In [1]:
```
import requests
from bs4 import BeautifulSoup
```

In [2]:
```
page = requests.get('http://dataquestio.github.io/web-scraping-pages/simple.html'
```

In [3]:
```
print(page.content)
```

```
b'<!DOCTYPE html>\n<html>\n    <head>\n        <title>A simple example page</ti
tle>\n    </head>\n    <body>\n        <p>Here is some simple content for this
page.</p>\n    </body>\n</html>'
```

In [4]:
```
soup = BeautifulSoup(page.content, 'html.parser')
```

In [5]:
```
print(soup.prettify())
```

```
<!DOCTYPE html>
<html>
 <head>
  <title>
   A simple example page
  </title>
 </head>
 <body>
  <p>
   Here is some simple content for this page.
  </p>
 </body>
</html>
```

In [6]:
```
list(soup.children)
```

Out[6]:
```
['html', '\n', <html>
 <head>
 <title>A simple example page</title>
 </head>
 <body>
 <p>Here is some simple content for this page.</p>
 </body>
 </html>]
```

In [7]:
```
html = list(soup.children)[2]
```

In [9]:
```
list(html.children)
```

Out[9]:
```
['\n', <head>
 <title>A simple example page</title>
 </head>, '\n', <body>
 <p>Here is some simple content for this page.</p>
 </body>, '\n']
```

In [10]:
```
body = list(html.children)[3]
```

In [11]:
```
list(body.children)
```

Out[11]:
```
['\n', <p>Here is some simple content for this page.</p>, '\n']
```

In [12]:
```
p = list(body.children)[1]
```

In [13]:
```
p.get_text()
```

Out[13]:
```
'Here is some simple content for this page.'
```

Finding all instances of a tag at once

In [14]:
```
soup.findAll('p')
```

Out[14]:
```
[<p>Here is some simple content for this page.</p>]
```

In [16]:
```
soup.find('p').get_text()
```

Out[16]:
```
'Here is some simple content for this page.'
```

Searching for tags by class and id

In [19]:
```
page = requests.get("http://dataquestio.github.io/web-scraping-pages/ids_and_clas
soup = BeautifulSoup(page.content, 'html.parser')
print(soup)
```

```
<html>
<head>
<title>A simple example page</title>
</head>
<body>
<div>
<p class="inner-text first-item" id="first">
                First paragraph.
            </p>
<p class="inner-text">
                Second paragraph.
            </p>
</div>
<p class="outer-text first-item" id="second">
<b>
                First outer paragraph.
            </b>
</p>
<p class="outer-text">
<b>
                Second outer paragraph.
            </b>
</p>
</body>
</html>
```

In [22]:
```
soup.find_all('p', class_='outer-text')
```

Out[22]:
```
[<p class="outer-text first-item" id="second">
 <b>
                 First outer paragraph.
             </b>
 </p>, <p class="outer-text">
 <b>
                 Second outer paragraph.
             </b>
 </p>]
```

In [23]:
```
soup.find_all(class_="outer-text")
```

Out[23]:
```
[<p class="outer-text first-item" id="second">
 <b>
                 First outer paragraph.
             </b>
 </p>, <p class="outer-text">
 <b>
                 Second outer paragraph.
             </b>
 </p>]
```

In [24]:
```
soup.find_all(id="first")
```

Out[24]:
```
[<p class="inner-text first-item" id="first">
                 First paragraph.
             </p>]
```

Using CSS Selectors

In [25]:
```
soup.select("div p")
```

Out[25]:
```
[<p class="inner-text first-item" id="first">
                 First paragraph.
             </p>, <p class="inner-text">
                 Second paragraph.
             </p>]
```

Downloading weather data

In [26]:
```
page = requests.get("http://forecast.weather.gov/MapClick.php?lat=37.7772&lon=-12
soup = BeautifulSoup(page.content, 'html.parser')
seven_day = soup.find(id="seven-day-forecast")
forecast_items = seven_day.find_all(class_="tombstone-container")
tonight = forecast_items[0]
print(tonight.prettify())
```

```
<div class="tombstone-container">
 <p class="period-name">
  Overnight
  <br/>
  <br/>
 </p>
 <p>
  <img alt="Overnight: Mostly cloudy, with a low around 51. West wind 11 to 14
mph. " class="forecast-icon" src="newimages/medium/nbkn.png" title="Overnight:
Mostly cloudy, with a low around 51. West wind 11 to 14 mph. "/>
 </p>
 <p class="short-desc">
  Mostly Cloudy
 </p>
 <p class="temp temp-low">
  Low: 51 °F
 </p>
</div>
```

Extracting information from the page

In [27]:
```
period = tonight.find(class_="period-name").get_text()
short_desc = tonight.find(class_="short-desc").get_text()
temp = tonight.find(class_="temp").get_text()

print(period)
print(short_desc)
print(temp)
```

```
Overnight
Mostly Cloudy
Low: 51 °F
```

In [28]:
```
img = tonight.find("img")
desc = img['title']

print(desc)
```

```
Overnight: Mostly cloudy, with a low around 51. West wind 11 to 14 mph.
```

Extracting all the information from the page

In [30]:
```
period_tags = seven_day.select(".tombstone-container .period-name")
periods = [pt.get_text() for pt in period_tags]
periods
```

Out[30]:
```
['Overnight',
 'Tuesday',
 'TuesdayNight',
 'Wednesday',
 'WednesdayNight',
 'Thursday',
 'ThursdayNight',
 'Friday',
 'FridayNight']
```

In [31]:
```
short_descs = [sd.get_text() for sd in seven_day.select(".tombstone-container .sh
temps = [t.get_text() for t in seven_day.select(".tombstone-container .temp")]
descs = [d["title"] for d in seven_day.select(".tombstone-container img")]

print(short_descs)
print(temps)
print(descs)
```

```
['Mostly Cloudy', 'Partly Sunny', 'Mostly Cloudy', 'ChanceShowers', 'ShowersLik
ely', 'Slight ChanceShowers thenMostly Sunny', 'Slight ChanceShowers', 'Rain Li
kely', 'Rain Likely']
['Low: 51 °F', 'High: 61 °F', 'Low: 51 °F', 'High: 59 °F', 'Low: 50 °F', 'High:
60 °F', 'Low: 51 °F', 'High: 57 °F', 'Low: 51 °F']
['Overnight: Mostly cloudy, with a low around 51. West wind 11 to 14 mph. ', 'T
uesday: Partly sunny, with a high near 61. West wind 6 to 14 mph, with gusts as
high as 18 mph. ', 'Tuesday Night: Mostly cloudy, with a low around 51. West so
uthwest wind 7 to 10 mph. ', 'Wednesday: A 30 percent chance of showers after 1
0am.  Mostly cloudy, with a high near 59. West southwest wind around 6 mph beco
ming calm  in the morning.  New precipitation amounts of less than a tenth of a
n inch possible. ', 'Wednesday Night: Showers likely, mainly before 4am.  Mostl
y cloudy, with a low around 50. West southwest wind 3 to 5 mph.  Chance of prec
ipitation is 60%. New precipitation amounts between a quarter and half of an in
ch possible. ', 'Thursday: A 20 percent chance of showers before 10am.  Mostly
sunny, with a high near 60. New precipitation amounts of less than a tenth of a
n inch possible. ', 'Thursday Night: A 20 percent chance of showers.  Mostly cl
oudy, with a low around 51.', 'Friday: Rain likely.  Cloudy, with a high near 5
7.', 'Friday Night: Rain likely.  Mostly cloudy, with a low around 51.']
```

Combining our data into a Pandas Dataframe

In [32]:
```
import pandas as pd
weather = pd.DataFrame({
        "period": periods,
        "short_desc": short_descs,
        "temp": temps,
        "desc":descs
    })
weather
```

Out[32]:

	period	short_desc	temp	desc
0	Overnight	Mostly Cloudy	Low: 51 °F	Overnight: Mostly cloudy, with a low around 51...
1	Tuesday	Partly Sunny	High: 61 °F	Tuesday: Partly sunny, with a high near 61. We...
2	TuesdayNight	Mostly Cloudy	Low: 51 °F	Tuesday Night: Mostly cloudy, with a low aroun...
3	Wednesday	ChanceShowers	High: 59 °F	Wednesday: A 30 percent chance of showers afte...
4	WednesdayNight	ShowersLikely	Low: 50 °F	Wednesday Night: Showers likely, mainly before...
5	Thursday	Slight ChanceShowers thenMostly Sunny	High: 60 °F	Thursday: A 20 percent chance of showers befor...
6	ThursdayNight	Slight ChanceShowers	Low: 51 °F	Thursday Night: A 20 percent chance of showers...
7	Friday	Rain Likely	High: 57 °F	Friday: Rain likely. Cloudy, with a high near...
8	FridayNight	Rain Likely	Low: 51 °F	Friday Night: Rain likely. Mostly cloudy, wit...

In [33]:
```
temp_nums = weather["temp"].str.extract("(?P<temp_num>\d+)", expand=False)
weather["temp_num"] = temp_nums.astype('int')
temp_nums
```

Out[33]:
```
0    51
1    61
2    51
3    59
4    50
5    60
6    51
7    57
8    51
Name: temp_num, dtype: object
```

In [34]:
```
weather["temp_num"].mean()
```

Out[34]:
```
54.55555555555556
```

In [35]:
```
is_night = weather["temp"].str.contains("Low")
weather["is_night"] = is_night
is_night
```

Out[35]:
```
0     True
1    False
2     True
3    False
4     True
5    False
6     True
7    False
8     True
Name: temp, dtype: bool
```

In [36]:
```
weather[is_night]
```

Out[36]:

	period	short_desc	temp	desc	temp_num	is_night
0	Overnight	Mostly Cloudy	Low: 51 °F	Overnight: Mostly cloudy, with a low around 51...	51	True
2	TuesdayNight	Mostly Cloudy	Low: 51 °F	Tuesday Night: Mostly cloudy, with a low aroun...	51	True
4	WednesdayNight	ShowersLikely	Low: 50 °F	Wednesday Night: Showers likely, mainly before...	50	True
6	ThursdayNight	Slight ChanceShowers	Low: 51 °F	Thursday Night: A 20 percent chance of showers...	51	True
8	FridayNight	Rain Likely	Low: 51 °F	Friday Night: Rain likely. Mostly cloudy, wit...	51	True