CARBO COUNTER : A STEP TOWARDS SUSTAINABILITY

Developing a **carbon footprint calculator** to help individuals as well as organizations to track and reduce their carbon footprint. The main goal is to make our environment sustainable and hence reducing one's carbon footprint is a step towards a more sustainable future. It can also have positive impacts on both environment as well as economy.

With the help of the following functionalities, we will be able to calculate the carbon footprint of an individual as well as an organization.

For an individual:

- **Household** (Electricity, Natural Gas, Heating Oil)
- **Plastic** (Bottles, Packaging, Personal Care Products)
- **Flight**
- **Car**

For an organization:

- ***Electricity***
- ***Plastic usage*** (Packagings)
- ***Mode of transport*** (Car,Bus,Train,Flight)

1. HOUSEHOLD

- This functionality helps us to track an individual's or a family's carbon footprint based on the amount of Electricity, Natural Gas and Heating oil used in a year.
- The average **CO2 emission** of electricity in india is around ***0.8 kgCO2e/KWh***
- The average **CO2 emission** of Natural Gas in india is around ***0.55 kgCO2e/KWh***
- The average **CO2 emission** of Heating Oil in india is around ***3.15 kgCO2e/litres***
- The average Carbon Footprint of a household in India is approximately ***1.8 tons of CO2 equivalent per year.***

```
def Household():
    print("Household Carbon Footprint Calculator:")
    Electricity_Emission_Factor = 0.8
    NaturalGas_Emission_Factor = 0.55
    HeatingOil_Emission_Factor = 3.15

    print("The average electricity emission factor in India is 0.8 kgCO2e/kWh")
    Electricity_Consumption = float(input("Enter your electricity consumption in one month :"))
    print(Electricity_Consumption)
    print("The average Natural Gas emission factor in India is 0.55 kgCO2e/kWh")
    NaturalGas_Consumption = float(input("Enter your natural gas consumption in one month:"))
    print(NaturalGas_Consumption)
    print("The average Heating Oil emission factor in India is 3.15 kgCO2e/litres")
    HeatingOil_Consumption = float(input("Enter your heating oil consumption in one month:"))
    print(HeatingOil_Consumption)

    #formulas
    Electricity_footprint = Electricity_Consumption * Electricity_Emission_Factor * 12
    NaturalGas_footprint = NaturalGas_Consumption * NaturalGas_Emission_Factor * 12
    Heatingoil_footprint = HeatingOil_Consumption * HeatingOil_Emission_Factor * 12
    Annual_Household_Footprint = Electricity_footprint + NaturalGas_footprint + Heatingoil_footprint

    print("The approximate annual carbon footprint is:")
    print(Annual_Household_Footprint)
Household()
```

```
Household Carbon Footprint Calculator:
The average electricity emission factor in India is 0.8 kgCO2e/kWh
Enter your electricity consumption in one month :0.8
0.8
The average Natural Gas emission factor in India is 0.55 kgCO2e/kWh
Enter your natural gas consumption in one month:0.9
0.9
The average Heating Oil emission factor in India is 3.15 kgCO2e/litres
Enter your heating oil consumption in one month:1.5
1.5
The approximate annual carbon footprint is:
70.32
```

2. PLASTIC

- This functionality helps us to track the plastic usage of an individual as well as an organization.

- Plastic usage in - for example: Bottles, plastic packagings (Bags, online delivery packages etc.), Personal care products (Toothbrush) etc.
- The average CO2 emission of plastic in India is 0.8 kgCO2e/kg

```
def Plastic():
    print("Plastic Carbon Footprint Calculator:")
    Plastic_Emission_Factor = 0.8
    print("The average plastic emission factor in India is 0.8 kgCO2e/kg")
    Bottles = int(input("Enter the number of plastic bottles used by you in one month:"))
    print(Bottles)
    Packaging = int(input("Enter the number of plastic packagings used by you in one month:"))
    print(Packaging)
    PersonalCare_Products = int(input("Enter the number of personal care products used by you in one month :"))
    print(PersonalCare_Products)

    Annual_Plastic_Footprint = ((Bottles * Plastic_Emission_Factor * 12) + (Packaging * Plastic_Emission_Factor * 12) + (Pers

    print("The approximate annual carbon footprint is:")
    print(Annual_Plastic_Footprint)
Plastic()
```

```
Plastic Carbon Footprint Calculator:
The average plastic emission factor in India is 0.8 kgCO2e/kg
Enter the number of plastic bottles used by you in one month:4
4
Enter the number of plastic packagings used by you in one month:2
2
Enter the number of personal care products used by you in one month :2
2
The approximate annual carbon footprint is:
76.80000000000001
```

3. FLIGHT

- This functionality helps us to to track the average carbon footprint of an individual according to his number of travels in one year.
- The carbon foot print of a flight is calculated on the basis of the total distance travelled, fuel consumption and the average emission factor
- The average CO2 emission of a flight in India is 0.25 kgCO2e/km

```
def Flight():
    print("Flight Carbon Footprint Calculator:")
    Flight_Emission_Factor = 0.25
    print("The average flight emission factor in India is 0.25 kgCO2e/km")

    Distance = int(input("Enter the distance travelled by you in one year:"))
    print(Distance)

    Annual_Flight_Footprint = Distance * Flight_Emission_Factor
    print("The approximate annual carbon footprint is:")
    print(Annual_Flight_Footprint)
Flight()
```

```
Flight Carbon Footprint Calculator:
The average flight emission factor in India is 0.25 kgCO2e/km
Enter the distance travelled by you in one year:600
600
The approximate annual carbon footprint is:
150.0
```

4. CAR

- This functionality helps us to track the average carbon footprint of an individual according to the amount of usage of his car.
- The carbon footprint of a car can be calculated with the help of various factors such as type of fuel, fuel efficiency, distance covered and the average emission factor.
- It also varies from the type of car model and its manufacturing emission factor.

```
def Car():
    print("Car Carbon Footprint Calculator:")
    CO2_Emission_Factor_Gasoline = 2.3
    CO2_Emission_Factor_Diesel = 2.7

    print(input("Enter your car model:"))
    print("The average CO2 emission of gasoline is 2.3 kgCO2e/litre")
    print("The average CO2 emission of diesel is 2.7 kgCO2e/litre")
    Fuel = input("Enter the type of fuel used by your car:")
    print(Fuel)

    Fuel = float(input("Enter the litres of fuel used by your vehicle in one month"))
    print(Fuel)
    Distance = int(input("Enter the distance travelled by you in one month:"))
    print(Distance)
    Fuel_Efficiency = (Distance) * (Fuel) * 12
    print("The Annual Fuel Efficiency is:")
    print(Fuel_Efficiency)

    if(Fuel == 'Gasoline'):

      CO2_Emission_Factor = 2.3

    else:
      CO2_Emission_factor = 2.7

    print("The average manufacturing emission of a car is 7.5 metric tons of CO2")
    Manufacturing_factor = 7.5

    Annual_Car_Footprint = Fuel_Efficiency + CO2_Emission_factor + Manufacturing_factor

    print("The approximate annual carbon footprint is:")
    print(Annual_Car_Footprint)
Car()
```

```
Car Carbon Footprint Calculator:
Enter your car model:Honda city
Honda city
The average CO2 emission of gasoline is 2.3 kgCO2e/litre
The average CO2 emission of diesel is 2.7 kgCO2e/litre
Enter the type of fuel used by your car:Gasoline
Gasoline
Enter the litres of fuel used by your vehicle in one month8
8.0
Enter the distance travelled by you in one month:20
20
The Annual Fuel Efficiency is:
1920.0
The average manufacturing emission of a car is 7.5 metric tons of CO2
The approximate annual carbon footprint is:
1930.2
```

ORGANISATION

- An organization can be a small business or a large business company.
- The carbon footprint of an organization depends on the following factors -

1. Number of employees,staff,workers etc
2. Total Electricity used
3. Total amount of plastic used
4. Mode of transport (Fuel used)
5. Water
6. Waste

- It is also possible to calculate the carbon footprint of a particular team with, for example: 3 members
- To bring in the innovative factor, the team member/employee with the lowest carbon footprint can be rewareded with bonus points.

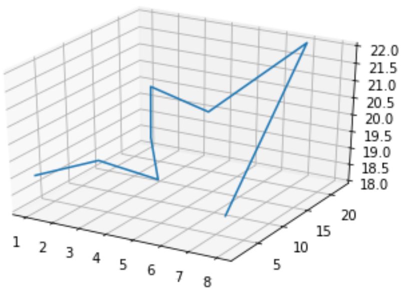
```
def organisation():
    user = int(input("Enter number of users in a team:"))
    print(user)
    print("To calculate the carbon footprint of a team in an organisation")

    print("The average CO2 emission of cars ranges from 140 to 160 grams of CO2 per kilometer (g CO2/km):")
    print("The average CO2 emission of buses ranges from 100 to 400 g CO2/km, depending on the size, fuel type:")
    print("The average CO2 emission of trains ranges from 30 to 90 g CO2/km, depending on the fuel type and efficiency of the
    print("The average CO2 emission of airplanes ranges from 150 to 550 g CO2/km, depending on the fuel efficiency:")
    Transport = str(input("Enter the most used mode of transport at your organisation - Car, Bus, Train, Flight:"))
    print(Transport)
    if(Transport == 'Car'):
      Transport_Emission_Factor = 1.5
    elif(Transport == 'Bus'):
      Transport_Emission_Factor = 2.5
    elif(Transport == 'Train'):
      Transport_Emission_Factor = 0.6
    elif(Transport == 'Flight'):
      Transport_Emission_Factor = 3.5
    else:
      print("Enter Again!")

    print("The average CO2 emission of electricity is 0.8 kgCO2e/KWh")
    Electricity_Emission_Factor = 0.8
    Electricity = float(input("Enter your electricity consumption in one month:"))
    print(Electricity)

    print("The average plastic emission factor in India is 0.8 kgCO2e/kg")
    Plastic_Emission_Factor = 0.8
    Plastic = float(input("Enter the number of plastic packagings used by you in one month:"))
    print(Plastic)

    Annual_Footprint = (Transport_Emission_Factor + (Electricity * Electricity_Emission_Factor) + (Plastic * Plastic_Emission_
    print("The annual carbon footprint of an organisation is:")
    print(Annual_Footprint)
organisation()
```

```
Enter number of users in a team:4
4
To calculate the carbon footprint of a team in an organisation
The average CO2 emission of cars ranges from 140 to 160 grams of CO2 per kilometer (g CO2/km):
The average CO2 emission of buses ranges from 100 to 400 g CO2/km, depending on the size, fuel type:
The average CO2 emission of trains ranges from 30 to 90 g CO2/km, depending on the fuel type and efficiency of the locom
The average CO2 emission of airplanes ranges from 150 to 550 g CO2/km, depending on the fuel efficiency:
Enter the most used mode of transport at your organisation - Car, Bus, Train, Flight:Car
Car
The average CO2 emission of electricity is 0.8 kgCO2e/KWh
Enter your electricity consumption in one month:2.4
2.4
The average plastic emission factor in India is 0.8 kgCO2e/kg
Enter the number of plastic packagings used by you in one month:5
5.0
The annual carbon footprint of an organisation is:
89.03999999999999
```

```
def Result():
    Annual_Household_Footprint = 70.32
    Annual_Plastic_Footprint = 76.80
    Annual_Flight_Footprint = 150.0
    Annual_Car_Footprint = 1930.2
    Total_Carbon_Footprint = (Annual_Household_Footprint + Annual_Plastic_Footprint + Annual_Flight_Footprint + Annual_Car_Fo
    print("Your total average carbon footprint is:") #unit for carbon footprint is ton
    print(Total_Carbon_Footprint)

Result()
```

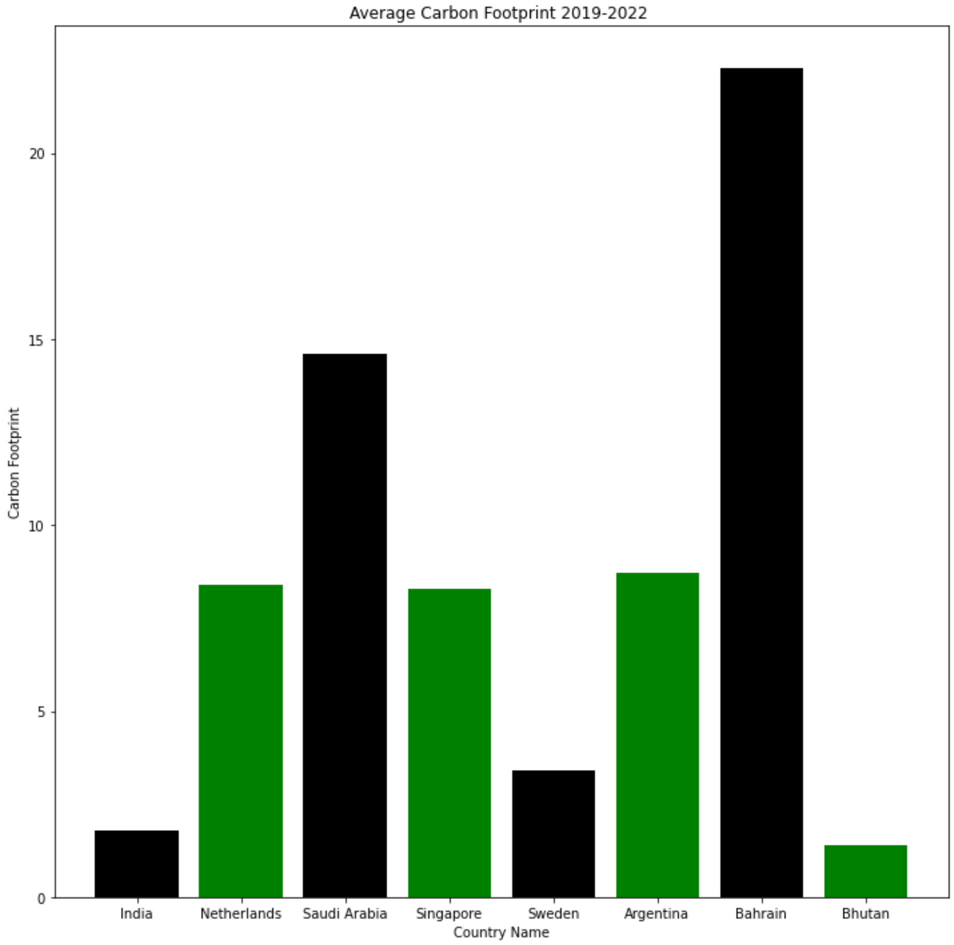
```
Your total average carbon footprint is:
2.22732
```

```
import matplotlib.pyplot as plt

# 1. India
# 2. Netherlands
# 3. Saudi arabia
# 4. Singapore
# 5. Sweden
# 6. Argentina
# 7. Bahrain
# 8. Bhutan

x = [1, 2, 3, 4, 5, 6, 7, 8] #country names
y = [1.8, 8.4,14.6,8.3,3.4,8.7,22.3,1.4] #carbon footprint
z = [19,19,18,20,22,21,22,19] #year

fig = plt.figure()

ax = plt.axes(projection = '3d')
ax.plot3D(x,y,z)
plt.show()
```

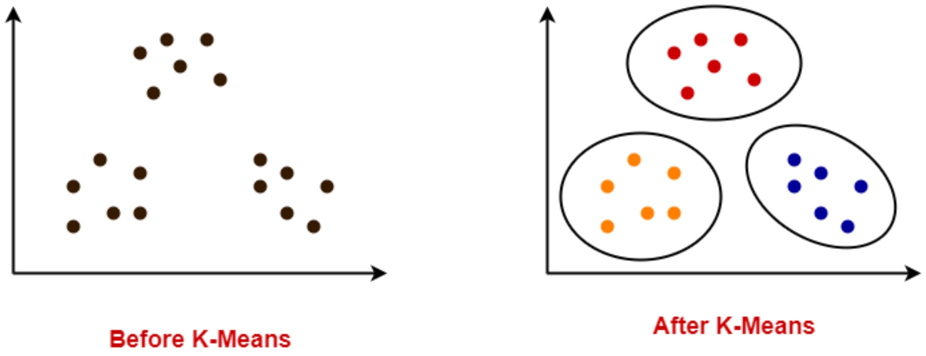


```
import matplotlib.pyplot as plt
plt.figure(figsize=(12,12))
# x-coordinates of left sides of bars
```

```
left = [1, 2, 3, 4, 5, 6, 7, 8]

# heights of bars
height = [1.8,8.4,14.6,8.3,3.4,8.7,22.3,1.4]

# labels for bars
tick_label = ['India', 'Netherlands', 'Saudi Arabia', 'Singapore', 'Sweden', 'Argentina', 'Bahrain', 'Bhutan']

# plotting a bar chart
plt.bar(left, height, tick_label = tick_label,
        width = 0.8, color = ['black', 'green'])

# naming the x-axis
plt.xlabel('Country Name')
# naming the y-axis
plt.ylabel('Carbon Footprint')
# plot title
plt.title('Average Carbon Footprint 2019-2022 ')

# function to show the plot
plt.show()
```



About K means clustering:



K- means cluster is used to identify clusters from a raw data

Step 1:

Identify the number of clusters one would require for a data Assuming number of clusters (k)

Step 2:

Randomly select k points. (assuming they belong to three differet clusters)

Step 3:

With respect to these 3 points, we calculate distance between individual points and these Cluster Centroids

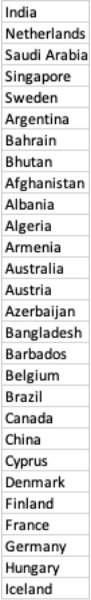
```
#importing libraries
import numpy as np #perform mathematical operations on arrays
import pandas as pd #data analysis
import matplotlib.pyplot as plt #creating graphs
import seaborn as sns #visualise random graphs
from sklearn.cluster import KMeans #to import KMeans
```

```
#loading the data from csv file to a panda dataframe
country_data=pd.read_csv("/content/carbon.csv")
```

```
#checking the shape of the dataset
country_data.shape  #28 rows and 2 coloums --> 28 different observations(countries)

     (28, 2)
```

```
country_data.info()  #getting information of the dataset

     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 28 entries, 0 to 27
     Data columns (total 2 columns):
      #   Column            Non-Null Count  Dtype
     ---  ------            --------------  -----
      0   Country Name      28 non-null     object
      1   Carbon Footprint  28 non-null     float64
     dtypes: float64(1), object(1)
     memory usage: 576.0+ bytes
```

```
#cheking for missing values
country_data.isnull().sum() #issum()--> takes the sum of number of null values
#if there are missing values, we conisder either the previous value,the next value, or the average value to replace missing va

     Country Name        0
     Carbon Footprint    0
     dtype: int64
```



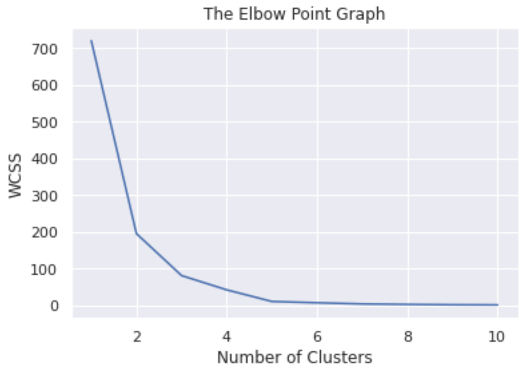
India
Netherlands
Saudi Arabia
Singapore
Sweden
Argentina
Bahrain
Bhutan
Afghanistan
Albania
Algeria
Armenia
Australia
Austria
Azerbaijan
Bangladesh
Barbados
Belgium
Brazil
Canada
China
Cyprus
Denmark
Finland
France
Germany
Hungary
Iceland

```
X= country_data.iloc[:,[1]]
print(X) #in the output printed, the first coloumn is the Annual income and the second coloums in the spendin

         Carbon Footprint
     0                1.8
     1                8.4
     2               14.6
```

```
3                 8.3
4                 3.4
5                 3.7
6                22.3
7                 1.4
8                 0.2
9                 1.7
10                4.0
11                2.2
12               15.3
13                7.3
14                3.5
15                0.5
16                4.5
17                8.1
18                2.1
19               15.4
20                7.6
21                5.9
22                5.1
23                7.4
24                4.5
25                7.9
26                4.7
27                4.5
```

```
 # finding wcss value for different number of clusters
# wcss--> within clusters sum of squares
wcss = []

for i in range(1,11):
  #we take range as 1-10 as we want to find the best number of clusters for the project
  kmeans = KMeans(n_clusters=i, init='k-means++', random_state=42)
   #k-means ++ is a initiation method, there are other methods but kmeans++ is the best model.
  kmeans.fit(X)
  #random state refers to the random way in which the data would be arranges by the compiler before performing operations


  wcss.append(kmeans.inertia_) #gives wcss value
```

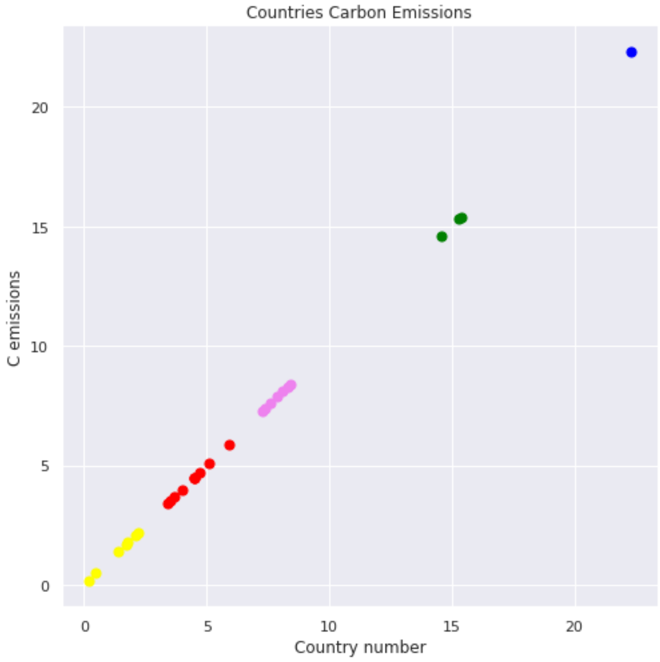
```
# plot an elbow graph. Done to predict the number of clusters

sns.set() #wrt to seaborn library, would allow use of basic parameters for graph
plt.plot(range(1,11), wcss)
plt.title('The Elbow Point Graph')
plt.xlabel('Number of Clusters')
plt.ylabel('WCSS')
plt.show()
```



```
kmeans = KMeans(n_clusters=5, init='k-means++', random_state=0)

# return a label for each data point based on their cluster
Y = kmeans.fit_predict(X)

print(Y)
```

