Ashoka Summer School

Web Scraping

Outline

Web scraping

Web page

BeautifulSoup Library

Practical Works

Web Scraping - What is it ?

Data Scraping ?

- Automated process
- Explore and download raw data
- Grab content
- Convert data in usable format for analysis
- Store data in database or text file

Web Scraping = Data Scraping of web pages

Web Scraping - What is a web page ?

Components of a web page

- HTML - Organize and contain the main content of a web page
- CSS - Add styling to make the page looks nicer
- JS - Javascript files add interactivity to web pages
- Media files - Images, Sounds, Videos, etc.

Interesting content for web scraping = **HTML**

Web Scraping - HTML

HTML is used to create documents on the Web

Very simple and logical

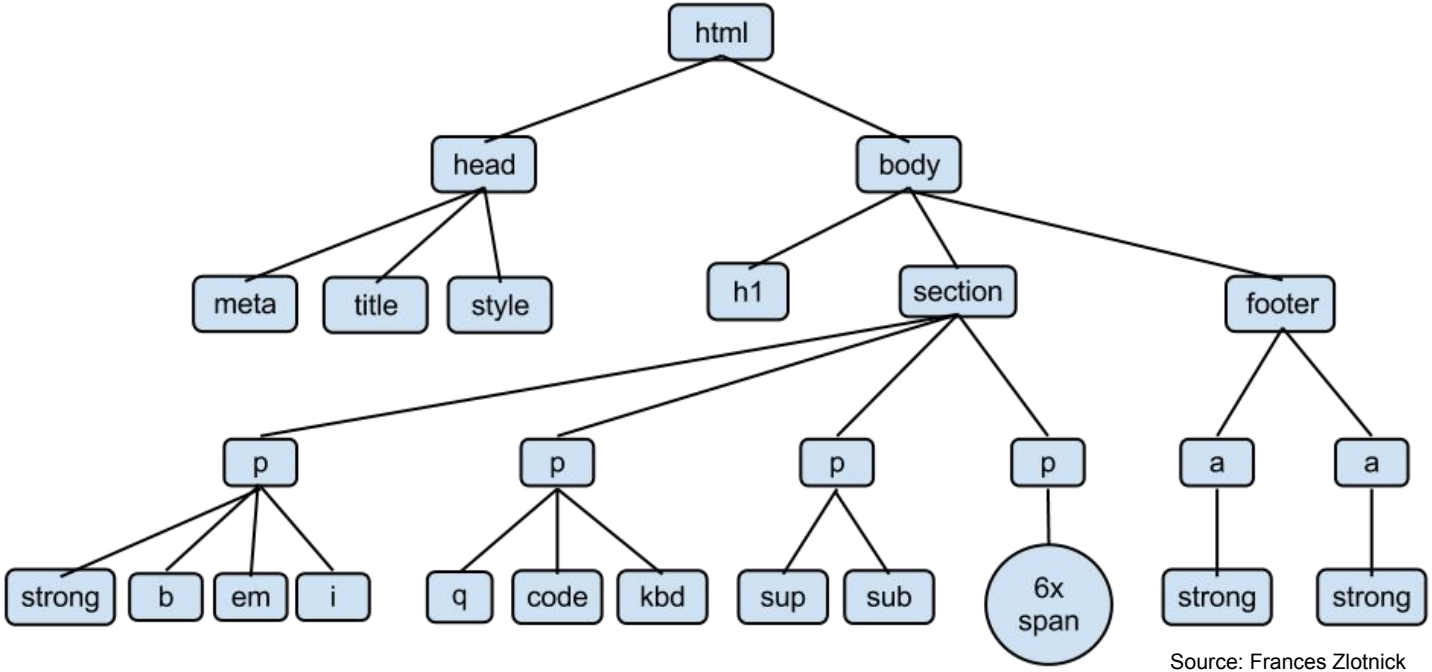
NOT a programming language but a **markups** language which use <tags> like this

The websites you view are basically HTML files rendered by web browsers

```
<!DOCTYPE html>
<html>
    <head>
        <title>Example<title>
        <link rel="stylesheet" href="styl
    </head>
    <body>
        <h1>
            <a href="/">Header</a>
        </h1>
        <nav>
            <a href="one/">One</a>
            <a href="two/">Two</a>
            <a href="three/">Three</a>
        </nav>
```

