Підготовка даних до аналізу

Ознайомитись з методикою первинної обробки даних. Після завершення цієї лабораторної роботи ви зможете:

- Досліджувати структуру завантажених даних
- Виправляти формати даних
- Знаходити та заповнювати пропуски в даних
- Знаходити викиди та некоректні значення
- Будувати прості візуалізації

Завдання, що оцінюються

1. Скачати дані із файлу 'Data2.csv'. Записати дані у dataframe. Дослідити структуру даних.
2. Виправити помилки в даних.
3. Заповнити пропуски.
4. Додати стовпчик із щільністю населення.
5. Побудувати діаграми розмаху та гістограми.

Завдання #1:

Зчитую дані з файлу у датафрейм

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
import pandas as pd
import numpy as np

df = pd.read_csv("Data2.csv", sep=';', encoding='cp1252')
print(df)
```

```
               Country Name                         Region GDP per capita
\
0               Afghanistan                   South Asia    561,7787463

1                   Albania       Europe & Central Asia     4124,98239

2                   Algeria  Middle East & North Africa    3916,881571

3            American Samoa         East Asia & Pacific    11834,74523

4                   Andorra       Europe & Central Asia    36988,62203
```

```
..                     ...                           ...             ...

212  Virgin Islands (U.S.)   Latin America & Caribbean             NaN

213     West Bank and Gaza  Middle East & North Africa    2943,404534

214            Yemen, Rep.  Middle East & North Africa     990,334774

215                 Zambia          Sub-Saharan Africa    1269,573537

216               Zimbabwe          Sub-Saharan Africa    1029,076649


     Populatiion CO2 emission     Area
0     34656032.0     9809,225   652860
1      2876101.0     5716,853    28750
2     40606052.0   145400,217  2381740
3        55599.0          NaN      200
4        77281.0      462,042      470
..           ...          ...      ...
212     102951.0          NaN      350
213    4551566.0          NaN     6020
214   27584213.0     22698,73   527970
215   16591390.0     4503,076   752610
216   16150362.0    12020,426   390760

[217 rows x 6 columns]
```

Досліджую структуру даних

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
df.info()
print(df.describe(include = "all"))
df

print("\n\nТипи даних:\n",df.dtypes)

print("\n\nКількість NaN:\n", df.isna().sum())

print('\n\nУнікальні записи у полі Region:\n', df['Region'].unique())
#тут все ок

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 217 entries, 0 to 216
Data columns (total 6 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Country Name    217 non-null    object
 1   Region          217 non-null    object
```

```
 2   GDP per capita  190 non-null    object
 3   Populatiion     216 non-null    float64
 4   CO2 emission    205 non-null    object
 5   Area            217 non-null    object
dtypes: float64(1), object(5)
memory usage: 10.3+ KB
       Country Name                  Region GDP per capita
Populatiion  \
count           217                     217            190
2.160000e+02
unique          217                       7            190
NaN
top     Afghanistan  Europe & Central Asia    561,7787463
NaN
freq              1                      58              1
NaN
mean            NaN                     NaN            NaN
3.432256e+07
std             NaN                     NaN            NaN
1.347600e+08
min             NaN                     NaN            NaN
1.109700e+04
25%             NaN                     NaN            NaN
7.900265e+05
50%             NaN                     NaN            NaN
6.221590e+06
75%             NaN                     NaN            NaN
2.350337e+07
max             NaN                     NaN            NaN
1.378665e+09

       CO2 emission Area
count           205  217
unique          202  213
top        6318,241  460
freq              2    3
mean            NaN  NaN
std             NaN  NaN
min             NaN  NaN
25%             NaN  NaN
50%             NaN  NaN
75%             NaN  NaN
max             NaN  NaN


Типи даних:
 Country Name       object
Region             object
GDP per capita     object
Populatiion       float64
```

```
CO2 emission       object
Area               object
dtype: object


Кількість NaN:
 Country Name       0
Region              0
GDP per capita     27
Populatiion         1
CO2 emission       12
Area                0
dtype: int64


Унікальні записи у полі Region:
 ['South Asia' 'Europe & Central Asia' 'Middle East & North Africa'
 'East Asia & Pacific' 'Sub-Saharan Africa' 'Latin America &
Caribbean'
 'North America']
```

Бачу наступні проблеми в даних:

1. У назві одного з полів є typo: 'Populatiion'
2. Невідповідність значень типам даних (у полів 'GDP per capita', 'CO2 emission' та 'Area' тип object, хоча має бути float, у поля 'Population' тип даних float, хоча кількість населення не може бути дробовим числом).
3. Ця проблема буде виявлена після вирішення перших двух.
4. Є пропущені значення в ознаках 'GDP per capita', 'CO2 emission', та 'Populatiion'

Завдання #2:

Проблема 1. Для виправлення зроблю наступне: зміню назву стовпця

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
df.rename(columns={'Populatiion':'Population'}, inplace=True)
print(df)
```

```
              Country Name                         Region GDP per capita
\
0              Afghanistan                     South Asia    561,7787463

1                  Albania       Europe & Central Asia     4124,98239

2                  Algeria  Middle East & North Africa    3916,881571

3           American Samoa         East Asia & Pacific    11834,74523
```

```
4                  Andorra       Europe & Central Asia    36988,62203

..                     ...                         ...            ...

