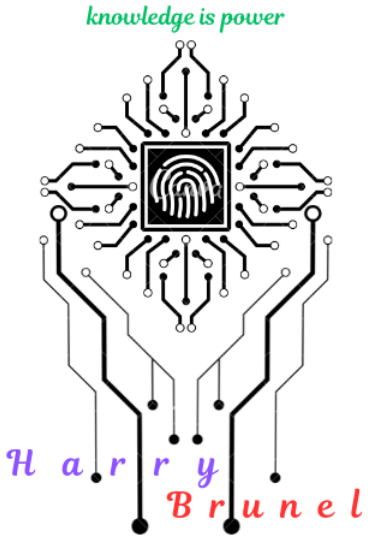


La donnée est l'actif stratégique de la révolution numérique

World Population Analysis

- Source du jeux de données : https://www.kaggle.com/datasets/rajkumarpandey02/2023-world-population-by-country

> A propos de l'ensemble de données
>
> Selon l'horloge démographique mondiale du Bureau du recensement des États-Unis, la population mondiale s'élevait à 7 922 312 800 personnes en septembre 2022 et devrait atteindre 8 milliards à la mi-novembre 2022. Ce total dépasse de loin la population mondiale de 2015, qui était de 7,2 milliards. La population mondiale continue d'augmenter d'environ 140 personnes par minute, les naissances l'emportant sur les décès dans la plupart des pays.
>
> Dans l'ensemble, cependant, le taux de croissance de la population ralentit depuis plusieurs décennies. Ce ralentissement devrait se poursuivre jusqu'à ce que le taux de croissance démographique atteigne zéro (nombre égal de naissances et de décès) vers 2080-2100, pour une population d'environ 10,4 milliards d'habitants. Après cette période, le taux de croissance démographique devrait devenir négatif, ce qui entraînera un déclin de la population mondiale.
>
> Pays comptant plus d'un milliard d'habitants La Chine est actuellement le pays le plus peuplé du monde, avec une population estimée à plus de 1,42 milliard d'habitants en septembre 2022. Un seul autre pays au monde peut se vanter d'avoir une population de plus d'un milliard d'habitants : L'Inde, dont la population est estimée à 1,41 milliard d'habitants et ne cesse d'augmenter.
>
> Traduit à partir du jeu de données de rajkumarpandey02 (Kaggle) </blocquote>

> Le jeu de données "world_population_by_country.csv"contient des informations détaillées sur la population mondiale par pays. Voici une brève description des colonnes :
>
> country: ==> le nom du pays.
> rank: ==> le rang du pays en fonction de la taille de sa population.
> area: ==> la superficie du pays en kilomètres carrés.
> landAreaKm: ==> la superficie terrestre du pays en kilomètres carrés.
> cca2 et cca3: ==> codes de pays en deux et trois lettres respectivement.
> netChange: ==> le changement net dans la population.
> growthRate: ==> le taux de croissance de la population.
> worldPercentage: ==> le pourcentage de la population mondiale que représente la population de ce pays.
> density et densityMi: ==> la densité de population par kilomètre carré et par mille respectivement.
> place: ==> rang du pays selon la densité de population.
> pop1980, pop2000, pop2010, pop2022, pop2023, pop2030, pop2050: ==> la population du pays pour chaque année respective.
> </blocquote>

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")

import plotly.express as px
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
from plotly import graph_objs as go
```

In [2]:
```
df = pd.read_csv('./data/world_population_by_country.csv')
df
```

Out[2]:

	country	rank	area	landAreaKm	cca2	cca3	netChange	growthRate	worldPercentage	dens
0	India	1	3287590.00	2973190.00	IN	IND	0.4184	0.0081	0.1785	480.50
1	China	2	9706961.00	9424702.90	CN	CHN	-0.0113	-0.0002	0.1781	151.26
2	United States	3	9372610.00	9147420.00	US	USA	0.0581	0.0050	0.0425	37.16
3	Indonesia	4	1904569.00	1877519.00	ID	IDN	0.0727	0.0074	0.0347	147.81
4	Pakistan	5	881912.00	770880.00	PK	PAK	0.1495	0.0198	0.0300	311.96
...	...	...	...	...	...	...	...	...	...	
229	Montserrat	230	102.00	102.00	MS	MSR	NaN	-0.0009	NaN	43.00
230	Falkland Islands	231	12173.00	12173.00	FK	FLK	NaN	0.0029	NaN	0.31
231	Niue	232	261.00	261.00	NU	NIU	0.0000	0.0005	NaN	7.41
232	Tokelau	233	12.00	10.00	TK	TKL	NaN	0.0118	NaN	189.30
233	Vatican City	234	0.44	0.44	VA	VAT	NaN	0.0157	NaN	1177.27

234 rows × 19 columns

In [3]:
```
df.shape
```

Out[3]: (234, 19)

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 234 entries, 0 to 233
Data columns (total 19 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   country          234 non-null    object 
 1   rank             234 non-null    int64  
 2   area             234 non-null    float64
 3   landAreaKm       234 non-null    float64
 4   cca2             233 non-null    object 
 5   cca3             234 non-null    object 
 6   netChange        226 non-null    float64
 7   growthRate       234 non-null    float64
 8   worldPercentage  228 non-null    float64
 9   density          234 non-null    float64
 10  densityMi        234 non-null    float64
 11  place            234 non-null    int64  
 12  pop1980          234 non-null    int64  
 13  pop2000          234 non-null    int64  
 14  pop2010          234 non-null    int64  
 15  pop2022          234 non-null    int64  
 16  pop2023          234 non-null    int64  
 17  pop2030          234 non-null    int64  
 18  pop2050          234 non-null    int64  
dtypes: float64(7), int64(9), object(3)
memory usage: 34.9+ KB
```

In [5]:
```
df.describe().T.sort_values("50%", ascending = False).style.background_gradient(cmap = "RdPu")
    .bar(subset = ["mean"], color = "red").bar(subset = ["max"], color = "green")
```

Out[5]:

	count	mean	std	min	25%	50%
pop2050	234.000000	41486278.790598	148167567.122402	731.000000	546605.750000	6352397.000000
pop2030	234.000000	36514605.33333	141782710.184894	561.000000	456149.000000	6178231.000000
pop2023	234.000000	34374424.74359	137386405.597263	518.000000	422598.250000	5643895.000000
pop2022	234.000000	34074414.71367	136766424.804728	510.000000	419738.500000	5559944.500000
pop2010	234.000000	29845235.04273	124218487.631581	596.000000	393149.000000	4942770.500000
pop2000	234.000000	26269468.8	111698206.719070	651.000000	327242.000000	4292907.000000
pop1980	234.000000	18984616.982906	81785186.081872	733.000000	229614.250000	3141145.500000
area	234.000000	581449.983590	1761840.665609	0.440000	2650.000000	81199.500000
landAreaKm	234.000000	557112.276239	1689971.526445	0.440000	2625.875000	75689.250000
place	234.000000	439.085470	253.295484	4.000000	223.000000	439.000000
densityMi	234.000000	1168.836388	5126.548664	0.357400	102.946450	252.475800
rank	234.000000	117.500000	67.694165	1.000000	59.250000	117.500000
density	234.000000	451.288182	1979.362419	0.138000	39.747650	97.481000
growthRate	234.000000	0.009737	0.012350	-0.074500	0.002325	0.008200
netChange	226.000000	0.010306	0.034774	-0.028600	0.000000	0.000900
worldPercentage	228.000000	0.004407	0.017375	0.000000	0.000100	0.000750

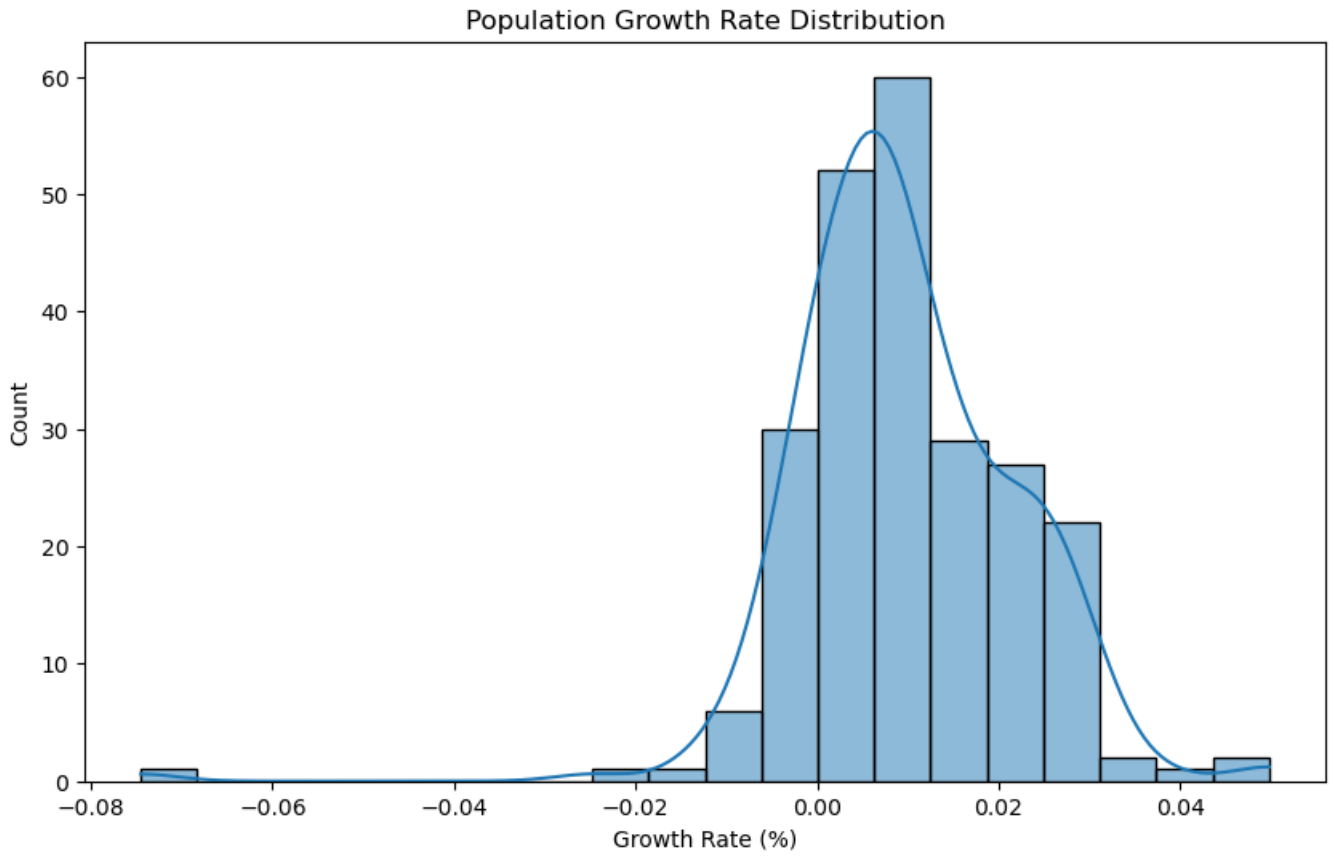
> Le tableau ci-dessus présente des statistiques descriptives pour chaque colonne numérique de notre ensemble de données, triées en fonction de la médiane ("50%").
> </blocquote>

In [6]:
```
df.isna().sum()
```

Out[6]:
```
country            0
rank               0
area               0
landAreaKm         0
cca2               1
cca3               0
netChange          8
growthRate         0
worldPercentage    6
density            0
densityMi          0
place              0
pop1980            0
pop2000            0
pop2010            0
pop2022            0
pop2023            0
pop2030            0
pop2050            0
dtype: int64
```

> Il semble que nous ayons quelques valeurs manquantes dans les colonnes netChange et worldPercentage. Comme le nombre de valeurs manquantes est relativement faible, nous pouvons choisir de supprimer ces lignes de notre ensemble de données. Cela devrait avoir un impact minimal sur notre analyse, car nous avons encore beaucoup de données disponibles.
> Nous y reviendrons plus tard
>
> </blocquote>

In [7]:
```
# Distribution du taux de croissance de la population
plt.figure(figsize=(10, 6))
sns.histplot(data=df, x='growthRate', bins=20, kde=True)
plt.title('Population Growth Rate Distribution')
plt.xlabel('Growth Rate (%)')
plt.ylabel('Count')
plt.show()
```



Commençons par examiner les pays ayant la plus grande population, la plus grande superficie, et la densité de population la plus élevée.

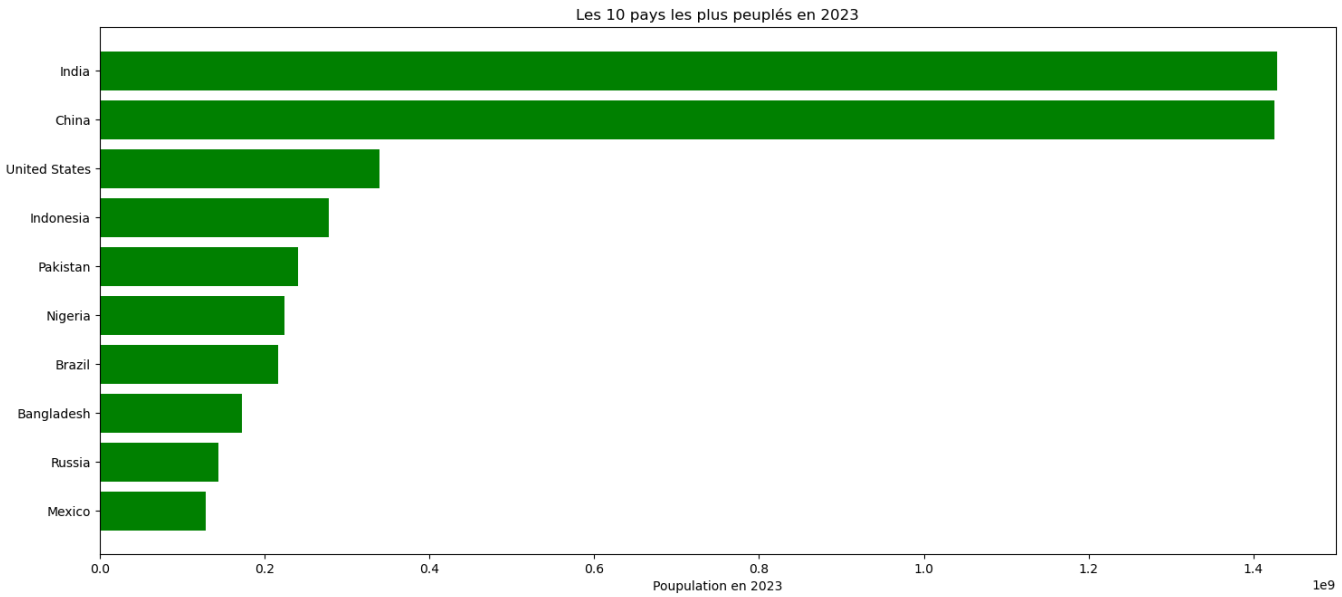
In [26]:
```
annee_23 = df.sort_values(by='pop2023', ascending=False)

fig1 = px.treemap(annee_23, path=[annee_23.country[:10]], values = annee_23['pop2023'][:10],
fig1.show()
```

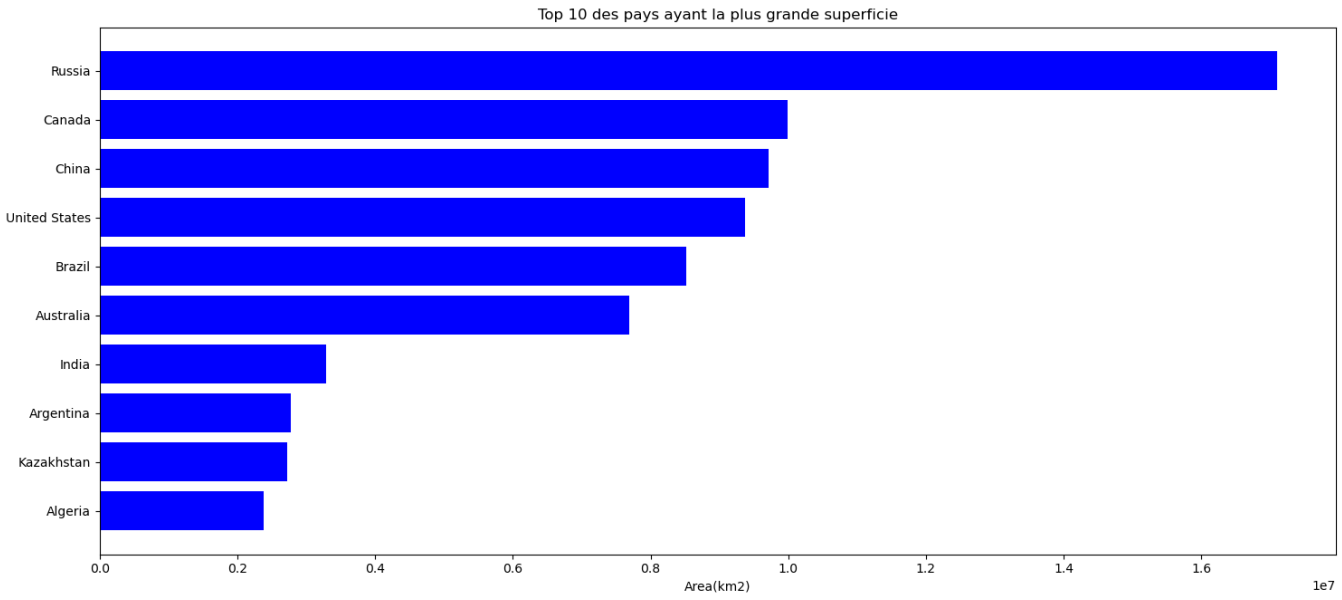
In [27]:

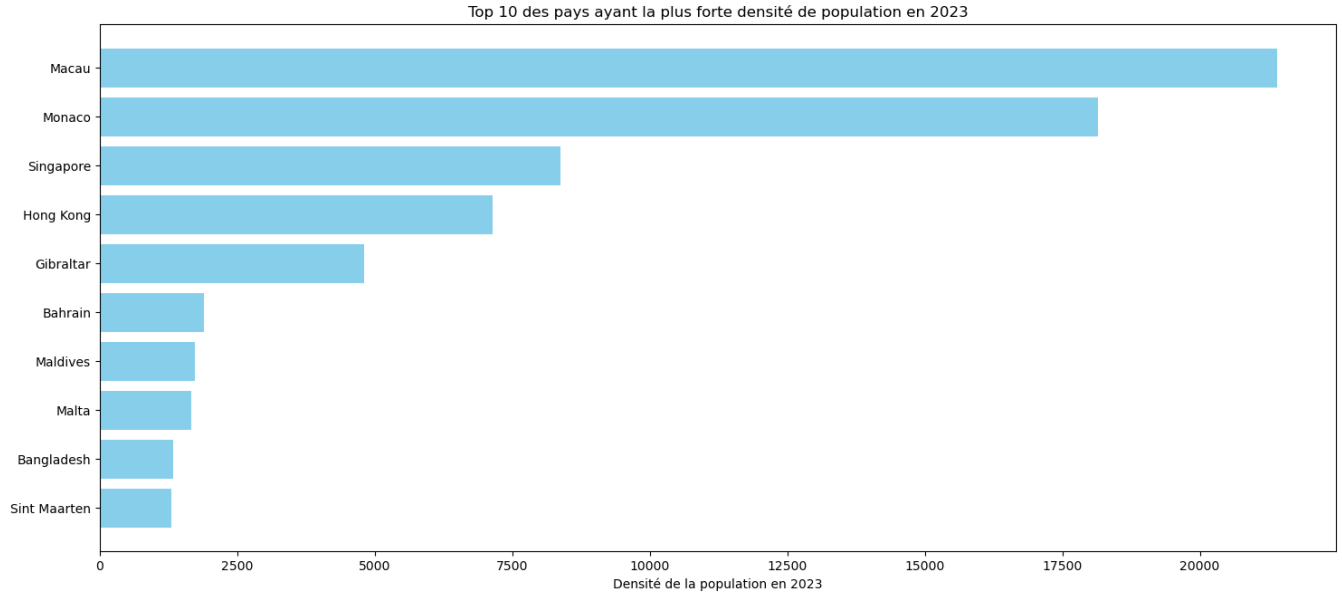
```
fig2 = px.bar(annee_23,x=annee_23.country[:10], y= annee_23['pop2023'][:10], color=annee_23.c
fig2.update_layout(xaxis_title = "Countries", yaxis_title="Population of countries")
fig2.show()

fig = px.choropleth(locations = annee_23.cca3, color=annee_23['pop2023'],
                    color_continuous_scale=px.colors.sequential.Rainbow,
                    title='population mondiale en 2023')
fig.show()
```

