Python: Data from the Web

Name: Rudy Martinez

abc123: Lpe538

Blank notebook to be used for class exercises.

Exercise 1

Write code to query http://duckduckgo.com/html/ with the key query "data science". Parse the resulting page by returning all the **unique** web URLs. Return only the base URLs (http://duckduckgo.com, www.duckduckgo.com, ...)

hint: Use re.findall()

In [11]:
```
import urllib.parse, urllib.request
from bs4 import BeautifulSoup
import re

baseurl = "http://duckduckgo.com/html?"

query = "Data Science"
params = {'q': query}

url = baseurl + urllib.parse.urlencode(params)
file = urllib.request.urlopen(url)
html_page = file.read().decode()

re.findall(r"www\.[^\" \n]+|http[^\" \n]+", html_page)
```

Out[11]:
```
['http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd',
 'http://www.w3.org/1999/xhtml',
 'http://www.w3.org/1999/xhtml',
 'http://www.w3.org/1999/xhtml',
 'http://www.w3.org/1999/xhtml',
 'http-equiv=',
 'https://en.wikipedia.org/wiki/Data_science',
 'https://en.wikipedia.org/wiki/Data_science',
 'https://i.duckduckgo.com/i/d8c64240.jpg',
 'https://en.wikipedia.org/wiki/Data_science',
 'https%3A%2F%2Fen.wikipedia.org%2Fwiki%2FData_science&amp;rut=262bcdd083c58fc75c0dc2bba7158769108a7b6749edc15a
```

```
99f06ef4c73d3d4e',
 'https%3A%2F%2Fen.wikipedia.org%2Fwiki%2FData_science&amp;rut=262bcdd083c58fc75c0dc2bba7158769108a7b6749edc15a
99f06ef4c73d3d4e',
 'https%3A%2F%2Fen.wikipedia.org%2Fwiki%2FData_science&amp;rut=262bcdd083c58fc75c0dc2bba7158769108a7b6749edc15a
99f06ef4c73d3d4e',
 'https%3A%2F%2Fen.wikipedia.org%2Fwiki%2FData_science&amp;rut=262bcdd083c58fc75c0dc2bba7158769108a7b6749edc15a
99f06ef4c73d3d4e',
 'https%3A%2F%2Fwww.edureka.co%2Fblog%2Fwhat%2Dis%2Ddata%2Dscience%2F&amp;rut=3549453c95c1ae3f9e8a6b2ea55330f16
e423c97c8970047033012be9c2a365f',
 'https%3A%2F%2Fwww.edureka.co%2Fblog%2Fwhat%2Dis%2Ddata%2Dscience%2F&amp;rut=3549453c95c1ae3f9e8a6b2ea55330f16
e423c97c8970047033012be9c2a365f',
 'www.edureka.co.ico',
 'https%3A%2F%2Fwww.edureka.co%2Fblog%2Fwhat%2Dis%2Ddata%2Dscience%2F&amp;rut=3549453c95c1ae3f9e8a6b2ea55330f16
e423c97c8970047033012be9c2a365f',
 'www.edureka.co/blog/what-is-data-science/',
 'https%3A%2F%2Fwww.edureka.co%2Fblog%2Fwhat%2Dis%2Ddata%2Dscience%2F&amp;rut=3549453c95c1ae3f9e8a6b2ea55330f16
e423c97c8970047033012be9c2a365f',
 'https%3A%2F%2Fbuiltin.com%2Fdata%2Dscience&amp;rut=8f6a95ea27ffb6207916f3ce1cd48d8ac9c5bbe98c96763a8fddd22440
34a5a7',
 'https%3A%2F%2Fbuiltin.com%2Fdata%2Dscience&amp;rut=8f6a95ea27ffb6207916f3ce1cd48d8ac9c5bbe98c96763a8fddd22440
34a5a7',
 'https%3A%2F%2Fbuiltin.com%2Fdata%2Dscience&amp;rut=8f6a95ea27ffb6207916f3ce1cd48d8ac9c5bbe98c96763a8fddd22440
34a5a7',
 'https%3A%2F%2Fbuiltin.com%2Fdata%2Dscience&amp;rut=8f6a95ea27ffb6207916f3ce1cd48d8ac9c5bbe98c96763a8fddd22440
34a5a7',
 'https%3A%2F%2Fwiki2.org%2Fen%2FData_science&amp;rut=c375c76f317a5f5d8d80be83387409af937534f7e29e3a07a2eb0a20e
8bce7ed',
 'https%3A%2F%2Fwiki2.org%2Fen%2FData_science&amp;rut=c375c76f317a5f5d8d80be83387409af937534f7e29e3a07a2eb0a20e
8bce7ed',
 'https%3A%2F%2Fwiki2.org%2Fen%2FData_science&amp;rut=c375c76f317a5f5d8d80be83387409af937534f7e29e3a07a2eb0a20e
8bce7ed',
 'https%3A%2F%2Fwiki2.org%2Fen%2FData_science&amp;rut=c375c76f317a5f5d8d80be83387409af937534f7e29e3a07a2eb0a20e
8bce7ed']
```

Exercise 2

Find another table on Wikipedia and use the BeautifulSoup package to parse the table.

In [31]:

```
import urllib
import requests
import pandas as pd
from bs4 import BeautifulSoup

wiki_url = 'https://en.wikipedia.org/wiki/List_of_S%26P_500_companies'       #Acquires Wikipedia page content f
response_1 = requests.get(wiki_url)
company_page_content = BeautifulSoup(response_1.text, 'html.parser')
```

```
table_id = "constituents"                                          #Stores the table with company inf
company_table = company_page_content.find('table', attrs={'id': table_id})

df = pd.read_html(str(company_table))                              #Creates a dataframe with company
df[0].to_csv('00. S&P500 Company Information.csv')

comp_df = pd.read_csv('00. S&P500 Company Information.csv')
comp_df.head()
```

Out[31]:

	Unnamed: 0	Symbol	Security	SEC filings	GICS Sector	GICS Sub-Industry	Headquarters Location	Date first added	CIK	Founded
0	0	MMM	3M Company	reports	Industrials	Industrial Conglomerates	St. Paul, Minnesota	1976-08-09	66740	1902
1	1	ABT	Abbott Laboratories	reports	Health Care	Health Care Equipment	North Chicago, Illinois	1964-03-31	1800	1888
2	2	ABBV	AbbVie Inc.	reports	Health Care	Pharmaceuticals	North Chicago, Illinois	2012-12-31	1551152	2013 (1888)
3	3	ABMD	Abiomed	reports	Health Care	Health Care Equipment	Danvers, Massachusetts	2018-05-31	815094	1981
4	4	ACN	Accenture	reports	Information Technology	IT Consulting & Other Services	Dublin, Ireland	2011-07-06	1467373	1989

In [ ]: