## **实验03（课程设计1）、HTML CSS Selector**

目的

1. 复习巩固线性表、栈等数据结构。
2. 练习使用树相关数据结构。
3. 熟悉和了解在实际问题中合理设计和使用数据结构。
4. 训练学生阅读和理解课外资料的能力。

问题描述

HTML CSS Selector是浏览器支撑Javascript脚本执行的核心和基础组建，也是众多互联网数据采集软件（如Scrapy）都支持的查询和数据抽取方式。只有熟悉和了解常用的数据结构，尤其是树，才能实现一个功能正确、方便易用的HTML CSS Selector。此次实验，希望学生能够一方面通过这个任务巩固数据结构知识，同时也基于数据结构开发未来自己能够使用的工具。

具体问题：

1. 通过树数据结构对HTML DOM层次结构进行建模。
2. 实现常用HTML CSS Selector以及对应的获取文本、HTML、链接等操作。
3. 开发测试或者演示程序，展示HTML CSS Selector的使用方法，验证正确性。

步骤

1. 读入HTML代码文件或给定一个网址，通过树数据结构存储该HTML页面内容。
2. 阅读理解CSS选择器的定义和功能，参考：
   1. 易懂入门资料：<https://www.w3school.com.cn/cssref/css_selectors.asp>
   2. CSS选择器国际标准：<https://www.w3.org/TR/selectors/>
   3. 其他工具如Scrapy中的CSS选择器：<https://www.educba.com/scrapy-css-selector/>；

<https://docs.scrapy.org/en/latest/topics/selectors.html>。

Beautiful Soup: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>等

* 1. 其他相关资料

1. 实现基本CSS选择器（请见下表），例如：
   1. \*,所有元素
   2. ID Selector，例如： #myId
   3. Class Selector，例如： .myClass
   4. Tag Selector，例如： DIV
   5. 空格选择器，例如 selectorA selectorB， DIV .myClass等。

更多选择器请见附表。仅要求实现其中标注为“是”的选择器

设计查询函数调用上述选择器，返回满足条件的elements/node的集合

1. 对于选择器返回的每个element/node，支持以下操作：
   1. 获取其innerText
   2. 获取该node的整个outerHTML
   3. 对于A标签，支持获取其包含的链接href的值
   4. 支持对返回的element再次调用选择器进行查询。例如，先通过一个选择器获取到一个新闻列表中的每一条新闻所在的节点，然后在每个节点上执行查询，分别抽取其中的标题、链接、日期等。
   5. 这些操作可以实现成函数，也可直接嵌入到选择器中，例如scrapy中的::text, ::attr(‘href’)
2. 开发测试或者演示程序，展示HTML CSS Selector的使用方法，验证正确性，演示方式和过程可参考Scrapy中Response.css的使用方法。
3. 完成实验的**实验报告**，报告的格式采用《数据结构题集》的模板格式。
4. 提供独立的你实现的CSS选择器的使用手册。
5. 提供独立的功能测试报告。报告要附充分的测试用例。
6. 提供完整的源代码、执行码以及生成执行码的项目工程文件。
7. 选作（10分）：在前面的基础上(满分90分)，考虑实现下面的功能之一。
   1. 实现XPATH选择器： <https://www.w3school.com.cn/xpath/xpath_syntax.asp>
   2. 实现5项以上附表表中标记为“否”的选择器

参考的交互形式（实现如下几个操作）：

**read(file\_name\_or\_url)**

从文件读入Html

**query(selector)**

根据selector从全文档检索所有符合条件的nodes，按顺序打印每个node的标签类型，id，class，同时将这些nodes保存，以供后续操作。

如：

> query ('.class1')

> [div.class0.class1, span.class1, img.class1]

**Out[k].innerText**

根据上次query操作得到的node list，找到第k个，输出他的innerText

**Out[k].outerHTML**

根据上次query操作得到的node list，找到第k个，输出他的outerHTML

**Out[k].query(selector)**

根据上次query操作得到的node list，找到第k个，再次调用选择器进行查询。按顺序打印每个node的标签类型，id，class，同时更新保存的nodes

**Out[k].href**

根据上次query操作得到的node list，找到第k个，输出他的href（仅对a标签）

测试query操作的正确性可以参考给的example.html

设备和环境

PC计算机、Windows操作系统、C/C++开发环境

附表：CSS选择器

|  |  |  |  |
| --- | --- | --- | --- |
| **需要实现** | **选择器** | **例子** | **例子描述** |
| 是 | [.*class*](https://www.w3school.com.cn/cssref/selector_class.asp) | .intro | 选择 class="intro" 的所有元素。 |
| 是 | .*class1*.*class2* | .name1.name2 | 选择 class 属性中同时有 name1 和 name2 的所有元素。 |
| 是 | .*class1* .*class2* | .name1 .name2 | 选择作为类名 name1 元素后代的所有类名 name2 元素。 |
| 是 | [#*id*](https://www.w3school.com.cn/cssref/selector_id.asp) | #firstname | 选择 id="firstname" 的元素。 |
| 是 | [\*](https://www.w3school.com.cn/cssref/selector_all.asp) | \* | 选择所有元素。 |
| 是 | [*element*](https://www.w3school.com.cn/cssref/selector_element.asp) | p | 选择所有 <p> 元素。 |
| 是 | [*element*.*class*](https://www.w3school.com.cn/cssref/selector_element_class.asp) | p.intro | 选择 class="intro" 的所有 <p> 元素。 |
| 是 | [*element*,*element*](https://www.w3school.com.cn/cssref/selector_element_comma.asp) | div, p | 选择所有 <div> 元素和所有 <p> 元素。 |
| 是 | [*element* *element*](https://www.w3school.com.cn/cssref/selector_element_element.asp) | div p | 选择 <div> 元素内的所有 <p> 元素。 |
| 是 | [*element*>*element*](https://www.w3school.com.cn/cssref/selector_element_gt.asp) | div > p | 选择父元素是 <div> 的所有 <p> 元素。 |
| 是 | [*element*+*element*](https://www.w3school.com.cn/cssref/selector_element_plus.asp) | div + p | 选择紧跟 <div> 元素的首个 <p> 元素。 |
| 是 | [*element1*~*element2*](https://www.w3school.com.cn/cssref/selector_gen_sibling.asp) | p ~ ul | 选择前面有 <p> 元素的每个 <ul> 元素。 |
|  |  |  |  |
| 否 | [[*attribute*]](https://www.w3school.com.cn/cssref/selector_attribute.asp) | [target] | 选择带有 target 属性的所有元素。 |
| 否 | [[*attribute*=*value*]](https://www.w3school.com.cn/cssref/selector_attribute_value.asp) | [target=\_blank] | 选择带有 target="\_blank" 属性的所有元素。 |
| 否 | [[*attribute*~=*value*]](https://www.w3school.com.cn/cssref/selector_attribute_value_contain.asp) | [title~=flower] | 选择 title 属性包含单词 "flower" 的所有元素。 |
| 否 | [[*attribute*|=*value*]](https://www.w3school.com.cn/cssref/selector_attribute_value_start.asp) | [lang|=en] | 选择 lang 属性值以 "en" 开头的所有元素。 |
| 否 | [[*attribute*^=*value*]](https://www.w3school.com.cn/cssref/selector_attr_begin.asp) | a[href^="https"] | 选择其 src 属性值以 "https" 开头的每个 <a> 元素。 |
| 否 | [[*attribute*$=*value*]](https://www.w3school.com.cn/cssref/selector_attr_end.asp) | a[href$=".pdf"] | 选择其 src 属性以 ".pdf" 结尾的所有 <a> 元素。 |
| 否 | [[*attribute*\*=*value*]](https://www.w3school.com.cn/cssref/selector_attr_contain.asp) | a[href\*="w3schools"] | 选择其 href 属性值中包含 "abc" 子串的每个 <a> 元素。 |
| 否 | [:active](https://www.w3school.com.cn/cssref/selector_active.asp) | a:active | 选择活动链接。 |
| 否 | [::after](https://www.w3school.com.cn/cssref/selector_after.asp) | p::after | 在每个 <p> 的内容之后插入内容。 |
| 否 | [::before](https://www.w3school.com.cn/cssref/selector_before.asp) | p::before | 在每个 <p> 的内容之前插入内容。 |
| 否 | [:checked](https://www.w3school.com.cn/cssref/selector_checked.asp) | input:checked | 选择每个被选中的 <input> 元素。 |
| 否 | [:default](https://www.w3school.com.cn/cssref/selector_default.asp) | input:default | 选择默认的 <input> 元素。 |
| 否 | [:disabled](https://www.w3school.com.cn/cssref/selector_disabled.asp) | input:disabled | 选择每个被禁用的 <input> 元素。 |
| 否 | [:empty](https://www.w3school.com.cn/cssref/selector_empty.asp) | p:empty | 选择没有子元素的每个 <p> 元素（包括文本节点）。 |
| 否 | [:enabled](https://www.w3school.com.cn/cssref/selector_enabled.asp) | input:enabled | 选择每个启用的 <input> 元素。 |
| 否 | [:first-child](https://www.w3school.com.cn/cssref/selector_first-child.asp) | p:first-child | 选择属于父元素的第一个子元素的每个 <p> 元素。 |
| 否 | [::first-letter](https://www.w3school.com.cn/cssref/selector_first-letter.asp) | p::first-letter | 选择每个 <p> 元素的首字母。 |
| 否 | [::first-line](https://www.w3school.com.cn/cssref/selector_first-line.asp) | p::first-line | 选择每个 <p> 元素的首行。 |
| 否 | [:first-of-type](https://www.w3school.com.cn/cssref/selector_first-of-type.asp) | p:first-of-type | 选择属于其父元素的首个 <p> 元素的每个 <p> 元素。 |
| 否 | [:focus](https://www.w3school.com.cn/cssref/selector_focus.asp) | input:focus | 选择获得焦点的 input 元素。 |
| 否 | [:fullscreen](https://www.w3school.com.cn/cssref/selector_fullscreen.asp) | :fullscreen | 选择处于全屏模式的元素。 |
| 否 | [:hover](https://www.w3school.com.cn/cssref/selector_hover.asp) | a:hover | 选择鼠标指针位于其上的链接。 |
| 否 | [:in-range](https://www.w3school.com.cn/cssref/selector_in-range.asp) | input:in-range | 选择其值在指定范围内的 input 元素。 |
| 否 | [:indeterminate](https://www.w3school.com.cn/cssref/selector_indeterminate.asp) | input:indeterminate | 选择处于不确定状态的 input 元素。 |
| 否 | [:invalid](https://www.w3school.com.cn/cssref/selector_invalid.asp) | input:invalid | 选择具有无效值的所有 input 元素。 |
| 否 | [:lang(*language*)](https://www.w3school.com.cn/cssref/selector_lang.asp) | p:lang(it) | 选择 lang 属性等于 "it"（意大利）的每个 <p> 元素。 |
| 否 | [:last-child](https://www.w3school.com.cn/cssref/selector_last-child.asp) | p:last-child | 选择属于其父元素最后一个子元素每个 <p> 元素。 |
| 否 | [:last-of-type](https://www.w3school.com.cn/cssref/selector_last-of-type.asp) | p:last-of-type | 选择属于其父元素的最后 <p> 元素的每个 <p> 元素。 |
| 否 | [:link](https://www.w3school.com.cn/cssref/selector_link.asp) | a:link | 选择所有未访问过的链接。 |
| 否 | [:not(*selector*)](https://www.w3school.com.cn/cssref/selector_not.asp) | :not(p) | 选择非 <p> 元素的每个元素。 |
| 否 | [:nth-child(*n*)](https://www.w3school.com.cn/cssref/selector_nth-child.asp) | p:nth-child(2) | 选择属于其父元素的第二个子元素的每个 <p> 元素。 |
| 否 | [:nth-last-child(*n*)](https://www.w3school.com.cn/cssref/selector_nth-last-child.asp) | p:nth-last-child(2) | 同上，从最后一个子元素开始计数。 |
| 否 | [:nth-of-type(*n*)](https://www.w3school.com.cn/cssref/selector_nth-of-type.asp) | p:nth-of-type(2) | 选择属于其父元素第二个 <p> 元素的每个 <p> 元素。 |
| 否 | [:nth-last-of-type(*n*)](https://www.w3school.com.cn/cssref/selector_nth-last-of-type.asp) | p:nth-last-of-type(2) | 同上，但是从最后一个子元素开始计数。 |
| 否 | [:only-of-type](https://www.w3school.com.cn/cssref/selector_only-of-type.asp) | p:only-of-type | 选择属于其父元素唯一的 <p> 元素的每个 <p> 元素。 |
| 否 | [:only-child](https://www.w3school.com.cn/cssref/selector_only-child.asp) | p:only-child | 选择属于其父元素的唯一子元素的每个 <p> 元素。 |
| 否 | [:optional](https://www.w3school.com.cn/cssref/selector_optional.asp) | input:optional | 选择不带 "required" 属性的 input 元素。 |
| 否 | [:out-of-range](https://www.w3school.com.cn/cssref/selector_out-of-range.asp) | input:out-of-range | 选择值超出指定范围的 input 元素。 |
| 否 | [::placeholder](https://www.w3school.com.cn/cssref/selector_placeholder.asp) | input::placeholder | 选择已规定 "placeholder" 属性的 input 元素。 |
| 否 | [:read-only](https://www.w3school.com.cn/cssref/selector_read-only.asp) | input:read-only | 选择已规定 "readonly" 属性的 input 元素。 |
| 否 | [:read-write](https://www.w3school.com.cn/cssref/selector_read-write.asp) | input:read-write | 选择未规定 "readonly" 属性的 input 元素。 |
| 否 | [:required](https://www.w3school.com.cn/cssref/selector_required.asp) | input:required | 选择已规定 "required" 属性的 input 元素。 |
| 否 | [:root](https://www.w3school.com.cn/cssref/selector_root.asp) | :root | 选择文档的根元素。 |
| 否 | [::selection](https://www.w3school.com.cn/cssref/selector_selection.asp) | ::selection | 选择用户已选取的元素部分。 |
| 否 | [:target](https://www.w3school.com.cn/cssref/selector_target.asp) | #news:target | 选择当前活动的 #news 元素。 |
| 否 | [:valid](https://www.w3school.com.cn/cssref/selector_valid.asp) | input:valid | 选择带有有效值的所有 input 元素。 |
| 否 | [:visited](https://www.w3school.com.cn/cssref/selector_visited.asp) | a:visited | 选择所有已访问的链接。 |