Web Scraping en Python: BeautifulSoup (II)

En esta sesión vamos a navegar hacia abajo (hijos o descendientes) y hacia arriba (parents) en una estructura HTML a partir de un nodo determinado. Y también a encontrar los Tags dentro de otros tags con métodos similares a los empleados con XML (es decir algo como findall pero que aquí se llama find_all)

Lo primero es importar y recuperar nuestra "página" ejemplo:

[1]:
```
import lxml
import pandas as pd
import requests
from bs4 import BeautifulSoup

# Suponemos que esta pagina de ejemplo la hemos descargado con una llamada a la función request de la librería requests

contenido = """
<html lang="es">
<head>
    <meta charset="UTF-8">
    <title>Página de prueba</title>
</head>
<body>
<div id="main" class="full-width">
    <h1>El título de la página</h1>
    <p>Este es el primer párrafo</p>
    <p>Este es el segundo párrafo</p>
    <div id="innerDiv">
        <div class="links">
            <a href="https://pagina1.xyz/">Enlace 1</a>
            <a href="https://pagina2.xyz/">Enlace 2</a>
        </div>
        <div class="right">
            <div class="links">
                <a href="https://pagina3.xyz/">Enlace 3</a>
                <a href="https://pagina4.xyz/">Enlace 4</a>
            </div>
        </div>
    </div>
    <div id="footer">
        <!-- El footer -->
        <p>Este párrafo está en el footer</p>
        <div class="links footer-links">
            <a href="https://pagina5.xyz/">Enlace 5</a>
        </div>
    </div>
</div>
</body>
</html>
```

"""
soup = BeautifulSoup(contenido, 'lxml')

Y ahora a "navegar"

Navegar a través de los elementos de Beautiful Soup

Hijos

- El atributo **contents**: Devuelve una lista con todos los hijos de primer nivel de un objeto.
- Atributo **descendants**: Este atributo devuelve un iterador que permite recorrer todos los hijos de un objeto. No importa el nivel de anidamiento.

```
inner_div = soup.div.div
inner_div.contents
```

```
['\n',
 <div class="links">
 <a href="https://pagina1.xyz/">Enlace 1</a>
 <a href="https://pagina2.xyz/">Enlace 2</a>
 </div>,
 '\n',
 <div class="right">
 <div class="links">
 <a href="https://pagina3.xyz/">Enlace 3</a>
 <a href="https://pagina4.xyz/">Enlace 4</a>
 </div>
 </div>,
 '\n']
```

Se ve, pero hay unos saltos de línea y esa forma no tabulada que podemos hacer un poco más estética (fijate además en el atributo name)

```
inner_div = soup.div.div
# contents
hijos = inner_div.contents
print(type(hijos))
print("\n")
for child in hijos:
    if child.name:  # Ignoramos los saltos de línea
        print(f'{child.name}:\n {child.prettify()}')
        print("\n")
```

```
<class 'list'>


div:
 <div class="links">
 <a href="https://pagina1.xyz/">
  Enlace 1
```

```
 </a>
 <a href="https://pagina2.xyz/">
  Enlace 2
 </a>
</div>
```

```
div:
 <div class="right">
 <div class="links">
  <a href="https://pagina3.xyz/">
   Enlace 3
  </a>
  <a href="https://pagina4.xyz/">
   Enlace 4
  </a>
 </div>
</div>
```

Veamos descendants:

```
for hijo in inner_div.descendants:
    print(hijo)
```

```
<div class="links">
<a href="https://pagina1.xyz/">Enlace 1</a>
<a href="https://pagina2.xyz/">Enlace 2</a>
</div>
```

```
<a href="https://pagina1.xyz/">Enlace 1</a>
Enlace 1
```

```
<a href="https://pagina2.xyz/">Enlace 2</a>
Enlace 2
```

```
<div class="right">
<div class="links">
<a href="https://pagina3.xyz/">Enlace 3</a>
<a href="https://pagina4.xyz/">Enlace 4</a>
</div>
</div>
```

```
<div class="links">
<a href="https://pagina3.xyz/">Enlace 3</a>
<a href="https://pagina4.xyz/">Enlace 4</a>
</div>
```

```
<a href="https://pagina3.xyz/">Enlace 3</a>
```

