Hamoye Stage C

In [1]:
```
#importing libraries
import pandas as pd
import plotly.express as px
```

In [2]:
```
#reading the data to pandas dataframe
df_prod=pd.read_csv("Africa Food Production (2004 - 2013).csv")
df_prod.head()
```

Out[2]:

	Country	Item	Year	Value
0	Algeria	Wheat and products	2004	2731
1	Algeria	Wheat and products	2005	2415
2	Algeria	Wheat and products	2006	2688
3	Algeria	Wheat and products	2007	2319
4	Algeria	Wheat and products	2008	1111

In [3]:
```
df_prod.rename(columns = {'Country':"Country Name"}, inplace = True)
```

In [4]:
```
#checking data type of columns
df_prod.dtypes
```

Out[4]:
```
Country Name    object
Item            object
Year             int64
Value            int64
dtype: object
```

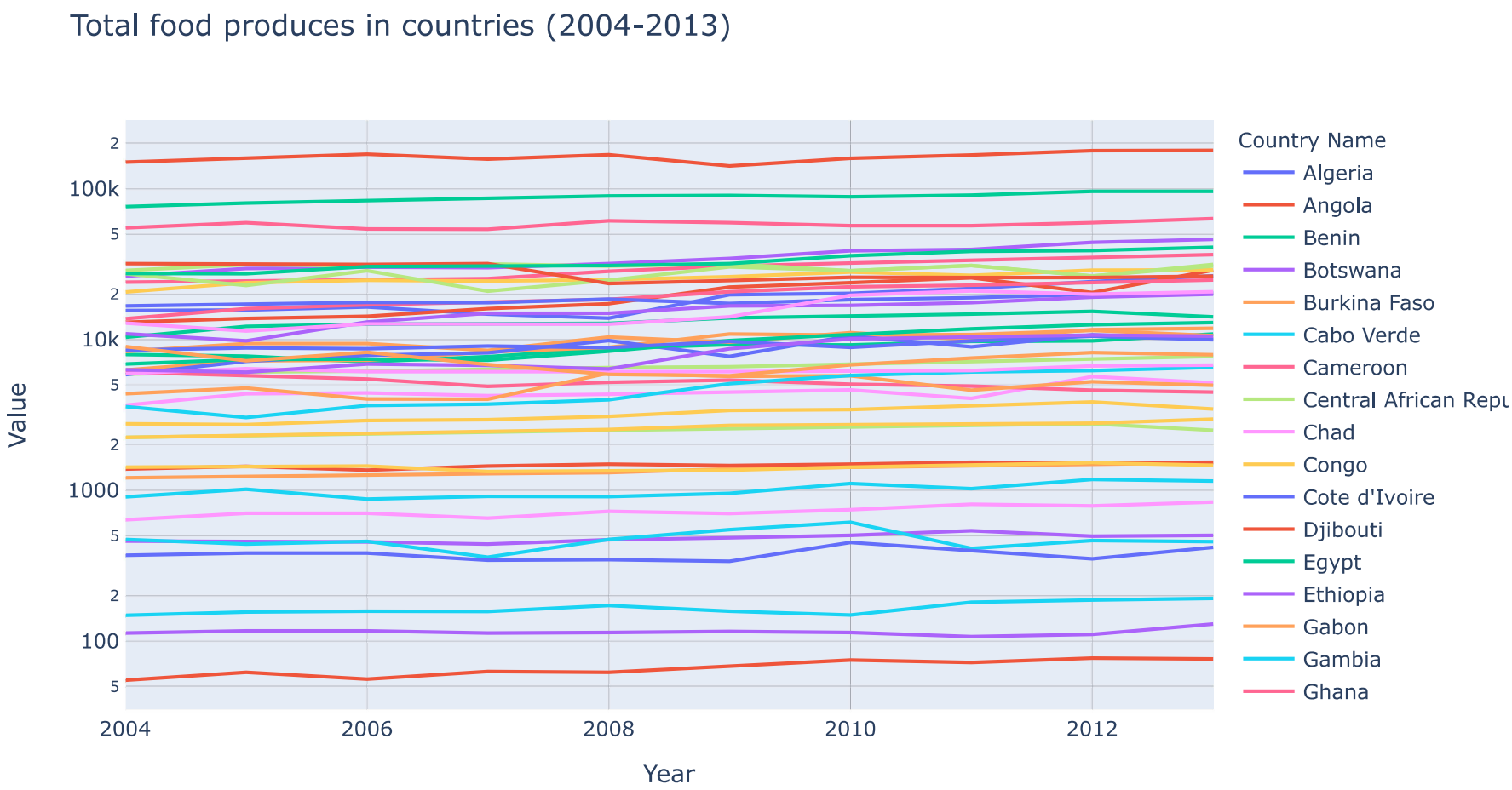
In [5]:
```
#grouping data
df_prod_grouped=pd.DataFrame(df_prod.groupby(["Country Name", "Year"])["Value"].sum().reset_index())
df_prod_grouped
```

Out[5]:

	Country Name	Year	Value
0	Algeria	2004	15536
1	Algeria	2005	15667
2	Algeria	2006	16417
3	Algeria	2007	14763
4	Algeria	2008	13841
...	...	...	...
445	Zimbabwe	2009	5754
446	Zimbabwe	2010	6777
447	Zimbabwe	2011	7551
448	Zimbabwe	2012	8173
449	Zimbabwe	2013	7914

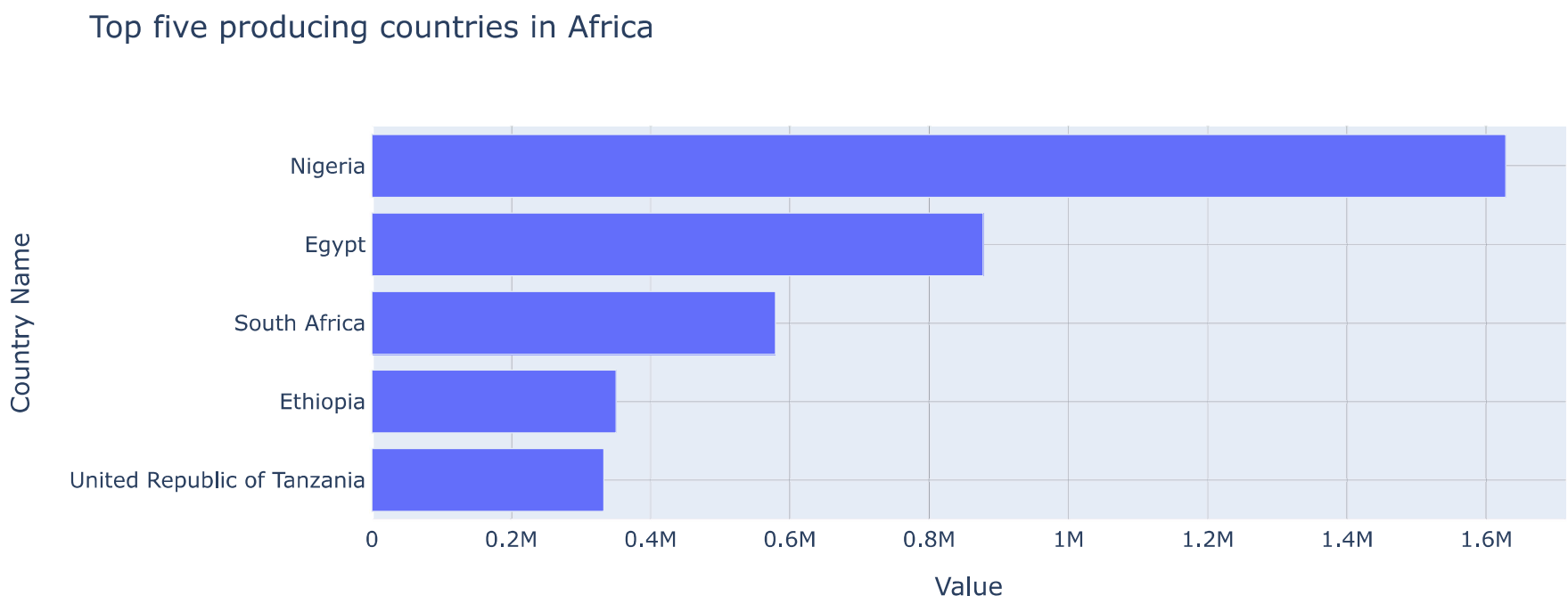
450 rows × 3 columns

In [6]:
```
#a line plot showing food production by country
fig = px.line(df_prod_grouped, x="Year", y="Value", color="Country Name", hover_name="Country Name", log_y=True,title
fig.show()
```

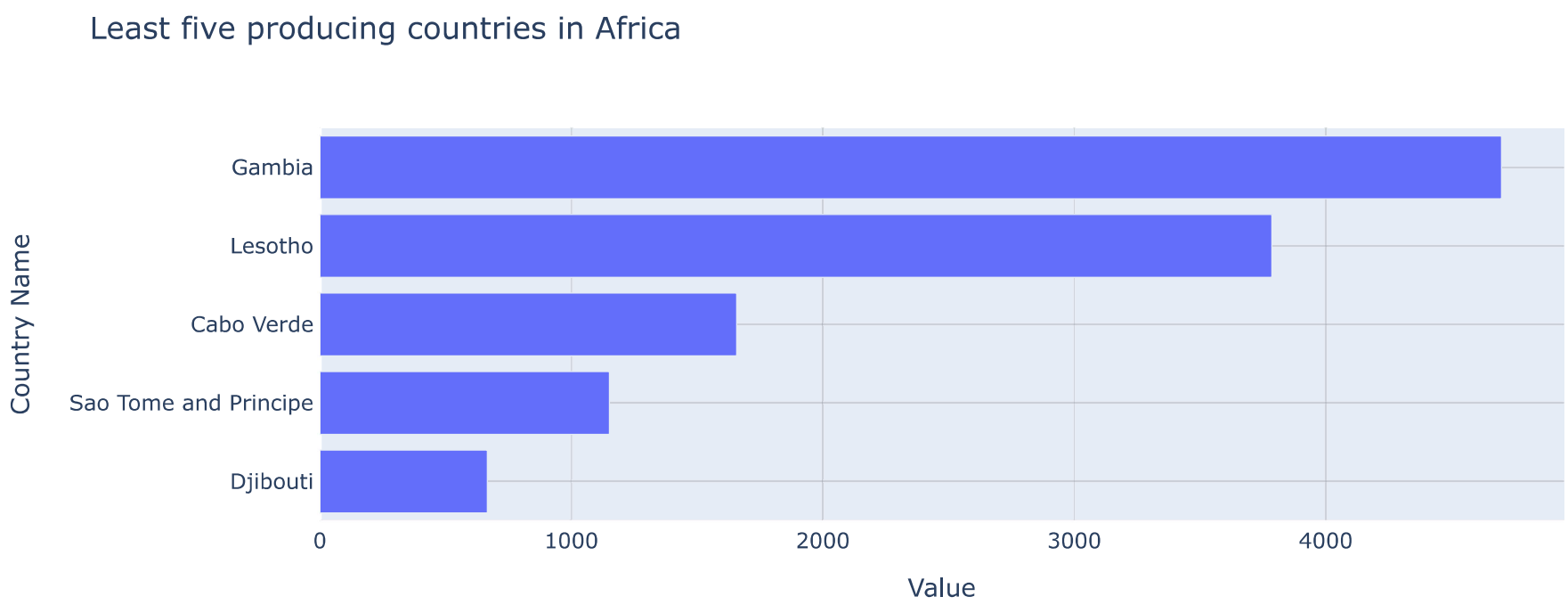


In [7]:
```
#grouping data
df_prod1=pd.DataFrame(df_prod.groupby(["Country Name"])["Value"].sum().reset_index())
df_prod1=df_prod1.sort_values(by="Value")
df_top5=df_prod1.tail()
df_last5=df_prod1.head()
```

