NAME: CHIRANJEEVI NERELLA **ASSIGNMENT 2** **STUD ID:11442777**

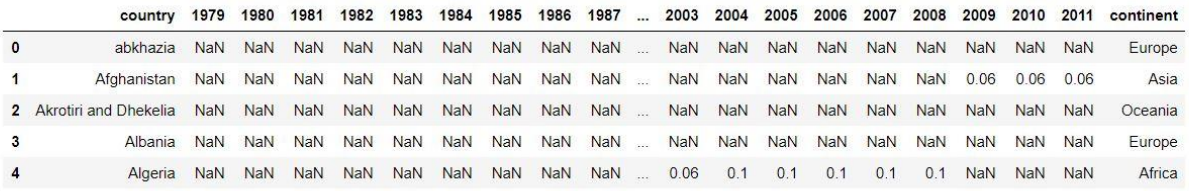
INFO COURSE: 5502 **DATE:02/14/2021**

The included Excel file lists HIV estimated prevalence of people ages from 15 to 49 in the world from1979 to 2011. Use the dataset to complete following tasks:

1) Add one column as "continent" in the dataset and label each country/region in the dataset to anappropriate continent such as "Europe", "Asia", "Africa", "North America", "South America" ,"Australia", or "Antarctica". Explain how do validated the correctness of your labelling. Output the updated dataset as a new CSV file. (1 point). (Note: You must write a Python program to complete the labelling, manually labelling will not get any credit).

Here I imported numpy and pandas these libraries are used to do data analysis by using some built-in functions.
Here pd. read_ excel is used to read the data and renaming the "Estimated HIV Prevalence% - (Ages 15-49)" to "country".

```
import  numpy as  np
import pandas as pd
data= pd. read_ excel('indicator hiv estimated prevalence% 15-49.xlsx')
data.rename(columns={"Estimated HIV Prevalence% - (Ages 15-49)":"country"},inplace=True)
data.head()
```



	country	1979	1980	1981	1982	1983	1984	1985	1986	1987	...	2003	2004	2005	2006	2007	2008	2009	2010	2011	continent
0	abkhazia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe
1	Afghanistan	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	0.06	0.06	0.06	Asia
2	Akrotiri and Dhekelia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Oceania
3	Albania	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe
4	Algeria	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	0.06	0.1	0.1	0.1	0.1	0.1	NaN	NaN	NaN	Africa

Here I have taken countries of all continents in list:

Africa=['Algeria','Angola','Benin','Botswana','Burkina','Burundi','Cameroon','Cape Verde','Central African Republic','Chad','Comoros','Congo','Congo,DemocraticRepublicof','Djibouti','Egypt','EquatorialGuinea','Eritrea','Ethiopia','Gabon','Gambia','Ghana','Guinea','GuineaBissau','IvoryCoast','Kenya','Lesotho','Liberia','Libya','Madagascar','Malawi','Mali','Mauritania','Mauritius','Morocco','Mozambique','Namibia','Niger','Nigeria','Rwanda','Sao Tome and Principe','Senegal','Seychelles','Sierra Leone','Somalia','South Africa','South Sudan','Sudan','Swaziland','Tanzania','Togo','Tunisia','Uganda','Zambia','Zimbabwe',"St. Helena","Western Sahara","Christian","Somaliland","Reunion","Eritrea and Ethiopia","Cote d'Ivoire","Congo", "Rep.","Congo", "Dem. Rep.","Burkina Faso"]

Asia=['Afghanistan','Bahrain','Bangladesh','Bhutan','Brunei','Burma(Myanmar)','Cambodia','China','EastTimor','India','Indonesia','Iran','Iraq','Israel','Japan','Jordan','Kazakhstan','Korea,North','Korea,South','Kuwait','Kyrgyzstan','Laos','Lebanon','Malaysia','Maldives','Mongolia','Nepal','Oman','Pakistan','Philippines','Qatar','RussianFederation','SaudiArabia','Singapore','SriLanka','Syria','Tajikistan','Thailand','Turkey','Turkmenistan','UnitedArabEmirates','Uzbekistan','Vietnam','Yemen',"South Yemen (former)","Coastline","North Yemen (former)","Taiwan","Russia","Timor-Leste","Holy See","india","HongKong","China","Northern Cyprus","Myanmar","Kyrgyz Republic","United Korea (former)\n","South Korea","North Korea","Macao"," China"]

Europe=['Albania','Andorra','Armenia','Austria','Azerbaijan','Belarus','Belgium','BosniaandHerzegovina','Bulgaria','Croatia','Cyprus','CzechRepublic','Denmark','Estonia','Finland','France','Georgia','Germany','Greece','Hungary','Iceland','Ireland','Italy','Latvia','Liechtenstein','Lithuania','Luxembourg','Macedonia','Malta','Moldova','Monaco','Montenegro','Netherlands','Norway','Poland','Portugal','Romania','San Marino','Serbia','Slovakia','Slovenia','Spain','Sweden','Switzerland','Ukraine','United Kingdom','Vatican City',"Kosovo","Jersey","Transnistria","Macedonia", "FYR","Isle of Man","Netherlands Antilles","Micronesia", "Fed. Sts.","St. Martin","Curaçao","SouthOssetia","Saba","SlovakRepublic","St.Barthélemy","Yugoslavia","EastGermany","Czechoslovakia","Åland","Faeroe Islands","Svalbard","Serbia excluding Kosovo","Gibraltar","USSR","West Germany","Serbia andMontenegro","Sint Maarten (Dutch part)","Sark","St. Martin (French part)","abkhazia"]

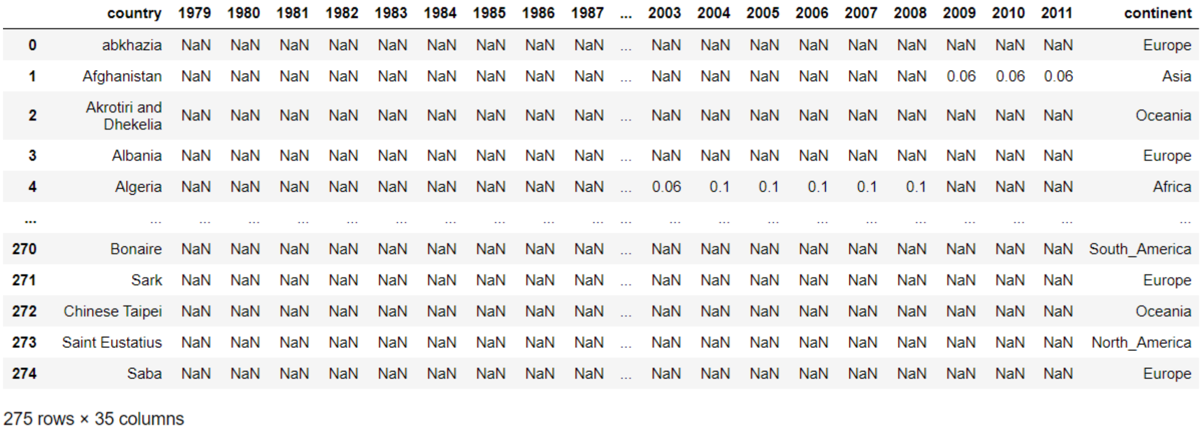
North_America=['Antigua and Barbuda','Bahamas','Barbados','Belize','Canada','Costa Rica','Cuba','Dominica','Dominican Republic','El Salvador','Grenada','Guatemala','Haiti','Honduras','Jamaica','Mexico','Nicaragua','Panama','Saint Kitts and Nevis','Saint Lucia','Saint Vincent and the Grenadines','Trinidad and Tobago','United States',"St.-Pierre-et-Miquelon","St. Lucia","SaintEustatius","TurksandCaicosIslands","U.S.PacificIslands","VirginIslands(U.S.)","PuertoRico","Guadeloupe","Greenland","British Virgin Islands","Bermuda","Martinique"]

South_America=['Argentina','Bolivia','Brazil','Chile','Colombia','Ecuador','Guyana','Paraguay','Peru','Suriname','Uruguay','Venezuela',"Virgin Islands", "British","Bonaire","French Guiana","Falkland Is (Malvinas)","Aruba"]

Australia=['Australia','Fiji','Kiribati','Marshall Islands','Micronesia','Nauru','New Zealand','Palau','Papua New Guinea','Samoa','Solomon Islands','Tonga','Tuvalu','Vanuatu',"Cocos Island","Christmas Island","Norfolk Island","Antartica-Antarctica","New Caledonia"]

```
def getconti(country):
    if country in Africa:
        return "Africa"
    elif country in Asia:
        return "Asia"
    elif country in Europe:
        return "Europe"
    elif country in North_America:
        return "North_America"
    elif country in South_America:
        return "South_America"
    elif country in Australia :
        return "Australia"
    else:
        return "Oceania"
data["continent"]=data["country"].apply(lambda x:getconti(x))
data
```

here I used def function variable name[getconti] and parameter[country].If countries in continents return continent names and assigning to new column in dataframe as continent.



	country	1979	1980	1981	1982	1983	1984	1985	1986	1987	...	2003	2004	2005	2006	2007	2008	2009	2010	2011	continent
0	abkhazia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe
1	Afghanistan	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	0.06	0.06	0.06	Asia
2	Akrotiri and Dhekelia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Oceania
3	Albania	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe
4	Algeria	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	0.06	0.1	0.1	0.1	0.1	0.1	NaN	NaN	NaN	Africa
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
270	Bonaire	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	South_America
271	Sark	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe
272	Chinese Taipei	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Oceania
273	Saint Eustatius	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	North_America
274	Saba	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe

275 rows × 35 columns

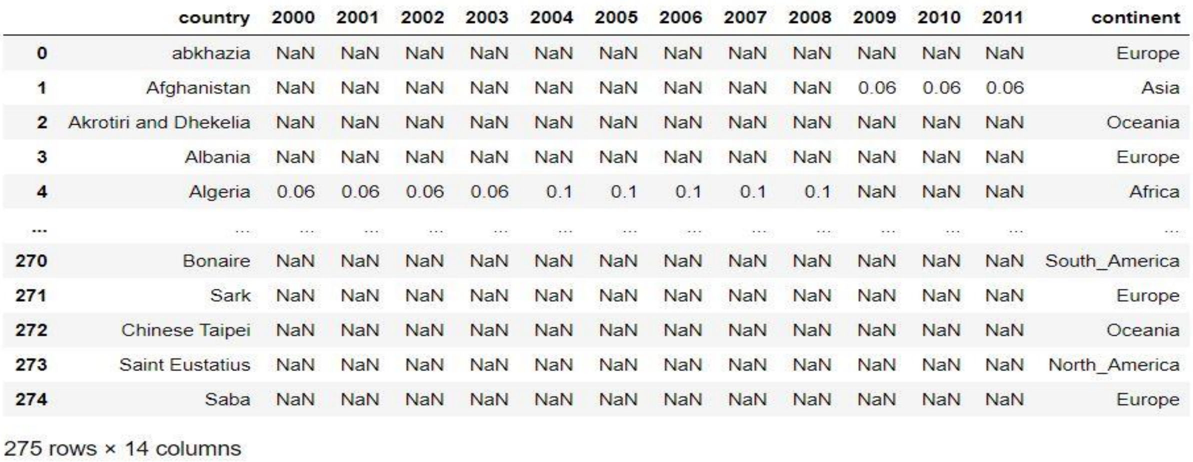
Create new csv file of above dataframe.

```
s=data
s.to_csv("New Data.csv")
```

2)Write a Python program to find the country/region in each continent that has the highest average HIV estimated prevalence of people ages from 15 to 49 of from year 2000 to 2011. Findthe country/region in each continent that has the lowest average HIV estimated prevalence ofpeople ages from 15 to 49 of from year 2000 to 2011. Create a bar chart to show the highestaverage HIV estimated prevalence of people ages from 15 to 49 of from year 2000 to 2011 ineach continent (1 point). Create a bar chart to show the lowest average HIV estimatedprevalence of people ages from 15 to 49 of from year 2000 to 2011 in each continent (1 point).Create an overlaid bar chart to show the highest and lowest average HIV estimated prevalence of people ages from 15 to 49 of from year 2000 to 2011 in each continent (1 point). Select a country/region that is different from the average highest or lowest HIV estimated prevalence of people ages from 15 to 49 from year 2000 to 2011 from each continent, then create an overlaid line chart for the selected country/region, the average highest and lowest HIV estimatedprevalence of people ages from 15 to 49 from year 2000 to 2011 for each continent

```
data1=data.drop(columns=data.iloc[:,1:22])
data1
```

Here collecting data from 2000 t0 2011 by using drop function

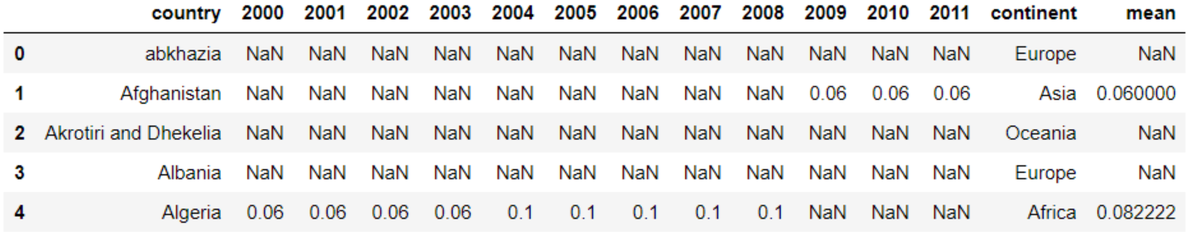


	country	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	continent
0	abkhazia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe
1	Afghanistan	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.06	0.06	0.06	Asia
2	Akrotiri and Dhekelia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Oceania
3	Albania	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe
4	Algeria	0.06	0.06	0.06	0.06	0.1	0.1	0.1	0.1	0.1	NaN	NaN	NaN	Africa
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
270	Bonaire	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	South_America
271	Sark	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe
272	Chinese Taipei	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Oceania
273	Saint Eustatius	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	North_America
274	Saba	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe

275 rows × 14 columns

```
data1 ["mean"]= data1.iloc[:, 1:-1].mean(axis=1)
data1 .head()
```

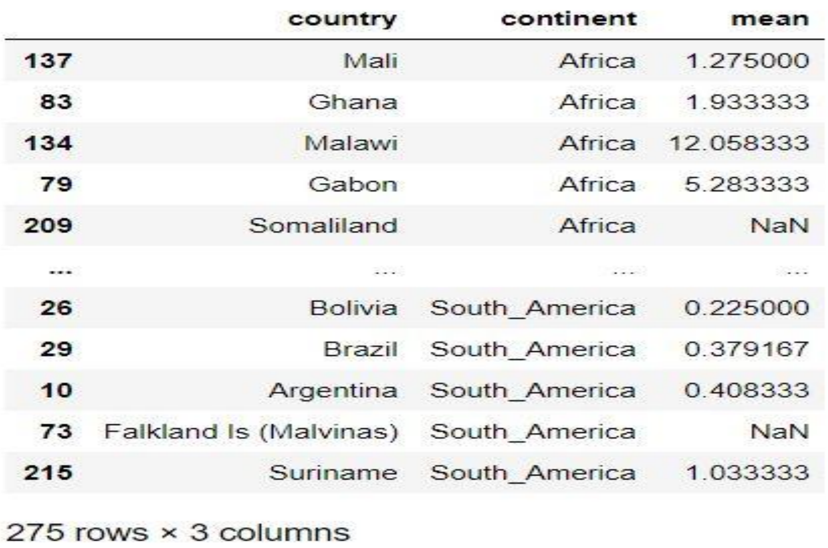
Mean of 2000 to 2011 and adding one column in dataframe as mean



	country	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	continent	mean
0	abkhazia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe	NaN
1	Afghanistan	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.06	0.06	0.06	Asia	0.060000
2	Akrotiri and Dhekelia	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Oceania	NaN
3	Albania	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Europe	NaN
4	Algeria	0.06	0.06	0.06	0.06	0.1	0.1	0.1	0.1	0.1	NaN	NaN	NaN	Africa	0.082222

# sorting continent by name

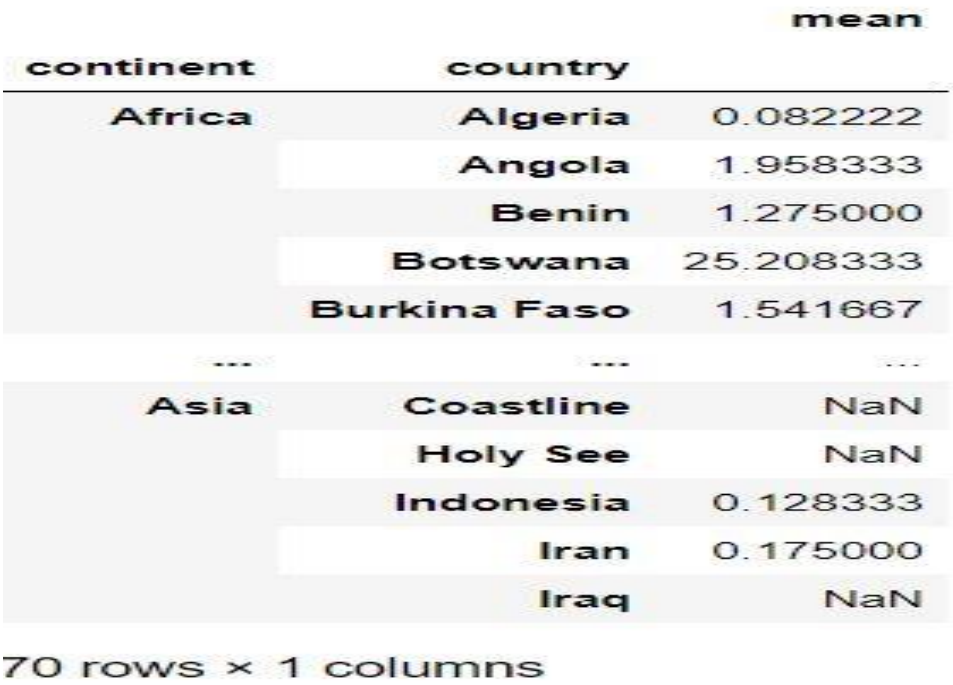
```
data1.sort_values(by="continent",ascending=True,inplace=True)
data1=data1.drop(columns=data1.iloc[:,1:-2])
data1
```



	country	continent	mean
137	Mali	Africa	1.275000
83	Ghana	Africa	1.933333
134	Malawi	Africa	12.058333
79	Gabon	Africa	5.283333
209	Somaliland	Africa	NaN
...	...	...	...
26	Bolivia	South_America	0.225000
29	Brazil	South_America	0.379167
10	Argentina	South_America	0.408333
73	Falkland Is (Malvinas)	South_America	NaN
215	Suriname	South_America	1.033333

275 rows × 3 columns

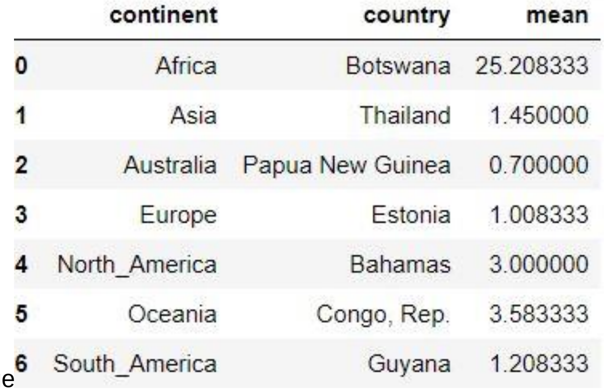
Grouping “continent” ,”country” and “mean” by groupby function



		mean
continent	country	
Africa	Algeria	0.082222
	Angola	1.958333
	Benin	1.275000
	Botswana	25.208333
	Burkina Faso	1.541667
...	...	...
Asia	Coastline	NaN
	Holy See	NaN
	Indonesia	0.128333
	Iran	0.175000
	Iraq	NaN

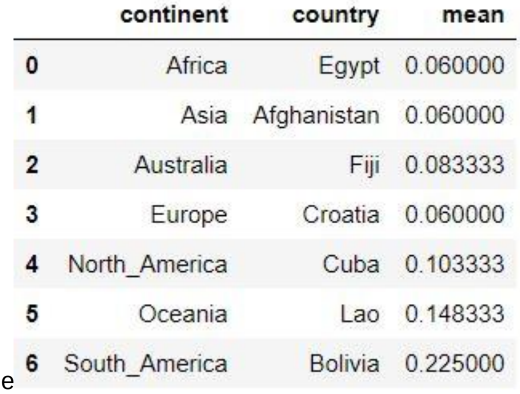
70 rows × 1 columns

```
df=pd.DataFrame(data1.groupby(["continent","country"])["mean"].max())
high_average = df.groupby('continent')['mean'].idxmax()
high_average=df.loc[high_average]
high_average=high_average.reset_index()
high_average
```



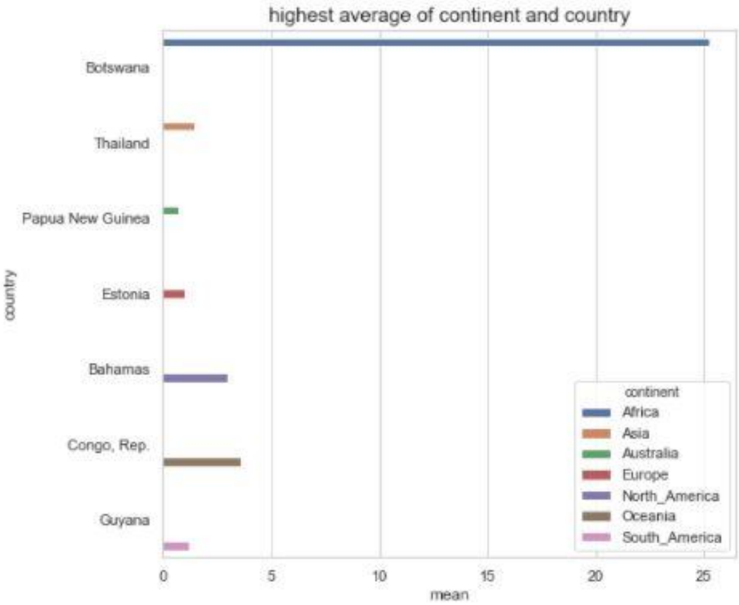
	continent	country	mean
0	Africa	Botswana	25.208333
1	Asia	Thailand	1.450000
2	Australia	Papua New Guinea	0.700000
3	Europe	Estonia	1.008333
4	North_America	Bahamas	3.000000
5	Oceania	Congo, Rep.	3.583333
6	South_America	Guyana	1.208333

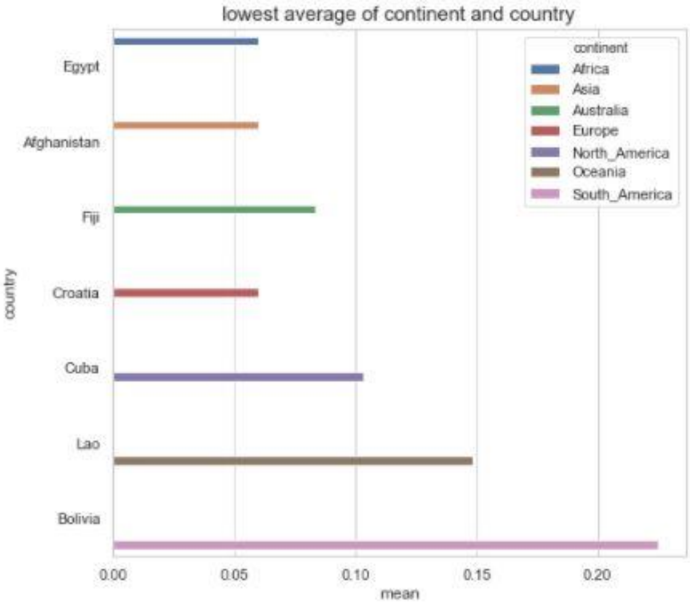
```
low=pd.DataFrame(data1.groupby(["continent","country"])["mean"].min())
low_average=df.groupby('continent')['mean'].idxmin()
low_average=low.l oc[low_average]
low_average=low_average.reset_index()
low_average
```



	continent	country	mean
0	Africa	Egypt	0.060000
1	Asia	Afghanistan	0.060000
2	Australia	Fiji	0.083333
3	Europe	Croatia	0.060000
4	North_America	Cuba	0.103333
5	Oceania	Lao	0.148333
6	South_America	Bolivia	0.225000

```
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(rc={'figure.figsize':(7.7,7.27)})
sns.set(style="whitegrid")
plt.title("highest average of continent and country",size=15)
sns.barplot(x="mean",y="country",hue="continent",data=high_average)
plt.show()
plt.title("lowest average of continent and country",size=15)
sns.barplot(x="mean",y="country",hue="continent",data=low_average)
plt.show()
```

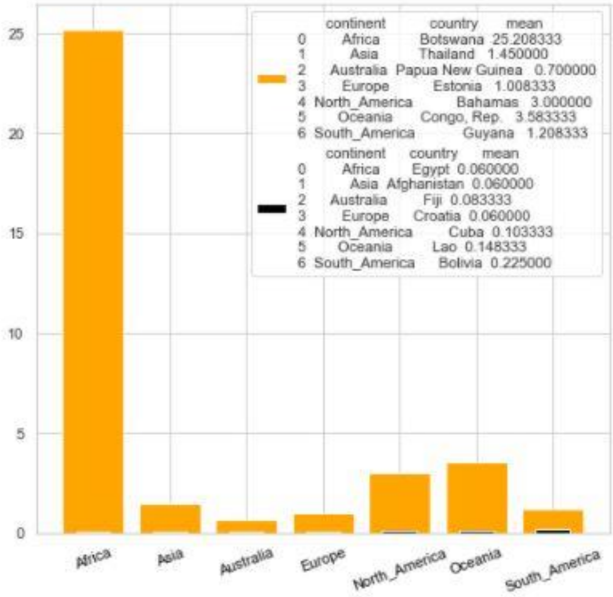




Create an overlaid bar chart to show the highest and lowest average HIV estimated prevalence of people ages from 15 to 49 of from year 2000 to 2011 in each continent

```
s=high_average["continent"]
continent=list(s)
plt.bar(continent, high_average["mean"] ,label=high_average[["continent","country","mean"]],color="orange")
plt.bar(continent,low_average["mean"],width=0.45,label=low_average[["continent","country","mean"]],color="black")
plt.xticks(color='black', rotation=20)
plt.legend()
```

Here continent column is taken into a list and plotting in bar graph .X-axis labelled with continent and y axis is labelled with the highest and lowest average HIV estimated prevalence of people ages from 15 to 49 of from year 2000 to 2011 in each continent and here I used label attribute to view all data in graph and high_average data in orange color and same as low_average and here I added width attribute it is used to compress and increase the bar graph.Xticks is used to customize the x axis data.

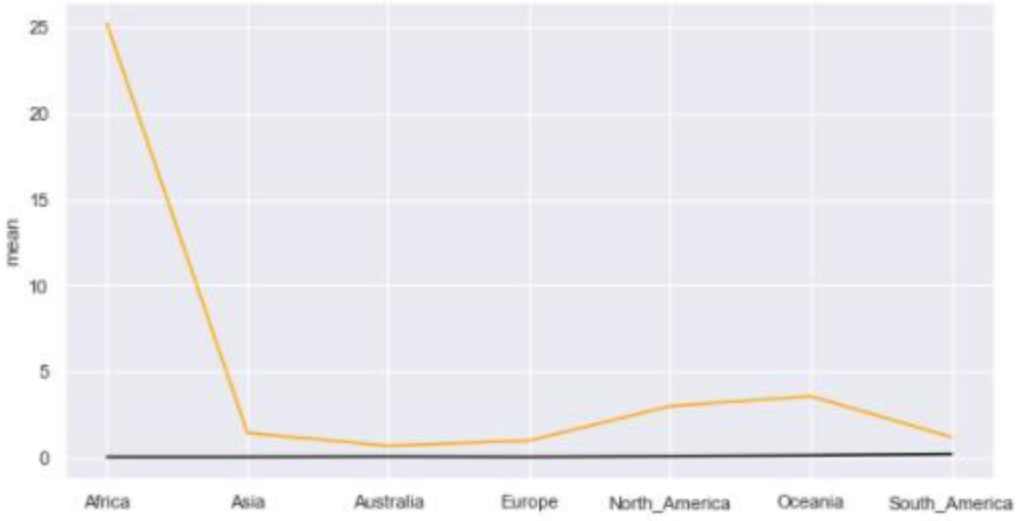


```
       continent           country       mean
0         Africa          Botswana  25.208333
1           Asia          Thailand   1.450000
2      Australia  Papua New Guinea   0.700000
3         Europe           Estonia   1.008333
4  North_America           Bahamas   3.000000
5        Oceania        Congo, Rep.  3.583333
6  South_America            Guyana   1.208333
       continent      country      mean
0         Africa        Egypt  0.060000
1           Asia  Afghanistan  0.060000
2      Australia         Fiji  0.083333
3         Europe      Croatia  0.060000
4  North_America         Cuba  0.103333
5        Oceania          Lao  0.148333
6  South_America      Bolivia  0.225000
```

Select acountry/region that is different from the average highest or lowest HIV estimated prevalence of people ages from 15 to 49 from year 2000 to 2011 from each continent, then create an overlaid line chart for the selected country/region, the average highest and lowest HIV estimated prevalence of people ages from 15 to 49 from year 2000 to 2011 for each continent

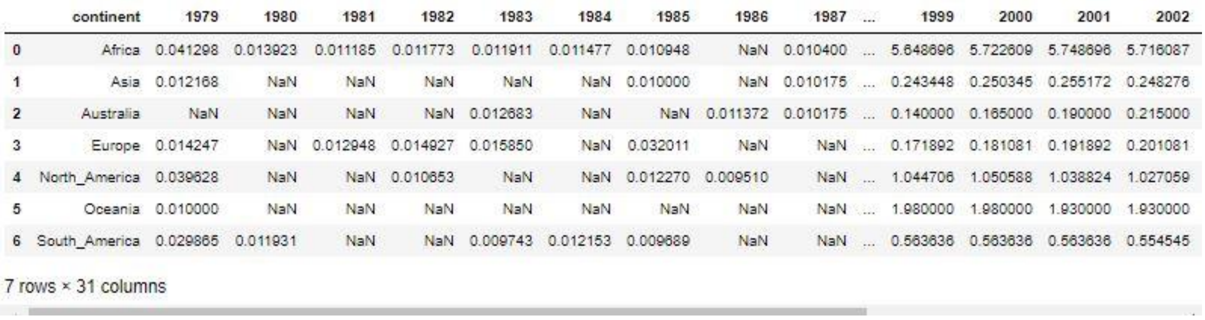
```
sns.set(rc={'figure.figsize':(10,5.27)})

sns.lineplot(continent,high_average["mean"],color="orange")

sns.lineplot(continent,low_average["mean"],color="black")
```

Here I have set function to resize the graph.And I imported seaborn library to plot the line chart.



3) Write a Python program to calculate the average HIV estimated prevalence of people ages from 15 to 49 for each year in the dataset for each continent (you only need simply add the estimate prevalence number of all countries/regions and divided by the number of the countries/regions in the continent). Based on the calculation, create a line chart for each continent to show the changes of the average HIV estimated prevalence from 1979 to 2011 (1 point). Create an overlaid line chart for all continents to show their changes of the average HIV estimated prevalence from 1 1979 to 2011

```
data["mean"]=data.iloc[:, 1:-1].mean(axis=1)

average_year=data.groupby(["continent"])[[x for x in range(1979,2009)]].mean()

average_year=average_year.reset_index()

average_year
```



