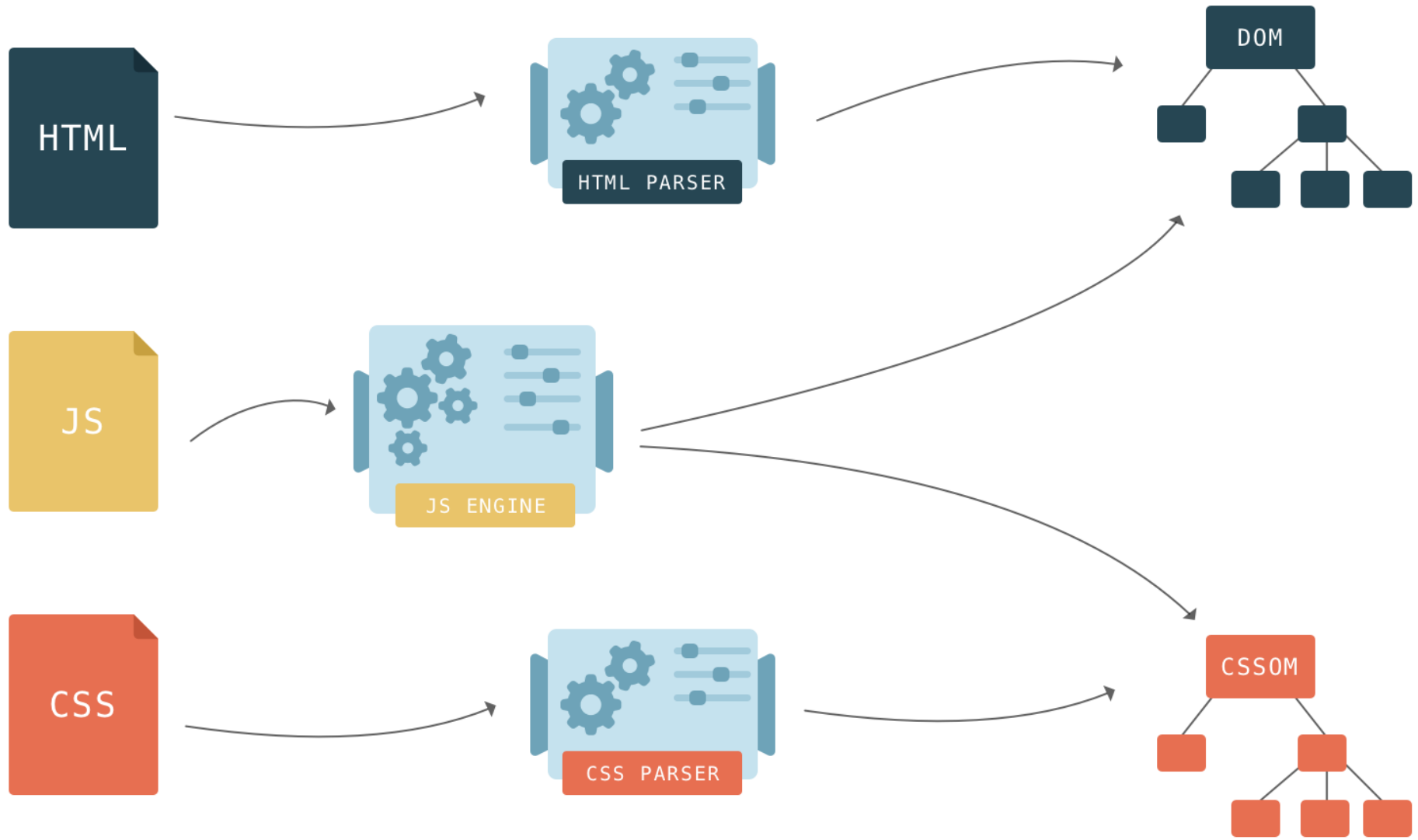
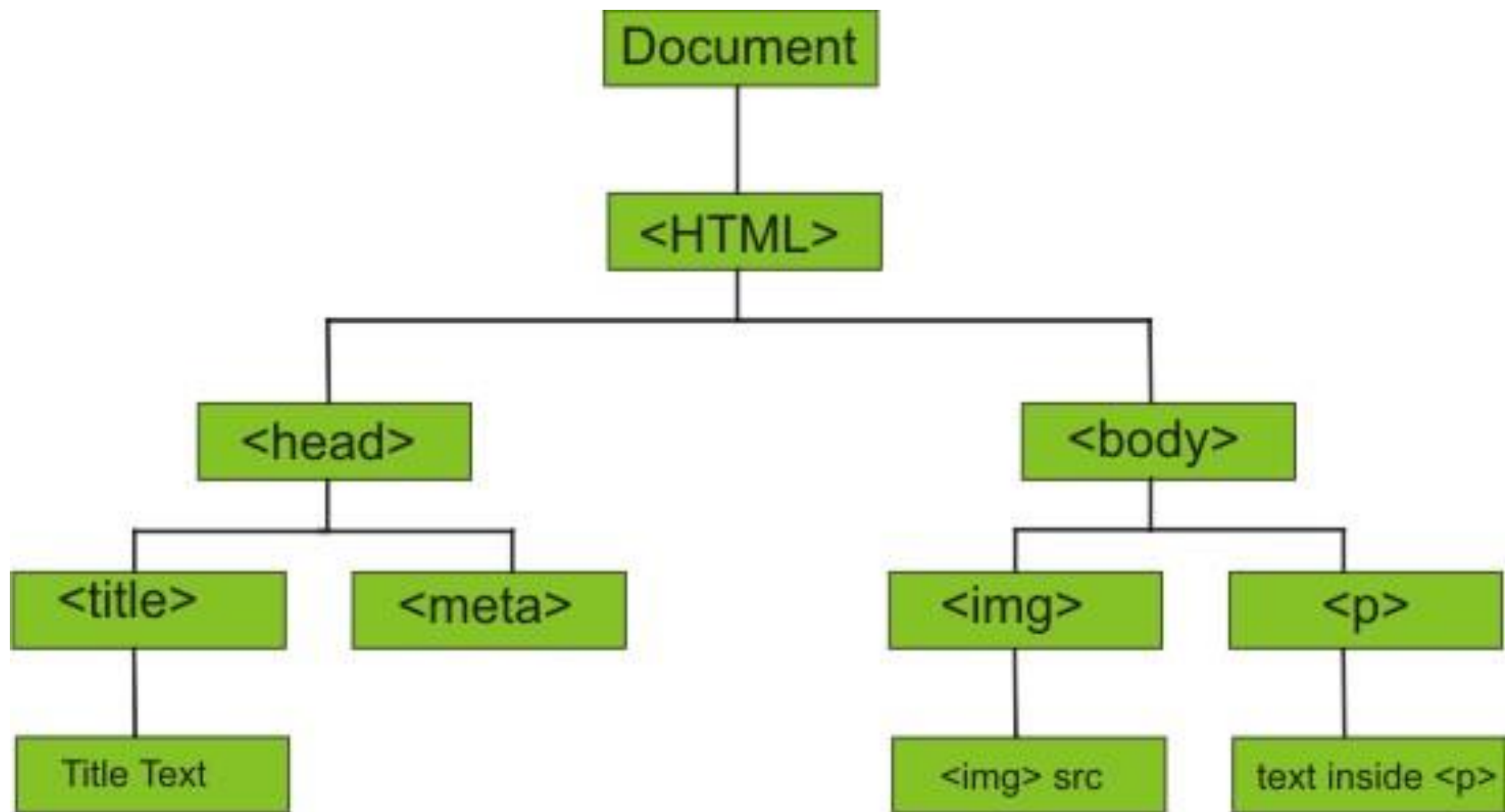


DOM





BeautifulSoup

# BeautifulSoup

## BeautifulSoup

Python library for pulling data out of HTML and XML files

Provide idiomatic ways of navigating, searching, and modifying DOM

Saves programmers hours or days of work

```
!pip install beautifulsoup4
```

```
from bs4 import BeautifulSoup
```

# Parser

**lxml** If you can, I recommend you install and use **lxml** for speed

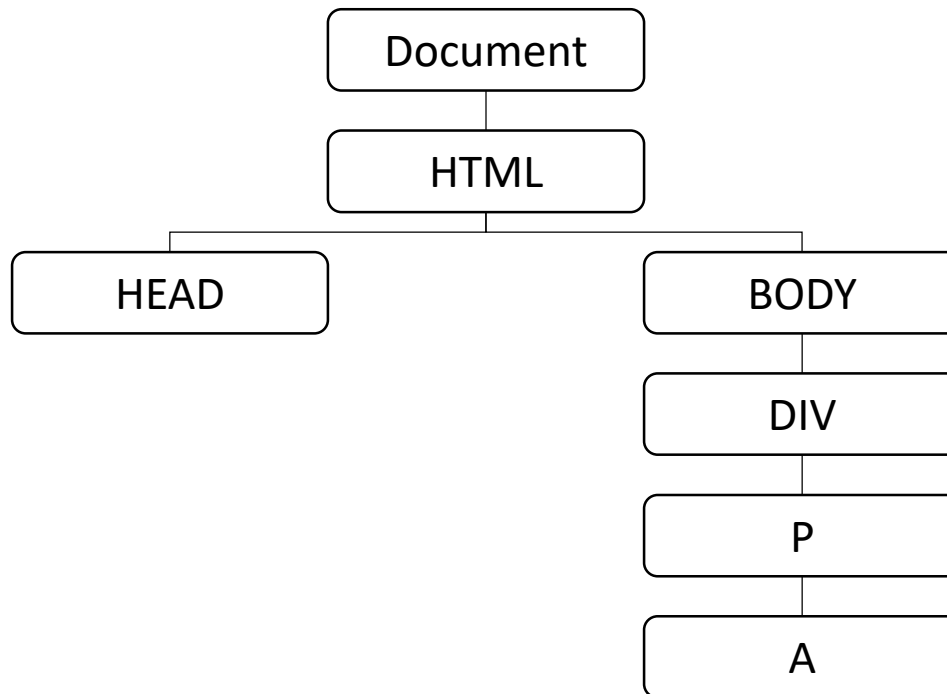
Parser	Typical usage	Advantages	Disadvantages
Python's html.parser	<code>BeautifulSoup(markup, "html.parser")</code>	<ul style="list-style-type: none"><li>• Batteries included</li><li>• Decent speed</li><li>• Lenient (as of Python 2.7.3 and 3.2.)</li></ul>	<ul style="list-style-type: none"><li>• Not very lenient (before Python 2.7.3 or 3.2.2)</li></ul>
lxml's HTML parser	<code>BeautifulSoup(markup, "lxml")</code>	<ul style="list-style-type: none"><li>• Very fast</li><li>• Lenient</li></ul>	<ul style="list-style-type: none"><li>• External C dependency</li></ul>
lxml's XML parser	<code>BeautifulSoup(markup, "lxml-xml")</code> <code>BeautifulSoup(markup, "xml")</code>	<ul style="list-style-type: none"><li>• Very fast</li><li>• The only currently supported XML parser</li></ul>	<ul style="list-style-type: none"><li>• External C dependency</li></ul>
html5lib	<code>BeautifulSoup(markup, "html5lib")</code>	<ul style="list-style-type: none"><li>• Extremely lenient</li><li>• Parses pages the same way a web browser does</li><li>• Creates valid HTML5</li></ul>	<ul style="list-style-type: none"><li>• Very slow</li><li>• External Python dependency</li></ul>

# Not Well-formed

- Every open tag must explicitly be **closed**
- Empty elements in XML are written **closed**
- Child markup must nest completely **within** parent markup

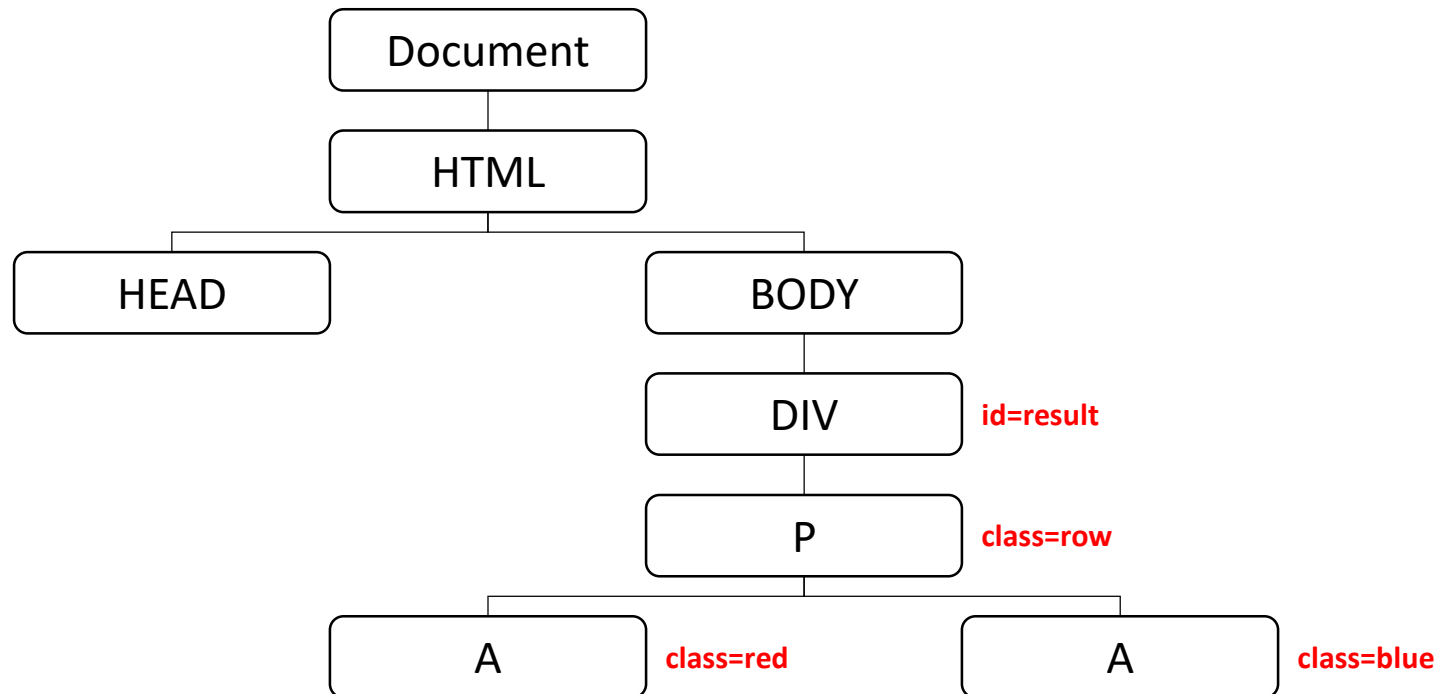
```
<html>
  <head><head>
  <body>
    <div>
      <ul>
        <li>
        <li> ← not closed / unclosed tag (meta, br, img, ...)
      </div>
    </ul> ← overlap with other elements
  </body>
</HTML> ← case mismatch
```

```
html = ""
<html>
  <head></head>
  <body>
    <div>
      <p>
        <a>go to page</a>
      </p>
    </div>
  </body>
</html>
""
```





```
html = """
<html>
  <head></head>
  <body>
    <div id="result">
      <p class="row">
        <a class="red">go to page1</a>
        <a class="blue">go to page2</a>
      </p>
    </div>
  </body>
</html>
"""
```



Scans the entire document looking for results

**find** (name, attrs, recursive, string, \*\*kwargs)

**find\_all** (name, attrs, recursive, string, limit, \*\*kwargs)

```
dom.find_all('a')
dom.find_all({'div', 'p'})
dom.find_all('div', {'id': 'result'})
dom.find_all("", attrs={'class': 'red'})
dom.find_all('a', recursive=False)
dom.find_all(text='go to page1')
dom.find_all('a', limit=1)
```

**find\_parent**            You can access an element's parent with the **.parent** attribute

**find\_parents**            You can iterate over all of an element's parents with **.parents**

**find\_all**                The **.descendants** attribute lets you iterate over all of a tag's children

**find\_all(recursive=false)**        The **.children** attributes only consider a tag's direct children

**find\_next\_sibling, find\_previous\_sibling**

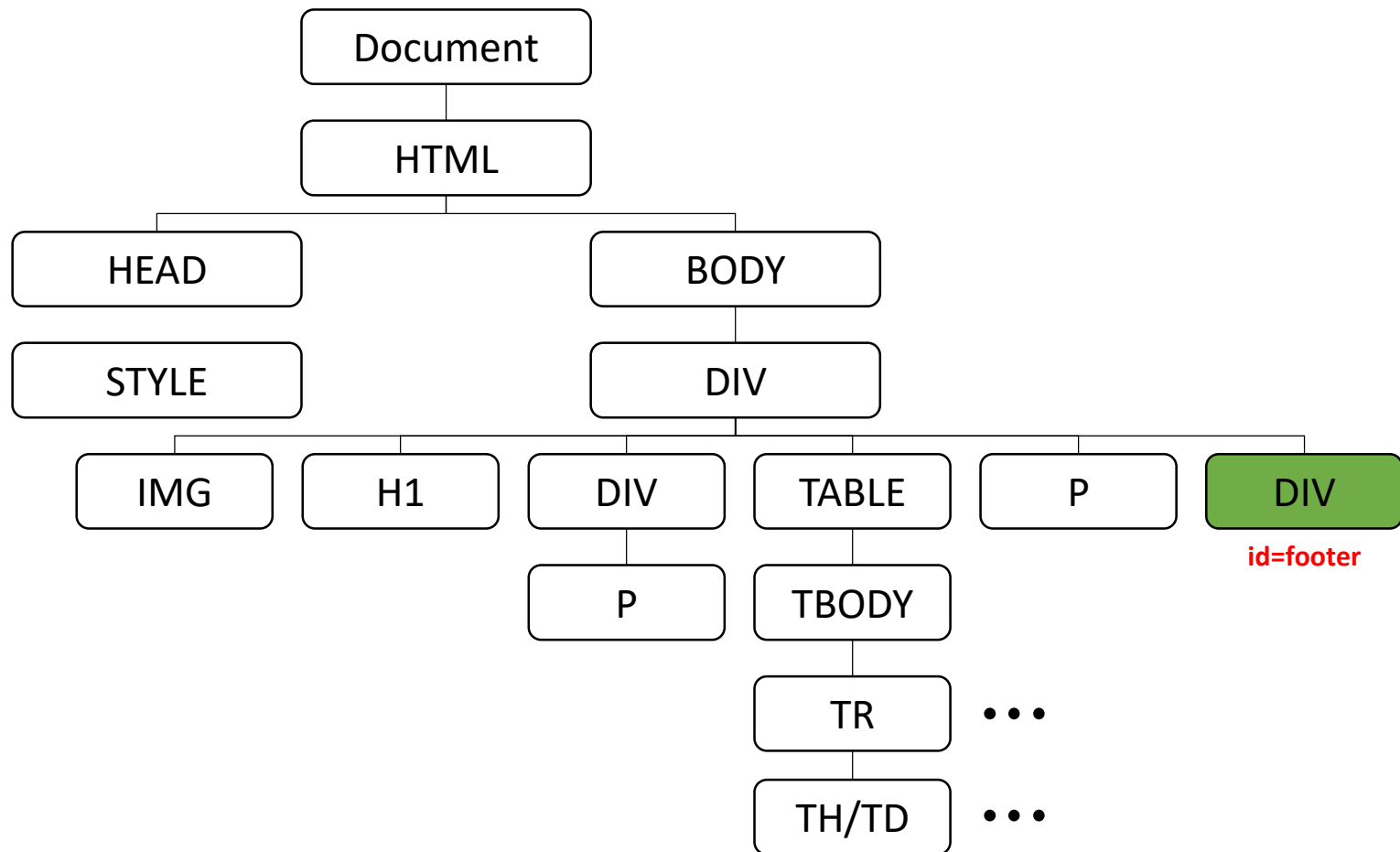
**find\_next\_siblings, find\_previous\_siblings**

Navigate between page elements

**on the same level** of the parse tree

# exercises

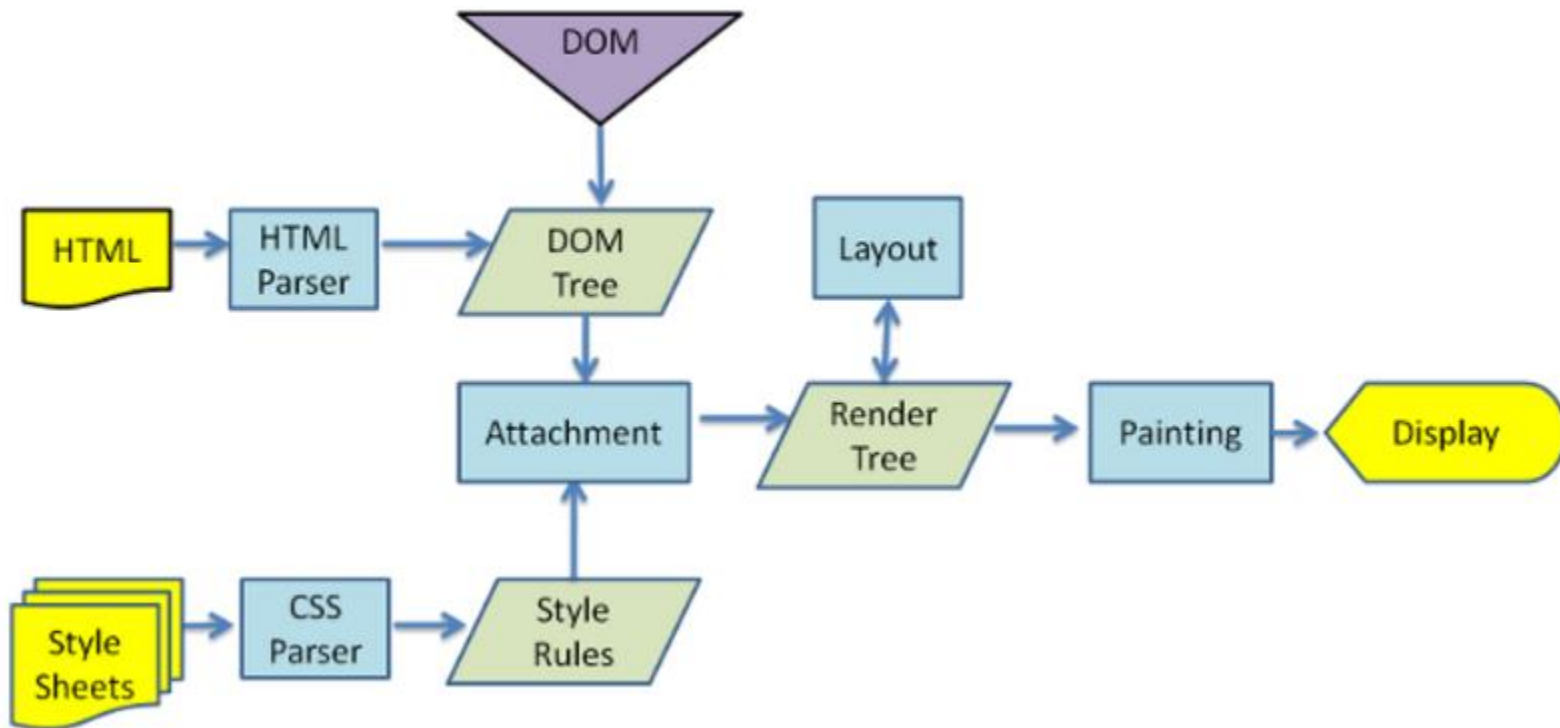
<http://pythonscraping.com/pages/page3.html>





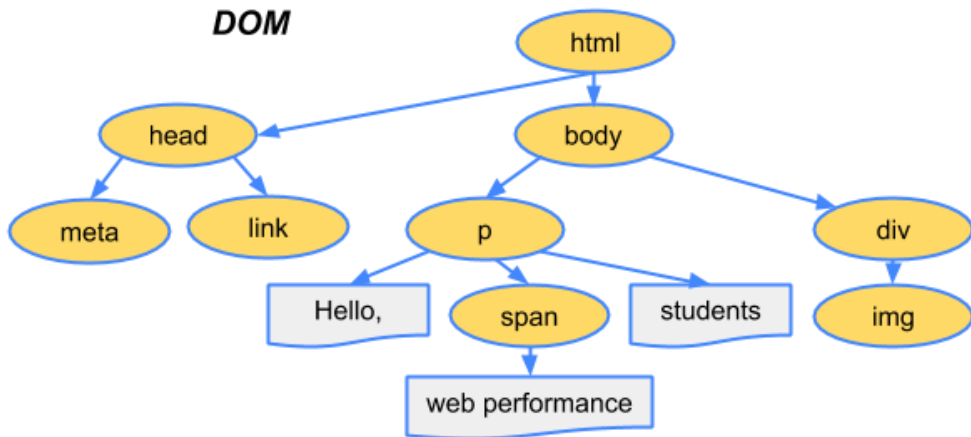
# Selector

# CSSOM

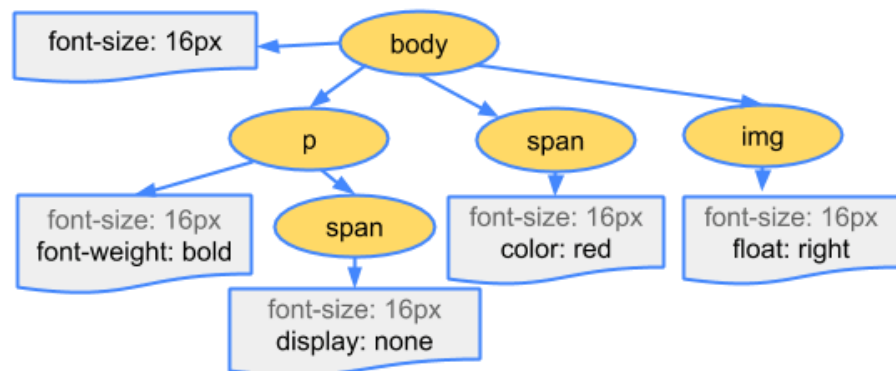




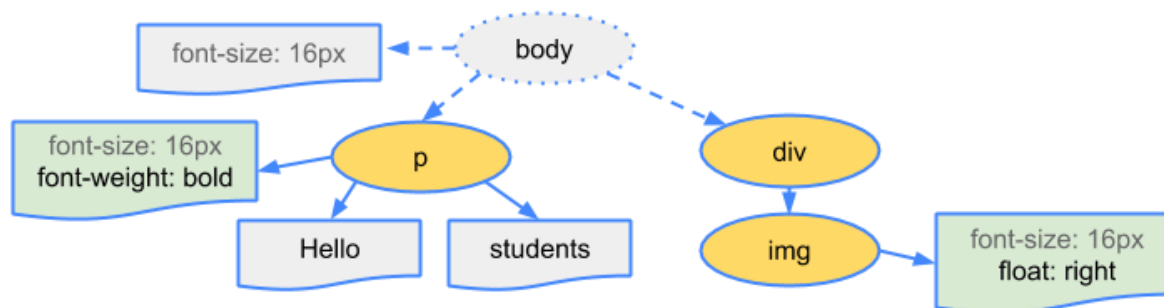
## DOM



## CSSOM



## Render Tree





# CSS Selector

Selector

Declaration

Declaration

**h1**

**{ color: red; font-size: 10 ; }**

↑  
Property

↑  
Value

↑  
Property

↑  
Value



# XPath

- XPath uses path expressions to select nodes in an XML document.  
The node is selected by following steps.

Expression	Description
<i>nodename</i>	Selects all nodes with the name " <i>nodename</i> "
/	Selects from the root node
//	Selects nodes in the document from the current node that match the selection no matter where they are
.	Selects the current node
..	Selects the parent of the current node
@	Selects attributes

exercises

**4대 검색사이트 검색결과 가져오기**