DigitalHouse > Coding School

DATA SCIENCE

UNIDAD 1
MÓDULO 2

Web Scraping con Python

Marzo 2017

OBJETIVOS DE ESTA CLASE

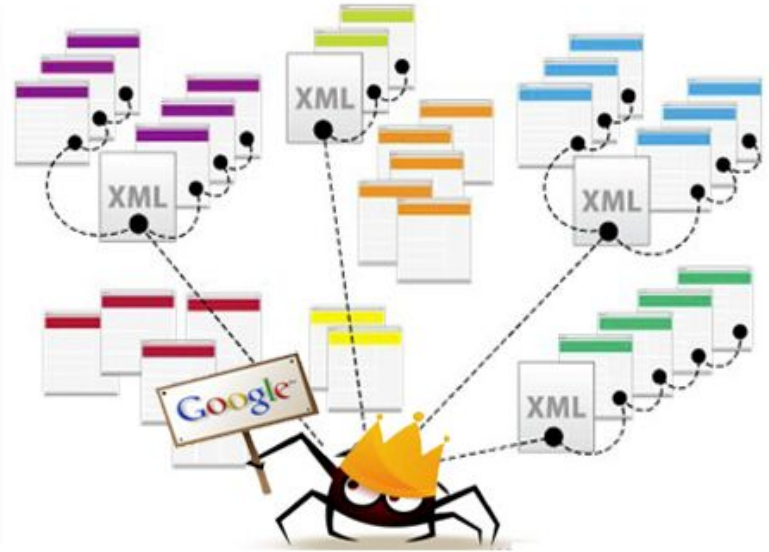
1. **Describir los fundamentos del web scraping**

2. **Conocer cómo implementar web scraping usando Python**
   - **Requests**
   - **BeautifulSoup**
   - **Regex** **https://regexr.com/**

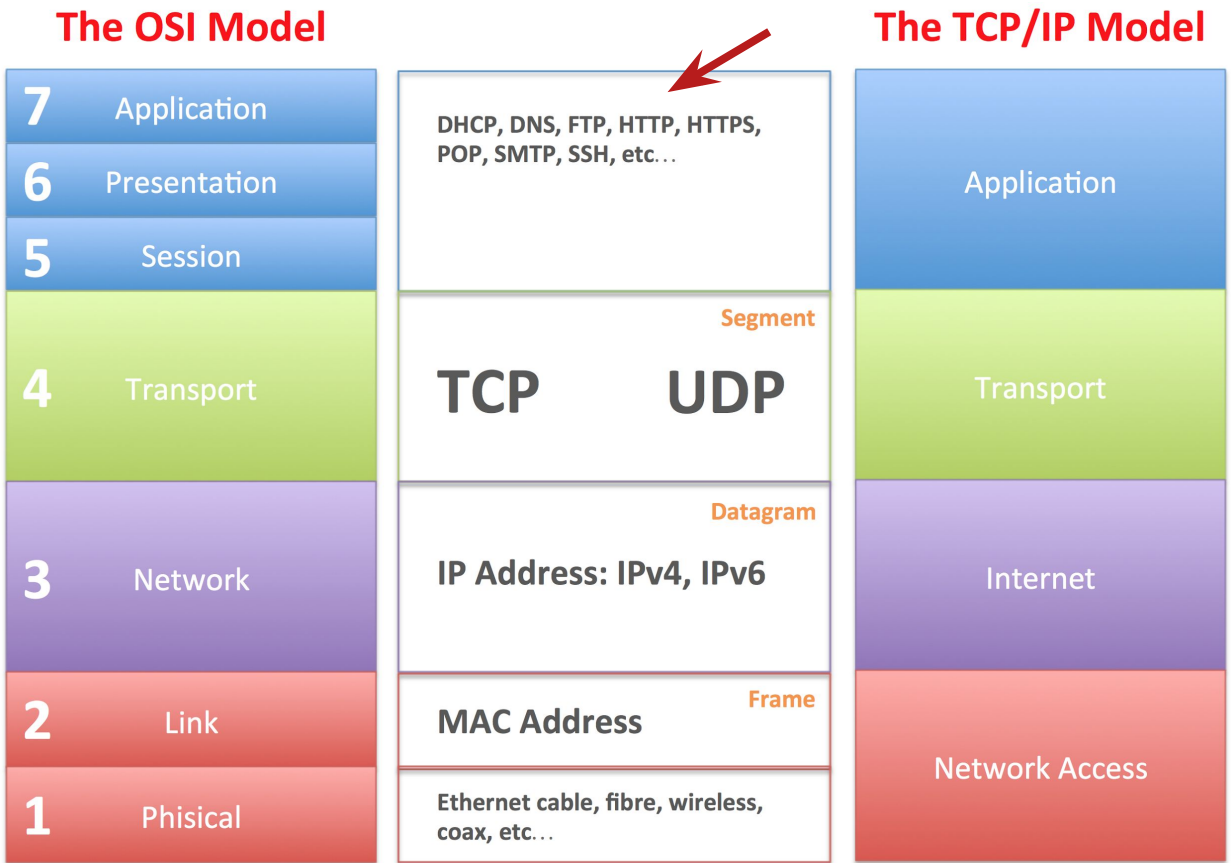
El web scraping es la extracción automática de información de la web mediante el uso de bots o crawlers.

Web Crawler

El web crawler o *spider* recorre e indexa las páginas web para el procesamiento de su contenido.



Internet

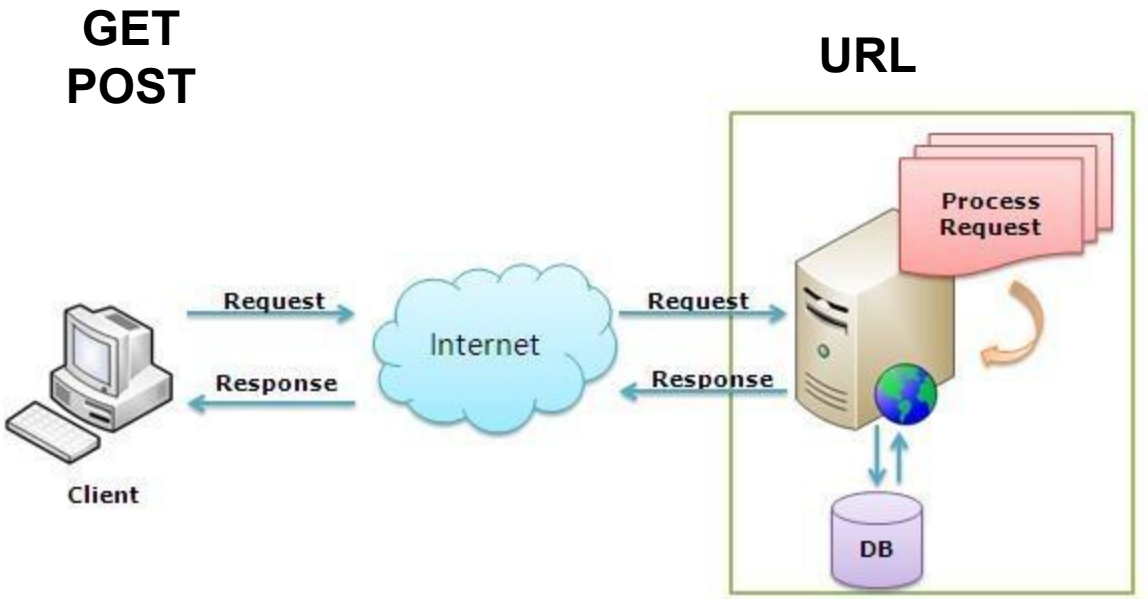

The OSI Model
The TCP/IP Model
7 Application
6 Presentation
5 Session
4 Transport
3 Network
2 Link
1 Phisical
DHCP, DNS, FTP, HTTP, HTTPS, POP, SMTP, SSH, etc…
Segment
TCP
UDP
Datagram
IP Address: IPv4, IPv6
Frame
MAC Address
Ethernet cable, fibre, wireless, coax, etc…
Application
Transport
Internet
Network Access

