

Zadanie 1 — Walidator listy operacji matematycznych

Napisz funkcję `calculate_operations(operations)`, która przyjmuje listę krotek w formacie:

```
[(liczba1, operator, liczba2), ...]
```

Operator to jeden z: "+", "-", "*", "/".

Funkcja powinna:

- obliczyć wynik dla każdej krotki,
- obsłużyć sytuacje:
 - dzielenie przez zero,
 - nieznany operator,
 - nieprawidłowy typ danych (np. tekst zamiast liczby).

Jeśli operacja się nie uda, do wyniku wpisz tekst "Błąd: opis".

Przykład wejścia:

```
[(2, "+", 3), (5, "/", 0), ("a", "*", 3), (10, "^", 2)]
```

Przykład wyjścia:

```
[5, "Błąd: dzielenie przez zero", "Błąd: nieprawidłowy typ danych",  
"Błąd: nieznany operator"]
```

Zadanie 2 — Symulator rezerwacji miejsc

Masz listę miejsc w kinie: ["A1", "A2", "A3", "B1"].

Napisz funkcję `reserve_seat(seats, seat_code)`, która:

- rezerwuje miejsce, jeśli jest dostępne,
- wyrzuca wyjątek, jeśli:
 - miejsce nie istnieje (`ValueError`),
 - miejsce jest już zajęte (`Exception` własny typ wyjątku).

Zadanie 3 — System koszyka z walidacją cen

Stwórz funkcję `add_to_cart(cart, product, price)`, która:

- dodaje produkt i cenę do koszyka (`dict`),
- jeśli cena jest ujemna, wyrzuca `ValueError`,
- jeśli produkt już jest w koszyku, wyrzuca `KeyError`,
- łapie wyjątki i zwraca komunikat błędu zamiast przerywać działanie.
-

Zadanie 4 — Mini parser JSON (bez bibliotek)

Napisz funkcję `parse_json_like(string)`, która:

- próbuje zamienić łańcuch w formacie `"{'key': 'value'}"` na słownik,
- jeśli wystąpi błąd składni, zgłasza wyjątek i zwraca `"Błąd: niepoprawny format"`.

UŻYJ FUNKCJI EVAL!