web-scraping

March 18, 2024

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

[2]:
```
from urllib.request import urlopen
from bs4 import BeautifulSoup
```

[3]:
```
url = "http://www.hubertiming.com/results/2017GPTR10K"
html = urlopen(url)
```

[4]:
```
soup = BeautifulSoup(html, 'lxml')
type(soup)
```

[4]:
```
bs4.BeautifulSoup
```

[5]:
```
# Get the title
title = soup.title
print(title)
```

```
<title>Race results for the 2017 Intel Great Place to Run \ Urban Clash
Games!</title>
```

[6]:
```
soup.find_all('a')
```

[6]:
```
[<a href="mailto:timing@hubertiming.com">timing@hubertiming.com</a>,
 <a href="https://www.hubertiming.com">Huber Timing Home</a>,
 <a class="btn btn-primary btn-lg" href="/results/2017GPTR" role="button"
style="margin: 0px 0px 5px 5px"><i aria-hidden="true" class="fa fa-user"></i> 5K
Individual</a>,
 <a class="btn btn-primary btn-lg" href="/results/team/2017GPTR" role="button"
style="margin: 0px 0px 5px 5px"><i aria-hidden="true" class="fa fa-users"></i>
5K Team</a>,
 <a class="btn btn-primary btn-lg" href="/results/team/2017GPTR10K"
role="button" style="margin: 0px 0px 5px 5px"><i aria-hidden="true" class="fa
fa-users"></i> 10K Team</a>,
 <a class="btn btn-primary btn-lg" href="/results/summary/2017GPTR10K"
```

```
role="button" style="margin: 0px 0px 5px 5px"><i class="fa fa-stream"></i>
Summary</a>,
 <a id="individual" name="individual"></a>,
 <a data-url="/results/2017GPTR10K" href="#tabs-1" id="rootTab" style="font-
size: 18px">10K Results</a>,
 <a href="https://www.hubertiming.com/"><img height="65" src="https://www.hubert
iming.com//sites/all/themes/hubertiming/images/clockWithFinishSign_small.png"
width="50"/>Huber Timing</a>,
 <a href="https://facebook.com/hubertiming/"><img
src="https://www.hubertiming.com/results/FB-f-Logo__blue_50.png"/></a>,
 <a class="small" id="bestFeatureEver" style="color:#007bff">Dark Mode</a>]
```

[7]:
```
all_links = soup.find_all("a")
for link in all_links:
    print(link.get("href"))
```

```
mailto:timing@hubertiming.com
https://www.hubertiming.com
/results/2017GPTR
/results/team/2017GPTR
/results/team/2017GPTR10K
/results/summary/2017GPTR10K
None
#tabs-1
https://www.hubertiming.com/
https://facebook.com/hubertiming/
None
```

[8]:
```
rows = soup.find_all('tr')
print(rows[:10])
```

```
[<tr colspan="2">
<b>10K:</b>
</tr>, <tr>
<td>Finishers:</td>
<td>577</td>
</tr>, <tr>
<td>Male:</td>
<td>414</td>
</tr>, <tr>
<td>Female:</td>
<td>163</td>
</tr>, <tr class="header">
<th>Place</th>
<th>Bib</th>
<th>Name</th>
<th>Gender</th>
<th>City</th>
```

```
<th>State</th>
<th>Time</th>
<th>Gun Time</th>
<th>Team</th>
</tr>, <tr data-bib="814">
<td>1</td>
<td>814</td>
<td>

                    JARED WILSON

                </td>
<td>M</td>
<td>TIGARD</td>
<td>OR</td>
<td>36:21</td>
<td>36:24</td>
<td></td>
</tr>, <tr data-bib="573">
<td>2</td>
<td>573</td>
<td>

                    NATHAN A SUSTERSIC

                </td>
<td>M</td>
<td>PORTLAND</td>
<td>OR</td>
<td>36:42</td>
<td>36:45</td>
<td>
<img class="lazy teamThumbs" data-src="/teamLogoThumbnail/logo?teamName=INTEL%20
TEAM%20F&amp;amp;raceId=1251&amp;amp;state=OR"/>
                            INTEL TEAM F
                        </td>
</tr>, <tr data-bib="687">
<td>3</td>
<td>687</td>
<td>

                    FRANCISCO MAYA

                </td>
<td>M</td>
<td>PORTLAND</td>
<td>OR</td>
<td>37:44</td>
```

```
<td>37:48</td>
<td></td>
</tr>, <tr data-bib="623">
<td>4</td>
<td>623</td>
<td>

                    PAUL MORROW

                </td>
<td>M</td>
<td>BEAVERTON</td>
<td>OR</td>
<td>38:34</td>
<td>38:37</td>
<td></td>
</tr>, <tr data-bib="569">
<td>5</td>
<td>569</td>
<td>

                    DEREK G OSBORNE

                </td>
<td>M</td>
<td>HILLSBORO</td>
<td>OR</td>
<td>39:21</td>
<td>39:24</td>
<td>
<img class="lazy teamThumbs" data-src="/teamLogoThumbnail/logo?teamName=INTEL%20
TEAM%20F&amp;amp;raceId=1251&amp;amp;state=OR"/>
                            INTEL TEAM F
                        </td>
</tr>]
```

[9]:
```
for row in rows:
    row_td = row.find_all('td')
print(row_td)
type(row_td)
```

```
[<td>577</td>, <td>443</td>, <td>

                    LIBBY B MITCHELL

                </td>, <td>F</td>, <td>HILLSBORO</td>, <td>OR</td>,
<td>1:41:18</td>, <td>1:42:10</td>, <td></td>]
```

[9]:
```
bs4.element.ResultSet
```

[10]:
```
str_cells = str(row_td)
cleantext = BeautifulSoup(str_cells, "lxml").get_text()
print(cleantext)
```

```
[577, 443,

                    LIBBY B MITCHELL

                , F, HILLSBORO, OR, 1:41:18, 1:42:10, ]
```

[11]:
```
import re

list_rows = []
for row in rows:
    cells = row.find_all('td')
    str_cells = str(cells)
    clean = re.compile('<.*?>')
    clean2 = (re.sub(clean, '',str_cells))
    list_rows.append(clean2)
print(clean2)
type(clean2)
```

```
[577, 443,

                    LIBBY B MITCHELL

                , F, HILLSBORO, OR, 1:41:18, 1:42:10, ]
```

[11]:
```
str
```

[12]:
```
df = pd.DataFrame(list_rows)
df.head(10)
```

[12]:

	0
0	[]
1	[Finishers:, 577]
2	[Male:, 414]
3	[Female:, 163]
4	[]
5	[1, 814, \r\n\r\n JARED WIL…
6	[2, 573, \r\n\r\n NATHAN A …
7	[3, 687, \r\n\r\n FRANCISCO…
8	[4, 623, \r\n\r\n PAUL MORR…
9	[5, 569, \r\n\r\n DEREK G O…

[13]:
```
df1 = df[0].str.split(',', expand=True)
df1.head(10)
```

[13]:
```
             0      1                                                  2  \
0           []   None                                               None
1  [Finishers:   577]                                               None
2       [Male:   414]                                               None
3     [Female:   163]                                               None
4           []   None                                               None
5           [1    814   \r\n\r\n                    JARED WILSON\r\n\…
6           [2    573   \r\n\r\n                    NATHAN A SUSTERSI…
7           [3    687   \r\n\r\n                    FRANCISCO MAYA\r\…
8           [4    623   \r\n\r\n                    PAUL MORROW\r\n\r…
9           [5    569   \r\n\r\n                    DEREK G OSBORNE\r…

      3           4     5       6       7  \
0  None        None  None    None    None
1  None        None  None    None    None
2  None        None  None    None    None
3  None        None  None    None    None
4  None        None  None    None    None
5     M      TIGARD    OR   36:21   36:24
6     M    PORTLAND    OR   36:42   36:45
7     M    PORTLAND    OR   37:44   37:48
8     M   BEAVERTON    OR   38:34   38:37
9     M   HILLSBORO    OR   39:21   39:24

                                                    8
0                                                None
1                                                None
2                                                None
3                                                None
4                                                None
5                                                   ]
6   \n\r\n                          INTEL TEAM …
7                                                   ]
8                                                   ]
9   \n\r\n                          INTEL TEAM …
```

