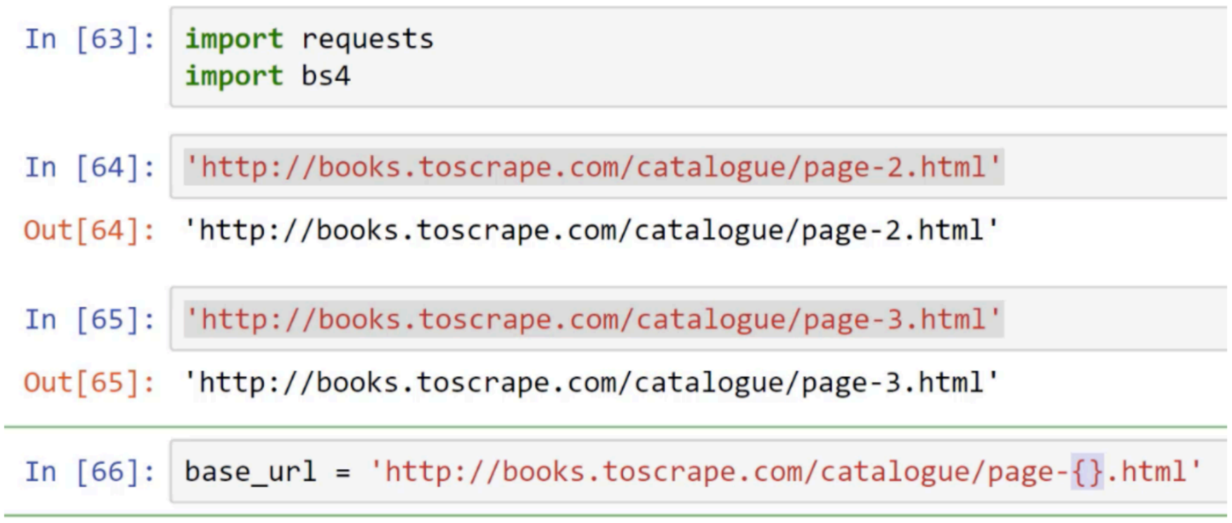
SECTION 15: WEB SCRAPING WITH PYTHON - 1 hour 40 minutes, 9 parts
6/9 Python Web Scraping - Book Examples Part One

- **working with multiple pages and items**
    - -> this is using the same notebook as the previous video
    - **-> we want to grab multiple elements off a webpage**
    - **-> web scraping libraries**
    - -> toscrape <- this is a website designed for practicing web scraping
    - -> scraping hub
    - -> the point of using this website is that no permissions are required

    - **-> bookstoscrape**
        - -> this is a fake bookstore website designed for practicing web scraping
        - -> it looks like Amazon but for books
        - -> there is information about books on there
        - -> we are scraping the titles of the books on there with two star ratings
        - -> importing requests and bs4 for this
        - -> there are 50 pages of books on the webpage and each has 20 books on it
        - **-> looping through all of the pages of books and scraping the books on each**

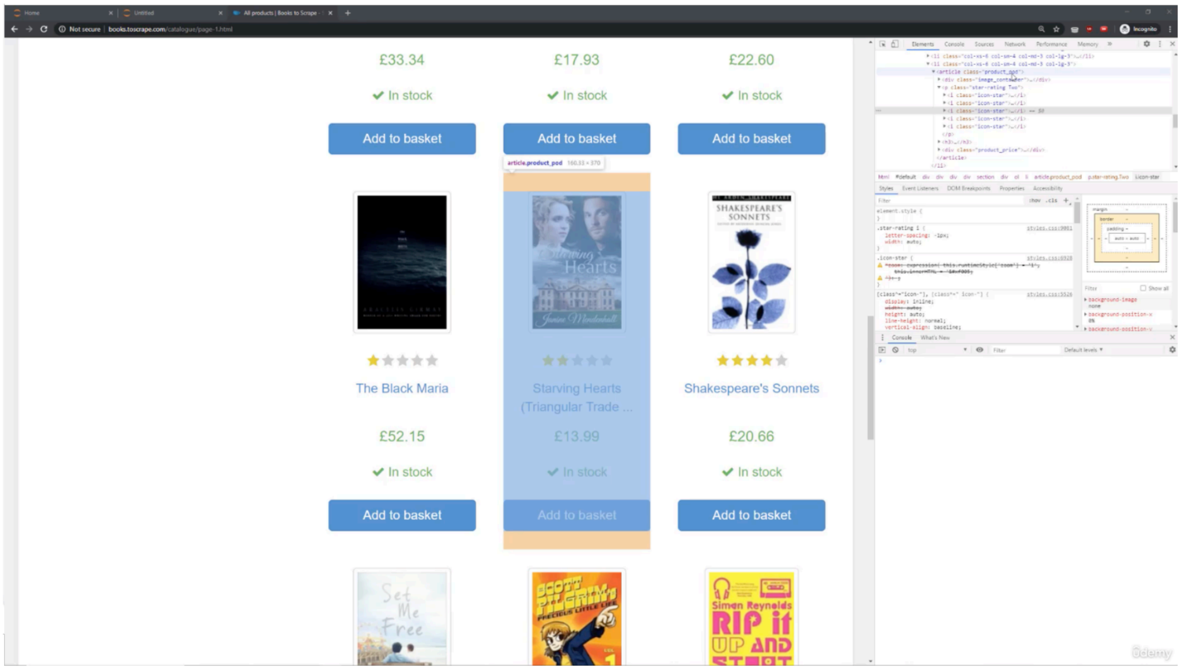
        - **-> we want to figure out what URL procedure is happening when we go from one page to the next**
            - -> we have 20 pages of books on the 'Amazon' site to scrape them from
            - -> each of those webpages follows a different URL syntax
            - -> so he's looking at what that is
            - -> then writing a while loop for the pattern
            - -> it's how we scrape the information from the webpage into Python for processing
            - -> he has stored two of the URLs in strings



```
In [63]: import requests
         import bs4

In [64]: 'http://books.toscrape.com/catalogue/page-2.html'

Out[64]: 'http://books.toscrape.com/catalogue/page-2.html'

In [65]: 'http://books.toscrape.com/catalogue/page-3.html'

Out[65]: 'http://books.toscrape.com/catalogue/page-3.html'

In [66]: base_url = 'http://books.toscrape.com/catalogue/page-{}.html'
```

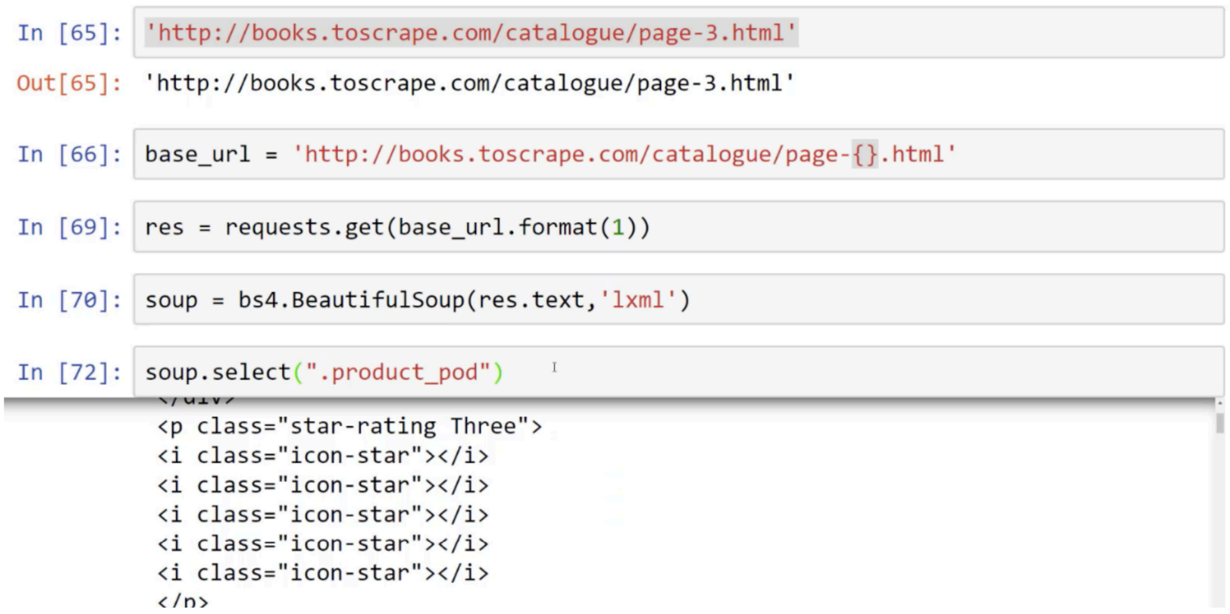
- **-> he generalises the URL of each page and uses the .format method to fill in the different numbers for the pages**

- **-> inspecting the stars on the webpage**



- -> he's inspecting the different stars for the books on the webpage
- -> these are in the form of html font awesome icons
- -> they have different classes
- **-> he's looking a the html for the different elements on the webpage which we want to scrape**

- **-> we can extract certain elements and then filter them**

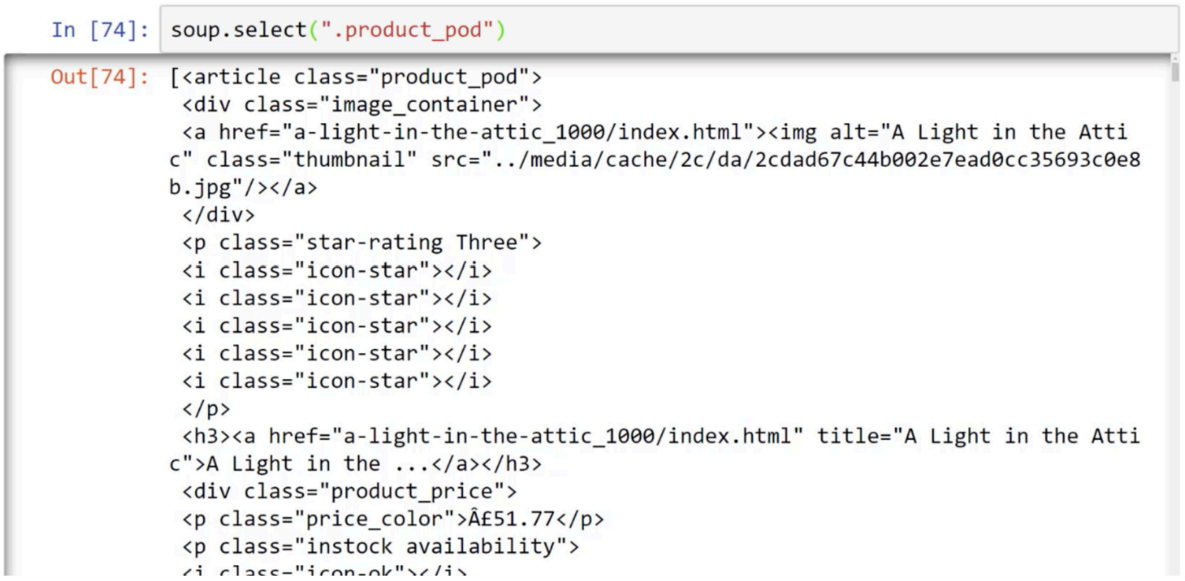


```
In [65]: 'http://books.toscrape.com/catalogue/page-3.html'
Out[65]: 'http://books.toscrape.com/catalogue/page-3.html'

In [66]: base_url = 'http://books.toscrape.com/catalogue/page-{}.html'

In [69]: res = requests.get(base_url.format(1))

In [70]: soup = bs4.BeautifulSoup(res.text,'lxml')

In [72]: soup.select(".product_pod")
```

```
</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>
```

  - -> all of the book elements have a certain class
  - -> he's used the requests.get method on a specific URL
  - -> then converted it into a beautiful soup object
  - -> then he's extracted the html for the books from one of the webpages
  - -> in this case, there are 20 different books, so we know it's for the books of one of the webpages

- **-> beautiful soup is the library we use for web scraping**
    - -> it's a soup object



```
In [74]: soup.select(".product_pod")

Out[74]: [<article class="product_pod">
          <div class="image_container">
          <a href="a-light-in-the-attic_1000/index.html"><img alt="A Light in the Atti
         c" class="thumbnail" src="../media/cache/2c/da/2cdad67c44b002e7ead0cc35693c0e8
         b.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 Atti
         c">A Light in the ...</a></h3>
          <div class="product_price">
          <p class="price_color">Â£51.77</p>
          <p class="instock availability">
```