In [9]:

```
#  Les 10 pays les plus peuplés en 2023
top_pop_2023 = df.nlargest(10, "pop2023")

# Top 10 des pays ayant la plus grande superficie
top_area = df.nlargest(10, "area")

# Top 10 des pays ayant la plus forte densité de population en 2023
top_density_2023 = df.nlargest(10, "density")

fig, ax = plt.subplots(3, 1, figsize=(15, 20))

ax[0].barh(top_pop_2023["country"], top_pop_2023["pop2023"], color="green")
ax[0].invert_yaxis()
ax[0].set_xlabel("Poupulation en 2023")
ax[0].set_title("Les 10 pays les plus peuplés en 2023")

ax[1].barh(top_area["country"], top_area["area"], color='blue')
ax[1].invert_yaxis()
ax[1].set_xlabel("Area(km2)")
ax[1].set_title("Top 10 des pays ayant la plus grande superficie")

ax[2].barh(top_density_2023["country"], top_density_2023["density"], color='skyblue')
ax[2].invert_yaxis()
ax[2].set_xlabel("Densité de la population en 2023")
ax[2].set_title("Top 10 des pays ayant la plus forte densité de population en 2023")

plt.tight_layout()
plt.show()
```

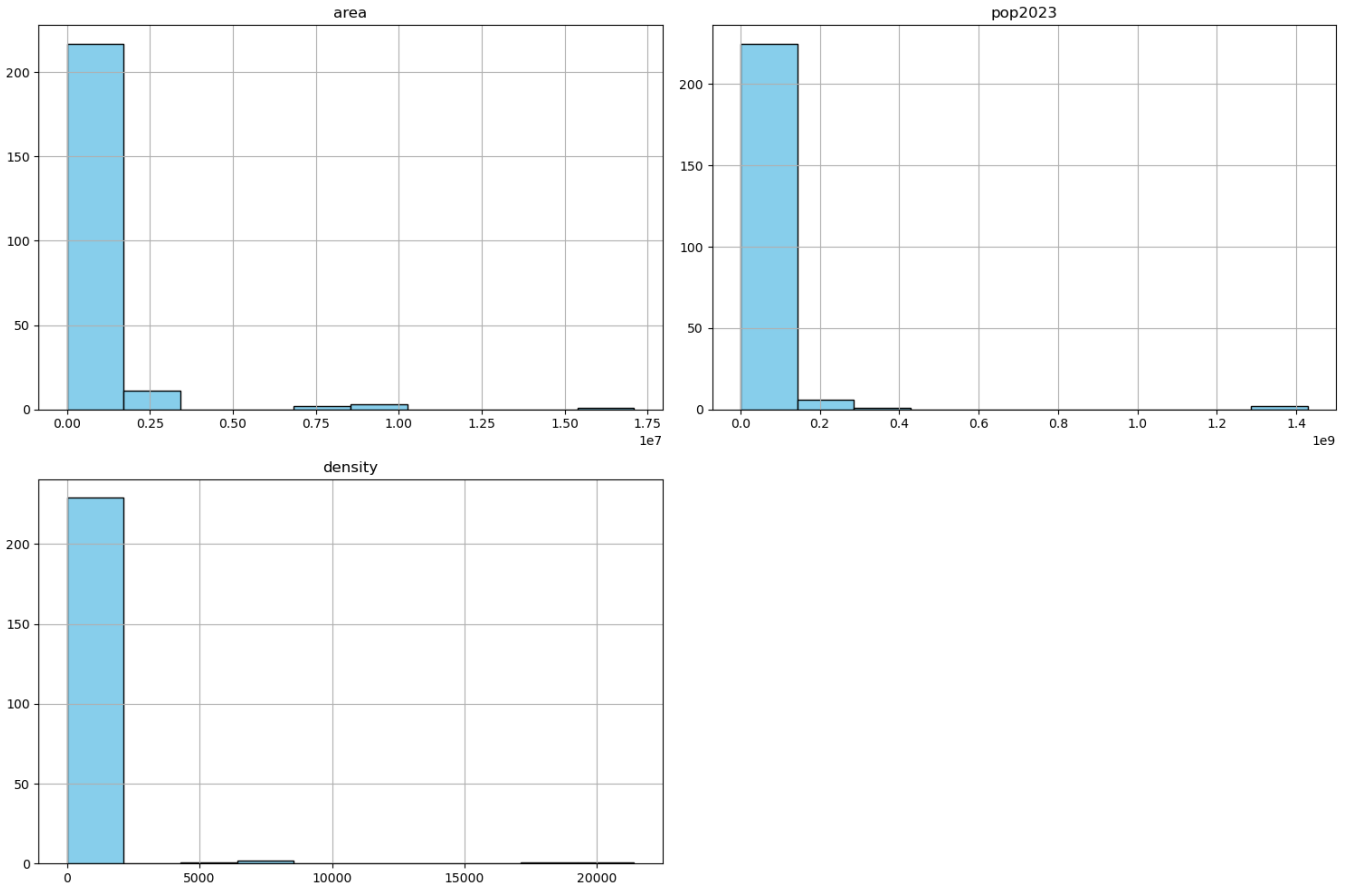

Les 10 pays les plus peuplés en 2023
India
China
United States
Indonesia
Pakistan
Nigeria
Brazil
Bangladesh
Russia
Mexico
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
Poupulation en 2023
1e9


Top 10 des pays ayant la plus grande superficie
Russia
Canada
China
United States
Brazil
Australia
India
Argentina
Kazakhstan
Algeria
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
Area(km2)
1e7


Top 10 des pays ayant la plus forte densité de population en 2023
Macau
Monaco
Singapore
Hong Kong
Gibraltar
Bahrain
Maldives
Malta
Bangladesh
Sint Maarten
0
2500
5000
7500
10000
12500
15000
17500
20000
Densité de la population en 2023

> D'après les graphiques ci-dessus, nous pouvons observer les points suivants :
>
> 1. Top 10 des pays par population en 2023 : L'Inde et la Chine ont la plus grande population en 2023, suivies des États-Unis et de l'Indonésie. Ces pays ont une population bien supérieure à celle des autres pays.
>
> 2. Top 10 des pays par superficie : La Russie a la plus grande superficie, suivie de près par l'Antarctique. Il est important de noter que l'Antarctique n'est pas un pays, mais il est inclus dans cet ensemble de données. Les États-Unis, la Chine et le Canada sont également parmi les pays les plus grands en termes de superficie.
>
>   </p>
>
> 3. Top 10 des pays par densité de population en 2023 : Macao (Chine), Monaco et Singapour ont la plus haute densité de population en 2023. Ces régions sont toutes assez petites en termes de superficie, ce qui peut expliquer leur densité de population élevée.
>
>   </p> </blocquote>
>
>   Ces informations pourraient être utiles pour comprendre les tendances démographiques mondiales et la distribution de la population à travers le monde. Elles pourraient également être utiles pour l'analyse des marchés potentiels, l'analyse de la politique d'immigration, l'urbanisme, entre autres.

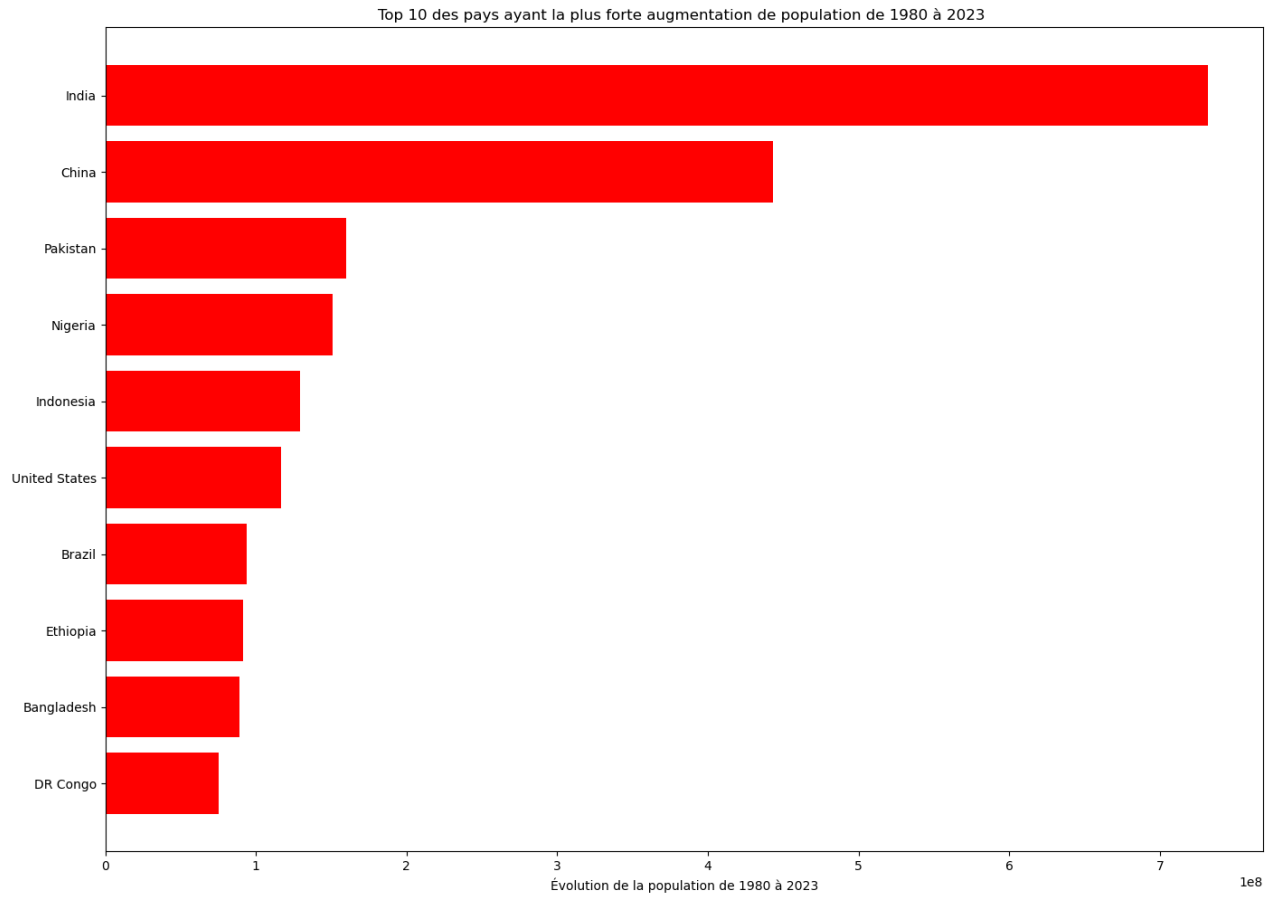
In [10]:
```
# Histogrammes de certaines colonnes numériques
df[['area', 'pop2023', 'density']].hist(bins=10, figsize=(15, 10), color='skyblue', edgecolor=
plt.tight_layout()
plt.show()
```

