

WORLD POPULATION RATE ANALYTICS :

```
In [1]:
import pandas as pd

In [2]:
df=pd.read_csv("C:\Users\HP\Downloads\World Population Live Dataset.csv")
df.head()
```

	CC3	Name	2022	2020	2015	2010	2000	1990	1980	1970	Area (km²)	Density (per km²)	GrowthRate	World Population Percentage	Rank
0	CN	China	1425887	1424930	1303715	1348159	1264409	1153704	982372	822534	9706961	146.8923	1.00	17.89%	1
1	IN	India	1411773	1396387	1322867	1240614	1099364	870452	696828	557001	3297590	431.0675	1.01	17.77%	2
2	US	United States	336290	336342	324608	311183	282399	248084	223140	200328	9372610	36.0935	1.00	4.24%	3
3	ID	Indonesia	275501	271858	250492	244619	214072	182160	148177	115528	1904569	144.6529	1.01	3.45%	4
4	PK	Pakistan	238825	227197	210999	194544	154370	115414	89624	56291	881912	267.4018	1.02	2.96%	5

```
In [3]:
df.tail()
```

	CC3	Name	2022	2020	2015	2010	2000	1990	1980	1970	Area (km²)	Density (per km²)	GrowthRate	World Population Percentage	Rank
229	MS	Montserrat	4	5	5	5	5	11	11	11	102	43.0282	0.99	0.00%	230
230	FK	Falkland Islands	4	4	3	3	2	2	2	2	12173	0.3105	1.00	0.00%	231
231	NJ	Niue	2	2	2	2	3	4	5	260	7.4385	1.00	0.00%	232	
232	TK	Tokelau	2	2	1	1	2	2	2	2	12	155.9167	1.01	0.00%	233
233	VA	Virgin City	1	1	1	1	1	1	1	1	510.0000	1.00	0.00%	234	

```
In [4]:
df.shape
```

```
Out[4]:
(234, 15)
```

```
In [5]:
df.describe()
```

	2022	2020	2015	2010	2000	1990	1980	1970	Area (km²)	Density (per km²)	GrowthRate	Rank
count	234.000000e+00	234.000000e+00	234.000000e+00	234.000000e+00	234.000000e+00	234.000000e+00	234.000000e+00	234.000000e+00	234.000000	234.000000	234.000000e+00	234.000000
mean	1.407441e+04	1.350206e+04	1.172956e+04	1.298452e+04	1.262987e+04	2.271024e+04	1.898464e+04	1.578887e+04	9.706961	157.8887	0.460200	452.127044
std	1.367654e+05	1.359999e+05	1.304050e+05	1.242185e+05	1.116982e+05	9.780216e+04	8.178513e+04	6.178513e+04	61785.136077	67785.064322	1.761841e+06	2066.121904
min	1.000000e+00	1.000000e+00	1.000000e+00	1.000000e+00	1.000000e+00	1.000000e+00	1.000000e+00	1.000000e+00	1.000000	1.000000	1.000000e+00	0.026100
25%	4.197500e+03	4.150000e+03	4.040000e+03	4.393000e+03	4.272500e+03	2.642500e+03	2.285000e+03	1.557500e+03	3141.000000	155.750000	2.650000e+03	38.417875
50%	5.560000e+03	5.490000e+03	5.307000e+03	4.943000e+03	4.293000e+03	3.825000e+03	3.14100000	2.60430000	8.119950e+04	85.346750	1.010000	117.500000
75%	2.447676e+04	2.144825e+04	1.973075e+04	1.915950e+04	1.576255e+04	1.138950e+04	9628.000000	8617.500000	4.304255e+05	238.923250	1.020000	176.500000
max	1.428887e+05	1.424930e+05	1.393731e+05	1.348159e+05	1.264099e+05	1.153704e+05	982372.000000	822534.000000	1.709854e+07	23172.286700	1.070000	234.000000

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 234 entries, 0 to 233
Data columns (total 15 columns):
 #   Column              Non-Null Count  Dtype
---  -
 0   CC3                  233 non-null    object
 1   Name                 234 non-null    object
 2   2022                 234 non-null    int64
 3   2020                 234 non-null    int64
 4   2015                 234 non-null    int64
 5   2010                 234 non-null    int64
 6   2000                 234 non-null    int64
 7   1990                 234 non-null    int64
 8   1980                 234 non-null    int64
 9   1970                 234 non-null    int64
10   Area (km²)           234 non-null    float64
11   Density (per km²)    234 non-null    float64
12   GrowthRate           234 non-null    float64
13   World Population Percentage  234 non-null    object
14   Rank                 234 non-null    int64
dtypes: float64(12), int64(10), object(3)
memory usage: 27.6+ KB
```

```
In [7]:
df.isnull().sum()
```

```
Out[7]:
CC3      1
Name      0
2022      0
2020      0
2015      0
2010      0
2000      0
1990      0
1980      0
1970      0
Area (km²)  0
Density (per km²)  0
GrowthRate  0
World Population Percentage  0
Rank      0
dtype: object
```

```
In [8]:
df.dropna(inplace = True)
```

```
Out[8]:
df.isnull().sum().sum()
0
```

```
Out[9]:
df.duplicated().sum()
0
```

```
Out[10]:
df.isnull().sum()
0
```

```
In [11]:
df.columns
```

```
Out[11]:
Index(['CC3', 'Name', '2022', '2020', '2015', '2010', '2000', '1990', '1980', '1970', 'Area (km²)', 'Density (per km²)', 'GrowthRate', 'World Population Percentage', 'Rank'],
      dtype='object')
```

```
In [12]:
df.corr()
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_15212\1134722465.py:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to silence this warning.
df.corr()
```

	2022	2020	2015	2010	2000	1990	1980	1970	Area (km²)	Density (per km²)	GrowthRate	Rank
2022	1.000000	0.999946	0.999890	0.999829	0.999804	0.987226	0.980281	0.973156	0.453619	-0.027840	-0.014641	-0.358126
2020	0.999946	1.000000	0.999950	0.999895	0.999860	0.987222	0.980278	0.973156	0.453619	-0.027840	-0.014641	-0.358126
2015	0.999890	0.999950	1.000000	0.999763	0.997339	0.991582	0.985721	0.979409	0.458444	-0.027074	-0.025938	-0.356900
2010	0.999829	0.999860	0.999763	1.000000	0.998983	0.993028	0.988784	0.983038	0.462138	-0.026720	-0.021799	-0.347231
2000	0.999804	0.999860	0.997339	0.998983	1.000000	0.998336	0.995159	0.990954	0.474133	-0.026349	-0.044500	-0.340830
1990	0.987226	0.987222	0.991582	0.993028	0.998336	1.000000	0.999041	0.996601	0.486965	-0.026434	-0.056737	-0.335924
1980	0.980281	0.980278	0.985721	0.988784	0.991582	0.995159	1.000000	0.999194	0.488370	-0.026799	-0.067230	-0.335015
1970	0.973156	0.973156	0.979409	0.983038	0.990954	0.996601	0.999194	1.000000	0.510147	-0.027095	-0.075685	-0.335146
Area (km²)	0.453619	0.453619	0.458444	0.462138	0.474133	0.486965	0.488370	0.510147	1.000000	-0.063007	-0.010959	-0.384169
Density (per km²)	-0.027840	-0.027840	-0.027074	-0.026720	-0.026349	-0.026434	-0.026799	-0.027095	-0.063007	1.000000	-0.088713	0.128876
GrowthRate	-0.014641	-0.014641	-0.025938	-0.021799	-0.044500	-0.056737	-0.067230	-0.075685	-0.010959	-0.088713	1.000000	-0.221209
Rank	-0.358126	-0.358126	-0.356900	-0.347231	-0.340830	-0.335924	-0.335015	-0.335146	-0.384169	0.128876	-0.221209	1.000000

```
In [13]:
df.dtypes
```

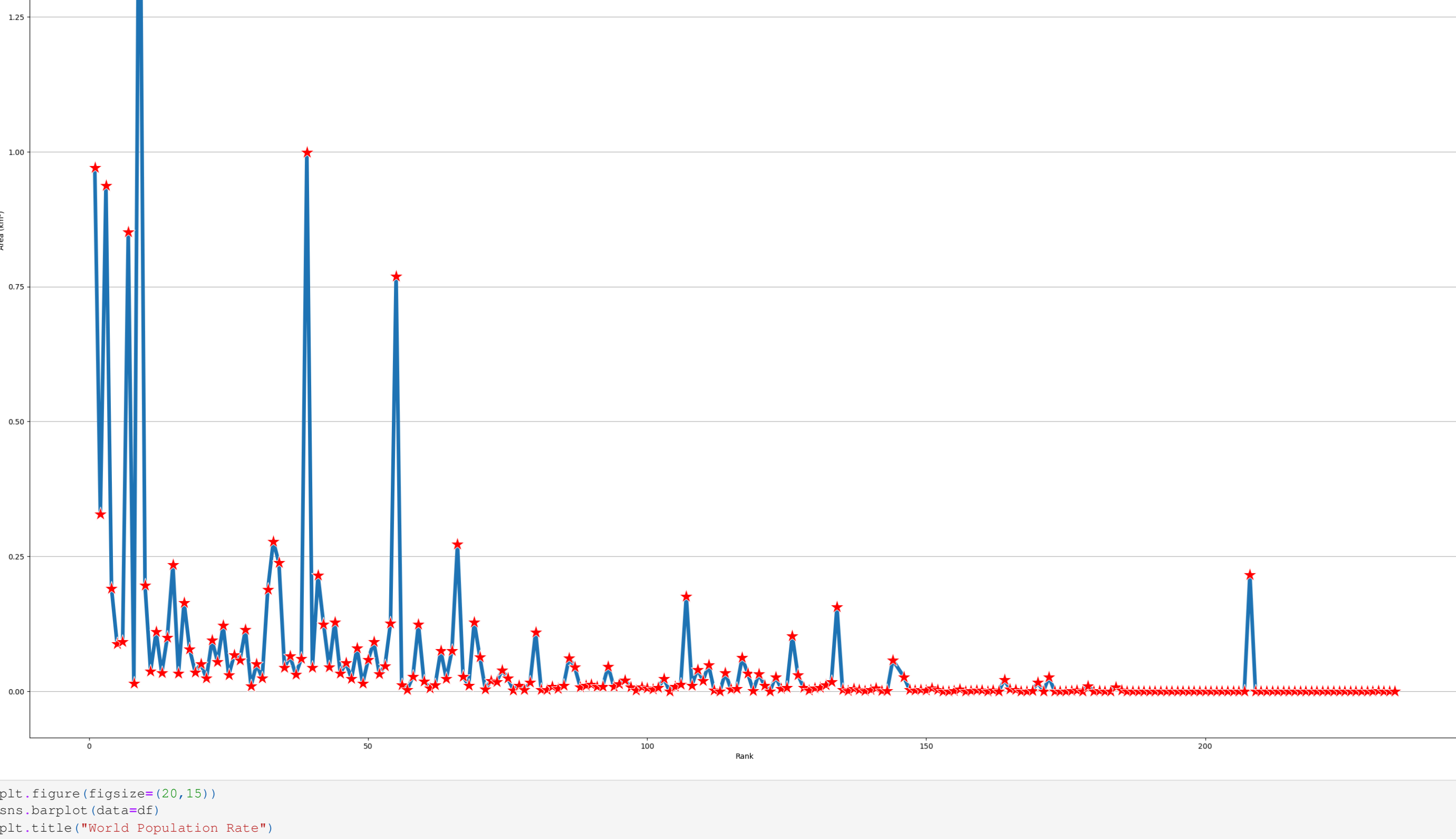
```
Out[13]:
CC3      object
Name      object
2022      int64
2020      int64
2015      int64
2010      int64
2000      int64
1990      int64
1980      int64
1970      int64
Area (km²)  float64
Density (per km²)  float64
GrowthRate  float64
World Population Percentage  object
Rank      int64
dtype: object
```

```
In [14]:
df.dtypes
```

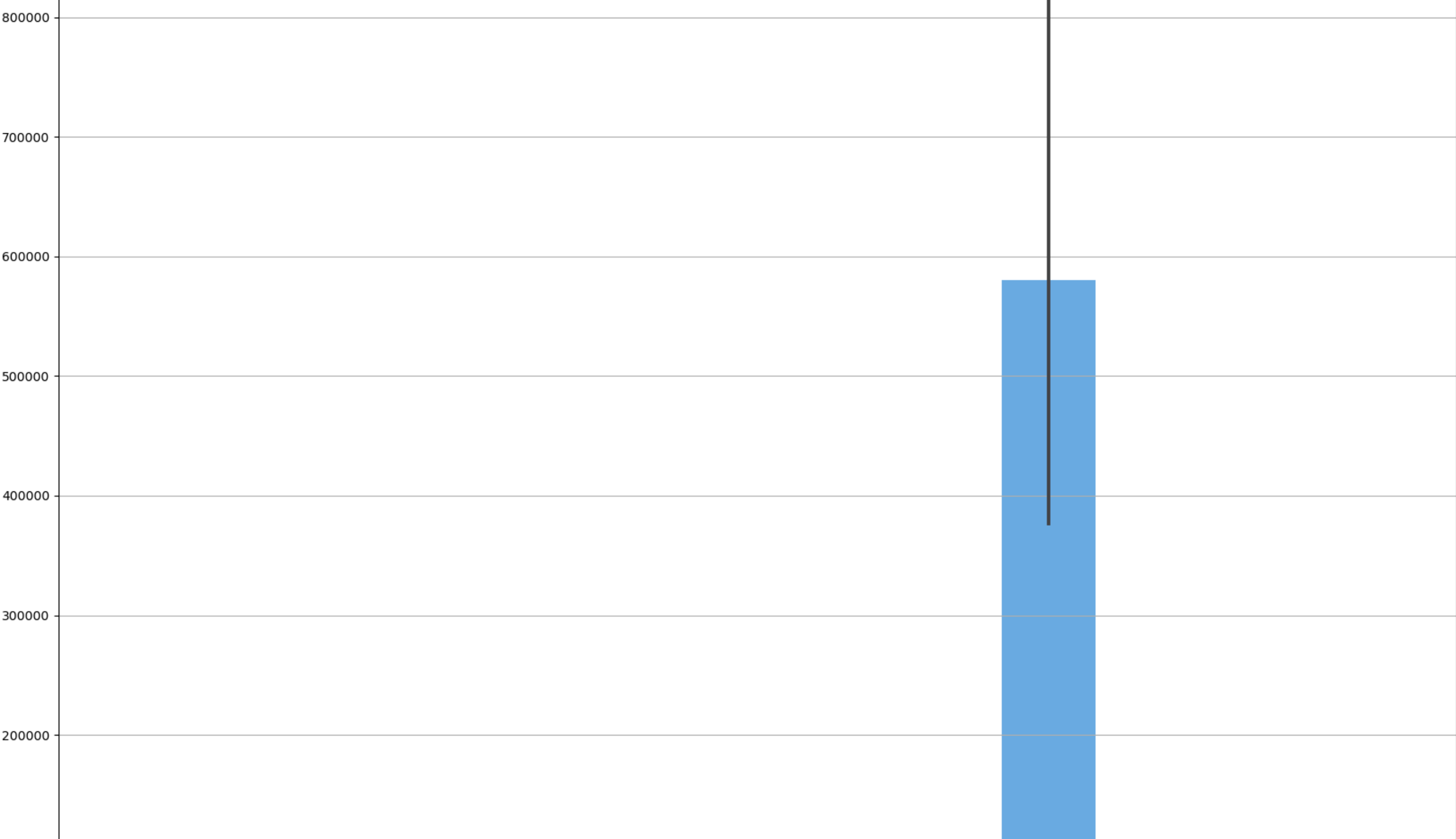
Data Analytics:

```
In [14]:
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

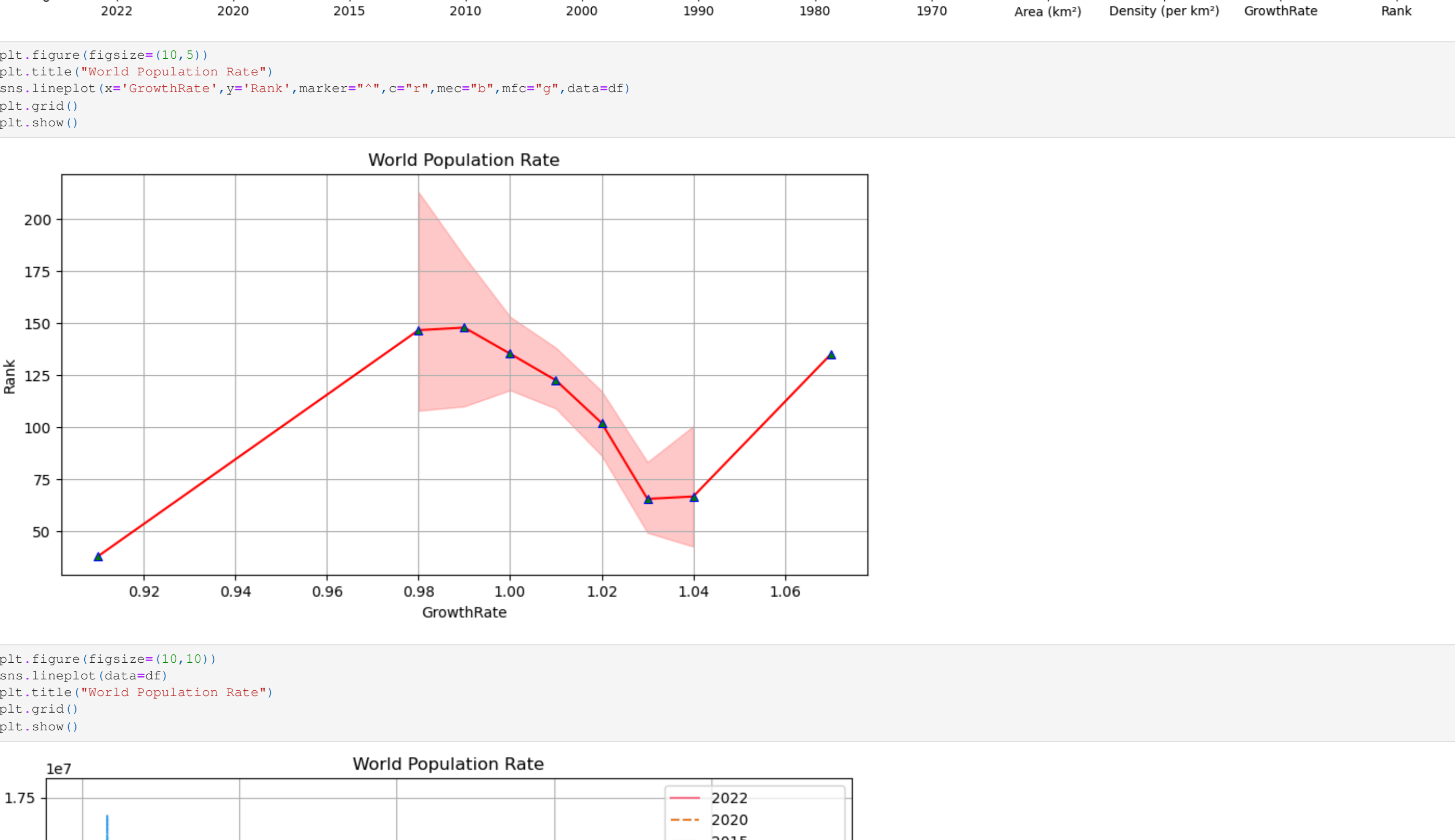
In [15]:
plt.figure(figsize=(15,25))
sns.lineplot(x='Rank',y='Area (km²)',marker='*',ms=20,mfc='r',lw=5,data=df)
plt.title('World Population Rate')
plt.show()
```



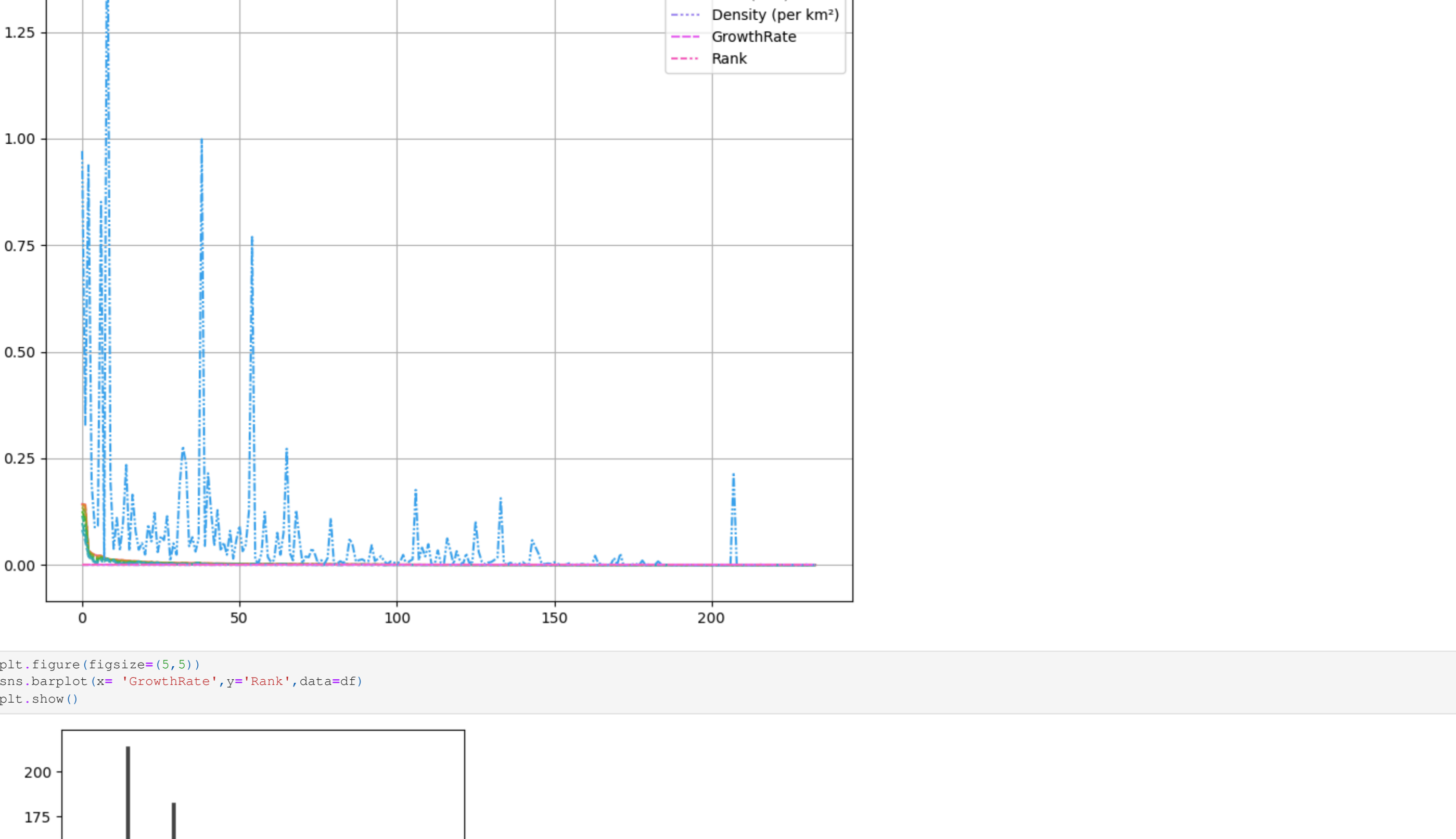