Enlace 3

```
<a href="https://pagina4.xyz/">Enlace 4</a>
```

Enlace 4

```
# descendants
hijos = inner_div.descendants
print(hijos)
print("\n")
for child in hijos:
    if child.name:
        print(f'{child.name}:\n {child}')
        print("\n")
```

```
<generator object Tag.descendants at 0x7fdd40136a50>
```

```
div:
 <div class="links">
<a href="https://pagina1.xyz/">Enlace 1</a>
<a href="https://pagina2.xyz/">Enlace 2</a>
</div>
```

```
a:
 <a href="https://pagina1.xyz/">Enlace 1</a>
```

```
a:
 <a href="https://pagina2.xyz/">Enlace 2</a>
```

```
div:
 <div class="right">
<div class="links">
```

Enlace 3
Enlace 4

div:

Enlace 3
Enlace 4

a:
Enlace 3

a:
Enlace 4

Padres

Además de a los hijos, es posible navegar hacia arriba en el árbol accediendo a los objetos padre de un elemento. Para ello, puedes usar las propiedades parent y parents:

- **parent** referencia al objeto padre de un elemento (Tag o NavigableString).
- **parents** es un generador que permite recorrer recursivamente todos los elementos padre de uno dado.

inner_div.parent

<h1>El título de la página</h1>
Este es el primer párrafo
Este es el segundo párrafo

Enlace 1
Enlace 2

Enlace 3
Enlace 4

```
<!-- El footer -->
<p>Este párrafo está en el footer</p>
<div class="links footer-links">
<a href="https://pagina5.xyz/">Enlace 5</a>
</div>
</div>
</div>

for parent in inner_div.parents:
    print(parent)
    print("\n\n")

<div class="full-width" id="main">
<h1>El título de la página</h1>
<p>Este es el primer párrafo</p>
<p>Este es el segundo párrafo</p>
<div id="innerDiv">
<div class="links">
<a href="https://pagina1.xyz/">Enlace 1</a>
<a href="https://pagina2.xyz/">Enlace 2</a>
</div>
<div class="right">
<div class="links">
<a href="https://pagina3.xyz/">Enlace 3</a>
<a href="https://pagina4.xyz/">Enlace 4</a>
</div>
</div>
</div>
<div id="footer">
<!-- El footer -->
<p>Este párrafo está en el footer</p>
<div class="links footer-links">
<a href="https://pagina5.xyz/">Enlace 5</a>
</div>
</div>
</div>



<body>
<div class="full-width" id="main">
<h1>El título de la página</h1>
<p>Este es el primer párrafo</p>
<p>Este es el segundo párrafo</p>
<div id="innerDiv">
<div class="links">
<a href="https://pagina1.xyz/">Enlace 1</a>
<a href="https://pagina2.xyz/">Enlace 2</a>
</div>
<div class="right">
```

```
<div class="links">
<a href="https://pagina3.xyz/">Enlace 3</a>
<a href="https://pagina4.xyz/">Enlace 4</a>
</div>
</div>
</div>
<div id="footer">
<!-- El footer -->
<p>Este párrafo está en el footer</p>
<div class="links footer-links">
<a href="https://pagina5.xyz/">Enlace 5</a>
</div>
</div>
</div>
</body>
```

```
<html lang="es">
<head>
<meta charset="utf-8"/>
<title>Página de prueba</title>
</head>
<body>
<div class="full-width" id="main">
<h1>El título de la página</h1>
<p>Este es el primer párrafo</p>
<p>Este es el segundo párrafo</p>
<div id="innerDiv">
<div class="links">
<a href="https://pagina1.xyz/">Enlace 1</a>
<a href="https://pagina2.xyz/">Enlace 2</a>
</div>
<div class="right">
<div class="links">
<a href="https://pagina3.xyz/">Enlace 3</a>
<a href="https://pagina4.xyz/">Enlace 4</a>
</div>
</div>
</div>
<div id="footer">
<!-- El footer -->
<p>Este párrafo está en el footer</p>
<div class="links footer-links">
<a href="https://pagina5.xyz/">Enlace 5</a>
</div>
</div>
</div>
</body>
</html>
```

```
<html lang="es">
<head>
<meta charset="utf-8"/>
<title>Página de prueba</title>
</head>
<body>
<div class="full-width" id="main">
<h1>El título de la página</h1>
<p>Este es el primer párrafo</p>
<p>Este es el segundo párrafo</p>
<div id="innerDiv">
<div class="links">
<a href="https://pagina1.xyz/">Enlace 1</a>
<a href="https://pagina2.xyz/">Enlace 2</a>
</div>
<div class="right">
<div class="links">
<a href="https://pagina3.xyz/">Enlace 3</a>
<a href="https://pagina4.xyz/">Enlace 4</a>
</div>
</div>
</div>
<div id="footer">
<!-- El footer -->
<p>Este párrafo está en el footer</p>
<div class="links footer-links">
<a href="https://pagina5.xyz/">Enlace 5</a>
</div>
</div>
</div>
</body>
</html>
```