212  Virgin Islands (U.S.)   Latin America & Caribbean            NaN

213     West Bank and Gaza  Middle East & North Africa    2943,404534

214            Yemen, Rep.  Middle East & North Africa     990,334774

215                 Zambia          Sub-Saharan Africa    1269,573537

216               Zimbabwe          Sub-Saharan Africa    1029,076649


     Population CO2 emission     Area
0    34656032.0     9809,225   652860
1     2876101.0     5716,853    28750
2    40606052.0   145400,217  2381740
3       55599.0          NaN      200
4       77281.0      462,042      470
..          ...          ...      ...
212    102951.0          NaN      350
213   4551566.0          NaN     6020
214  27584213.0     22698,73   527970
215  16591390.0     4503,076   752610
216  16150362.0    12020,426   390760

[217 rows x 6 columns]
```

Проблема 2. Для виправлення зроблю наступне (опишіть, що хочете зробити)

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
df['GDP per capita'] = df['GDP per capita'].str.replace(',', '.')
df[['GDP per capita']] = df[['GDP per capita']].astype('float')

df['CO2 emission'] = df['CO2 emission'].str.replace(',', '.')
df[['CO2 emission']] = df[['CO2 emission']].astype('float')

df = df[df['Area'].notna()]
df['Area'] = df['Area'].str.replace(',', '.')
df[['Area']] = df[['Area']].astype('float')

df = df[df['Population'].notna()]
df[['Population']] = df[['Population']].astype('int')

print("\n\nТипи даних:\n",df.dtypes)
```

```
Типи даних:
```

```
 Country Name       object
Region             object
GDP per capita    float64
Population          int64
CO2 emission      float64
Area              float64
dtype: object
```

Проблема 3. Для виявлення перевіримо чи всі дані є допутимими

```
print(df[(df['Population'] <= 0) | (df['Area'] <= 0) | (df['CO2
emission'] <= 0) | (df['GDP per capita'] <= 0)])
```

```
           Country Name                     Region  GDP per capita  \
56   Dominican Republic  Latin America & Caribbean    -6722.223536
135             Myanmar        East Asia & Pacific     1195.515372

     Population  CO2 emission      Area
56     10648791     21539.958   48670.0
135    52885223     21631.633 -676590.0
```

Проблема 3 полягає у тому, що значення 56 запису в полі 'GDP per capita' і 135 запису в полі 'Area' менше 0. Тобто дані невалідні.

Для вирішення цієї проблеми замінимо невалідні дані на NaN, а що із ними робити вирішемо в завданні #3.

```
df.at[135, 'Area'] = np.nan
df.at[56, 'GDP per capita'] = np.nan

print(df.loc[56], '\n\n', df.loc[135])
```

```
Country Name              Dominican Republic
Region            Latin America & Caribbean
GDP per capita                          NaN
Population                         10648791
CO2 emission                      21539.958
Area                                48670.0
Name: 56, dtype: object

 Country Name                    Myanmar
Region            East Asia & Pacific
GDP per capita            1195.515372
Population                   52885223
CO2 emission                21631.633
Area                              NaN
Name: 135, dtype: object
```

Завдання #3:

У ознаки Population і Area відсутнє лише одне значення

```
print(df[df['Population'].isna()])
```

```
   Country Name              Region GDP per capita  Population CO2
emission  \
61      Eritrea  Sub-Saharan Africa            NaN         NaN
696,73

      Area
61  117600
Empty DataFrame
Columns: [Country Name, Region, GDP per capita, Population, CO2
emission, Area]
Index: []
```

```
print(df[df['Area'].isna()])
```

```
    Country Name               Region  GDP per capita  Population  \
135      Myanmar  East Asia & Pacific     1195.515372    52885223

     CO2 emission  Area
135     21631.633   NaN
```

У запису під номером 61 також відсутню значення ознаки GDP per capita. Знайдемо пропущені поля цих країн в інтернеті.

