```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib as plt
import warnings
warnings.filterwarnings('ignore')
%matplotlib inline
```

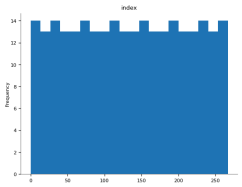
```
import pandas as pd

# Try specifying a different delimiter or using the 'sep' argument
# to automatically detect the delimiter
try:
    df = pd.read_csv('/content/drive/MyDrive/API_SP.POP.TOTL_DS2_en_csv_v2_56/API_SP.POP.TOTL_DS2_en_csv_v2_56.csv', sep=None, engine='py
except pd.errors.ParserError:
    # If automatic detection fails, try common delimiters like tab or semicolon
    try:
        df = pd.read_csv('/content/drive/MyDrive/API_SP.POP.TOTL_DS2_en_csv_v2_56/API_SP.POP.TOTL_DS2_en_csv_v2_56.csv', sep='\t')  # Try
    except pd.errors.ParserError:
        df = pd.read_csv('/content/drive/MyDrive/API_SP.POP.TOTL_DS2_en_csv_v2_56/API_SP.POP.TOTL_DS2_en_csv_v2_56.csv', sep=';')  # Try
df
```

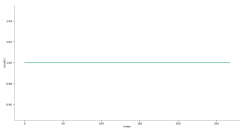
	Data Source,"World Development Indicators",
0	Last Updated Date,"2024-11-13",
1	Country Name,"Country Code","Indicator Name","...
2	Aruba,"ABW","Population, total","SP.POP.TOTL",...
3	Africa Eastern and Southern,"AFE","Population,...
4	Afghanistan,"AFG","Population, total","SP.POP....
...	...
263	Kosovo,"XKX","Population, total","SP.POP.TOTL"...
264	Yemen, Rep.,"YEM","Population, total","SP.POP....
265	South Africa,"ZAF","Population, total","SP.POP...
266	Zambia,"ZMB","Population, total","SP.POP.TOTL"...
267	Zimbabwe,"ZWE","Population, total","SP.POP.TOT...

268 rows × 1 columns

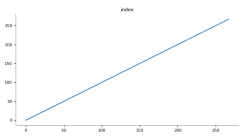
Distributions



Time series



Values

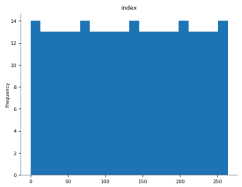


```
df_1 = pd.read_csv('/content/drive/MyDrive/API_SP.POP.TOTL_DS2_en_csv_v2_56/Metadata_Country_API_SP.POP.TOTL_DS2_en_csv_v2_56.csv')
df_1
```

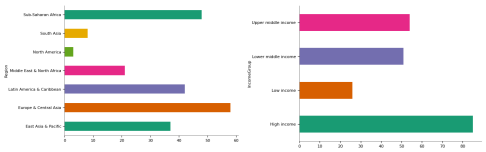
	Country Code	Region	IncomeGroup	SpecialNotes	TableName	Unnamed: 5
0	ABW	Latin America & Caribbean	High income	NaN	Aruba	NaN
1	AFE	NaN	NaN	26 countries, stretching from the Red Sea in t...	Africa Eastern and Southern	NaN
2	AFG	South Asia	Low income	The reporting period for national accounts dat...	Afghanistan	NaN
3	AFW	NaN	NaN	22 countries, stretching from the westernmost ...	Africa Western and Central	NaN
4	AGO	Sub-Saharan Africa	Lower middle income	The World Bank systematically assesses the app...	Angola	NaN
...	...	...	...	...	...	...
260	XKX	Europe & Central Asia	Upper middle income	NaN	Kosovo	NaN
261	YEM	Middle East & North Africa	Low income	The World Bank systematically assesses the app...	Yemen, Rep.	NaN
262	ZAF	Sub-Saharan Africa	Upper middle income	Fiscal year end: March 31; reporting period fo...	South Africa	NaN
263	ZMB	Sub-Saharan Africa	Lower middle income	National accounts data were rebased to reflect...	Zambia	NaN
264	ZWE	Sub-Saharan Africa	Lower middle income	National Accounts data are reported in Zimbabw...	Zimbabwe	NaN

265 rows × 6 columns

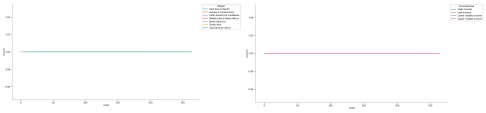
Distributions



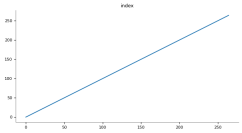
Categorical distributions



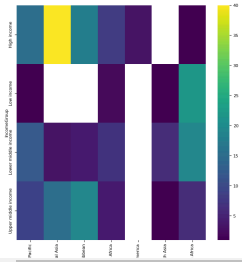
Time series



Values



2-d categorical distributions



```
df_2 = pd.read_csv('/content/drive/MyDrive/API_SP.POP.TOTL_DS2_en_csv_v2_56/Metadata_Indicator_API_SP.POP.TOTL_DS2_en_csv_v2_56.csv')
df_2
```

	INDICATOR_CODE	INDICATOR_NAME	SOURCE_NOTE	SOURCE_ORGANIZATION	Unnamed: 4
0	SP.POP.TOTL	Population, total	Total population is based on the de facto defi...	(1) United Nations Population Division. World ...	NaN

```
df.shape
```

```
(268, 1)
```

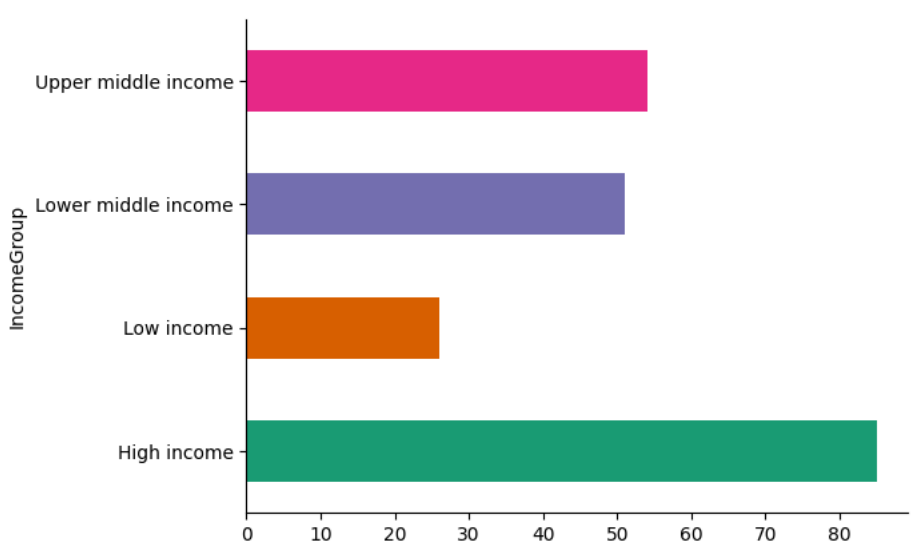
```
df_1.shape
```

```
(265, 6)
```

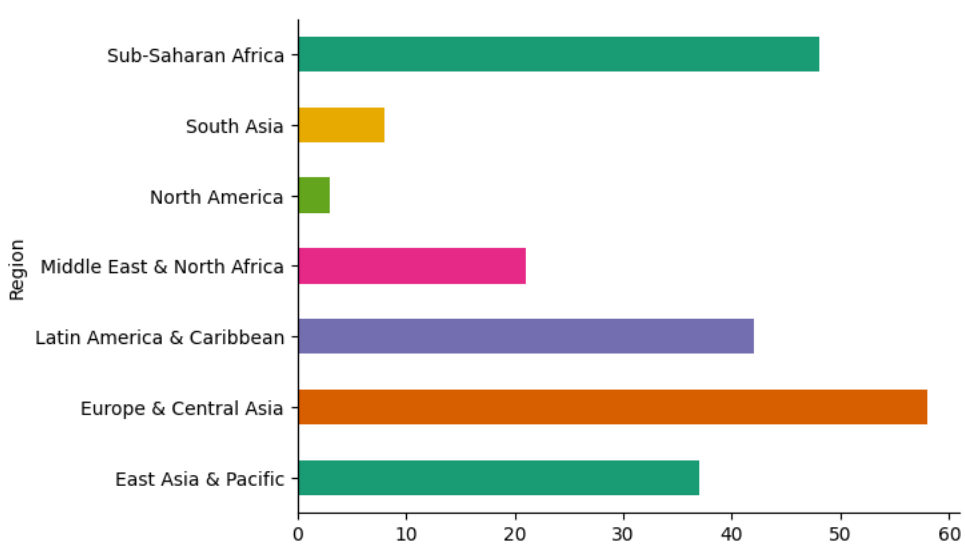
```
df_2.shape
```

```
(1, 5)
```

```
from matplotlib import pyplot as plt
import seaborn as sns
_df_2.groupby('IncomeGroup').size().plot(kind='barh', color=sns.palettes.mpl_palette('Dark2'))
plt.gca().spines[['top', 'right',]].set_visible(False)
```

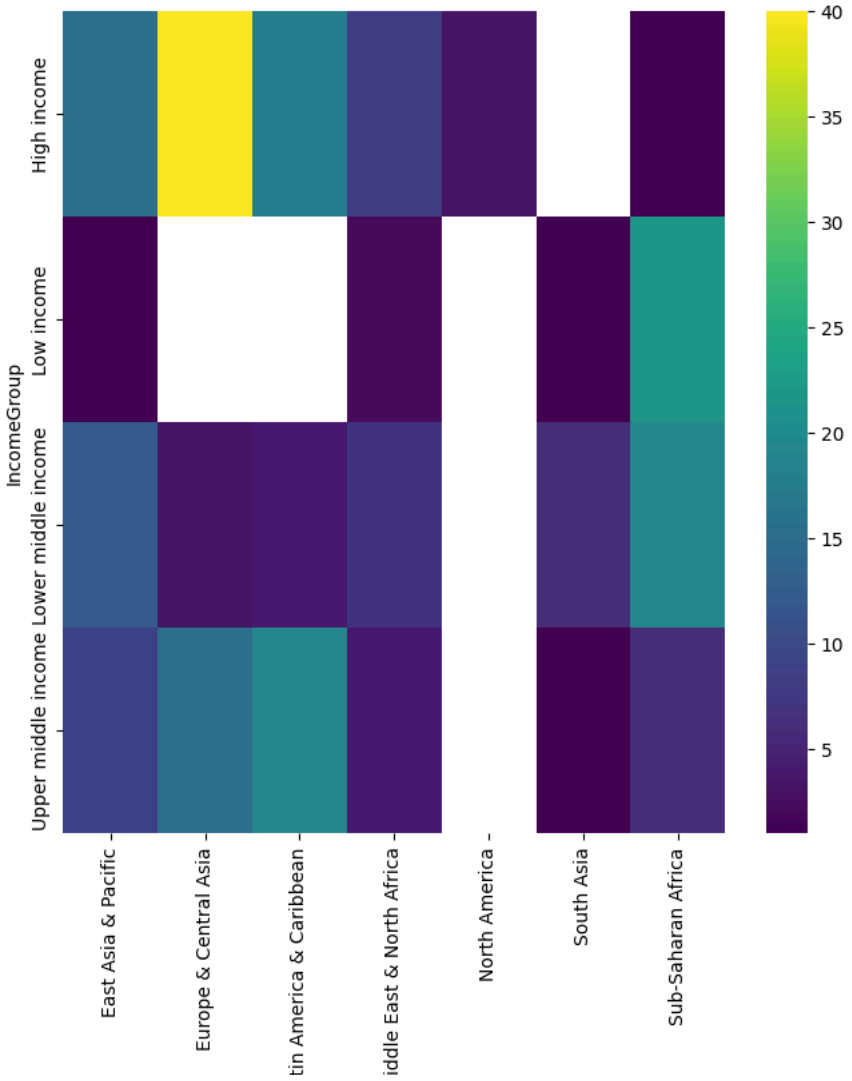



```
from matplotlib import pyplot as plt
import seaborn as sns
_df_1.groupby('Region').size().plot(kind='barh', color=sns.palettes.mpl_palette('Dark2'))
plt.gca().spines[['top', 'right',]].set_visible(False)
```




```
from matplotlib import pyplot as plt
import seaborn as sns
import pandas as pd
plt.subplots(figsize=(8, 8))
df_2dhist = pd.DataFrame({
    x_label: grp['IncomeGroup'].value_counts()
    for x_label, grp in _df_6.groupby('Region')
})
sns.heatmap(df_2dhist, cmap='viridis')
plt.xlabel('Region')
_ = plt.ylabel('IncomeGroup')
```

[+]


IncomeGroup
High income
Low income
Lower middle income
Upper middle income
East Asia & Pacific
Europe & Central Asia
tin America & Caribbean
iddle East & North Africa
North America
South Asia
Sub-Saharan Africa
40
35
30
25
20
15
10
5