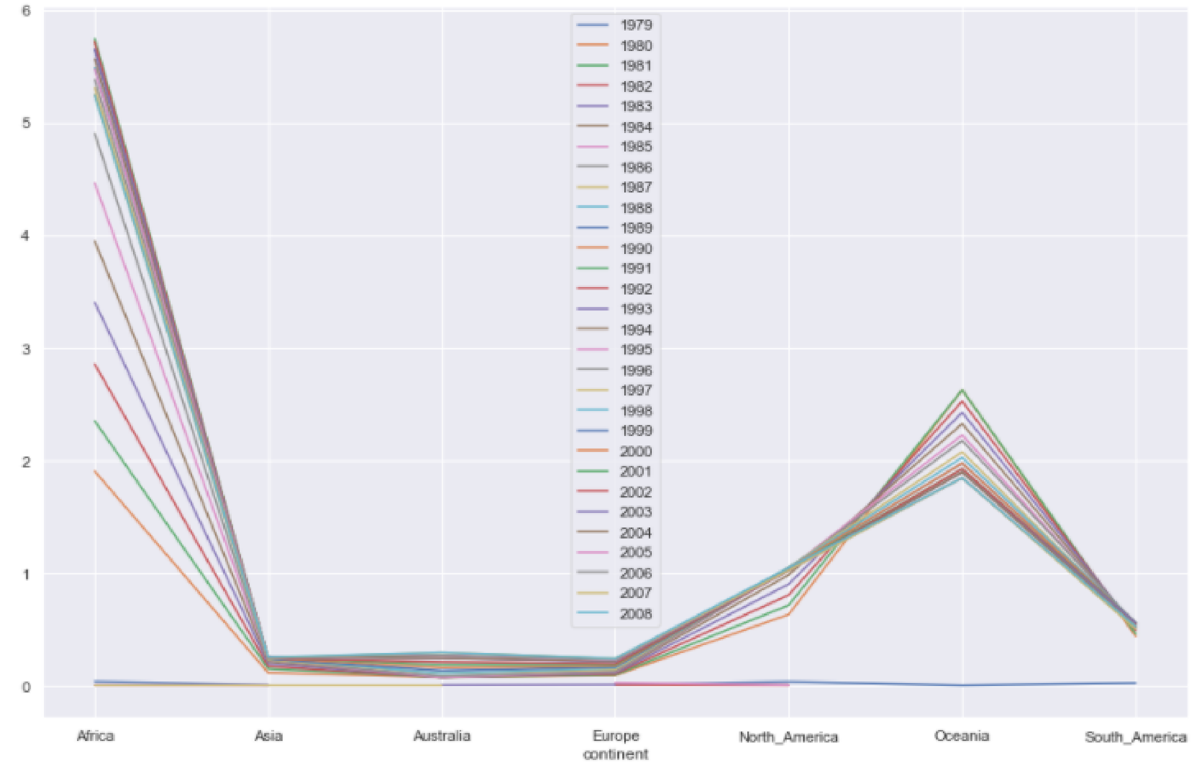
	continent	1979	1980	1981	1982	1983	1984	1985	1986	1987	...	1999	2000	2001	2002
0	Africa	0.041298	0.013923	0.011185	0.011773	0.011911	0.011477	0.010948	NaN	0.010400	...	5.648696	5.722609	5.748696	5.716087
1	Asia	0.012168	NaN	NaN	NaN	NaN	NaN	0.010000	NaN	0.010175	...	0.243448	0.250345	0.255172	0.248276
2	Australia	NaN	NaN	NaN	NaN	0.012683	NaN	NaN	0.011372	0.010175	...	0.140000	0.165000	0.190000	0.215000
3	Europe	0.014247	NaN	0.012948	0.014927	0.015850	NaN	0.032011	NaN	NaN	...	0.171892	0.181081	0.191892	0.201081
4	North_America	0.039628	NaN	NaN	0.010653	NaN	NaN	0.012270	0.009510	NaN	...	1.044706	1.050588	1.038824	1.027059
5	Oceania	0.010000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	1.980000	1.980000	1.930000	1.930000
6	South_America	0.029865	0.011931	NaN	NaN	0.009743	0.012153	0.009689	NaN	NaN	...	0.563636	0.563636	0.563636	0.554545

7 rows × 31 columns

Above program I defind average of each year by using iloc function and I used groupby function to caluculate the average of each year by continent. And reset the index.

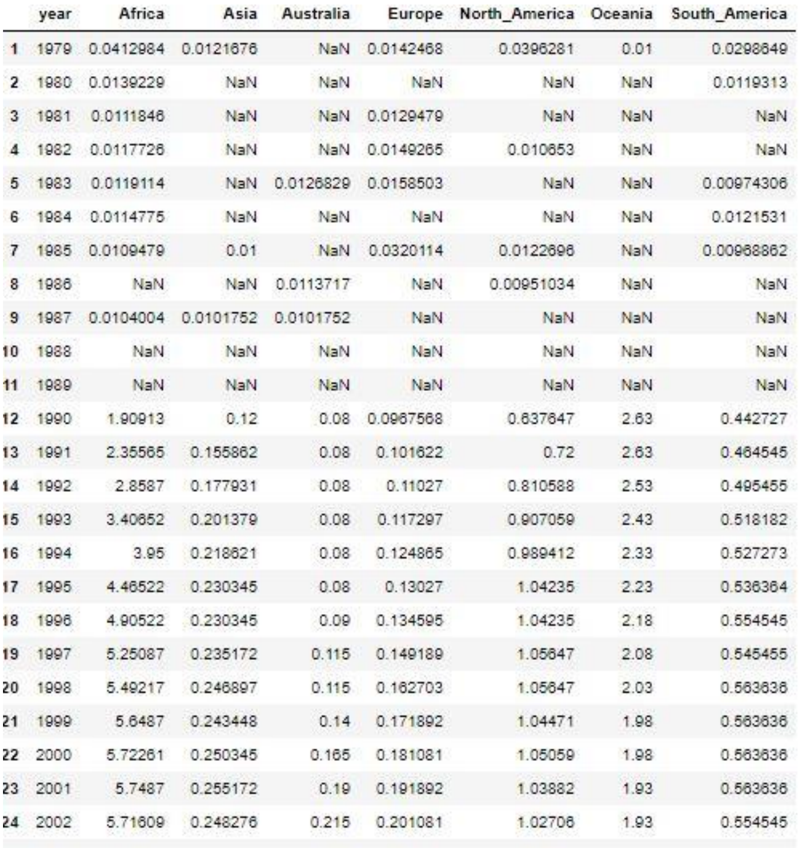
```
average_year.plot(x="continent", y=[i for i in range(1979,2009)], figsize=(15,10), grid=True)
```

Here plotting the line graph x axis is continent and y axis averages of all years

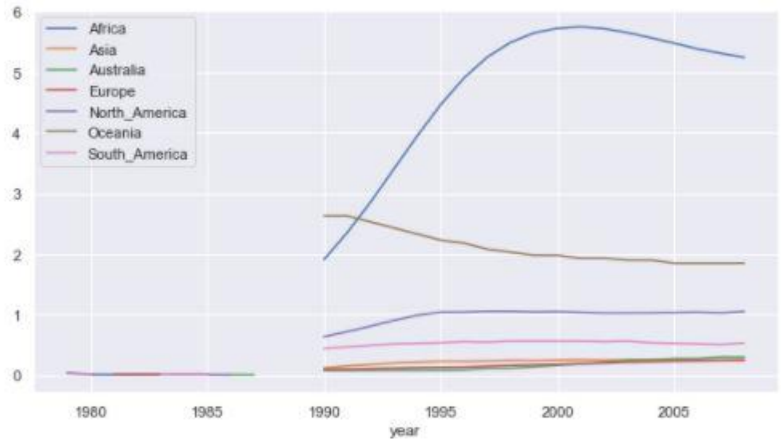


```
s=average_year.T

s.rename(columns=s.iloc[0], inplace = True)

s=s.reset_index()

s.drop(0,inplace=True)

s.rename(columns={"index":"year"},inplace=True)

s.plot(x="year")
```