```
In [16]:
plt.figure(figsize=(20,15))
sns.barplot(x='GrowthRate',y='Rank',data=df)
plt.title('World Population Rate')
plt.show()
```



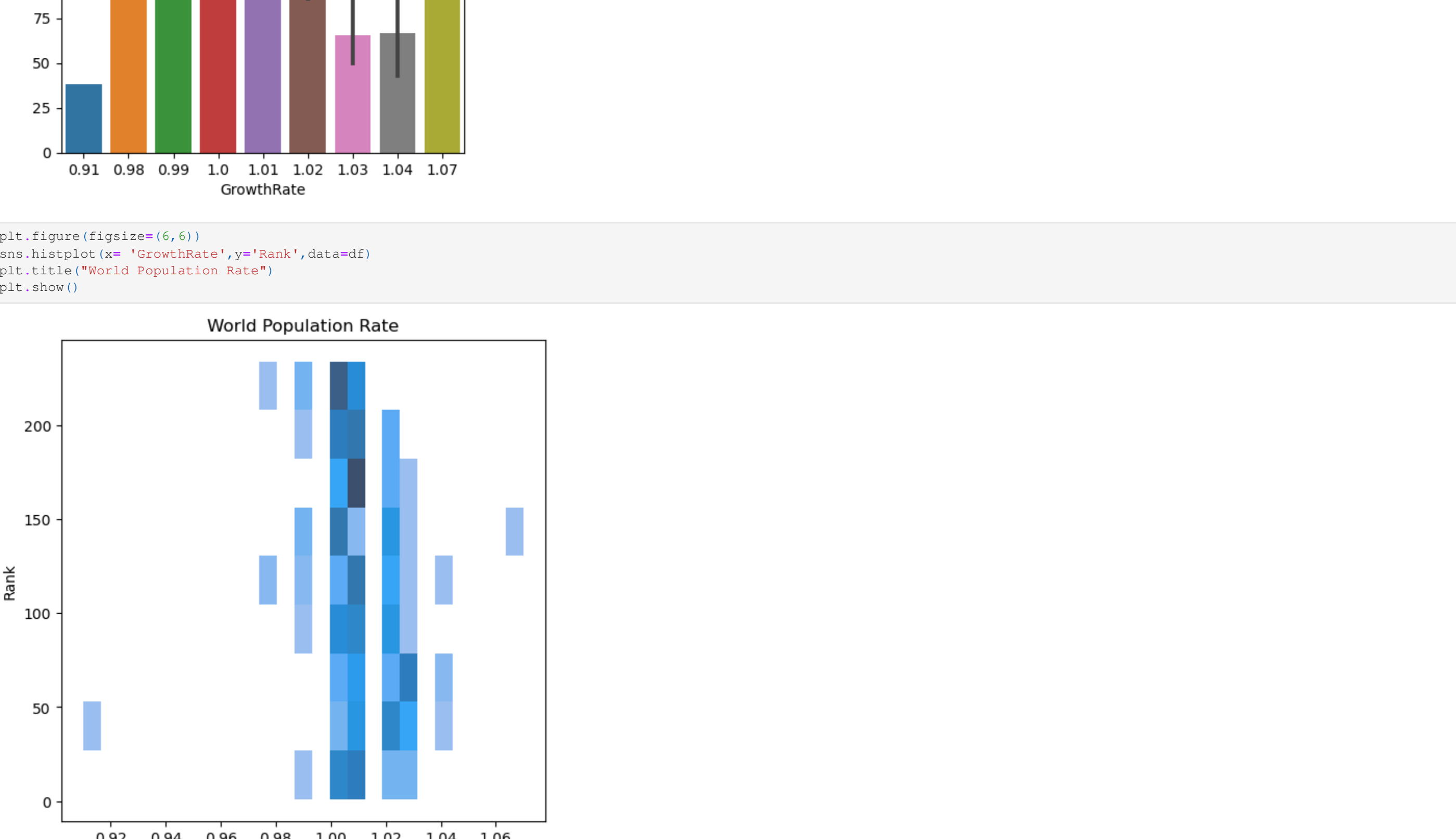
```
In [17]:
plt.figure(figsize=(15,5))
sns.histplot(x='GrowthRate',y='Rank',data=df)
plt.title('World Population Rate')
plt.show()
```



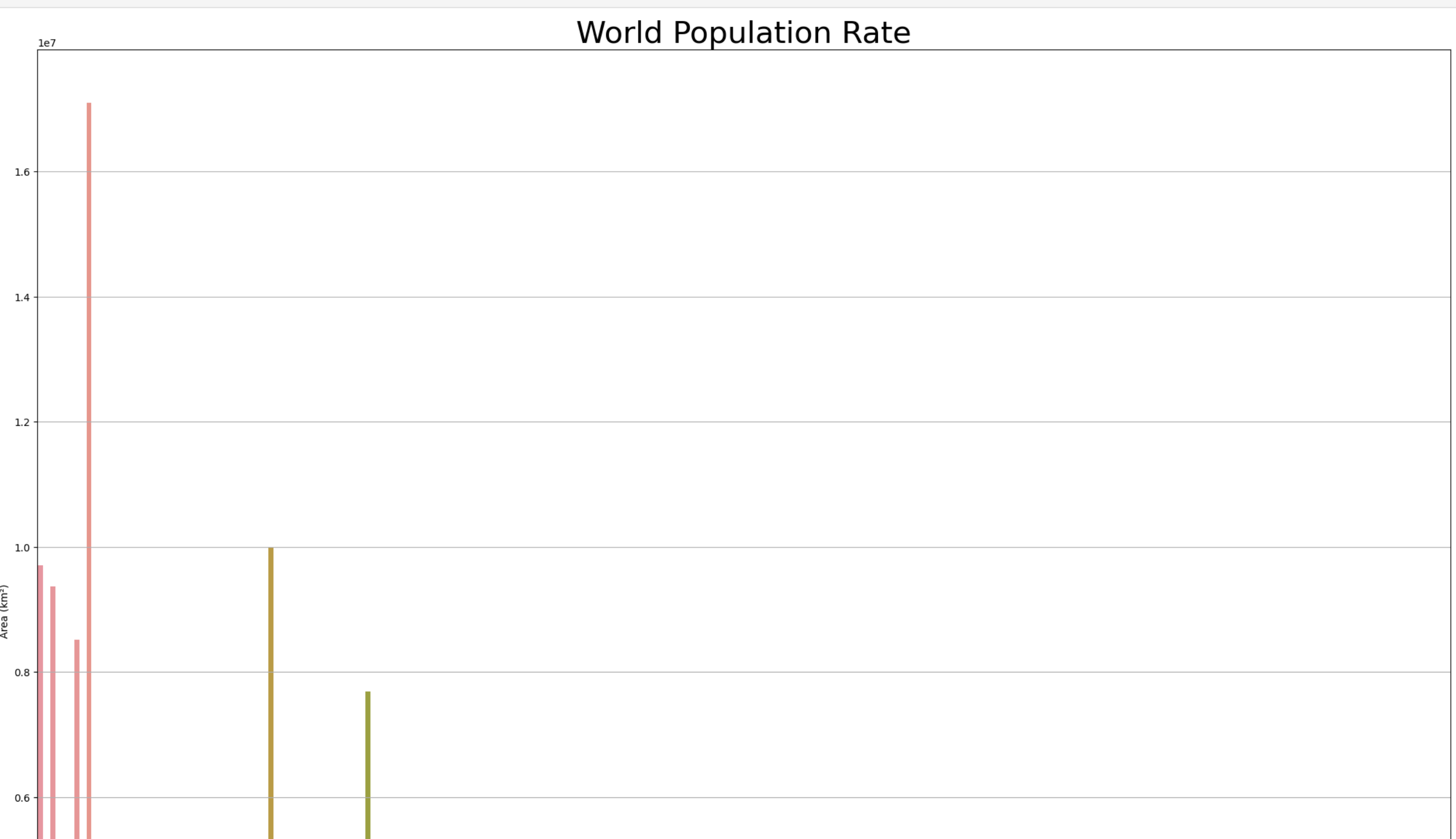
```
In [18]:
plt.figure(figsize=(15,10))
sns.lineplot(x='GrowthRate',y='Rank',data=df)
plt.title('World Population Rate')
plt.grid()
plt.show()
```



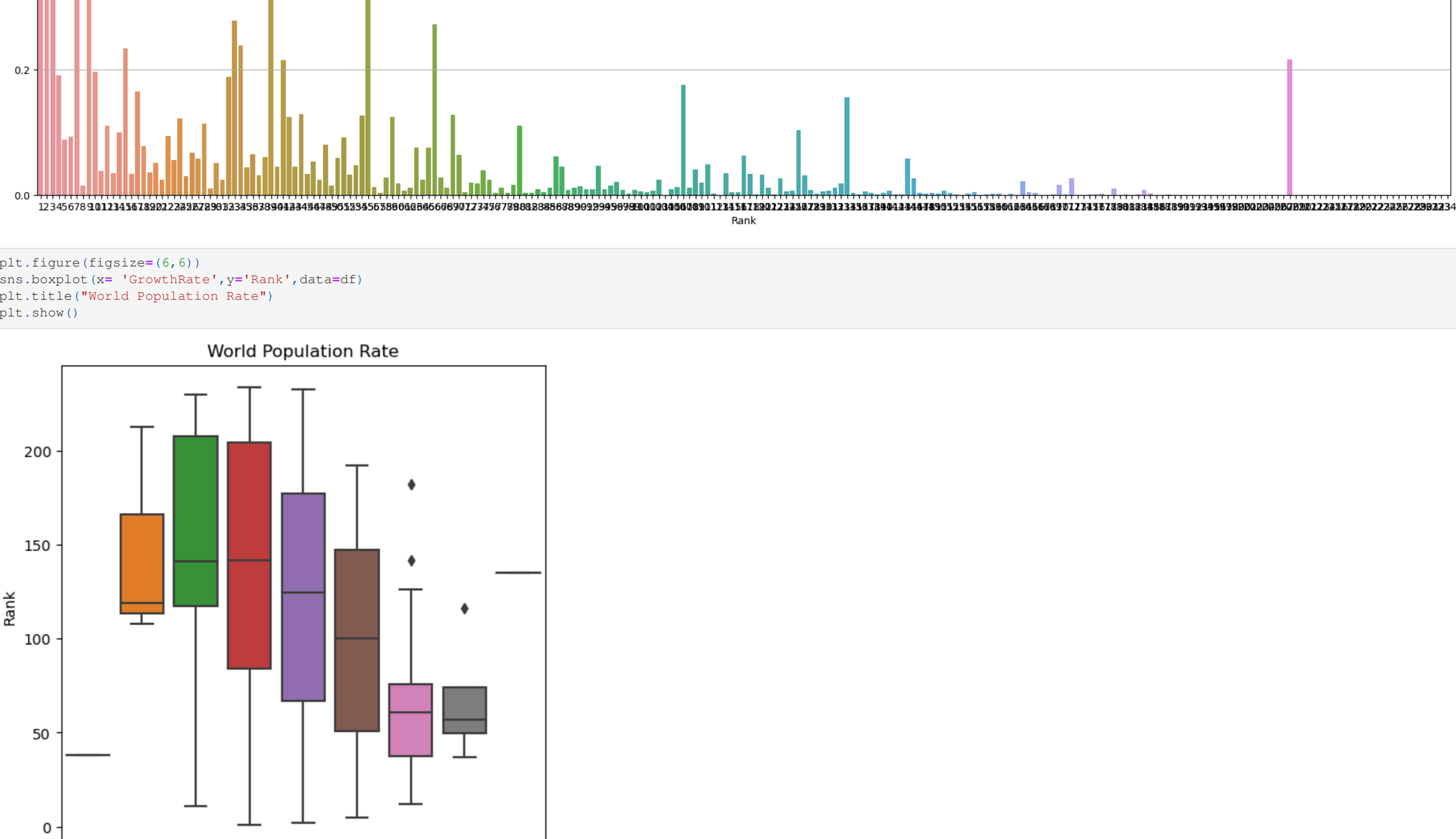
```
In [19]:
