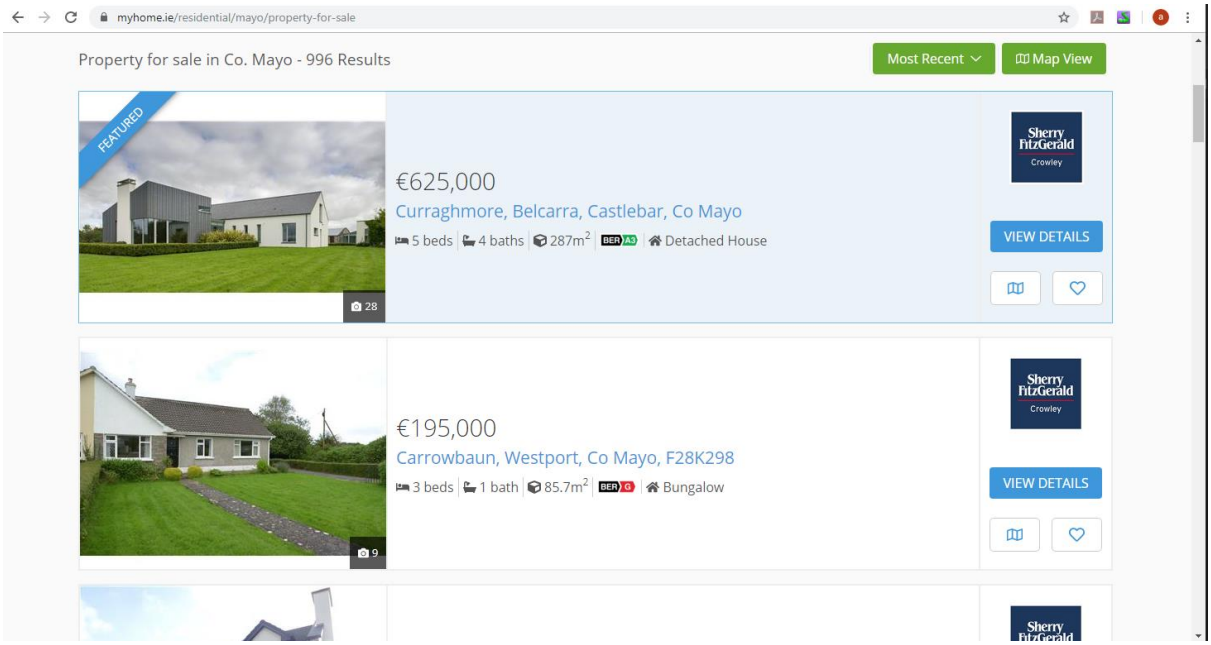
Data Representation
Lab 3: Web Scraping

Lecturer: Andrew Beatty

Extract house prices from myhome.ie and store in a TabSV :



Will be stored in a file like

Price	address
625,000	Curraghmore, Belcarra, Castlebar, Co Mayo
195,000	Croghan, Killala, Mayo

1. Test that we can retrieve a web page from the web. save this file at PY01-testRequest.py in a folder called week03-webScraping

```
import requests
page = requests.get("http://dataquestio.github.io/web-scraping-pages/simple.html")
print (page)
print("-------------------")
print (page.content)
```

2. Test that BeautifulSoup is installed by modifying the program to read

```
import requests
from bs4 import BeautifulSoup
page = requests.get("http://dataquestio.github.io/web-scraping-pages/simple.html")
print (page)
print("-------------------")
print (page.content)
soup1 = BeautifulSoup(page.content, 'html.parser')
print("-------------------")
print (soup1.prettify())
```

3. Test that you can read a file, we will use the carviewer2.html file that we made last week should be up a directory and in the week02 folder ie ("../week02/carviewer2.html")

```
from bs4 import BeautifulSoup

with open("../week02/carviewer2.html") as fp:
    soup = BeautifulSoup(fp,'html.parser')
```

```
print (soup.prettify())
```

(If wish you can save another html file in same directory as this and remove the javascript, I will not be doing this, but it might make the html clearer for you)

Reading the data from our html file

4. Extract the first <tr> from the file (make a file called (PY03-readOutFile.py)

```
from bs4 import BeautifulSoup

with open("../week02/carviewer2.html") as fp:
    soup = BeautifulSoup(fp,'html.parser')
```

```
print (soup.tr)
```