Here I converted rows and columns by using T function(Transpose function) and reset the index and rename the index column to year after that I plotted the graph



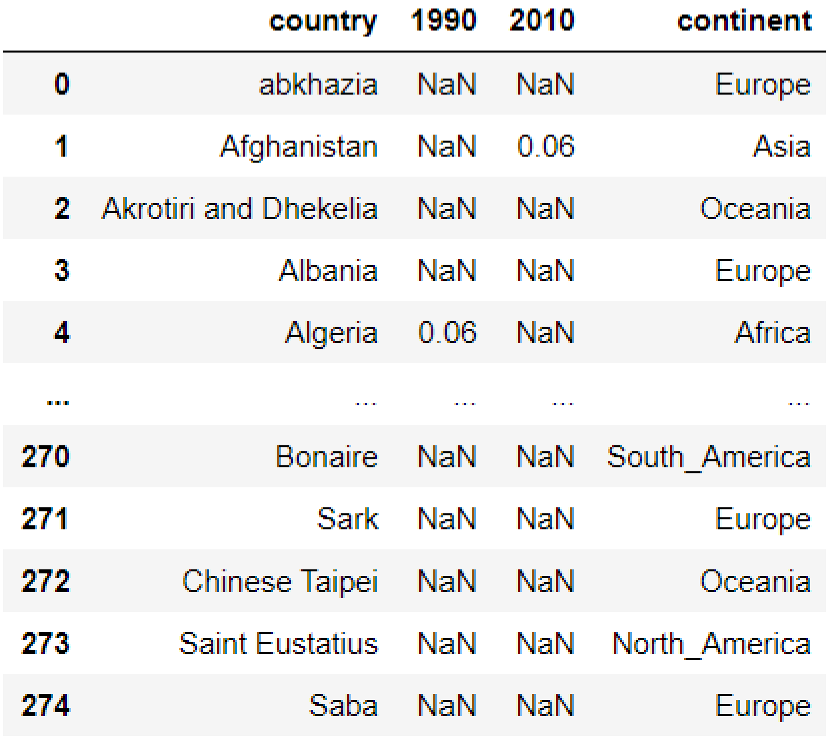
	year	Africa	Asia	Australia	Europe	North_America	Oceania	South_America
1	1979	0.0412984	0.0121676	NaN	0.0142468	0.0396281	0.01	0.0298649
2	1980	0.0139229	NaN	NaN	NaN	NaN	NaN	0.0119313
3	1981	0.0111846	NaN	NaN	0.0129479	NaN	NaN	NaN
4	1982	0.0117726	NaN	NaN	0.0149265	0.010653	NaN	NaN
5	1983	0.0119114	NaN	0.0126829	0.0158503	NaN	NaN	0.00974306
6	1984	0.0114775	NaN	NaN	NaN	NaN	NaN	0.0121531
7	1985	0.0109479	0.01	NaN	0.0320114	0.0122696	NaN	0.00968862
8	1986	NaN	NaN	0.0113717	NaN	0.00951034	NaN	NaN
9	1987	0.0104004	0.0101752	0.0101752	NaN	NaN	NaN	NaN
10	1988	NaN	NaN	NaN	NaN	NaN	NaN	NaN
11	1989	NaN	NaN	NaN	NaN	NaN	NaN	NaN
12	1990	1.90913	0.12	0.08	0.0967568	0.637647	2.63	0.442727
13	1991	2.35565	0.155862	0.08	0.101622	0.72	2.63	0.464545
14	1992	2.8587	0.177931	0.08	0.11027	0.810588	2.53	0.495455
15	1993	3.40652	0.201379	0.08	0.117297	0.907059	2.43	0.518182
16	1994	3.95	0.218621	0.08	0.124865	0.989412	2.33	0.527273
17	1995	4.46522	0.230345	0.08	0.13027	1.04235	2.23	0.536364
18	1996	4.90522	0.230345	0.09	0.134595	1.04235	2.18	0.554545
19	1997	5.25087	0.235172	0.115	0.149189	1.05647	2.08	0.545455
20	1998	5.49217	0.246897	0.115	0.162703	1.05647	2.03	0.563636
21	1999	5.6487	0.243448	0.14	0.171892	1.04471	1.98	0.563636
22	2000	5.72261	0.250345	0.165	0.181081	1.05059	1.98	0.563636
23	2001	5.7487	0.255172	0.19	0.191892	1.03882	1.93	0.563636
24	2002	5.71609	0.248276	0.215	0.201081	1.02706	1.93	0.554545



4) Create two scatter plots to show the data (i.e. each country/region) in year 1990 and year 2010,respectively. The vertical axis in the scatter plot is the HIV estimated prevalence, and the horizontal axis is the corresponding year average HIV estimated prevalence in each continent, which you calculated above. Using different color to show data from different continent

```
v=data.iloc[:,[0,12,32,-2]]

v.head()
```

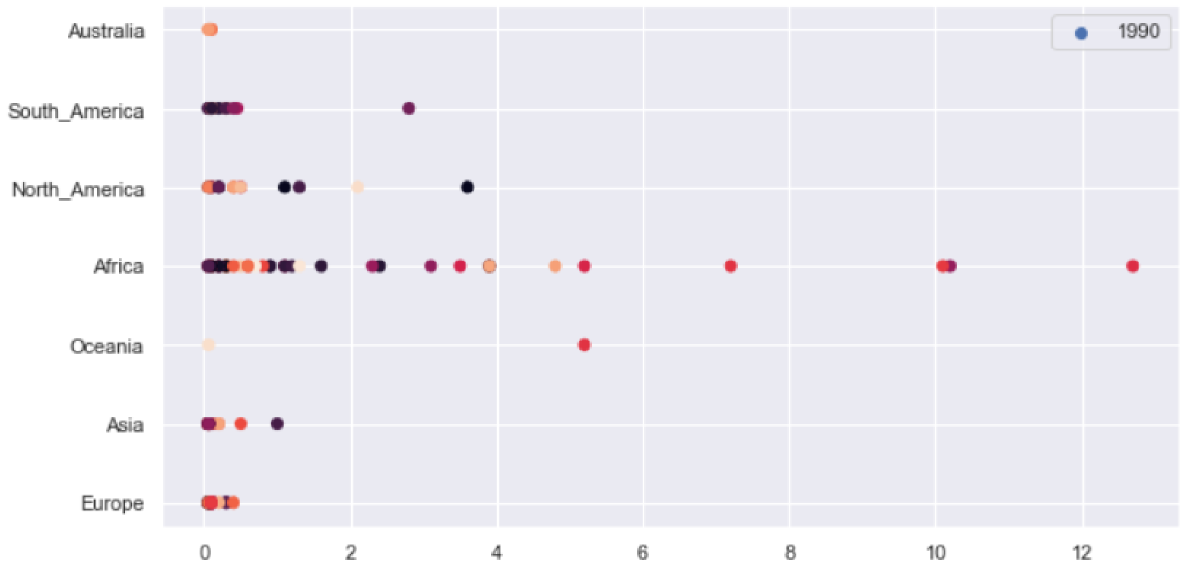
here I just call the country and 1990 ,2010 columns by using iloc function .