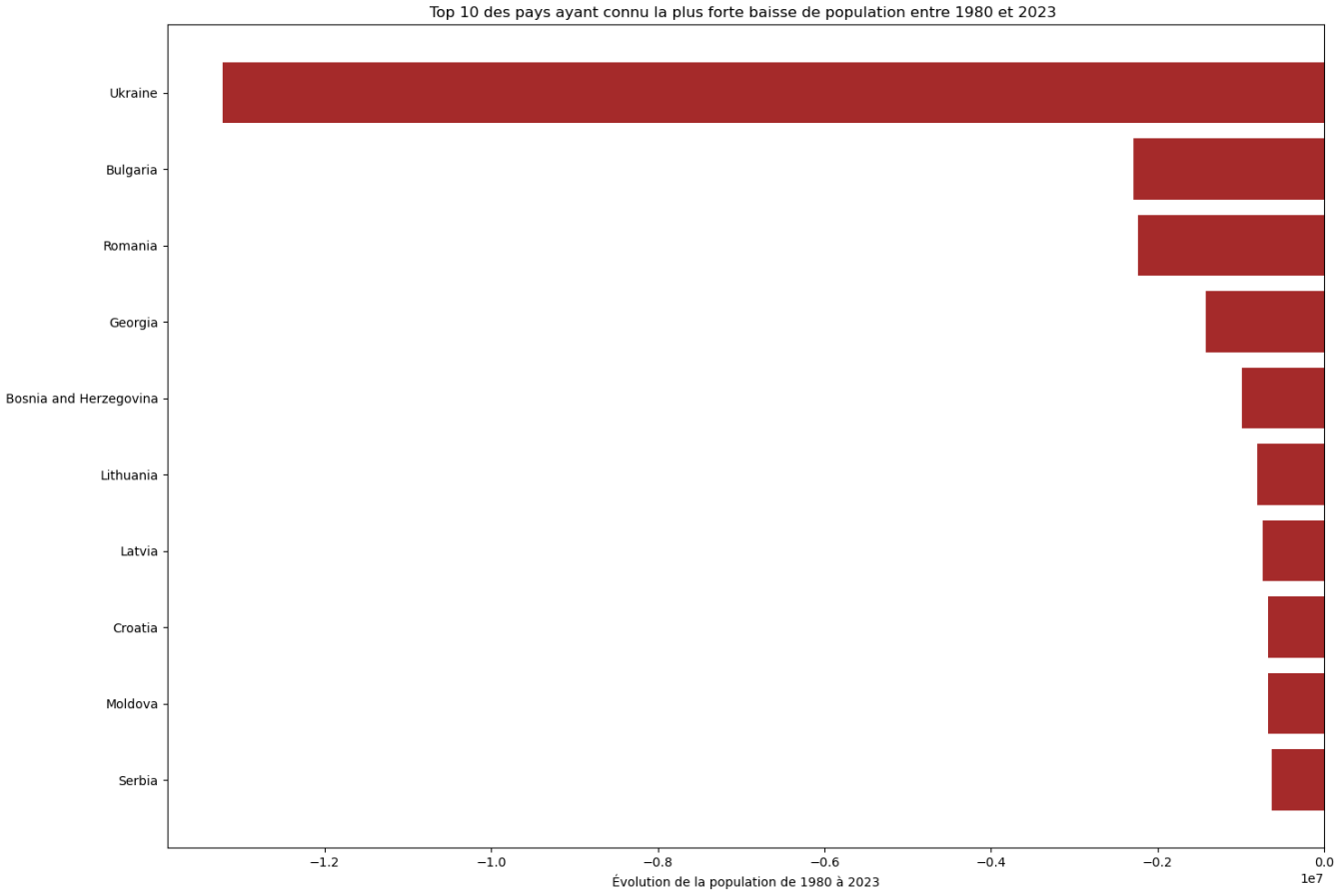


Les histogrammes ci-dessus montrent les distributions de la superficie (area), de la population en 2023 (pop2023), et de la densité de population (density). Voici quelques observations :

> - Superficie (area) : La plupart des pays ont une superficie inférieure à 2 millions de km². Il y a quelques pays avec une superficie beaucoup plus grande, ce qui crée une distribution fortement asymétrique vers la droite.
> - Population en 2023 (pop2023) : La majorité des pays ont une population inférieure à 200 millions d'habitants. Comme pour l'aire, il y a quelques pays (comme la Chine et l'Inde) qui ont une population beaucoup plus importante, créant une distribution fortement asymétrique vers la droite.
>
> </p>
>
> - Densité de population (density) : La densité de population est également fortement asymétrique vers la droite. La plupart des pays ont une densité de population inférieure à 5000 personnes par km², mais il y a quelques exceptions avec une densité de population très élevée.
>
> </p>
>
> </blocquote>

In [11]:
```
# Evolution de la population de 1980 à 2023 pour chaque pays
df["pop_change_1980_2023"] = df["pop2023"] - df["pop1980"]

# Top 10 des pays avec la plus forte augmentation de population
top_pop_increase = df.nlargest(10, "pop_change_1980_2023")

# Top 10 des pays ayant la plus forte diminution de population
top_pop_decrease = df.nsmallest(10, "pop_change_1980_2023")

fig, ax = plt.subplots(2,1, figsize=(15,20))

ax[0].barh(top_pop_increase["country"], top_pop_increase["pop_change_1980_2023"], color="red"
ax[0].invert_yaxis()
ax[0].set_xlabel("Évolution de la population de 1980 à 2023")
ax[0].set_title("Top 10 des pays ayant la plus forte augmentation de population de 1980 à 202


ax[1].barh(top_pop_decrease["country"], top_pop_decrease["pop_change_1980_2023"], color="brow
ax[1].invert_yaxis()
ax[1].set_xlabel("Évolution de la population de 1980 à 2023")
ax[1].set_title("Top 10 des pays ayant connu la plus forte baisse de population entre 1980 et

plt.tight_layout()
plt.show()
```


Top 10 des pays ayant la plus forte augmentation de population de 1980 à 2023
India
China
Pakistan
Nigeria
Indonesia
United States
Brazil
Ethiopia
Bangladesh
DR Congo
0
1
2
3
4
5
6
7
1e8
Évolution de la population de 1980 à 2023


Top 10 des pays ayant connu la plus forte baisse de population entre 1980 et 2023
Ukraine
Bulgaria
Romania
Georgia
Bosnia and Herzegovina
Lithuania
Latvia
Croatia
Moldova
Serbia
−1.2
−1.0
−0.8
−0.6
−0.4
−0.2
0.0
1e7
Évolution de la population de 1980 à 2023

D'après les graphiques ci-dessus, nous pouvons observer les points suivants :

> - Top 10 des pays avec la plus grande augmentation de population de 1980 à 2023 : L'Inde, la Chine et les États-Unis ont connu la plus grande augmentation de la population au cours de cette période. Il convient de noter que malgré une croissance plus faible que celle de l'Inde, la Chine a encore une augmentation significative de la population en raison de sa grande population de départ.
>
> </p>
>
> - Top 10 des pays avec la plus grande diminution de population de 1980 à 2023 : Certains pays, comme la Bulgarie, la Lettonie et la Lituanie, ont connu une diminution de la population au cours de cette période. Cela pourrait être dû à divers facteurs, tels que l'émigration, le vieillissement de la population, ou des taux de fécondité plus faibles.
>
> </p> </blocquote:>
>
> Ces informations pourraient être utiles pour comprendre les tendances démographiques à long terme dans différents pays. Par exemple, les pays avec une croissance démographique rapide pourraient avoir besoin de plus d'infrastructures et de services pour répondre à la demande croissante, tandis que les pays avec une diminution de la population pourraient avoir besoin de stratégies pour gérer une population vieillissante ou pour attirer de nouveaux résidents.

Qu'en est il du canada et de la France ?

In [12]:
```
# Information sur le Canada
canada_data = df[df['country'] == 'Canada']
canada_data.transpose()
```

Out[12]:

	37
country	Canada
rank	38
area	9984670.0
landAreaKm	8965590.0
cca2	CA
cca3	CAN
netChange	0.0104
growthRate	0.0085
worldPercentage	0.0048
density	4.3256
densityMi	11.2032
place	124
pop1980	24511510
pop2000	30683313
pop2010	33963412
pop2022	38454327
pop2023	38781291
pop2030	41008596
pop2050	45890819
pop_change_1980_2023	14269781

> Le Canada est le 38ème pays en termes de population en 2023.
>
> - Il a une superficie d'environ 9,98 millions de km², ce qui en fait l'un des plus grands pays du monde en termes de superficie.
> - Le taux de croissance de la population est de 0,0085, ce qui est relativement faible par rapport à certains autres pays.
> - La densité de population est également faible, avec environ 4,33 personnes par km². Cela est dû à la grande superficie du pays.
> - La population est passée de 24,5 millions en 1980 à prévu de 38,8 millions en 2023, soit une augmentation de 14,3 millions.

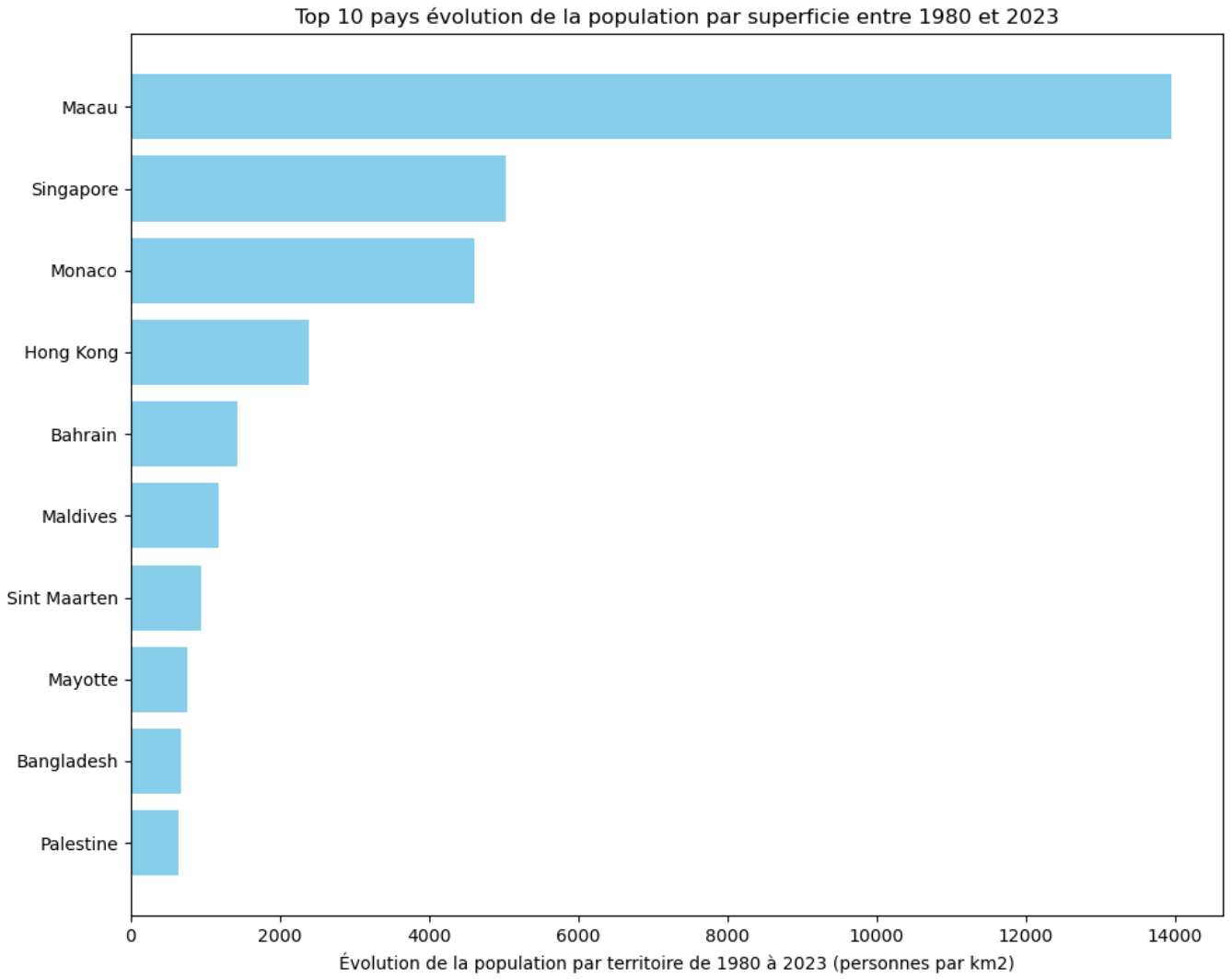
In [13]:
```
# Information sur la France
France_data = df[df['country'] == 'France']
France_data.transpose()
```

Out[13]:

	22
country	France
rank	23
area	551695.0
landAreaKm	547557.0
cca2	FR
cca3	FRA
netChange	0.004
growthRate	0.002
worldPercentage	0.0081
density	118.2646
densityMi	306.3052
place	250
pop1980	53713830
pop2000	58665453
pop2010	62444567
pop2022	64626628
pop2023	64756584
pop2030	65543452
pop2050	65827072
pop_change_1980_2023	11042754

> La France est le 23ème pays en termes de population en 2023.
>
> - Il a une superficie d'environ 5,51 millions de km²
> - Le taux de croissance de la population est de 0,002, ce qui est relativement faible par rapport à certains autres pays.
> - La densité de population est grande comparée au Canada, avec environ 118,26 personnes par km².
> - La population est passée de 53,71 millions en 1980 à 65,82 millions en 2023, soit une augmentation de 11,04 millions.

In [14]:
```
# Evolution de la population par superficie pour chaque pays
df['pop_change_per_area'] = df['pop_change_1980_2023'] / df['landAreaKm']

#  Top 10 des pays où la population a le plus changé par rapport à la superficie du territoire
top_pop_change_per_area = df.nlargest(10, 'pop_change_per_area')

fig, ax = plt.subplots(figsize=(10, 8))

ax.barh(top_pop_change_per_area['country'], top_pop_change_per_area['pop_change_per_area'], c
ax.invert_yaxis()
ax.set_xlabel('Évolution de la population par territoire de 1980 à 2023 (personnes par km2)')
ax.set_title('Top 10 pays évolution de la population par superficie entre 1980 et 2023')

plt.tight_layout()
plt.show()
```

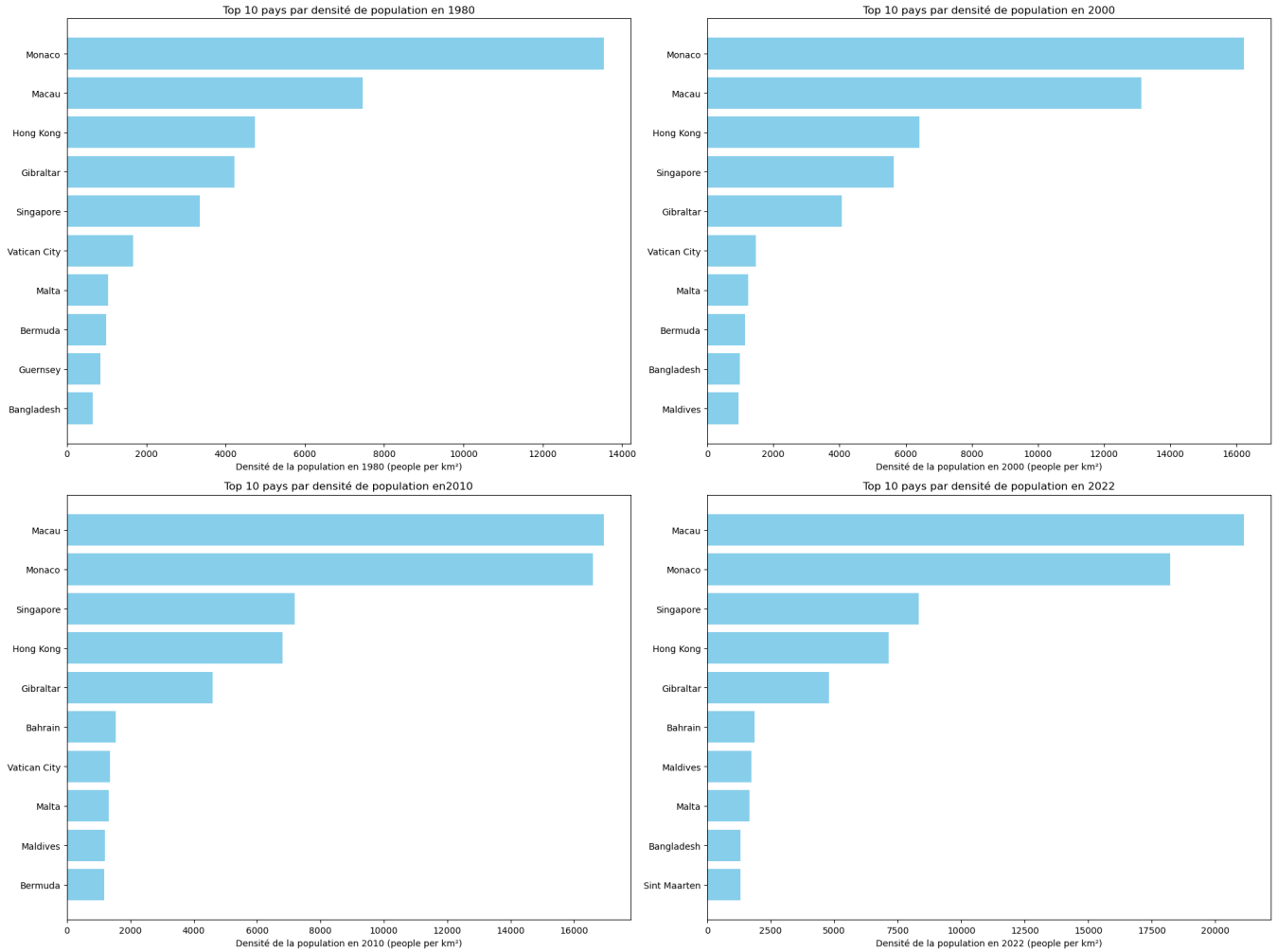


L'histogramme ci-dessus montre les 10 pays avec le plus grand changement de population par unité de superficie terrestre de 1980 à 2023.

> - Singapour, le Koweït et le Liban sont les pays où l'augmentation de la population par km^2 de superficie terrestre a été la plus forte au cours de cette période.
> - Cela suggère que ces pays ont vu une augmentation significative de la densité de population au cours de cette période.
> - Il convient de noter que ces pays sont relativement petits en termes de superficie terrestre, ce qui peut expliquer pourquoi une augmentation de la population peut entraîner une augmentation significative de la densité de population.

In [15]:

```
# densité de population pour chaque année et pour chaque pays
df['density_1980'] = df['pop1980'] / df['landAreaKm']
df['density_2000'] = df['pop2000'] / df['landAreaKm']
df['density_2010'] = df['pop2010'] / df['landAreaKm']
df['density_2022'] = df['pop2022'] / df['landAreaKm']

# Top 10 pays ayant la plus forte densité de population en 1980, 2000, 2010, 2022
top_density_1980 = df.nlargest(10, 'density_1980')
top_density_2000 = df.nlargest(10, 'density_2000')
top_density_2010 = df.nlargest(10, 'density_2010')
top_density_2022 = df.nlargest(10, 'density_2022')

# Plotting
fig, ax = plt.subplots(2, 2, figsize=(20, 15))

ax[0, 0].barh(top_density_1980['country'], top_density_1980['density_1980'], color='skyblue')
ax[0, 0].invert_yaxis()
ax[0, 0].set_xlabel('Densité de la population en 1980 (people per km²)')
ax[0, 0].set_title('Top 10 pays par densité de population en 1980')

ax[0, 1].barh(top_density_2000['country'], top_density_2000['density_2000'], color='skyblue')
ax[0, 1].invert_yaxis()
ax[0, 1].set_xlabel('Densité de la population en 2000 (people per km²)')
ax[0, 1].set_title('Top 10 pays par densité de population en 2000')

ax[1, 0].barh(top_density_2010['country'], top_density_2010['density_2010'], color='skyblue')
ax[1, 0].invert_yaxis()
ax[1, 0].set_xlabel('Densité de la population en 2010 (people per km²)')
ax[1, 0].set_title('Top 10 pays par densité de population en2010')

ax[1, 1].barh(top_density_2022['country'], top_density_2022['density_2022'], color='skyblue')
ax[1, 1].invert_yaxis()
ax[1, 1].set_xlabel('Densité de la population en 2022 (people per km²)')
ax[1, 1].set_title('Top 10 pays par densité de population en 2022')

plt.tight_layout()
plt.show()
```