Web Scraping - HTML

HTML is organized like a hierarchical tree



Source: Frances Zlotnick

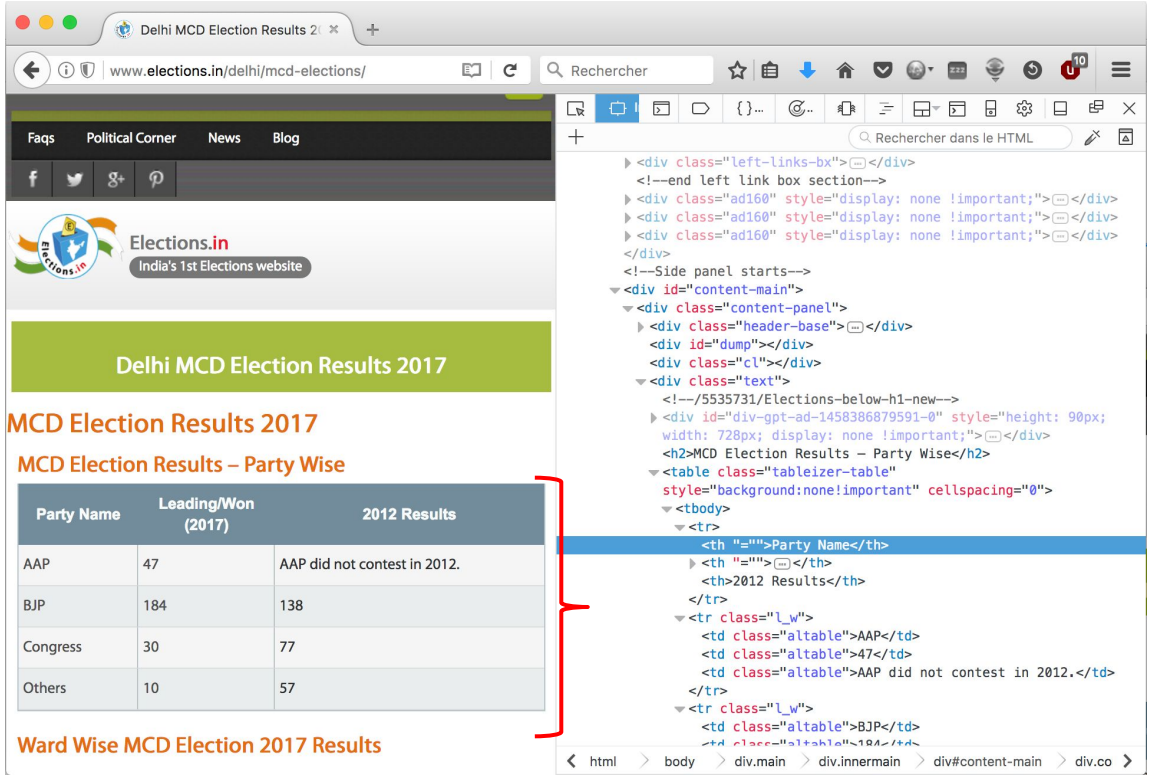
Web Scraping - Inspect the source

Inspect the element

Find de HTML node

<table> defines a table
<tr> defines a row in a table
<th> defines a table header cell
<td> defines a cell in table

Use BeautifulSoup to grab it



Web Scraping - BeautifulSoup

Python library

Pull out data out of HTML/XML files

Designed for quick turnaround projects

Charged with some superb methods

Open-source, free & well documented

Web Scraping - Jump into the code

Grab the node with BeautifulSoup

```
from BeautifulSoup import BeautifulSoup
import urllib

raw_html =
urllib.urlopen('http://www.elections.in/delhi/mcd-elections/').read()

soup = BeautifulSoup(raw_html)

attrs = { 'class':'tableizer-table' }
tables = soup.findAll(attrs=attrs)
table = tables[0]
rows = table.findAll('tr')
```

Import librairies

Download data

Instantiate BeautifulSoup object

Access the data

Web Scraping - Jump into the code

Use grabbed data to write a CSV file

```
import csv

with open('export.csv', 'wb') as f:
 writer = csv.writer(f, delimiter=';')
 for row in rows:
   csv_row = []
   headers = row.findAll('th')
   for header in headers:
     csv_row.append(header.text)
   cells = row.findAll('td')
   for cell in cells:
     csv_row.append(cell.text)
   writer.writerow(csv_row)
```

Import the CSV library

Open a file with write permissions
Handle it with CSV lib's methods

Make loops for selecting data inside table cells.
Write them in a python list

Write the list in the CSV handle file

Web Scraping - Jump into the code

Extraction Result

export.csv — jerem

```
Party Name;Leading/Won(2017);2012 Results
AAP;47;AAP did not contest in 2012.
BJP;184;138
Congress;30;77
Others;10;57
```

Présentation | 125 % Zoom | Formule | Tableau | Graphique

Feuille 1

Tableau 1

	Party Name	Leading/Won(2017)	2012 Results
1	Party Name	Leading/Won(2017)	2012 Results
2	AAP	47	AAP did not contest in 2012.
3	BJP	184	138
4	Congress	30	77
5	Others	10	57

Let’s play !

https://ashoka.cdsp.sciences-po.fr