In [3]:
```
import pandas as pd
import requests
from bs4 import BeautifulSoup
url = 'https://en.wikipedia.org/wiki/History_of_Python'

# Extract tables
dfs = pd.read_html(url)

# Get first table
df = dfs[0]

# Extract columns
df2 = df[['Version','Release date']]
print(df2)
```

```
                                              Version  \
0                                                 NaN
1                                                 NaN
2                                                 NaN
3                                                 NaN
4                                                 NaN
5                                                 NaN
6                                                 NaN
7                                                 NaN
8                                                 NaN
9                                                 NaN
10                                                NaN
11                                                NaN
12                                                NaN
13                                                NaN
14                                                NaN
15                                                NaN
16                                                NaN
17                                                NaN
18                                                NaN
19                                                NaN
20                                                NaN
21                                                NaN
22                                                NaN
23                                                NaN
24                                                NaN
25                                                NaN
26                                               3.10
27                                     [needs update]
28                                                NaN
29                                            Legend:
30  Italics indicates the latest micro version of ...

                                        Release date
0                                       1991-02-20[2]
1                                       1994-01-26[2]
2                                       1994-10-11[2]
3                                       1995-04-13[2]
4                                       1995-10-13[2]
5                                       1996-10-25[2]
6                                       1998-01-03[2]
7                                      2000-09-05[43]
```

```
8                                          2000-10-16[45]
9                                          2001-04-15[46]
10                                         2001-12-21[47]
11                                         2003-06-29[48]
12                                         2004-11-30[49]
13                                         2006-09-19[50]
14                                         2008-10-01[27]
15                                         2010-07-03[32]
16                                         2008-12-03[27]
17                                         2009-06-27[52]
18                                         2011-02-20[54]
19                                         2012-09-29[55]
20                                         2014-03-16[56]
21                                         2015-09-13[58]
22                                         2016-12-23[60]
23                                         2018-06-27[61]
24                                         2019-10-14[62]
25                                         2020-10-05[63]
26                                         2021-10-04[65]
27                                         2022-10-03[66]
28                                            2023-10[64]
29  Old versionOlder version, still maintainedLate...
30  Italics indicates the latest micro version of ...
```

In [4]:
```
#Write a Python program to test if a given page is found or not on the server
from urllib.request import urlopen  #This module helps to define functions and c
from urllib.error import HTTPError
from urllib.error import URLError

try:
    html = urlopen("https://abcxyz.com")
except HTTPError as e:
    print("HTTP error")
except URLError as e:
    print("Server not found!")
else:
    print(html.read())

try:
    html = urlopen("http://www.example.com/")
except HTTPError as e:
    print("HTTP error")
except URLError as e:
    print("Server not found!")
else:
    print("HTML Details")
    print(html.read())
```

```
Server not found!
HTML Details
b'<!doctype html>\n<html>\n<head>\n    <title>Example Domain</title>\n\n    <me
ta charset="utf-8" />\n    <meta http-equiv="Content-type" content="text/html;
charset=utf-8" />\n    <meta name="viewport" content="width=device-width, initi
al-scale=1" />\n    <style type="text/css">\n    body {\n        background-col
or: #f0f0f2;\n        margin: 0;\n        padding: 0;\n        font-family: -ap
ple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans", "Helvetica
Neue", Helvetica, Arial, sans-serif;\n        \n    }\n    div {\n        widt
h: 600px;\n        margin: 5em auto;\n        padding: 2em;\n        background
-color: #fdfdff;\n        border-radius: 0.5em;\n        box-shadow: 2px 3px 7p
x 2px rgba(0,0,0,0.02);\n    }\n    a:link, a:visited {\n        color: #38488
f;\n        text-decoration: none;\n    }\n    @media (max-width: 700px) {\n
div {\n            margin: 0 auto;\n            width: auto;\n        }\n    }
\n    </style>    \n</head>\n\n<body>\n<div>\n    <h1>Example Domain</h1>\n
<p>This domain is for use in illustrative examples in documents. You may use th
is\n    domain in literature without prior coordination or asking for permissio
n.</p>\n    <p><a href="https://www.iana.org/domains/example">More informatio
n...</a></p>\n</div>\n</body>\n</html>\n'
```

In [6]:
```
from urllib.request import urlopen
from urllib.error import HTTPError
from urllib.error import URLError

try:
    html = urlopen("http://www.example.com/")
except HTTPError as e:
    print("HTTP error")
except URLError as e:
    print("Server not found!")
else:
    print("HTML DETAILS")
    print(html.read())
```

```
HTML DETAILS
b'<!doctype html>\n<html>\n<head>\n    <title>Example Domain</title>\n\n    <me
ta charset="utf-8" />\n    <meta http-equiv="Content-type" content="text/html;
charset=utf-8" />\n    <meta name="viewport" content="width=device-width, initi
al-scale=1" />\n    <style type="text/css">\n    body {\n        background-col
or: #f0f0f2;\n        margin: 0;\n        padding: 0;\n        font-family: -ap
ple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans", "Helvetica
Neue", Helvetica, Arial, sans-serif;\n        \n    }\n    div {\n        widt
h: 600px;\n        margin: 5em auto;\n        padding: 2em;\n        background
-color: #fdfdff;\n        border-radius: 0.5em;\n        box-shadow: 2px 3px 7p
x 2px rgba(0,0,0,0.02);\n    }\n    a:link, a:visited {\n        color: #38488
f;\n        text-decoration: none;\n    }\n    @media (max-width: 700px) {\n
div {\n            margin: 0 auto;\n            width: auto;\n        }\n    }
\n    </style>    \n</head>\n\n<body>\n<div>\n    <h1>Example Domain</h1>\n
<p>This domain is for use in illustrative examples in documents. You may use th
is\n    domain in literature without prior coordination or asking for permissio
n.</p>\n    <p><a href="https://www.iana.org/domains/example">More informatio
n...</a></p>\n</div>\n</body>\n</html>\n'
```

In [8]:
```
from urllib.request import urlopen
from urllib.error import HTTPError
from urllib.error import URLError

try:
    html = urlopen("file:///C:/Users/MADHU/Desktop/WEB/sample.html")
except HTTPError as e:
    print("HTTP Error")
except URLError as e:
    print("Server not found!")
else:
    print("HTML DETAILS")
    print(html.read())
```

```
HTML DETAILS
b'<html>\r\n<body style="background-color:powderblue">\r\n<h1 style="text-alig
n:center">STUDENT REGISTRATION FORM</h1>\r\n<form>\r\nStudent\'s Name:  <input
type="text" name="Name"><br>\r\n<br>\r\nFather\'s Name:  <input type="text" nam
e="Name"><br>\r\n<br>\r\nMother\'s Name:  <input type="text" name="Name"><br><b
r>\r\nPhone Number: <input type="text" name="Name"><br><br>\r\nEmail:<input typ
e="text" name="Name"><br><br>\r\nGender: <input type="radio" name="male" value
="on">Male\r\n<input type="radio" name="male" value="on">Female\r\n<input type
="radio" name="male" value="on">Other<br><br>\r\nAddress:<br>\r\n<textarea rows
="10" cols="50" name="description">\r\nEnter your address here...\r\n</textarea
><br><br>\r\nBlood Group: \r\n<select name="dropdown">\r\n     <option value="s
elect">select</option>\r\n     <option value="O+">O+</option>\r\n     <option v
alue="O-">O-</option>\r\n     <option value="AB+">AB+</option>\r\n     <option
value="AB-">AB-</option>\r\n     <option value="A+">A+</option>\r\n     <option
value="A-">A-</option>\r\n</select><br> <br>\r\nCourse:\r\n<input type="checkbo
x" name="C" value="on">C \r\n<input type="checkbox" name="C++" value="on">C++
\r\n<input type="checkbox" name="Java" value="on">Java\r\n<input type="checkbo
x" name="Robotics" value="on">Robotics \r\n<input type="checkbox" name="AI" val
ue="on">AI <br><br>\r\nPhoto:\r\n<input type = "file" name = "fileupload" accep
t = "image/*" /><br><br>\r\n<input type = "submit" name = "submit" value = "Sub
mit" />\r\n <input type = "reset" name = "reset"  value = "Reset" />\r\n <input
type = "button" name = "ok" value = "OK" />\r\n     \r\n</form>\r\n\r\n</body>
\r\n</html>'
```

In [11]:
```
from urllib.request import urlopen
from urllib.error import HTTPError
from urllib.error import URLError
try:
    html = urlopen("https://www.google.co.in/")
except HTTPError as e:
    print("HTTP Error")
except URLError as e:
    print("Server not found")
else:
    print("HTML DETAILS")
    print(html.read())
```

```
HTML DETAILS
b'<!doctype html><html itemscope="" itemtype="http://schema.org/WebPage" lang
="en-IN"><head><meta content="text/html; charset=UTF-8" http-equiv="Content-T
ype"><meta content="/images/branding/googleg/1x/googleg_standard_color_128dp.
png" itemprop="image"><title>Google</title><script nonce="QFlJFHVwUiw5SfqmveD
R8A">(function(){window.google={kEI:\'iCbAYv-oBcmhoASh5K7QDg\',kEXPI:\'0,1302
536,56873,6058,207,4804,2316,383,246,5,1354,4013,1123753,1197768,380723,1611
4,28684,17572,4859,1361,9291,3026,17582,4020,978,13228,3847,10622,7432,15309,
5081,889,704,1279,2742,149,562,541,840,6297,3514,606,2023,1777,520,14670,322
7,2845,7,4808,12642,15768,552,1850,2615,3784,9358,3,346,230,1014,1,5444,149,1
1325,989,1661,4,1528,2304,7039,20309,1714,3050,2658,7357,31723,3158,651,5161,
2545,4094,4052,3,3541,1,42154,2,14022,2715,3533,7868,11623,5679,1021,2380,271
8,18261,2,2,5,7754,4568,6255,23421,1249,5838,14968,4332,8,6082,1394,445,2,2,
1,10790,13836,1928,78,8155,6581,800,2,2958,82,8730,2908,7341,2650,11805,7,192
2,5703,3469,54,553,24,5415,902,547,1278,1662,2,4050,428,1415,1496,420,4296,10
79,1409,6040,4911,751,29,41,3742,1060,527,583,117,41,420,28,2027,910,3245,81,
2,2656,371,180,417,568,122,156,259,285,4,1,2,2,2,2,2179,545,150,845,1684,71,4
79,879,97,924,18,337,1376,95,44,1878,688,231,155,181,852,208,921,1748,21,127,
532,19,1701,808,57,10,4,588,266,3,494,754,2,11,490,5393140,474,124,23,599522
```

In [12]:
```
#Write a Python program to get the number of datasets currently listed on data.g
from lxml import html  #lxml is a Python library which allows for easy handling 
import requests
response = requests.get("http://www.data.gov/")
doc_gov = html.fromstring(response.text) #fromstring() function create a new one
link_gov = doc_gov.cssselect('small a')[0] #cssselect is a BSD-licensed Python l
print("Number of datasets currently listed on data.gov:")
print(link_gov.text)
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
~\Anaconda3\lib\site-packages\lxml\cssselect.py in <module>
     12 try:
---> 13     import cssselect as external_cssselect
     14 except ImportError:

ModuleNotFoundError: No module named 'cssselect'

During handling of the above exception, another exception occurred:

ImportError                               Traceback (most recent call last)
<ipython-input-12-642cde662e49> in <module>
      4 response = requests.get("http://www.data.gov/")
      5 doc_gov = html.fromstring(response.text) #fromstring() function create
 a new one-dimensional array initialized from text data in a string.
----> 6 link_gov = doc_gov.cssselect('small a')[0] #cssselect is a BSD-licensed
Python library to parse CSS3 selectors and translate them to XPath 1.0 expressi
ons.
      7 print("Number of datasets currently listed on data.gov:")
      8 print(link_gov.text)

~\Anaconda3\lib\site-packages\lxml\html\__init__.py in cssselect(self, expr, tr
anslator)
    429         """
    430         # Do the import here to make the dependency optional.
--> 431         from lxml.cssselect import CSSSelector
    432         return CSSSelector(expr, translator=translator)(self)
    433

~\Anaconda3\lib\site-packages\lxml\cssselect.py in <module>
     14 except ImportError:
     15     raise ImportError(
---> 16         'cssselect does not seem to be installed. '
     17         'See http://packages.python.org/cssselect/') (http://packages.p
ython.org/cssselect/'))
     18

ImportError: cssselect does not seem to be installed. See http://packages.pytho
n.org/cssselect/ (http://packages.python.org/cssselect/)
```

In [15]:
```
#Write a Python program to extract h1 tag from example.com.
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen('http://www.example.com/')
bsh = BeautifulSoup(html.read(), 'html.parser')
print(bsh.h1)
```

```
<h1>Example Domain</h1>
```

In [16]:
```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen('file:///C:/Users/MADHU/Desktop/WEB/sample.html')
bsh = BeautifulSoup(html.read(), 'html.parser')
print(bsh.h1)
```

```
<h1 style="text-align:center">STUDENT REGISTRATION FORM</h1>
```

In [17]:
```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen('file:///C:/Users/MADHU/Desktop/WEB/sample.html')
bsh = BeautifulSoup(html, 'html.parser')
titles = bsh.find_all(['h1','h2','h3'])
print("ALL HEADER TAGS: ",*titles,sep='\n\n')
```

```
ALL HEADER TAGS:

<h1 style="text-align:center">STUDENT REGISTRATION FORM</h1>
```

In [18]:
```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen('https://www.udemy.com/')
bsh = BeautifulSoup(html, 'html.parser')
titles = bsh.find_all(['h1','h2','h3'])
print("ALL HEADER TAGS: ",*titles,sep='\n\n')
```

```
ALL HEADER TAGS:

<h1 class="udlite-heading-serif-xxl billboard-banner--short-title--3HjCw" data-
purpose="billboard-title">New to Udemy? Lucky you.</h1>

<h2 class="udlite-heading-xl top-categories--title--261i0">Top categories</h2>

<h2 class="udlite-heading-xl trending-topics--section-title--3UH9I">Featured to
pics by category</h2>

<h2 class="udlite-heading-md trending-topics--title--kvhmu" data-purpose="categ
ory-title">Development</h2>

<h2 class="udlite-heading-md trending-topics--title--kvhmu" data-purpose="categ
ory-title">Business</h2>

<h2 class="udlite-heading-md trending-topics--title--kvhmu" data-purpose="categ
ory-title">IT and Software</h2>

<h2 class="udlite-heading-md trending-topics--title--kvhmu" data-purpose="categ
ory-title">Design</h2>

<h3 class="udlite-heading-serif-xl non-student-cta__header">Become an instructo
r</h3>

<h3 class="udlite-heading-serif-lg partners__title">
Trusted by companies of all sizes
</h3>

<h3 class="udlite-heading-serif-xl non-student-cta__header">Transform your life
through education</h3>
```

