Top100_ebooks_program

April 6, 2024

0.0.1 Extracting the Top 100 eBooks from Gutenberg

1. Import the necessary libraries, including regex and beautifulsoup.
2. Check the SSL certificate.
3. Read the HTML from the URL.
4. Write a small function to check the status of the web request.
5. Decode the response and pass this on to BeautifulSoup for HTML parsing.
6. Find all the href tags and store them in the list of links. Check what the list looks like – print the first 30 elements.
7. Use a regular expression to find the numeric digits in these links. These are the file numbers for the top 100 eBooks.
8. Initialize the empty list to hold the file numbers over an appropriate range and use regex to find the numeric digits in the link href string. Use the findall method.
9. What does the soup object's text look like? Use the .text method and print only the first 2,000 characters (do not print the whole thing, as it is too long).
10. Search in the extracted text (using a regular expression) from the soup object to find the names of the top 100 eBooks (yesterday's ranking).
11. Create a starting index. It should point at the text Top 100 Ebooks yesterday. Use the splitlines method of soup.text. It splits the lines of text of the soup object.
12. Loop 1-100 to add the strings of the next 100 lines to this temporary list. Hint: use the splitlines method.
13. Use a regular expression to extract only text from the name strings and append it to an empty list. Use match and span to find the indices and use them.

[2]:

```
# Import the necessary libraries, including regex and beautifulsoup.
import urllib.request, urllib.parse, urllib.error
import requests
from bs4 import BeautifulSoup
import ssl
import re
```

[3]:

```
# Ignore SSL certificate errors
ctx = ssl.create_default_context()
ctx.check_hostname = False
ctx.verify_mode = ssl.CERT_NONE
```

[4]:

```
# Read the HTML from the URL and pass on to BeautifulSoup
top100url = 'https://www.gutenberg.org/browse/scores/top'
response = requests.get(top100url)
```

[5]:

```
# Write a small function to check the status of web request
def status_check(r):
    if r.status_code==200:
        print("Success!")
        return 1
    else:
        print("Failed!")
        return -1
```

[6]:

```
status_check(response)
```

```
Success!
```

[6]:

```
1
```

[7]:

```
# Decode the response and pass on to BeautifulSoup for HTML parsing

contents = response.content.decode(response.encoding)
```

[8]:

```
soup = BeautifulSoup(contents, 'html.parser')
```

[10]:

```
# Empty list to hold all the http links in the HTML page
lst_links=[]

# Find all the href tags and store them in the list of links
for link in soup.find_all('a'):
    #print(link.get('href'))
    lst_links.append(link.get('href'))
```

[12]:

```
# Print first 30 rows

lst_links[:30]
```

[12]:

```
['/',
 '/about/',
 '/about/',
 '/policy/collection_development.html',
 '/about/contact_information.html',
 '/about/background/',
 '/policy/permission.html',
 '/policy/privacy_policy.html',
 '/policy/terms_of_use.html',
 '/ebooks/',
 '/ebooks/',
 '/ebooks/bookshelf/',
 '/browse/scores/top',
 '/ebooks/offline_catalogs.html',
 '/help/',
 '/help/',
 '/help/copyright.html',
 '/help/errata.html',
 '/help/file_formats.html',
 '/help/faq.html',
 '/policy/',
 '/help/public_domain_ebook_submission.html',
 '/help/submitting_your_own_work.html',
 '/help/mobile.html',
 '/attic/',
 '/donate/',
 '/donate/',
 '#books-last1',
 '#authors-last1',
 '#books-last7']
```

[13]:

```
# Use regular expression to find the numeric digits in these links.These are␣
 ↪the file number for the Top 100 books.

#Initialize empty list to hold the file numbers

booknum=[]
```

[14]:

```
# For loop the selected range from num 19 to 119 from the link list
for i in range(19,119):
    link=lst_links[i]
    link=link.strip()
    # Regular expression to find the numeric digits in the link (href) string
    n=re.findall('[0-9]+',link)
    if len(n)==1:
        # Append the filenumber casted as integer
        booknum.append(int(n[0]))
```