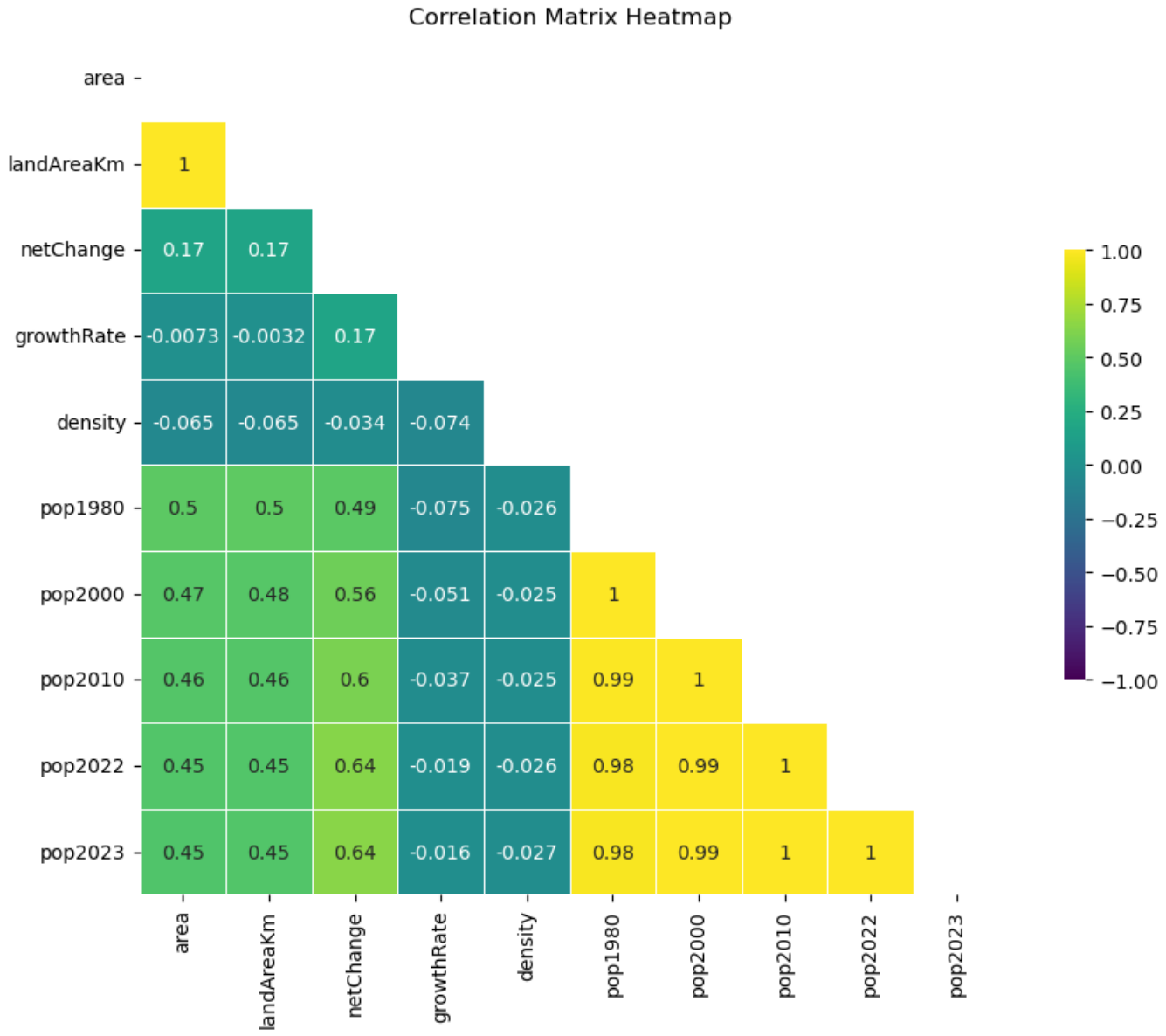
Les graphiques ci-dessus montrent les 10 pays avec la plus grande densité de population en 1980, 2000, 2010 et 2022. Voici quelques observations :

> - 1980 : En 1980, Monaco avait la plus grande densité de population, suivie de près par Macao et Singapour.
> - 2000 : En 2000, la densité de population de Macao a dépassé celle de Monaco, faisant de Macao le pays avec la plus grande densité de population.
>
>   </p>
> - 2010 : En 2010, la densité de population de Macao a continué à augmenter, tandis que celle de Monaco a légèrement diminué.
>
>   </p>
> - 2022 : En 2022, la densité de population de Macao est toujours la plus élevée, suivie de Monaco et de Singapour.
>
>   </p> </blockquote>
>
>   Dans l'ensemble, on observe que les pays avec les plus grandes densités de population sont généralement des petits territoires ou des cités-États comme Monaco, Macao et Singapour. Cependant, il est intéressant de noter que certains pays plus grands, comme le Liban et Taïwan, figurent également parmi les 10 pays les plus densément peuplés.

Correlations

In [16]:

```
# matrice de correlation
correlation_matrix = df[['area', 'landAreaKm', 'netChange', 'growthRate', 'density', 'pop1980

plt.figure(figsize=(12, 8))

# Générer un masque pour le triangle supérieur
mask = np.triu(np.ones_like(correlation_matrix, dtype=bool))

#  heatmap with avec le masque et le bon rapport hauteur/largeur
sns.heatmap(correlation_matrix, mask=mask, cmap='viridis', vmax=1, vmin=-1,
            square=True, linewidths=.5, cbar_kws={"shrink": .5}, annot=True)

plt.title('Correlation Matrix Heatmap')
plt.show()
```

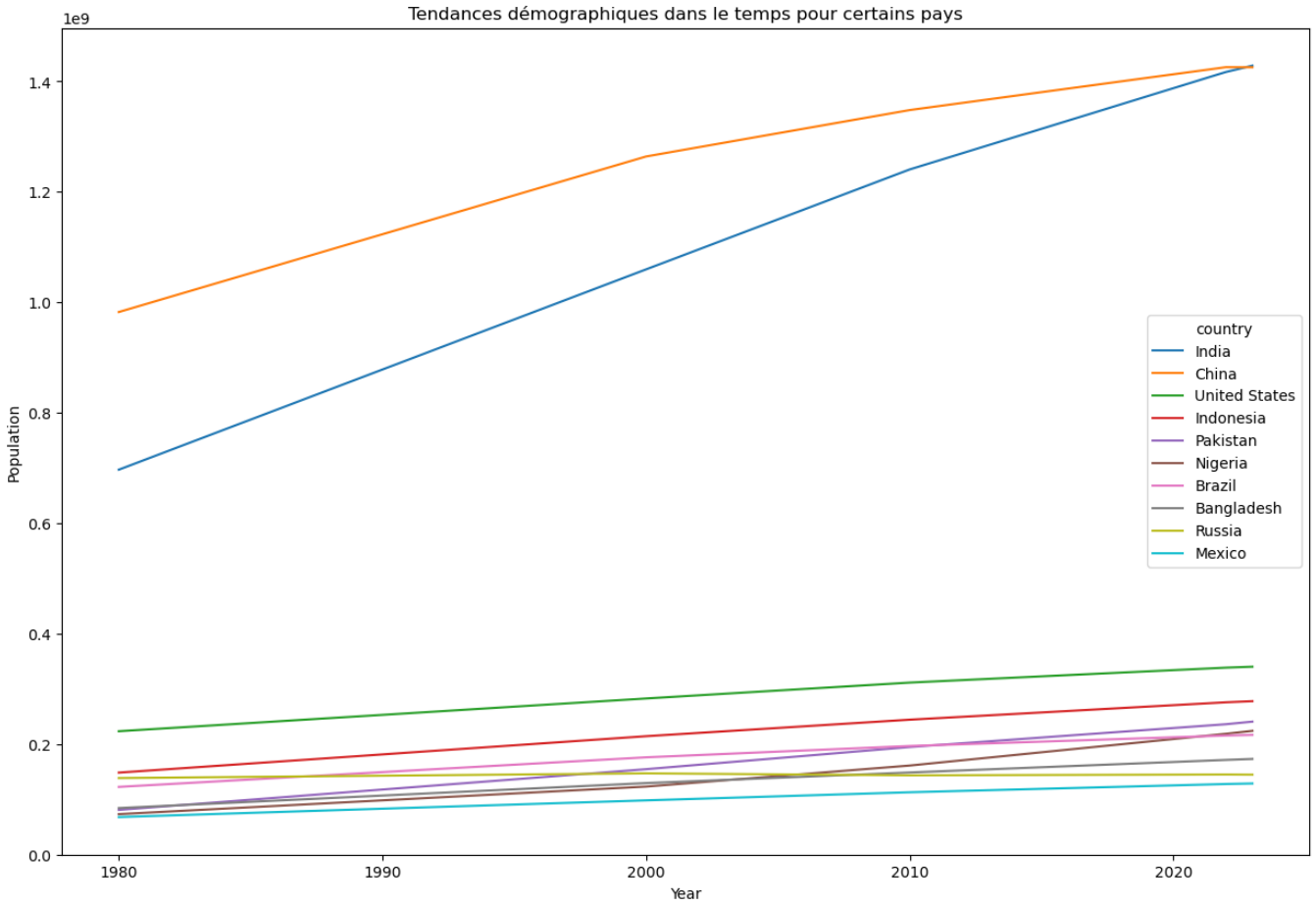


La matrice de corrélation ci-dessus montre les coefficients de corrélation entre différentes paires de variables. Voici quelques observations :

> - area et landAreaKm sont fortement corrélées (0.997406), ce qui est attendu puisqu'ils mesurent essentiellement la même chose (la superficie du pays).
> - Les variables pop1980, pop2000, pop2010, pop2022, et pop2023 sont fortement corrélées entre elles. Cela indique que les pays qui avaient une grande population dans le passé ont tendance à avoir une grande population dans le présent et vice versa.
> - netChange est modérément corrélé avec pop1980, pop2000, pop2010, pop2022, et pop2023, indiquant que les pays avec une population plus importante ont tendance à avoir un changement net plus important.
> - growthRate a une faible corrélation négative avec density, ce qui suggère que les pays avec une densité de population plus élevée ont tendance à avoir un taux de croissance de la population plus faible. Cependant, cette corrélation est assez faible, donc cette relation peut ne pas être très forte.
> - growthRate a également une faible corrélation positive avec netChange, ce qui suggère que les pays avec un taux de croissance de la population plus élevé ont tendance à avoir un changement net plus important.
>
> </blockquote>
>
> Il est important de noter que la corrélation ne prouve pas la causalité et qu'il pourrait y avoir d'autres facteurs non inclus dans ce jeu de données qui influencent ces variables. De plus, ces coefficients de corrélation sont basés sur des données agrégées pour chaque pays, donc ils ne capturent pas nécessairement les variations à l'intérieur des pays.

Analyse des tendances temporelles

In [17]:
```
# Tendances dans le temps pour un sous-ensemble de pays
subset_countries = ['China', 'India', 'United States', 'Indonesia', 'Pakistan', 'Brazil', 'Nig

subset_df = df[df['country'].isin(subset_countries)]
subset_df = subset_df[['country', 'pop1980', 'pop2000', 'pop2010', 'pop2022', 'pop2023']]

# Transformation en format long
subset_df_melted = subset_df.melt(id_vars='country', var_name='year', value_name='population'

# Conversion
subset_df_melted['year'] = subset_df_melted['year'].str[3:].astype(int)

# Plotting
plt.figure(figsize=(15, 10))
sns.lineplot(data=subset_df_melted, x='year', y='population', hue='country')
plt.title('Tendances démographiques dans le temps pour certains pays')
plt.xlabel('Year')
plt.ylabel('Population')

plt.show()
```



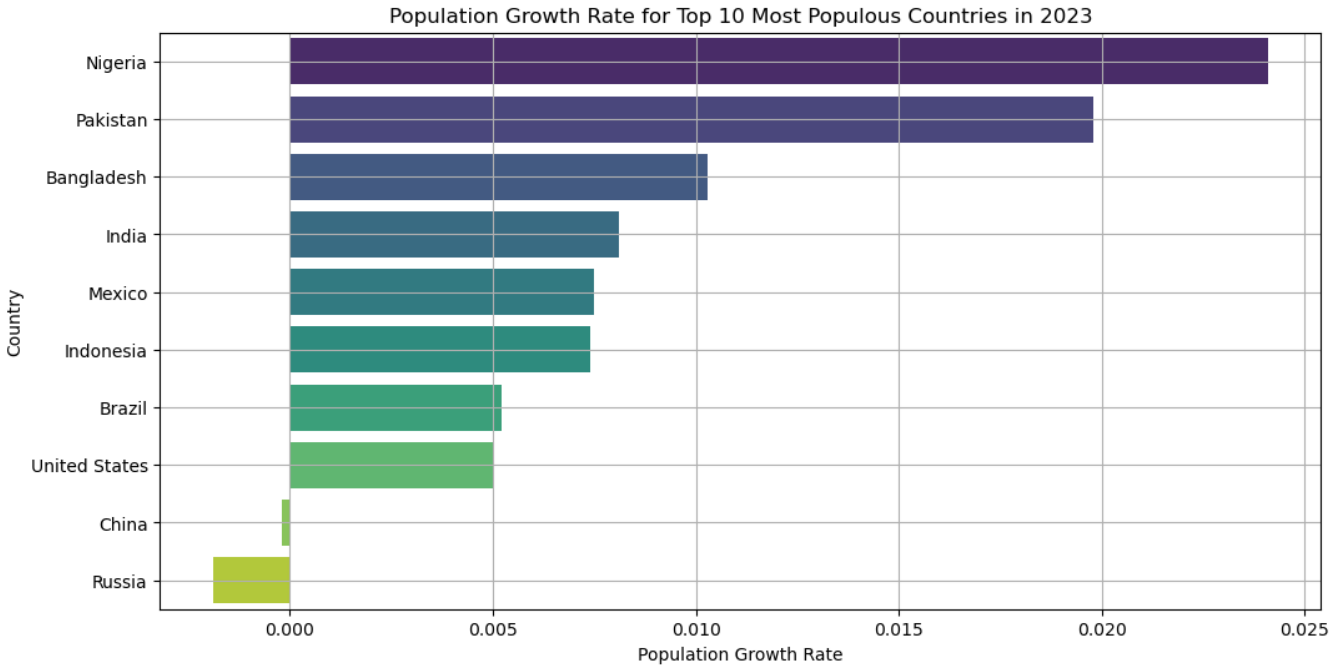
Le graphique ci-dessus montre les tendances de la population au fil du temps pour une sélection de pays. Voici quelques observations :

> - La population de la Chine et de l'Inde a augmenté de manière significative au cours de cette période. L'Inde, en particulier, semble connaître une croissance rapide de sa population.
> - Les États-Unis, l'Indonésie et le Pakistan ont également connu une augmentation de leur population, bien que à un rythme plus lent.
> - La population de la Russie semble avoir diminué ou être restée relativement stable au cours de cette période.

Ces tendances suggèrent que la croissance de la population est un phénomène complexe qui peut être influencé par une variété de facteurs, y compris les politiques gouvernementales, les taux de fécondité, l'espérance de vie, et les taux d'immigration et d'émigration.

Taux de croissance de la population pour les 10 pays les plus peuplés en 2023

In [18]:
```
# diagramme à barres montrant le taux de croissance de la population des 10 pays les plus peup

top10_populous_2023 = df.nlargest(10, 'pop2023')['country']

#  taux de croissance de la population pour ces pays
growth_rate_top10 = df[df['country'].isin(top10_populous_2023)][['country', 'growthRate']]

# Trier les pays par taux de croissance
growth_rate_top10 = growth_rate_top10.sort_values('growthRate', ascending=False)

# Plotting
plt.figure(figsize=(12, 6))
sns.barplot(x='growthRate', y='country', data=growth_rate_top10, palette='viridis')
plt.title('Population Growth Rate for Top 10 Most Populous Countries in 2023')
plt.xlabel('Population Growth Rate')
plt.ylabel('Country')
plt.grid(True)
plt.show()
```



Nous allons maintenant passer à l'analyse de regroupement des pays. Pour cela, nous allons utiliser une technique de clustering, telle que le K-means clustering, pour regrouper les pays en fonction de caractéristiques similaires.

In [19]:
```
df_clean = df.dropna(subset=['netChange', 'worldPercentage', 'cca2'])
```

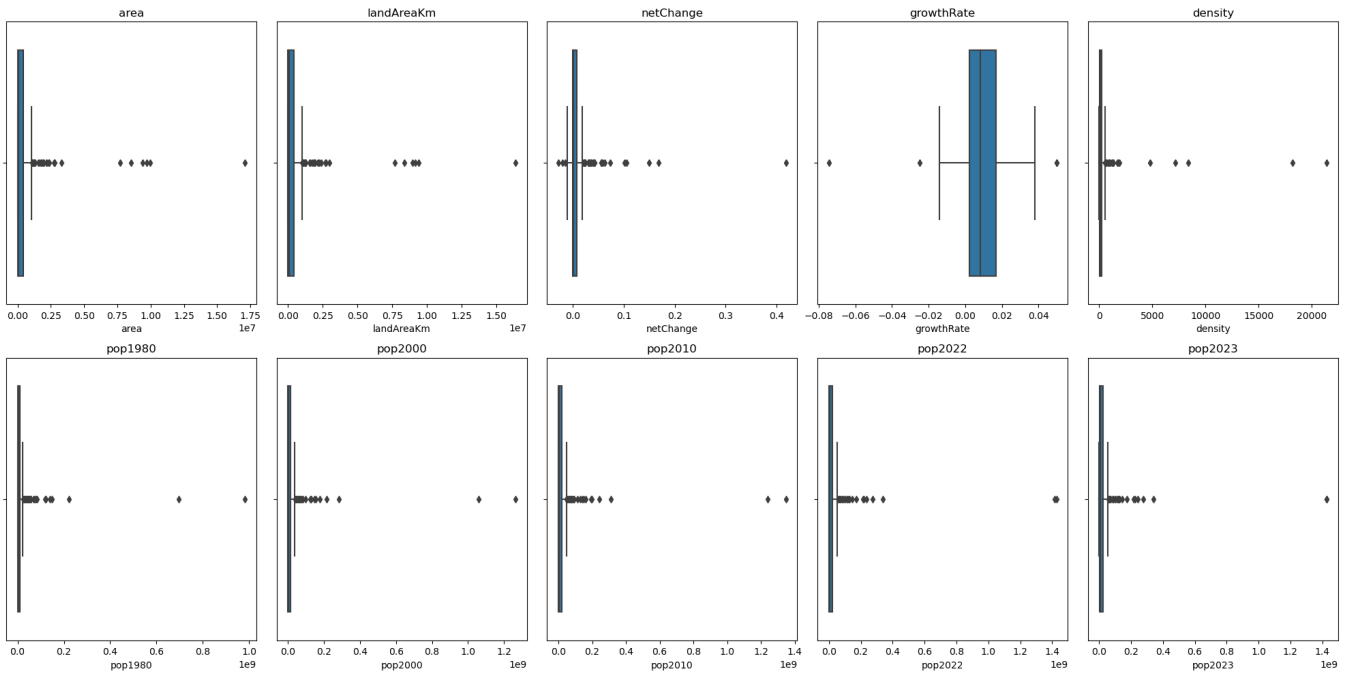
In [20]:
```
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler

# Select features for clustering
features = ['area', 'landAreaKm', 'netChange', 'growthRate', 'density', 'pop1980', 'pop2000',

# Standardize the features
scaler = StandardScaler()
scaled_features = scaler.fit_transform(df_clean[features])

# Perform KMeans clustering
kmeans = KMeans(n_clusters=5, random_state=42)
kmeans.fit(scaled_features)

# Add cluster labels to the dataframe
df_clean['cluster'] = kmeans.labels_

# Display the size of each cluster
df_clean['cluster'].value_counts().sort_index()
```

Out[20]:
```
0    142
1      2
2      5
3      2
4     73
Name: cluster, dtype: int64
```

Nous avons réussi à créer 5 clusters à partir de nos données. Voici le nombre de pays dans chaque cluster :

> Cluster 0 : 142 pays
>
> Cluster 1 : 2 pays
>
> Cluster 2 : 5 pays
>
> Cluster 3 : 2 pays
>
> Cluster 4 : 73 pays
>
> </blockquote> Cela suggère que la plupart des pays se situent dans les clusters 0 et 4, tandis que les clusters 1, 2 et 3 contiennent un nombre relativement faible de pays.

