

# Python - Analiza danych z modulem PANDAS

[www.udemy.com](http://www.udemy.com) (<http://www.udemy.com>) (R)

## S02-L006 - LAB - Filtrowanie Data Series

1. Zaimportuj module: pandas, numpy, matplotlib (tylko pyplot), math i nadaj im standardowe aliasy
2. Na stronie <https://danepubliczne.gov.pl> (<https://danepubliczne.gov.pl>) publikowane są ciekawe dane statystyczne. Ta lista prezentuje informacje o wypadkach powodowanych przez osoby w różnym wieku: 14 ,334,312,5823,9491,7486,4343. Wiek tych osób to kolejno: do 6, 7-14, 15-17, 18-24, 25-39, 40-59, 60 i więcej). Utwórz serię zawierającą takie dane. Indeksami niech będzie wiek, a wartościami ilość wypadków. Serię nazwij incidents. Wyświetl serię.
3. Wyświetl tylko te grupy, które spowodowały więcej niż 1000 wypadków. Najpierw pozwól na wyświetlenie NaN dla grup, które mają mniej wypadków, a potem zupełnie je wyeliminuj
4. Zapisz wynik poprzedniego polecenia w zmiennej incident1000. Wyświetl tę nową zmienną, żeby się przekonać, że nie zawiera grup o małej ilości wypadków.
5. Upewnij się, że oryginalna seria incidents nie zmieniła zawartości. Jeżeli zmieniła, wykonaj ponownie instrukcje z kroku (2)
6. Wyświetl tylko dane dla grup osób w wieku 18-59 (3 grupy) - filtrowanie po indeksie
7. Wyfiltruj w serii incidents tylko te pozycje, które liczbę wypadków mają mniejszą lub równą 1000. Po operacji filtrowania obiekt powinien zawierać tylko te wartości. Wyświetl serię incidents, aby upewnić się, że nie ma w niej innych grup.

## Rozwiązania:

Poniżej znajdują się propozycje rozwiązań zadań. Prawdopodobnie istnieje wiele dobrych rozwiązań, dlatego jeżeli rozwiązujesz zadania samodzielnie, to najprawdopodobniej zrobisz to inaczej, może nawet lepiej :) Możesz pochwalić się swoimi rozwiązaniami w sekcji Q&A

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import math as math
```

```
In [2]: ix = ["do 6", "7-14", "15-17", "18-24", "25-39", "40-59", "60 i więcej"]
val = [14 ,334,312,5823,9491,7486,4343]
incidents = pd.Series(data = val, index=ix)
incidents
```

```
Out[2]: do 6          14
7-14         334
15-17        312
18-24       5823
25-39       9491
40-59       7486
60 i więcej  4343
dtype: int64
```

```
In [3]: incidents.where(incidents>1000)
```

```
Out[3]: do 6          NaN
        7-14         NaN
        15-17        NaN
        18-24        5823.0
        25-39        9491.0
        40-59        7486.0
        60 i więcej  4343.0
        dtype: float64
```

```
In [4]: incidents.where(incidents>1000).dropna()
```

```
Out[4]: 18-24        5823.0
        25-39        9491.0
        40-59        7486.0
        60 i więcej  4343.0
        dtype: float64
```

```
In [5]: incidents1000 = incidents.where(incidents>1000).dropna()
        incidents1000
```

```
Out[5]: 18-24        5823.0
        25-39        9491.0
        40-59        7486.0
        60 i więcej  4343.0
        dtype: float64
```

```
In [6]: incidents
```

```
Out[6]: do 6          14
        7-14         334
        15-17        312
        18-24        5823
        25-39        9491
        40-59        7486
        60 i więcej  4343
        dtype: int64
```

```
In [7]: incidents.filter(items=["18-24", "25-39", "40-59"])
```

```
Out[7]: 18-24    5823
        25-39    9491
        40-59    7486
        dtype: int64
```

```
In [8]: incidents.where(incidents <= 1000, inplace = True)
        incidents.dropna(inplace=True)
        incidents
```

```
Out[8]: do 6         14.0
        7-14        334.0
        15-17        312.0
        dtype: float64
```