

Zadanie 2

```
In [2]: import pandas as pd
import numpy as np

# Wczytaj dane z pliku CSV
df = pd.read_csv("data.csv")

# Zamiana wartości "brak" na 0 w jedenastej kolumnie
df.iloc[:, 3] = pd.to_numeric(df.iloc[:, 10].replace("brak", 0), errors='coerce').fillna(0)
df.iloc[:, 10] = pd.to_numeric(df.iloc[:, 10].replace("brak", 0), errors='coerce').fillna(0)

print(df)
```

	observation_id	submitted_time	gender	age	\
0	wmn_4503683847159808	2020-07-09 23:19:01.982 UTC	Female	3.0	
1	wmn_4503772699295744	2020-07-09 21:22:15.864 UTC	Female	13.0	
2	wmn_4504010469146624	2020-07-10 05:09:07.359 UTC	Female	5.0	
3	wmn_4504035500752896	2020-07-11 16:59:49.85 UTC	Female	7.0	
4	wmn_4504181395423232	2020-07-11 18:43:35.954 UTC	Female	3.0	
...	
12349	wmn_6754210441068544	2020-07-16 15:46:12.095 UTC	Female	7.0	
12350	wmn_6754415891709952	2020-07-10 09:57:24.863 UTC	Female	3.0	
12351	wmn_6754483574145024	2020-07-19 17:50:01.295 UTC	Female	4.0	
12352	wmn_6755256899993600	2020-07-11 16:09:09.78 UTC	Female	4.0	
12353	wmn_6755376524689408	2020-07-17 03:19:00.388 UTC	Female	3.0	

	geography	\
0	City center or metropolitan area	
1	Rural	
2	Rural	
3	Suburban/Peri-urban	
4	Suburban/Peri-urban	
...	...	
12349	Suburban/Peri-urban	
12350	City center or metropolitan area	
12351	City center or metropolitan area	
12352	Rural	
12353	City center or metropolitan area	

	financial_situation	\
0	I cannot afford enough food for my family	
1	I cannot afford enough food for my family	
2	I can afford food, but nothing else	
3	I can afford food and regular expenses, and bu...	
4	I can afford food and regular expenses, and bu...	
...	...	
12349	I can afford food and regular expenses, but no...	
12350	I can afford food and regular expenses, but no...	
12351	I can afford food and regular expenses, but no...	
12352	I can comfortably afford food, clothes, and fu...	
12353	I can afford food and regular expenses, and bu...	

	education	employment_status	ethnicity	\
0	College or university	Unemployed	Mestizo	
1	Secondary/high school	Student	Tagalog	
2	College or university	Student	Hiligaynon	
3	College or university	Unemployed	Thai	
4	College or university	Employed full-time	African	
...	
12349	College or university	Unemployed	African	
12350	Secondary/high school	Self-employed	Kru	
12351	Technical school	Unemployed	Non-hispanic White	
12352	College or university	Self-employed	Thai	
12353	Secondary/high school	Employed full-time	Mestizo	

	religion	... wmn_pre_safe_place	wmn_post_safe_place	\
0	Catholicism	...	NaN	NaN
1	Muslim	...	NaN	NaN
2	Christianity	...	NaN	NaN
3	Buddhism	...	NaN	NaN
4	Christianity	...	NaN	NaN
...
12349	Muslim	...	NaN	NaN
12350	Catholicism	...	NaN	NaN
12351	Prefer Not To Answer	...	Every day	Rarely
12352	Buddhism	...	NaN	NaN
12353	Catholicism	...	NaN	NaN

	wmn_safe_place_no_access	wmn_safe_place_no_access_why	wmn_pre_help	\
0	NaN	NaN	NaN	
1	NaN	NaN	NaN	
2	NaN	NaN	NaN	
3	NaN	NaN	NaN	
4	NaN	NaN	NaN	
...	
12349	NaN	NaN	NaN	
12350	NaN	NaN	NaN	
12351	No	NaN	Yes	
12352	NaN	NaN	NaN	
12353	NaN	NaN	NaN	

	wmn_post_help	wmn_post_no_help	wmn_no_help_why	\
0	NaN	NaN	NaN	
1	NaN	NaN	NaN	
2	NaN	NaN	NaN	
3	NaN	NaN	NaN	
4	NaN	NaN	NaN	
...	
12349	NaN	NaN	NaN	
12350	NaN	NaN	NaN	
12351	Yes	Yes	Decline to Answer	
12352	NaN	NaN	NaN	
12353	NaN	NaN	NaN	

	country	user_id
0	Ecuador	wmn_5900473574883328
1	Philippines	wmn_5702261783658496
2	Philippines	wmn_5652767014387712
3	Thailand	wmn_6411372690669568
4	United Republic of Tanzania	wmn_6215734184378368
...
12349	United Republic of Tanzania	wmn_6151550260215808
12350	Ivory Coast	wmn_5222308327456768
12351	United States	wmn_4706368994148352
12352	Thailand	wmn_6730228637892608
12353	Ecuador	wmn_6585614141489152

[12354 rows x 46 columns]

```
In [3]: print(df['wmn_hh'])
```

0	3.0
1	13.0
2	5.0
3	7.0
4	3.0
...	...
12349	7.0
12350	3.0
12351	4.0
12352	4.0
12353	3.0

Name: wmn_hh, Length: 12354, dtype: object

Srednia arytmetyczna (Mean)

```
In [5]: mean = np.mean (df['wmn_hh'])
print(mean)
```

4.344260968107496

Mediana (Median)

```
In [9]: median = np.median(df['wmn_hh'])
print(median)
```

4.0

Odchylenie standardowe (Standard Deviation)

```
In [ ]: stddev = np.std(df['wmn_hh'])
print(stddev)
```

Wariancja (Variance)

```
In [ ]: variance = np.var(df['wmn_hh'])
print (variance)
```

Korelacja (Correlation)

```
In [ ]: data1 = [ 1 , 2 , 3 , 4 , 5 ]
data2 = [ 5 , 4 , 3 , 2 , 1 ]
correlation = np.corrcoef(data1 ,data2 ) [0,1]
print(correlation)
```

```
In [ ]: Kowariancja (Covariance)
```

```
In [ ]: covariance = np.cov (data1, data2) [0, 1]
print (covariance)
```