plt.figure(figsize=(5,5))
sns.histplot(x='GrowthRate',y='Rank',data=df)
plt.title('World Population Rate')
plt.show()
```



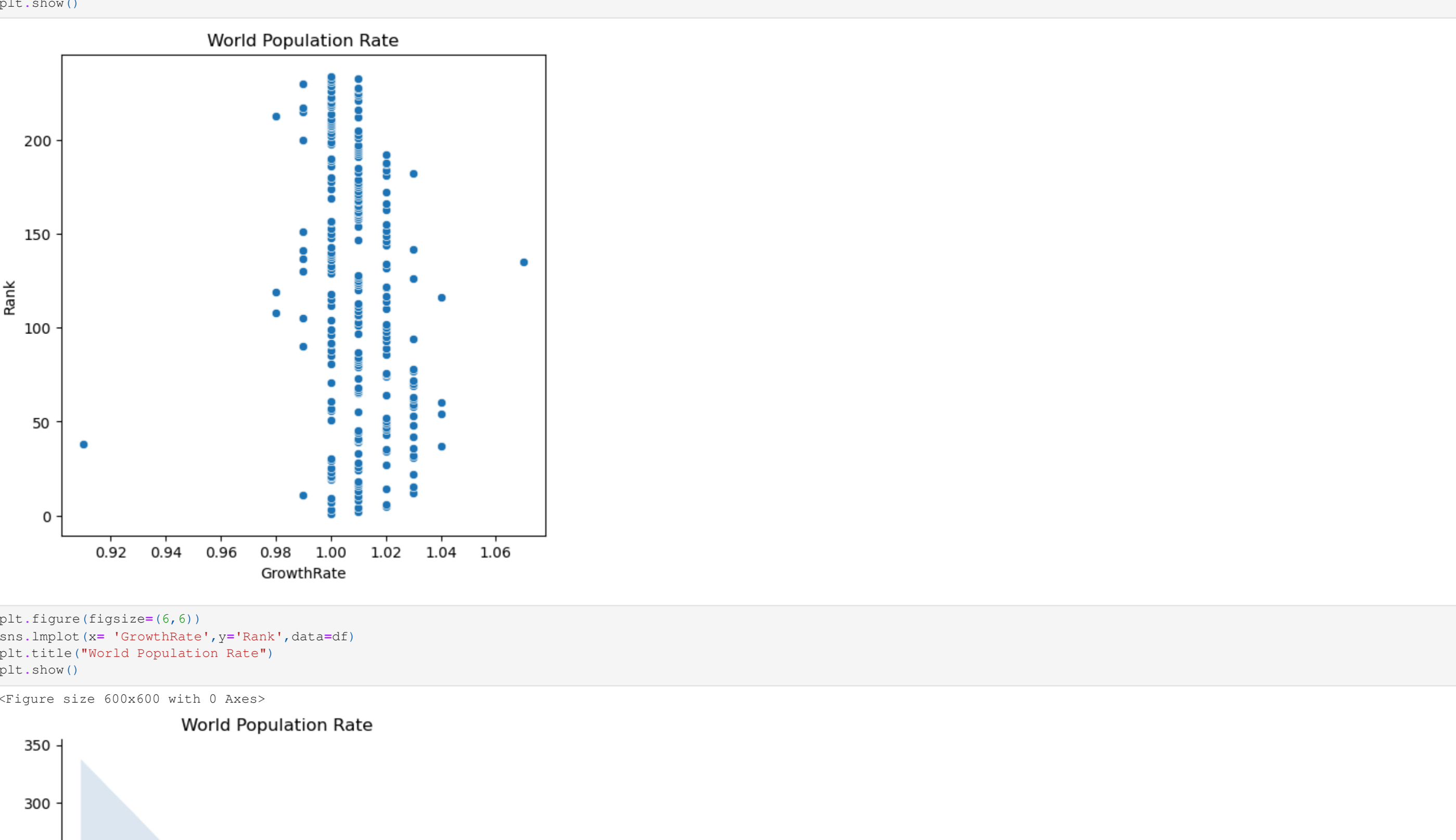
```
In [20]:
plt.figure(figsize=(6,6))
sns.histplot(x='GrowthRate',y='Rank',data=df)
plt.title('World Population Rate')
plt.show()
```




