##Use http-based scraping techniques like Beautiful soup library with requests to fetch the

##necessary details from the website

import requests

website\_url = requests.get('https://en.wikipedia.org/wiki/List\_of\_countries\_and\_dependencies\_by\_population').text

from bs4 import BeautifulSoup

soup = BeautifulSoup(website\_url,'lxml')

## Display the extracted html content in proper , neat format

print(soup.prettify())

print('\n')

My\_table = soup.find('table',{'class':'wikitable'})

# My\_table

rows = My\_table.select('tbody > tr')

header = [th.text.rstrip() for th in rows[0].find\_all('th')]

print("#######Header#######")

print(header)

allRows = My\_table.tbody.findAll('tr')

allRows = allRows[1:10]

countries = []

population = []

for row in allRows[1:]:

countries.append(row.find('a').text)

population.append(int(row.findAll('td')[1].text.replace(',','')))

print('\n')

print("\*\*\*\*\*\*\*\*Countries list\*\*\*\*\*\*\*\*\*")

for c in countries:

print(c)

print('\n')

print("\*\*\*\*\*\*\*\*Population List\*\*\*\*\*\*\*\*\*")

for pop in population:

print(pop)

print('\n')

from matplotlib import pyplot as plt

fig = plt.figure(figsize =(10, 7))

plt.legend(population, countries,

title="Countries and Population: A pie",

loc="center left",

bbox\_to\_anchor=(1, 0, 0.5, 1))

## Draw a pie chart for the extracted information

plt.pie(population, labels = countries,autopct='%1.1f%%',pctdistance=1.25, labeldistance=.6)