

### Zadanie 1: Łączenie i filtrowanie wielu źródeł danych

Masz dwie listy: jedna zawiera imiona użytkowników, druga ich wyniki:

```
users = ["Ala", "Ola", "Jan", "Ela", "Kot"]
```

```
scores = [75, 45, 90, 60, 50]
```

Napisz funkcję funkcyjną, która:

- Połączy dane w pary (name, score)
- Przetnie tylko tych użytkowników, którzy mają wynik co najmniej 60
- Zwróci posortowaną (malejąco) listę ich imion w wersji uppercase

Oczekiwany wynik:

```
['JAN', 'ALA', 'ELA']
```

### Zadanie 2: Oceny uczniów i klasyfikacja

Masz dwie listy:

```
students = ["Ania", "Bartek", "Celina", "Darek", "Ewa"]
```

```
grades = [4.5, 2.0, 3.0, 5.0, 3.5]
```

- Połącz uczniów z ich ocenami.
- Odfiltruj tylko tych uczniów, którzy **zdali** (ocena  $\geq 3.0$ ).
- Zamień dane na formę tekstową: "IMIĘ - ZDAŁ(A) z oceną X.X" (imię wielkimi literami).
- Posortuj listę alfabetycznie po imieniu

Oczekiwany wynik:

```
[  
    "ANIA - ZDAŁ(A) z oceną 4.5",  
    "CELINA - ZDAŁ(A) z oceną 3.0",  
    "DAREK - ZDAŁ(A) z oceną 5.0",  
    "EWA - ZDAŁ(A) z oceną 3.5"  
]
```

### Zadanie 3: Transformacja macierzy (funkcyjnie)

Dla danej macierzy (listy list):

```
matrix = [  
    [1, 2, 3],  
    [4, 5, 6],  
    [7, 8, 9]  
]
```

Użyj map i lambda, aby:

- Transponować macierz (zamienić wiersze z kolumnami),
- Następnie przekształcić każdy wiersz w sumę elementów podniesionych do kwadratu.

Oczekiwany wynik:

```
[66, 93, 126] # (1^2 + 4^2 + 7^2), (2^2 + 5^2 + 8^2), (3^2 + 6^2 + 9^2)
```

### Zadanie 4: Potok funkcji (pipeline)

Zdefiniuj funkcję wyższego rzędu pipeline, która:

- Przyjmuje listę funkcji jednoargumentowych,
- Zwraca nową funkcję, która stosuje je wszystkie kolejno do danego argumentu.

Następnie użyj jej, by utworzyć przekształcenie:

dodaj 3, pomnóż przez 2, odejmij 5, podnieś do kwadratu.

Dla  $x = 4$  powinno to dać:

```
((4 + 3) * 2 - 5)^2 = (14 - 5)^2 = 9^2 = 81
```

### **Zadanie 5: Przekształcenia tekstowe (czysto funkcyjne)**

Dla danego tekstu:

text = "omg JPWP jest super"

Zadanie:

- Podziel tekst na słowa,
- Usuń wszystkie słowa krótsze niż 4 litery,
- Przekształć każde słowo na: długość + dwie ostatnie litery (np. "super" → "5er"),
- Zwróć wynikową listę tych przekształconych słów.

Oczekiwany wynik:

['4WP', '4st', '5er']