МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Практикум №3

з курсу «Аналіз даних в інформаційнних системах»

на тему: «Описова статистика»

Викладач:
Ліхоузова Т.А.

Виконав:
студент 2 курсу
групи ІП-11 Сідак
Кирил з ФІОТ

Київ-2023

ЗМІСТ

1. ЗАВДАННЯ

Основне завдання

Скачати дані із файлу Data2.csv

1. Записати дані у data frame
2. Дослідити структуру даних
3. Виправити помилки в даних
4. Побудувати діаграми розмаху та гістограми
5. Додати стовпчик із щільністю населення

Додаткове завдання

Відповісти на питання (файл Data2.csv):

1. Чи є пропущені значення? Якщо є, замінити середніми
2. Яка країна має найбільший ВВП на людину (GDP per capita)? Яка має найменшу площу?
3. В якому регіоні середня площа країни найбільша?
4. Знайдіть країну з найбільшою щільністю населення у світі? У Європі та центральній Азії?
5. Чи співпадає в якомусь регіоні середнє та медіана ВВП?
6. Вивести топ 5 країн та 5 останніх країн по ВВП та кількості CO2 на душу населення.

2. ОСНОВНЕ ЗАВДАННЯ

1. Записати дані у data frame

1. Reading data from csv into DataFrame

```
df: pd.DataFrame = pd.read_csv('Lab_3/data/Data2.csv', sep=';', encoding="ISO-8859-1")
df.head()
```

5 rows × 6 columns

	Country Name	Region	GDP per capita	Populatiion	CO2 emission	Area
0	Afghanistan	South Asia	561,7787463	34656032.0	9809,225	652860
1	Albania	Europe & Central Asia	4124,98239	2876101.0	5716,853	28750
2	Algeria	Middle East & North Africa	3916,881571	40606052.0	145400,217	2381740
3	American Samoa	East Asia & Pacific	11834,74523	55599.0	NaN	200
4	Andorra	Europe & Central Asia	36988,62203	77281.0	462,042	470

```
df.tail()
```

5 rows × 6 columns

	Country Name	Region	GDP per capita	Populatiion	CO2 emission	Are
212	Virgin Islands (U.S.)	Latin America & Caribbean	NaN	102951.0	NaN	
213	West Bank and Gaza	Middle East & North Africa	2943,404534	4551566.0	NaN	6
214	Yemen, Rep.	Middle East & North Africa	990,334774	27584213.0	22698,73	527
215	Zambia	Sub-Saharan Africa	1269,573537	16591390.0	4503,076	752
216	Zimbabwe	Sub-Saharan Africa	1029,076649	16150362.0	12020,426	390

2. Дослідити структуру даних

У даному датасеті присутні пропущені значення (NaN), дублікати відсутні, наявні від’ємні значення, типи даних стовпців з числовими значеннями не є числовими, замість крапки використана кома для розділення дробової частини числа, назви країн мають зайві частини.

3. Виправити помилки в даних

2. Data cleaning

2.1 Renaming columns

```
df = df.rename(columns={'Populatiion': 'Population'})
df.head()
```

5 rows × 6 columns

	Country Name	Region	GDP per capita	Population	CO2 emission	Area
0	Afghanistan	South Asia	561,7787463	34656032.0	9809,225	652860
1	Albania	Europe & Central Asia	4124,98239	2876101.0	5716,853	28750
2	Algeria	Middle East & North Africa	3916,881571	40606052.0	145400,217	2381740
3	American Samoa	East Asia & Pacific	11834,74523	55599.0	NaN	200
4	Andorra	Europe & Central Asia	36988,62203	77281.0	462,042	470

2.2 Checking for duplicates

```
df.duplicated().sum()
```

```
0
```

2.3 Formatting columns

```
for column in ['GDP per capita', 'CO2 emission', 'Area']:
    df[column] = df[column].str.replace(',', '.')
    df[column] = df[column].str.replace('-', '')
    df[column] = df[column].astype(float)
df.head()
```

5 rows | 5 rows × 6 columns | CSV

	Country Name	Region	GDP per capita	Population	CO2 emission	Area
0	Afghanistan	South Asia	561.778746	34656032.0	9809.225	652860.0
1	Albania	Europe & Central Asia	4124.982390	2876101.0	5716.853	28750.0
2	Algeria	Middle East & North Africa	3916.881571	40606052.0	145400.217	2381740.0
3	American Samoa	East Asia & Pacific	11834.745230	55599.0	NaN	200.0
4	Andorra	Europe & Central Asia	36988.622030	77281.0	462.042	470.0

```
df[df['Country Name'].str.contains('Congo')]
```

2 rows | 2 rows × 6 columns | CSV

	Country Name	Region	GDP per capita	Population	CO2 emission	Area
44	Congo, Dem. Rep.	Sub-Saharan Africa	405.542501	78736153.0	4671.758	2344860.0
45	Congo, Rep.	Sub-Saharan Africa	1528.244720	5125821.0	3094.948	342000.0

```
country_mapping = {'Congo, Dem. Rep.': 'Democratic Republic of the Congo', 'Congo, Rep.': 'Republic of the
  Congo'}
df['Country Name'] = df['Country Name'].replace(country_mapping)
df[df['Country Name'].str.contains('Congo')]
```

2 rows | 2 rows × 6 columns | CSV

	Country Name	Region	GDP per capita	Population	CO2 emission	
44	Democratic Republic of the Congo	Sub-Saharan Africa	405.542501	78736153.0	4671.758	23
45	Republic of the Congo	Sub-Saharan Africa	1528.244720	5125821.0	3094.948	3

```
df['Country Name'] = df['Country Name'].str.extract(r'([a-zA-Z\s\.]+)')
df.head()
```

5 rows | 5 rows × 6 columns | CSV

	Country Name	Region	GDP per capita	Population	CO2 emission	Area
0	Afghanistan	South Asia	561.778746	34656032.0	9809.225000	652860.0
1	Albania	Europe & Central Asia	4124.982390	2876101.0	5716.853000	28750.0
2	Algeria	Middle East & North Africa	3916.881571	40606052.0	145400.217000	2381740.0
3	American Samoa	East Asia & Pacific	11834.745230	55599.0	165114.116337	200.0
4	Andorra	Europe & Central Asia	36988.622030	77281.0	462.042000	470.0

2.4 Mean imputation (additional task)

```
df.isna().sum()
```

6 rows · Length: 6, dtype: int64 · CSV

	data
Country Name	0
Region	0
GDP per capita	27
Population	1
CO2 emission	12
Area	0

```
df.fillna(df.mean(numeric_only=True), inplace=True)
df.head()
```

5 rows · 5 rows × 6 columns · CSV

	Country Name	Region	GDP per capita	Population	CO2 emission	Area
0	Afghanistan	South Asia	561.778746	34656032.0	9809.225000	652860.0
1	Albania	Europe & Central Asia	4124.982390	2876101.0	5716.853000	28750.0
2	Algeria	Middle East & North Africa	3916.881571	40606052.0	145400.217000	2381740.0
3	American Samoa	East Asia & Pacific	11834.745230	55599.0	165114.116337	200.0
4	Andorra	Europe & Central Asia	36988.622030	77281.0	462.042000	470.0

```
df['Population'] = df['Population'].astype('int')
df.head()
```

5 rows · 5 rows × 6 columns · CSV

	Country Name	Region	GDP per capita	Population	CO2 emission	Area
0	Afghanistan	South Asia	561.778746	34656032	9809.225000	652860.0
1	Albania	Europe & Central Asia	4124.982390	2876101	5716.853000	28750.0
2	Algeria	Middle East & North Africa	3916.881571	40606052	145400.217000	2381740.0
3	American Samoa	East Asia & Pacific	11834.745230	55599	165114.116337	200.0
4	Andorra	Europe & Central Asia	36988.622030	77281	462.042000	470.0

4. Побудувати діаграми розмаху та гістограми

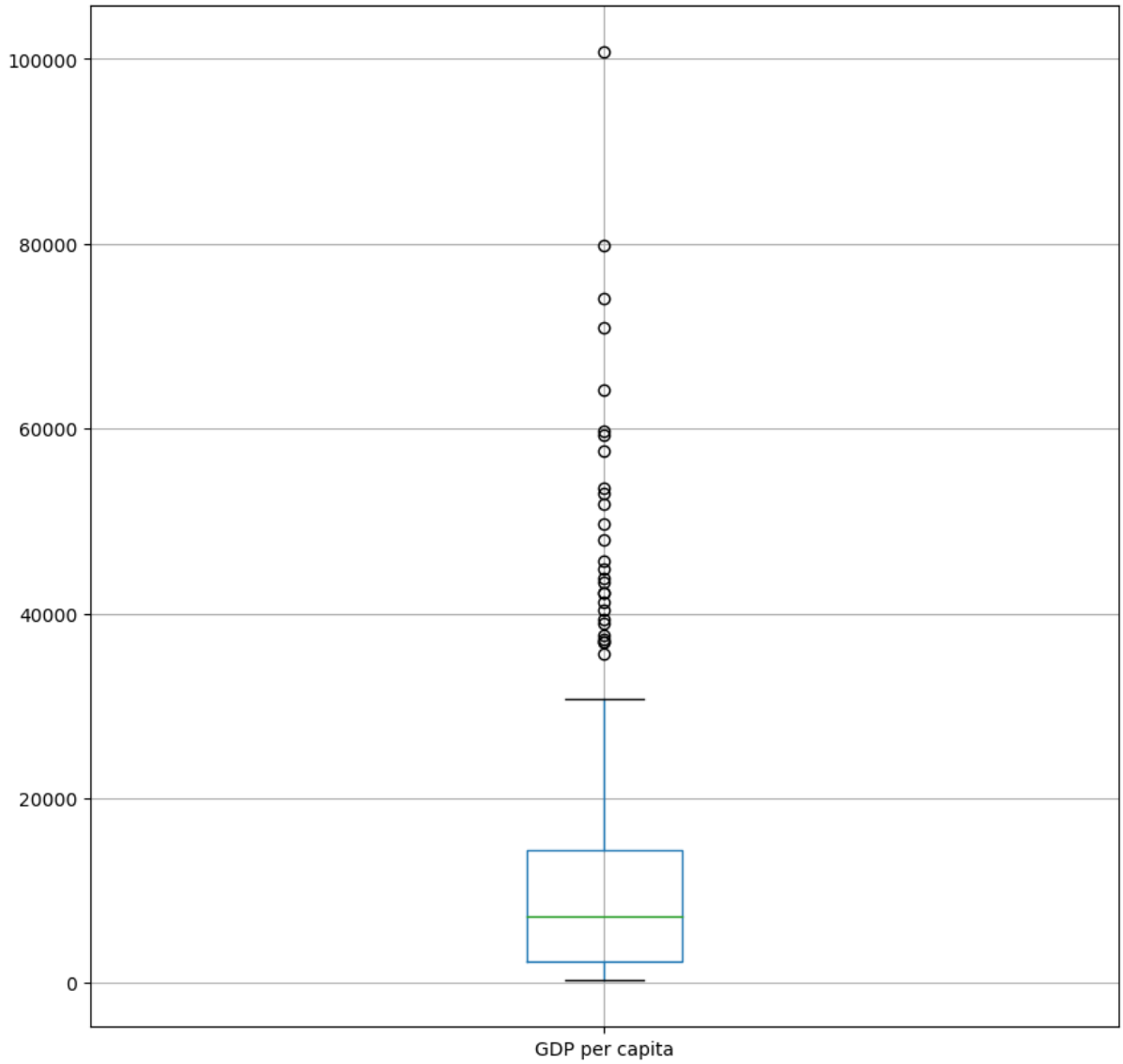


Рисунок 4.1 – Діаграма розмаху для ВВП на душу населення

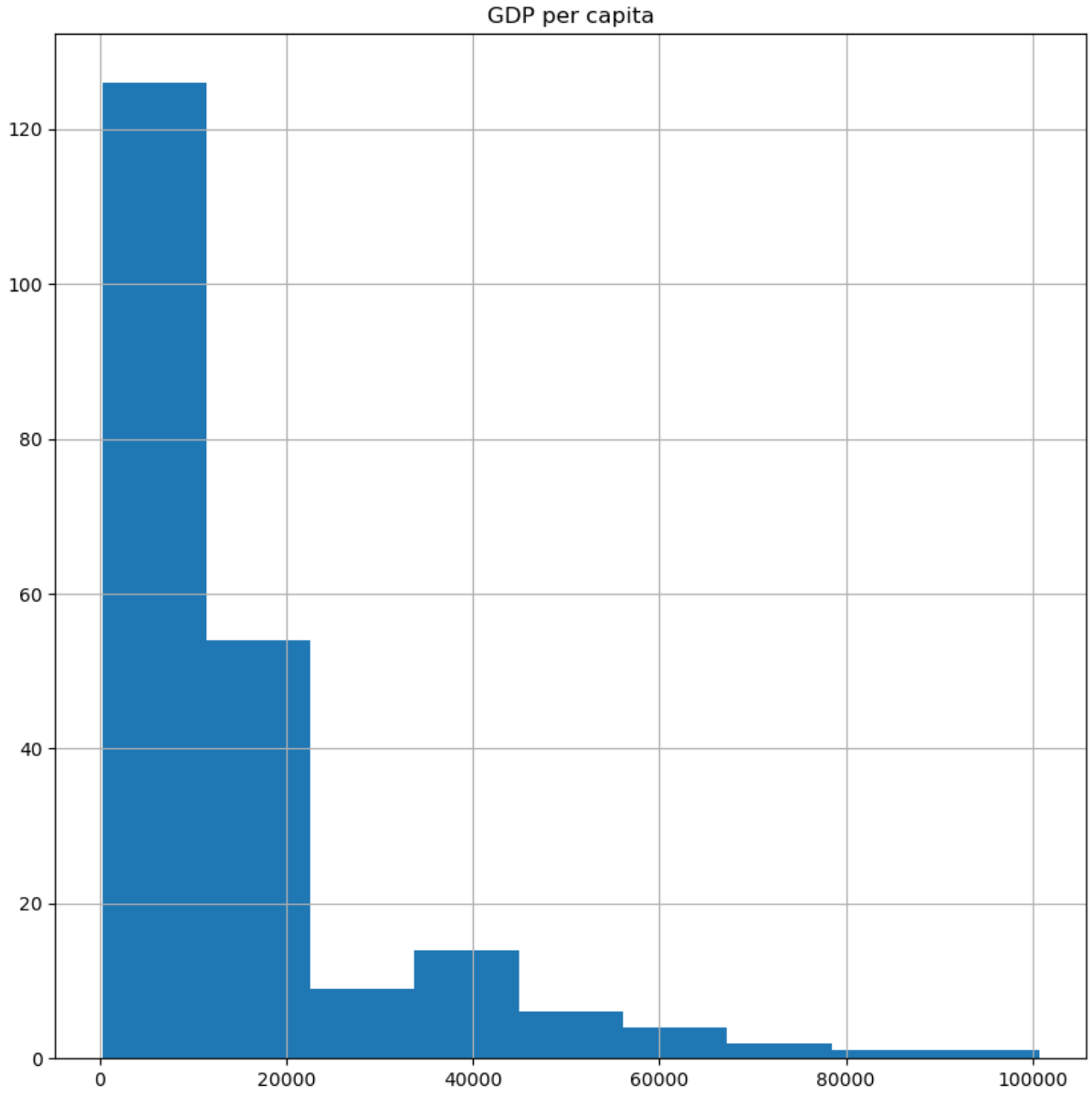


Рисунок 4.2 – Гістограма для ВВП на душу населення

5. Додати стовпчик із щільністю населення

4. Addition of the column “Population Density”

```
df['Population Density'] = df['Population'] / df['Area']
df.head()
```

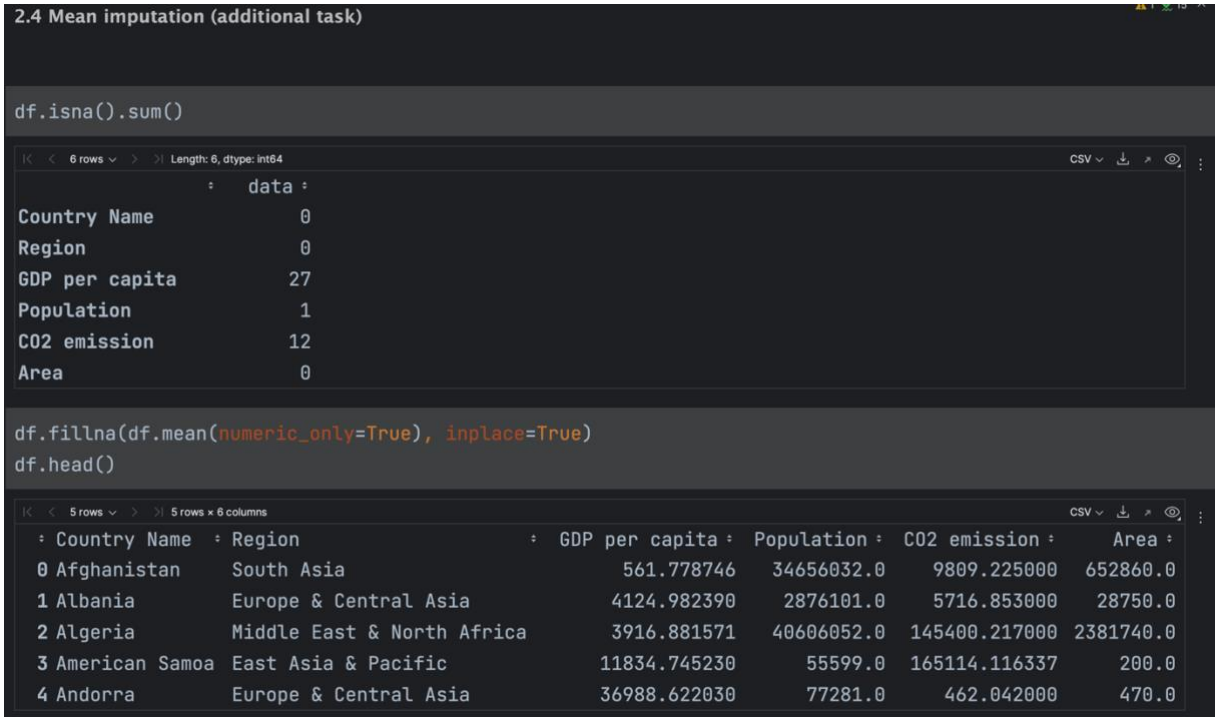
5 rows × 7 columns

	Country ...	Region	GDP per...	Population	CO2 e...	Area	Population Density
0	Afghanistan	South Asia	561.778746	34656032	9809.2250…	652860.0	53.083405
1	Albania	Europe & Central…	4124.982390	2876101	5716.8530…	28750.0	100.038296
2	Algeria	Middle East & No…	3916.881571	40606052	145400.21…	2381740.0	17.048902
3	American Sa…	East Asia & Paci…	11834.745230	55599	165114.11…	200.0	277.995000
4	Andorra	Europe & Central…	36988.622030	77281	462.042000	470.0	164.427660

3. ДОДАТКОВЕ ЗАВДАННЯ

1. Чи є пропущені значення? Якщо є, замінити середніми

Пропущені значення наявні в наступних стовпцях: “GDP per capita”, “Population”, “CO2 emission”.



2.4 Mean imputation (additional task)

```
df.isna().sum()
```

6 rows · Length: 6, dtype: int64 · CSV

	data
Country Name	0
Region	0
GDP per capita	27
Population	1
CO2 emission	12
Area	0

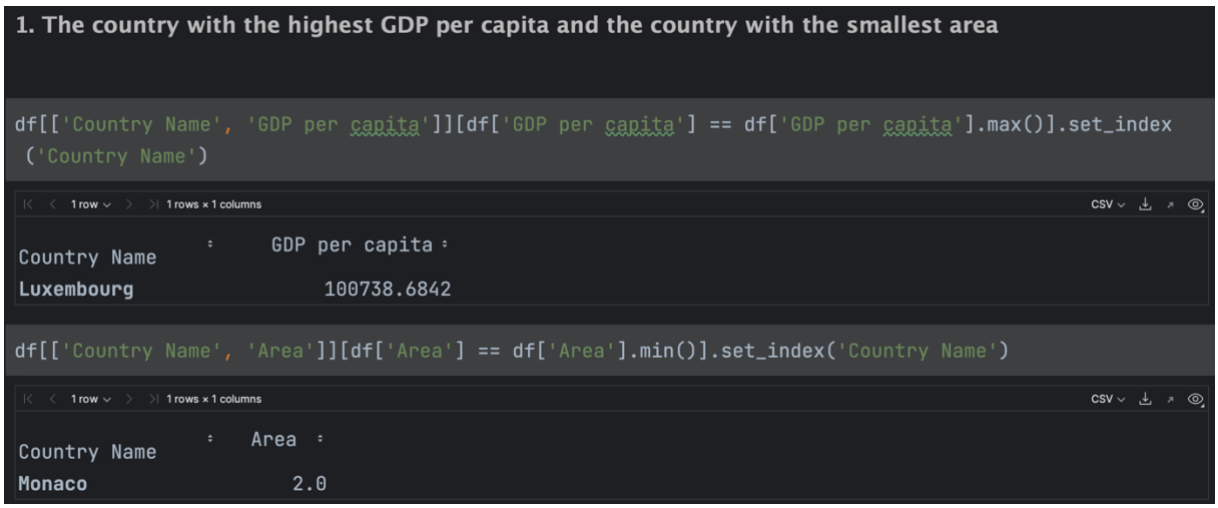
```
df.fillna(df.mean(numeric_only=True), inplace=True)
df.head()
```

5 rows · 5 rows × 6 columns · CSV

	Country Name	Region	GDP per capita	Population	CO2 emission	Area
0	Afghanistan	South Asia	561.778746	34656032.0	9809.225000	652860.0
1	Albania	Europe & Central Asia	4124.982390	2876101.0	5716.853000	28750.0
2	Algeria	Middle East & North Africa	3916.881571	40606052.0	145400.217000	2381740.0
3	American Samoa	East Asia & Pacific	11834.745230	55599.0	165114.116337	200.0
4	Andorra	Europe & Central Asia	36988.622030	77281.0	462.042000	470.0

2. Яка країна має найбільший ВВП на людину (GDP per capita)? Яка має найменшу площу?

Найбільший ВВП на людину має Люксембург, а найменшу площу має Монако.



1. The country with the highest GDP per capita and the country with the smallest area

```
df[['Country Name', 'GDP per capita']][df['GDP per capita'] == df['GDP per capita'].max()].set_index
 ('Country Name')
```

1 row · 1 rows × 1 columns · CSV

Country Name	GDP per capita
Luxembourg	100738.6842

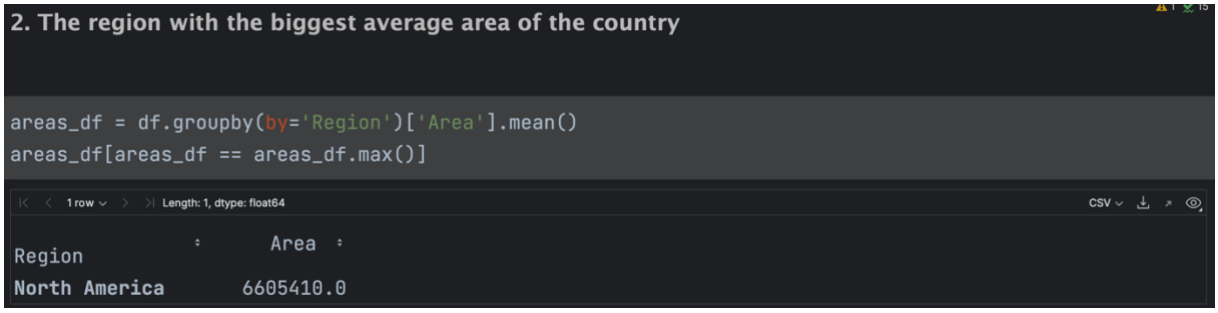
```
df[['Country Name', 'Area']][df['Area'] == df['Area'].min()].set_index('Country Name')
```

1 row · 1 rows × 1 columns · CSV

Country Name	Area
Monaco	2.0

3. В якому регіоні середня площа країни найбільша?

Найбільша середня площа країни в Північній Америці.



4. Знайдіть країну з найбільшою щільністю населення у світі? У Європі та центральній Азії?

Країна з найбільшою щільністю населення у світі – це Макао (особливий адміністративний район (SAR) Китаю), а в Європі та центральній Азії – Монако.

5. Чи співпадає в якомусь регіоні середнє та медіана ВВП?

У жодному регіоні середнє та медіана ВВП не співпадають. У регіоні Латинська Америка та Кариби різниця між цими значеннями є мінімальною і становить трохи більше 347.

4. Check if there is a region with the same GDP mean and GDP median

```
mean_median_df = df.groupby('Region').agg({'GDP per capita': ['mean', 'median']})
mean_median_df.columns = ['GDP mean', 'GDP median']
mean_median_df[mean_median_df['GDP mean'] == mean_median_df['GDP median']]
```

0 rows | 0 rows × 2 columns | CSV

Region	GDP mean	GDP median

```
mean_median_df['Difference'] = abs(mean_median_df['GDP median'] - mean_median_df['GDP mean'])
mean_median_df[mean_median_df['Difference'] == mean_median_df['Difference'].min()]
```

1 row | 1 rows × 3 columns | CSV

Region	GDP mean	GDP median	Difference
Latin America & Caribbean	10485.343136	10833.201075	347.857939

6. Вивести топ 5 країн та 5 останніх країн по ВВП та кількості CO2 на душу населення.

Топ-5 та 5 останніх країн по ВВП:

```
top_gdp_df = df[['Country Name']].assign(
    **{'GDP': df['GDP per capita'] * df['Population']})
top_gdp_df = top_gdp_df.sort_values(by='GDP', ascending=False).set_index('Country Name')['GDP']
top_gdp_df.head()
```

5 rows | Length: 5, dtype: float64 | CSV

Country Name	GDP
United States	1.862448e+13
China	1.119915e+13
Japan	4.940159e+12
Germany	3.485379e+12
United Kingdom	2.649581e+12

```
top_gdp_df.tail()
```

5 rows | Length: 5, dtype: float64 | CSV

Country Name	GDP
Palau	3.102483e+08
Marshall Islands	1.944979e+08
Kiribati	1.815515e+08
Nauru	1.020601e+08
Tuvalu	3.421888e+07

Топ-5 та 5 останніх країн по кількості CO2 на душу населення:

```
top_co2_df = df[['Country Name']].assign(
    **{'CO2 per capita': df['CO2 emission'] / df['Population']}
)
top_co2_df = top_co2_df.sort_values(by='CO2 per capita', ascending=False).set_index('Country Name')
top_co2_df.head()
```

5 rows × 1 columns

Country Name	CO2 per capita
St. Martin	5.168053
San Marino	4.972867
Monaco	4.288790
Northern Mariana Islands	3.000820
American Samoa	2.969732

```
top_co2_df.tail()
```

5 rows × 1 columns

Country Name	CO2 per capita
Democratic Republic of the Congo	0.000059
Chad	0.000050
Somalia	0.000043
Burundi	0.000042
Eritrea	0.000020

4. ВИСНОВОК

Отже, при виконанні даної лабораторної було записано дані у DataFrame, виправлені помилки в цих даних, а саме: пропущені значення (NaN), від’ємні значення, стовпців з числовими значеннями з не числовими типами даних, крапки замість кома для розділення дробової частини числа, невідформатовані назви країн, побудовано діаграму розмаху та гістограму для числового стовпця DataFrame, додано стовпчик із щільністю населення та досліджено країни та регіони з найбільшими чи найменшими значеннями певних параметрів.