```
In [21]:
plt.figure(figsize=(25,20))
sns.barplot(x='Rank',y='Area (km²)',width=0.8,data=df)
plt.title('World Population Rate')
plt.grid(axis='y')
plt.show()
```



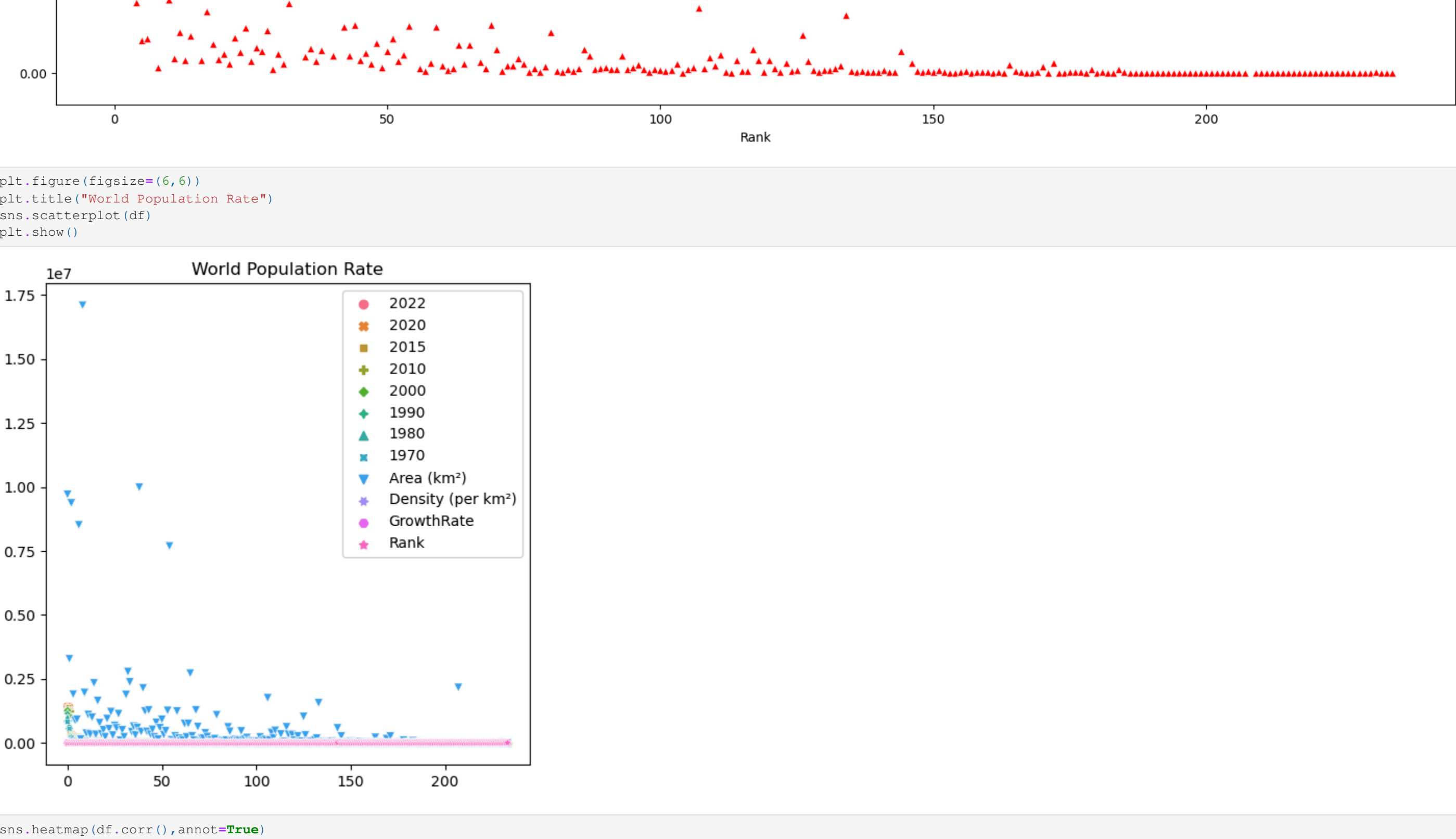
```
In [22]:
plt.figure(figsize=(6,6))
sns.histplot(x='GrowthRate',y='Rank',data=df)
plt.title('World Population Rate')
plt.show()
```




```
In [23]:
plt.figure(figsize=(6,6))
sns.scatterplot(x='GrowthRate',y='Rank',data=df)
plt.title('World Population Rate')
plt.show()
```



```
In [24]:
plt.figure(figsize=(6,6))
sns.scatterplot(x='GrowthRate',y='Rank',data=df)
plt.title('World Population Rate')
plt.show()
```



```
In [25]:
plt.figure(figsize=(20,10))
sns.scatterplot(x='Rank',y='Area (km²)',c='r',marker='*',data=df)
plt.title('World Population Rate')
plt.show()
```



```
In [26]:
sns.heatmap(df.corr(),annot=True)
```



```
In [27]:
sns.heatmap(df.corr(),annot=False)
```



```
In [28]:
# Thanking You!
```

```
In [29]:
# Thiswillalways 0
```