Question #1. load the dataset csv file as a dataframe using Pandas

In [210…

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import matplotlib
df = pd.read_csv("./world_population.csv")
```

Question#2. compute annual rate of population growth over 50 years from 1970 to 2020. Express the result as percentage rounded to 2 decimal points. What 5 countries had the highest an- nual growth and which had the lowest annual growth?

In [211…

```
#print(df)
from pickle import APPEND


annual_pop_rate={}
annual_pop_rate_avg={}
population_rate_arr=annual_pop_rate_avg_arr=[]
country=df['Country'].unique()
print(len(country))
#print(country)
for i in country:
    temp_d=((df.loc[(df["Country"]==i),:]["2020 Population"].sum()-df.loc[(df["C
    annual_pop_rate[i]=round((temp_d*100),2)
    population_rate_arr.append(round((temp_d*100),2))
    annual_pop_rate_avg[i]=round(((temp_d*100)/50),2)
    #annual_pop_rate_avg_arr.append(round(((temp_d*100)/50),2))
df_hw3_aws=pd.DataFrame(columns=["Country"],data=(country))
df_hw3_aws["Q1_aws"]=population_rate_arr
print(df_hw3_aws)
```

```
234
                Country  Q1_aws
0           Afghanistan  262.43
1               Albania   23.32
2               Algeria  214.96
3        American Samoa   70.60
4               Andorra  291.24
..                  ...     ...
229  Wallis and Futuna   24.29
230     Western Sahara  628.09
231              Yemen  371.74
232             Zambia  342.06
233           Zimbabwe  201.17

[234 rows x 2 columns]
```

Question# 3. compute annual rate of population growth over 10 years from 2010 to 2020. Express the result as percentage rounded to 2 decimal points. What 5 countries had the highest an- nual growth and which had the lowest annual growth? Are the countries the same as in the previous question.

In [212…

```
pop_rise_rate_2020_2010={}
for i in country:
```

```
    temp_d=((df.loc[(df["Country"]==i),:]["2020 Population"].sum()-df.loc[(df["C
    pop_rise_rate_2020_2010[i]=round(((temp_d*100)/50),2)
q3_max=""
q3_max_value=0
for key,value in pop_rise_rate_2020_2010.items():
    if(value == max(pop_rise_rate_2020_2010.values())):
        q3_max=key
        q3_max_value=value
q3_min=""
q3_min_value=0
for key,value in pop_rise_rate_2020_2010.items():
    if(value == min(pop_rise_rate_2020_2010.values())):
        q3_min=key
        q3_min_value=value
print("highest annual rate of population growth over 10 years from 2010 to 2020
print("lowest annual rate of population growth over 10 years from 2010 to 2020 i
```

```
highest annual rate of population growth over 10 years from 2010 to 2020 is Qata
r value is 1.22
lowest annual rate of population growth over 10 years from 2010 to 2020 is Marsh
all Islands value is -0.37
```

Question#4. for 2020 what are the 5 countries with the highest and what are the 5 countries with the lowest population?

In [213…

```
q4_max_5={}
q4_min_5={}

q4_min=df["2020 Population"].sort_values()[0:5].values
q4_max=df["2020 Population"].sort_values()[-5:].values
#print(q4_min,"\n",q4_max)
for i in q4_max:
    #print(type(str(df.loc[(df["2020 Population"]==i),:]["Country"].values)))
    q4_max_5[str(df.loc[(df["2020 Population"]==i),:]["Country"].values)[2:-2]]=
print("5 countries with highest population are",q4_max_5)
for i in q4_min:
    q4_min_5[str(df.loc[(df["2020 Population"]==i),:]["Country"].values)[2:-2]]=
print("5 countries with lowest population are",q4_min_5)
```

```
5 countries with highest population are {'Pakistan': 227196741, 'Indonesia': 271
857970, 'United States': 335942003, 'India': 1396387127, 'China': 1424929781}
5 countries with lowest population are {'Vatican City': 520, 'Tokelau': 1827, 'N
iue': 1942, 'Falkland Islands': 3747, 'Montserrat': 4500}
```

Question#5. for 1970 what are the 5 countries with the highest and what are the 5 countries with the lowest population? Which countries remained in 2022?

In [214…

```
q5_max_5={}
q5_max_5_arr=[]
q5_min_5={}
q5_min_5_arr=[]
q5_min=df["1970 Population"].sort_values()[0:5].values
q5_max=df["1970 Population"].sort_values()[-5:].values
for i in q5_max:
    q5_max_5[str(df.loc[(df["1970 Population"]==i),:]["Country"].values)[2:-2]]=
for i in q5_min:
    q5_min_5[str(df.loc[(df["1970 Population"]==i),:]["Country"].values)[2:-2]]=
for key,value in q5_max_5.items():
```

```
    if key in q4_max_5.keys():
        #print(key)
        q5_max_5_arr.append(key)
for key,value in q5_min_5.items():
    if key in q4_min_5.keys():
        q5_min_5_arr.append(key)
        #print(key)

print("highest 5 countries remained in 2022 are",q5_max_5_arr)
print("lowest 5 countries remained in 2022 are",q5_min_5_arr)
```

```
highest 5 countries remained in 2022 are ['Indonesia', 'United States', 'India',
'China']
lowest 5 countries remained in 2022 are ['Vatican City', 'Tokelau', 'Falkland Is
lands', 'Niue']
```

Question#6. for 2020, compute the mean μ and the quartiles Q1 (25%), Q2 (50% or median M), and Q3 (75%).

In [215…

```
print("for 2020, the median or Q2 is ",df["2020 Population"].iloc[:].median())
print("for 2020, the mean is ",df["2020 Population"].iloc[:].mean())
print("for 2020, the Q1 is ",df["2020 Population"].quantile(0.25))
print("for 2020, the Q3 is",df["2020 Population"].quantile(0.75))
```

```
for 2020, the median or Q2 is  5493074.5
for 2020, the mean is  33501070.952991452
for 2020, the Q1 is  415284.5
for 2020, the Q3 is 21447979.5
```

Question#7. find 3 countries ”around” Q1, M, Q3 and μ

In [216…

```
#print("for 2020, the nearest country of Q1 is ",df["2020 Population"].quantile(
print("for 2020, the nearest country of Q1",str(df.loc[(df["2020 Population"]==d
#print("for 2020, the nearest country of Q2 or median is ",df["2020 Population"]
print("for 2020, the nearest country of Q2 or median is",str(df.loc[(df["2020 Po
#print("for 2020, the nearest country of Q3 is ",df["2020 Population"].quantile(
print("for 2020, the nearest country of Q3 or median is",str(df.loc[(df["2020 Po
print("for 2020, the nearest country of means is ",df["Country"][(df["2020 Popul
```

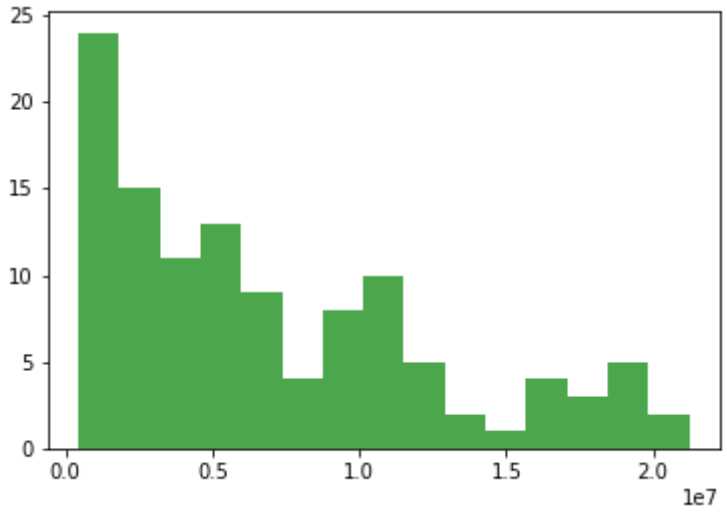
```
for 2020, the nearest country of Q1 Bahamas value is 406471
for 2020, the nearest country of Q2 or median is Slovakia value is 5456681
for 2020, the nearest country of Q3 or median is Burkina Faso value is 21522626
for 2020, the nearest country of means is  Uzbekistan value is 33501070.95299145
2
```

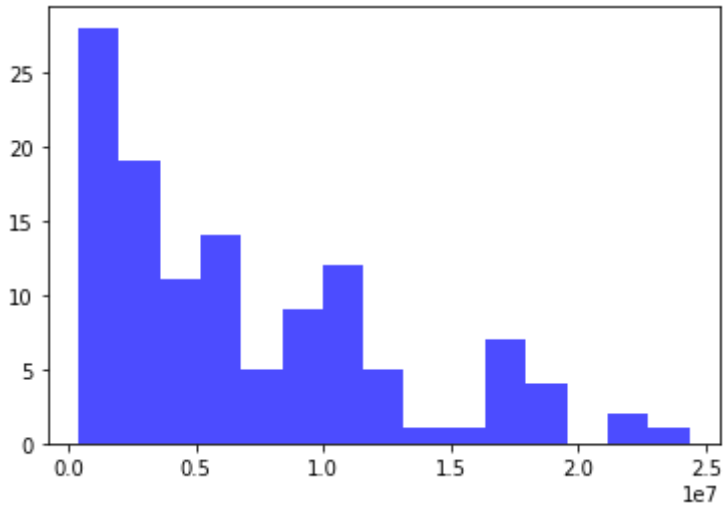
Question#8. consider the columns for years 2020, 2010, 2000, 1990 and 1980. For each columns compute q1 and q3. For each of these years, construct 5 histograms by considering only countries with populations from q1 to q3 for that year. use the following colors: 2020 (in green), 2010 (in blue), 2000 (in cyan), 1990 (in black), and 1980 (in red). Write each histogram to a separate pdf file (with a descriptive name).

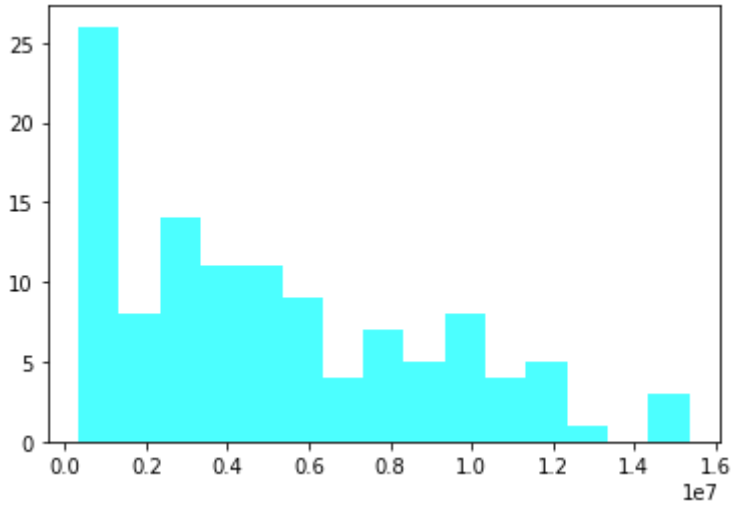
In [217…

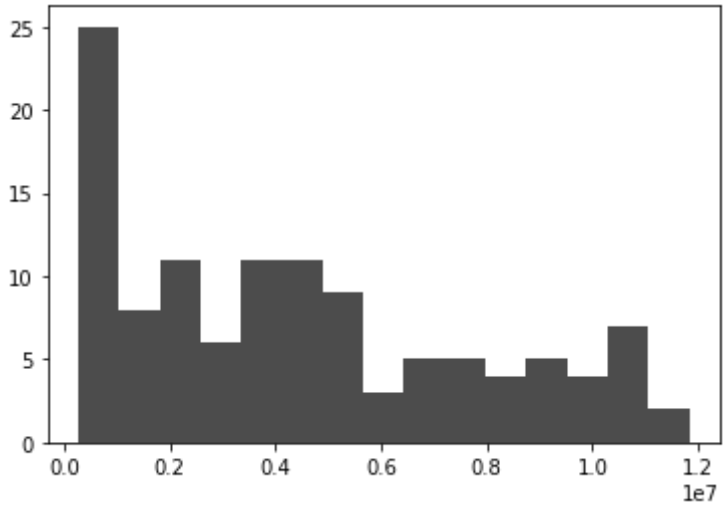
```
Q8_2020_q1=df["2020 Population"].quantile(0.25)
Q8_2020_q3=df["2020 Population"].quantile(0.75)
df_2020=df.sort_values(by="2020 Population",ascending=False)
Q8_2020_arr=df_2020.loc[(df["2020 Population"]<Q8_2020_q3) & (df["2020 Populatio
plt.hist(Q8_2020_arr,color="g", bins=15,alpha=0.7)
plt.show()
```

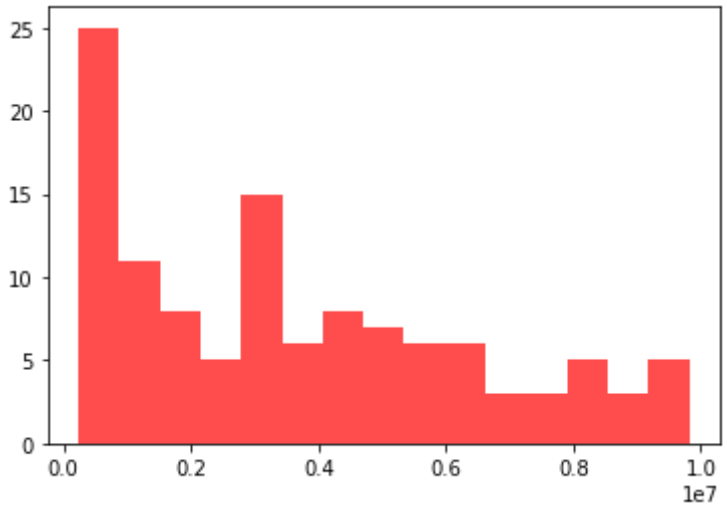
```
#print(Q8_2020_arr)

Q8_2010_q1=df["2010 Population"].quantile(0.25)
Q8_2010_q3=df["2010 Population"].quantile(0.75)
df_2010=df.sort_values(by="2010 Population",ascending=False)
Q8_2010_arr=df_2010.loc[(df["2010 Population"]<Q8_2010_q3) & (df["2020 Populatio
plt.hist(Q8_2010_arr,color="b", bins=15,alpha=0.7)
plt.show()

Q8_2000_q1=df["2000 Population"].quantile(0.25)
Q8_2000_q3=df["2000 Population"].quantile(0.75)
df_2000=df.sort_values(by="2000 Population",ascending=False)
Q8_2000_arr=df_2000.loc[(df["2000 Population"]<Q8_2000_q3) & (df["2000 Populatio
plt.hist(Q8_2000_arr,color="cyan", bins=15,alpha=0.7)
plt.show()

Q8_1990_q1=df["1990 Population"].quantile(0.25)
Q8_1990_q3=df["1990 Population"].quantile(0.75)
df_1990=df.sort_values(by="1990 Population",ascending=False)
Q8_1990_arr=df_1990.loc[(df["1990 Population"]<Q8_1990_q3) & (df["1990 Populatio
plt.hist(Q8_1990_arr,color="black", bins=15,alpha=0.7)
plt.show()

Q8_1980_q1=df["1980 Population"].quantile(0.25)
Q8_1980_q3=df["1980 Population"].quantile(0.75)
df_1980=df.sort_values(by="1980 Population",ascending=False)
Q8_1980_arr=df_1980.loc[(df["1980 Population"]<Q8_1980_q3) & (df["1980 Populatio
plt.hist(Q8_1980_arr,color="red", bins=15,alpha=0.7)
plt.show()
```




25
20
15
10
5
0
0.0
0.5
1.0
1.5
2.0
2.5
1e7


25
20
15
10
5
0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
1e7


25
20
15
10
5
0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1e7



Question#9. examine your histograms. Any interesting observations?

In 1970, the population in Q1 to Q3 in 2010 generally increased, but it did decline from 2010 to 2020

Question#10. compute the rank by population density

In [218…

```
df_density_rank=df.sort_values(by="Density (per km²)",ascending=False)
density_rank=df_density_rank['Country']
print(density_rank)
```

```
119               Macau
134              Monaco
187           Singapore
89            Hong Kong
76            Gibraltar
             ...
141             Namibia
135            Mongolia
230      Western Sahara
64     Falkland Islands
78            Greenland
Name: Country, Length: 234, dtype: object
```

Question#11. compute the rank of countries (by population) in 1970 and compute the difference in rank from 1970 to 2020. Which 5 countries experienced the largest positive change (by value, not percentage) in rank and which 5 countries experiences the largest negative change in rank

In [219…

```
df_1970_rank=df.sort_values(by="1970 Population",ascending=False)
test=list(range(1,235))
#print(test)
df_1970_rank['df_1970_rank']=test

#print(df_1970_rank)
df_2020_rank=df.sort_values(by="2020 Population",ascending=False)
df_2020_rank['df_2020_rank']=test
df_2020_rank.to_csv("asdasd.csv")
Q11_res={}
Q11_res_ans=[]
for i in country:
```

```
    Q11_res[i]=int(df_1970_rank.loc[(df_1970_rank["Country"]==i),:]["df_1970_ran
    Q11_res_ans.append(int(df_1970_rank.loc[(df_1970_rank["Country"]==i),:]["df_

df_q11=pd.DataFrame(columns=['country'],data=(country))
df_q11["Q11_aws"]=Q11_res_ans

q11_min=df_q11["Q11_aws"].sort_values()[0:5].values
q11_max=df_q11["Q11_aws"].sort_values()[-5:].values
q11_min_dict={}
q11_max_dict={}
#print(q11_min,q11_max)
for i in q11_min:
    q11_min_dict[str(df_q11.loc[(df_q11["Q11_aws"]==i),:]["country"].values)]=i
print("The top 5 negtive countries are",q11_min_dict)
for i in q11_max:
    q11_max_dict[str(df_q11.loc[(df_q11["Q11_aws"]==i),:]["country"].values)]=i
print("The five most positive countries are",q11_max_dict)
```

```
The top 5 negtive countries are {"['Georgia']": -47, "['Bulgaria']": -45, "['Hun
gary']": -42, "['Croatia']": -39, "['Belarus' 'Serbia']": -38}
The five most positive countries are {"['Angola']": 33, "['Saudi Arabia']": 34,
"['Qatar']": 40, "['Jordan']": 49, "['United Arab Emirates']": 70}
```

Question#12. take the population in 2020 (all countries). Compute the first digit and compute the % of occurrence of this digit.

In [220…

```
population_2020=df["2000 Population"]
Q12_aws={}
numbers=list(range(1,10))
#print(str(population_2020[0])[0])
for i in numbers:
    count=0
    for j in population_2020:
        if i == int(str(j)[0]):
            count+=1
            Q12_aws[i]=((count/len(df))*100)
print("here is the % of occurrence of this digit. ",Q12_aws)
```

```
here is the % of occurrence of this digit.  {1: 29.48717948717949, 2: 14.9572649
57264957, 3: 11.965811965811966, 4: 10.683760683760683, 5: 8.974358974358974, 6:
8.11965811965812, 7: 5.128205128205128, 8: 8.11965811965812, 9: 2.56410256410256
4}
```

In [ ]: