Web Scraping with Python

```
In [1]: import requests
```

```
In [2]: import bs4
```

```
In [3]: result=requests.get("http://www.example.com")
```

```
In [5]: type(result)

Out[5]: requests.models.Response
```

```
In [6]: result.text

Out[6]: '<!doctype html>\n<html>\n<head>\n    <title>Example Domain</title>\n\n
        <meta charset="utf-8" />\n    <meta http-equiv="Content-type" content="t
        ext/html; charset=utf-8" />\n    <meta name="viewport" content="width=de
        vice-width, initial-scale=1" />\n    <style type="text/css">\n    body
        {\n        background-color: #f0f0f2;\n        margin: 0;\n        paddi
        ng: 0;\n        font-family: -apple-system, system-ui, BlinkMacSystemFon
        t, "Segoe UI", "Open Sans", "Helvetica Neue", Helvetica, Arial, sans-ser
        if;\n        \n    }\n    div {\n        width: 600px;\n        margin:
        5em auto;\n        padding: 2em;\n        background-color: #fdfdff;\n
        border-radius: 0.5em;\n        box-shadow: 2px 3px 7px 2px rgba(0,0,0,0.
        02);\n    }\n    a:link, a:visited {\n        color: #38488f;\n        t
        ext-decoration: none;\n    }\n    @media (max-width: 700px) {\n        d
        iv {\n            margin: 0 auto;\n            width: auto;\n        }\n
        }\n    </style>    \n</head>\n\n<body>\n<div>\n    <h1>Example Domain</h
        1>\n    <p>This domain is for use in illustrative examples in documents.
        You may use this\n    domain in literature without prior coordination or
        asking for permission.</p>\n    <p><a href="https://www.iana.org/domain
        s/example">More information...</a></p>\n</div>\n</body>\n</html>\n'
```

```
In [7]: import bs4
```

```
In [8]: soup= bs4.BeautifulSoup(result.text,'lxml')
```

```
In [9]: type(soup)

Out[9]: bs4.BeautifulSoup
```

Grabbing a Title

```
In [10]: soup.select("title") #to get specific text with the string code
```

```
Out[10]: [<title>Example Domain</title>]
```

```
In [13]: soup.select("p")[0].getText() #to only get the text
```

```
Out[13]: 'This domain is for use in illustrative examples in documents. You may u
         se this\n    domain in literature without prior coordination or asking f
         or permission.'
```

Grabbing a Class

```
In [17]: res=requests.get('https://en.wikipedia.org/wiki/Grace_Hopper')
```

```
In [18]: soup=bs4.BeautifulSoup(res.text,'lxml')
```

```
In [19]: type(soup)
```

```
Out[19]: bs4.BeautifulSoup
```

In [36]:

```
first= soup.select('.vector-toc-text')
first
```

Out[36]:

```
[<div class="vector-toc-text">(Top)</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">1</span>Early life and education</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">2</span>Career</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">2.1</span>World War II</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">2.2</span>UNIVAC</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">2.3</span>COBOL</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">2.4</span>Standards</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">3</span>Retirement</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">4</span>Post-retirement</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">5</span>Anecdotes</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">6</span>Death</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">7</span>Dates of rank</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">8</span>Awards and honors</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">8.1</span>Military awards</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">8.2</span>Other awards</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">9</span>Legacy</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">9.1</span>Places</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">9.2</span>Programs</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">9.3</span>In popular culture</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">9.3.1</span>Grace Hopper Celebration of W
omen in Computing</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">10</span>See also</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">11</span>Notes</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">12</span>References</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">13</span>Obituary notices</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">14</span>Further reading</div>,
 <div class="vector-toc-text">
 <span class="vector-toc-numb">15</span>External links</div>]
```

In [38]:

```python
for item in soup.select('.vector-toc-text'):
    print(item.text)
```

```
(Top)

1Early life and education

2Career

2.1World War II

2.2UNIVAC

2.3COBOL

2.4Standards

3Retirement

4Post-retirement

5Anecdotes

6Death

7Dates of rank

8Awards and honors

8.1Military awards

8.2Other awards

9Legacy

9.1Places

9.2Programs

9.3In popular culture

9.3.1Grace Hopper Celebration of Women in Computing

10See also

11Notes

12References

13Obituary notices

14Further reading

15External links
```

Grabbing an image

```
In [39]: res= requests.get('https://en.wikipedia.org/wiki/Grace_Hopper')
```

```
In [42]: soup= bs4.BeautifulSoup(res.text,'lxml')
```

```
In [50]: computer= soup.select('.mw-file-element')[0]
```

```
In [51]: type(computer)
```

```
Out[51]: bs4.element.Tag
```

```
In [53]: computer['src']
```

```
Out[53]: '//upload.wikimedia.org/wikipedia/commons/thumb/a/ad/Commodore_Grace_M._
Hopper%2C_USN_%28covered%29.jpg/240px-Commodore_Grace_M._Hopper%2C_USN_%
28covered%29.jpg'
```

```
In [54]: image_link= requests.get('https://upload.wikimedia.org/wikipedia/commons/t
```

In [55]:
```
image_link.content
```

Out[55]:
```
b'\xff\xd8\xff\xe1\x00rExif\x00\x00MM\x00*\x00\x00\x00\x08\x00\x05\x0
1\x1a\x00\x05\x00\x00\x00\x01\x00\x00\x00J\x01\x1b\x00\x05\x00\x00\x0
0\x01\x00\x00\x00R\x01(\x00\x03\x00\x00\x00\x01\x00\x02\x00\x00\x0
1;\x00\x02\x00\x00\x00\x0f\x00\x00\x00Z\x02\x13\x00\x03\x00\x00\x00\x
01\x00\x01\x00\x00\x00\x00\x00\x00\x00\x00\x00H\x00\x00\x00\x01\x00\x
00\x00H\x00\x00\x00\x01James S. Davis\x00\x00\xff\xe1\x0bwhttp://ns.a
dobe.com/xap/1.0/\x00<?xpacket begin=\'\xef\xbb\xbf\' id=\'W5M0MpCehi
HzreSzNTczkc9d\'?>\n<x:xmpmeta xmlns:x=\'adobe:ns:meta/\' x:xmptk=\'I
mage::ExifTool 9.74\'>\n<rdf:RDF xmlns:rdf=\'http://www.w3.org/1999/0
2/22-rdf-syntax-ns#\'>\n\n <rdf:Description rdf:about=\'\'\n  xmlns:d
c=\'http://purl.org/dc/elements/1.1/\'>\n  <dc:description>\n   <rdf:
Alt>\n    <rdf:li xml:lang=\'x-default\'>Commodore Grace M. Hopper, U
SN (covered).</rdf:li>\n   </rdf:Alt>\n  </dc:description>\n </rdf:De
scription>\n</rdf:RDF>\n</x:xmpmeta>\n
\n
\n
\n
\n
\n
```

In [56]:
```
f=open('computer_image.jpg','wb')
```

In [57]:
```
f.write(image_link.content)
```

Out[57]:
```
24355
```

In [58]:
```
f.close
```

Out[58]:
```
<function BufferedWriter.close>
```

In [59]:
```
pwd
```

Out[59]:
```
'C:\\Users\\siby9\\PYSPARK'
```

Book Examples

In [1]:
```
import requests
import bs4
```

In [2]:
```
#https://books.toscrape.com/catalogue/page-2.html
#https://books.toscrape.com/catalogue/page-5.html
base_url= 'https://books.toscrape.com/catalogue/page-{}.html' #to iterate
```

In [3]:
```
base_url.format(5)
```

Out[3]:
```
'https://books.toscrape.com/catalogue/page-5.html'
```

In [4]:

```python
base_url.format(10)
```

Out[4]:

```
'https://books.toscrape.com/catalogue/page-10.html'
```

In [5]:

```python
res=requests.get(base_url.format(1))
```

In [6]:

```python
soup= bs4.BeautifulSoup(res.text,'lxml')
```

In [21]:

```python
products=soup.select('.product_pod')
```

In [36]:

```python
products
```

```
 <button class="btn btn-primary btn-block" data-loading-text="Addin
g..." type="submit">Add to basket</button>
 </form>
 </div>
 </article>,
 <article class="product_pod">
 <div class="image_container">
 <a href="scott-pilgrims-precious-little-life-scott-pilgrim-1_987/ind
ex.html"><img alt="Scott Pilgrim's Precious Little Life (Scott Pilgri
m #1)" class="thumbnail" src="../media/cache/94/b1/94b1b8b244bce9677c
2f29ccc890d4d2.jpg"/></a>
 </div>
 <p class="star-rating Five">
 <i class="icon-star"></i>
 <i class="icon-star"></i>
 <i class="icon-star"></i>
 <i class="icon-star"></i>
 <i class="icon-star"></i>
 </p>
 <h3><a href="scott-pilgrims-precious-little-life-scott-pilgrim-1_98
```

In [22]:

```python
example=products[0]
```

In [23]:

```
example
```

Out[23]:

```
<article class="product_pod">
<div class="image_container">
<a href="a-light-in-the-attic_1000/index.html"><img alt="A Light in the
Attic" class="thumbnail" src="../media/cache/2c/da/2cdad67c44b002e7ead0c
c35693c0e8b.jpg"/></a>
</div>
<p class="star-rating Three">
<i class="icon-star"></i>
<i class="icon-star"></i>
<i class="icon-star"></i>
<i class="icon-star"></i>
<i class="icon-star"></i>
</p>
<h3><a href="a-light-in-the-attic_1000/index.html" title="A Light in the
Attic">A Light in the ...</a></h3>
<div class="product_price">
<p class="price_color">Â£51.77</p>
<p class="instock availability">
<i class="icon-ok"></i>

        In stock

</p>
<form>
<button class="btn btn-primary btn-block" data-loading-text="Adding..."
type="submit">Add to basket</button>
</form>
</div>
</article>
```

In [24]:

```
example.select('.star-rating.Three') # to derive the rating
```

Out[24]:

```
[<p class="star-rating Three">
 <i class="icon-star"></i>
 <i class="icon-star"></i>
 <i class="icon-star"></i>
 <i class="icon-star"></i>
 <i class="icon-star"></i>
 </p>]
```

In [25]:

```
example.select('a') #to derive the title name
```

Out[25]:

```
[<a href="a-light-in-the-attic_1000/index.html"><img alt="A Light in the
Attic" class="thumbnail" src="../media/cache/2c/da/2cdad67c44b002e7ead0c
c35693c0e8b.jpg"/></a>,
 <a href="a-light-in-the-attic_1000/index.html" title="A Light in the At
tic">A Light in the ...</a>]
```

In [26]:

```
example.select('a')[1]['title']
```

Out[26]:

```
'A Light in the Attic'
```

In [27]:
```
example.select('.price_color')
```

Out[27]:
```
[<p class="price_color">Â£51.77</p>]
```

In [30]:
```
stock=example.select('.instock.availability')[0] #to get stock availabilit
```

In [31]:
```
stock.get_text()
```

Out[31]:
```
'\n\n    \n        In stock\n    \n'
```

In [35]:
```
example.select('.price_color')
```

Out[35]:
```
[<p class="price_color">Â£51.77</p>]
```

1. To grab the 5 star rating books

2. with their title name

In [42]:
```
two_star_titles=[]
for n in range(1,51):
    scrape_url=base_url.format(n)
    res=requests.get(scrape_url)

    soup=bs4.BeautifulSoup(res.text,'lxml')
    books=soup.select('.product_pod')
    for book in books:
        if len(book.select('.star-rating.Five'))!=0:
            book_title=book.select('a')[1]['title']
            two_star_titles.append(book_title)
```

In [43]:

```
two_star_titles
```

```
 'Agnostic: A Spirited Manifesto',
 'You (You #1)',
 "Walt Disney's Alice in Wonderland",
 "The White Queen (The Cousins' War #1)",
 'The Time Keeper',
 'The Star-Touched Queen',
 'The Songs of the Gods',
 'The Song of Achilles',
 'The Darkest Lie',
 'Superman Vol. 1: Before Truth (Superman by Gene Luen Yang #1)',
 'Steve Jobs',
 'Someone Like You (The Harrisons #2)',
 'Quarter Life Poetry: Poems for the Young, Broke and Hangry',
 'Old School (Diary of a Wimpy Kid #10)',
 'Made to Stick: Why Some Ideas Survive and Others Die',
 'Looking for Lovely: Collecting the Moments that Matter',
 'Let It Out: A Journey Through Journaling',
 'Lady Midnight (The Dark Artifices #1)',
 'Hyperbole and a Half: Unfortunate Situations, Flawed Coping Mechani
sms, Mayhem, and Other Things That Happened',
```

In [ ]:

```
```