Find y Findall

Beautiful Soup pone a nuestra disposición diferentes métodos para buscar elementos en el árbol. Sin embargo, dos de los principales son find_all() y find().

Ambos métodos buscan entre los descendientes de un objeto de tipo Tag y recuperan todos aquellos que cumplan una serie de filtros.

El filtro más básico consiste en pasar el nombre de la etiqueta a buscar como primer argumento de la función (parámetro name).

Si quieres recuperar todos los enlaces (etiqueta <a>) que hay en el texto HTML del ejemplo:

```
enlaces = soup.find_all('a')
for enlace in enlaces:
    print(enlace)
```

```
<a href="https://pagina1.xyz/">Enlace 1</a>
<a href="https://pagina2.xyz/">Enlace 2</a>
<a href="https://pagina3.xyz/">Enlace 3</a>
<a href="https://pagina4.xyz/">Enlace 4</a>
<a href="https://pagina5.xyz/">Enlace 5</a>
```

Además del nombre de la etiqueta, puedes especificar parámetros con nombre. Si estos no coinciden con los nombres de los parámetros de la función, serán tratados como atributos de la etiqueta entre los que filtrar.

Por ejemplo, si quisieras encontrar el bloque div con id="footer", podrías aplicar el siguiente filtro:

```
footer = soup.find_all(id='footer')
print(footer)
```

```
[<div id="footer">
<!-- El footer -->
<p>Este párrafo está en el footer</p>
<div class="links footer-links">
<a href="https://pagina5.xyz/">Enlace 5</a>
</div>
</div>]
```

Si queremos aplicar filtros a los Tags por sus atributos tendremos que pasarle un diccionario, por ejemplo, si queremos quedarnos sólo con los Tags "div" con su atributo "class" igual a "right":

```
for elemento in soup.find_all("div", attrs = {"class":"right"}):
    print(elemento.prettify())
```

```
<div class="right">
 <div class="links">
  <a href="https://pagina3.xyz/">
   Enlace 3
  </a>
  <a href="https://pagina4.xyz/">
   Enlace 4
  </a>
 </div>
</div>
```

Si sabemos que aparece una única vez, podemos usar el método find, que directamente nos muestra la primera aparición. Es equivalente a fijar limit=1en el método find_all.

```
soup.find('title')
```

```
<title>Página de prueba</title>
```

```
soup.find_all('div', limit=1)

[<div class="full-width" id="main">
 <h1>El título de la página</h1>
 <p>Este es el primer párrafo</p>
 <p>Este es el segundo párrafo</p>
 <div id="innerDiv">
 <div class="links">
 <a href="https://pagina1.xyz/">Enlace 1</a>
 <a href="https://pagina2.xyz/">Enlace 2</a>
 </div>
 <div class="right">
 <div class="links">
 <a href="https://pagina3.xyz/">Enlace 3</a>
 <a href="https://pagina4.xyz/">Enlace 4</a>
 </div>
 </div>
 </div>
 <div id="footer">
 <!-- El footer -->
 <p>Este párrafo está en el footer</p>
 <div class="links footer-links">
 <a href="https://pagina5.xyz/">Enlace 5</a>
 </div>
 </div>
 </div>]
```

Y con esto terminamos nuestro paseo por BeautifulSoap antes de hacer un ejemplo de cómo scrapear una página.