[14]:
```
df1[0] = df1[0].str.strip('[')
df1.head(10)
```

[14]:
```
             0      1                                                  2     3  \
0           ]   None                                               None  None
1  Finishers:   577]                                               None  None
2       Male:   414]                                               None  None
3     Female:   163]                                               None  None
```

```
4           ]   None                                                  None  None
5           1    814   \r\n\r\n                  JARED WILSON\r\n\…     M
6           2    573   \r\n\r\n                  NATHAN A SUSTERSI…    M
7           3    687   \r\n\r\n                  FRANCISCO MAYA\r\…    M
8           4    623   \r\n\r\n                  PAUL MORROW\r\n\r…    M
9           5    569   \r\n\r\n                  DEREK G OSBORNE\r…    M

            4     5       6       7  \
0        None  None    None    None
1        None  None    None    None
2        None  None    None    None
3        None  None    None    None
4        None  None    None    None
5      TIGARD    OR   36:21   36:24
6    PORTLAND    OR   36:42   36:45
7    PORTLAND    OR   37:44   37:48
8   BEAVERTON    OR   38:34   38:37
9   HILLSBORO    OR   39:21   39:24

                                                  8
0                                              None
1                                              None
2                                              None
3                                              None
4                                              None
5                                                 ]
6   \n\r\n                          INTEL TEAM …
7                                                 ]
8                                                 ]
9   \n\r\n                          INTEL TEAM …
```

[15]:
```
col_labels = soup.find_all('th')
all_header = []
col_str = str(col_labels)
cleantext2 = BeautifulSoup(col_str, "lxml").get_text()
all_header.append(cleantext2)
print(all_header)
```

```
['[Place, Bib, Name, Gender, City, State, Time, Gun Time, Team]']
```

[16]:
```
df2 = pd.DataFrame(all_header)
df2.head()
```

[16]:
```
                                                   0
0  [Place, Bib, Name, Gender, City, State, Time, …
```

[17]:
```
df3 = df2[0].str.split(',', expand=True)
df3.head()
```

[17]:

	0	1	2	3	4	5	6	7	8
0	[Place	Bib	Name	Gender	City	State	Time	Gun Time	Team]

[18]:
```
frames = [df3, df1]

df4 = pd.concat(frames)
df4.head(10)
```

[18]:

	0	1	2	3	4	5	6	7	8
0	[Place	Bib	Name	Gender	City	State	Time	Gun Time	Team]
0	]	None	None	None	None	None	None	None	None
1	Finishers:	577]	None	None	None	None	None	None	None
2	Male:	414]	None	None	None	None	None	None	None
3	Female:	163]	None	None	None	None	None	None	None
4	]	None	None	None	None	None	None	None	None
5	1	814	\r\n\r\n	JARED WILSON\r\n\…	M	TIGARD	OR	36:21	36:24	\n\r\n
6	2	573	\r\n\r\n	NATHAN A SUSTERSI…	M	PORTLAND	OR	36:42	36:45	INTEL TEAM …
7	3	687	\r\n\r\n	FRANCISCO MAYA\r\…	M	PORTLAND	OR	37:44	37:48	]
8	4	623	\r\n\r\n	PAUL MORROW\r\n\r…	M	BEAVERTON	OR	38:34	38:37	]

[19]:
```
df5 = df4.rename(columns=df4.iloc[0])
df5.head()
```

[19]:

	[Place	Bib	Name	Gender	City	State	Time	Gun Time	Team]
0	[Place	Bib	Name	Gender	City	State	Time	Gun Time	Team]
0	]	None	None	None	None	None	None	None	None
1	Finishers:	577]	None	None	None	None	None	None	None
2	Male:	414]	None	None	None	None	None	None	None
3	Female:	163]	None	None	None	None	None	None	None

[20]:
```
df5.info()
df5.shape
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 583 entries, 0 to 581
Data columns (total 9 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   [Place     583 non-null    object
 1    Bib       581 non-null    object
 2    Name      578 non-null    object
 3    Gender    578 non-null    object
 4    City      578 non-null    object
 5    State     578 non-null    object
 6    Time      578 non-null    object
 7    Gun Time  578 non-null    object
 8    Team]     578 non-null    object
dtypes: object(9)
memory usage: 45.5+ KB
```

[20]: (583, 9)

[ ]:
```

```