

▼ Importing Libraries

```
import pandas as pd
import requests
from bs4 import BeautifulSoup as bs
```

▼ Retrieving the page with the drop-down list

```
r=requests.get("https://www.nps.gov/index.htm")
soup=bs(r.content,'html.parser')
print(soup.prettify())
```

```
<!DOCTYPE doctype html>
<html class="no-js" lang="en">
  <!-- Content Copyright National Park Service -->
  <!-- JavaScript & DHTML Code Copyright &copy; 1998-2017, PaperThin, Inc. All Rights
  <head>
    <meta content="IE=edge" http-equiv="X-UA-Compatible"/>
    <title>
      NPS.gov Homepage (U.S. National Park Service)
    </title>
    <script>
      var jsDlglLoader = '/loader.cfm';
var jsSiteResourceLoader = '/cs-resources.cfm?r=';
var jsSiteResourceSettings = {canCombine: false, canMinify: false};
    </script>
    <!-- beg (1) PrimaryResources -->
    <script src="/ADF/thirdParty/jquery/jquery-1.12.js" type="text/javascript">
    </script>
    <script src="/commonspot/javascript/browser-all.js" type="text/javascript">
    </script>
    <!-- end (1) PrimaryResources -->
    <!-- beg (2) SecondaryResources -->
    <!-- end (2) SecondaryResources -->
    <!-- beg (3) StyleTags -->
    <link href="/common/commonspot/templates/assets/css/commonspotRTE.css" rel="stylesh
    <!-- end (3) StyleTags -->
    <!-- beg (4) JavaScript -->
    <script type="text/javascript">
      <!--
var gMenuControlID = 0;
var menus_included = 0;
var jsSiteID = 1;
var jsSubSiteID = 1;
var js_gvPageID = 5102204;
var jsPageID = 5102204;
var jsPageSetID = 0;
```

```

var jsPageType = 0;
var jsSiteSecurityCreateControls = 0;
var jsShowRejectForApprover = 1;
// -->
</script>
<script>
  var jsDlgLoader = '/loader.cfm';
var jsSiteResourceLoader = '/cs-resources.cfm?r=';
var jsSiteResourceSettings = {canCombine: false, canMinify: false};
</script>
<!-- end (4) JavaScript -->
<!-- beg (5) CustomHead -->
<!-- override Global Homepage from NPS Homepage-->
<meta charset="utf-8"/>
<link href="/common/commonspot/templates/images/icons/favicon.ico" rel="shortcut ic
<meta content="The National Park Service cares for special places saved by the Amer
<meta content="no-cache" http-equiv="Pragma">
<meta content="0" http-equiv="Expires"/>
<meta content="width=device-width, initial-scale=1.0" name="viewport"/>
<!--[if gt IE 8]>
<script src="/common/commonspot/templates/assetsCT/javascripts/app.early.min.js?v=3ht

```

▼ Retrieving the drop-down list

```

s=soup.find("ul",attrs={"class":"dropdown-menu SearchBar-keywordSearch"})
print(s.prettify())

```

```

<ul class="dropdown-menu SearchBar-keywordSearch" role="menu" style="max-height: 300p
<li>
  <a href="/state/al/index.htm">
    Alabama
  </a>
</li>
<li>
  <a href="/state/ak/index.htm">
    Alaska
  </a>
</li>
<li>
  <a href="/state/as/index.htm">
    American Samoa
  </a>
</li>
<li>
  <a href="/state/az/index.htm">
    Arizona
  </a>
</li>
<li>
  <a href="/state/ar/index.htm">
    Arkansas
  </a>

```

```

</li>
<li>
  <a href="/state/ca/index.htm">
    California
  </a>
</li>
<li>
  <a href="/state/co/index.htm">
    Colorado
  </a>
</li>
<li>
  <a href="/state/ct/index.htm">
    Connecticut
  </a>
</li>
<li>
  <a href="/state/de/index.htm">
    Delaware
  </a>
</li>
<li>
  <a href="/state/dc/index.htm">
    District of Columbia
  </a>
</li>
<li>
  <a href="/state/fl/index.htm">
    Florida
  </a>
</li>

```

▼ Retrieving the links to all states from page 1 (List of States)

```

states=s.select("a")
Links=[link['href'] for link in states]
df=pd.DataFrame(states,columns=['Name of States'])
df1=pd.DataFrame(Links,columns=['Links for States'])
list_of_states=pd.merge(df,df1,left_index=True,right_index=True)
list_of_states.to_csv('list_of_states.csv',index=False)
list_of_states

```

	Name of States	Links for States
0	Alabama	/state/al/index.htm
1	Alaska	/state/ak/index.htm
2	American Samoa	/state/as/index.htm
3	Arizona	/state/az/index.htm
4	Arkansas	/state/ar/index.htm
5	California	/state/ca/index.htm
6	Colorado	/state/co/index.htm
7	Connecticut	/state/ct/index.htm
8	Delaware	/state/de/index.htm
9	District of Columbia	/state/dc/index.htm
10	Florida	/state/fl/index.htm
11	Georgia	/state/ga/index.htm
12	Guam	/state/gu/index.htm
13	Hawaii	/state/hi/index.htm
14	Idaho	/state/id/index.htm
15	Illinois	/state/il/index.htm
16	Indiana	/state/in/index.htm
17	Iowa	/state/ia/index.htm
18	Kansas	/state/ks/index.htm
19	Kentucky	/state/ky/index.htm
20	Louisiana	/state/la/index.htm
21	Maine	/state/me/index.htm
22	Maryland	/state/md/index.htm
23	Massachusetts	/state/ma/index.htm
24	Michigan	/state/mi/index.htm
25	Minnesota	/state/mn/index.htm

▼ Retrieving each state's page

28 Montana /state/mt/index.htm

```
url = 'https://www.nps.gov'
for link in Links:
```

```
res = requests.get(url + link)
Page = bs(res.content, 'html.parser')
```

Retrieving the list of all parks from page 2 for each state (List of Parks)

```
url = 'https://www.nps.gov'
list_of_parks = []
for link in Links:
    res = requests.get(url + link)
    Page = bs(res.content, 'html.parser')
    parks = Page.find("ul",{ "id": "list_parks" }).find_all("li",{ "class": "clearfix" })
    for eachpark in parks:
        park = {}
        park['List of Parks'] = eachpark.find('h3').getText()
        list_of_parks.append(park)
df2=pd.DataFrame(list_of_parks,columns=['List of Parks'])
df2.to_csv('List_of_Parks.csv',index=False)
df2
```

List of Parks	
0	Birmingham Civil Rights
1	Freedom Riders
2	Horseshoe Bend
3	Little River Canyon
4	Natchez Trace
...	...
626	Grand Teton
627	Mormon Pioneer
628	Oregon
629	Pony Express
630	Yellowstone
631 rows × 1 columns	

Retrieving each park’s name, category, and description from the above list of parks in each state

```
url = 'https://www.nps.gov'
Parks_Info = []
for link in Links:
    res = requests.get(url + link)
    Page = bs(res.content, 'html.parser')
    parks = Page.find("ul",{"id":"list_parks"}).find_all("li",{"class":"clearfix"})
    for eachpark in parks:
        park = {}
        park['Name'] = eachpark.find('h3').getText()
        park['Category'] = eachpark.find('h2').getText()
        park['Description'] = eachpark.find('p').getText().replace('\n','')
        Parks_Info.append(park)
df3=pd.DataFrame(Parks_Info,columns=['Name', 'Category', 'Description'])
df3.to_csv('Parks_Info.csv',index=False)
df3
```

	Name	Category	Description
0	Birmingham Civil Rights	National Monument	In 1963, images of snarling police dogs unleas
1	Freedom Riders	National Monument	In 1961, a small interracial band of “Freedom
2	Horseshoe Bend	National Military Park	On 27 March 1814, Major General Andrew Jackson
3	Little River Canyon	National Preserve	Little River is unique because it flows for mc
4	Natchez Trace	Parkway	The Natchez Trace Parkway is a 444-mile recrea
...	
626	Grand Teton	National Park	Rising above a scene rich with extraordinary w
627	Mormon Pioneer	National Historic Trail	Explore the Mormon Pioneer National Historic T
628	Oregon	National Historic Trail	Imagine yourself an emigrant headed for Oregon
629	Pony Express	National Historic Trail	It is hard to believe that young men once rode
630	Yellowstone	National Park	On March 1, 1872, Yellowstone became the first

631 rows × 3 columns

Retrieving each park’s page

```

Parks_Links = []
for link in Links:
    res = requests.get(url + link)
    Page = bs(res.content, 'html.parser')
    parks = Page.find("ul",{"id":"list_parks"}).find_all("li",{"class":"clearfix"})
    for eachpark in parks:
        park = {}
        park = eachpark.find('h3').findChild('a')['href']
        Parks_Links.append(str(park))
df4=pd.DataFrame(Parks_Links,columns=['Parks_Links'])
df4.to_csv('Parks_Links.csv',index=False)
df4

```

	Parks_Links
0	/bicr/
1	/frri/
2	/hobe/
3	/liri/
4	/natr/
...	...
626	/grte/
627	/mopi/
628	/oreg/
629	/poex/
630	/yell/

631 rows × 1 columns

Retrieving each park's contact information from page 3 for each park

```

url = 'https://www.nps.gov'
Parks_Data=[]
for links in Parks_Links:
    L = requests.get(url + links)
    bs_page = bs(L.content, 'html.parser')
    pages=bs_page.find_all('div',attrs={'class':'vcard'})
    for eachpage in pages:
        parkdata={}

```

```
Address = eachpage.find("span",attrs={"class":"street-address"})
if Address:
    addresstxt = Address.getText()
    address_lines = addresstxt.splitlines()
    parkdata['Street Address Line 1'] = len(address_lines)>1 and address_lines[1] or 'null'
    parkdata['Line 2'] = len(address_lines)>2 and address_lines[2] or 'null'
    parkdata['Line 3'] = len(address_lines)>3 and address_lines[3] or 'null'
City=eachpage.find('span',attrs={'itemprop':'addressLocality'})
if City:
    parkdata['City']=City.getText()
State=eachpage.find('span',attrs={'class':'region'})
if State:
    parkdata['State']=State.getText()
Zip_Code=eachpage.find('span',attrs={'class':'postal-code'})
if Zip_Code:
    parkdata['Zip Code']=Zip_Code.getText()
Phone_Number=eachpage.find('span',attrs={'class':'tel'})
if Phone_Number:
    parkdata['Phone Number']=Phone_Number.getText().replace('\n','')
Parks_Data.append(parkdata)
df5=pd.DataFrame(Parks_Data,columns=['Street Address Line 1','Line 2','Line 3','City','State']
All_Parks_Data=pd.merge(df3,df5,left_index=True,right_index=True)
All_Parks_Data.to_csv('All_Parks_Data.csv',index=False)
All_Parks_Data
```


12/9/21, 2:14 AMNational_Parks.ipynb - Colaboratory

	Name	Category	Description	Street Address Line 1	Line 2	Line 3	City	State	
0	Birmingham Civil Rights	National Monument	In 1963, images of snarling police dogs unleas...	Birmingham Civil Rights National Monument	1914 4th Street North	Suite 440	Birmingham	AL	3
1	Freedom Riders	National Monument	In 1961, a small interracial band of "Freedom ...	1304 Noble Street	null	null	Anniston	AL	3
2	Horseshoe Bend	National Military Park	On 27 March 1814, Major General Andrew Jackson...	11288 Horseshoe Bend Road	null	null	Daviston	AL	3
3	Little River Canyon	National Preserve	Little River is unique because it flows for mo...	4322 Little River Trail NE Ste 100	null	null	Fort Payne	AL	3

X