[15]:

```
# Print the numbers
print ("\nThe file numbers for the top 100 ebooks on Gutenberg are shown␣
 ↪below\n"+"-"*70)
print(booknum)
```

```

The file numbers for the top 100 ebooks on Gutenberg are shown below
----------------------------------------------------------------------
[1, 1, 7, 7, 30, 30, 1513, 2641, 145, 37106, 16389, 2701, 67979, 100, 394, 6761,
2160, 4085, 6593, 5197, 1259, 84, 1342, 17607, 70055, 11, 64317, 174, 98, 68891,
1952, 2542, 70052, 28054, 844, 1080, 76, 1661, 345, 5200, 2554, 408, 4300, 2591,
1260, 42108, 205, 1400, 1232, 46, 25344, 6130, 70051, 5827, 43, 219, 20228,
67098, 2600, 1184, 74, 70056, 36, 30254, 70054, 768, 20203, 3206, 2814, 45,
70049, 730, 158, 996, 37134, 1497, 23, 120, 1727, 514, 4363, 16328, 15399, 5740,
161, 58585, 55, 1399, 829, 47629, 70048, 3207]
```

[16]:

```python
# How does the soup object's text look like? Use .text() method and print only first 2000 characters (i.e. do not print the whole thing, it is long).

print(soup.text[:2000])
```

```




Top 100 | Project Gutenberg






























Menu
```

About

About Project Gutenberg
Collection Development
Contact Us
History & Philosophy
Permissions & License
Privacy Policy
Terms of Use

Search and Browse

Book Search
Bookshelves
Frequently Downloaded
Offline Catalogs

Help

All help topics →
Copyright Procedures
Errata, Fixes and Bug Reports
File Formats
Frequently Asked Questions
Policies →
Public Domain eBook Submission
Submitting Your Own Work
Tablets, Phones and eReaders
The Attic →

Donate

Donation

Frequently Viewed or Downloaded
These listings are based on the number of times each eBook gets downloaded.

Multiple downloads from the same Internet address on the same day count as one download, and addresses that download more than 100 eBooks in a day are considered robots and are not counted.

Downloaded Books
2023-02-17255824
last 7 days1847060
last 30 days8050233

Top 100 EBooks yesterday
Top 100 Authors yesterday
Top 100 EBooks last 7 days
Top 100 Authors last 7 days
Top 100 EBooks last 30 days
Top 100 Authors last 30 days

Top 100 EBooks yesterday

Romeo and Juliet by William Shakespeare (6472)
A Room with a View by E. M. Forster (5652)
Middlemarch by George Eliot (5551)
Little Women; Or, Meg, Jo, Beth, and Amy by Louisa May Alcott (5200)
The Enchanted April by Elizabeth Von Arnim (5114)
Moby Dick; Or, The Whale by Herman Melville (5098)
The Blue Castle: a novel by L. M. Montgomery (5013)

```
The Complete Works of William Shakespeare by William Shakespeare (4979)
Cranford by Elizabeth Cleghorn Gaskell (4844)
The Adventures of Ferdinand Count Fathom - Complete by T.  Smollett (4815)
The Expedition of Humphry Clinker by T.  Smollett (4737)
The Adventures of Roderick Random by T.  Smollett (4691)
History of Tom Jones, a Foundling by Henry Fielding (4501)
My Life - Volume 1 by Richard Wagner (4384)
Twenty Years After by Alexandre Dumas (4355)
Frankenstein; Or, The Moder
```

[17]:

```
# Search in the extracted text (using regular expression) from the soup object␣
 ↪to find the names of top 100 Ebooks (Yesterday's rank)

lst_titles_temp=[]
```

[18]:

```
# Create a starting index. It should point at the text "Top 100 Ebooks␣
 ↪yesterday". Hint: Use splitlines() method of the soup.text. It splits the␣
 ↪lines of the text of the soup object.

start_idx=soup.text.splitlines().index('Top 100 EBooks yesterday')
```

[19]:

```
# Loop 1-100 to add the strings of next 100 lines to this temporary list.

for i in range(100):
    lst_titles_temp.append(soup.text.splitlines()[start_idx+2+i])
```

[20]:

```
# Use regular expression to extract only text from the name strings and append␣
 ↪to an empty list

lst_titles=[]
for i in range(100):
    id1,id2=re.match('^[a-zA-Z ]*',lst_titles_temp[i]).span()
    lst_titles.append(lst_titles_temp[i][id1:id2])
```

[21]:

```
# Print the list of titles

for l in lst_titles:
    print(l)
```

```
Top
Top
Top
Top


Top
```

Romeo and Juliet by William Shakespeare
A Room with a View by E
Middlemarch by George Eliot
Little Women
The Enchanted April by Elizabeth Von Arnim
Moby Dick
The Blue Castle
The Complete Works of William Shakespeare by William Shakespeare
Cranford by Elizabeth Cleghorn Gaskell
The Adventures of Ferdinand Count Fathom
The Expedition of Humphry Clinker by T
The Adventures of Roderick Random by T
History of Tom Jones
My Life
Twenty Years After by Alexandre Dumas
Frankenstein
Pride and Prejudice by Jane Austen
Superstition In All Ages
Satan
Alice
The Great Gatsby by F
The Picture of Dorian Gray by Oscar Wilde
A Tale of Two Cities by Charles Dickens
The alley cat
The Yellow Wallpaper by Charlotte Perkins Gilman
A Doll
Ancient calendars and constellations by Emmeline M
The Brothers Karamazov by Fyodor Dostoyevsky
The Importance of Being Earnest
A Modest Proposal by Jonathan Swift
Adventures of Huckleberry Finn by Mark Twain
The Adventures of Sherlock Holmes by Arthur Conan Doyle
Dracula by Bram Stoker
Metamorphosis by Franz Kafka
Crime and Punishment by Fyodor Dostoyevsky
The Souls of Black Folk by W
Ulysses by James Joyce
Grimms
Jane Eyre
The Slang Dictionary
Walden
Great Expectations by Charles Dickens
The Prince by Niccol
A Christmas Carol in Prose
The Scarlet Letter by Nathaniel Hawthorne
The Iliad by Homer
Gambler
The Problems of Philosophy by Bertrand Russell

The Strange Case of Dr
Heart of Darkness by Joseph Conrad
Noli Me Tangere by Jos
Winnie
War and Peace by graf Leo Tolstoy
The Count of Monte Cristo
The Adventures of Tom Sawyer
Historical record of the
The War of the Worlds by H
The Romance of Lust

Wuthering Heights by Emily Bront
Autobiography of Benjamin Franklin by Benjamin Franklin
Moby Multiple Language Lists of Common Words by Grady Ward
Dubliners by James Joyce
Anne of Green Gables by L
Egyptian decorative art
Oliver Twist by Charles Dickens
Emma by Jane Austen
Don Quixote by Miguel de Cervantes Saavedra
The Elements of Style by William Strunk
The Republic by Plato
Narrative of the Life of Frederick Douglass
Treasure Island by Robert Louis Stevenson
The Odyssey by Homer
Little Women by Louisa May Alcott
Beyond Good and Evil by Friedrich Wilhelm Nietzsche
Beowulf
The Interesting Narrative of the Life of Olaudah Equiano
Tractatus Logico
Sense and Sensibility by Jane Austen
The Prophet by Kahlil Gibran
The Wonderful Wizard of Oz by L
Anna Karenina by graf Leo Tolstoy
Gulliver
Ang
The great Skene mystery by Bernard Capes
Leviathan by Thomas Hobbes
Peter Pan by J
Thus Spake Zarathustra
Second Treatise of Government by John Locke
The peaceful atom by Bernice Kohn Hunt
The Call of the Wild by Jack London
A Study in Scarlet by Arthur Conan Doyle