# PYTHON

Web scraping

# WEB SCRAPING

- Técnica consistente en extraer información de la web tomando como fuentes documentos HTML.
- En Python la biblioteca Beautiful Soup facilita la realización de esta técnica.
- https://www.crummy.com/software/BeautifulSoup/
- Instalación:
  - pip install beautifulsoup4

  o
  - pip install bs4 (alias de beautifulsoup4)

# WEB SCRAPING

- Fases:
  - Obtención del código HTML
  - Conversión a un objeto de la clase BeautifulSoup.
  - Análisis.

# WEB SCRAPING

- Fases:
  - Obtención del código HTML

```
from urllib.request import urlopen
from urllib.error import HTTPError
from urllib.error import URLError
try:
    html =
urlopen("https://www.python.org/")
except HTTPError:
    print("Error HTTP")
except URLError:
    print("Error URL")
else:
    print(html.read())
```

# WEB SCRAPING

- Fases:
  - Conversión a un objeto de la clase BeautifulSoup.

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
```

# WEB SCRAPING

- Fases:
  - Análisis. Métodos:
    - `bs.h1#El primer h1`
    - `bs.h1.a)#a que está dentro del primer h1`
    - `bs.h1.text#El texto contenido en el elemento h1`
    - `bs.a.attrs['href']#El valor del atributo href del elemento a`
    - `bs.findAll("h1")#Todos los h1`
    - `bs.findAll("h1",{'class':'site-headline'})#Todos los h1 de la clase site-headline`
    - `bs.findAll(["h1","h2","h3"])#Todos los h1, h2, h3`

# WEB SCRAPING

- Fases:
  - Análisis. Métodos:
    - `bs.findAll("span",{"class":{"icon-get-started","icon-download"}})#Todos los span de las clases indicada`
    - `bs.findAll(class_="icon-get-started")#Los elementos de la clase indicada (atención al atributo _class)`
    - `bs.find(id="close-python-network")#Elemento con el id indicado`
    - `bs.find(id="close-python-network").children #Descendiente del elemento seleccionado`
    - `bs.find(id="touchnav-wrapper").next_siblings #Elemento siguiente (hermano)`
    - `bs.find("h3", string="Película") #Elemento h3 con el texto Película`

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtención de un documento html.

```
from urllib.request import urlopen
html = urlopen("https://www.python.org/")
print(html.read())
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtención de un documento HTML con control de errores.

```
from urllib.request import urlopen
from urllib.error import HTTPError
from urllib.error import URLError
try:
    html = urlopen("https://www.python.org/")
except HTTPError:
    print("Error HTTP")
except URLError:
    print("Error URL")
else:
    print(html.read())
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtención del primer elemento h1

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
print(bs.h1)#Primer h1
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtención del elemento a que está dentro del primer elemento h1

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
print(bs.h1.a)#a que está dentro de un h1
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtención de todos los elementos h1

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
h1s =bs.findAll("h1")#Todos los h1
for h1 in h1s:
    print(h1)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: obtención de los textos contenidos en todos los elementos `h1`

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
h1s =bs.findAll("h1")#Todos los h1
for h1 in h1s:
    print(h1.text)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtención de todos los elementos a contenidos en todos los elementos h1

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
h1s =bs.findAll("h1")#Todos los h1
for h1 in h1s:
    enlaces = h1.findAll("a")
    if (len(enlaces)>0):
        for enlace in enlaces:
            print(enlace)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtención de todos los elementos h1 de la clase 'site-headline'

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
h1s =bs.findAll("h1",{'class':'site-headline'})#Todos los h1
de la clase site-headline
for h1 in h1s:
    print(h1)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtención de todos los elementos h1, h2 y h3

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
hs =bs.findAll(["h1","h2","h3"])#Todos los h1, h2, h3
for h in hs:
    print(h)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtención de todos los elementos span de las clases “icon-get-started” y “icon-download”.

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
spans =bs.findAll("span",{"class":{"icon-get-started","icon-download"}})#Todos los
span de las clases indicadas en el set
for span in spans:
    print(span)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Todos los elementos de clase “icon-get_started”. El atributo referente a la clase es **class_**

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
elementos = bs.findAll(class_="icon-get-started")#Los elementos de la
clase indicada (atención al atributo _class)
for elemento in elementos:
    print(elemento)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtiene el elemento con el id “close-python-network”

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
elemento =bs.find(id="close-python-network")
print(elemento)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtiene los elementos descendientes (hijos) del elemento con el id “close-Python-network”

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
elemento =bs.find(id="close-python-network")
hijos = elemento.children
for hijo in hijos:
    print(hijo)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtiene los elementos hermanos siguientes del elemento con el id “touchnav-wrapper”

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.python.org/")
bs = BeautifulSoup(html.read(), 'html.parser')
elemento =bs.find(id="touchnav-wrapper")
hermanos = elemento.next_siblings
for hermano in hermanos:
    print(hermano)
```

# WEB SCRAPING

- Fases:
  - Ejemplos: Obtiene los valores del atributo href de todos los elementos a

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen('https://www.python.org/')
bs = BeautifulSoup(html, 'html.parser')
for link in bs.find_all('a'):
    if 'href' in link.attrs:
        print(link.attrs['href'])
```