In [102]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [103]:
```
df=pd.read_csv("C:/Users/zoaah/Downloads/world_population.csv")
df
```

Out[103]:

	Rank	CCA3	Country	Capital	Continent	2022 Population	2020 Population	2015 Population	Popu
0	36	AFG	Afghanistan	Kabul	Asia	41128771.00	38972230.00	33753499.00	28189
1	138	ALB	Albania	Tirana	Europe	2842321.00	2866849.00	2882481.00	2913
2	34	DZA	Algeria	Algiers	Africa	44903225.00	43451666.00	39543154.00	35856
3	213	ASM	American Samoa	Pago Pago	Oceania	44273.00	46189.00	51368.00	54
4	203	AND	Andorra	Andorra la Vella	Europe	79824.00	77700.00	71746.00	71
...	...	...	...	...	...	...	...	...	
229	226	WLF	Wallis and Futuna	Mata-Utu	Oceania	11572.00	11655.00	12182.00	13
230	172	ESH	Western Sahara	El Aaiún	Africa	575986.00	556048.00	491824.00	413
231	46	YEM	Yemen	Sanaa	Asia	33696614.00	32284046.00	28516545.00	24743
232	63	ZMB	Zambia	Lusaka	Africa	20017675.00	18927715.00	NaN	13792
233	74	ZWE	Zimbabwe	Harare	Africa	16320537.00	15669666.00	14154937.00	12839

234 rows × 17 columns

In [104]:
```
pd.set_option('display.float_format',lambda x: '%.2f' % x)
```

In [105]:
```
df.head()
```

Out[105]:

	Rank	CCA3	Country	Capital	Continent	2022 Population	2020 Population	2015 Population	2 Popula
0	36	AFG	Afghanistan	Kabul	Asia	41128771.00	38972230.00	33753499.00	2818967
1	138	ALB	Albania	Tirana	Europe	2842321.00	2866849.00	2882481.00	291339
2	34	DZA	Algeria	Algiers	Africa	44903225.00	43451666.00	39543154.00	3585634
3	213	ASM	American Samoa	Pago Pago	Oceania	44273.00	46189.00	51368.00	5484
4	203	AND	Andorra	Andorra la Vella	Europe	79824.00	77700.00	71746.00	7151

In [106]:
```
df.shape
```

Out[106]:
```
(234, 17)
```

In [107]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 234 entries, 0 to 233
Data columns (total 17 columns):
 #   Column                        Non-Null Count  Dtype  
---  ------                        --------------  -----  
 0   Rank                          234 non-null    int64  
 1   CCA3                          234 non-null    object 
 2   Country                       234 non-null    object 
 3   Capital                       234 non-null    object 
 4   Continent                     234 non-null    object 
 5   2022 Population               230 non-null    float64
 6   2020 Population               233 non-null    float64
 7   2015 Population               230 non-null    float64
 8   2010 Population               227 non-null    float64
 9   2000 Population               227 non-null    float64
 10  1990 Population               229 non-null    float64
 11  1980 Population               229 non-null    float64
 12  1970 Population               230 non-null    float64
 13  Area (km²)                    232 non-null    float64
 14  Density (per km²)             230 non-null    float64
 15  Growth Rate                   232 non-null    float64
 16  World Population Percentage   234 non-null    float64
dtypes: float64(12), int64(1), object(4)
memory usage: 31.2+ KB
```

In [108]:
```
df.describe()
```

Out[108]:

	Rank	2022 Population	2020 Population	2015 Population	2010 Population	2000 Population	P
count	234.00	230.00	233.00	230.00	227.00	227.00	
mean	117.50	34632250.88	33600710.95	32066004.16	30270164.48	26840495.26	19
std	67.69	137889172.44	135873196.61	131507146.34	126074183.54	113352454.57	81
min	1.00	510.00	520.00	564.00	596.00	651.00	
25%	59.25	419738.50	406471.00	394295.00	382726.50	329470.00	
50%	117.50	5762857.00	5456681.00	5244415.00	4889741.00	4491202.00	3
75%	175.75	22653719.00	21522626.00	19730853.75	16825852.50	15625467.00	11
max	234.00	1425887337.00	1424929781.00	1393715448.00	1348191368.00	1264099069.00	1153

In [109]:
```
#to check for unique values
df.nunique()
```

Out[109]:
```
Rank                             234
CCA3                             234
Country                          234
Capital                          234
Continent                          6
2022 Population                  230
2020 Population                  233
2015 Population                  230
2010 Population                  227
2000 Population                  227
1990 Population                  229
1980 Population                  229
1970 Population                  230
Area (km²)                       231
Density (per km²)                230
Growth Rate                      178
World Population Percentage       70
dtype: int64
```

In [110]:
```
df.isnull().sum()
```

Out[110]:
```
Rank                             0
CCA3                             0
Country                          0
Capital                          0
Continent                        0
2022 Population                  4
2020 Population                  1
2015 Population                  4
2010 Population                  7
2000 Population                  7
1990 Population                  5
1980 Population                  5
1970 Population                  4
Area (km²)                       2
Density (per km²)                4
Growth Rate                      2
World Population Percentage      0
dtype: int64
```

In [111]:
```
df.count()
```

Out[111]:
```
Rank                            234
CCA3                            234
Country                         234
Capital                         234
Continent                       234
2022 Population                 230
2020 Population                 233
2015 Population                 230
2010 Population                 227
2000 Population                 227
1990 Population                 229
1980 Population                 229
1970 Population                 230
Area (km²)                      232
Density (per km²)               230
Growth Rate                     232
World Population Percentage     234
dtype: int64
```

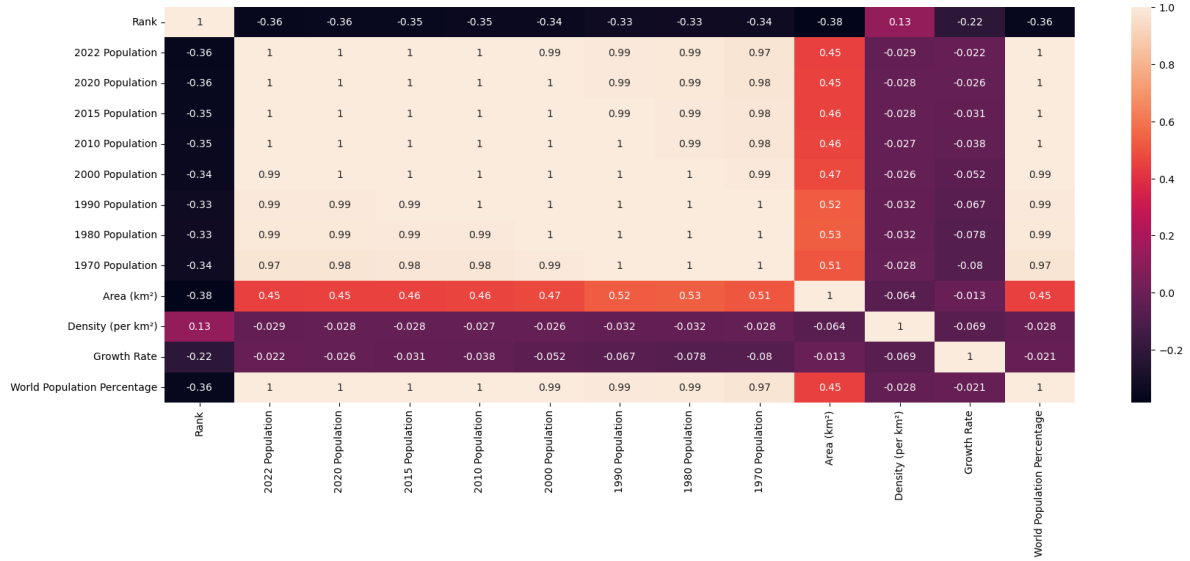
In [112]:
```
#if we want to sort the values in a particular order based on a specific columr
df.sort_values(by='2022 Population',ascending=False).head()
```

Out[112]:

	Rank	CCA3	Country	Capital	Continent	2022 Population	2020 Population	201 Populatio
41	1	CHN	China	Beijing	Asia	1425887337.00	1424929781.00	1393715448.0
92	2	IND	India	New Delhi	Asia	1417173173.00	1396387127.00	1322866505.0
221	3	USA	United States	Washington, D.C.	North America	338289857.00	335942003.00	324607776.0
93	4	IDN	Indonesia	Jakarta	Asia	275501339.00	271857970.00	259091970.0
156	5	PAK	Pakistan	Islamabad	Asia	235824862.00	227196741.00	210969298.0

In [113]:
```
#correlation
correlation=df.corr()
```

In [114]:
```
sns.heatmap(correlation,xticklabels=correlation.columns,yticklabels=correlatio
plt.show()
```



In [115]:
```
df.columns
```

Out[115]:
```
Index(['Rank', 'CCA3', 'Country', 'Capital', 'Continent', '2022 Population',