In [21]:
```
# boxplots pour visualiser les valeurs aberrantes.
fig, ax = plt.subplots(2, 5, figsize=(20, 10))

for i, feature in enumerate(features):
    sns.boxplot(x=df[feature], ax=ax[i//5, i%5])
    ax[i//5, i%5].set_title(feature)

plt.tight_layout()
plt.show()
```



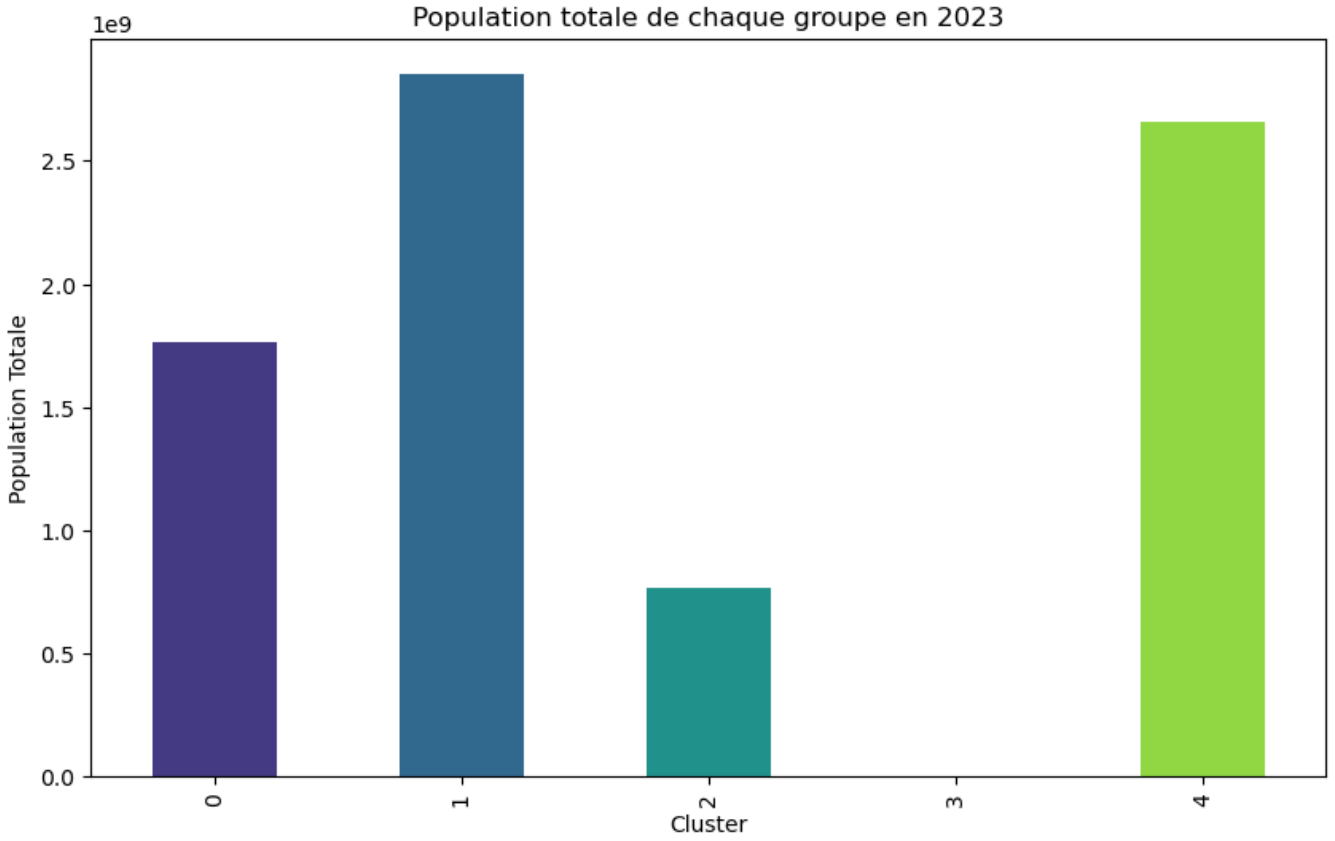
> Les graphiques à moustaches ci-dessus montrent la distribution de chaque caractéristique dans nos données. Les points en dehors des "moustaches" des graphiques à moustaches sont considérés comme des valeurs aberrantes. Voici quelques observations :
>
> Ces valeurs aberrantes sont probablement dues à la présence de pays très peuplés ou très grands.
>
> Ces valeurs aberrantes peuvent avoir une grande influence sur les résultats de l'analyse. cependant, je vais les conserver car elles peuvent représenter des informations précieuses ou intéressantes.
>
> </blocquote>

In [22]:
```
# valeurs moyennes pour chaque groupe
cluster_means = df_clean.groupby('cluster')[features].mean()

# Affichage des valeurs moyennes des clusters
cluster_means.transpose()
```

Out[22]:

cluster	0	1	2	3	4
area	1.849054e+05	6.497276e+06	1.053266e+07	17.46000	5.932312e+05
landAreaKm	1.690771e+05	6.198946e+06	1.010801e+07	17.45000	5.832978e+05
netChange	1.660563e-03	2.035500e-01	2.006000e-02	0.00015	2.170548e-02
growthRate	3.612676e-03	3.950000e-03	5.360000e-03	0.00410	2.300822e-02
density	3.705139e+02	3.158865e+02	1.592900e+01	19775.60260	1.402950e+02
pop1980	8.545361e+06	8.396004e+08	1.045807e+08	136204.00000	1.404854e+07
pop2000	1.063570e+07	1.161866e+09	1.309637e+08	232180.50000	2.268328e+07
pop2010	1.149305e+07	1.294402e+09	1.413523e+08	295237.50000	2.812929e+07
pop2022	1.239412e+07	1.421530e+09	1.525897e+08	365818.50000	3.567204e+07
pop2023	1.241911e+07	1.427150e+09	1.532168e+08	370223.00000	3.638757e+07

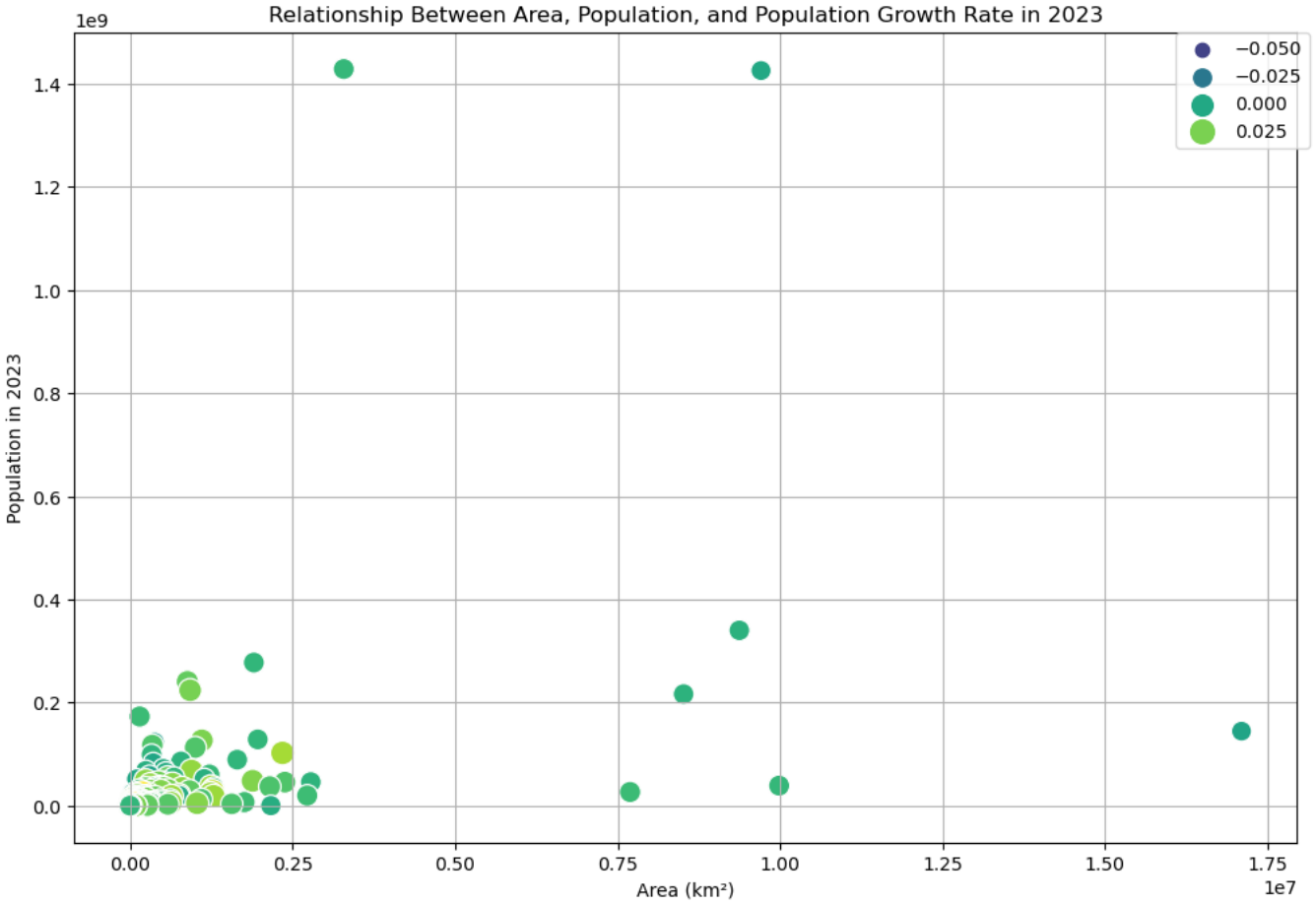
In [23]:
```
# population totale de chaque groupe en 2023
cluster_pop2023 = df_clean.groupby('cluster')['pop2023'].sum()

# Plotting
plt.figure(figsize=(10, 6))
cluster_pop2023.sort_index().plot(kind='bar', stacked=True, color=sns.color_palette('viridis'
plt.title('Population totale de chaque groupe en 2023')
plt.xlabel('Cluster')
plt.ylabel('Population Totale')
plt.show()
```



In [ ]:

Nous allons créer un diagramme à bulles montrant la relation entre la superficie, la population et le taux de croissance de la population pour chaque pays

In [24]:
```
# 5. Bubble Chart of Population vs Area, colored by Growth Rate
plt.figure(figsize=(12, 8))
sns.scatterplot(x='area', y='pop2023', size='growthRate', sizes=(20, 200), hue='growthRate', 
plt.title('Relationship Between Area, Population, and Population Growth Rate in 2023')
plt.xlabel('Area (km²)')
plt.ylabel('Population in 2023')
plt.grid(True)
plt.legend(bbox_to_anchor=(1.01, 1),borderaxespad=0)
plt.show()
```



Le diagramme à bulles ci-dessus montre la relation entre la superficie, la population et le taux de croissance de la population pour chaque pays en 2023. Les couleurs représentent le taux de croissance de la population. Voici quelques observations :

- Les pays avec une grande superficie ont généralement une grande population, mais il y a de nombreuses exceptions à cette tendance. Par exemple, certains pays ont une grande superficie mais une petite population, tandis que d'autres ont une petite superficie mais une grande population.
- Le taux de croissance de la population, représenté par la taille et la couleur des bulles, varie considérablement d'un pays à l'autre. Certains pays ont un taux de croissance élevé (grandes bulles, couleur foncée), tandis que d'autres ont un taux de croissance faible (petites bulles, couleur claire).

Il est intéressant de noter que certains pays avec une grande population et une grande superficie ont un taux de croissance relativement faible, tandis que certains pays plus petits ont un taux de croissance plus élevé.

Fin