

Методы сбора и обработки данных при помощи Python

Урок 6



Парсинг HTML

Краткий обзор технологий для понимания сбора и обработки данных

План урока

- 1) Структура страницы HTML - DOM
- 2) Язык запросов Xpath
- 3) Парсинг HTML в Python – библиотека lxml



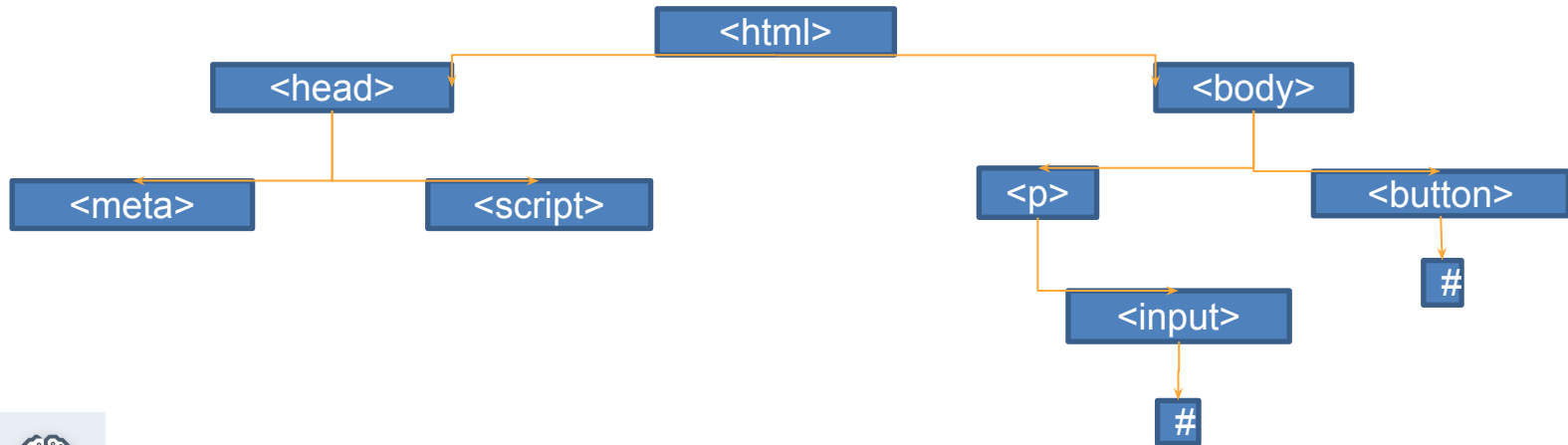
Структура страницы HTML



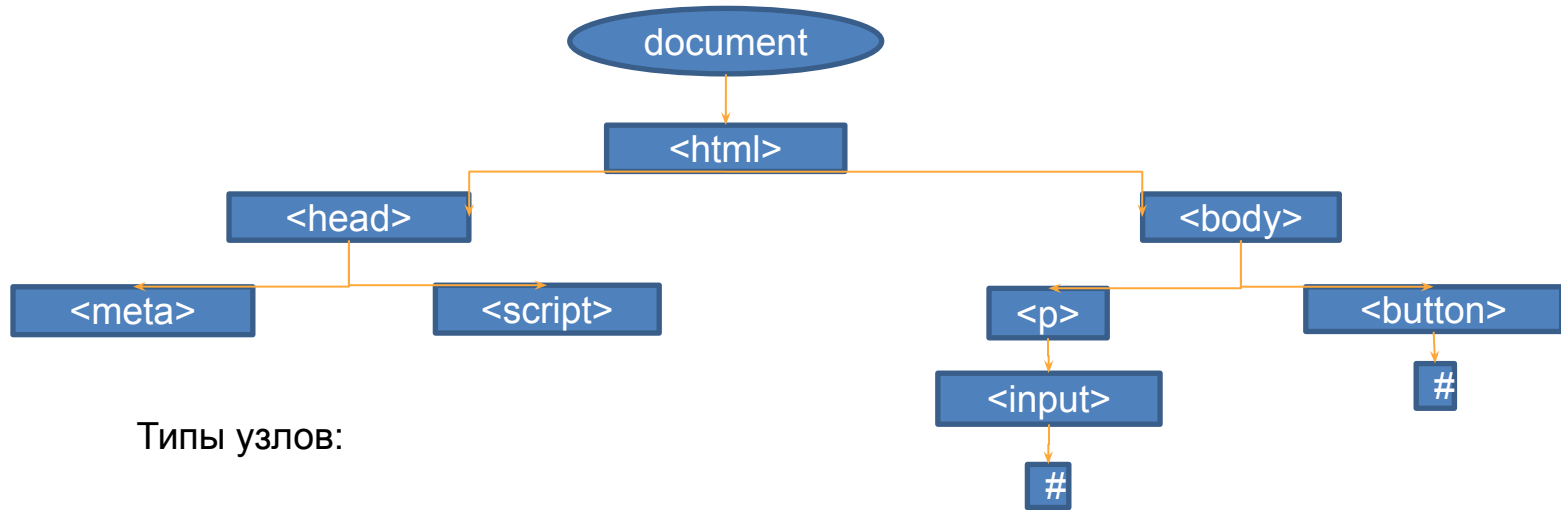
Структура страницы HTML

DOM

(от англ. *Document Object Model* — «объектная модель документа») — программный интерфейс, позволяющий программам и скриптам получить доступ к содержимому HTML-документов, а также изменять их содержимое, структуру и оформление таких документов.