In [1]:
```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.wikipedia.org")
bsh =BeautifulSoup(html,'html.parser')
titles=bsh.find_all(['h1','h2','h3'])
print("ALL HEADER TAGS: ",*titles, sep='\n\n')
```

```
ALL HEADER TAGS:

<h1 class="central-textlogo-wrapper">
<span class="central-textlogo__image sprite svg-Wikipedia_wordmark">
Wikipedia
</span>
<strong class="jsl10n localized-slogan" data-jsl10n="portal.slogan">The Free En
cyclopedia</strong>
</h1>

<h2 class="bookshelf-container">
<span class="bookshelf">
<span class="text">
<bdi dir="ltr">
1 000 000+
</bdi>
<span class="jsl10n" data-jsl10n="entries">
articles
</span>
</span>
</span>
</h2>

<h2 class="bookshelf-container">
<span class="bookshelf">
<span class="text">
<bdi dir="ltr">
100 000+
</bdi>
<span class="jsl10n" data-jsl10n="portal.entries">
articles
</span>
</span>
</span>
</h2>

<h2 class="bookshelf-container">
<span class="bookshelf">
<span class="text">
<bdi dir="ltr">
10 000+
</bdi>
<span class="jsl10n" data-jsl10n="portal.entries">
articles
</span>
</span>
</span>
</h2>

<h2 class="bookshelf-container">
```

```
<span class="bookshelf">
<span class="text">
<bdi dir="ltr">
1 000+
</bdi>
<span class="jsl10n" data-jsl10n="portal.entries">
articles
</span>
</span>
</span>
</h2>

<h2 class="bookshelf-container">
<span class="bookshelf">
<span class="text">
<bdi dir="ltr">
100+
</bdi>
<span class="jsl10n" data-jsl10n="portal.entries">
articles
</span>
</span>
</span>
</h2>
```

In [2]:
```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen("https://www.facebook.com")
bsh = BeautifulSoup(html,'html.parser')
titles = bsh.find_all(['h1','h2','h3'])
print("ALL HEADER TAGS: ",*titles,sep='\n\n')
```

```
ALL HEADER TAGS:

<h2 class="_8eso">Facebook helps you connect and share with the people in your
life.</h2>
```

In [3]:
```
#Write a Python program to extract and display all the image links
from urllib.request import urlopen
from bs4 import BeautifulSoup
import re
html = urlopen("https://en.wikipedia.org/wiki/Peter_Jeffrey_(RAAF_officer)")
bsh = BeautifulSoup(html,'html.parser')
images = bsh.find_all('img',{'src':re.compile('.jpg')})
for image in images:
    print(image['src']+'\n')
```

```
//upload.wikimedia.org/wikipedia/commons/thumb/a/af/NlaJeffrey1942-43.jpg/220px
-NlaJeffrey1942-43.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/c/c5/008315JeffreyTurnbull1941.j
pg/260px-008315JeffreyTurnbull1941.jpg

//upload.wikimedia.org/wikipedia/commons/e/ea/021807CameronJeffrey1941.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/9/92/AC0072JeffreyTruscottKittyh
awks1942.jpg/280px-AC0072JeffreyTruscottKittyhawks1942.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/2/26/VIC1689Jeffrey1945.jpg/280p
x-VIC1689Jeffrey1945.jpg
```

