

```
In [10]: from urllib.request import Request, urlopen
from bs4 import BeautifulSoup
import pandas as pd
url = "https://www.daftlogic.com/information-appliance-power-consumption.htm"
url
```

```
Out[10]: 'https://www.daftlogic.com/information-appliance-power-consumption.htm'
```

```
In [14]: headers={'User-Agent': 'Mozilla/5.0'}
```

```
In [15]: req=Request(url=url, headers=headers)
page=urlopen(req).read()
```

```
In [16]: soup = BeautifulSoup(page, 'html.parser')
```

```
In [17]: table = soup.find('table', id="tblApp")
table
<td style="font-weight: bold;"><a name="tablet charger"></a><a search="Ta
let Charger" type="amzn">Tablet Charger</a></td>
<td>10W</td>
<td>15W</td>
<td>N/A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td style="font-weight: bold;"><a name="tablet computer"></a><a search="Ta
blet Computer" type="amzn">Tablet Computer</a></td>
<td>5W</td>
<td>10W</td>
<td>N/A</td>
<td></td>
<td>[<a href="https://discussions.apple.com/thread/4049197?tstart=0" targe
t="_blank">1</a>]</td>
<td></td>
</tr>
...
```

```
In [18]: df = pd.read_html(str(table))[0]
```

```
In [19]: df.head()
```

Out[19]:

	Appliance	Minimum	Maximum	Standby	Other Name(s)	References	Notes
0	100W light bulb (Incandescent)	100W	100W	0W	NaN	[1]	NaN
1	22 Inch LED TV	17W	17W	0.5W	NaN	NaN	NaN
2	25" colour TV	150W	150W	NaN	NaN	NaN	NaN
3	3" belt sander	1000W	1000W	NaN	NaN	NaN	NaN
4	32 Inch LED TV	20W	60W	1W	NaN	NaN	NaN

```
In [31]: df.to_csv("POWER_Consumption.csv")
```

```
In [21]: df
```

Out[21]:

	Appliance	Minimum	Maximum	Standby	Other Name(s)	References	Notes
0	100W light bulb (Incandescent)	100W	100W	0W	NaN	[1]	NaN
1	22 Inch LED TV	17W	17W	0.5W	NaN	NaN	NaN
2	25" colour TV	150W	150W	NaN	NaN	NaN	NaN
3	3" belt sander	1000W	1000W	NaN	NaN	NaN	NaN
4	32 Inch LED TV	20W	60W	1W	NaN	NaN	NaN
...
150	WiFi Booster	1W	2W	NaN	WiFi Repeater,WiFi Extender,Range Extender	NaN	NaN
151	WiFi Router	4W	10W	4W	Router	NaN	NaN
152	Window Air Conditioner	500W	1500W	NaN	Window AC	NaN	NaN
153	Wine cooler (18 bottles)	83W	83W	0W	NaN	[1]	NaN
154	Xbox One	50W	110W	14W	NaN	NaN	NaN

155 rows × 7 columns

Task 2

```
In [24]: df = pd.DataFrame(pd.read_csv('iris.csv'))
df.head()
```

Out[24]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

```
In [26]: df.describe()
```

Out[26]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

```
In [25]: df.groupby('Species').SepalLengthCm.mean()
```

Out[25]: Species
 Iris-setosa 5.006
 Iris-versicolor 5.936
 Iris-virginica 6.588
 Name: SepalLengthCm, dtype: float64

```
In [27]: df.corr().loc['SepalLengthCm', 'PetalLengthCm']
```

Out[27]: 0.8717541573048716

```
In [30]: max_petal = df['PetalLengthCm'].max()
max_petal_spec = df.loc[df['PetalLengthCm'].idxmax(), 'Species']
print(max_petal_spec)
```

Iris-virginica

In []: