Vizualizace dat

Zadání

V jednom ze cvičení jste probírali práci s moduly pro vizualizaci dat. Mezi nejznámější moduly patří matplotlib (a jeho nadstavby jako seaborn), pillow, opencv, aj. Vyberte si nějakou zajímavou datovou sadu na webovém portále Kaggle a proveďte datovou analýzu datové sady. Využijte k tomu různé typy grafu a interpretujte je (minimálně alespoň 5 zajímavých grafu). Příklad interpretace: z datové sady pro počasí vyplynulo z liniového grafu, že v létě je vyšší rozptyl mezi minimální a maximální hodnotou teploty. Z jiného grafu vyplývá, že v létě je vyšší průměrná vlhkost vzduchu. Důvodem vyššího rozptylu může být absorpce záření vzduchem, který má v létě vyšší tepelnou kapacitu.

Řešení

Pro vizualizaci dat jsem si vybral data set, obsahující různé údaje o populaci mnoha zemí světa a zaměřil se na 10 největších dle celkové populace. Data set jsem získal ze stránky kaggle. Struktura dat je následující.

A	B	C	D	E	F	G	H	I	J	K	L
Country	Year	Total Population	Urban Population	Rural Population	Population Density	Life Expectancy	Birth Rate	Death Rate	Fertility Rate	Infant Mortality Rate	Growth Rate
Afghanistan	2017	0	0	0	55	63	37.342	7.027	5.129	49.4	0
Afghanistan	2018	36,686,784.00	9,353,296.00	27,333,488.00	56	63	36.927	6.981	5.002	47.8	3
Afghanistan	2019	37,769,499.00	9,727,157.00	28,042,342.00	58	64	36.466	6.791	4.87	46.3	3
Afghanistan	2020	38,972,230.00	10,142,913.00	28,829,317.00	60	63	36.051	7.113	4.75	44.8	3
Afghanistan	2021	40,099,462.00	10,551,772.00	29,547,690.00	0	62	35.842	7.344	4.643	43.4	3
Albania	2017	0	0	0	105	79	10.87	8.15	1.491	8.3	0
Albania	2018	2,866,376.00	1,728,969.00	1,137,407.00	105	79	10.517	8.308	1.44	8.3	0
Albania	2019	2,854,191.00	1,747,593.00	1,106,598.00	104	79	10.343	8.48	1.414	8.4	0
Albania	2020	2,837,849.00	1,762,645.00	1,075,204.00	104	77	10.285	10.785	1.4	8.4	1
Albania	2021	2,811,666.00	1,770,478.00	1,041,188.00	0	76	10.24	11.325	1.39	8.4	1
Algeria	2017	0	0	0	17	76	24.755	4.542	3.05	21	0
Algeria	2018	41,927,007.00	30,451,166.00	11,475,841.00	18	76	24.074	4.482	3.023	20.6	2
Algeria	2019	42,705,368.00	31,255,632.00	11,449,736.00	18	76	23.298	4.392	2.988	20.1	2
Algeria	2020	43,451,666.00	32,038,217.00	11,413,449.00	18	74	22.431	5.398	2.942	19.6	2
Algeria	2021	44,177,969.00	32,807,002.00	11,370,967.00	0	76	21.524	4.546	2.889	19.2	2

Pro práci s daty a jejich vizualizaci jsem si vybral knihovny matplotlib, numpy a pandas.

Vizualizace.py > ...

```
1  import matplotlib.pyplot as plt
2  import matplotlib.ticker as ticker
3  import pandas as pd
4  import numpy as np
```

Nejdříve jsem si načetl data do proměnné csvreader pomocí pandas. A nastavil defaultní velikost okna, které zobrazuje grafy.

```
# Načtení CSV datového modelu
csv_path = r'C:\Users\lubos\OneDrive\Dokumenty\Škola\KMSW\PopulationDataNew.csv'
csvreader = pd.read_csv(csv_path, sep=';')
plt.figure(figsize=(20, 10))  # Nastavení defaultní velikosti okna
```

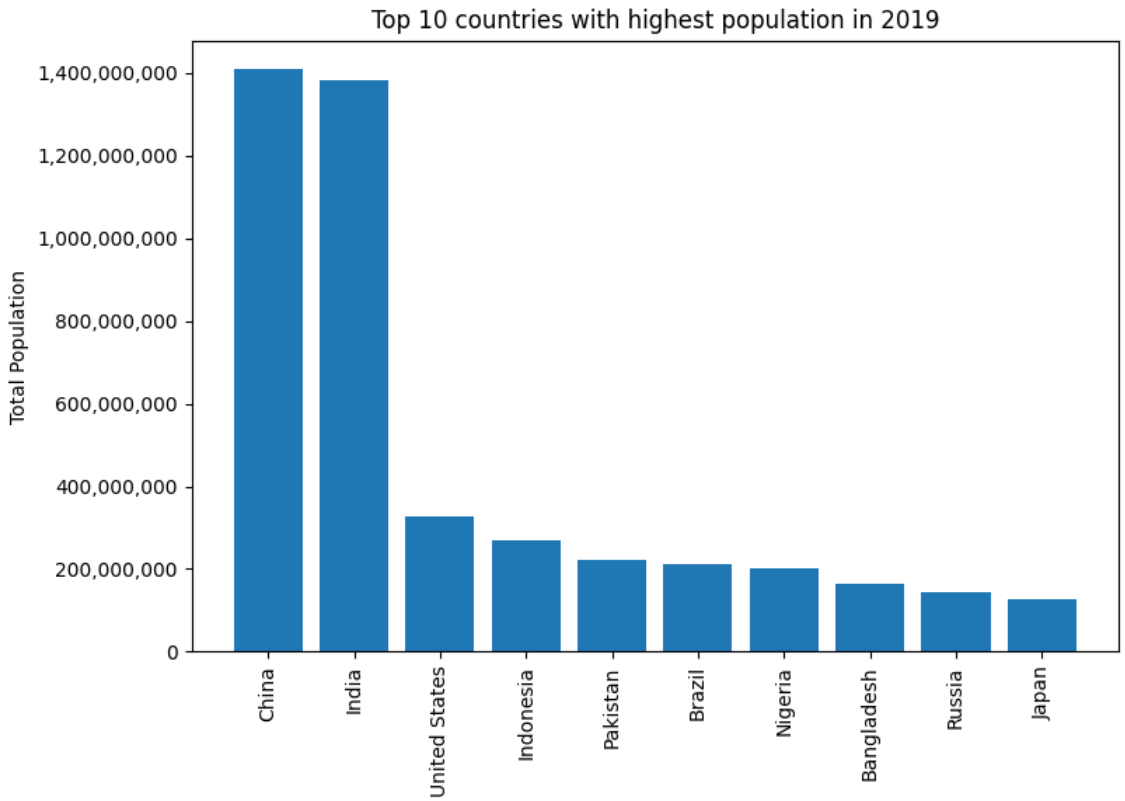
Poté jsem si vytvořil funkci **barChart()** pro graf typu bar. Tato funkce vytváří podgrafy, které zobrazují celkovou populaci 10 nejlidnatějších zemí světa. Nejdříve v roce 2019 a poté v roce

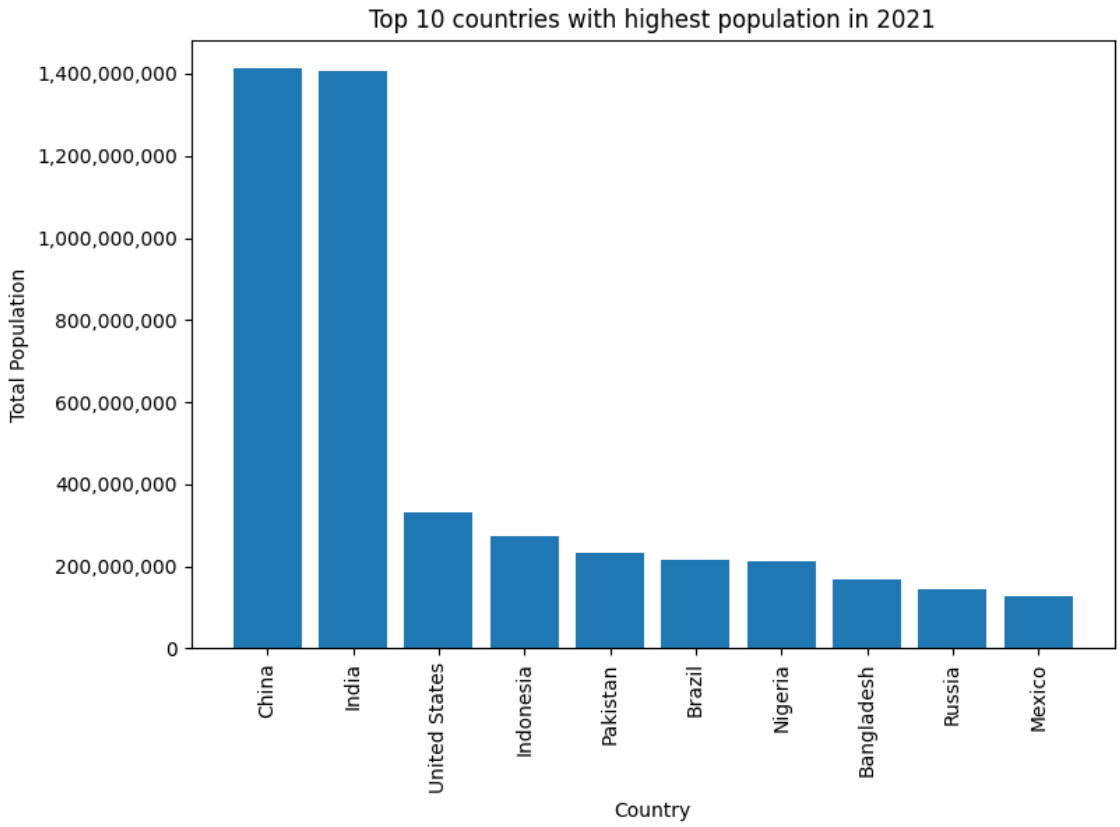
2021. Data si tedy nejdříve převedu do správného formátu a následně vyfiltruji dle daného roku. Následně je seřadím dle celkové populace a vyberu těch top 10 států.

```
# Filtrování dat podle roku
data = csvreader[csvreader['Year'] == year]

# Převedení pole na numerické
data['Total Population'] = data['Total Population'].astype(str).str.replace(',', '').astype(float)

# Seřazení dat dle celkové populace
data_sorted = data.sort_values(by='Total Population', ascending=False)
```

Pomocí těchto dat a příslušných sloupců nastavím osy x a y grafu. Také zde nastavím popisky os a další formátování.

```
#Vytvoření barového grafu a určení osy x a y
plt.bar(top_10_data['Country'], top_10_data['Total Population'])
#Nastavení nadpisů
plt.xlabel('Country')
plt.ylabel('Total Population')
plt.title(f'Top 10 countries with highest population in {year}')
plt.xticks(rotation=90)  # Rotace osy x k lepšímu přečtení delších hodnot
plt.ticklabel_format(style='plain', axis='y')
#Formátování hodnot osy y
plt.gca().get_yaxis().set_major_formatter(ticker.FuncFormatter(formatter))
```



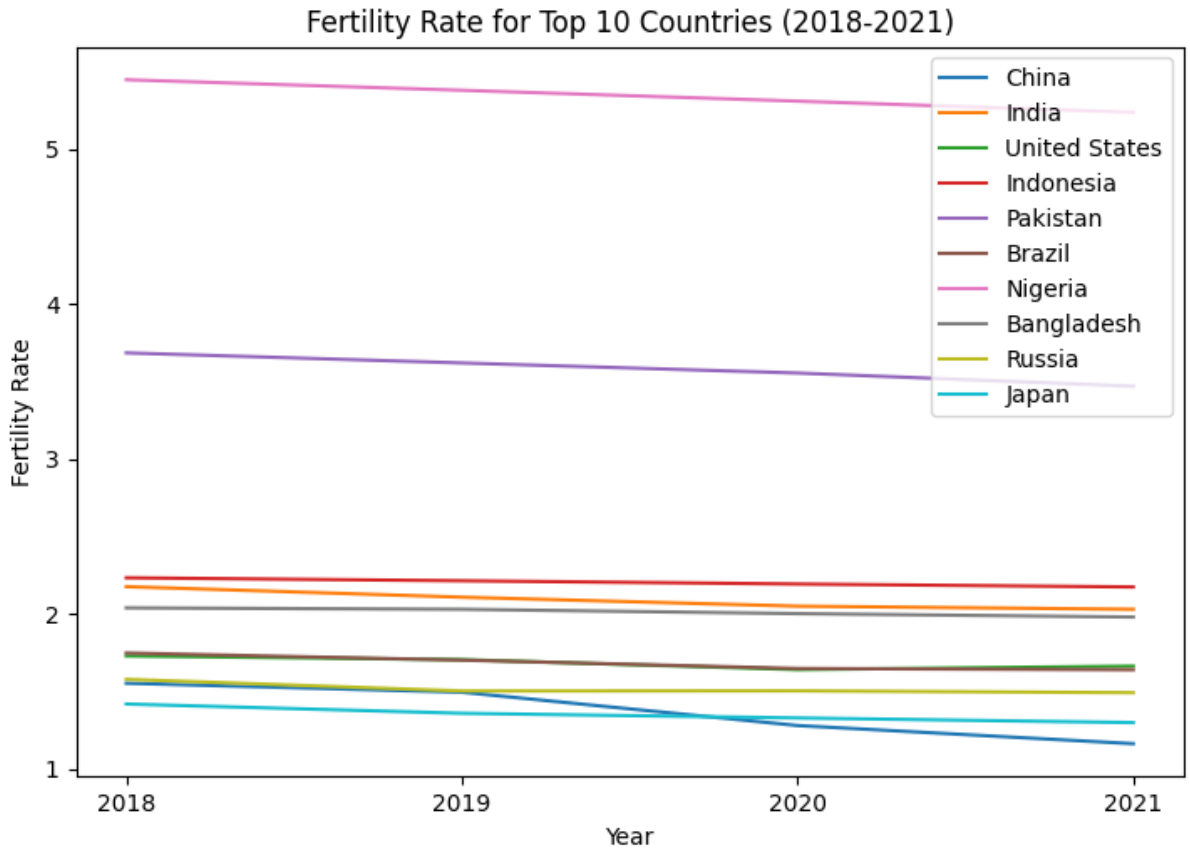


Další funkci, kterou jsem si vytvořil je **lineChart().** Ta slouží pro vytvoření line grafu, který zobrazuje hodnotu plodnosti mezi lety 2018–2021. Hodnota plodnosti udává přibližný počet potomků na jednu ženu v dané zemi. Data opět vyfiltruji jako v předchozím grafu. Poté iteruji těmito vyfiltrovanými daty a vždy přidám novou čáru do grafu pro danou zemi, a její data o plodnosti napříč lety 2018–2021.

```
for country in top_10_countries:
    country_data = data_filtered[data_filtered['Country'] == country]
    plt.plot(country_data['Year'],country_data['Fertility Rate'], label=country)
```

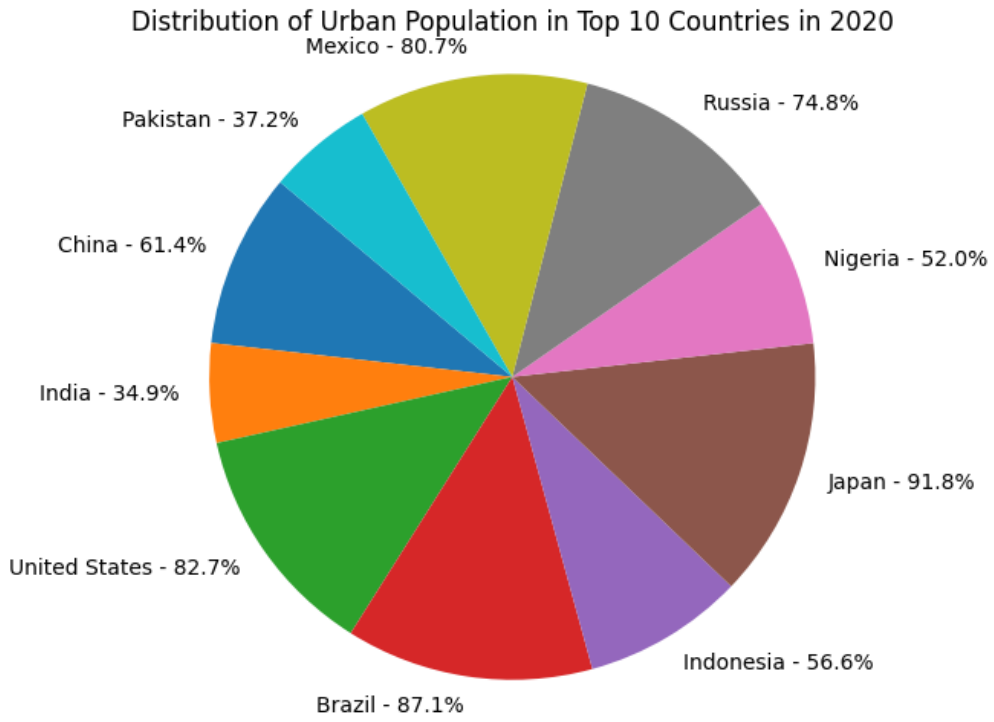
Následně provedu formátování grafu.

```
plt.xlabel('Year')
plt.ylabel('Fertility Rate')
plt.title(f'Fertility Rate for Top 10 Countries (2018-2021)')
plt.ticklabel_format(style='plain', axis='y')
plt.gca().get_yaxis().set_major_formatter(ticker.FuncFormatter(formatter))
plt.xticks(range(2018, 2022))
#Pozice legendy
plt.legend(loc='upper right')
```



