Scrapy

Mais um pouco sobre Selector

XPath e CSS

Scrapy

Mais um pouco sobre Selector XPath e CSS

Na seção sobre LXML e também nesta seção já vimos como utilizar XPath. Vimos também nesta seção como utilizar Selector do Scrapy, mas vamos fazer mais uma aula sobre estes assuntos com mais exemplos.

Scrapy

Mais um pouco sobre Selector XPath e CSS

Este é o endereço utilizado nesta aula:

https://pythonparatodos.com.br/formulario.html

O mesmo utilizado na aula sobre LXML e XPath.


```html
<html>
    <head>
        <title>Aula Python Web Scraping</title>
    </head>
    <body>
        <h1>Curso Python Web Scraping</h1>

        <form action="formulario_pythonwebscraping1.php" method="POST">
            Informe os dados abaixo:<br><br>
            <table>
                <tr>
                    <td>Nome:</td>
                    <td><input type="text" name="nome" size="50" maxlength="100"></input></td>
                </tr>
                <tr>
                    <td>E-mail:</td>
                    <td><input type="text" name="email" size="50" maxlength="100"></input></td>
                </tr>
                <tr>
                    <td>Celular:</td>
                    <td><input type="text" name="celular" size="20" maxlength="14"></input></td>
                </tr>
                <tr>
                    <td><input type="submit" name="enviar" value="Enviar dados"></td>
                </tr>
            </table>
        </form>
    </body>
</html>
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Estamos utilizando XPath com o Selector do Scrapy.

Estamos buscando todas tags input cujo atributo type seja igual a “text”.

```
from scrapy import Selector
from urllib.request import urlopen



html = urlopen("https://www.pythonparatodos.com.br/formulario.html")
sel = Selector(text = html.read())
lista = sel.xpath('//input[@type="text"]')
print(lista)
for selector in lista:
    print(selector)
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Aqui estamos utilizando o extract() para extrair apenas os dados.

```
from scrapy import Selector
from urllib.request import urlopen



html = urlopen("https://www.pythonparatodos.com.br/formulario.html")
sel = Selector(text = html.read())
lista = sel.xpath('//input[@type="text"]')
print(lista.extract())
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Aqui estamos utilizando o extract_first() para extrair apenas os dados do primeiro elemento.

```
from scrapy import Selector
from urllib.request import urlopen



html = urlopen("https://www.pythonparatodos.com.br/formulario.html")
sel = Selector(text = html.read())
lista = sel.xpath('//input[@type="text"]')
print(lista.extract_first())
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Aqui estamos montando uma lista de elementos input e estamos recuperando o quarto elemento da lista.

```
from scrapy import Selector
from urllib.request import urlopen


html = urlopen("https://www.pythonparatodos.com.br/formulario.html")
sel = Selector(text = html.read())
lista = sel.xpath('//input’)
quarto_input = lista[3]
print(quarto_input.extract())
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Agora vamos ver alguns exemplos utilizando CSS de Selector.

Comparando o CSS com o XPath:

- A “/” é substituída por “>” (exceto quando for o primeiro caracter)

  XPath: “/html/body/div”

  CSS Locator: “html > body > div”

- As duas barras “//” são substituídas por um espaço (exceto quando for o primeiro caracter)

  XPath: “//div/span//p”

  CSS Locator: “div > span p”

- O [N] é substituído por :nth-of-type(N)

  XPath: “//div/p[2]”

  CSS Locator: “div > p:nth-of-type(2)”

Scrapy

Mais um pouco sobre Selector XPath e CSS

Veja como selecionar todos objetos tr da tabela usando CSS.

```
from scrapy import Selector
from urllib.request import urlopen


html = urlopen("https://www.pythonparatodos.com.br/formulario.html")
sel = Selector(text = html.read())
lista = sel.css('html > body > form > table > tr')
print(lista)
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Vamos agora selecionar o segundo objeto tr (e-mail).

```
from scrapy import Selector
from urllib.request import urlopen



html = urlopen("https://www.pythonparatodos.com.br/formulario.html")
sel = Selector(text = html.read())
lista = sel.css('tr:nth-of-type(2)')
for x in lista:
    print(x)
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Para selecionar com base na classe do objeto podemos utilizar o caracter ponto “.”.

Como no formulário do site não tem nenhuma classe específica, nesse exemplo, vamos colocar o HTML modificado em uma string.

E vamos definir uma classe para o input do nome.

```
"""
<html>
...         <tr>
            <td>Nome:</td>
            <td><input class="teste" type="text" name="nome" size="50" maxlength="100"></input></td>
          </tr>
          <tr>
            <td>E-mail:</td>
            <td><input type="text" name="email" size="50" maxlength="100"></input></td>
...
</html>
"""
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Cortei o HTML aqui do slide devido à limitação de tamanho. Veja a utilização de input.teste que retorna o objeto da tag input cuja classe é “teste”.

```
from scrapy import Selector
from urllib.request import urlopen

html = """
<html>
  ...
          <td>Nome:</td>
          <td><input class="teste" type="text" name="nome" size="50" maxlength="100"></input></td>
        </tr>
  ...
</html>
"""

sel = Selector(text = html)
lista = sel.css('input.teste')
for x in lista:
   print(x)
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Para selecionar um elemento com base no ID, utilizamos “#id”.

Para este exemplo coloquei um id em cada input.

```
<tr>
  <td>Nome:</td>
  <td><input id="nome" type="text" name="nome" size="50" maxlength="100"></input></td>
</tr>
<tr>
  <td>E-mail:</td>
  <td><input id="email" type="text" name="email" size="50" maxlength="100"></input></td>
</tr>
<tr>
  <td>Celular:</td>
  <td><input id="celular" type="text" name="celular" size="20" maxlength="14"></input></td>
</tr>
<tr>
  <td><input id="enviar" type="submit" name="enviar" value="Enviar dados"></td>
</tr>
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Veja a utilização de input#celular para retornar o objeto input cujo id seja “celular”.

```
from scrapy import Selector
from urllib.request import urlopen

html = """
<html>
  ...
</html>
"""

sel = Selector(text=html)
lista = sel.css('input#celular')
for x in lista:
   print(x)
```

Scrapy

Mais um pouco sobre Selector XPath e CSS

Veja agora como selecionar dados informando um atributo e valor.

Nesse exemplo, vamos retornar todos os “ids” de inputs que tenham o atributo id (para esse exemplo, deixei somente com id o input do e-mail).

```
from scrapy import Selector
from urllib.request import urlopen

html = """
<html>
...
          <td>Nome:</td>
          <td><input type="text" name="nome" size="50" maxlength="100"></input></td>
        </tr>
        <tr>
          <td>E-mail:</td>
          <td><input id="email" type="text" name="email" size="50" maxlength="100"></input></td>
        </tr>
        <tr>
          <td>Celular:</td>
          <td><input type="text" name="celular" size="20" maxlength="14"></input></td>
...
</html>
"""

sel = Selector(text=html)
lista = sel.css('input::attr(id)')
for x in lista:
   print(x)
```

FIM