Еритрея

Населення: 3.684 млн = 3684000

ВВП на душу населення: 643,79

М'янма

Площа: 676578

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
df.at[61, 'Population'] = 3684000
df.at[61, 'GDP per capita'] = 643.79

df.at[135, 'Area'] = 676578

print(df.loc[61],'\n\n', df.loc[135])
```

```
Country Name                 Eritrea
Region           Sub-Saharan Africa
GDP per capita               643.79
Population                  3684000
```

```
CO2 emission                   696.73
Area                         117600.0
Name: 61, dtype: object

 Country Name                    Myanmar
Region              East Asia & Pacific
GDP per capita              1195.515372
Population                     52885223
CO2 emission                  21631.633
Area                           676578.0
Name: 135, dtype: object
```

Обчислимо середнє арифметичне мат. сподівання відношення площі до викидів CO2 і кількості населення до викидів CO2. За цією величиною визначимо кількість викидів у пропущених даних.

```
for_co2 = ((df['Area']/df['CO2 emission']).mean() +
(df['Population']/df['CO2 emission']).mean())/2

print(for_co2)

897.0671670946242
```

Маємо формулу для знаходження викидів вуглекислого газу

CO2 emission = (Area+Population)/(2*for_co2)

```
df['CO2 emission'] = df['CO2 emission'].fillna((df['Area'] +
df['Population']) / (2 * for_co2))

print("\n\nКількість NaN:\n", df.isna().sum())



Кількість NaN:
 Country Name       0
Region              0
GDP per capita     27
Population          0
CO2 emission        0
Area                0
dtype: int64
```

Пропущені дані для GDP визначимо таким самим чином

```
for_gdp = ((df['Area']/df['GDP per capita']).mean() +
(df['Population']/df['GDP per capita']).mean())/2

df['GDP per capita'] = df['GDP per capita'].fillna((df['Area'] +
df['Population']) / (2 * for_gdp))
```

Досліджую структуру даних, чи всі пропуски заповнено

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
print("\n\nКількість NaN:\n", df.isna().sum())
```

```
Кількість NaN:
 Country Name      0
Region             0
GDP per capita     0
Population         0
CO2 emission       0
Area               0
dtype: int64
```

Завдання #4:

Щільність населення розрахую по формулі Population/Area і додам у стовпчик Density.

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
df['Density'] = df['Population']/df['Area']
print(df)
```

```
              Country Name                       Region  GDP per capita
\
0              Afghanistan                   South Asia      561.778746

1                  Albania       Europe & Central Asia     4124.982390

2                  Algeria  Middle East & North Africa     3916.881571

3           American Samoa         East Asia & Pacific    11834.745230

4                  Andorra       Europe & Central Asia    36988.622030

..                     ...                          ...             ...