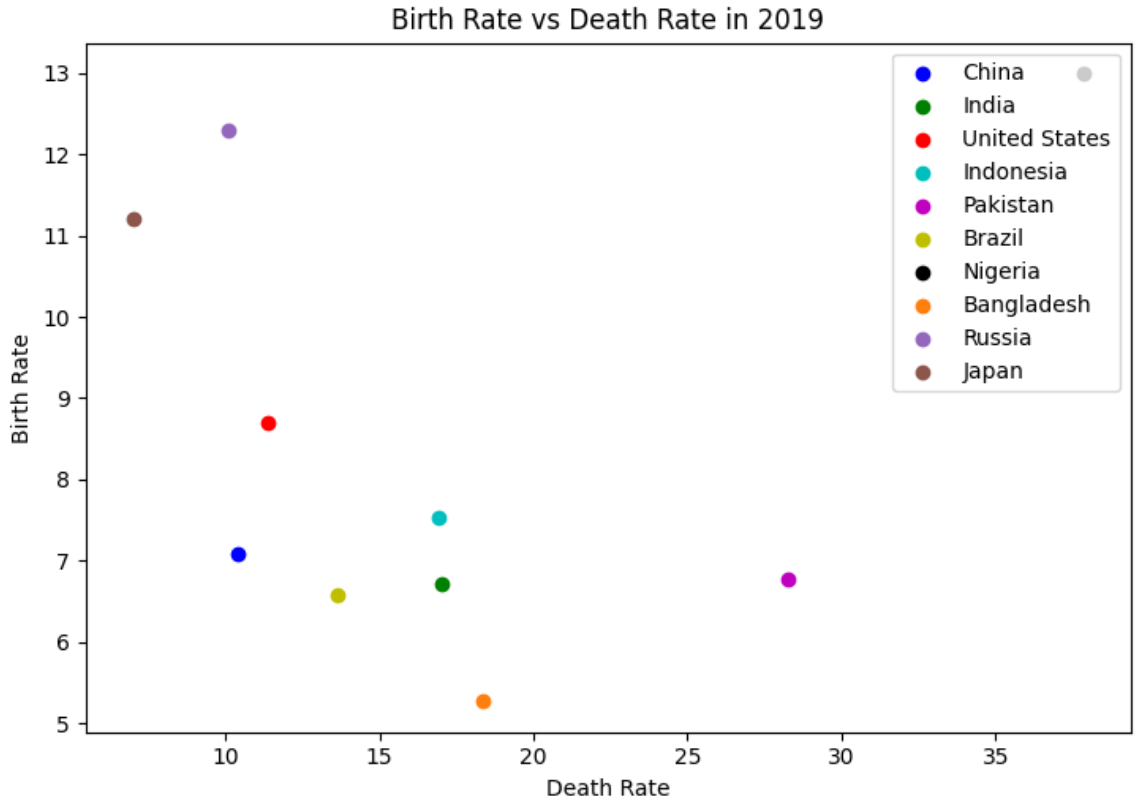
Třetí funkce **pieChart()** vytváří koláčový graf, který zobrazuje procentuální podíl populace žijící ve městech pro danou zemi. Graf opět používá vyfiltrovaná data pro rok 2020 a 10 nejlidnatějších zemí. Vypočítal jsem si procentuální podíl městské populace vůči celkové populaci. Nastavil jsem startovní úhel pro vytváření grafu. Také jsem si vytvořil seznam labels, který bude obsahovat zemi a její procentuální hodnotu.

```
top_10_data['Urban Percentage'] = (top_10_data['Urban Population'] / top_10_data['Total Population']) * 100
labels = [f"{country} - {percentage:.1f}%" for country, percentage in zip(top_10_data['Country'], top_10_data['Urban Percentage'])]
plt.pie(top_10_data['Urban Percentage'],labels=labels, startangle=140)
# Nastavení stejného poměru v koláčovém grafu
plt.axis('equal')
plt.title(f'Distribution of Urban Population in Top 10 Countries in {year}')
```



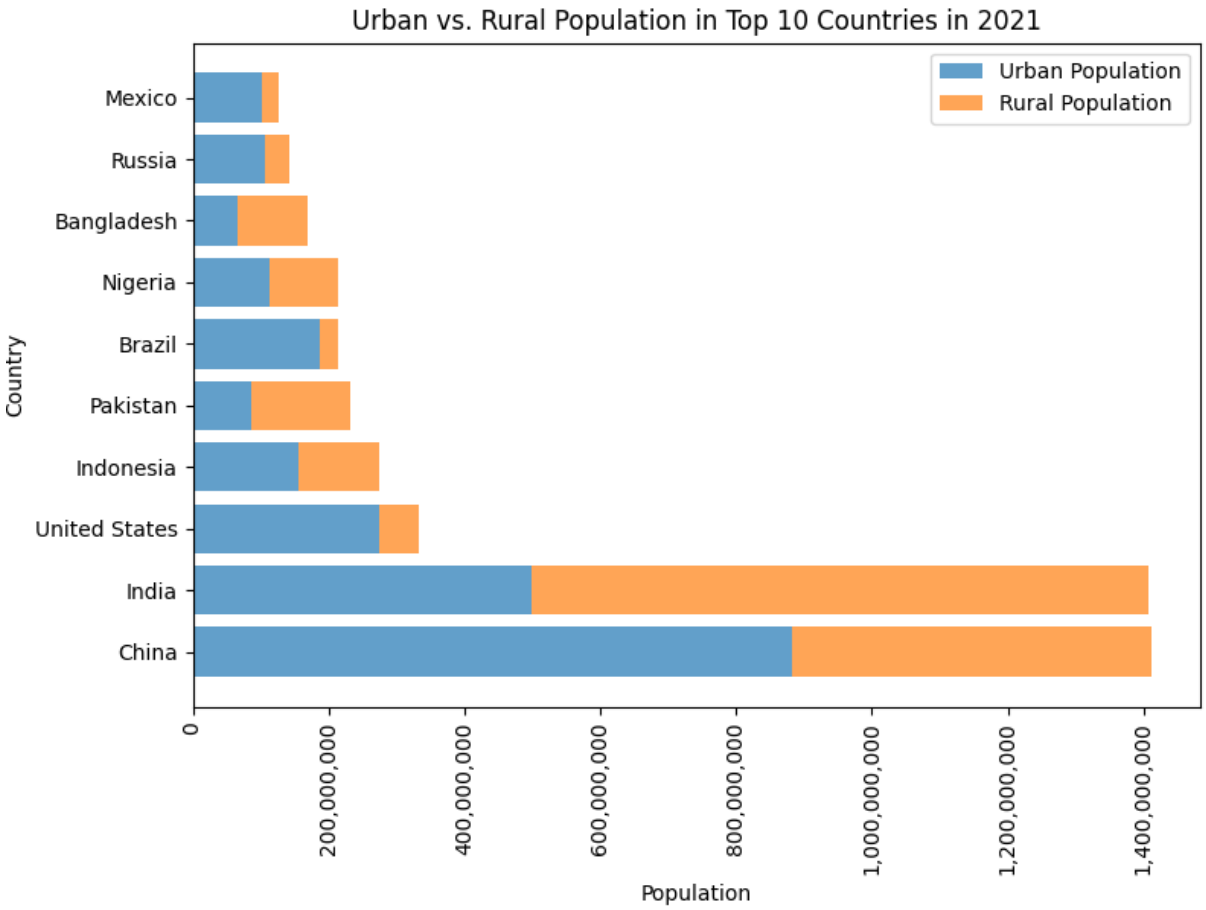
Funkce **scatter()** slouží pro vytvoření grafu typu scatter. Tento graf zobrazuje vztah mezi hodnotou úmrtnosti a porodnosti dané země v roce 2019. Vyfiltrovaná a seřazená data použiju k nastavení hodnot os x a y pro bod, který reprezentuje danou zemi. Každá země má přiřazenou svou barvu, aby se lišily v zobrazení grafu.

```
top_10_data = data_sorted.head(10)
for i, (_, country_data) in enumerate(top_10_data.iterrows()):
    country = country_data['Country']
    x_value = country_data['Birth Rate']
    y_value = country_data['Death Rate']
    color = colors[i]
# Přidání země do grafu dle jejích hodnot
    plt.scatter(x_value, y_value, color=color, label=country)
```



Poslední funkce **stacked()** vytváří graf typu stacked bar, který zobrazuje podíl městské populace a populace mimo velká města v dané zemi. Osa y pro tento graf je název dané země. Pro osu x nejdříve nastavíme hodnotu městské populace a následně mimo městské populace. Pomocí argumentu **left** určíme, že hodnoty mimo městské populace mají začínat vpravo od hodnot městské populace.

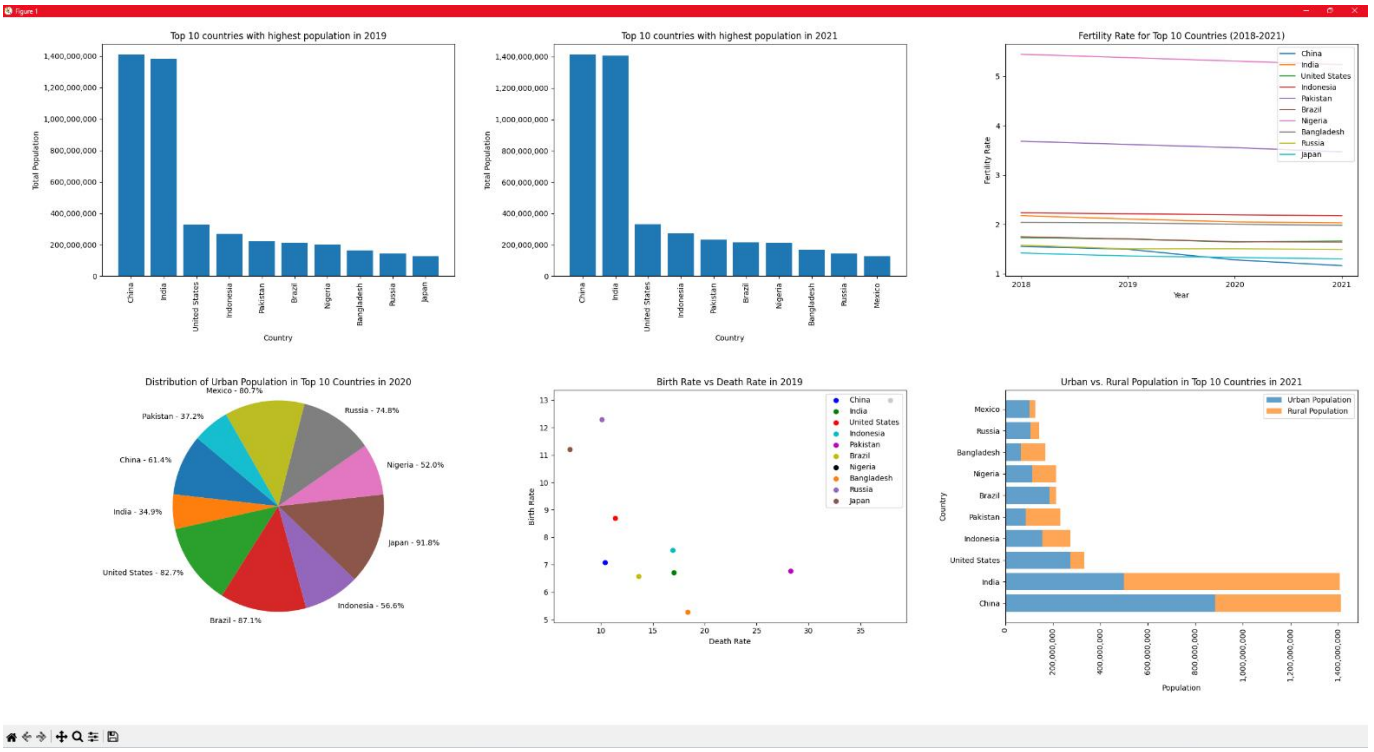
```
# Vytvoření stacked grafu a nastavení jeho os a popisu. Alpha nastavuje průhlednost sloupců v grafu
plt.barh(top_10_data['Country'], top_10_data['Urban Population'],label='Urban Population', alpha=0.7)
plt.barh(top_10_data['Country'], top_10_data['Rural Population'],label='Rural Population', alpha=0.7, left=top_10_data['Urban Population'])
```



Nakonec zavolám jednotlivé funkce, kterým předám v parametru daný rok a také hodnoty pro umístění jednotlivých grafů. Poté grafy zobrazím pomocí metody **show().**

```
barChart(2019, 2, 3, 1)
barChart(2021, 2, 3, 2)
lineChart(2,3,3)
pieChart(2020, 2, 3, 4)
scatter(2019, 'China', 2, 3, 5)
stacked(2021, 2, 3, 6)

#Nastavení layoutu, aby nedošlo k překrytí podgrafů
plt.tight_layout()
plt.show()
```

Výsledek a rozložení jednotlivých grafů.



Závěr

Vizualizace dat pomocí grafů je skvělá pomůcka pro zobrazení dat všeho druhu. Slouží hlavně k prezentaci dat při různých událostech. Ať už pro prezentaci dat, které se mohou týkat financí a obchodu nebo pro prezentaci právě demografických a vědeckých dat veřejnosti.
V tomto mém příkladu si můžeme všimnout například rozdílů růstu populace před a po Covidu-19. Také si můžeme všimnout velkých rozdílů v podílu městské populace mezi jednotlivými státy.