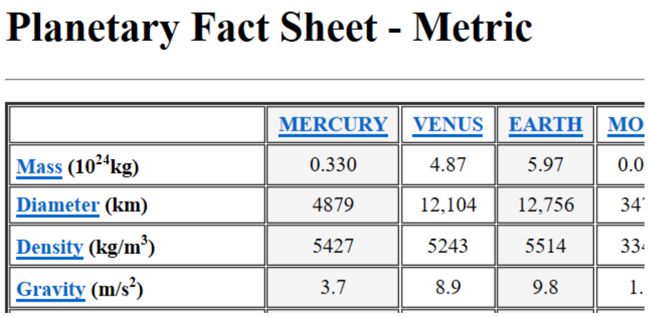
ISTA 350 Scraping Planet Data Worksheet   Name:

What the webpage `'http://nssdc.gsfc.nasa.gov/planetary/factsheet/'` looks like:



Planetary Fact Sheet - Metric

	MERCURY	VENUS	EARTH	MO
Mass (10^{24}kg)	0.330	4.87	5.97	0.0
Diameter (km)	4879	12,104	12,756	34
Density (kg/m^3)	5427	5243	5514	33
Gravity (m/s^2)	3.7	8.9	9.8	1.

The html for the first table row looks like this:

```
<tr>
  <td align=left><b>&nbsp;</b></td>
  <td align=center bgcolor=F5F5F5><b>&nbsp;<a
      href="mercuryfact.html">MERCURY</a>&nbsp;</b></td>
  <td align=center><b>&nbsp;<a
      href="venusfact.html">VENUS</a>&nbsp;</b></td>
  <td align=center bgcolor=F5F5F5><b>&nbsp;<a
      href="earthfact.html">EARTH</a>&nbsp;</b></td>
  ...
</tr>
```

The html for the second row looks like:

```
<tr>
  <td align=left><b><a
      href="planetfact_notes.html#mass">Mass</a>
      (10<sup>24</sup>kg)</b></td>
  <td align=center bgcolor=F5F5F5>0.330</td>
  <td align=center bgcolor=FFFFFF>4.87</td>
  ...
</tr>
```

The rest of the rows follow this pattern except for the last one, which you do not want. Write a function called `scrape_planets` that scrapes this webpage and stores the html in a file called `'planets.html'`.

Write a function called `get_planet_frame` that reads in `planets.html`, turns it into a `BeautifulSoup` object, and returns a `DataFrame` that looks like this:

```
           Mass Diameter Density Gravity Escape Velocity ...
Mercury   0.330     4879    5427     3.7             4.3
Venus      4.87   12,104    5243     8.9            10.4
Earth      5.97   12,756    5514     9.8            11.2
Moon      0.073     3475    3340     1.6             2.4
...
```

Recall your `find_all` method, which returns a list of html elements. Make your column labels from the first row, i.e. `tr` element. Make your row labels from the first `td` in each succeeding `tr`, except the last one. Note that those `td`'s have an `a` element in them that contains the text you want for the row label. You can grab it with the `find` method.