Структура страницы HTML DOM, как дерево тэгов



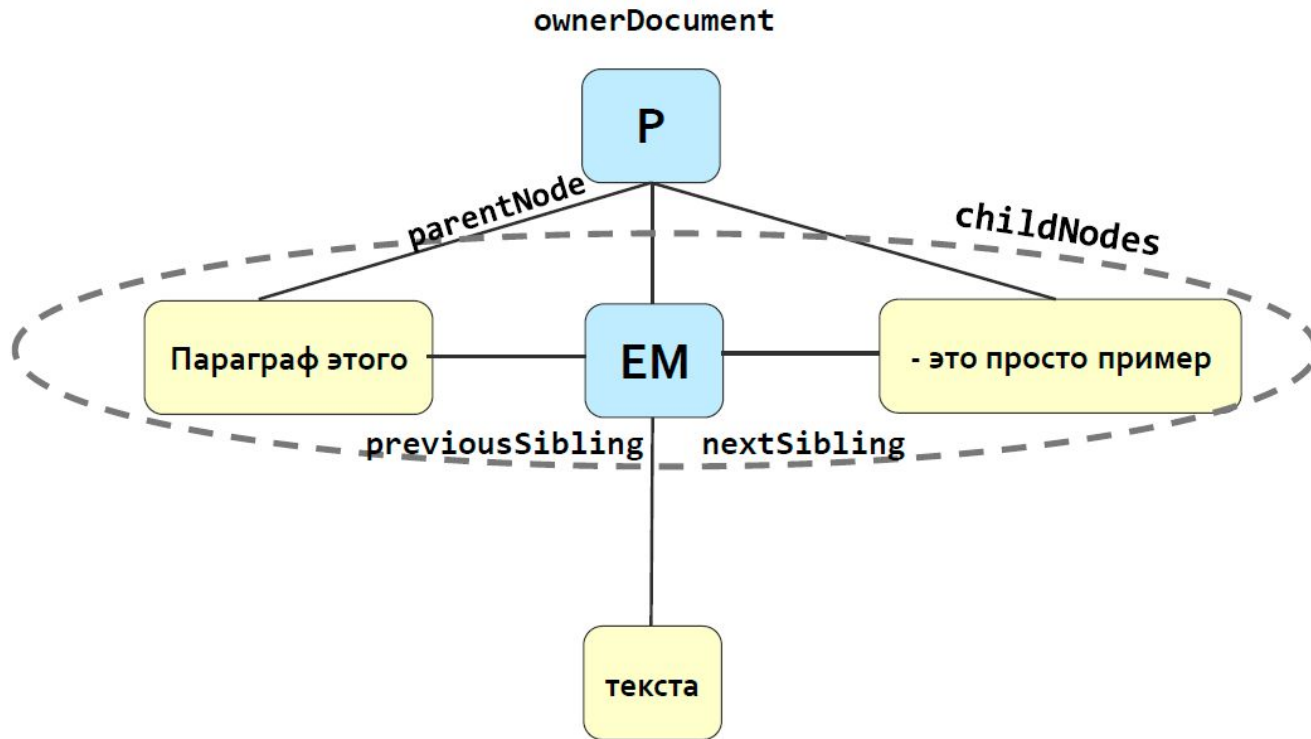
Типы узлов:

- документ
- элементы
- текстовые узлы
- комментарии



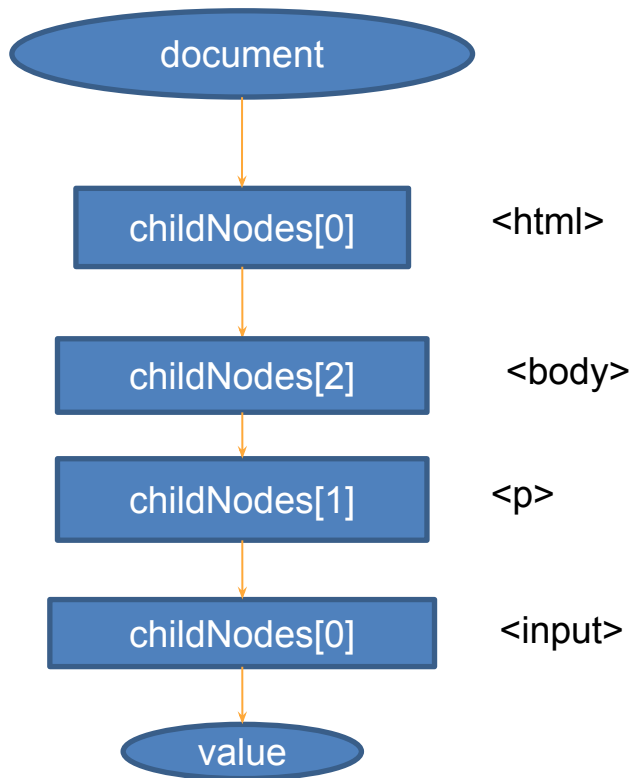
Связи между объектами

`<p>Параграф этого текста - это просто пример</p>`



Навигация по DOM

```
document.childNodes[0].childNodes[2].childNodes[1].childNodes[0].value
```



Язык запросов XPath

XPath (XML Path Language) — язык запросов к элементам XML-документа. Разработан для организации доступа к частям документа XML в файлах трансформации XSLT и является стандартом консорциума W3C. XPath призван реализовать навигацию по DOM в XML. В XPath используется компактный синтаксис, отличный от принятого в XML.

```
1 <html>
2 <body>
3   <div>Первый слой
4     <span>блок текста в первом слое</span>
5   </div>
6   <div>Второй слой</div>
7   <div>Третий слой
8     <span class="text">первый блок в третьем слое</span>
9     <span class="text">второй блок в третьем слое</span>
10    <span>третий блок в третьем слое</span>
11  </div>
12  <span>четвёртый слой</span>
13  <img />
14 </body>
15 </html>
```

XPath-путь `/html/body/*/span[@class]` будет соответствовать в нём двум элементам исходного документа — `первый блок в третьем слое` и `второй блок в третьем слое`



Язык запросов XPath

Выражение XPath	Результат
messages	Выбирает все узлы с именем "messages"
/messages	Выбирает корневой элемент сообщений Примечание: Если путь начинается с косой черты (/), то он всегда представляет абсолютный путь к элементу!
messages/note	Выбирает все элементы note, являющиеся потомками элемента messages
//note	Выбирает все элементы note независимо от того, где в документе они находятся
messages//note	Выбирает все элементы note, являющиеся потомками элемента messages независимо от того, где они находятся от элемента messages
//@date	Выбирает все атрибуты с именем date



Язык запросов XPath

Выражение XPath	Результат
<code>/messages/note[1]</code>	Выбирает первый элемент <code>note</code> , который является прямым потомком элемента <code>messages</code> . Примечание: В IE 5,6,7,8,9 первым узлом будет <code>[0]</code> , однако согласно W3C это должен быть <code>[1]</code> . Чтобы решить эту проблему в IE, нужно установить опцию <code>SelectionLanguage</code> в значение <code>XPath</code> . В JavaScript: <code>xml.setProperty("SelectionLanguage","XPath");</code>
<code>/messages/note[last()]</code>	Выбирает последний элемент <code>note</code> , который является прямым потомком элемента <code>messages</code> .
<code>/messages/note[last()-1]</code>	Выбирает предпоследний элемент <code>note</code> , который является прямым потомком элемента <code>messages</code> .
<code>/messages/note[position()]</code>	Выбирает первые два элемента <code>note</code> , которые являются прямыми потомками элемента <code>messages</code> .
<code>//heading[@date]</code>	Выбирает все элементы <code>heading</code> , у которых есть атрибут <code>date</code>
<code>//heading[@date="10/01/2008"]</code>	Выбирает все элементы <code>heading</code> , у которых есть атрибут <code>date</code> со значением "10/01/2008"



Язык запросов XPath

Выбор неизвестных заранее узлов

Чтобы найти неизвестные заранее узлы XML документа, XPath позволяет использовать специальные символы.

Спецсимвол	Описание
*	Соответствует любому узлу элемента
@*	Соответствует любому узлу атрибута
node()	Соответствует любому узлу любого типа

В следующей таблице приводятся некоторые выражения XPath со спецсимволами, позволяющие сделать выборки по демонстрационному XML документу:

Выражение XPath	Результат
/messages/*	Выбирает все элементы, которые являются прямыми потомками элемента messages
//*	Выбирает все элементы в документе
//heading[@*]	Выбирает все элементы heading, у которых есть по крайней мере один атрибут любого типа



Язык запросов XPath

Выбор нескольких путей

Использование оператора `|` в выражении XPath позволяет делать выбор по нескольким путям.

В следующей таблице приводятся некоторые выражения XPath, позволяющие сделать выборки по демонстрационному XML документу:

Выражение XPath	Результат
<code>//note/heading //note/body</code>	Выбирает все элементы <code>heading</code> И <code>body</code> из всех элементов <code>note</code>
<code>//heading //body</code>	Выбирает все элементы <code>heading</code> И <code>body</code> во всем документе



Язык запросов Xpath и python

```
from lxml import html

links = html.fromstring(html_page).xpath('//ul[28]//li/a/@href')
```



Домашнее задание

1) Необходимо собрать информацию о вакансиях на должность программиста или разработчика с сайта job.ru или hh.ru. (Можно с обоих сразу) Приложение должно анализировать несколько страниц сайта. Получившийся список должен содержать в себе:

- *Наименование вакансии,
- *Предлагаемую зарплату
- *Ссылку на самую вакансию

2) Доработать приложение таким образом, чтобы можно было искать разработчиков на разные языки программирования (Например Python, Java, C++)



Ваши вопросы?

