

▼ Importing Libraries:

```
import pandas as pd
from bs4 import BeautifulSoup
import numpy as np
import seaborn as sbs
!pip install bs4
from urllib.request import urlopen
import requests
from bs4 import BeautifulSoup
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: bs4 in /usr/local/lib/python3.10/dist-packages (0.0.1)
Requirement already satisfied: beautifulsoup4 in /usr/local/lib/python3.10/dist-packages (from bs4) (4.11.2)
Requirement already satisfied: soupsieve>1.2 in /usr/local/lib/python3.10/dist-packages (from beautifulsoup4->bs4) (2.4.1)
```

TASK#2:

▼ Importing DataSet:

```
file=pd.read_csv("Iris.csv")
file.head()
```

```
file.columns
```

```
Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm',  
      'Species'],  
      dtype='object')
```

```
file.groupby("Species")["SepalLengthCm"].mean()
```

```
Species  
Iris-setosa      5.006  
Iris-versicolor  5.936  
Iris-virginica   6.588  
Name: SepalLengthCm, dtype: float64
```

```
file.SepalLengthCm.corr(file.PetalLengthCm)
```

```
0.8717541573048718
```

```
print(file.groupby("Species")["PetalLengthCm"].max().idxmax(),"Has Largest Petal.")  
X=file.groupby("Species")["PetalLengthCm"].max()  
max=0  
for i in X:  
    if i>max:  
        max=i  
    else:  
        i=max  
print("Value =",max,"\n",X)
```

```
Iris-virginica Has Largest Petal.  
Value = 6.9  
Species  
Iris-setosa      1.9  
Iris-versicolor  5.1  
Iris-virginica   6.9  
Name: PetalLengthCm, dtype: float64
```

TASK#1:

```
link= "https://www.daftlogic.com/information-appliance-power-consumption.htm"
link
```

```
'https://www.daftlogic.com/information-appliance-power-consumption.htm'
```

```
response = requests.get(link)
html_content = response.text
soup = BeautifulSoup(html_content, 'html.parser')
table = soup.find('table', id="tblApp")
print(table)
```

```
<table class="alt" id="tblApp">
<thead>
<tr>
<th>Appliance</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Standby</th>
<th>Other Name(s)</th>
<th>References</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td style="font-weight: bold;"><a name="100w light bulb (incandescent)"></a><a search="100W light bulb (Incandescent)" type="amzn">100W
<td>100W</td>
<td>100W</td>
<td>0W</td>
<td></td>
<td>[<a href="https://en.wikipedia.org/wiki/Incandescent_light_bulb#Electrical_characteristics" target="_blank">1</a>]</td>
<td></td>
</tr>
<tr>
<td style="font-weight: bold;"><a name="22 inch led tv"></a><a search="22 Inch LED TV" type="amzn">22 Inch LED TV</a></td>
<td>17W</td>
<td>17W</td>
```

```

<td>0.5W</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td style="font-weight: bold;"><a name='25" colour tv'></a><a search='25" colour TV' type="amzn">25" colour TV</a></td>
<td>150W</td>
<td>150W</td>
<td>N/A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td style="font-weight: bold;"><a name='3" belt sander'></a><a search='3" belt sander' type="amzn">3" belt sander</a></td>
<td>1000W</td>
<td>1000W</td>
<td>N/A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td style="font-weight: bold;"><a name="32 inch led tv"></a><a search="32 Inch LED TV" type="amzn">32 Inch LED TV</a></td>
<td>20W</td>
<td>60W</td>
<td>1W</td>
<td></td>
<td></td>
<td></td>

```

```

df_list = pd.read_html(str(table))
df = df_list[0]
df.to_csv("DATASET.csv")

```