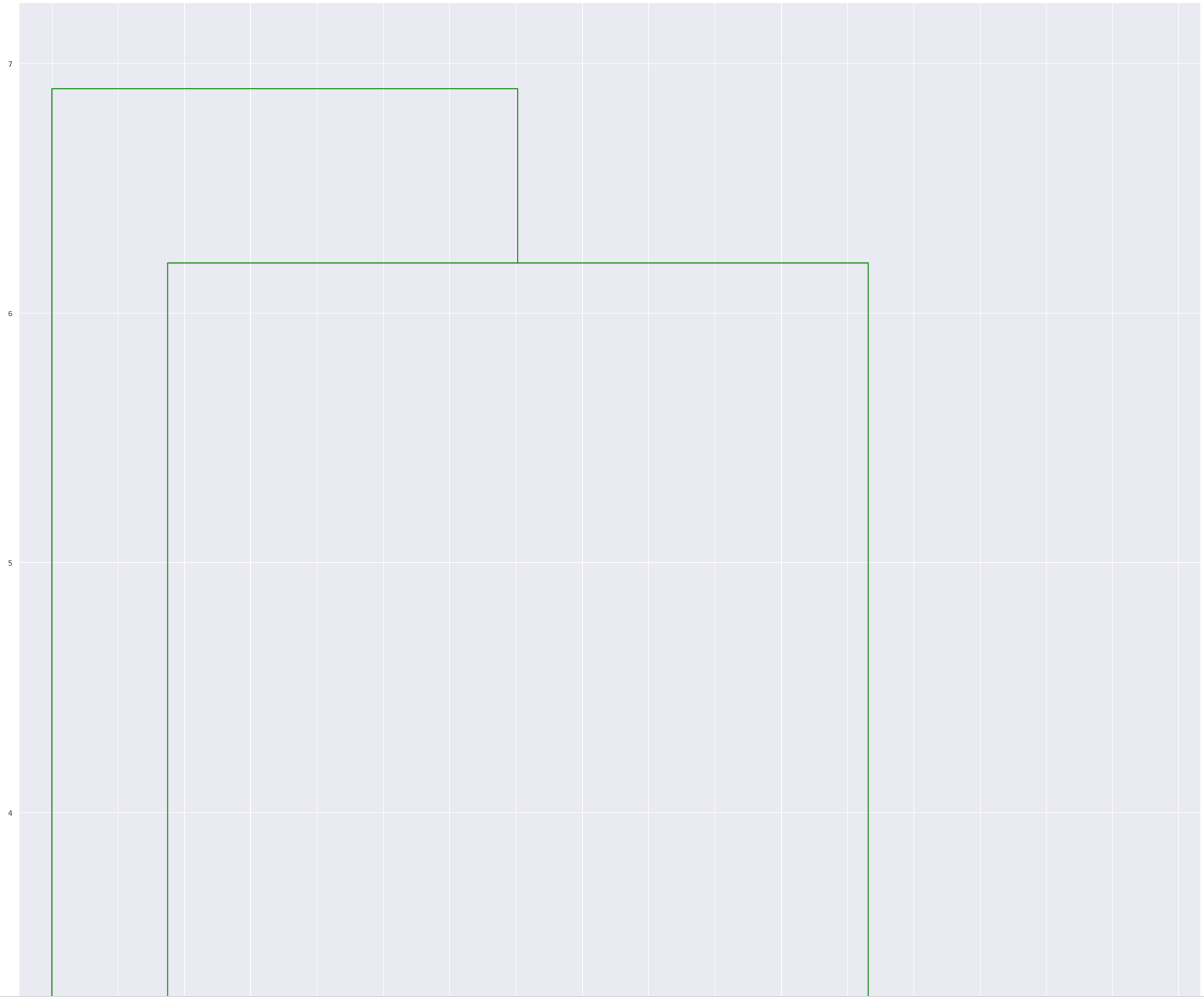
```
[2 3 0 3 1 1 4 2 2 2 1 2 0 3 1 2 1 3 2 0 3 1 1 3 1 3 1 1]
```

```
# plotting all the clusters and their Centroids
# (X[Y==0,0]---> First 0 represents the cluster identity and the second represents the first coloumn of the original data
#X[Y==1,0] ---> First 1 represents the cluster identity and the 2nd zero represents the first coloumn of the original data
# s--> size
plt.figure(figsize=(8,8))
plt.scatter(X[Y==0], X[Y==0], s=50, c='green', label='Cluster 1')
plt.scatter(X[Y==1], X[Y==1], s=50, c='red', label='Cluster 2')
plt.scatter(X[Y==2], X[Y==2], s=50, c='yellow', label='Cluster 3')
plt.scatter(X[Y==3], X[Y==3], s=50, c='violet', label='Cluster 4')
plt.scatter(X[Y==4], X[Y==4], s=50, c='blue', label='Cluster 5')

# plot the centroids
#plt.scatter(kmeans.cluster_centers_[:,0], kmeans.cluster_centers_[:,1], s=100, c='cyan', label='Centroids') # 0 represents x

plt.title('Countries Carbon Emissions')
plt.xlabel('Country number')
plt.ylabel('C emissions')
plt.show()
```



Based on the clusters formed, we can group the countries into 5 main clusters. The Yellow cluster represents those countries that have a very low carbon footprint. The Red cluster mainly consists of countries that have an increasing trend of carbon footrprint. The Pink cluster represents the countries that have violated the permissiable limit of the carbon content in the environment. The Green and the Blue clusters are those countries that have severly violated the carbon limits and need to immediately take strict actions to safeguards humanlife

```
# Plot the dendrogram in vertical orientation
#dendogram shows the way in which clusters are formed.
# Import the python libraries
import numpy as np
from scipy.cluster import hierarchy
import matplotlib.pyplot as plt

temp = hierarchy.linkage(X)
plt.figure(figsize=(50,50))
dn = hierarchy.dendrogram(
    temp, above_threshold_color="green", color_threshold=.7, orientation='top')
```


7
6
5
4