

Zadanie 1.

Stwórz abstrakcyjną klasę `Animal` i trzy klasy (nieabstrakcyjne) które po niej dziedziczą. Klasa `Animal` powinna mieć w sobie abstrakcyjną metodę `makeSound()`, a klasy dziedziczące implementację tej metody.

Stwórz obiekty klas dziedziczących, umieść je we wspólnej tablicy i pokaż, że działają.

Zadanie 2.

Utwórz klasy `Figura`, `Figura2D` i `Figura3D`.

Niech klasa `Figura` będzie abstrakcyjna i umożliwi obliczenie pola całkowitego (`double`) oraz obliczenie sumy krawędzi (`int`). Niech klasa `Figura` posiada konstruktor w postaci tablicy typu `int` stanowiącej zbiór wszystkich krawędzi.

Klasy `Figura2D` i `Figura3D` powinny dziedziczyć po klasie `Figura`.

`Figura3D` dodatkowo posiada metodę do obliczenia objętości figury (`double`).

Następnie zaimplementuj klasy `Prostokat` oraz `Kwadrat` pamiętając o odpowiednim dziedziczeniu.

Następnie utwórz klasę `Prostopadloscian`, która będzie dziedziczyć po odpowiedniej klasie z wyżej wymienionych i zaimplementuj wszystkie metody odziedziczone po klasach nadrzędnych. Na koniec utwórz klasę `Szescian` i odpowiednio podepnij do łańcucha dziedziczenia.

Jeżeli użytkownik przy tworzeniu jakiegokolwiek z obiektów jako argument zostanie podane 0, program powinien rzucić wyjątek i zakończyć działanie.

Przetestuj swoje rozwiązanie w bloku `psvm` klasy `Main`.

Podpowiedź

W klasie dziedziczącej możesz zmienić konstruktor tak, aby przyjmował np. 1 parametr typu `int`, a następnie wewnątrz konstruktora przekazać ten parametr w innej formie, tak aby spełnić wymagania konstruktora nadrzędnego np.

```
Kwadrat(int bok){
    super(new int[]{bok,bok,bok,bok});
}
```

Zadanie 3

Napisz program, który odczytuje od użytkownika cyfrę, a następnie wykonuje odpowiednie działanie

matematyczne :

1. Dodawanie
2. Odejmowanie
3. Mnożenie
4. Dzielenie

Jeśli użytkownik nie poda cyfry albo poda spoza zakresu to program wyświetla komunikat o niepoprawnym wyborze i zachęca do podania cyfry ponownie.

(ciąg dalszy na drugiej stronie)

Po poprawnym wskazaniu cyfry 1-4 program prosi użytkownika kolejno o podanie pierwszej i drugiej cyfry do wykonania wskazanej operacji. W przypadku niepoprawnej cyfry program wyświetla komunikat o błędzie i zachęca do podania cyfry ponownie.

Na koniec program wyświetla wynik w następującej formie :

Wynik {operacja} na liczbach '{liczba1}' oraz '{liczba2}' wynosi '{wynik}'

Np. Wynik dodawania na liczbach '2' oraz '3' wynosi '5'

W zadaniu wykorzystaj `scanner.nextLine()` do pobrania danych od użytkownika oraz `Integer.parseInt()` do zamiany na liczbę. Pamiętaj o odpowiedniej obsłudze wyjątku.