       '2020 Population', '2015 Population', '2010 Population',
       '2000 Population', '1990 Population', '1980 Population',
       '1970 Population', 'Area (km²)', 'Density (per km²)', 'Growth Rate',
       'World Population Percentage'],
      dtype='object')
```

In [116]:
```
df.groupby('Continent').mean()
```

Out[116]:

Continent	Rank	2022 Population	2020 Population	2015 Population	2010 Population	2000 Population	1990 Population
Africa	92.16	25455879.68	23871435.26	21419703.57	18898197.31	14598365.95	11376964.52
Asia	77.56	96327387.31	94955134.37	89165003.64	89087770.00	80580835.11	48639995.33
Europe	124.50	15055371.82	14915843.92	15027454.12	14712278.68	14817685.71	14785203.94
North America	160.93	15007403.40	14855914.82	14259596.25	13568016.28	12151739.60	10531660.62
Oceania	188.52	2046386.32	1910148.96	1756664.48	1613163.65	1357512.09	1162774.87
South America	97.57	31201186.29	30823574.50	29509599.71	26789395.54	25015888.69	21224743.93

In [117]:
```
df.groupby('Continent').mean().sort_values(by="2022 Population",ascending=Fals
```

Out[117]:

Continent	Rank	2022 Population	2020 Population	2015 Population	2010 Population	2000 Population	1990 Population
Asia	77.56	96327387.31	94955134.37	89165003.64	89087770.00	80580835.11	48639995.33
South America	97.57	31201186.29	30823574.50	29509599.71	26789395.54	25015888.69	21224743.93
Africa	92.16	25455879.68	23871435.26	21419703.57	18898197.31	14598365.95	11376964.52
Europe	124.50	15055371.82	14915843.92	15027454.12	14712278.68	14817685.71	14785203.94
North America	160.93	15007403.40	14855914.82	14259596.25	13568016.28	12151739.60	10531660.62
Oceania	188.52	2046386.32	1910148.96	1756664.48	1613163.65	1357512.09	1162774.87

In [118]:
```
df2=df.groupby('Continent')[df.columns[5:13].tolist()[::-1]].mean(numeric_only
df2
```

Out[118]:

Continent	1970 Population	1980 Population	1990 Population	2000 Population	2010 Population	2015 Population	Popu
Asia	43839877.83	40278333.33	48639995.33	80580835.11	89087770.00	89165003.64	94955
South America	13781939.71	17270643.29	21224743.93	25015888.69	26789395.54	29509599.71	30823
Africa	6567175.27	8586031.98	11376964.52	14598365.95	18898197.31	21419703.57	23871
Europe	13118479.82	14200004.52	14785203.94	14817685.71	14712278.68	15027454.12	14915
North America	7885865.15	9207334.03	10531660.62	12151739.60	13568016.28	14259596.25	14855
Oceania	846968.26	996532.17	1162774.87	1357512.09	1613163.65	1756664.48	1910

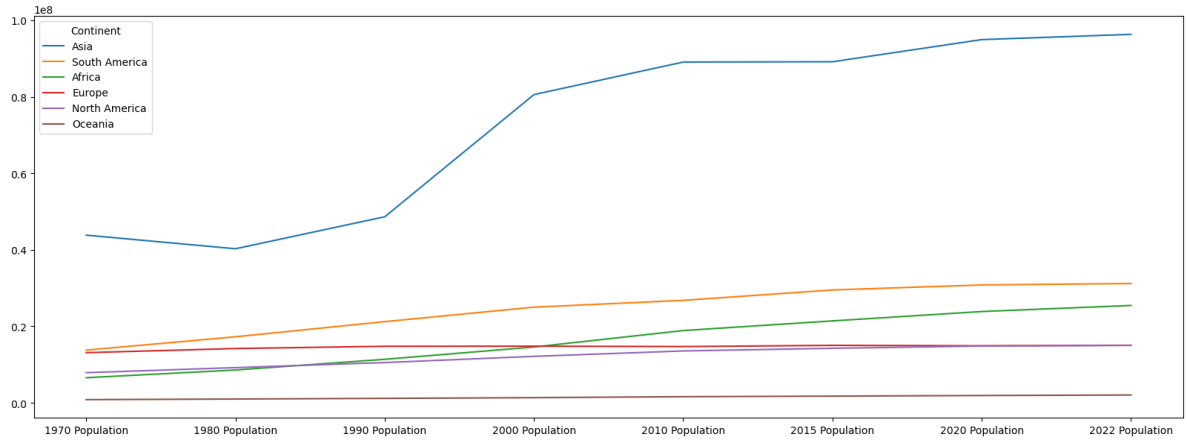
In [119]:
```
df3=df2.transpose()
df3
```

Out[119]:

Continent	Asia	South America	Africa	Europe	North America	Oceania
1970 Population	43839877.83	13781939.71	6567175.27	13118479.82	7885865.15	846968.26
1980 Population	40278333.33	17270643.29	8586031.98	14200004.52	9207334.03	996532.17
1990 Population	48639995.33	21224743.93	11376964.52	14785203.94	10531660.62	1162774.87
2000 Population	80580835.11	25015888.69	14598365.95	14817685.71	12151739.60	1357512.09
2010 Population	89087770.00	26789395.54	18898197.31	14712278.68	13568016.28	1613163.65
2015 Population	89165003.64	29509599.71	21419703.57	15027454.12	14259596.25	1756664.48
2020 Population	94955134.37	30823574.50	23871435.26	14915843.92	14855914.82	1910148.96
2022 Population	96327387.31	31201186.29	25455879.68	15055371.82	15007403.40	2046386.32

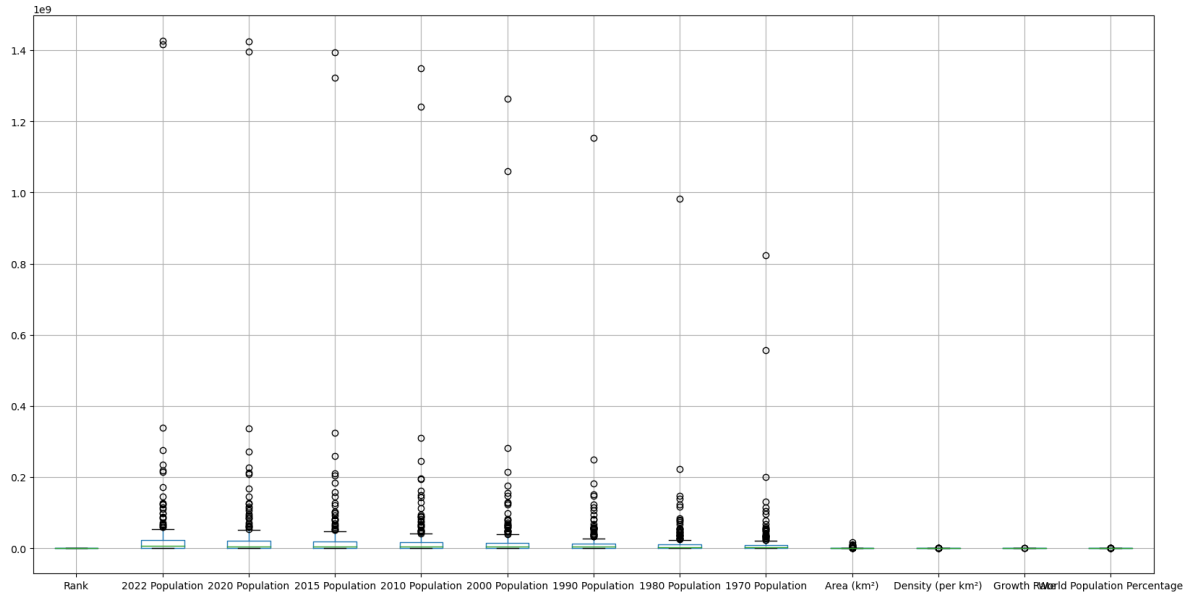
In [120]:
```
df3.plot()
```

Out[120]: <Axes: >

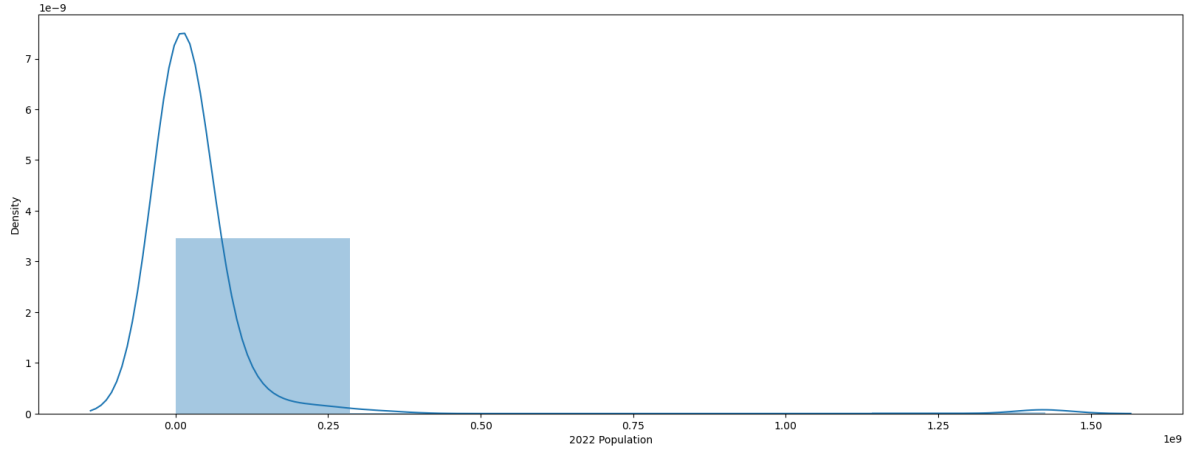


In [121]:
```
#detection of outliers
df.boxplot(figsize=(20,10))
```

Out[121]: <Axes: >

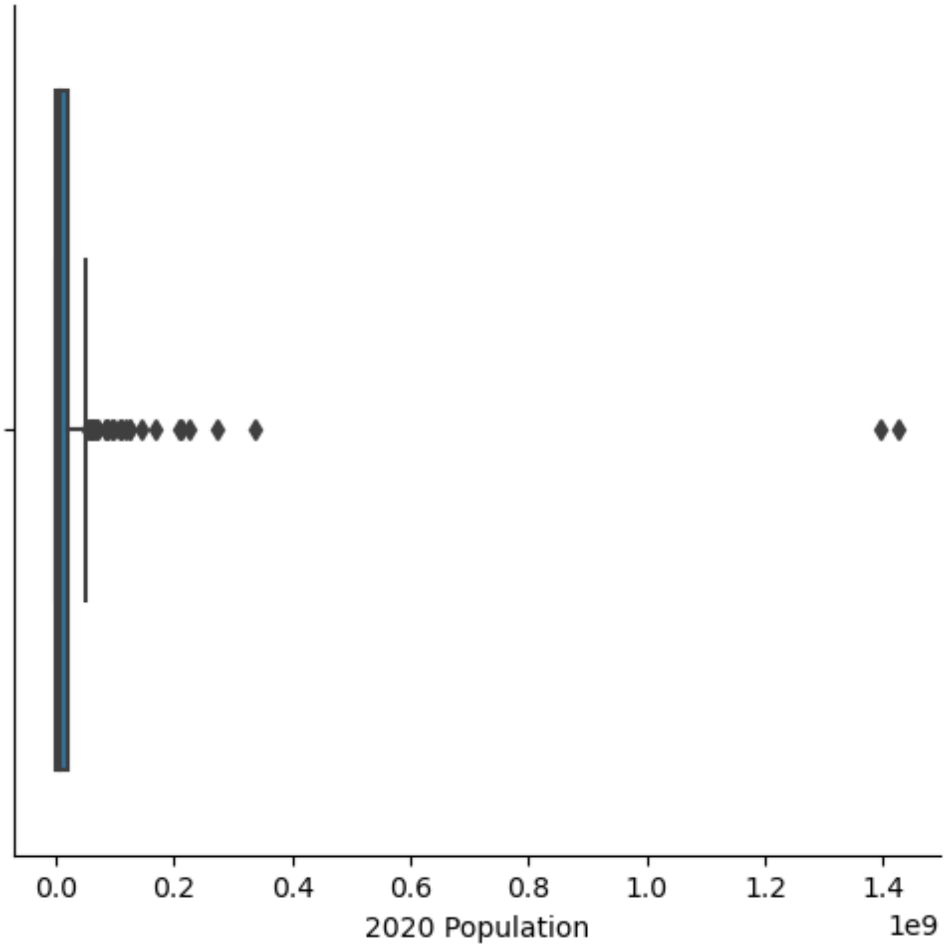


In [122]:
```
#using histograms to analyze the relationship
sns.distplot(df['2022 Population'],bins=5)
plt.show()
```



In [123]:
```
sns.catplot(x='2020 Population',kind='box',data=df)
```

Out[123]:
```
<seaborn.axisgrid.FacetGrid at 0x1a30a86cd90>
```



In [124]:
```
df.select_dtypes(include='number')
```

Out[124]:

	Rank	2022 Population	2020 Population	2015 Population	2010 Population	2000 Population	1990 Population	Pop
0	36	41128771.00	38972230.00	33753499.00	28189672.00	19542982.00	10694796.00	1248
1	138	2842321.00	2866849.00	2882481.00	2913399.00	3182021.00	3295066.00	2941
2	34	44903225.00	43451666.00	39543154.00	35856344.00	30774621.00	25518074.00	1873
3	213	44273.00	46189.00	51368.00	54849.00	58230.00	47818.00	32
4	203	79824.00	77700.00	71746.00	71519.00	66097.00	53569.00	3
...	...	...	...	...	...	...	...	
229	226	11572.00	11655.00	12182.00	13142.00	14723.00	13454.00	1
230	172	575986.00	556048.00	491824.00	413296.00	270375.00	178529.00	116
231	46	33696614.00	32284046.00	28516545.00	24743946.00	18628700.00	13375121.00	9204
232	63	20017675.00	18927715.00	NaN	13792086.00	9891136.00	7686401.00	572
233	74	16320537.00	15669666.00	14154937.00	12839771.00	11834676.00	10113893.00	704

234 rows × 13 columns

In [125]:
```
df.select_dtypes(include='object')
```

Out[125]:

	CCA3	Country	Capital	Continent
0	AFG	Afghanistan	Kabul	Asia
1	ALB	Albania	Tirana	Europe
2	DZA	Algeria	Algiers	Africa
3	ASM	American Samoa	Pago Pago	Oceania
4	AND	Andorra	Andorra la Vella	Europe
...	...	...	...	...
229	WLF	Wallis and Futuna	Mata-Utu	Oceania
230	ESH	Western Sahara	El Aaiún	Africa
231	YEM	Yemen	Sanaa	Asia
232	ZMB	Zambia	Lusaka	Africa
233	ZWE	Zimbabwe	Harare	Africa

234 rows × 4 columns

In [126]:
```
#to remove warnings
import warnings
warnings.filterwarnings('ignore')
```

CONCLUSION

Exploratory data analysis is a Data exploration technique to understand the sequence of data.
objectives-:
1.Identify the type of data (numeric, categorical, etc.)and recognizes the data distribution and patterns.
2.Identify any unusual patterns or extreme values that might indicate errors or anomalies in the data.
3.Calculate summary statistics (mean, median, standard deviation, etc.) to describe the central tendency and variability of the data.
4.Identify missing values and assess their impact on the analysis.

In the exploratory data analysis (EDA) report conducted on world population data using the Pandas library, a comprehensive examination of the dataset was performed to gain insights into its key characteristics. The analysis involved understanding the structure of the data, detecting anomalies or outliers, exploring relationships between variables such as population size, growth rates, and geographic regions, and summarizing essential statistics to describe the distribution of population figures worldwide. Visualizations such as histograms and scatter plots were utilized to identify patterns and trends, providing a clear overview of the dataset. The report also addressed the presence of missing data and potential implications for analysis, laying the foundation for informed decision-making and further statistical modeling.