212  Virgin Islands (U.S.)   Latin America & Caribbean        6.327732

213     West Bank and Gaza  Middle East & North Africa     2943.404534

214            Yemen, Rep.  Middle East & North Africa      990.334774

215                 Zambia          Sub-Saharan Africa     1269.573537

216               Zimbabwe          Sub-Saharan Africa     1029.076649
```

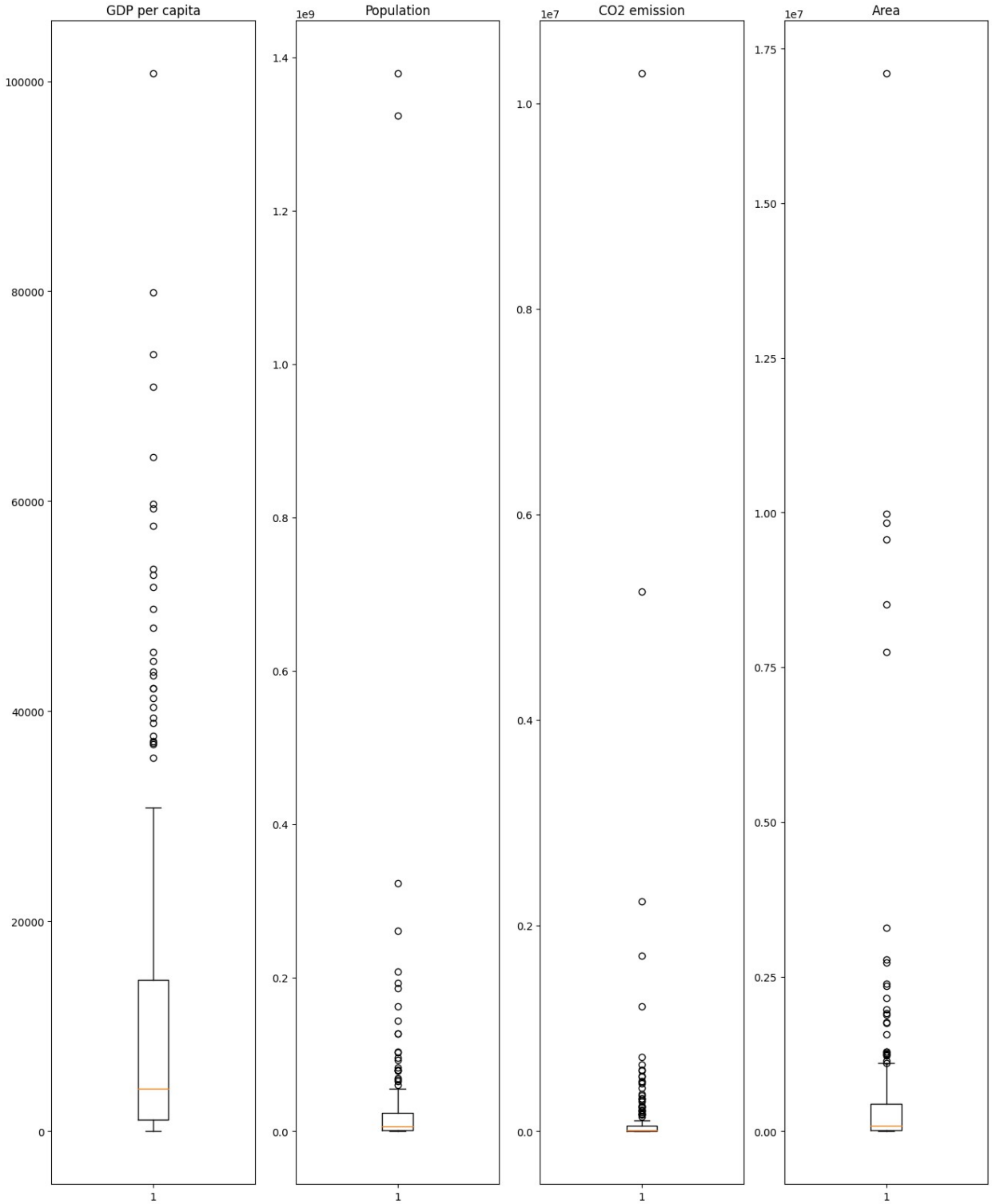
```
     Population   CO2 emission       Area     Density
0      34656032    9809.225000   652860.0   53.083405
1       2876101    5716.853000    28750.0  100.038296
2      40606052  145400.217000  2381740.0   17.048902
3         55599      31.100793      200.0  277.995000
4         77281     462.042000      470.0  164.427660
..          ...            ...        ...         ...
212      102951      57.577071      350.0  294.145714
213     4551566    2540.270209     6020.0  756.074086
214    27584213   22698.730000   527970.0   52.245796
215    16591390    4503.076000   752610.0   22.045136
216    16150362   12020.426000   390760.0   41.330643

[217 rows x 7 columns]
```

Завдання #5:

Для побудови графіків скористайтесь бібліотекою Matplotlib. Спробуйте погратись з кольорами, розмірами та підписами.

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
import matplotlib.pyplot as plt

fig, axs = plt.subplots(1, 4, figsize=(16, 20))

fig.suptitle('Діаграми розмаху', fontsize=16)

axs[0].set_title('GDP per capita')
axs[0].boxplot(df['GDP per capita'])

axs[1].set_title('Population')
axs[1].boxplot(df['Population'])

axs[2].set_title('CO2 emission')
axs[2].boxplot(df['CO2 emission'])

axs[3].set_title('Area')
axs[3].boxplot(df['Area'])

plt.show()
```

Діаграми розмаху



Додаткове завдання:

1. Яка країна має найбільший ВВП на людину (GDP per capita)?
2. Яка країна має найменшу площу?
3. Знайдіть країну з найбільшою щільністю населення у світі? У Європі та центральній Азії?
4. Покажіть топ 5 країн та 5 останніх країн по ВВП на людину.

```
# 1
print('\n' + df.loc[df['GDP per capita'].idxmax(), 'Country Name'] + ' має найбільший ВВП на людину')
```

```
Luxembourg має найбільший ВВП на людину
```

```
#2
print('\n' + df.loc[df['Area'].idxmin(), 'Country Name'] + ' має найменшу площу')
```

```
Monaco має найменшу площу
```

```
#3
print('\n' + df.loc[df['Density'].idxmax(), 'Country Name'] + ' має найбільшу щільність населення у світі')
print('\n' + df[df['Region'] == 'Europe & Central Asia'].loc[df[df['Region'] == 'Europe & Central Asia']['Density'].idxmax(), 'Country Name'] + ' має найбільшу щільність населення у Європі та центральній Азії')
```

```
Macao SAR, China має найбільшу щільність населення у світі

Monaco має найбільшу щільність населення у Європі та центральній Азії
```

Збережіть дані у новий файл 'clean_data2.csv':

```
# Напишіть ваш код нижче та натисніть Shift+Enter для виконання
df.to_csv('clean_data2.csv', index=False)
```