This is the first <tr>. This is not what we want

5. Modify the program to get all the <tr>

```
#print (soup.tr)
rows = soup.findAll("tr")
for row in rows:
     print("------")
     print(row)
```

6. Now for each row let's get the contents of the TD

```
for row in rows:
     #print(row)
     cols = row.findAll("td")
     for col in cols:
         print(col.text)
```

7. Modify this so that the text in the columns are stored in a list

```
dataList = []
    for col in cols:
        dataList.append(col.text)
    print (dataList)
```

8. We want to write this to a CSV file for that we will need the csv package, lets test it. Write a file called PY04-testCSV.py

```
import csv

employee_file = open('employee_file.csv', mode='w')
employee_writer = csv.writer(employee_file, delimiter=',', quotechar='"', quoting=csv.QUOTE_MINIMAL)

employee_writer.writerow(['John Smith', 'Accounting', 'November'])
employee_writer.writerow(['Erica Meyers', 'IT', 'March'])

employee_file.close()
```

Look at the directory and check if an employee_file.csv was made

Bring it together

9. Make a file called PY05-readFileFinal.py, copy in the code from PY03-readOutFile.py, that brings it all together

```
from bs4 import BeautifulSoup
import csv

with open("../week02/carviewer2.html") as fp:
    soup = BeautifulSoup(fp,'html.parser')

#print (soup.tr)
employee_file = open('week02data.csv', mode='w')
employee_writer = csv.writer(employee_file, delimiter=',', quotechar='"', quoting=csv.QUOTE_MINIMAL)

rows = soup.findAll("tr")
for row in rows:

    cols = row.findAll("td")
    dataList = []
    for col in cols:
        dataList.append(col.text)
    employee_writer.writerow(dataList)
employee_file.close()
```

10. How would you modify the code so that the update and delete text is not outputted?

Tricky bit: Read data from myhome.ie

11. Open myhome.ie in a browser and search for houses for sale in a county, you will see that the search results are in multiple pages navigate to the second page and note the URL

    https://www.myhome.ie/residential/mayo/property-for-sale?page=1

12. Create a file called py06-myhome.py, and write the code to read the page. (this may take a little time to run

```
<import requests

from bs4 import BeautifulSoup
page = requests.get("https://www.myhome.ie/residential/
mayo/property-for-sale?page=1")

soup = BeautifulSoup(page.content, 'html.parser')
print (soup.prettify())
```

13. Look at the do a view page source, search for some text you recognise, I did a search for "belcarra" and looked at the containing divs and saw that the listing are in a containing div with class=" PropertyListingCard"

14. Modify the code to retrieve the first with class="PropertyListingCard "

```
listings = soup.find("div", class_="PropertyListingCard" )

print (listings)
```

15. Modify that code to get the price
    `class="PropertyListingCard__Price"`

```
price = listings.find(class_="PropertyListingCard__Price").text

print (price)
```

16. Also get the address

```
price = listings.find(class_="PropertyListingCard__Price").text

print (price)
```

17. Now get all the entries in the page and store each in a list

```
listings = soup.findAll("div", class_="PropertyListingCard" )

for listing in listings:
    entry = []

    price = listing.find(class_="PropertyListingCard__Price").text
    entry.append(price)
    address = listing.find(class_="PropertyListingCard__Address").text
    entry.append(address)

    print(entry)
```

18. Now output into a CSV with tabs as delimiters, (you can open this in excel if you wish)

```
import requests
import csv
from bs4 import BeautifulSoup
page = requests.get("https://www.myhome.ie/residential/mayo/property-for-
sale?page=1")

soup = BeautifulSoup(page.content, 'html.parser')

home_file = open('week03MyHome.csv', mode='w')
home_writer = csv.writer(home_file, delimiter='\t', quotechar='"', quoting=csv
.QUOTE_MINIMAL)

listings = soup.findAll("div", class_="PropertyListingCard" )

for listing in listings:
    entryList = []

    price = listing.find(class_="PropertyListingCard__Price").text
    entryList.append(price)
    address = listing.find(class_="PropertyListingCard__Address").text
    entryList.append(address)

    home_writer.writerow(entryList)
home_file.close()
```