In [1]:
```
#Exploratory data analysis to discover patterns to check assumptions with the help of graphical represen
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()
plt.style.use('seaborn-v0_8')
```

In [65]:
```
df=pd.read_csv(r"C:\Users\Chinenye Claire\Desktop\cleaned_data (3).csv")
```

In [66]:
```
df.head()
```

Out[66]:

	Country Name	Year	Incidence of malaria (per 1,000 population at risk)	Malaria cases reported	Malaria death	Use of insecticide-treated bed net in total population	Children with fever receiving antimalarial drugs (% of children under age 5 with fever)	Intermittent preventive treatment (IPT) of malaria in pregnancy (% of pregnant women)	Total Population	Rural Population	...	People using a least basi drinking wate services urban (% of urbai population
0	Algeria	2007-01-01	0.01	26.0	0.0	4.7625	4.9125	19.163636	33983827.0	11776076.0	...	94.78
1	Angola	2007-01-01	286.72	1533485.0	0.0	18.0000	29.8000	1.500000	20909684.0	8881597.0	...	65.8:
2	Benin	2007-01-01	480.24	0.0	0.0	2.8125	18.6750	15.000000	8647761.0	5053924.0	...	76.2
3	Botswana	2007-01-01	1.03	390.0	3.0	21.6500	73.8625	8.600000	1966977.0	827547.0	...	94.3
4	Burkina Faso	2007-01-01	503.80	44246.0	0.0	24.9200	67.0625	7.000000	14757074.0	11363537.0	...	76.1

5 rows × 27 columns

In [5]:
```
df.dtypes
```

Out[5]:
```
Country Name                                                                                  object
Year                                                                                          object
Incidence of malaria (per 1,000 population at risk)                                          float64
Malaria cases reported                                                                       float64
Malaria death                                                                                float64
Use of insecticide-treated bed net in total population                                       float64
Children with fever receiving antimalarial drugs (% of children under age 5 with fever)      float64
Intermittent preventive treatment (IPT) of malaria in pregnancy (% of pregnant women)        float64
Total Population                                                                             float64
Rural Population                                                                             float64
Urban Population                                                                             float64
Rural population (% of total population)                                                     float64
Rural population growth (annual %)                                                           float64
Urban population (% of total population)                                                     float64
Urban population growth (annual %)                                                           float64
People using at least basic drinking water services (% of population)                        float64
People using at least basic drinking water services, rural (% of rural population)           float64
People using at least basic drinking water services, urban (% of urban population)           float64
People using at least basic sanitation services (% of population)                            float64
People using at least basic sanitation services, rural (% of rural population)               float64
People using at least basic sanitation services, urban  (% of urban population)              float64
latitude                                                                                     float64
longitude                                                                                    float64
geometry                                                                                      object
Total Malaria Cases                                                                          float64
Mortality Rate                                                                               float64
Prevalence Rate                                                                              float64
dtype: object
```

In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 462 entries, 0 to 461
Data columns (total 27 columns):
 #   Column                                                                          Non-Null
Count  Dtype
---  ------                                                                          --------
------  -----
 0   Country Name                                                                    462 non-
null    object
 1   Year                                                                            462 non-
null    object
 2   Incidence of malaria (per 1,000 population at risk)                             462 non-
null    float64
 3   Malaria cases reported                                                          462 non-
null    float64
 4   Malaria death                                                                   462 non-
null    float64
 5   Use of insecticide-treated bed net in total population                          462 non-
null    float64
 6   Children with fever receiving antimalarial drugs (% of children under age 5 with fever)  462 non-
null    float64
 7   Intermittent preventive treatment (IPT) of malaria in pregnancy (% of pregnant women)    462 non-
null    float64
 8   Total Population                                                                462 non-
null    float64
 9   Rural Population                                                                462 non-
null    float64
 10  Urban Population                                                                462 non-
null    float64
 11  Rural population (% of total population)                                        462 non-
null    float64
 12  Rural population growth (annual %)                                              462 non-
null    float64
 13  Urban population (% of total population)                                        462 non-
null    float64
 14  Urban population growth (annual %)                                              462 non-
null    float64
 15  People using at least basic drinking water services (% of population)           462 non-
null    float64
 16  People using at least basic drinking water services, rural (% of rural population)   462 non-
null    float64
 17  People using at least basic drinking water services, urban (% of urban population)   462 non-
null    float64
 18  People using at least basic sanitation services (% of population)               462 non-
null    float64
 19  People using at least basic sanitation services, rural (% of rural population)  462 non-
null    float64
 20  People using at least basic sanitation services, urban  (% of urban population)  462 non-
null    float64
 21  latitude                                                                        462 non-
null    float64
 22  longitude                                                                       462 non-
null    float64
 23  geometry                                                                        462 non-
null    object
 24  Total Malaria Cases                                                             462 non-
null    float64
 25  Mortality Rate                                                                  462 non-
null    float64
 26  Prevalence Rate                                                                 462 non-
null    float64
dtypes: float64(24), object(3)
memory usage: 97.6+ KB
```

In [67]:
```
data=df.rename(columns={'Incidence of malaria (per 1,000 population at risk)':'incidence rate','Use of
```

In [68]:
```
data.head()
```

Out[68]:

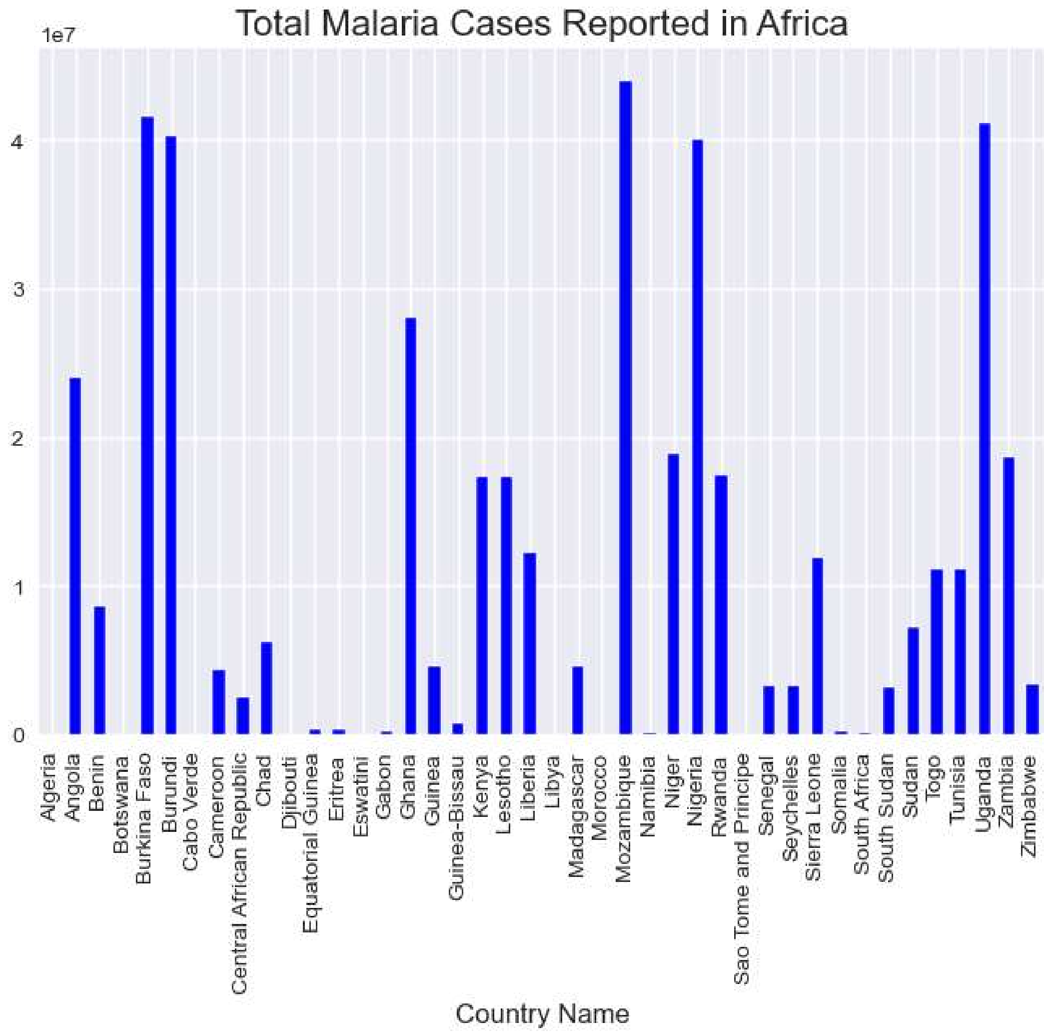
	Country Name	Year	incidence rate	Malaria cases reported	Malaria death	ITN total	% under 5 children on ACT	% pregnant women on IPT	Total Population	Rural Population	...	% Urban Pop using BDWS	% Pop using BSS	R u
0	Algeria	2007-01-01	0.01	26.0	0.0	4.7625	4.9125	19.163636	33983827.0	11776076.0	...	94.78	85.85	7
1	Angola	2007-01-01	286.72	1533485.0	0.0	18.0000	29.8000	1.500000	20909684.0	8881597.0	...	65.83	37.26	1
2	Benin	2007-01-01	480.24	0.0	0.0	2.8125	18.6750	15.000000	8647761.0	5053924.0	...	76.24	11.80	
3	Botswana	2007-01-01	1.03	390.0	3.0	21.6500	73.8625	8.600000	1966977.0	827547.0	...	94.35	61.60	3
4	Burkina Faso	2007-01-01	503.80	44246.0	0.0	24.9200	67.0625	7.000000	14757074.0	11363537.0	...	76.15	15.60	

In [9]:
```
#statistics summary
data.describe().T
#huge difference between min and max values shows evidence of outliers
#minimum value of o incidence rates, reported cases and deaths shows malaria was eliminated in some cou
```

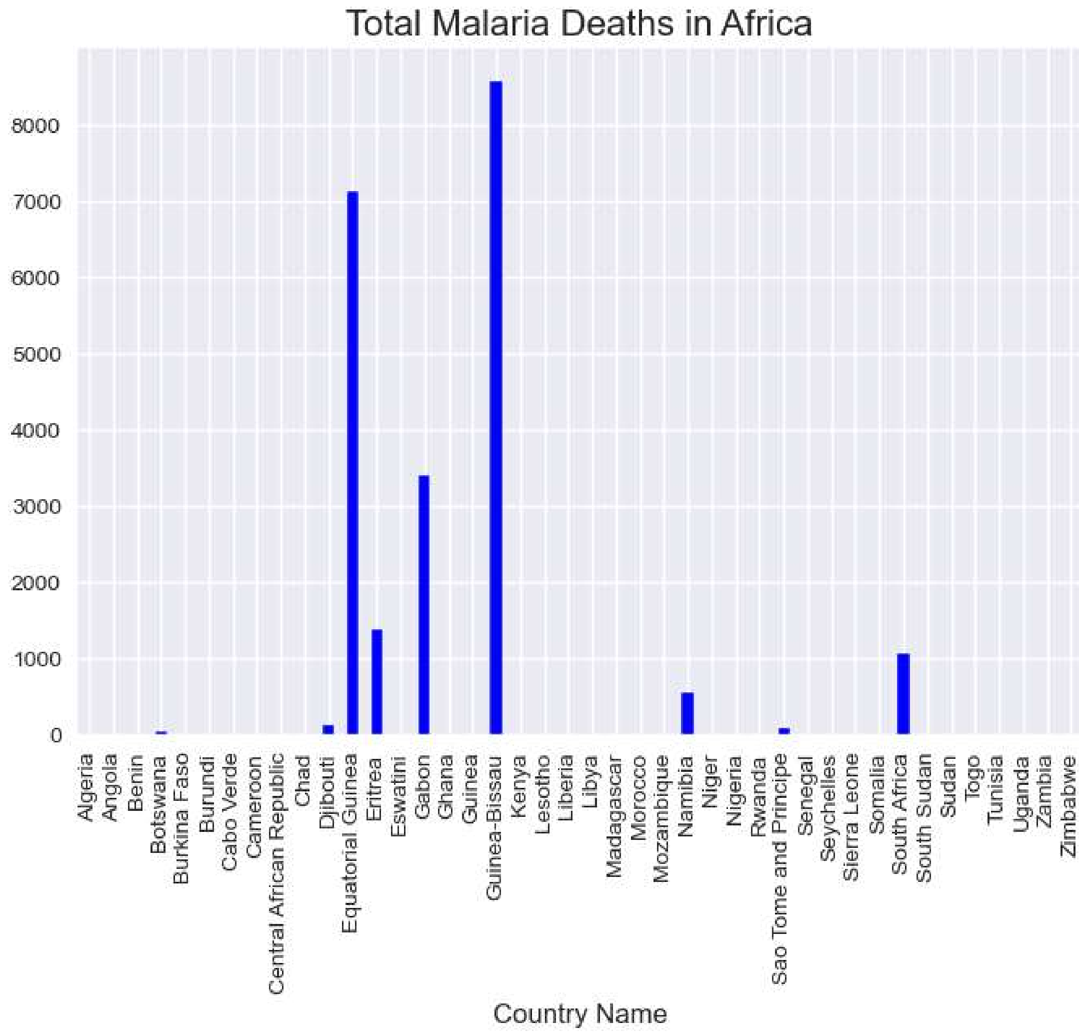
Out[9]:

	count	mean	std	min	25%	50%	75%	max
incidence rate	462.0	1.836460e+02	1.633838e+02	0.000000	2.581250e+01	1.560450e+02	3.466700e+02	5.855400e+02
Malaria cases reported	462.0	9.712805e+05	1.912862e+06	0.000000	2.345250e+03	1.711445e+05	1.041084e+06	1.229382e+07
Malaria death	462.0	4.844589e+01	1.602435e+02	0.000000	0.000000e+00	0.000000e+00	0.000000e+00	8.140000e+02
ITN total	462.0	4.028751e+01	2.466112e+01	0.160000	1.931812e+01	4.197500e+01	5.937500e+01	9.550000e+01
% under 5 children on ACT	462.0	3.020410e+01	2.126452e+01	0.200000	1.200000e+01	2.765625e+01	4.334375e+01	9.887143e+01
% pregnant women on IPT	462.0	1.705291e+01	1.610125e+01	0.000000	3.465909e+00	1.290000e+01	2.680000e+01	7.280000e+01
Total Population	462.0	1.762875e+07	2.733676e+07	85033.000000	2.288321e+06	1.104398e+07	2.233601e+07	1.934959e+08
Rural Population	462.0	1.005121e+07	1.507862e+07	40468.000000	1.301555e+06	7.357656e+06	1.228312e+07	9.767867e+07
Urban Population	462.0	7.577541e+06	1.314700e+07	44460.000000	1.239060e+06	3.369878e+06	7.852774e+06	9.581724e+07
Rural population (% of total population)	462.0	5.540307e+01	1.906638e+01	11.020000	3.886250e+01	5.829500e+01	6.854500e+01	9.014000e+01
Rural population growth (annual %)	462.0	1.278723e+00	1.295397e+00	-3.450000	1.500000e-01	1.605000e+00	2.057500e+00	7.090000e+00
Urban population (% of total population)	462.0	4.459773e+01	1.906612e+01	9.860000	3.145500e+01	4.171000e+01	6.113750e+01	8.898000e+01
Urban population growth (annual %)	462.0	3.494329e+00	1.440270e+00	-4.650000	2.390000e+00	3.710000e+00	4.360000e+00	7.400000e+00
% Pop using BDWS	462.0	6.558255e+01	1.648065e+01	32.910000	5.227750e+01	6.314500e+01	7.945500e+01	9.853000e+01
Rural % Pop using BDWS	462.0	5.056481e+01	1.600283e+01	17.050000	3.816500e+01	5.051000e+01	6.078250e+01	8.871000e+01
% Urban Pop using BDWS	462.0	8.398857e+01	9.415290e+00	52.010000	7.735000e+01	8.432000e+01	9.130000e+01	9.970000e+01
% Pop using BSS	462.0	4.025043e+01	2.605920e+01	6.630000	1.739500e+01	3.436000e+01	5.832750e+01	1.000000e+02
% Rural Pop using BSS	462.0	2.712803e+01	2.209490e+01	1.890000	7.817500e+00	1.831000e+01	3.989500e+01	8.221000e+01
% Urban Pop using BSS	462.0	4.852110e+01	2.065029e+01	12.580000	3.077500e+01	4.520000e+01	6.309750e+01	9.529000e+01
latitude	462.0	2.693280e+00	1.605725e+01	-30.559482	-4.679574e+00	6.744051e+00	1.223833e+01	3.388692e+01
longitude	462.0	1.650710e+01	1.901266e+01	-24.013197	1.659626e+00	1.818215e+01	3.021764e+01	5.549198e+01
Total Malaria Cases	462.0	3.758988e+06	9.211784e+06	0.000000	5.474830e+04	1.404877e+06	3.793560e+06	6.523623e+07
Mortality Rate	462.0	3.119739e-05	1.113219e-04	0.000000	0.000000e+00	0.000000e+00	0.000000e+00	6.429079e-04
Prevalence Rate	462.0	1.545961e-01	5.412884e-01	0.000000	7.091058e-04	1.933285e-02	8.790226e-02	5.269303e+00

In [10]:
```
Cases=data.groupby("Country Name")["Malaria cases reported"].sum()
Cases.plot(kind='bar', color = 'blue')
plt.title('Total Malaria Cases Reported in Africa', fontsize=16)
plt.show()
#no malaria cases reported in eight (8) African countries;
```

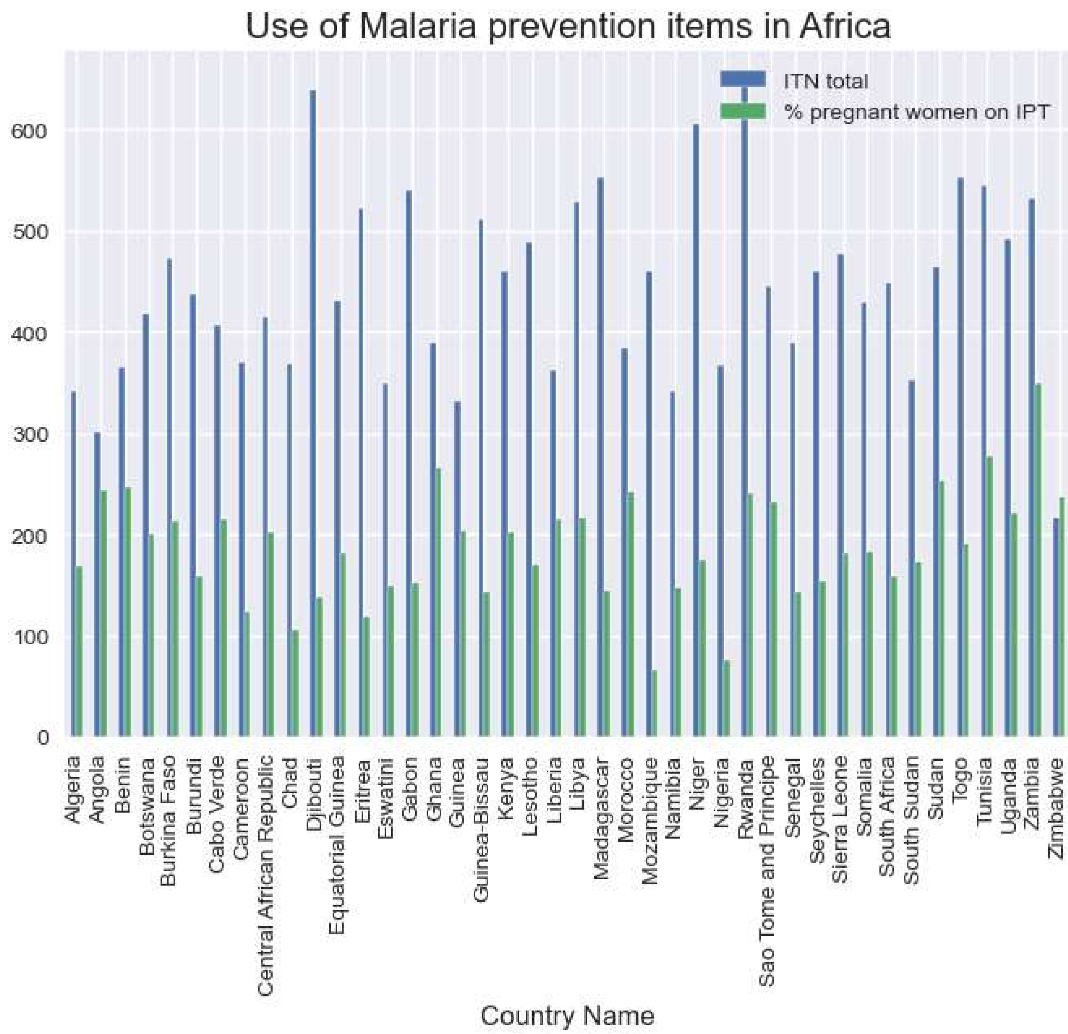


In [11]:
```
deaths=data.groupby("Country Name")["Malaria death"].sum()
deaths.plot(kind='bar', color = 'blue')
plt.title('Total Malaria Deaths in Africa', fontsize=16)
plt.show()
#deaths due to malaria has been eliminated in some parts of Africa
```

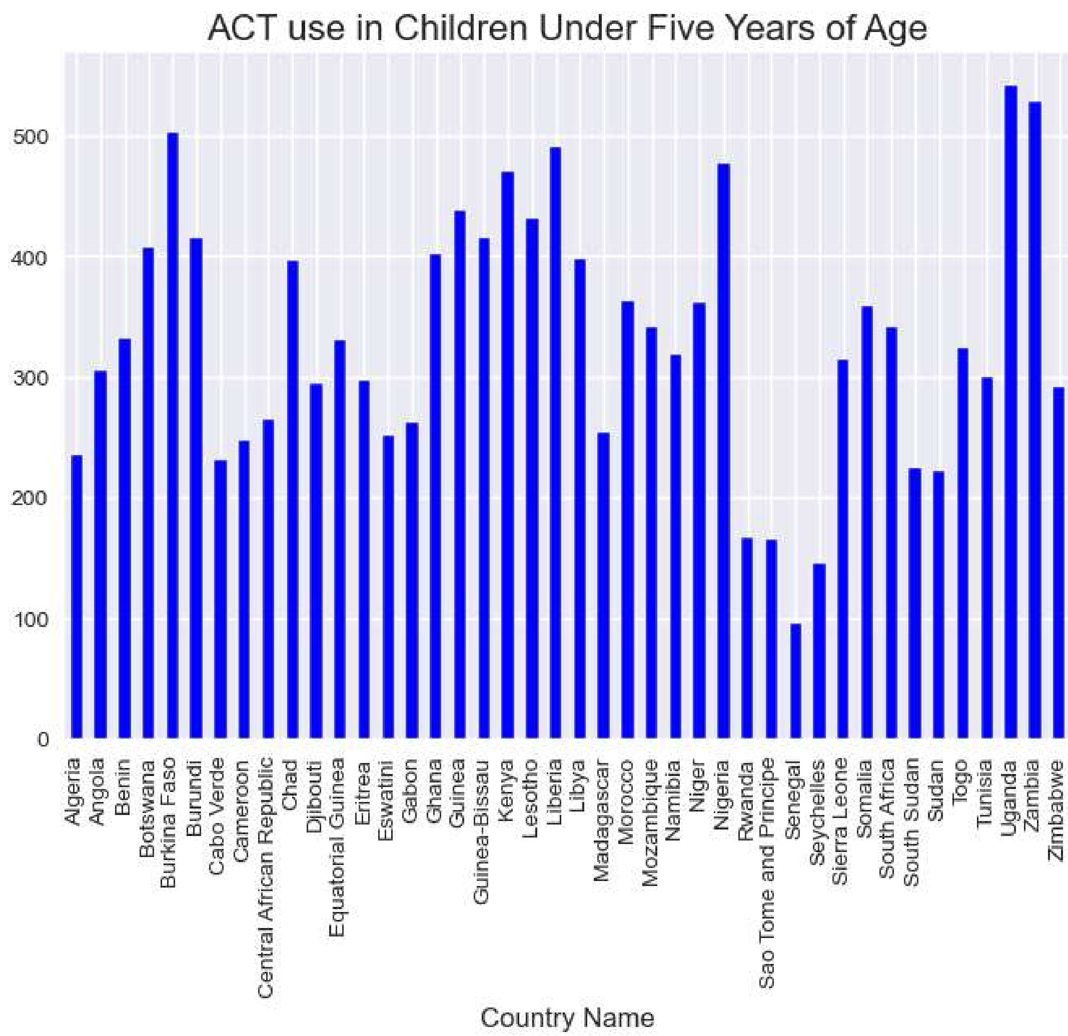


In [12]:
```
ITNUse=data.groupby("Country Name")["ITN total", "% pregnant women on IPT"].sum()
ITNUse.plot(kind='bar')
plt.title('Use of Malaria prevention items in Africa', fontsize=16)
plt.show()
```

```
C:\Users\Chinenye Claire\AppData\Local\Temp\ipykernel_10516\4274846624.py:1: FutureWarning: Indexing w
ith multiple keys (implicitly converted to a tuple of keys) will be deprecated, use a list instead.
  ITNUse=data.groupby("Country Name")["ITN total", "% pregnant women on IPT"].sum()
```

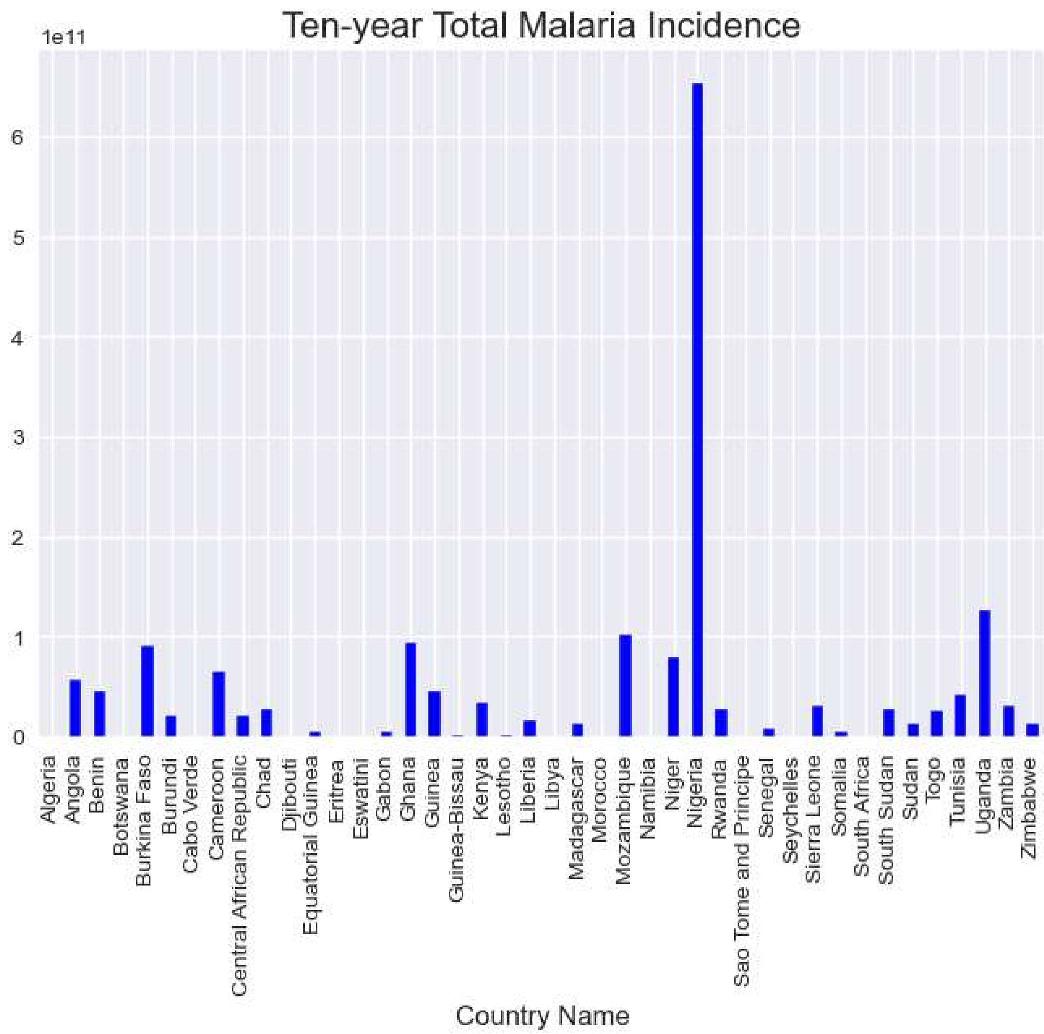


In [13]:
```
Treated_Children=data.groupby("Country Name")["% under 5 children on ACT"].sum()
Treated_Children.plot(kind='bar', color = 'blue')
plt.title('ACT use in Children Under Five Years of Age', fontsize=16)
plt.show()
```



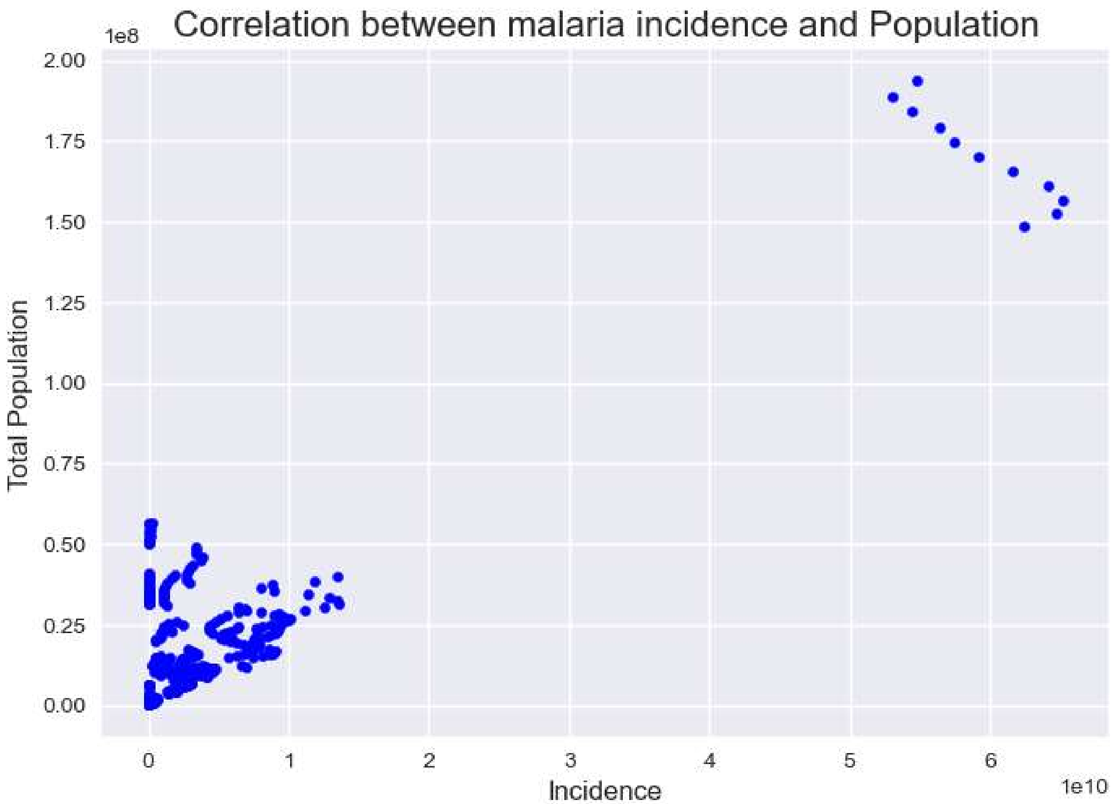
In [74]:
```
#engineering a new feature
data['Incidence']=data['incidence rate'] * data['Total Population']
```

In [75]:
```
National_Malaria_Incidence=data.groupby("Country Name")["Incidence"].sum()
National_Malaria_Incidence.plot(kind='bar', color = 'blue')
plt.title('Ten-year Total Malaria Incidence', fontsize=16)
plt.show()
#Nigeria, Uganda and Mozambique bear the highest burden of malaria in Africa
#Burkina Faso evidently has malaria but have reporting issues
```



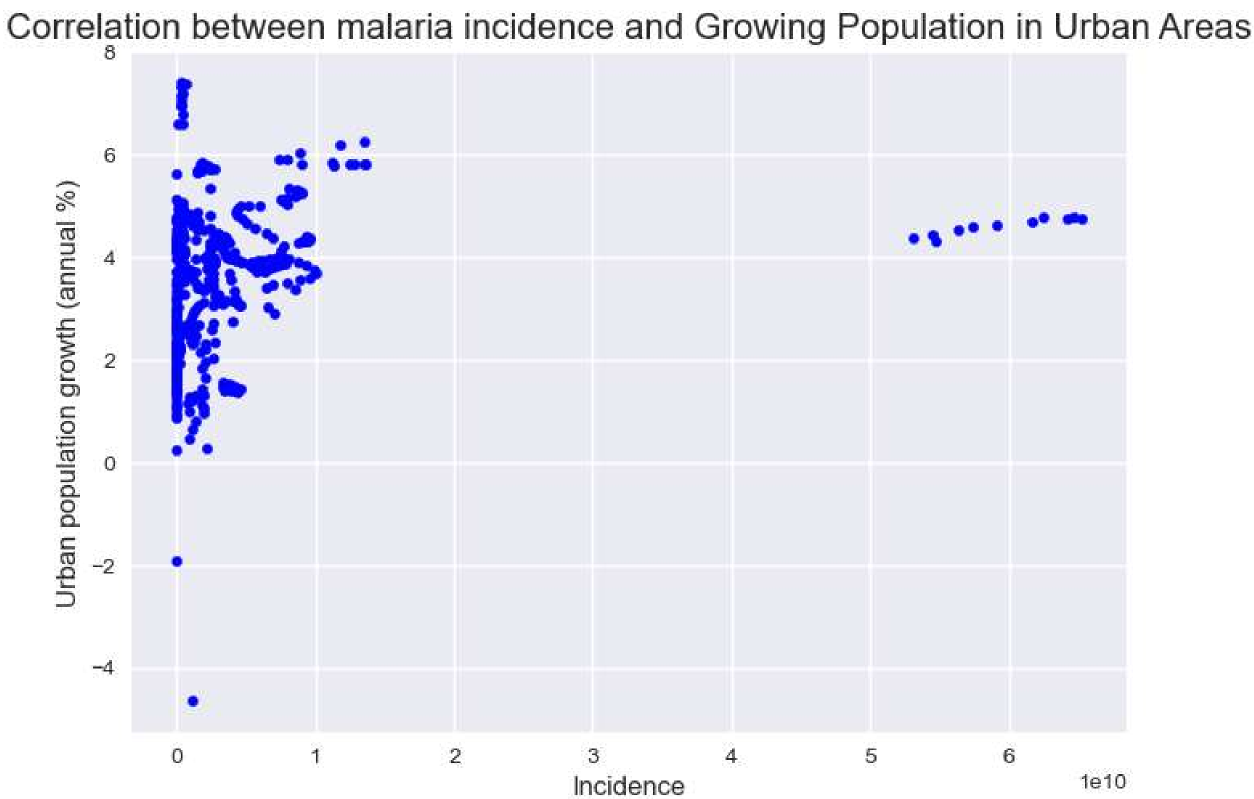
In [88]:
```
#Exploratory factor analysis
#Correlation between malaria incidence and population
plt.figure(figsize=[3, 2])
data.plot.scatter(x='Incidence', y='Total Population', color = 'blue')
plt.title('Correlation between malaria incidence and Population', fontsize=16)
plt.show()
#there is some correlation between total malaria incidence and the population in African countries
```

```
<Figure size 300x200 with 0 Axes>
```



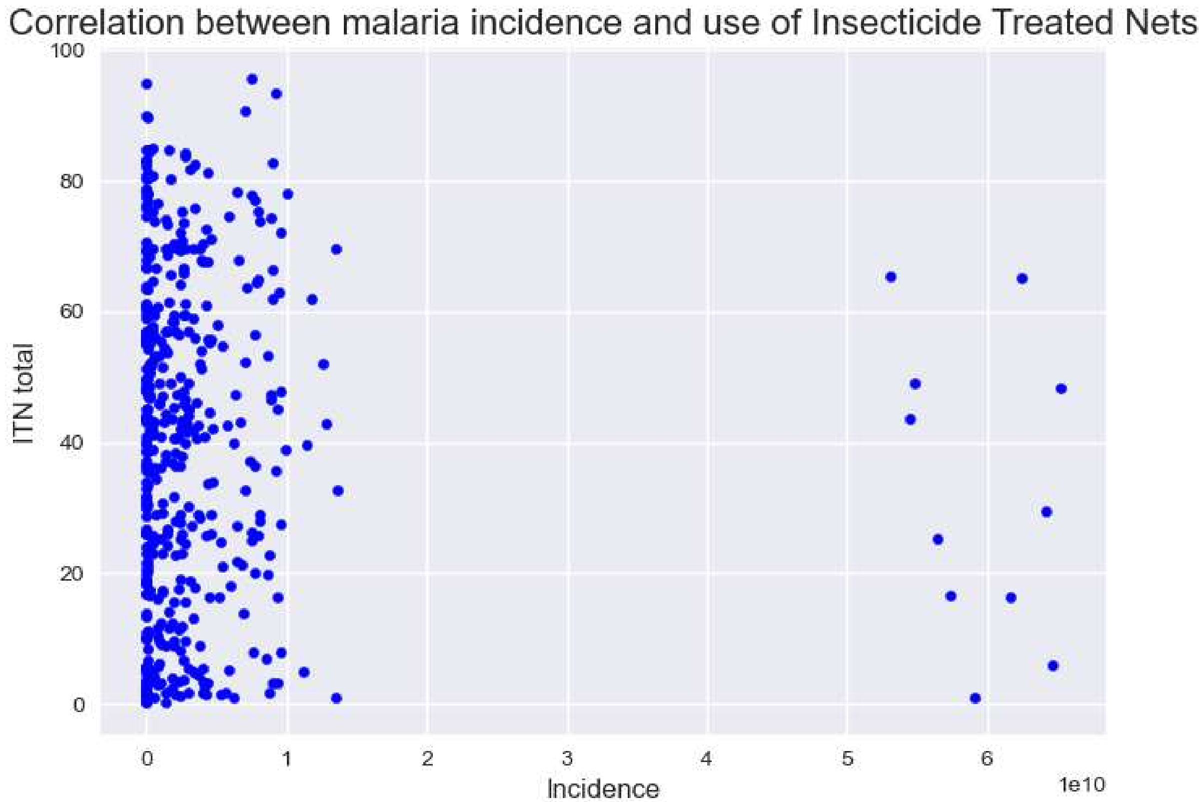
In [89]:
```
plt.figure(figsize=[3, 2])
data.plot.scatter(x='Incidence', y='Urban population growth (annual %)', color = 'blue')
plt.title('Correlation between malaria incidence and Growing Population in Urban Areas', fontsize=16)
plt.show()
#strong correlation between malaria incidence and urban population growth
```

<Figure size 300x200 with 0 Axes>



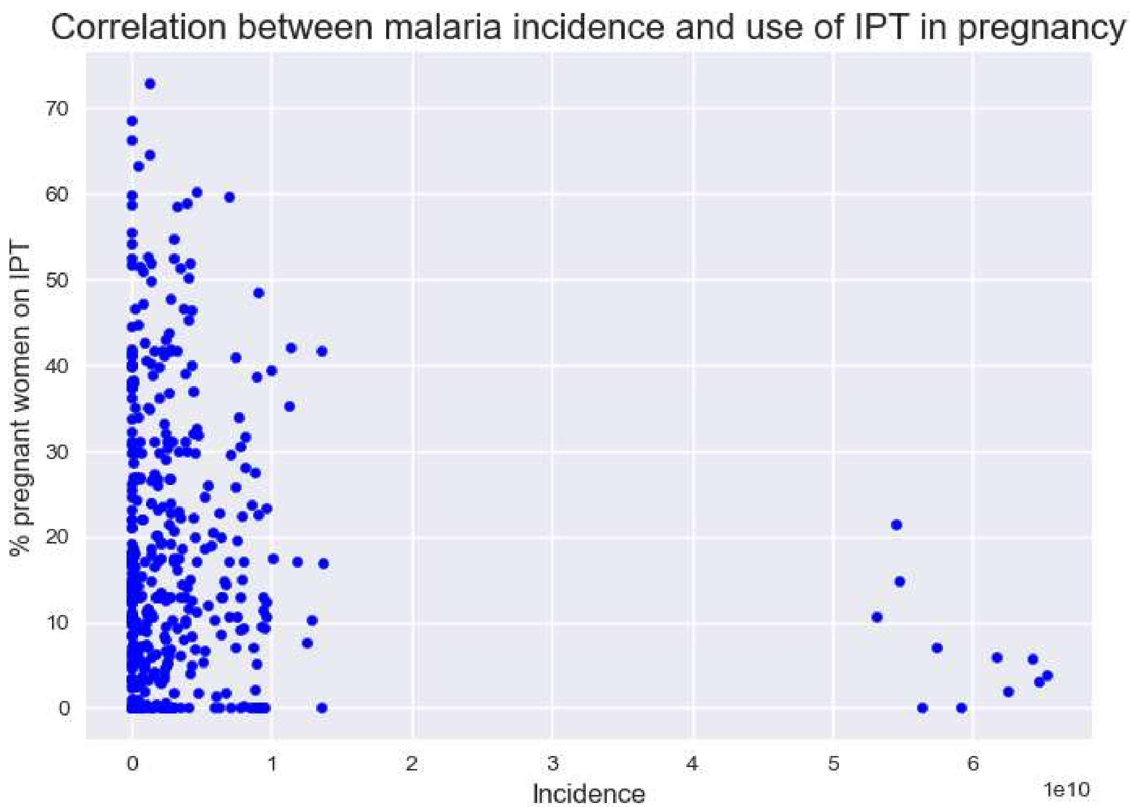
In [90]:
```
plt.figure(figsize=[3, 2])
data.plot.scatter(x='Incidence', y='ITN total', color = 'blue')
plt.title('Correlation between malaria incidence and use of Insecticide Treated Nets', fontsize=16)
plt.show()
#strong correlation between malaria incidence and use of ITNs
```

```
<Figure size 300x200 with 0 Axes>
```



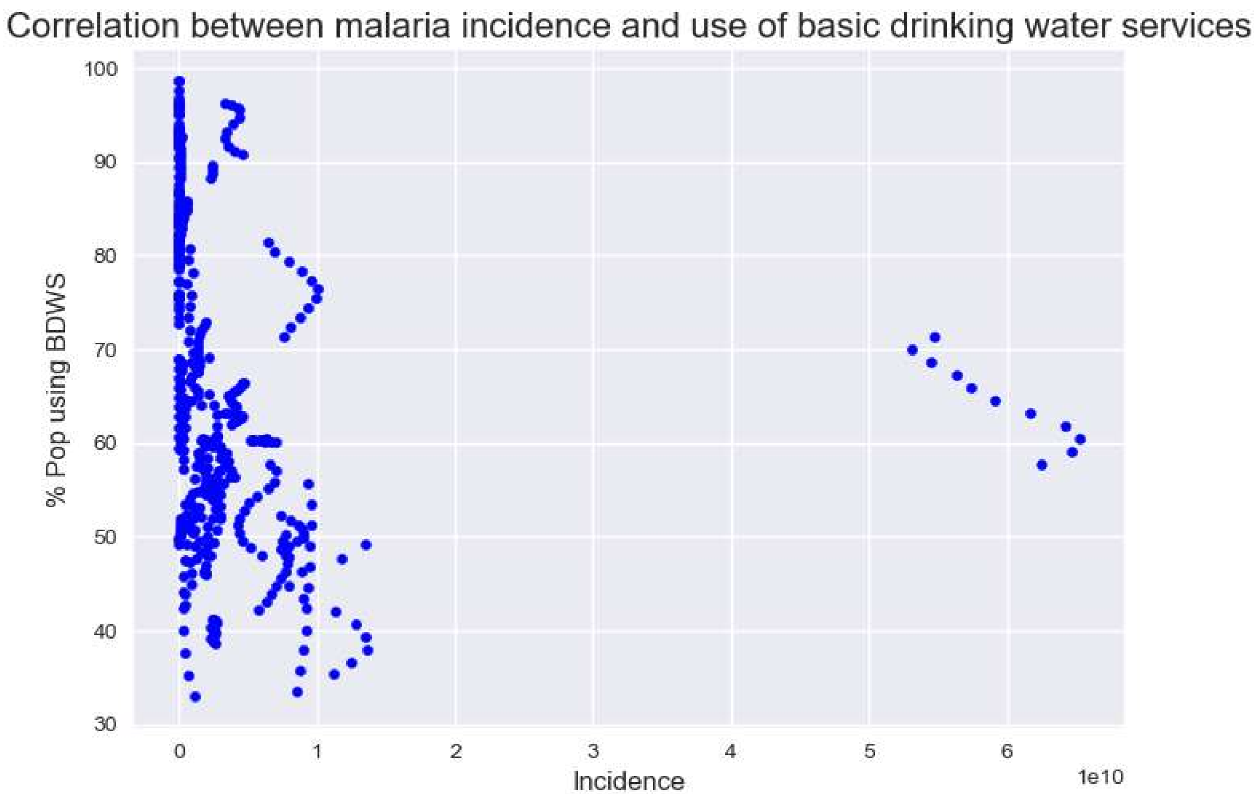
In [91]:
```
plt.figure(figsize=[3, 2])
data.plot.scatter(x='Incidence', y='% pregnant women on IPT', color = 'blue')
plt.title('Correlation between malaria incidence and use of IPT in pregnancy', fontsize=16)
plt.show()
#strong correlation between malaria incidence and use of IPTs in pregnant women
```

```
<Figure size 300x200 with 0 Axes>
```



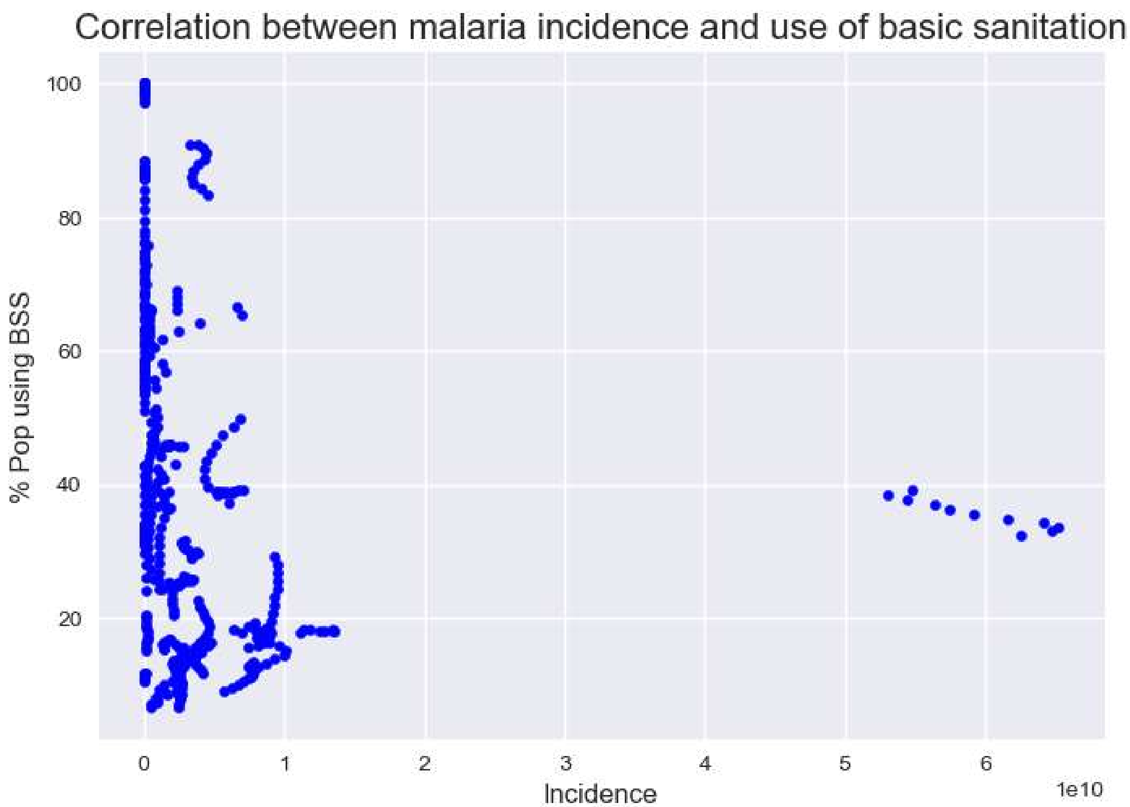
In [92]:
```
plt.figure(figsize=[3, 2])
data.plot.scatter(x='Incidence', y='% Pop using BDWS', color = 'blue')
plt.title('Correlation between malaria incidence and use of basic drinking water services', fontsize=16
plt.show()
#strong correlation between malaria incidence and use of basic drinking water
```

```
<Figure size 300x200 with 0 Axes>
```



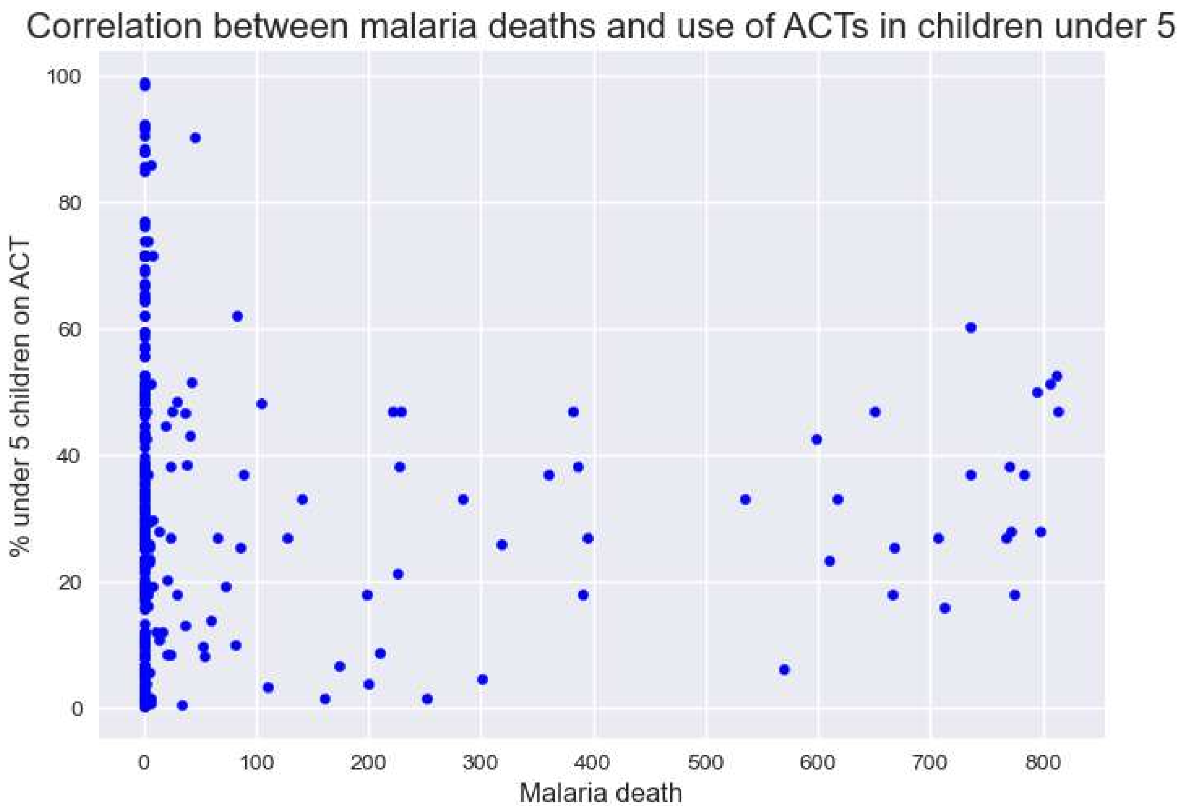
In [93]:
```
plt.figure(figsize=[3, 2])
data.plot.scatter(x='Incidence', y='% Pop using BSS', color = 'blue')
plt.title('Correlation between malaria incidence and use of basic sanitation', fontsize=16)
plt.show()
#strong correlation between malaria incidence and use of basic sanitation
```

```
<Figure size 300x200 with 0 Axes>
```



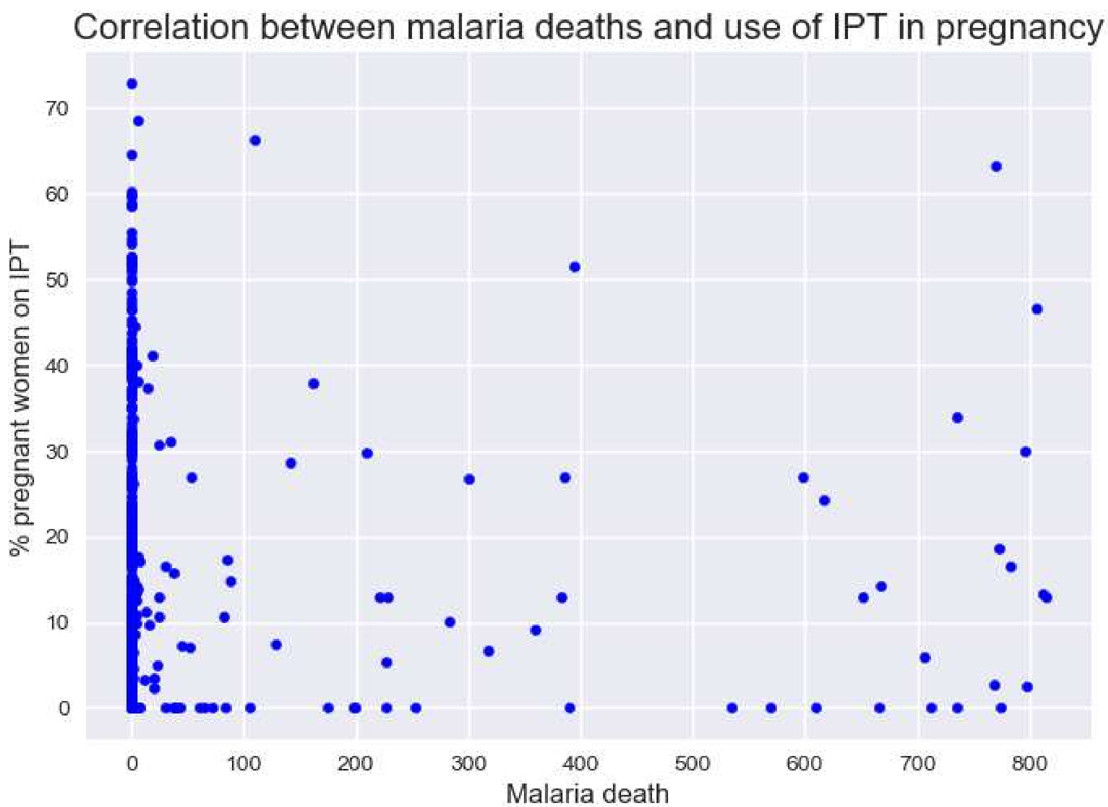
In [94]:
```
plt.figure(figsize=[3, 2])
data.plot.scatter(x='Malaria death', y='% under 5 children on ACT', color = 'blue')
plt.title('Correlation between malaria deaths and use of ACTs in children under 5', fontsize=16)
plt.show()
#minimal correlation between malaria deaths and administration of ACTs in children under the age of five
```

<Figure size 300x200 with 0 Axes>



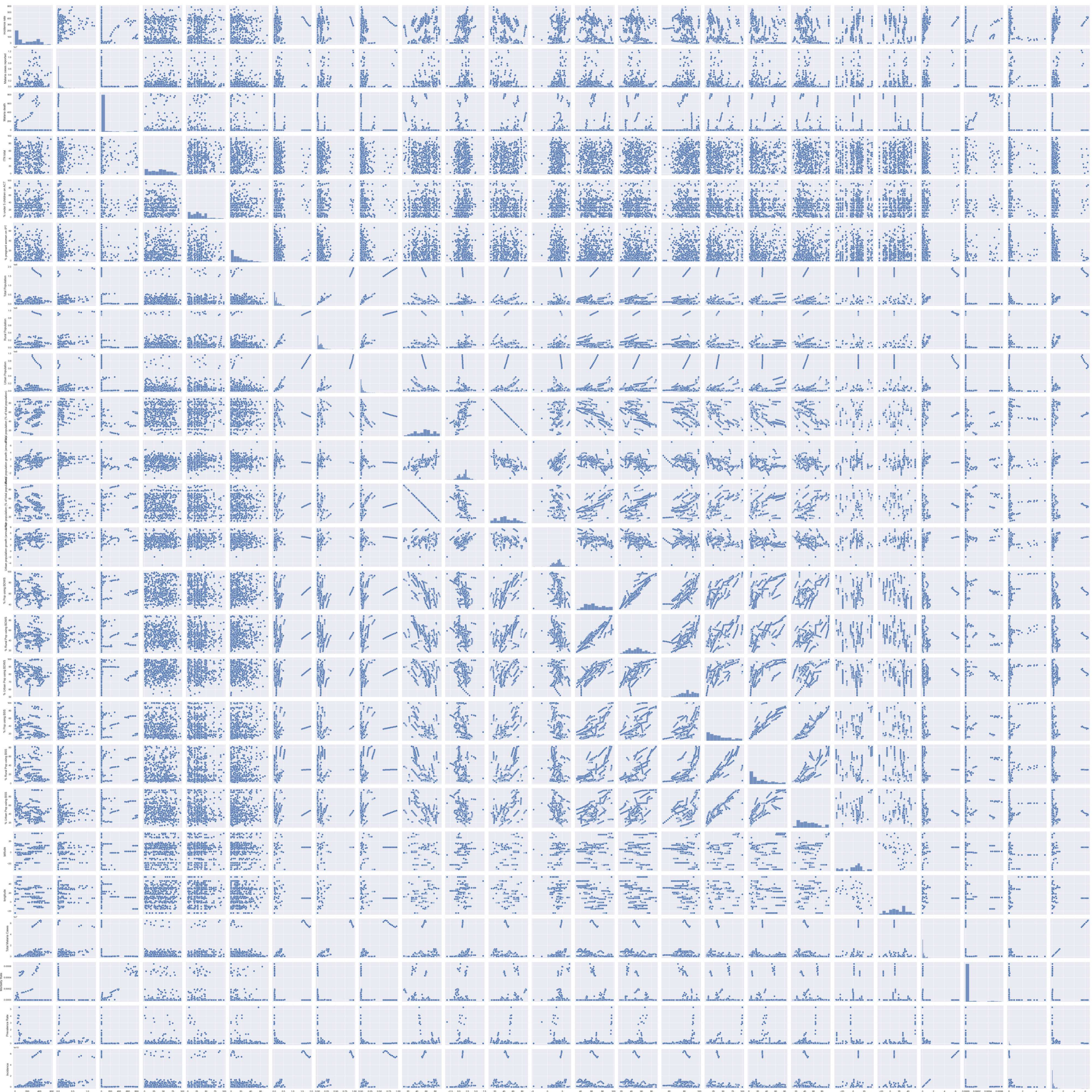
In [95]:
```
plt.figure(figsize=[3, 2])
data.plot.scatter(x='Malaria death', y='% pregnant women on IPT', color = 'blue')
plt.title('Correlation between malaria deaths and use of IPT in pregnancy', fontsize=16)
plt.show()
#minimal correlation between malaria deaths and use of IPT in pregnancy
```

```
<Figure size 300x200 with 0 Axes>
```



In [97]:
```
sns.pairplot(data=data)
plt.show
```

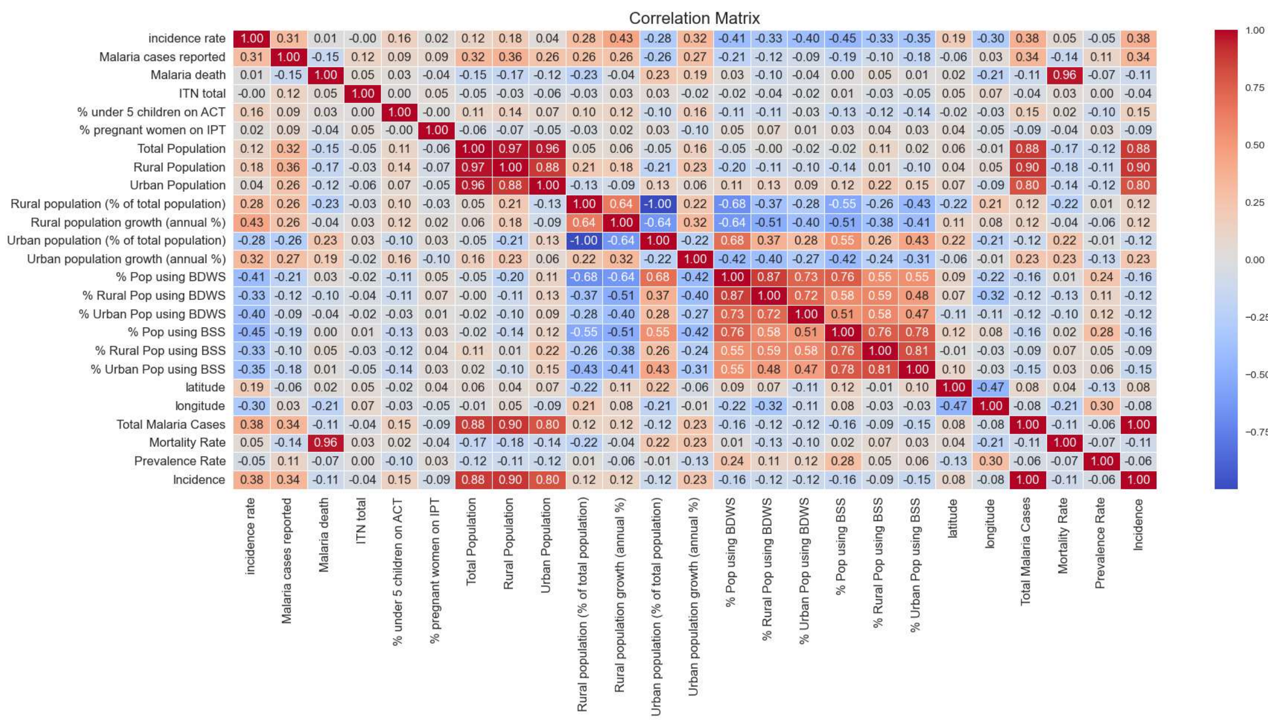
Out[97]: <function matplotlib.pyplot.show(close=None, block=None)>



In [56]:
```
corr = data.corr()
plt.figure(figsize=[20, 8])
sns.heatmap(corr, annot=True, cmap='coolwarm', fmt=".2f", linewidths=0.5)
plt.title('Correlation Matrix', fontsize=16)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.show();
```

```
C:\Users\Chinenye Claire\AppData\Local\Temp\ipykernel_7848\2754446453.py:1: FutureWarning:

The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will defaul
t to False. Select only valid columns or specify the value of numeric_only to silence this warning.
```



In [ ]:
```
#some variables are quite highly correlated
#total malaria cases has 100% correlation with the malaria incidence. It makes sense; incidence is the r
#malaria incidence has a very high correlation with total, rural and urban populations
#we will do a Confirmatory Factor Analysis to detect the structure of the relationship between the varia
#We will not be using all variables
```

In [105]:
```
x=data[['Incidence', 'ITN total', '% pregnant women on IPT', '% Pop using BSS', '% Pop using BDWS', 'To
```

In [106]:
```
x.head()
```

Out[106]:

	Incidence	ITN total	% pregnant women on IPT	% Pop using BSS	% Pop using BDWS	Total Population	Rural Population	Rural population growth (annual %)	Urban Population	Urban population growth (annual %)
0	3.398383e+05	4.7625	19.163636	85.85	91.68	33983827.0	11776076.0	-0.60	22207751.0	2.71
1	5.995225e+09	18.0000	1.500000	37.26	47.96	20909684.0	8881597.0	1.91	12028087.0	5.01
2	4.153001e+09	2.8125	15.000000	11.80	63.78	8647761.0	5053924.0	1.99	3593837.0	4.09
3	2.025986e+06	21.6500	8.600000	61.60	78.89	1966977.0	827547.0	-1.44	1139430.0	4.80
4	7.434614e+09	24.9200	7.000000	15.60	52.27	14757074.0	11363537.0	2.16	3393537.0	5.91

In [26]:
```
pip install factor_analyzer
```

```
Requirement already satisfied: factor_analyzer in c:\users\chinenye claire\anaconda3\lib\site-packages
(0.5.0)
Requirement already satisfied: pandas in c:\users\chinenye claire\anaconda3\lib\site-packages (from fa
ctor_analyzer) (1.5.3)
Requirement already satisfied: numpy in c:\users\chinenye claire\anaconda3\lib\site-packages (from fac
tor_analyzer) (1.23.5)
Requirement already satisfied: scikit-learn in c:\users\chinenye claire\anaconda3\lib\site-packages (f
rom factor_analyzer) (1.2.1)
Requirement already satisfied: pre-commit in c:\users\chinenye claire\anaconda3\lib\site-packages (fro
m factor_analyzer) (3.3.3)
Requirement already satisfied: scipy in c:\users\chinenye claire\anaconda3\lib\site-packages (from fac
tor_analyzer) (1.10.0)
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\chinenye claire\anaconda3\lib\site-p
ackages (from pandas->factor_analyzer) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\chinenye claire\anaconda3\lib\site-packages (f
rom pandas->factor_analyzer) (2022.7)
Requirement already satisfied: nodeenv>=0.11.1 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from pre-commit->factor_analyzer) (1.8.0)
Requirement already satisfied: identify>=1.0.0 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from pre-commit->factor_analyzer) (2.5.26)
Requirement already satisfied: cfgv>=2.0.0 in c:\users\chinenye claire\anaconda3\lib\site-packages (fr
om pre-commit->factor_analyzer) (3.4.0)
Requirement already satisfied: virtualenv>=20.10.0 in c:\users\chinenye claire\anaconda3\lib\site-pack
ages (from pre-commit->factor_analyzer) (20.24.3)
Requirement already satisfied: pyyaml>=5.1 in c:\users\chinenye claire\anaconda3\lib\site-packages (fr
om pre-commit->factor_analyzer) (6.0)
Requirement already satisfied: joblib>=1.1.1 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from scikit-learn->factor_analyzer) (1.1.1)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\chinenye claire\anaconda3\lib\site-pac
kages (from scikit-learn->factor_analyzer) (2.2.0)
Requirement already satisfied: setuptools in c:\users\chinenye claire\anaconda3\lib\site-packages (fro
m nodeenv>=0.11.1->pre-commit->factor_analyzer) (65.6.3)
Requirement already satisfied: six>=1.5 in c:\users\chinenye claire\anaconda3\lib\site-packages (from
python-dateutil>=2.8.1->pandas->factor_analyzer) (1.16.0)
Requirement already satisfied: distlib<1,>=0.3.7 in c:\users\chinenye claire\anaconda3\lib\site-packag
es (from virtualenv>=20.10.0->pre-commit->factor_analyzer) (0.3.7)
Requirement already satisfied: filelock<4,>=3.12.2 in c:\users\chinenye claire\anaconda3\lib\site-pack
ages (from virtualenv>=20.10.0->pre-commit->factor_analyzer) (3.12.2)
Requirement already satisfied: platformdirs<4,>=3.9.1 in c:\users\chinenye claire\anaconda3\lib\site-p
ackages (from virtualenv>=20.10.0->pre-commit->factor_analyzer) (3.10.0)
Note: you may need to restart the kernel to use updated packages.
```

In [28]:
```
from factor_analyzer import FactorAnalyzer
```

In [107]:
```
#Bartlett’s test of sphericity to check whether or not the observed variables intercorrelate at all usi
from factor_analyzer.factor_analyzer import calculate_bartlett_sphericity
chi_square_value,p_value=calculate_bartlett_sphericity(x)
chi_square_value, p_value
#The chi-square value is a measure of the difference between the observed correlation matrix and the id
#Since the p test statistic is less than 0.05, we can conclude that correlation is present among the va
```

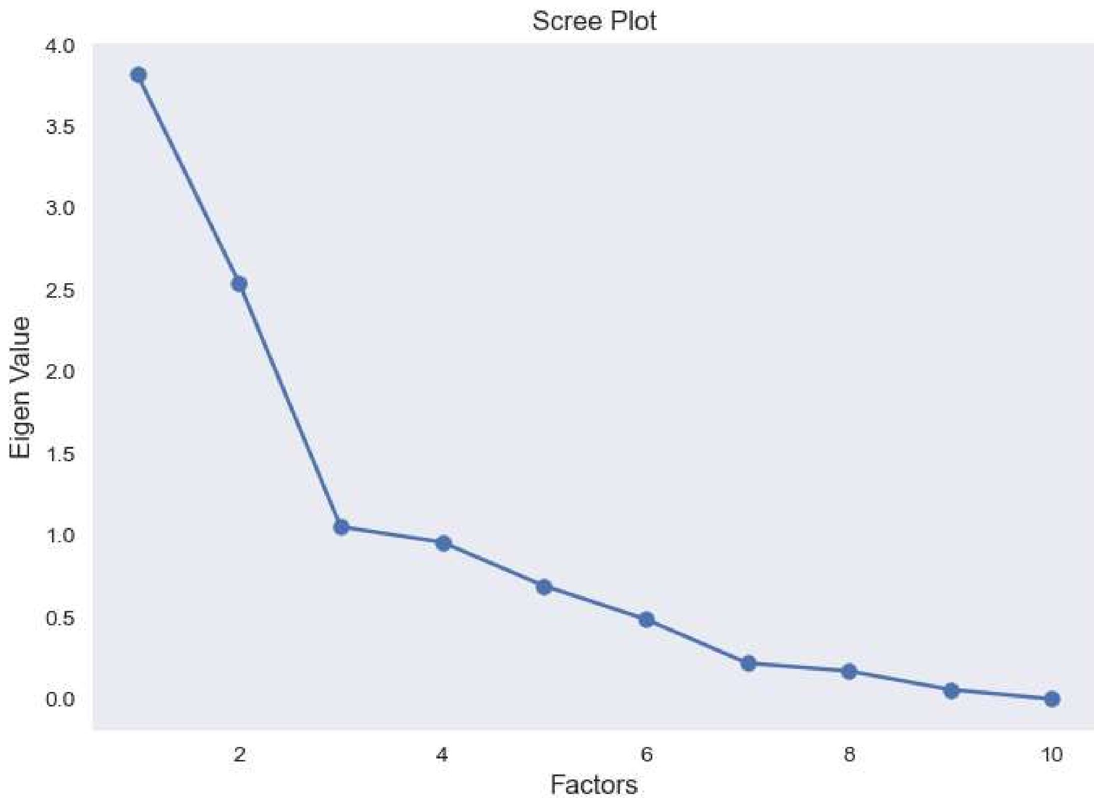
Out[107]:
```
(20333.00575869285, 0.0)
```

In [108]:
```
from factor_analyzer.factor_analyzer import calculate_kmo
kmo_all,kmo_model=calculate_kmo(x)
```

In [109]:
```
kmo_model
#The overall KMO for our data is 0.66, which is big. This value indicates that we can proceed with the
```

Out[109]:
```
0.6605585349842014
```

In [110]:
```
#determining the number of factors
fa = FactorAnalyzer(rotation = None,impute = "drop",n_factors=x.shape[1])
fa.fit(x)
ev,_ = fa.get_eigenvalues()
plt.scatter(range(1,x.shape[1]+1),ev)
plt.plot(range(1,x.shape[1]+1),ev)
plt.title('Scree Plot')
plt.xlabel('Factors')
plt.ylabel('Eigen Value')
plt.grid()
#Create scree plot using matplotlib
```



In [111]:
```
#only 3-factors eigenvalues are greater than one. It means we need to choose only 3 factors (or unobserv
# Create factor analysis object and perform factor analysis
fa = FactorAnalyzer(n_factors=3,rotation='varimax')
fa.fit(x)
print(pd.DataFrame(fa.loadings_,index=x.columns))
#Loadings indicate how much a factor explains a variable. The loading score will range from -1 to 1.Valu
```

```
                                        0         1         2
Incidence                        0.860020 -0.172282  0.193023
ITN total                       -0.030081 -0.015036 -0.129700
% pregnant women on IPT         -0.032664  0.041556 -0.276231
% Pop using BSS                  0.011231  0.815405 -0.096682
% Pop using BDWS                -0.024358  0.923799  0.045686
Total Population                 0.997166 -0.030892  0.098884
Rural Population                 0.967359 -0.194548  0.068070
Rural population growth (annual %)  0.054400 -0.676904 -0.175426
Urban Population                 0.937931  0.151935  0.131032
Urban population growth (annual %)  0.117983 -0.486599  0.225908
```

In [ ]:
```
#the higher a factor loading, the more important a variable is for said factor. A loading cutoff of 0.5
1. Population: Total Population, Rural Population and Urban Population
2. Interventions: % Pop using BSS, % Pop using BDWS
```

In [114]:
```
print(pd.DataFrame(fa.get_factor_variance(),index=['Variance','Proportional Var','Cumulative Var']))
#the 3 factors together are able to explain 61.3% of the total variance.
```

```
                         0         1         2
Variance          3.569044  2.306789  0.255208
Proportional Var  0.356904  0.230679  0.025521
Cumulative Var    0.356904  0.587583  0.613104
```

In [115]:
```
#The proportion of each variable's variance that is explained by the factors
print(pd.DataFrame(fa.get_communalities(),index=x.columns,columns=['Communalities']))
#only the same variables have over 0.5 communalities
```

```
                                    Communalities
Incidence                                0.806574
ITN total                                0.017953
% pregnant women on IPT                  0.079097
% Pop using BSS                          0.674359
% Pop using BDWS                         0.856085
Total Population                         1.005073
Rural Population                         0.978266
Rural population growth (annual %)       0.491933
Urban Population                         0.919968
Urban population growth (annual %)       0.301733
```

In [118]:
```
!pip install pingouin
import pingouin as pg
```

```
Collecting pingouin
  Downloading pingouin-0.5.3-py3-none-any.whl (198 kB)
     ------------------------------------ 198.6/198.6 kB 131.0 kB/s eta 0:00:00
Requirement already satisfied: matplotlib>=3.0.2 in c:\users\chinenye claire\anaconda3\lib\site-packag
es (from pingouin) (3.7.0)
Collecting outdated
  Downloading outdated-0.2.2-py2.py3-none-any.whl (7.5 kB)
Requirement already satisfied: scipy>=1.7 in c:\users\chinenye claire\anaconda3\lib\site-packages (fro
m pingouin) (1.10.0)
Requirement already satisfied: scikit-learn in c:\users\chinenye claire\anaconda3\lib\site-packages (f
rom pingouin) (1.2.1)
Requirement already satisfied: tabulate in c:\users\chinenye claire\anaconda3\lib\site-packages (from
pingouin) (0.8.10)
Requirement already satisfied: pandas>=1.0 in c:\users\chinenye claire\anaconda3\lib\site-packages (fr
om pingouin) (1.5.3)
Requirement already satisfied: numpy>=1.19 in c:\users\chinenye claire\anaconda3\lib\site-packages (fr
om pingouin) (1.23.5)
Requirement already satisfied: seaborn>=0.11 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from pingouin) (0.12.2)
Requirement already satisfied: statsmodels>=0.13 in c:\users\chinenye claire\anaconda3\lib\site-packag
es (from pingouin) (0.13.5)
Collecting pandas-flavor>=0.2.0
  Downloading pandas_flavor-0.6.0-py3-none-any.whl (7.2 kB)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\chinenye claire\anaconda3\lib\site-package
s (from matplotlib>=3.0.2->pingouin) (1.0.5)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\chinenye claire\anaconda3\lib\site-packag
es (from matplotlib>=3.0.2->pingouin) (4.25.0)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\chinenye claire\anaconda3\lib\site-pac
kages (from matplotlib>=3.0.2->pingouin) (2.8.2)
Requirement already satisfied: pillow>=6.2.0 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from matplotlib>=3.0.2->pingouin) (9.4.0)
Requirement already satisfied: packaging>=20.0 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from matplotlib>=3.0.2->pingouin) (22.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\chinenye claire\anaconda3\lib\site-package
s (from matplotlib>=3.0.2->pingouin) (3.0.9)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\chinenye claire\anaconda3\lib\site-packag
es (from matplotlib>=3.0.2->pingouin) (1.4.4)
Requirement already satisfied: cycler>=0.10 in c:\users\chinenye claire\anaconda3\lib\site-packages (f
rom matplotlib>=3.0.2->pingouin) (0.11.0)
Requirement already satisfied: pytz>=2020.1 in c:\users\chinenye claire\anaconda3\lib\site-packages (f
rom pandas>=1.0->pingouin) (2022.7)
Requirement already satisfied: xarray in c:\users\chinenye claire\anaconda3\lib\site-packages (from pa
ndas-flavor>=0.2.0->pingouin) (2022.11.0)
Requirement already satisfied: patsy>=0.5.2 in c:\users\chinenye claire\anaconda3\lib\site-packages (f
rom statsmodels>=0.13->pingouin) (0.5.3)
Requirement already satisfied: requests in c:\users\chinenye claire\anaconda3\lib\site-packages (from
outdated->pingouin) (2.28.1)
Collecting littleutils
  Downloading littleutils-0.2.2.tar.gz (6.6 kB)
  Preparing metadata (setup.py): started
  Preparing metadata (setup.py): finished with status 'done'
Requirement already satisfied: setuptools>=44 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from outdated->pingouin) (65.6.3)
Requirement already satisfied: joblib>=1.1.1 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from scikit-learn->pingouin) (1.1.1)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\chinenye claire\anaconda3\lib\site-pac
kages (from scikit-learn->pingouin) (2.2.0)
Requirement already satisfied: six in c:\users\chinenye claire\anaconda3\lib\site-packages (from patsy
>=0.5.2->statsmodels>=0.13->pingouin) (1.16.0)
Requirement already satisfied: idna<4,>=2.5 in c:\users\chinenye claire\anaconda3\lib\site-packages (f
rom requests->outdated->pingouin) (3.4)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\chinenye claire\anaconda3\lib\site-packa
ges (from requests->outdated->pingouin) (2022.12.7)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\chinenye claire\anaconda3\lib\site-pa
ckages (from requests->outdated->pingouin) (1.26.14)
Requirement already satisfied: charset-normalizer<3,>=2 in c:\users\chinenye claire\anaconda3\lib\site
-packages (from requests->outdated->pingouin) (2.0.4)
Building wheels for collected packages: littleutils
  Building wheel for littleutils (setup.py): started
  Building wheel for littleutils (setup.py): finished with status 'done'
```

```
  Created wheel for littleutils: filename=littleutils-0.2.2-py3-none-any.whl size=7034 sha256=66c30175
b1a031c7943ce5ce1109b2e87f8133652af34800b330ad679bd494f9
  Stored in directory: c:\users\chinenye claire\appdata\local\pip\cache\wheels\e0\3b\9c\d55ff5bc6cfbe7
0537c4731a22f2ee2462c2e5010b56ac9726
Successfully built littleutils
Installing collected packages: littleutils, outdated, pandas-flavor, pingouin
Successfully installed littleutils-0.2.2 outdated-0.2.2 pandas-flavor-0.6.0 pingouin-0.5.3
```

In [119]:
```
factor1= data[['Total Population', 'Rural Population', 'Urban Population']]
factor2= data[['% Pop using BSS', '% Pop using BDWS']]
```

In [120]:
```
factor1_alpha = pg.cronbach_alpha(factor1)
factor2_alpha = pg.cronbach_alpha(factor2)
print(factor1_alpha, factor2_alpha)
```

```
(0.9241724975523246, array([0.911, 0.935])) (0.8158867002354504, array([0.779, 0.847]))
```

In [ ]:
```
#the alphas are evaluated at 0.91 and 0.77, which indicates they are useful and coherent. we could use t
```

In [132]:
```
import numpy as np
from sklearn.linear_model import LinearRegression
```

In [136]:
```
#use of regression to determine the relationship between these variables and malaria incidence(dependent
import pickle
feature_cols= ['Total Population','Urban Population', 'Rural Population', '% Pop using BSS', '% Pop usi
x = data[feature_cols]
y = data.Incidence
lm = LinearRegression()
lm.fit(x,y)
print(lm.intercept_)
print(lm.coef_)
#RELATIONSIPS
#Urban Population and % Pop using BSS are not contributing in a positive way to malaria incidence in Afr
#Total Population, Rural Population and % Pop using BDWS have a positive influnce on malaria incidence
```

```
-3592835586.7755466
[ 1.86058089e+02 -1.42506847e+02  3.28565402e+02 -4.56556389e+07
  5.62174307e+07]
```

In [16]:
```
#Determining variables with combined effect on reducing malaria incidence in Africa using multiple linea
from sklearn import linear_model
x= data[['Rural Population', '% Pop using BDWS']]
y= data['Incidence']
regr= linear_model.LinearRegression()
regr.fit(x,y)
print(regr.coef_)
#If the total population increases by 1, malaria incidence increases by about 5 reports and if % Pop us
#A growing rural population and population of people using basic drinking water service can have a comb
```

```
[5.49244797e+02 1.13331813e+07]
```

In [19]:
```
a= data[['Total Population', 'Rural Population']]
y= data['Incidence']
regr= linear_model.LinearRegression()
regr.fit(a,y)
print(regr.coef_)
#If the total population increases by 1, malaria incidence increases by almost 40 and if rural populatio
#Total and Rural Population growth can have combinatorial positive effect on malaria incidence
```

```
[ 36.90606032 481.72252644]
```

In [20]:
```
b= data[['Total Population', '% Pop using BDWS']]
y= data['Incidence']
regr= linear_model.LinearRegression()
regr.fit(b,y)
print(regr.coef_)
#Total population growth when there is growing population of people using basic drinking water services
```

```
[ 2.93318020e+02 -6.12744687e+07]
```

In [22]:
```
#predict malaria incidence using sample values
regr.fit(x,y)
predictedIncidence = regr.predict([[100, 250]])
print(predictedIncidence)
```

```
[3.28504447e+08]

C:\Users\Chinenye Claire\anaconda3\lib\site-packages\sklearn\base.py:420: UserWarning: X does not have
valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```

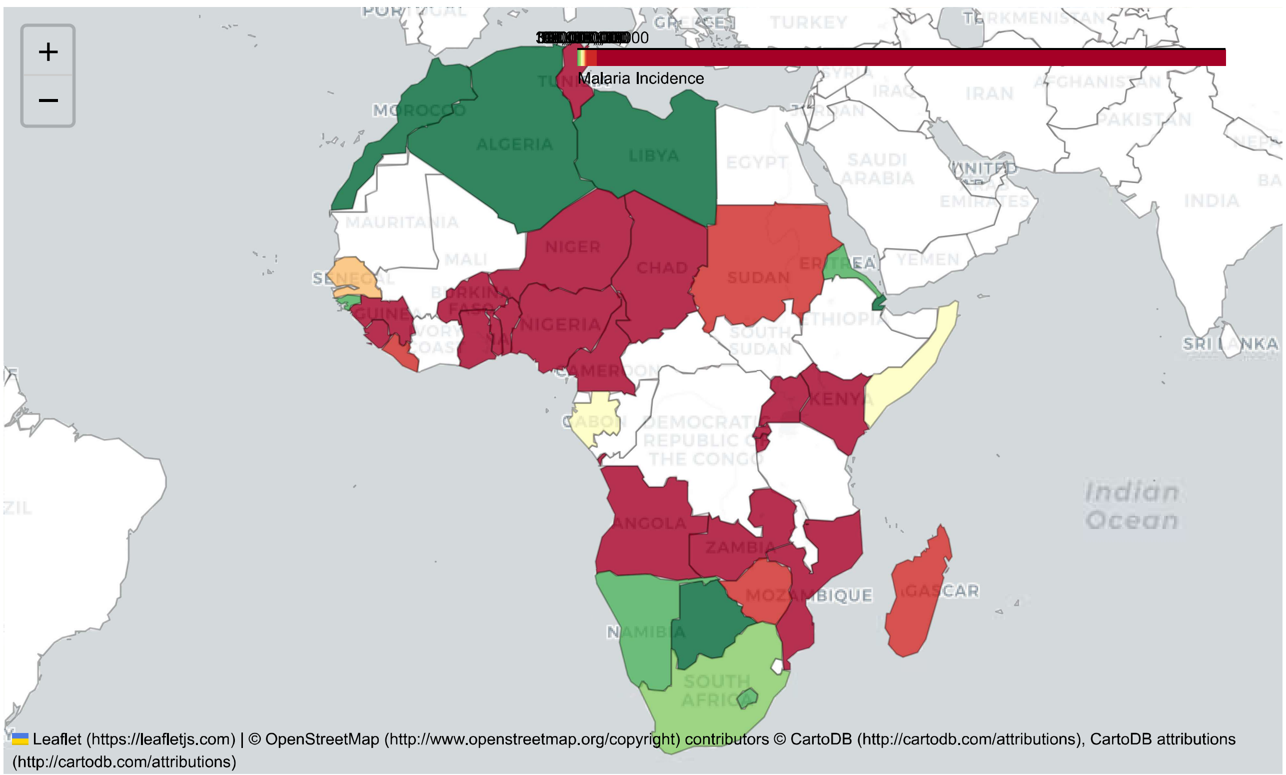
In [72]:
```
#Mapping malaria incidence to show hot spots in Africa
!pip install folium
```

```
Requirement already satisfied: folium in c:\users\chinenye claire\anaconda3\lib\site-packages (0.14.0)
Requirement already satisfied: jinja2>=2.9 in c:\users\chinenye claire\anaconda3\lib\site-packages (fr
om folium) (3.1.2)
Requirement already satisfied: branca>=0.6.0 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from folium) (0.6.0)
Requirement already satisfied: requests in c:\users\chinenye claire\anaconda3\lib\site-packages (from
folium) (2.28.1)
Requirement already satisfied: numpy in c:\users\chinenye claire\anaconda3\lib\site-packages (from fol
ium) (1.23.5)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\chinenye claire\anaconda3\lib\site-packages
(from jinja2>=2.9->folium) (2.1.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\chinenye claire\anaconda3\lib\site-packa
ges (from requests->folium) (2022.12.7)
Requirement already satisfied: charset-normalizer<3,>=2 in c:\users\chinenye claire\anaconda3\lib\site
-packages (from requests->folium) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\chinenye claire\anaconda3\lib\site-packages (f
rom requests->folium) (3.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\chinenye claire\anaconda3\lib\site-pa
ckages (from requests->folium) (1.26.14)
```

In [76]:
```
WORKING_COLS = data[["Country Name","Incidence", "Total Population", "latitude", "longitude", "geometry
```

In [78]:
```
import folium as folium
max_Malaria_Incidence = WORKING_COLS["Incidence"].max()
political_countries_url = (
    "http://geojson.xyz/naturalearth-3.3.0/ne_50m_admin_0_countries.geojson"
)
m= folium.Map(location=[WORKING_COLS["latitude"].mean(), WORKING_COLS["longitude"].mean()], zoom_start=
folium.Choropleth(
    geo_data=political_countries_url,
    data=WORKING_COLS,
    columns=("Country Name", "Incidence"),
    key_on="feature.properties.name",
    bins=[0, 50000000, 100000000, 200000000, 300000000, 400000000, 500000000, 600000000, 700000000, 800(
    fill_color="RdYlGn_r",
    fill_opacity=0.8,
    line_opacity=0.3,
    nan_fill_color="white",
    legend_name="Malaria Incidence",
).add_to(m)
m
```

Out[78]:



In [4]:
```
df2=pd.read_csv(r"C:\Users\Chinenye Claire\Desktop\cleaned_data (4) (1).csv")
```

In [5]:
```
df2.head()
```

Out[5]:

Country Name	Year	Incidence of malaria (per 1,000 population at risk)	Malaria cases reported	Malaria death	Malaria Confirmed Cases	Use of insecticide-treated bed nets (% of under-5 population)	Children with fever receiving antimalarial drugs (% of children under age 5 with fever)	Intermittent preventive treatment (IPT) of malaria in pregnancy (% of pregnant women)	Total Population	...	People using at least basic drinking water services, urban (% of urban population)
Algeria	2010-01-01	0.00	1.0	1	408	32.8750	NaN	6.5	35856344.0	...	95.03
Angola	2010-01-01	185.52	1682870.0	12398	1 682 870	3.2500	NaN	46.5	23364185.0	...	67.53
Benin	2010-01-01	387.76	0.0	8632	1 682 870	29.1000	NaN	46.6	9445710.0	...	76.08
Botswana	2010-01-01	1.69	1046.0	5	1 046	43.8125	NaN	0.0	2091664.0	...	95.18
Burkina Faso	2010-01-01	551.24	804539.0	35034	804 539	47.4000	35.1	5.2	16116845.0	...	77.25

ows × 28 columns

In [29]:
```
Annual_Figures=df2.groupby(['Year', 'Country Name']).sum()
print(Annual_Figures)
```

```
C:\Users\Chinenye Claire\AppData\Local\Temp\ipykernel_5224\222043580.py:1: FutureWarning: The defaul
t value of numeric_only in DataFrameGroupBy.sum is deprecated. In a future version, numeric_only wil
l default to False. Either specify numeric_only or select only columns which should be valid for the
function.
  Annual_Figures=df2.groupby(['Year', 'Country Name']).sum()

                         Incidence of malaria (per 1,000 population at risk)  \
Year       Country Name                                                        
2010-01-01 Algeria                                                    0.00   
           Angola                                                   185.52   
           Benin                                                    387.76   
           Botswana                                                   1.69   
           Burkina Faso                                             551.24   
...                                                                    ...   
2017-01-01 Sudan                                                     46.75   
           Togo                                                     278.20   
           Uganda                                                   336.76   
           Zambia                                                   160.05   
           Zimbabwe                                                 108.55   
```

In [34]:
```
Annual_Figures['Malaria Incidence']= Annual_Figures['Incidence of malaria (per 1,000 population at risk
```

In [36]:
```
Annual_Figures['Prevalence']= Annual_Figures['Malaria Incidence']*100
```

In [37]:
```
print((Annual_Figures['Prevalence']))
```

```
Year        Country Name
2010-01-01  Algeria          0.000
            Angola          18.552
            Benin           38.776
            Botswana         0.169
            Burkina Faso    55.124
                             ...
2017-01-01  Sudan            4.675
            Togo            27.820
            Uganda          33.676
            Zambia          16.005
            Zimbabwe        10.855
Name: Prevalence, Length: 288, dtype: float64
```

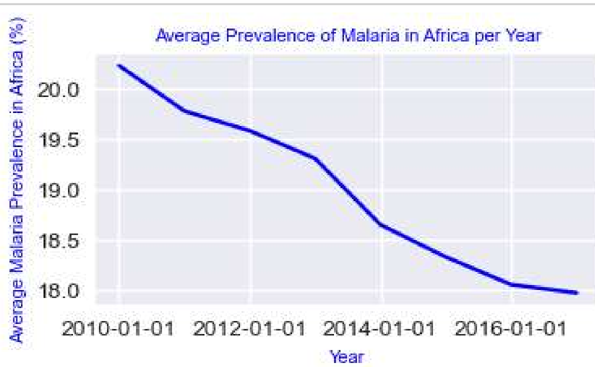
In [60]:
```
Average_Prevalence=Annual_Figures['Prevalence'].groupby(['Year']).mean()
print(Average_Prevalence)
```

```
Year
2010-01-01    20.234639
2011-01-01    19.782417
2012-01-01    19.585278
2013-01-01    19.308833
2014-01-01    18.648972
2015-01-01    18.327417
2016-01-01    18.051556
2017-01-01    17.969806
Name: Prevalence, dtype: float64
```

In [62]:
```
Average_National_Prevalence=Annual_Figures['Prevalence'].groupby(['Country Name']).mean()
print(Average_National_Prevalence)
#Average Malaria prevalence over the 10-year period considered is less than 50% in all African Countrie
```

```
Country Name
Algeria                      0.000375
Angola                      19.420875
Benin                       39.538750
Botswana                     0.096000
Burkina Faso                46.796000
Burundi                     18.398625
Cabo Verde                   0.052750
Cameroon                    25.924000
Central African Republic    39.450625
Chad                        18.636125
Djibouti                     0.990375
Equatorial Guinea           33.737625
Eritrea                      2.416250
Eswatini                     0.175000
Gabon                       25.049875
Ghana                       31.359375
Guinea                      38.227125
Guinea-Bissau               10.505125
Kenya                        7.349125
Liberia                     34.666375
Madagascar                   6.133875
Mozambique                  36.737750
Namibia                      1.322750
Niger                       39.873875
Nigeria                     32.784625
Sao Tome and Principe        2.587000
Senegal                      5.516625
Sierra Leone                42.013125
Somalia                      3.127500
South Africa                 0.167625
South Sudan                 23.921875
Sudan                        3.499750
Togo                        33.867750
Uganda                      31.087000
Zambia                      19.494250
Zimbabwe                     8.664375
Name: Prevalence, dtype: float64
```

In [95]:
```
plt.figure(figsize=(4, 2))
Average_Prevalence.plot(kind='line', color='blue')
plt.title('Average Prevalence of Malaria in Africa per Year', fontsize=8, color='blue')
plt.xlabel('Year', fontsize=8, color='blue')
plt.ylabel('Average Malaria Prevalence in Africa (%)', fontsize=8, color='blue')
plt.grid(True)
plt.show()
#Average malaria prevalence in Africa dropped over the 10-year period by over 2%
```



In [92]:
```
import folium as folium
max_Malaria_Prevalence = Average_National_Prevalence.max()
political_countries_url = (
    "http://geojson.xyz/naturalearth-3.3.0/ne_50m_admin_0_countries.geojson"
)
m= folium.Map(location=[WORKING_COLS["latitude"].mean(), WORKING_COLS["longitude"].mean()], zoom_start=
folium.Choropleth(
    geo_data=political_countries_url,
    data=Average_National_Prevalence,
    columns=("Country", "Prevalence"),
    key_on="feature.properties.name",
    bins=[0, 10, 20, 30, 40, 50, 60, 70, 80, 90, max_Malaria_Incidence],
    fill_color="RdYlGn_r",
    fill_opacity=0.8,
    line_opacity=0.3,
    nan_fill_color="white",
    legend_name="Malaria Prevalence",
).add_to(m)
m
```

Out[92]: