In [2]:
```
##### learned from this link
#https://www.analyticsvidhya.com/blog/2015/10/beginner-guide-web-scraping-beau
```

In [5]:
```
#Urllib2: It is a Python module which can be used for fetching URLs
#BeautifulSoup: It is an incredible tool for pulling out information from a we
     #You can use it to extract tables, lists, paragraph and you can also put

#BeautifulSoup does not fetch the web page for us. That's why, I use urllib2 i

#learn basics of html from the link mentionedd above because you should know b

from bs4 import BeautifulSoup
import urllib.request
```

In [6]:
```
wiki = "https://en.wikipedia.org/wiki/List_of_state_and_union_territory_capita
```

In [8]:
```
page = urllib.request.urlopen(wiki)
page
```

Out[8]:
```
<http.client.HTTPResponse at 0x20dbe801b38>
```

In [9]:
```
soup = BeautifulSoup(page)
soup.prettify()
```

Out[9]:
```
'<!DOCTYPE html>\n<html class="client-nojs" dir="ltr" lang="en">\n <head>\n
<meta charset="utf-8"/>\n  <title>\n   List of state and union territory ca
pitals in India - Wikipedia\n  </title>\n  <script>\n   document.documentEl
ement.className=document.documentElement.className.replace(/(^|\\s)client-n
ojs(\\s|$)/,"$1client-js$2");RLCONF={"wgCanonicalNamespace":"","wgCanonical
SpecialPageName":!1,"wgNamespaceNumber":0,"wgPageName":"List_of_state_and_u
nion_territory_capitals_in_India","wgTitle":"List of state and union territ
ory capitals in India","wgCurRevisionId":904438853,"wgRevisionId":90443885
3,"wgArticleId":2371868,"wgIsArticle":!0,"wgIsRedirect":!1,"wgAction":"vie
w","wgUserName":null,"wgUserGroups":["*"],"wgCategories":["Use dmy dates fr
om November 2018","Use Indian English from November 2018","All Wikipedia ar
ticles written in Indian English","States and union territories of India-re
lated lists","Indian capital cities","Lists of cities in India","Lists of c
apitals of country subdivisions","Cities and towns in India by state or ter
ritory"],"wgBreakFrames":!1,"wgPageContentLanguage":"en","wgPageContentMode
l":"wikitext","wgSeparatorTransformTable":["",""],"wgDigitTransformTabl
e":\n["",""],"wgDefaultDateFormat":"dmy","wgMonthNames":["","January","Febr
uary","March","April","May","June","July","August","September","October","N
ovember","December"],"wgMonthNamesShort":["","Jan","Feb","Mar","Apr","Ma
```

In [13]:
```
# Return content between opening and closing tag including tag.
soup.title
```

Out[13]:
```
<title>List of state and union territory capitals in India - Wikipedia</title
>
```

In [15]:
```
# Return string within given tag
soup.title.string
```

Out[15]:
```
'List of state and union territory capitals in India - Wikipedia'
```

In [17]:
```
#We know that, we can tag a link using tag “<a>”.
#So, we should go with option soup.a and it should return the links available

soup.a
```

Out[17]:
```
<a id="top"></a>
```

In [21]:
```
#Above, you can see that, we have only one output. Now to extract all the link

all_link =soup.findAll("a")
all_link
```

Out[21]:
```
[<a id="top"></a>,
 <a class="mw-jump-link" href="#mw-head">Jump to navigation</a>,
 <a class="mw-jump-link" href="#p-search">Jump to search</a>,
 <a href="/wiki/States_and_union_territories_of_India" title="States and un
ion territories of India">States and union <br/> territories of India</a>,
 <a class="image" href="/wiki/File:Flag_of_India.svg"><img alt="Flag of Ind
ia.svg" data-file-height="900" data-file-width="1350" decoding="async" heig
ht="47" src="//upload.wikimedia.org/wikipedia/en/thumb/4/41/Flag_of_India.s
vg/70px-Flag_of_India.svg.png" srcset="//upload.wikimedia.org/wikipedia/en/
thumb/4/41/Flag_of_India.svg/105px-Flag_of_India.svg.png 1.5x, //upload.wik
imedia.org/wikipedia/en/thumb/4/41/Flag_of_India.svg/140px-Flag_of_India.sv
g.png 2x" width="70"/></a>,
 <a href="/wiki/List_of_states_and_union_territories_of_India_by_area" titl
e="List of states and union territories of India by area">Area</a>,
 <a href="/wiki/List_of_states_and_union_territories_of_India_by_populatio
n" title="List of states and union territories of India by population">Popu
lation</a>,
 <a href="/wiki/List_of_Indian_states_and_union_territories_by_GDP" title
="List of Indian states and union territories by GDP">GDP</a>,
```

In [23]:
```
#href is present in 'a tag'
#The href attribute specifies the URL of the page the link goes to.
#If the href attribute is not present, the <a> tag is not a hyperlink

for link in all_link:
    print(link.get('href'))
```

```
None
#mw-head
#p-search
/wiki/States_and_union_territories_of_India
/wiki/File:Flag_of_India.svg
/wiki/List_of_states_and_union_territories_of_India_by_area
/wiki/List_of_states_and_union_territories_of_India_by_population
/wiki/List_of_Indian_states_and_union_territories_by_GDP
/wiki/List_of_Indian_states_and_union_territories_by_GDP_per_capita
/wiki/ISO_3166-2:IN
None
/wiki/List_of_Indian_states_by_Child_Nutrition
/wiki/List_of_states_and_union_territories_of_India_by_crime_rate
/wiki/List_of_states_and_union_territories_of_India_by_households_having_el
ectricity
/wiki/List_of_states_and_union_territories_of_India_by_fertility_rate
/wiki/Forest_cover_by_state_in_India
/wiki/Ease_of_doing_business_ranking_of_states_of_India
/wiki/List_of_Indian_states_and_territories_by_highest_point
```

In [26]:
```
#Find the right table: As we are seeking a table to extract information about
#we should identify the right table first. Let's write the command to extract

all_tables=soup.find_all('table')
all_tables
```

Out[26]:
```
[<table class="vertical-navbox nowraplinks" style="float:right;clear:right;
width:22.0em;margin:0 0 1.0em 1.0em;background:#f9f9f9;border:1px solid #aa
a;padding:0.2em;border-spacing:0.4em 0;text-align:center;line-height:1.4em;
font-size:88%"><tbody><tr><th style="padding:0.2em 0.4em 0.2em;font-size:14
5%;line-height:1.2em"><a href="/wiki/States_and_union_territories_of_India"
title="States and union territories of India">States and union <br/> territ
ories of India</a> <br/> ordered by</th></tr><tr><td style="padding:0.2em 0
0.4em"><div class="center"><div class="floatnone"><a class="image" href="/w
iki/File:Flag_of_India.svg"><img alt="Flag of India.svg" data-file-height
="900" data-file-width="1350" decoding="async" height="47" src="//upload.wi
kimedia.org/wikipedia/en/thumb/4/41/Flag_of_India.svg/70px-Flag_of_India.sv
g.png" srcset="//upload.wikimedia.org/wikipedia/en/thumb/4/41/Flag_of_Indi
a.svg/105px-Flag_of_India.svg.png 1.5x, //upload.wikimedia.org/wikipedia/e
n/thumb/4/41/Flag_of_India.svg/140px-Flag_of_India.svg.png 2x" width="70"/>
</a></div></div></td></tr><tr><td class="hlist" style="padding:0 0.1em 0.4e
m">
 <ul><li><a href="/wiki/List_of_states_and_union_territories_of_India_by_ar
ea" title="List of states and union territories of India by area">Area</a>
</li>
```

In [41]:
```
#Now to identify the right table, we will use attribute “class” of table and u
#In chrome, you can check the class name by right click on the required table
#Inspect element -> Copy the class name OR go through the output of above comm

soup.find_all('table', class_="wikitable sortable plainrowheaders")
```

Out[41]:
```
[<table class="wikitable sortable plainrowheaders">
 <tbody><tr>
 <th scope="col">No.
 </th>
 <th scope="col">State or<br/>union territory
 </th>
 <th scope="col">Administrative capital
 </th>
 <th scope="col">Legislative capital
 </th>
 <th scope="col">Judicial capital
 </th>
 <th scope="col">Year of establishment
 </th>
 <th scope="col">Former capital
 </th></tr>
 <tr>
 <td>1
 </td>
```

In [42]:
```
#Extract the information to DataFrame: Here, we need to iterate through each r
#to a variable and append it to a list. Let’s first look at the HTML structure
#(I am not going to extract information for table heading <th>)
right_table = soup.find('table' , {"class" : "wikitable sortable plainrowheade
right_table
```

Out[42]:
```
<table class="wikitable sortable plainrowheaders">
<tbody><tr>
<th scope="col">No.
</th>
<th scope="col">State or<br/>union territory
</th>
<th scope="col">Administrative capital
</th>
<th scope="col">Legislative capital
</th>
<th scope="col">Judicial capital
</th>
<th scope="col">Year of establishment
</th>
<th scope="col">Former capital
</th></tr>
<tr>
<td>1
</td>
```

In [47]:
```
#Above, you can notice that second element of <tr> is within tag <th> not <td>
#Now to access value of each element, we will use “find(text=True)” option wit

#Generate lists
A=[]
B=[]
C=[]
D=[]
E=[]
F=[]
G=[]
for row in right_table.findAll("tr"):
    cells = row.findAll('td')
    states=row.findAll('th') #To store second column data
    if len(cells)==6: #Only extract table body not heading
        A.append(cells[0].find(text=True))
        B.append(states[0].find(text=True))
        C.append(cells[1].find(text=True))
        D.append(cells[2].find(text=True))
        E.append(cells[3].find(text=True))
        F.append(cells[4].find(text=True))
        G.append(cells[5].find(text=True))
```

In [48]:
```
#import pandas to convert list to data frame
import pandas as pd
df=pd.DataFrame(A,columns=['Number'])
df['State/UT']=B
df['Admin_Capital']=C
df['Legislative_Capital']=D
df['Judiciary_Capital']=E
df['Year_Capital']=F
df['Former_Capital']=G
df
```

Out[48]:

	Number	State/UT	Admin_Capital	Legislative_Capital	Judiciary_Capital	Year_Capit
0	1	Andaman and Nicobar Islands	Port Blair	—	Kolkata	195
1	2	Andhra Pradesh	Hyderabad	Amaravati	Amaravati	195
2	3	Arunachal Pradesh	Itanagar	Itanagar	Guwahati	198
3	4	Assam	Dispur	Guwahati	Guwahati	197
4	5	Bihar	Patna	Patna	Patna	191
5	6	Chandigarh	Chandigarh	—	Chandigarh	196
6	7	Chhattisgarh	Raipur	Raipur	Bilaspur	200
7	8	Dadra and	Silvassa	—	Mumbai	194

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: