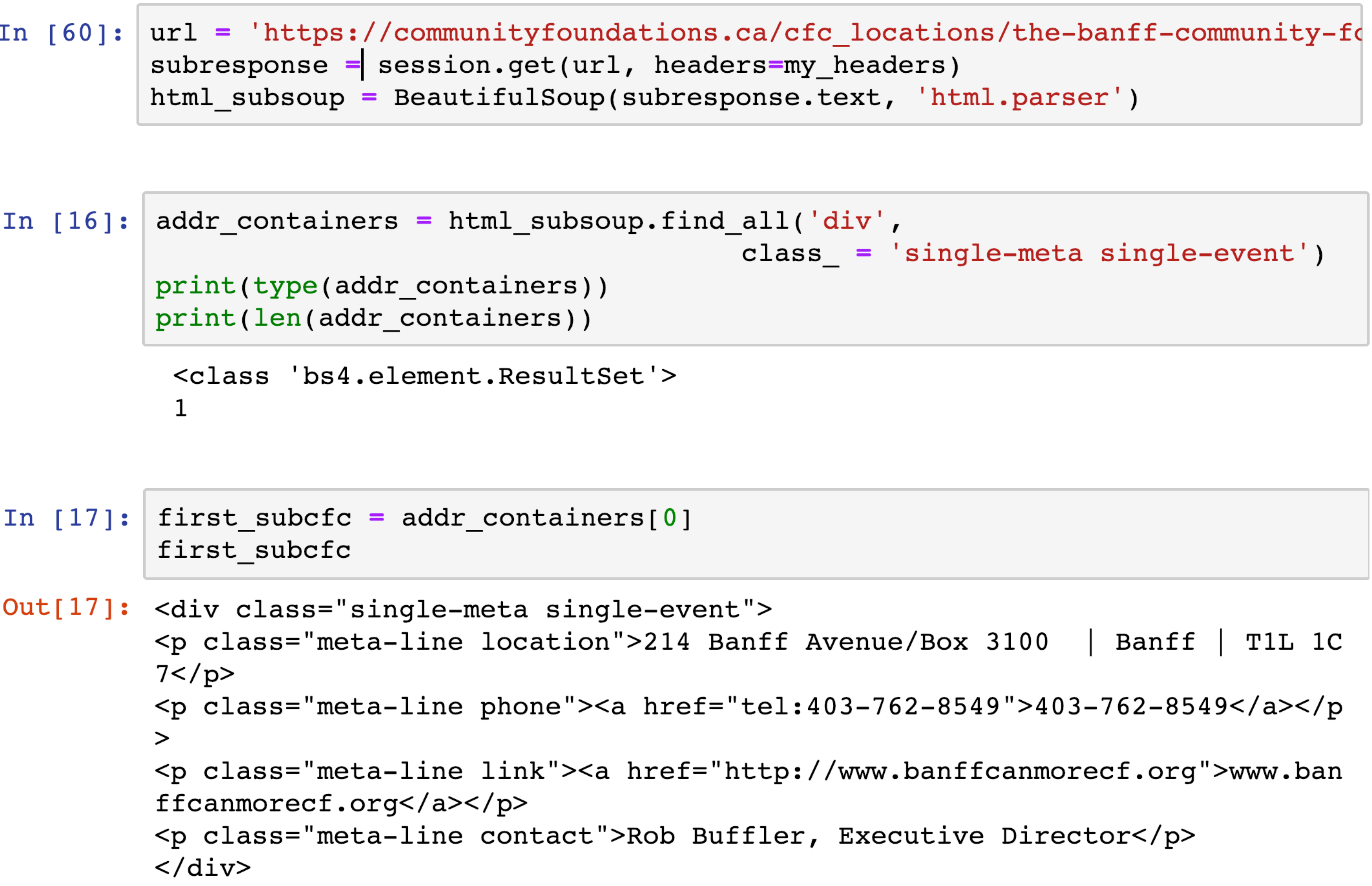


```
In [125]: # import libraries
          from requests import get
          from bs4 import BeautifulSoup
          import regex as re
          import csv
          import pandas as pd
          import time
          from genderize import Genderize
          import nltk
          from nltk.tokenize import word_tokenize
          from nltk.tag import pos_tag
          import spacy
          from spacy import displacy
          from collections import Counter
          import en_core_web_sm
          nlp = en_core_web_sm.load()
```

```
In [49]: my_headers={"User-Agent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_3)\
                        AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98\
                        Safari/537.36",
                        "Accept":"text/html,application/xhtml+xml,application/xml;\
                        q=0.9,image/webp,image/apng,*/*;q=0.8"}

In [57]: url = 'https://communityfoundations.ca/find-a-community-foundation/'
         response = session.get(url, headers=my_headers)

 In [4]: html_soup = BeautifulSoup(response.text, 'html.parser')
         type(html_soup)

 Out[4]: bs4.BeautifulSoup

In [59]: info_containers = html_soup.find_all('h3')
         print(type(info_containers))
         print(len(info_containers))

          <class 'bs4.element.ResultSet'>
          194

In [64]: first_cfc = info_containers[0]
         first_cfc

Out[64]: <h3><a href="https://communityfoundations.ca/cfc_locations/wood-buffalo-c
         ommunity-foundation/">Wood Buffalo Community Foundation</a></h3>

In [123]: location_title = first_cfc.text
          location_title

Out[123]: 'Wood Buffalo Community Foundation'
```



```
In [211]: info_containers_all = html_soup.find_all(["h2", "h3"],
                                                   class_=lambda x: x != 'hidden')
          print(type(info_containers_all))
          print(len(info_containers_all))

          <class 'bs4.element.ResultSet'>
          207

In [268]: for lines in info_containers_all:
              if lines.name == 'h2':
                  province = lines.text
                  print ('In Province', lines.text)
              if lines.name == 'h3':
                  print('Foundation: ', lines.text)
                  foundation = lines.text
                  print ('Foundation url: ', lines.find_all("a",
                                    href=re.compile("cfc_locations"))[0].get('href'))

          In Province AB
          Foundation:  Wood Buffalo Community Foundation
          Foundation url:  https://communityfoundations.ca/cfc_locations/wood-buff
          alo-community-foundation/
          In Province Alberta
          Foundation:  Airdrie and District Community Foundation
          Foundation url:  https://communityfoundations.ca/cfc_locations/airdrie-a
          nd-district-community-foundation/
          Foundation:  Banff Canmore Community Foundation
          Foundation url:  https://communityfoundations.ca/cfc_locations/the-banff
          -community-foundation/
```



```
In [60]: url = 'https://communityfoundations.ca/cfc_locations/the-banff-community-f
         subresponse = session.get(url, headers=my_headers)
         html_subsoup = BeautifulSoup(subresponse.text, 'html.parser')
```

```
In [16]: addr_containers = html_subsoup.find_all('div',
                                                  class_ = 'single-meta single-event')
         print(type(addr_containers))
         print(len(addr_containers))

         <class 'bs4.element.ResultSet'>
         1
```

```
In [17]: first_subcfc = addr_containers[0]
         first_subcfc

Out[17]: <div class="single-meta single-event">
         <p class="meta-line location">214 Banff Avenue/Box 3100  | Banff | T1L 1C
         7</p>
         <p class="meta-line phone"><a href="tel:403-762-8549">403-762-8549</a></p
         >
         <p class="meta-line link"><a href="http://www.banffcanmorecf.org">www.ban
         ffcanmorecf.org</a></p>
         <p class="meta-line contact">Rob Buffler, Executive Director</p>
         </div>
```

```
In [37]: c_location = html_subsoup.find_all('p', class_ = 'meta-line location')
         print(type(c_location))
         print(len(c_location))
         ctext_location = c_location[0]
         ctext_location.text

          <class 'bs4.element.ResultSet'>
          1

Out[37]: '214 Banff Avenue/Box 3100  | Banff | T1L 1C7'

In [128]: address_array = c_location[0].text
          address_array

Out[128]: '214 Banff Avenue/Box 3100  | Banff | T1L 1C7'
```

How about using NLP for parsing the address?

NLTK

```
In [131]: def preprocess(sent):
              sent = nltk.word_tokenize(sent)
              sent = nltk.pos_tag(sent)
              return sent
```

```
In [132]: sent = preprocess(address_array)
          sent
```

```
Out[132]: [('214', 'CD'),
           ('Banff', 'NNP'),
           ('Avenue/Box', 'NNP'),
           ('3100', 'CD'),
           ('|', 'NNP'),
           ('Banff', 'NNP'),
           ('|', 'NNP'),
           ('T1L', 'NNP'),
           ('1C7', 'CD')]
```

Proper Nouns

Tokenization split zip code

Train model for geographical data?

spaCy

```
In [135]: article = nlp(address_array)
          len(article.ents)
```

```
Out[135]: 5
```

```
In [141]: sentences = [x for x in article.sents]
          print(sentences[0])
```

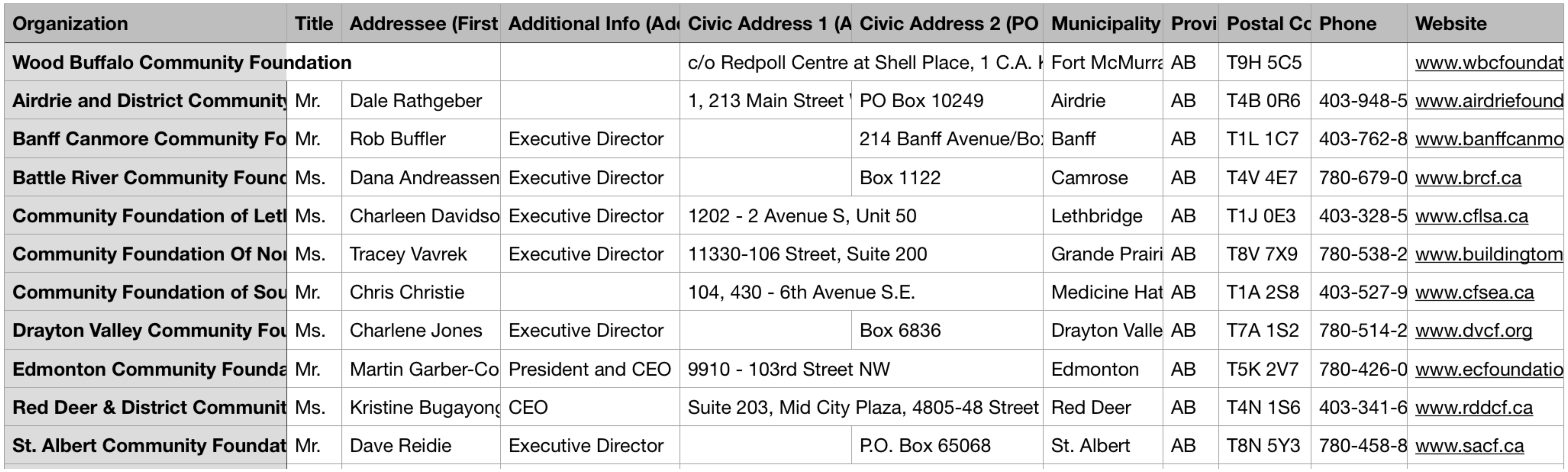
```
214 Banff Avenue/Box 3100  | Banff | T1L 1C7
```

```
In [142]: displacy.render(nlp(str(sentences[0])), jupyter=True, style='ent')
```

```
214 CARDINAL  Banff Avenue/Box FAC  3100 PERCENT  |  Banff PERSON  | T1L  1C7
CARDINAL
```

The Code

cfcMailingAddresses



Organization	Title	Addressee (First	Additional Info (Ad	Civic Address 1 (A	Civic Address 2 (PO	Municipality	Provi	Postal Cc	Phone	Website
Wood Buffalo Community Foundation				c/o Redpoll Centre at Shell Place, 1 C.A. I		Fort McMurra	AB	T9H 5C5		www.wbcfoundat
Airdrie and District Community	Mr.	Dale Rathgeber		1, 213 Main Street	PO Box 10249	Airdrie	AB	T4B 0R6	403-948-5	www.airdriefound
Banff Canmore Community Fo	Mr.	Rob Buffler	Executive Director		214 Banff Avenue/Bo	Banff	AB	T1L 1C7	403-762-8	www.banffcanmo
Battle River Community Founc	Ms.	Dana Andreassen	Executive Director		Box 1122	Camrose	AB	T4V 4E7	780-679-0	www.brcf.ca
Community Foundation of Letl	Ms.	Charleen Davidso	Executive Director	1202 - 2 Avenue S, Unit 50		Lethbridge	AB	T1J 0E3	403-328-5	www.cflsa.ca
Community Foundation Of Noi	Ms.	Tracey Vavrek	Executive Director	11330-106 Street, Suite 200		Grande Prairi	AB	T8V 7X9	780-538-2	www.buildingtom
Community Foundation of Sou	Mr.	Chris Christie		104, 430 - 6th Avenue S.E.		Medicine Hat	AB	T1A 2S8	403-527-9	www.cfsea.ca
Drayton Valley Community Fou	Ms.	Charlene Jones	Executive Director		Box 6836	Drayton Valle	AB	T7A 1S2	780-514-2	www.dvcf.org
Edmonton Community Founda	Mr.	Martin Garber-Co	President and CEO	9910 - 103rd Street NW		Edmonton	AB	T5K 2V7	780-426-0	www.ecfoundatio
Red Deer & District Communit	Ms.	Kristine Bugayong	CEO	Suite 203, Mid City Plaza, 4805-48 Street		Red Deer	AB	T4N 1S6	403-341-6	www.rddcf.ca
St. Albert Community Foundat	Mr.	Dave Reidie	Executive Director		P.O. Box 65068	St. Albert	AB	T8N 5Y3	780-458-8	www.sacf.ca

Summary

- Use web scraping to gather and process information.
- Inspect the webpage, view the source.
- BeautifulSoup can help parse HTML and XML.
  - find( ), findAll( ), tag names and attributes, works with regEx, search by CSS class.
- Fortunately, CFC page was straightforward to process.
- Refining code: adjust headers, throw exceptions.