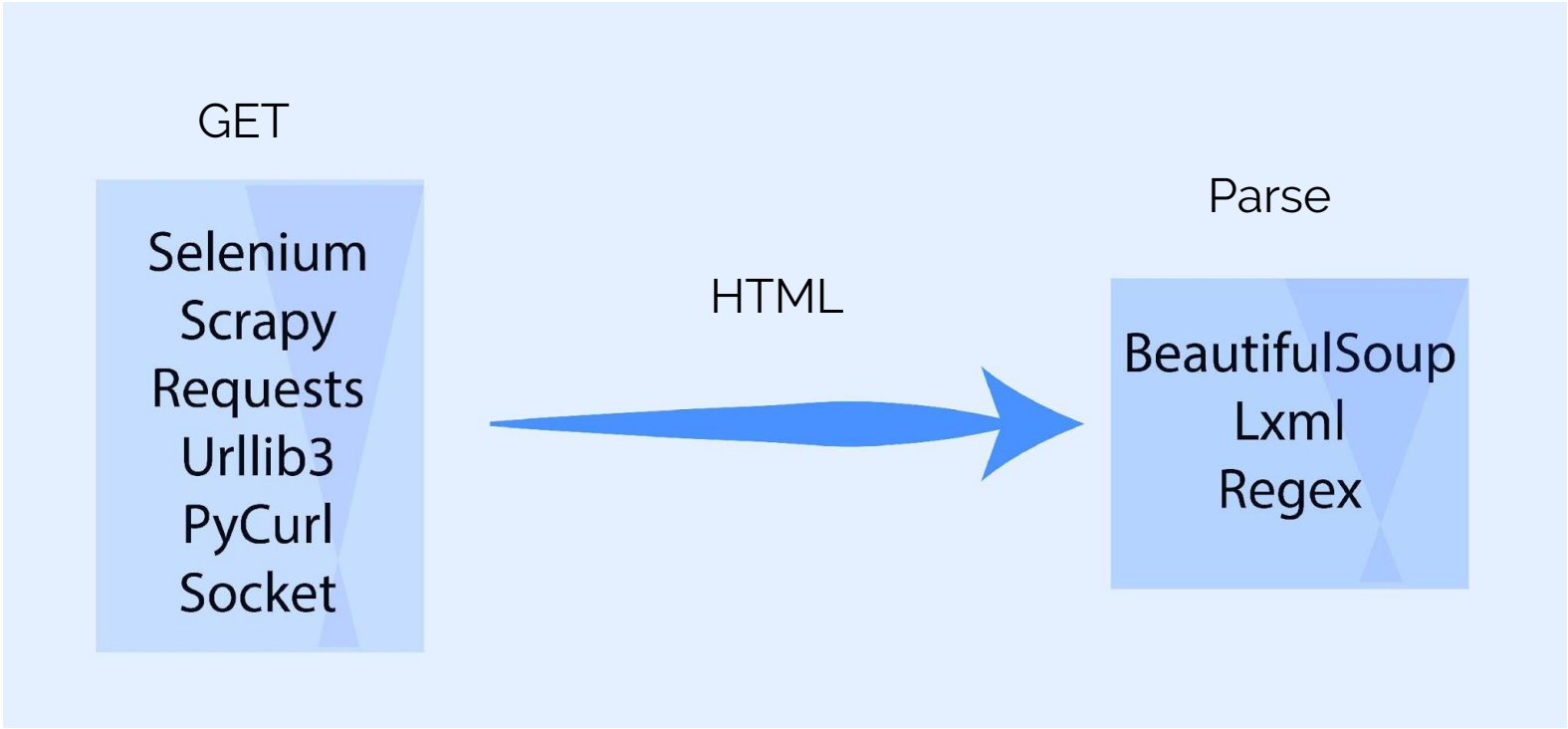
headers

```
HTTP/1.1 200 OK
Date: Mon, 27 Jul 2009 12:28:53 GMT
Server: Apache/2.2.14 (Win32)
Last-Modified: Wed, 22 Jul 2009 19:15:56 GMT
Content-Length: 88
Content-Type: text/html
Connection: Closed
```

content

```
<html>
   <body>

   <h1>Hello, World!</h1>

   </body>
</html>
```


GET
POST
URL
Process
Request
Request
Response
Internet
Request
Response
Client
DB

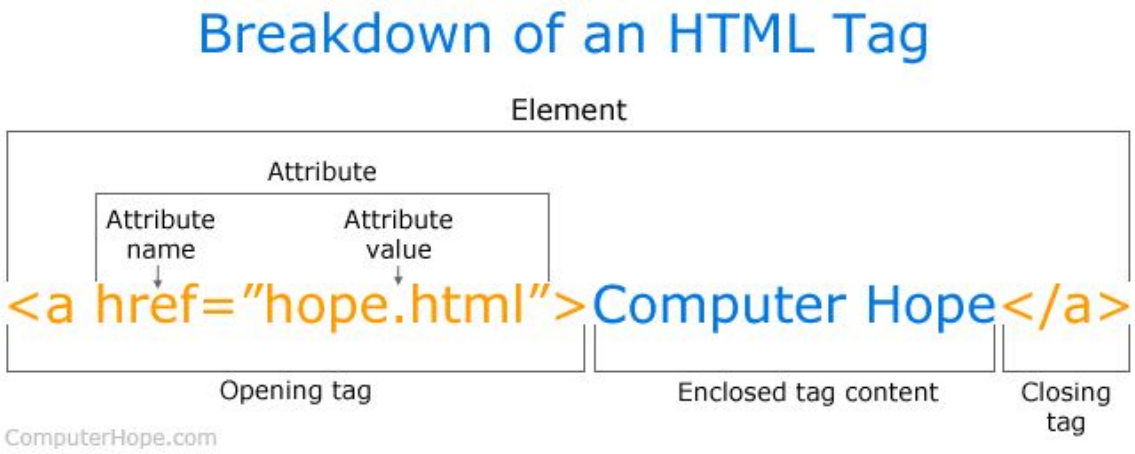

GET
Selenium
Scrapy
Requests
Urllib3
PyCurl
Socket
HTML
Parse
BeautifulSoup
Lxml
Regex