	country	1990	2010	continent
0	abkhazia	NaN	NaN	Europe
1	Afghanistan	NaN	0.06	Asia
2	Akrotiri and Dhekelia	NaN	NaN	Oceania
3	Albania	NaN	NaN	Europe
4	Algeria	0.06	NaN	Africa
...	...	...	...	...
270	Bonaire	NaN	NaN	South_America
271	Sark	NaN	NaN	Europe
272	Chinese Taipei	NaN	NaN	Oceania
273	Saint Eustatius	NaN	NaN	North_America
274	Saba	NaN	NaN	Europe

here I have used numpy and random libraries to get different colors of every point in graph and create a scatter plot for every continent in 1990 year by using matplotlib library and x axis is averages and y.axis is continents.

```
colr=np.random.RandomState(0)

colors=colr.rand(275)

plt.scatter(v[1990],"continent",data=v,c=colors,label="1990")

plt.legend()
```

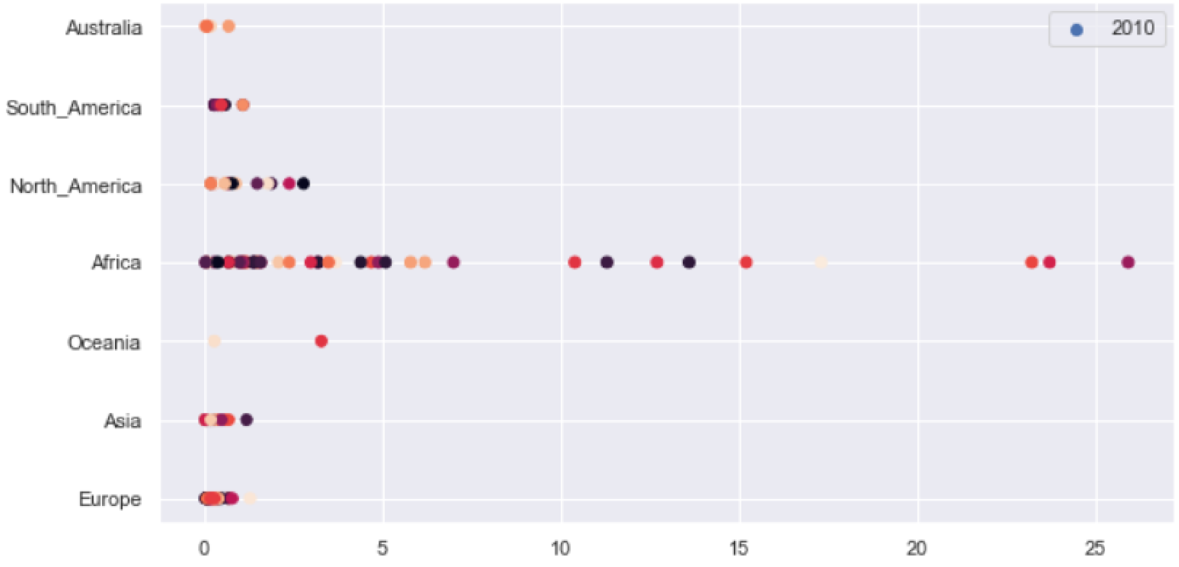


colr=np.random.RandomState(0)

colors=colr.rand(275)

plt.scatter(v["2010"],"continent",data=v,c=colors,label="2010")

plt.legend()



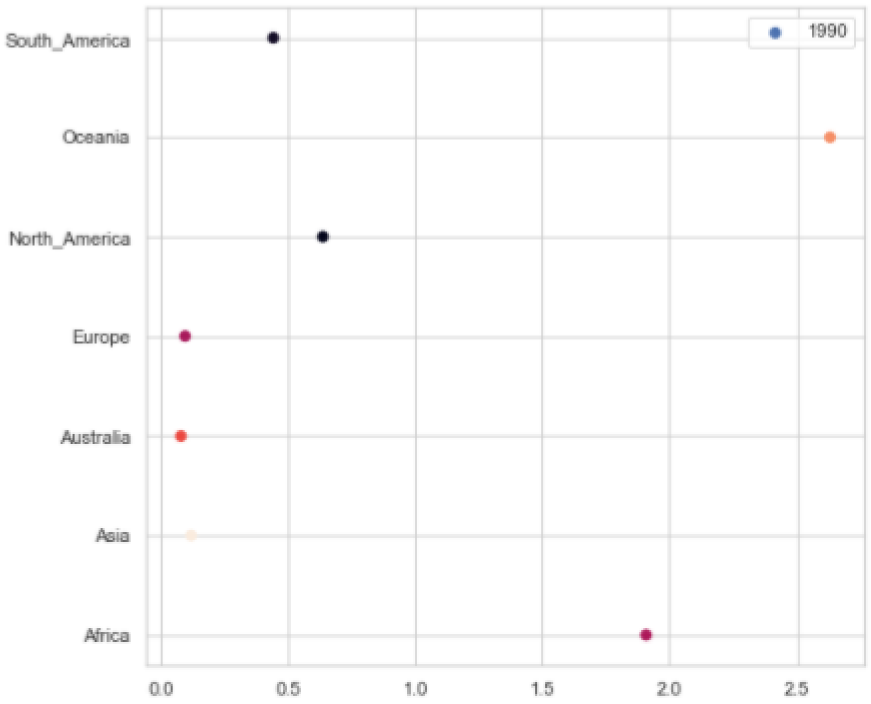
colr=np.random.RandomState(0)

colors=colr.rand(7)

plt.scatter(average_year[1990],"continent",data=average_year,c=colors)

plt.legend(["1990"])

create a scatter plot for every continent in 1990 year by using matplotlib library and x axis is averages and yaxis is continent.



South_America
Oceania
North_America
Europe
Australia
Asia
Africa
0.0
0.5
1.0
1.5
2.0
2.5
● 1990