In [8]:
```
#plotting top five producing countries in Africa
fig = px.bar(df_top5, y="Country Name", x="Value", orientation='h',height=400,
             title='Top five producing countries in Africa')
fig.show()
```



In [9]:
```
#plotting least five producing countries in Africa
fig = px.bar(df_last5, y="Country Name", x="Value", orientation='h',height=400,
             title='Least five producing countries in Africa')
fig.show()
```



The plot above shows that production has been constant over years with little changes over years, there is no major increase or decrease for any of countries.

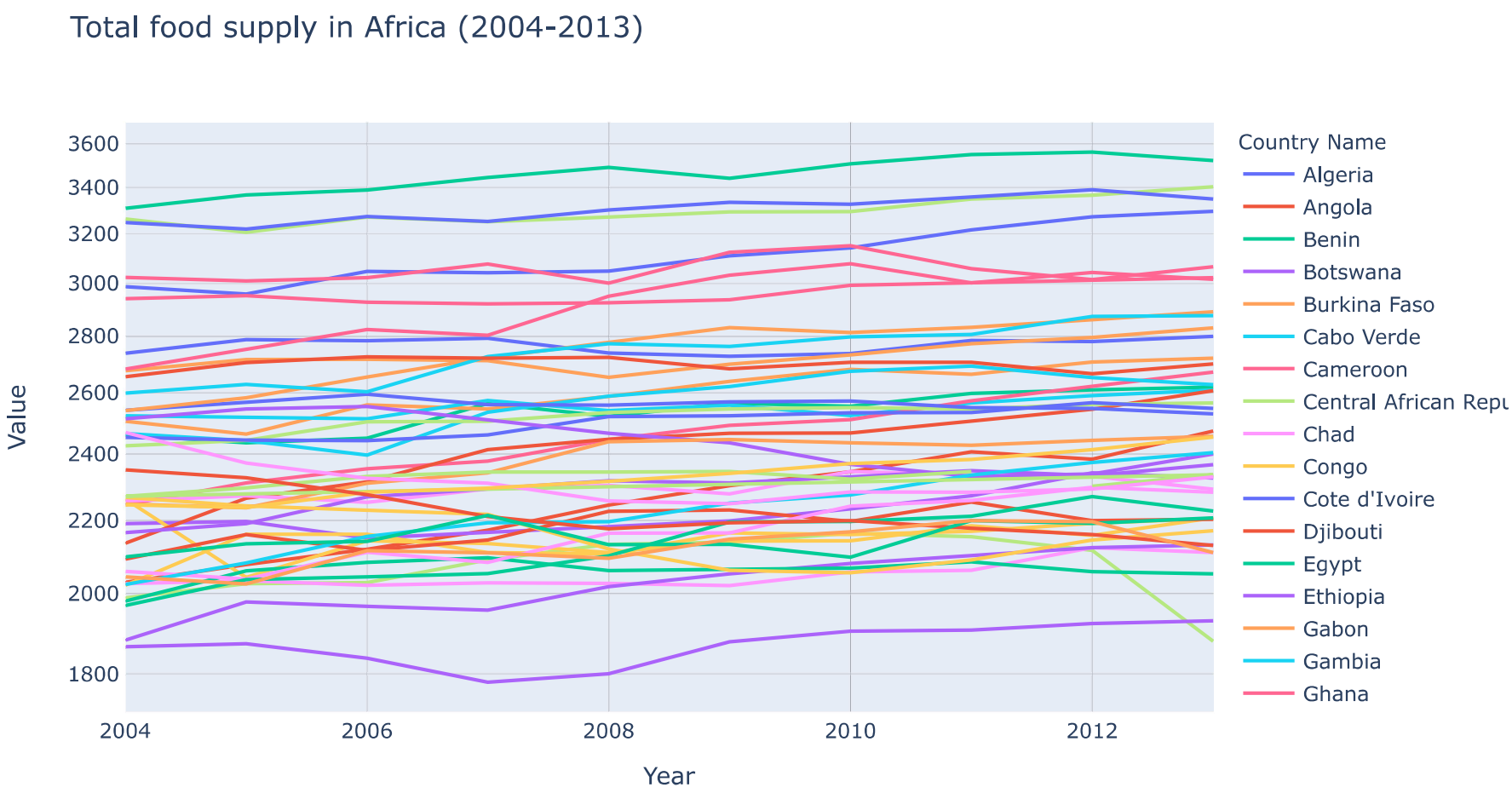
In [10]:

```
#loading the food supply dataset
df_s= pd.read_csv("Africa Food Supply (2004 - 2013).csv")
df_s.rename(columns = {'Country':"Country Name"}, inplace = True)
df_s.head()
```

Out[10]:

	Country Name	Year	Value
0	Algeria	2004	2987
1	Algeria	2005	2958
2	Algeria	2006	3047
3	Algeria	2007	3041
4	Algeria	2008	3048

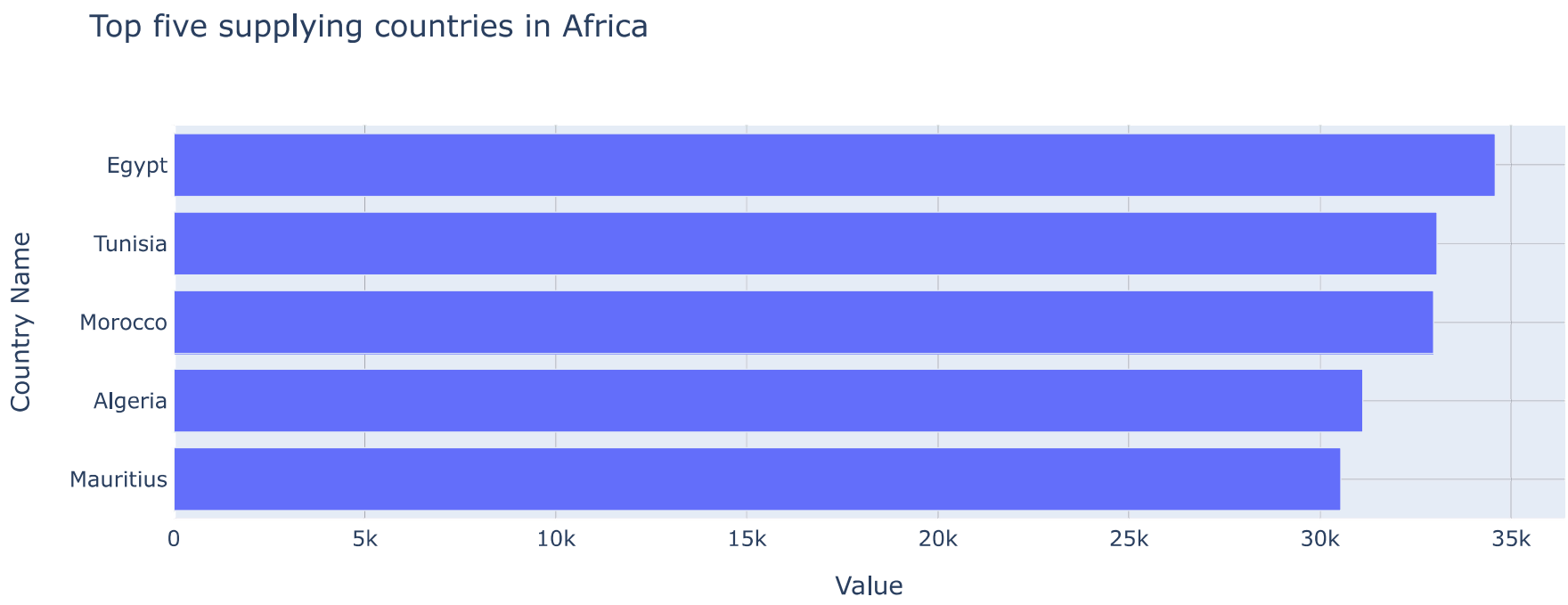
In [11]:
```
#a line plot showing food supply by country
fig=px.line(df_s, x="Year", y="Value", color="Country Name",hover_name="Country Name", log_y=True,title="Total food s
fig.show()
```



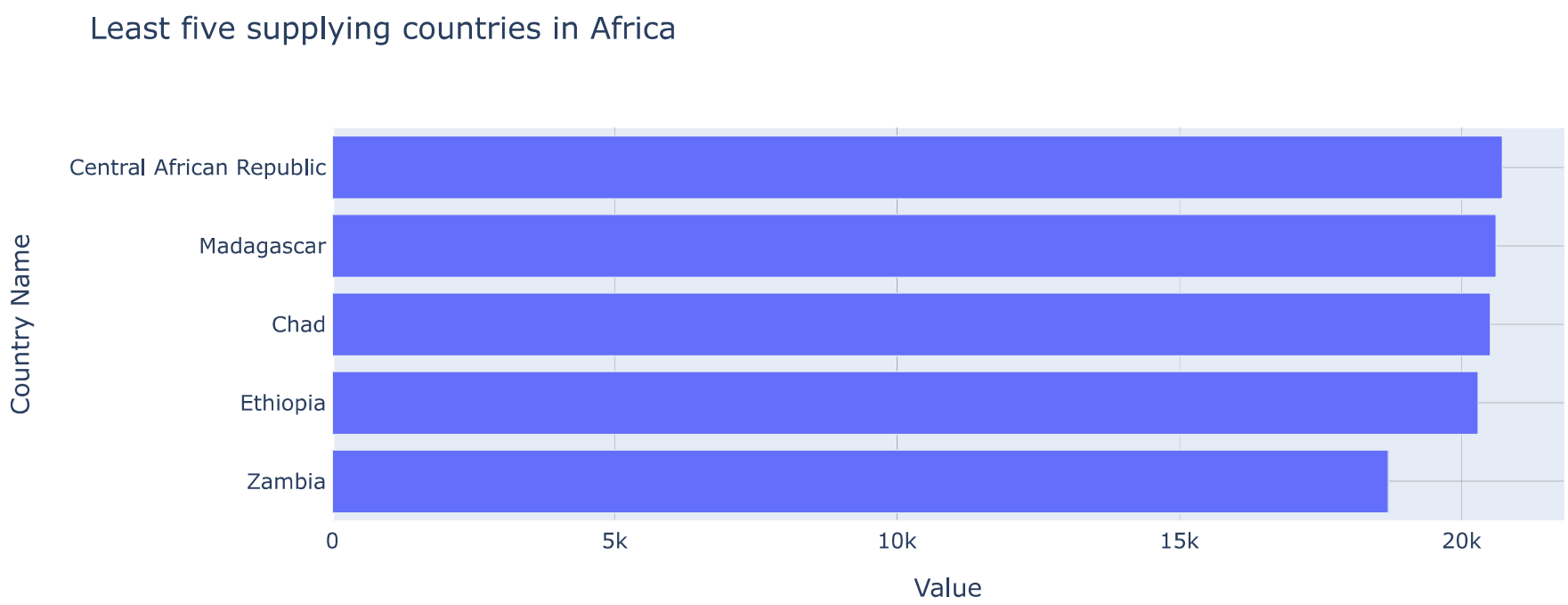
As seen in the plot above,it is easy to say that while most countries don't major increase in food supply, some countries had decrease in supply at some point, while some countries increased their supply some kept decresing like in the case of Central African Republic which was caused by drought and increase in prices of food.

In [12]:
```
#grouping data
df_sup1=pd.DataFrame(df_s.groupby(["Country Name"])["Value"].sum().reset_index())
df_sup1=df_sup1.sort_values(by="Value")
df_tops5=df_sup1.tail()
df_lasts5=df_sup1.head()
```

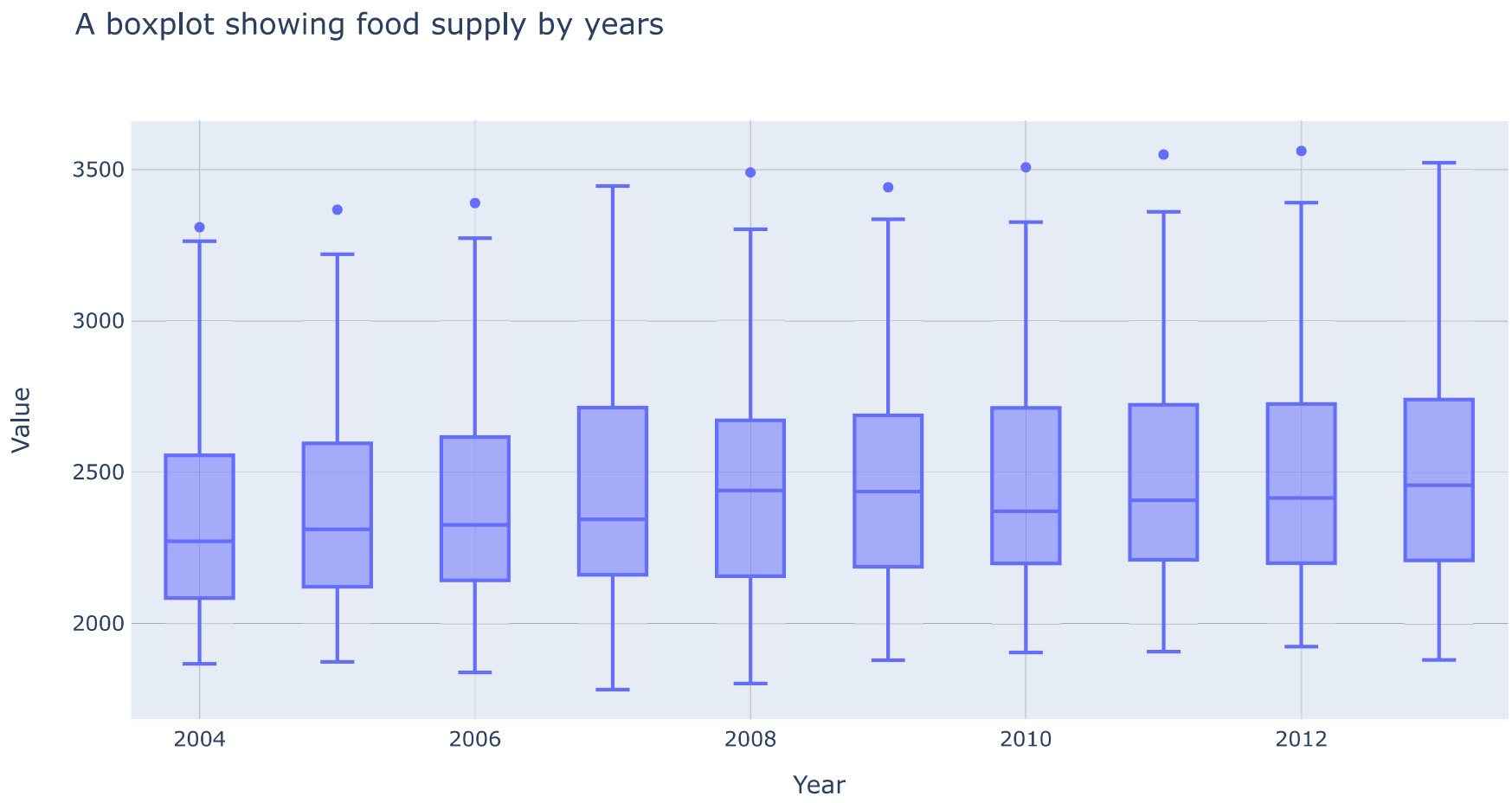
In [13]:
```
#plotting top five supplying countries in Africa
fig = px.bar(df_tops5, y="Country Name", x="Value", orientation='h',height=400,
             title='Top five supplying countries in Africa')
fig.show()
```



In [14]:
```
#plotting least five supplying countries in Africa
fig = px.bar(df_lasts5, y="Country Name", x="Value", orientation='h',height=400,
             title='Least five supplying countries in Africa')
fig.show()
```



In [15]:
```
#a boxplot showing the distribution by year
fig=px.box(df_s, x="Year", y="Value", title="A boxplot showing food supply by years")
fig.show()
```



In [16]:
```
#finding the exact country that has outlier in 2012
upper_fence=3468
target_year=df_s[df_s["Year"]==2012]

outlier= target_year[target_year.Value>=upper_fence]
outlier.head()
```

Out[16]:

	Country Name	Year	Value
128	Egypt	2012	3561

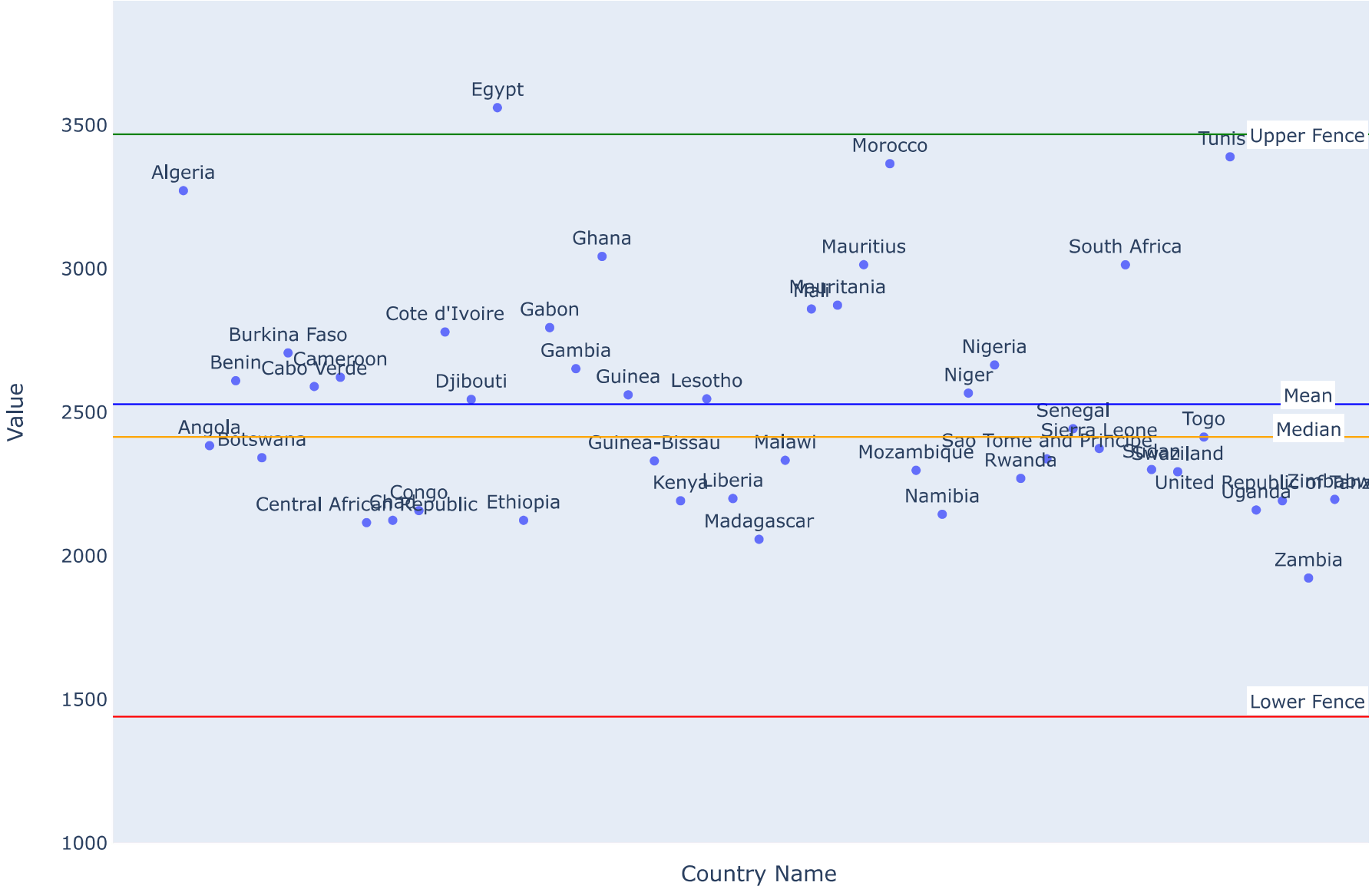
In [17]:
```
#finding statistical measures for ploting the outlier
target_year_des=target_year.describe()
target_year_des
```

Out[17]:

	Year	Value
count	45.0	45.000000
mean	2012.0	2527.644444
std	0.0	388.466926
min	2012.0	1923.000000
25%	2012.0	2200.000000
50%	2012.0	2414.000000
75%	2012.0	2707.000000
max	2012.0	3561.000000

In [18]:

```
#plotting outliers
fig=px.scatter(target_year,x="Country Name" ,y="Value", text="Country Name")
#adding lines to show mean, median, Q3 and Q1
fig.update_layout(width=1000,
                  height=700,
                  xaxis_showgrid=False,
                  yaxis_showgrid=False,
                  annotations=[
                      {"x":"Zambia", "y":1495, "xref": "x", "yref":"y", "showarrow":False, "text":"Lower Fence",
                       "bgcolor":"white"},
                      {"x":"Zambia", "y":2444, "xref": "x", "yref":"y", "showarrow":False, "text":"Median",
                       "bgcolor":"white"},
                      {"x":"Zambia", "y":3468, "xref": "x", "yref":"y", "showarrow":False, "text":"Upper Fence",
                      "bgcolor":"white"},
                      {"x":"Zambia", "y":2563, "xref": "x", "yref":"y", "showarrow":False, "text":"Mean",
                      "bgcolor":"white"}
                  ],
                  yaxis = dict(
                        tickmode = 'linear',
                        range=(1000,4000),
                        dtick = 500),
                  autosize=False,
                  title_text='Food supply (Africa, 2012)',
                  shapes=[
                      {"type": "line","xref":"paper", "x0": 0, "x1": 1, "yref": "y","y0":1440, "y1": 1440 , "line":
                       {"color":"red", "width": 1}},
                      {"type": "line","xref":"paper", "x0": 0, "x1": 1, "yref": "y","y0":2414, "y1": 2414, "line":
                       {"color":"orange", "width": 1}},
                      {"type": "line","xref":"paper", "x0": 0, "x1": 1, "yref": "y","y0":3468, "y1": 3468 , "line":
                       {"color":"green", "width": 1}},
                      {"type": "line","xref":"paper", "x0": 0, "x1": 1, "yref": "y","y0":2528, "y1": 2528 , "line":
                       {"color":"blue", "width": 1}}
                  ])
fig.update_traces(textposition="top center")
fig.update_xaxes(showticklabels=False)
fig.show()
```



Nigeria is the largest producer of food judging by the point it has on this line plot showing produces, but Egypt is the largest supplier, it supplies even above the upper fence while Nigeria the largest producer is somewhere between the mean and the upper fence, Djibouti which is the lowest

producer was able to make it to the mean but no country supplies below the lower fence.

In [19]:
```
#loading the population dataset
df_p=pd.read_csv("./API_SP.POP.TOTL_DS2_en_csv_v2_1308146.csv")
df_p.head()
```

Out[19]:

	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	1962	1963	1964	1965	...	2011	2012
0	Aruba	ABW	Population, total	SP.POP.TOTL	54211.0	55438.0	56225.0	56695.0	57032.0	57360.0	...	102046.0	102560.0
1	Afghanistan	AFG	Population, total	SP.POP.TOTL	8996973.0	9169410.0	9351441.0	9543205.0	9744781.0	9956320.0	...	30117413.0	31161376.0
2	Angola	AGO	Population, total	SP.POP.TOTL	5454933.0	5531472.0	5608539.0	5679458.0	5735044.0	5770570.0	...	24220661.0	25107931.0
3	Albania	ALB	Population, total	SP.POP.TOTL	1608800.0	1659800.0	1711319.0	1762621.0	1814135.0	1864791.0	...	2905195.0	2900401.0
4	Andorra	AND	Population, total	SP.POP.TOTL	13411.0	14375.0	15370.0	16412.0	17469.0	18549.0	...	83747.0	82427.0

5 rows × 65 columns

In [20]:
```
#dropping unnecessary columns
df_p.drop(["Indicator Code","Indicator Name", "Unnamed: 64"], axis=1, inplace=True)
```

In [21]:
```
#melting df_p to met the merging requirements
df_p_melt=pd.melt(df_p, id_vars=["Country Name", "Country Code"], var_name="Year", value_name="Value")
df_p_melt.head()
```

Out[21]:

	Country Name	Country Code	Year	Value
0	Aruba	ABW	1960	54211.0
1	Afghanistan	AFG	1960	8996973.0
2	Angola	AGO	1960	5454933.0
3	Albania	ALB	1960	1608800.0
4	Andorra	AND	1960	13411.0

In [22]:
```
#renaming values to population
df_p_melt.rename(columns={"Value":"Population"}, inplace=True)
```

In [23]:
```
#changing the type of year
df_p_melt['Year']=df_p_melt['Year'].astype(int)
```

In [24]:
```
#changing columns to give equal number of rows as df_prod after merging
df_p_melt["Country Name"] = df_p_melt["Country Name"].replace(["Egypt, Arab Rep."], "Egypt")
df_p_melt["Country Name"] = df_p_melt["Country Name"].replace(["Congo, Rep."], "Congo")
df_p_melt["Country Name"] = df_p_melt["Country Name"].replace(["Gambia, The"], "Gambia")
df_p_melt["Country Name"] = df_p_melt["Country Name"].replace(["Eswatini"], "Swaziland")
df_p_melt["Country Name"] = df_p_melt["Country Name"].replace(["Tanzania"], "United Republic of Tanzania")
```

In [25]:
```
#extracting the values for rice only
Rice_data = df_prod[df_prod['Item']=='Rice (Milled Equivalent)'].reset_index()
Rice_data.head()
```

Out[25]:

	index	Country Name	Item	Year	Value
0	10	Algeria	Rice (Milled Equivalent)	2004	0
1	11	Algeria	Rice (Milled Equivalent)	2005	0
2	12	Algeria	Rice (Milled Equivalent)	2006	0
3	13	Algeria	Rice (Milled Equivalent)	2007	0
4	14	Algeria	Rice (Milled Equivalent)	2008	0

In [26]:
```
#extracting value for rice by country
Rice_data1=pd.DataFrame(Rice_data.groupby("Country Name")["Value"].sum())
Rice_data1=Rice_data1.sort_values(by="Value")
Rice_data1.tail()
```

Out[26]:

	Value
Country Name	
Guinea	10400
United Republic of Tanzania	10956
Madagascar	26118
Nigeria	26964
Egypt	40370

In [27]:
```
#extracting the values for oats only
oats_data = df_prod[df_prod['Item']=='Oats'].reset_index()
oats_data.head()
```

Out[27]:

	index	Country Name	Item	Year	Value
0	40	Algeria	Oats	2004	89
1	41	Algeria	Oats	2005	78
2	42	Algeria	Oats	2006	89
3	43	Algeria	Oats	2007	92
4	44	Algeria	Oats	2008	27

In [28]:
```
target_year1=pd.DataFrame(oats_data[oats_data["Year"]==2010])
target_year1=target_year1.sort_values(by="Value")
target_year1.tail()
```

Out[28]:

	index	Country Name	Item	Year	Value
36	9286	Kenya	Oats	2010	3
66	17816	South Africa	Oats	2010	34
56	13186	Morocco	Oats	2010	37
16	6406	Ethiopia	Oats	2010	48
6	46	Algeria	Oats	2010	88

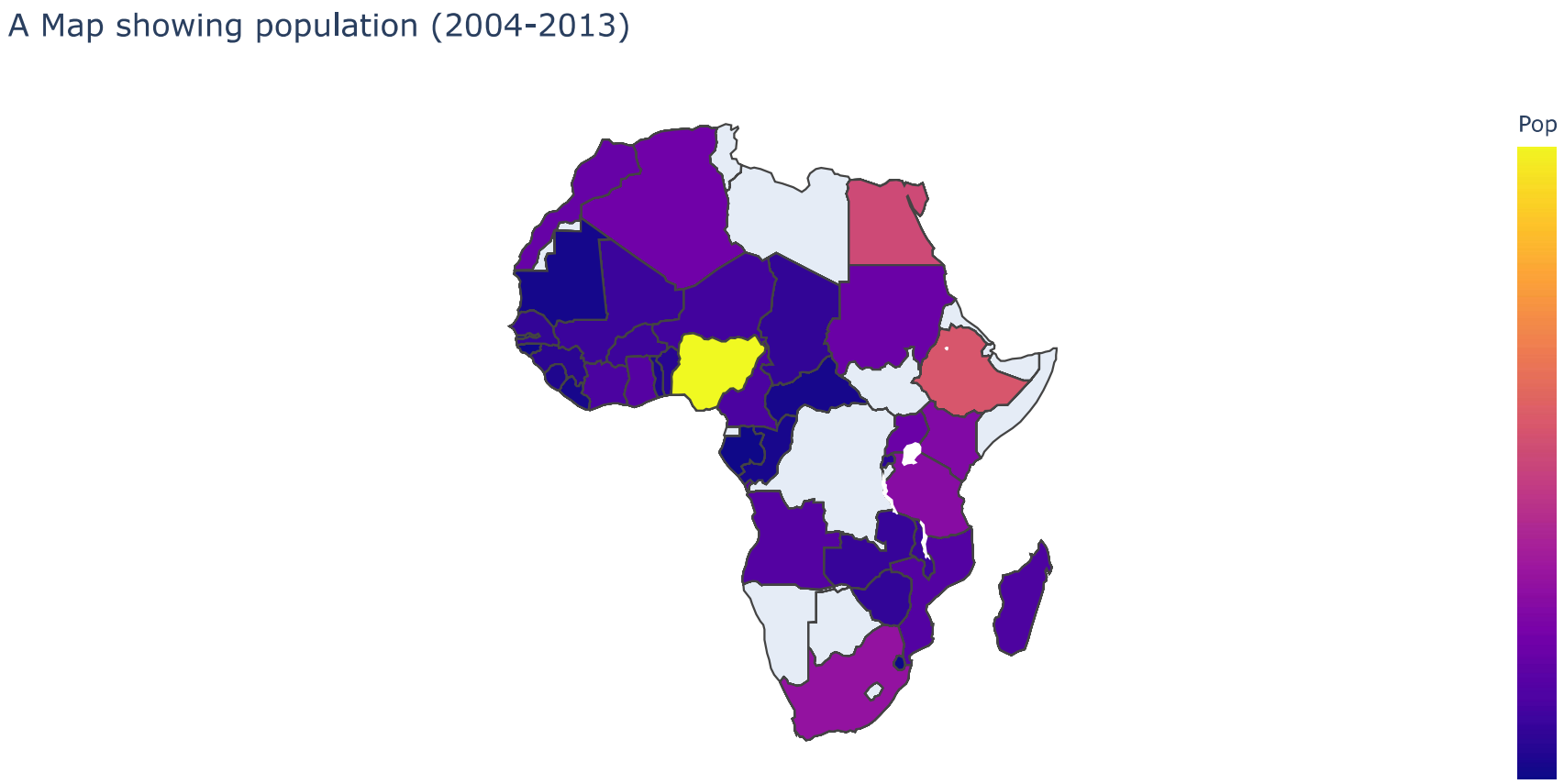
In [29]:

```
#merging rice_data and df_p_melt
df_rpop = pd.merge(Rice_data,
                   df_p_melt,
                   on=["Country Name", "Year"],
                   how="left"
                   )
df_rpop.rename(columns = {'Value':"Rice (in KT)"}, inplace = True)
df_rpop.head()
```

Out[29]:

	index	Country Name	Item	Year	Rice (in KT)	Country Code	Population
0	10	Algeria	Rice (Milled Equivalent)	2004	0	DZA	32692163.0
1	11	Algeria	Rice (Milled Equivalent)	2005	0	DZA	33149724.0
2	12	Algeria	Rice (Milled Equivalent)	2006	0	DZA	33641002.0
3	13	Algeria	Rice (Milled Equivalent)	2007	0	DZA	34166972.0
4	14	Algeria	Rice (Milled Equivalent)	2008	0	DZA	34730608.0

In [30]:
```
#showing map view of population
fig = px.choropleth(df_rpop, locations="Country Code", color="Population", hover_name="Country Name",
                    scope="africa", color_continuous_scale="Plasma")
fig.update_layout(title="A Map showing population (2004-2013)")
fig.show()
```



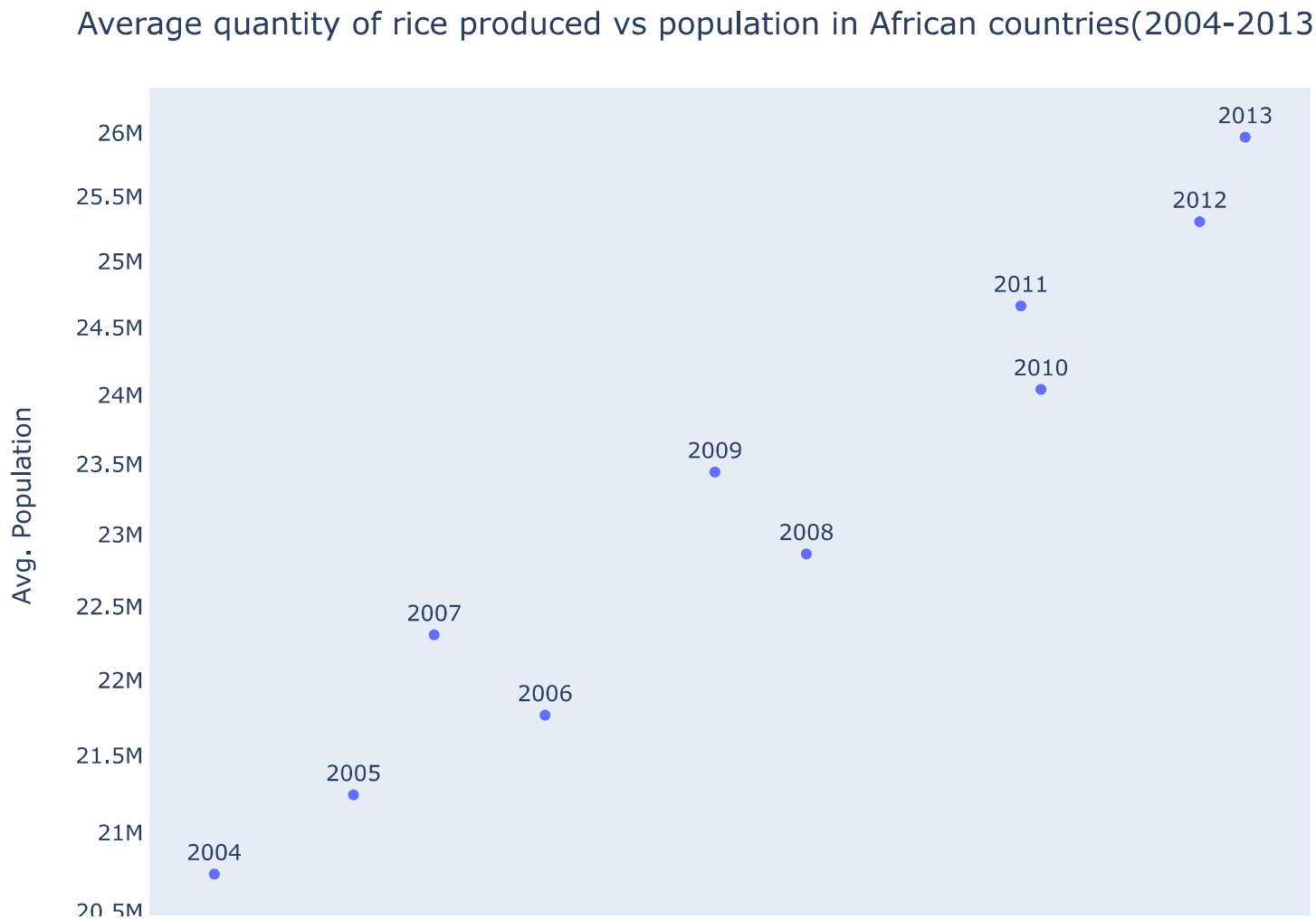
The map shows that Nigeria is able to attain largest producer because of its poppulation which is approximately 136million(as at 2013).

In [31]:
```
#getting mean of the dataframes by years
df_rpop_grouped=pd.DataFrame(df_rpop.groupby("Year")[["Rice (in KT)", "Population"]].mean().reset_index())
df_rpop_grouped.rename(columns = {'Rice (in KT)':"Avg. Rice (in KT)", "Population":"Avg. Population"}, inplace = True
df_rpop_grouped
```

Out[31]:

	Year	Avg. Rice (in KT)	Avg. Population
0	2004	326.789474	2.073717e+07
1	2005	348.763158	2.124461e+07
2	2006	379.026316	2.176808e+07
3	2007	361.500000	2.230803e+07
4	2008	420.289474	2.286609e+07
5	2009	405.842105	2.344414e+07
6	2010	457.315789	2.404335e+07
7	2011	454.157895	2.466432e+07
8	2012	482.394737	2.530639e+07
9	2013	489.578947	2.596753e+07

In [32]:
```
#plotting df_rpop_grouped to show the correlation between avarage rice produced and population by year
fig=px.scatter(df_rpop_grouped,x="Avg. Rice (in KT)" ,y="Avg. Population", hover_name="Year", log_y=True, text="Year'
fig.update_layout(width=800,
                  height=600,
                  xaxis_showgrid=False,
                  yaxis_showgrid=False,
                  title=" Average quantity of rice produced vs population in African countries(2004-2013)",
                 )
fig.update_traces(textposition="top center")
fig.show()
```



The plot shows that there is a positive correlation between average quantity of rice produced and population.

In [33]:

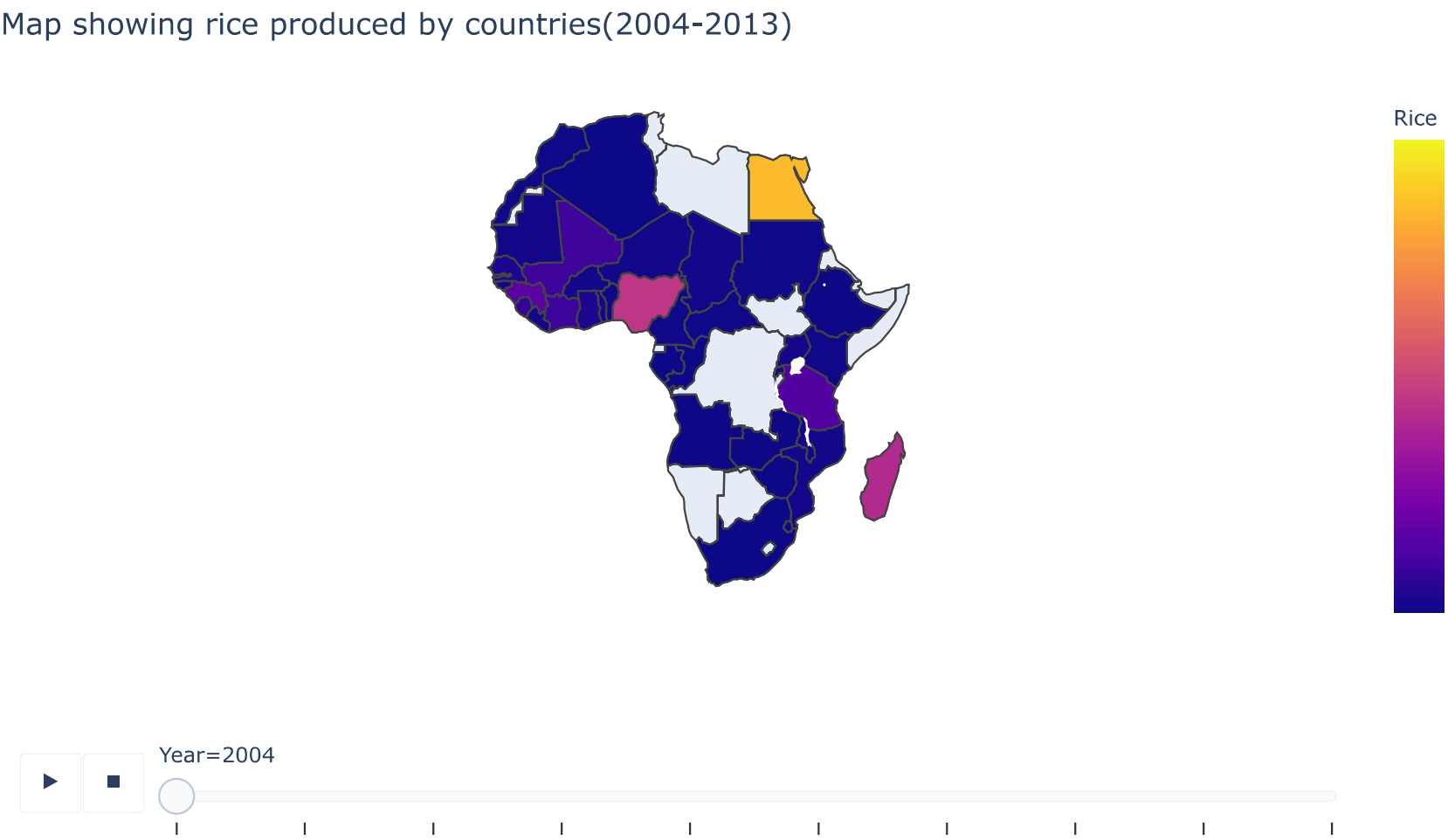
```
#checking the correlation coefficient
from scipy import stats
a=df_rpop_grouped["Avg. Rice (in KT)"]
b=df_rpop_grouped["Avg. Population"]
corr = stats.pearsonr(a, b)
print("Correlation coefficient:", corr[0])
```

```
Correlation coefficient: 0.9706032040326851
```

This further confirms that there is a positive relationship between average rice produced and population.

In [34]:

```
#using to map to show countries that produces more rice
fig = px.choropleth(df_rpop, locations="Country Code", color="Rice (in KT)", hover_name="Country Name",
                    scope="africa",animation_frame="Year", range_color=[0, 5000], color_continuous_scale="Plasma")
fig.update_layout(title="A Map showing rice produced by countries(2004-2013)")
fig.show()
```



In [35]:
```
#merging df_prod_grouped and df_p_melt
df_ppop = pd.merge(df_prod_grouped,
                   df_p_melt,
                   on=["Country Name", "Year"]
                   )
df_ppop.rename(columns = {'Value':"Produces(in KT)"}, inplace = True)
df_ppop
```

Out[35]:

	Country Name	Year	Produces(in KT)	Country Code	Population
0	Algeria	2004	15536	DZA	32692163.0
1	Algeria	2005	15667	DZA	33149724.0
2	Algeria	2006	16417	DZA	33641002.0
3	Algeria	2007	14763	DZA	34166972.0
4	Algeria	2008	13841	DZA	34730608.0
...	...	...	...	...	...
445	Zimbabwe	2009	5754	ZWE	12526968.0
446	Zimbabwe	2010	6777	ZWE	12697723.0
447	Zimbabwe	2011	7551	ZWE	12894316.0
448	Zimbabwe	2012	8173	ZWE	13115131.0
449	Zimbabwe	2013	7914	ZWE	13350356.0

450 rows × 5 columns

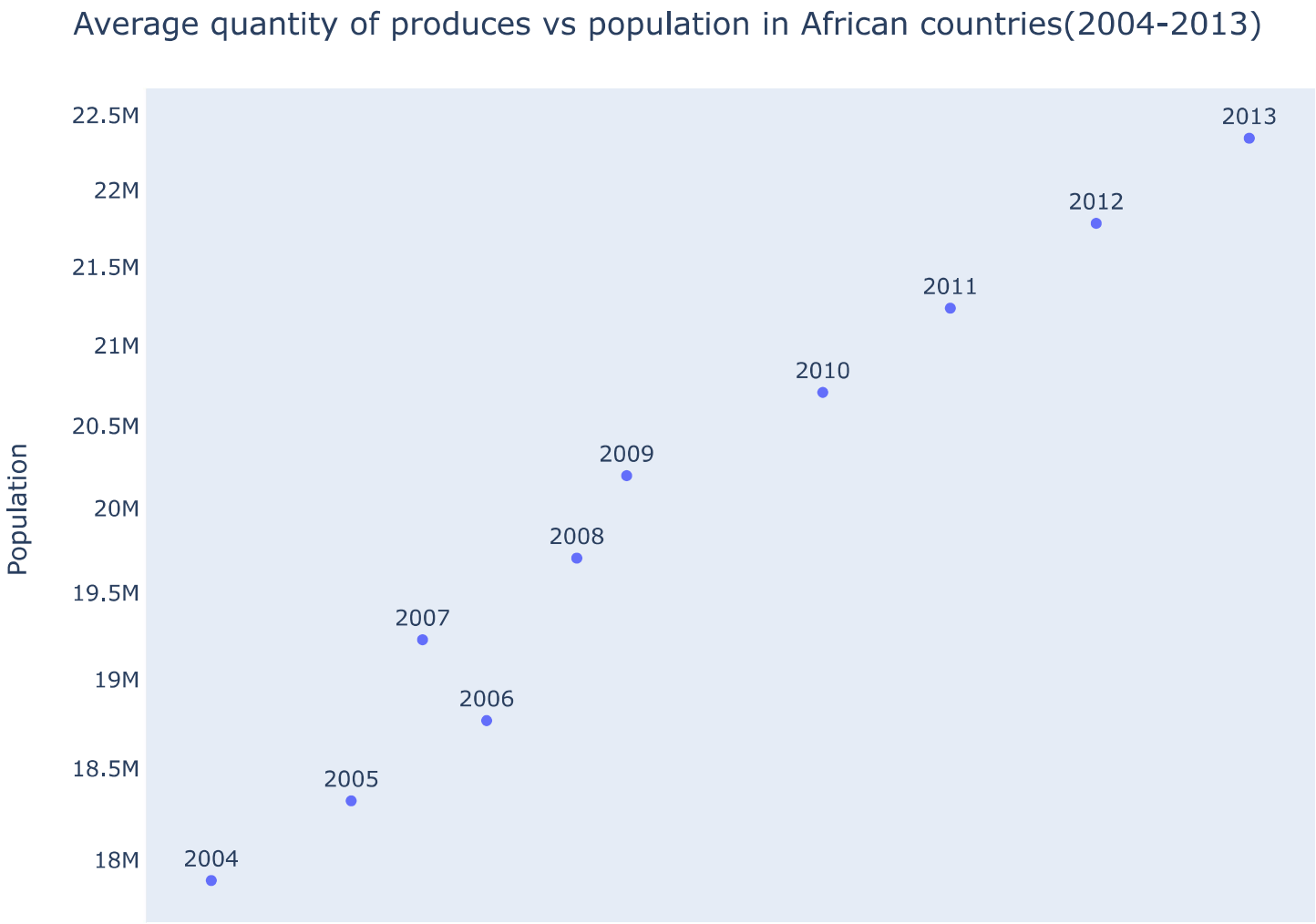
In [36]:
```
#getting mean of the dataframes by years
df_ppop_grouped=df_ppop.groupby("Year").mean().reset_index()
df_ppop_grouped
```

Out[36]:

	Year	Produces(in KT)	Population
0	2004	14733.466667	1.789102e+07
1	2005	15361.266667	1.832310e+07
2	2006	15968.933333	1.876913e+07
3	2007	15681.311111	1.922945e+07
4	2008	16373.422222	1.970534e+07
5	2009	16597.111111	2.019818e+07
6	2010	17477.022222	2.070883e+07
7	2011	18049.200000	2.123775e+07
8	2012	18703.711111	2.178446e+07
9	2013	19390.466667	2.234735e+07

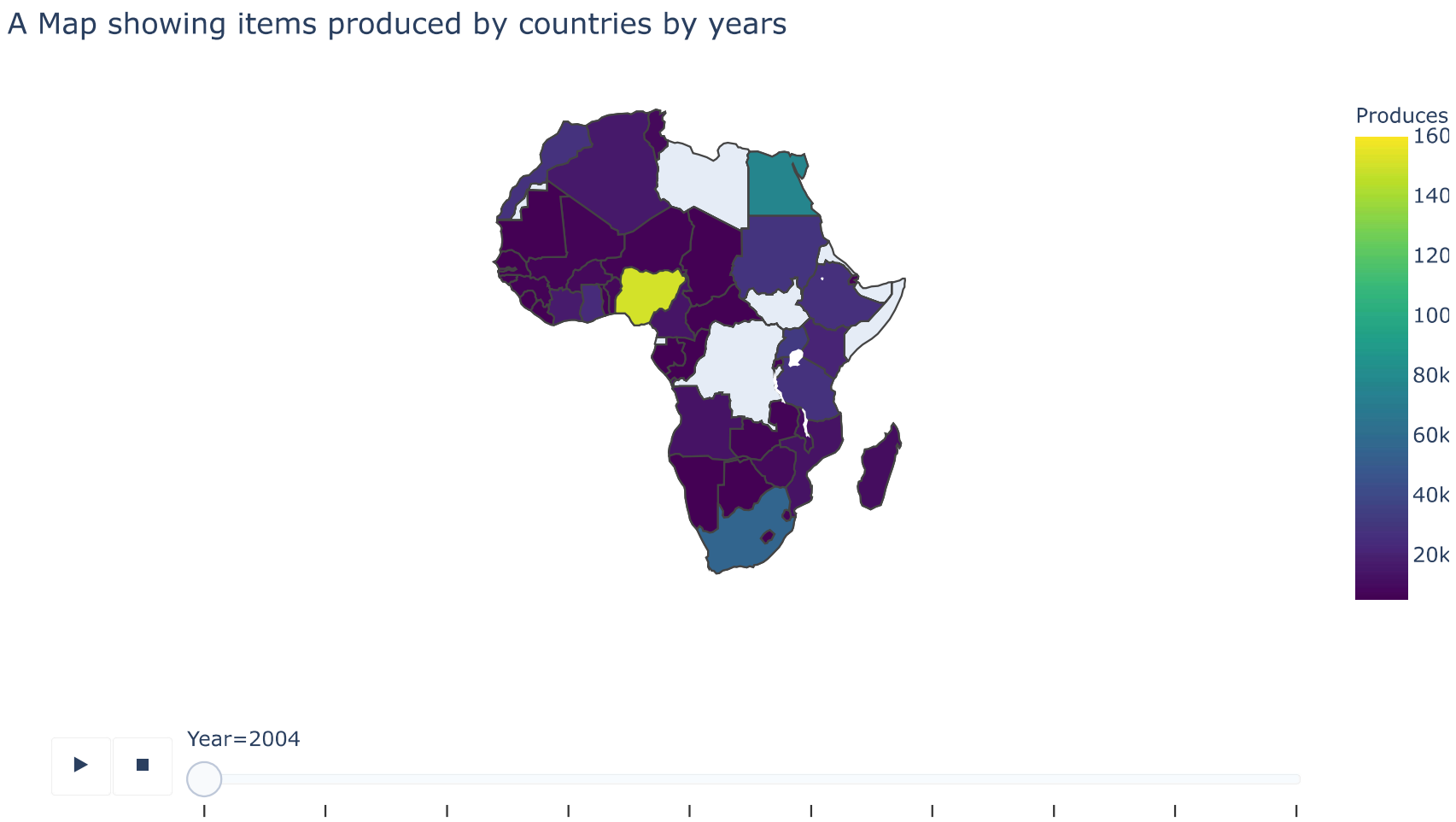
In [37]:

```
#plotting df_ppop_grouped to show the correlation between avarage produces and population by year
fig=px.scatter(df_ppop_grouped,x="Produces(in KT)" ,y="Population", hover_name="Year", log_y=True, text="Year")
fig.update_layout(width=800,
                  height=600,
                  xaxis_showgrid=False,
                  yaxis_showgrid=False,
                  title=" Average quantity of produces vs population in African countries(2004-2013)",
                 )
fig.update_traces(textposition="top center")
fig.show()
```



The plot shows that there is a positive correlation between taverage produces and population.

In [38]:
```
#using to map to show countries that has more produces
fig = px.choropleth(df_ppop, locations="Country Code", color="Produces(in KT)", hover_name="Country Name",
                    scope="africa",animation_frame="Year", range_color=[5000,160000], color_continuous_scale="Viridis
fig.update_layout(title="A Map showing items produced by countries by years")
fig.show()
```



In [39]:

```
#merging df_s and df_p_melt
df_spop = pd.merge(df_s,
                   df_p_melt,
                   on=["Country Name", "Year"]
                   )
df_spop.rename(columns = {'Value':"Supply(in KT)"}, inplace = True)
df_spop
```

Out[39]:

	Country Name	Year	Supply(in KT)	Country Code	Population
0	Algeria	2004	2987	DZA	32692163.0
1	Algeria	2005	2958	DZA	33149724.0
2	Algeria	2006	3047	DZA	33641002.0
3	Algeria	2007	3041	DZA	34166972.0
4	Algeria	2008	3048	DZA	34730608.0
...	...	...	...	...	...
445	Zimbabwe	2009	2147	ZWE	12526968.0
446	Zimbabwe	2010	2168	ZWE	12697723.0
447	Zimbabwe	2011	2200	ZWE	12894316.0
448	Zimbabwe	2012	2197	ZWE	13115131.0
449	Zimbabwe	2013	2110	ZWE	13350356.0

450 rows × 5 columns

In [40]:
```
#getting mean of the dataframes by years
df_spop_grouped=df_spop.groupby("Year").mean().reset_index()
df_spop_grouped
```

Out[40]:

	Year	Supply(in KT)	Population
0	2004	2394.222222	1.789102e+07
1	2005	2409.288889	1.832310e+07
2	2006	2430.800000	1.876913e+07
3	2007	2447.755556	1.922945e+07
4	2008	2460.755556	1.970534e+07
5	2009	2482.222222	2.019818e+07
6	2010	2497.400000	2.070883e+07
7	2011	2515.422222	2.123775e+07
8	2012	2527.644444	2.178446e+07
9	2013	2532.244444	2.234735e+07

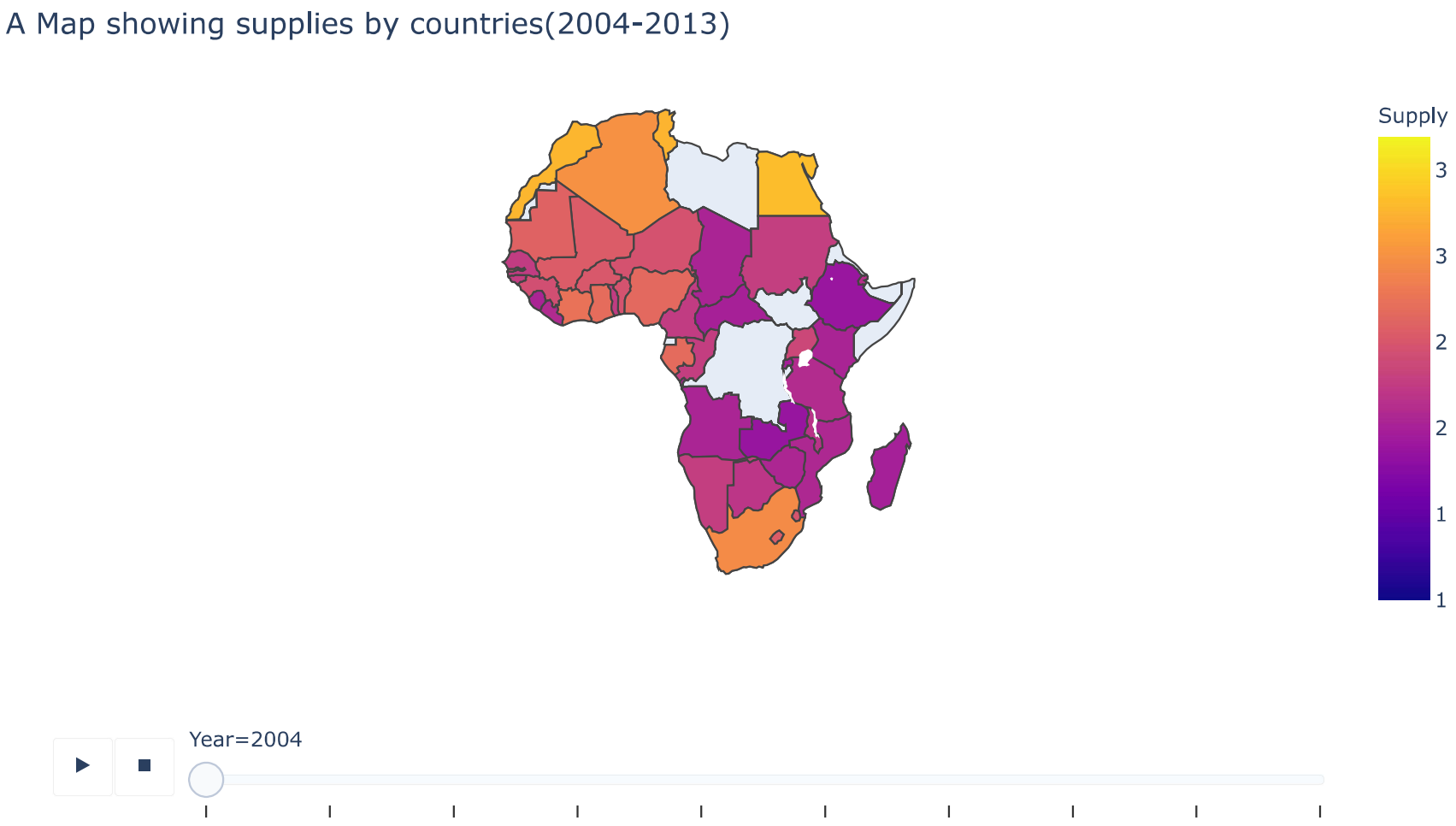
In [41]:

```
#plotting df_spop_grouped to show the correlation between avarage supplies and population
fig=px.scatter(df_spop_grouped,x="Supply(in KT)" ,y="Population", hover_name="Year", log_y=True, text="Year")
fig.update_layout(width=800,
                  height=600,
                  xaxis_showgrid=False,
                  yaxis_showgrid=False,
                  title=" Average quantity of produces vs population in African countries(2004-2013)",
                 )
fig.update_traces(textposition="top center")
fig.show()
```

The plot shows that there is a positive correlation between average supplies and population.

In [42]:

```
#using to map to show countries that supplies more
fig = px.choropleth(df_spop, locations="Country Code", color="Supply(in KT)", hover_name="Country Name",
                    scope="africa",animation_frame="Year", range_color=[1000, 3700], color_continuous_scale="Plasma")
fig.update_layout(title="A Map showing supplies by countries(2004-2013)")
fig.show()
```



In [43]:

```
#merging df_prod_grouped and df_s and df_p_melt
df_sprod = pd.merge(df_s,
                df_prod_grouped,
                on=["Country Name", "Year"]
                )
df_sprod.rename(columns = {'Value_y':"Produces(in KT)", "Value_x":"Supply(in KT)"}, inplace = True)

df_sprod
```

Out[43]:

	Country Name	Year	Supply(in KT)	Produces(in KT)
0	Algeria	2004	2987	15536
1	Algeria	2005	2958	15667
2	Algeria	2006	3047	16417
3	Algeria	2007	3041	14763
4	Algeria	2008	3048	13841
...	...	...	...	...
445	Zimbabwe	2009	2147	5754
446	Zimbabwe	2010	2168	6777
447	Zimbabwe	2011	2200	7551
448	Zimbabwe	2012	2197	8173
449	Zimbabwe	2013	2110	7914

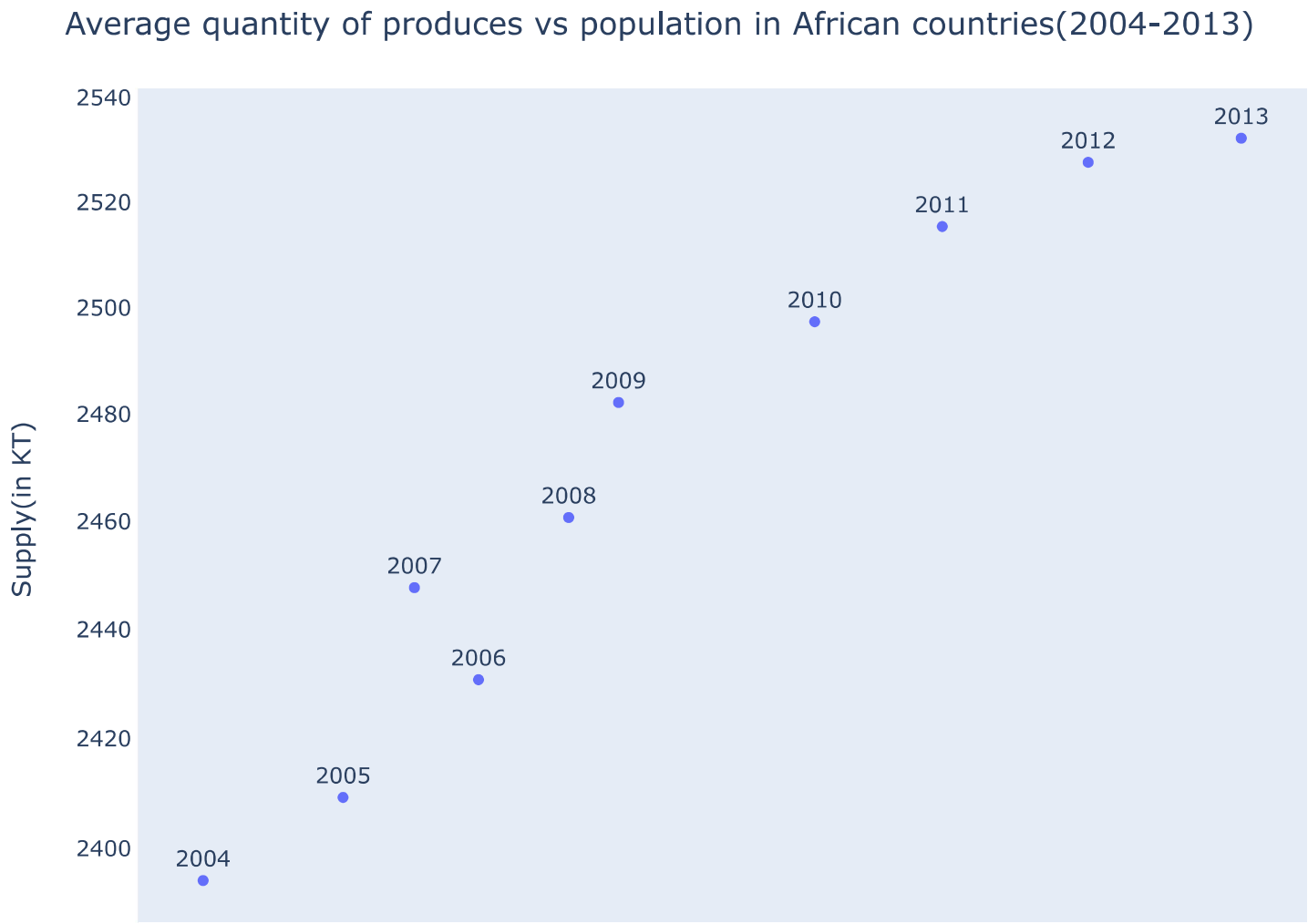
450 rows × 4 columns

In [44]:
```
#getting mean of the dataframes by years
df_sprod_grouped=df_sprod.groupby("Year").mean().reset_index()
df_sprod_grouped
```

Out[44]:

	Year	Supply(in KT)	Produces(in KT)
0	2004	2394.222222	14733.466667
1	2005	2409.288889	15361.266667
2	2006	2430.800000	15968.933333
3	2007	2447.755556	15681.311111
4	2008	2460.755556	16373.422222
5	2009	2482.222222	16597.111111
6	2010	2497.400000	17477.022222
7	2011	2515.422222	18049.200000
8	2012	2527.644444	18703.711111
9	2013	2532.244444	19390.466667

In [45]:
```
#plotting df_sprod_grouped to show the correlation between avarage rice produced and population
fig=px.scatter(df_sprod_grouped,x="Produces(in KT)" ,y="Supply(in KT)", hover_name="Year", log_y=True, text="Year")
fig.update_layout(width=800,
                  height=600,
                  xaxis_showgrid=False,
                  yaxis_showgrid=False,
                  title=" Average quantity of produces vs population in African countries(2004-2013)",
                 )
fig.update_traces(textposition="top center")
fig.show()
```



The plot shows that there is a positive correlation between total produces and supplies.

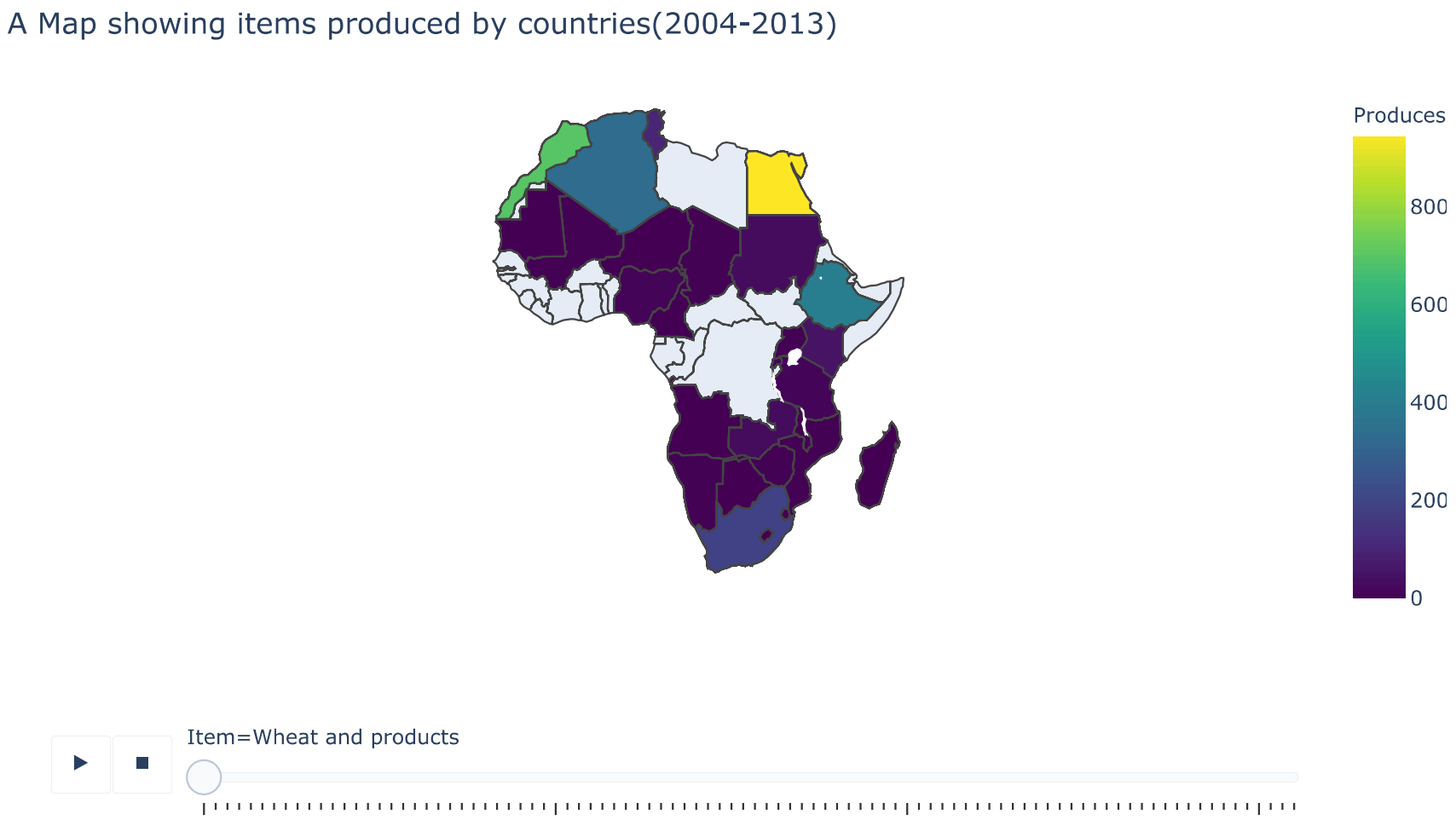
In [46]:

```
#merging df_proddf_p_melt
df_pprod = pd.merge(df_prod,
                    df_p_melt,
                    on=["Country Name", "Year"]
                    )
df_pprod.rename(columns = {'Value':"Produces(in KT)"}, inplace = True)
df_pprod.head()
```

Out[46]:

	Country Name	Item	Year	Produces(in KT)	Country Code	Population
0	Algeria	Wheat and products	2004	2731	DZA	32692163.0
1	Algeria	Rice (Milled Equivalent)	2004	0	DZA	32692163.0
2	Algeria	Barley and products	2004	1212	DZA	32692163.0
3	Algeria	Maize and products	2004	1	DZA	32692163.0
4	Algeria	Oats	2004	89	DZA	32692163.0

In [47]:
```
#using to map to show countries that has more produces of a specific item
fig = px.choropleth(df_pprod, locations="Country Code", color="Produces(in KT)", hover_name="Country Name",
                    scope="africa",animation_frame="Item",  color_continuous_scale="Viridis")
fig.update_layout(title="A Map showing items produced by countries(2004-2013)")
fig.show()
```



The map shows that not all countries produces all items even Nigeria the top producer doesn't produce all items, She thrives well with meats, tuber, grains and cash crops as compared to aquatic farming, Uganda and Tanzania thrive well in that aspect. It goes further to show that only three

African countries(South Africa, Egypt, and Morocco) produces Rye and three African countries(Tanzania,Madagascar and Kenya) produces cloves (as at 2013).

In [48]:
```
#merging df_prod_grouped and df_s and df_p_melt
df_sppop = pd.merge(df_pprod,
                df_s,
                on=["Country Name", "Year"]
                )
df_sppop.rename(columns = {'Value_y':"Produces(in KT)", "Value":"Supply(in KT)"}, inplace = True)
df_sppop.head()
```

Out[48]:

	Country Name	Item	Year	Produces(in KT)	Country Code	Population	Supply(in KT)
0	Algeria	Wheat and products	2004	2731	DZA	32692163.0	2987
1	Algeria	Rice (Milled Equivalent)	2004	0	DZA	32692163.0	2987
2	Algeria	Barley and products	2004	1212	DZA	32692163.0	2987
3	Algeria	Maize and products	2004	1	DZA	32692163.0	2987
4	Algeria	Oats	2004	89	DZA	32692163.0	2987

In [49]:
```
df_sppop.to_csv("./supply and produce in africa (2004-2013).csv", index=False, header=True)
```