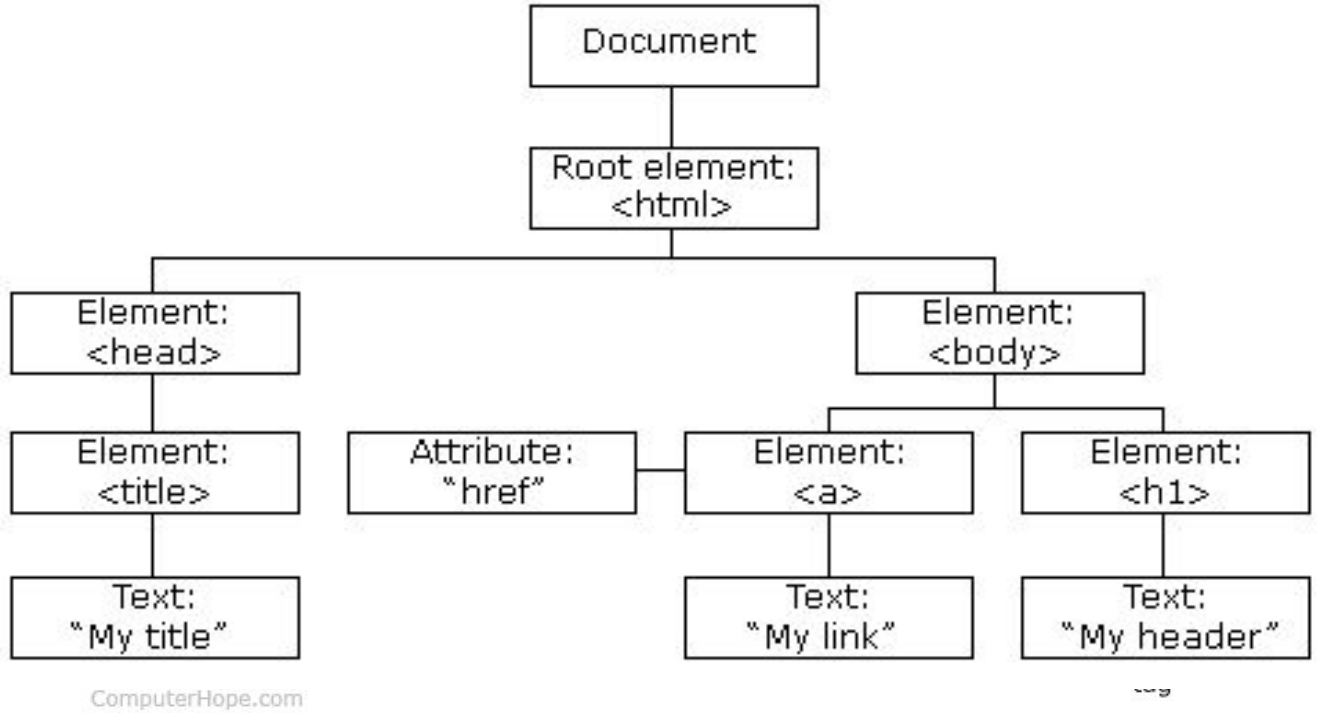
- El *HyperText Markup Language* es el lenguaje de marcado (*markup language)* estándar en la web.
- Permite armar textos estructurados enriquecidos con archivos multimedia. Junto con CSS y JavaScript forman la tríada de tecnologías centrales de la Web.

```
<!DOCTYPE html>
    <html>

        <head>
            <title> Title here </title>
        </head>

        <body>
            Web page content goes here.
        </body>

    </html>
```

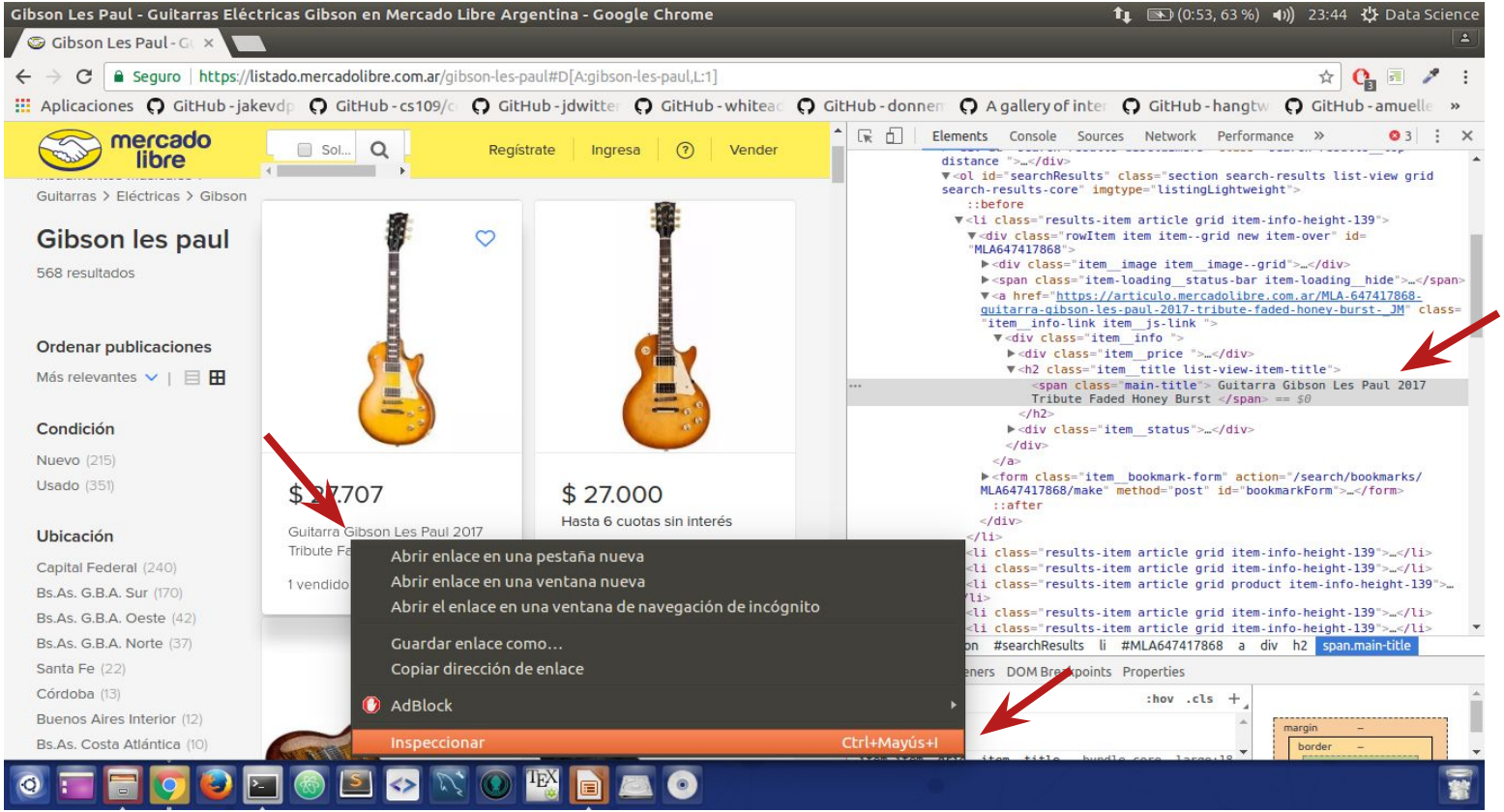
Se compone por elementos semánticos delimitados por etiquetas o *tags*, que se encapsulan en ángulos <>.

Los mismos pueden poseer atributos separados con espacio y con el valor entre comillas




Document
Root element:
<html>
Element:
<head>
Element:
<body>
Element:
<title>
Attribute:
"href"
Element:
<a>
Element:
<h1>
Text:
"My title"
Text:
"My link"
Text:
"My header"
ComputerHope.com

Document Object Model



Entre las opciones para usar HTTP con Python, la librería recomendada oficialmente es Requests.

Está escrita sobre urllib3 y http.client, y ofrece un cómodo nivel de abstracción.

Resuelve fácilmente autenticaciones, HTTPS, persistencia de cookies...

- Una expresión regular es una secuencia de caracteres que determina un patrón de búsqueda

- Es un lenguaje muy flexible que sirve para identificar y extraer información de un cuerpo de caracteres no estructurado.

()	capturing group	(?:)	non-capturing group
\w	carácter alfanumérico	.	cualquier cosa menos \n
\d	dígito	|	operador “or”
\s	whitespace	[m-z3-9]	rangos
[ ]	conjunto	Cuantificadores:	

+ uno o más del elemento anterior

* cero o más del elemento anterior

{4,} cuatro o más del elemento anterior

? cambia el operador anterior de "greedy" a "lazy". Ejemplo:

Lazy
(.+?)<

contacto@digitalhouse.comDigital House

Greedy (.+)< contacto@digitalhouse.comDigital House

Beautiful Soup

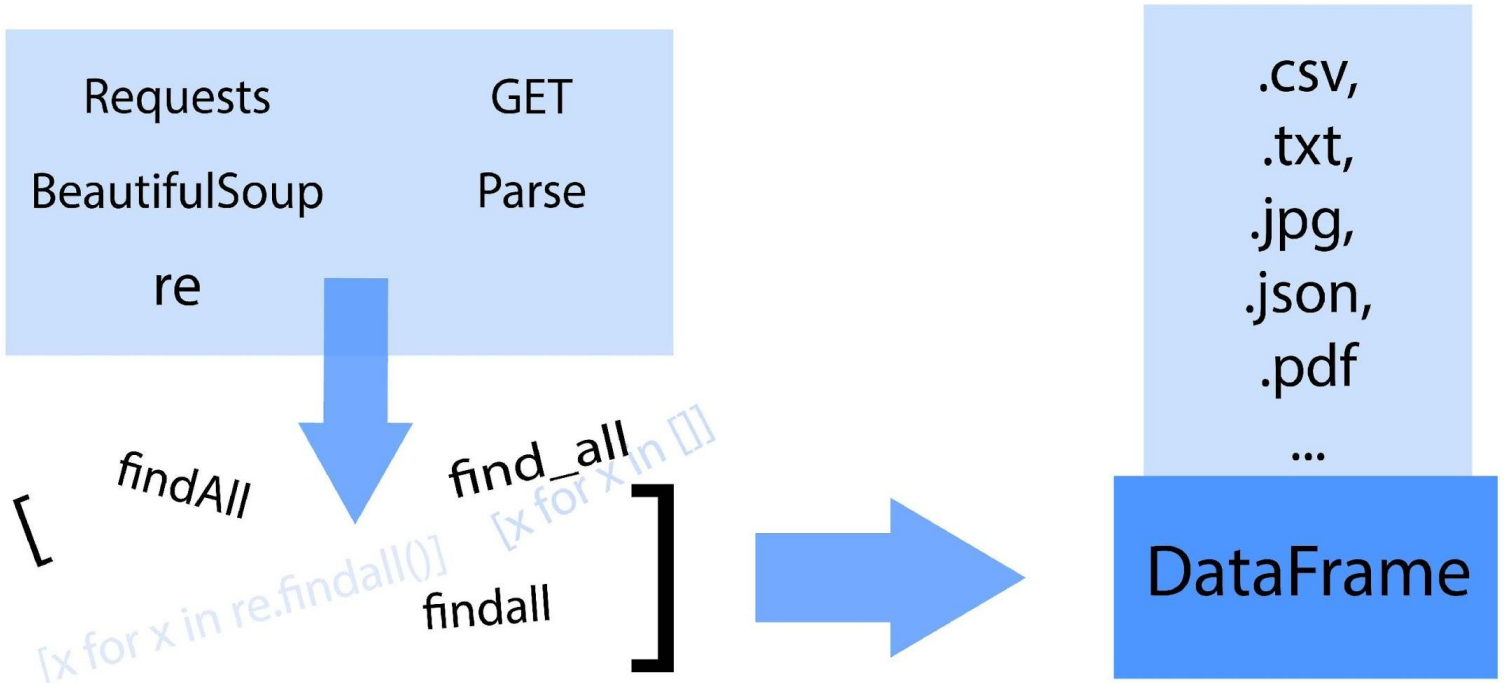
Esta librería permite obtener la información contenida en HTML/XML y extraerla en un formato ordenado.

Tolera documentos HTML mal formados, ya que está implementado con Regular Expressions, y no parseando el HTML a una estructura DOM.

Permite distintos parsers, entre los que se incluye lxml.

Práctica Guiada I

Práctica Guiada II


Requests
GET
BeautifulSoup
Parse
re
findAll
find_all
findall
.csv,
.txt,
.jpg,
.json,
.pdf
...
DataFrame

Laboratorio

- Esta técnica permite obtener información no estructurada y normalmente poco accesible en un formato simple para un posterior análisis.

- El web scraping es de interés cuando se necesita acceder a información dinámica.

- Python nos facilita el acceso a la información de interés sin la necesidad de tener un conocimiento exhaustivo de HTML o XML, utilizando librerías de distinto nivel entre las cuales están Requests y BeautifulSoup.

Para persistencia de cookies:

requests.Session()

Cuando se necesita ejecutar JavaScript:

Selenium

Para mayor rendimiento temporal:

Lxml

Extras de infraestructura:

Threading

Queue