In [ ]:
```
```

In [6]:
```
from urllib.request import urlopen
from bs4 import BeautifulSoup
import re
html = urlopen("https://en.wikipedia.org/wiki/Visakhapatnam")
bsh = BeautifulSoup(html,'html.parser')
images = bsh.find_all('img',{'src':re.compile('.jpg')})
for image in images:
    print(image['src']+'\n')
```

```
//upload.wikimedia.org/wikipedia/commons/thumb/7/7f/Vizag_View_from_Kailasagir
i.jpg/288px-Vizag_View_from_Kailasagiri.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/1/12/Vizag_seaport.jpg/109px-Viz
ag_seaport.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/2/22/Varaha_Lakshmi_Narasimha_te
mple_in_Simhachalam.jpg/109px-Varaha_Lakshmi_Narasimha_temple_in_Simhachalam.jp
g

//upload.wikimedia.org/wikipedia/commons/thumb/4/44/Visakhapatnam_beach_road_fr
om_Kailsagiri_hill.jpg/61px-Visakhapatnam_beach_road_from_Kailsagiri_hill.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/5/55/Ramanaidu_studios.jpg/73px-
Ramanaidu_studios.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/8/84/Novotel%2C_Vizag.jpg/110px-
Novotel%2C_Vizag.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/9/9c/Vizag_submarine_museum.jpg/
97px-Vizag_submarine_museum.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/f/fe/Vizagcity.jpg/142px-Vizagci
ty.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/7/7f/Kambalakonda_wildlife_sanct
uary_Landscape.jpg/142px-Kambalakonda_wildlife_sanctuary_Landscape.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/6/62/Yaksha_Sculpture_Relief_at_
Pavurallakonda_Buddhist_Remnant_Site_near_Bheemunipatnam.jpg/220px-Yaksha_Sculp
ture_Relief_at_Pavurallakonda_Buddhist_Remnant_Site_near_Bheemunipatnam.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/1/1a/Rock-cut_Lord_--Buddha--_St
atue_at_Bojjanakonda_near_Anakapalle_of_Visakhapatnam_dist_in_AP.jpg/220px-Rock
-cut_Lord_--Buddha--_Statue_at_Bojjanakonda_near_Anakapalle_of_Visakhapatnam_di
st_in_AP.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/c/c0/Mahastupa_in_Thotlakonda%2C
_Visakhapatnam_%282%29.jpg/220px-Mahastupa_in_Thotlakonda%2C_Visakhapatnam_%28
2%29.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/0/05/Boats_in_Kondakarla_ava_2.j
pg/220px-Boats_in_Kondakarla_ava_2.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/c/c6/Vizag_Steel.jpg/220px-Vizag
_Steel.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/7/7f/Vizag_View_from_Kailasagir
```

```
i.jpg/300px-Vizag_View_from_Kailasagiri.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/b/bc/INS_Kursura_%28S20%29.jpg/2
20px-INS_Kursura_%28S20%29.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/4/42/Beautiful_view_of_Visakhapa
tnam_and_Bay_of_Bengal_from_Tenneti_park_1.jpg/220px-Beautiful_view_of_Visakhap
atnam_and_Bay_of_Bengal_from_Tenneti_park_1.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/6/64/Love_Vizag_near_TUV_aircraf
t_museum.jpg/220px-Love_Vizag_near_TUV_aircraft_museum.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/c/c6/Simhachalam-temple-2_big.jp
g/220px-Simhachalam-temple-2_big.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/1/1a/Visakhapatnam_Highway_Servi
ce.jpg/220px-Visakhapatnam_Highway_Service.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/4/4a/Visakhapatnam_railway_stati
on.jpg/220px-Visakhapatnam_railway_station.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/e/e2/Vizag_airport_terminal_full
_view.jpg/220px-Vizag_airport_terminal_full_view.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/8/81/St_Aloysius_Anglo_Indian_Hi
gh_School_%28SAS%29_established_in_1847_in_Visakhapatnam%2C_Andhra_Pradesh.jpg/
220px-St_Aloysius_Anglo_Indian_High_School_%28SAS%29_established_in_1847_in_Vis
akhapatnam%2C_Andhra_Pradesh.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/f/f0/INS_Karmuk_P64_at_Visakhapa
tnam.jpg/220px-INS_Karmuk_P64_at_Visakhapatnam.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/9/96/Vizagacavdca.jpg/220px-Viza
gacavdca.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/d/df/ENC_Cricket_Team_which_won_
the_VDCA_Institutional_League_Cricket_Championship_2015-16.jpg/220px-ENC_Cricke
t_Team_which_won_the_VDCA_Institutional_League_Cricket_Championship_2015-16.jpg

//upload.wikimedia.org/wikipedia/commons/thumb/2/25/Sarvepalli_Radhakrishnan_19
67_stamp_of_India.jpg/150px-Sarvepalli_Radhakrishnan_1967_stamp_of_India.jpg
```

In [ ]:
```
#Write a Python program to check whether a page contains a title or not
```

In [ ]:
```

```

In [ ]:
```

```