BeautifulSoup:

soup = BeautifulSoup(source\_code, parser)

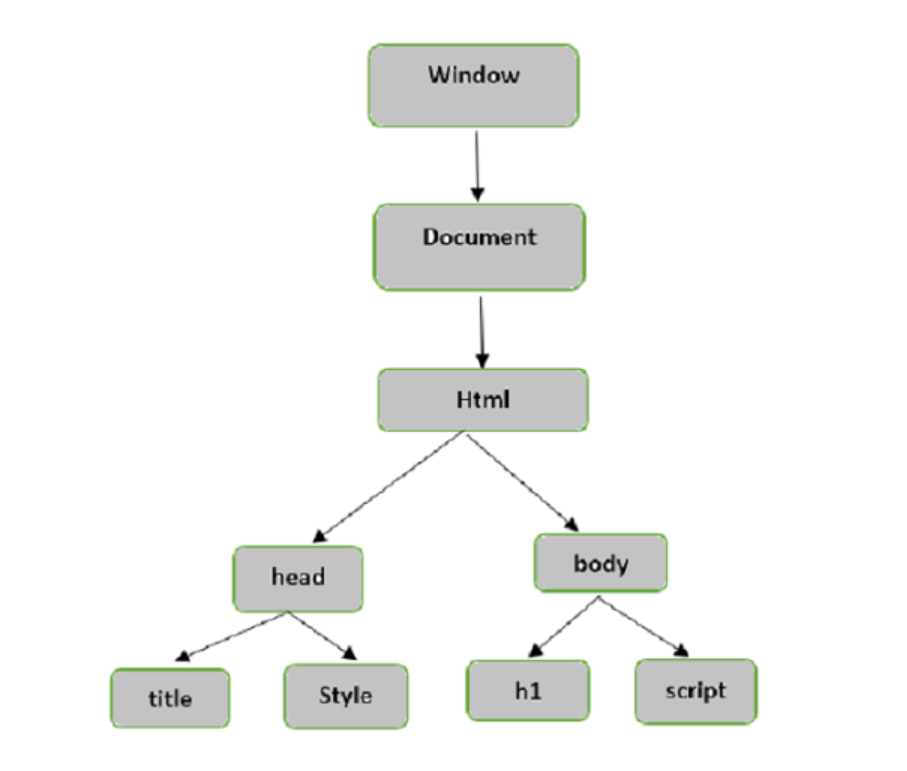
-here we have to pass a source code of web page , and a parser

-common parsers: lxml, urllib

With open(“x.html”) as fp:

soup= BeautifulSoup(fp, ‘lxml’)

HTML Dom structure:



When soup object is created:

-it has all the objects of html as Dom Structure

-To access the objects say for all the code of body we can use – soup.body

-to access the <h1> from <body> we can use: soup.body.h1

- likewise we can access all the tags of html

Each Tag properties:

-the properties of the tags can be accessed using :

soup.div[“class”] : it returns the “class” of particular div

* We can summarize as a tag is dictionary of all its properties:
* Ex- table:{ “class”:”form-control”, “name”:”table”,”id”:”id”}
* We can access children of a tag using ‘.’

Find() and findAll():

find(): returns only first occurrence of a tag with matching filters

find\_all : returns all the tags with specified filters

soup.find\_all(“div”,class=”table-wrap”)

**find\_next\_siblings()** and **find\_next\_sibling()** methods will iterate over all the siblings of the element that come after the current one.

**find\_previous\_siblings()** and **find\_previous\_sibling()** methods will iterate over all the siblings that come before the current element.

**find\_all\_next()** and **find\_next()** methods will iterate over all the tags and strings that come after the current element.

**find\_all\_previous** and **find\_previous()** methods will iterate over all the tags and strings that come before the current element.

Navigable Strings:

We can access contents of a tag using ‘.string’