

GUI03

March 26, 2022

1 Zadanie 1

Stwórz klasę **Kontener**, która będzie przyjmowała typ generyczny T. Klasa ma posiadać pole podanego typu oraz konstruktor je inicjujący. Następnie, napisz metodę **wypisz()** wypisującą wartość pola. Stwórz dwa obiekty klasy **Kontener**, parametryzując je różnymi typami np. String, Integer. Wywołaj metodę **wypisz()**, na obu obiektach.

2 Zadanie 2

Zmodyfikuj poprzednią Klasę **Kontener**, tak, aby posiadała kolejne pole typu generycznego S. Zapewnij aby było ono również inicjalizowane w konstruktorze, oraz żeby metoda **wypisz()** wypisywała wartości obu pól.

3 Zadanie 3

Stwórz obiekt klasy **ArrayList** sparametryzowany na obiekty typu **String**. Dodaj do listy następujące ciągi znaków: "Ala", "ma", "Kota". Wyświetl długość danej listy, następnie za pomocą petli wypisz zawartość całej listy.

4 Zadanie 4

Napisz program, w którym utworzysz obiekt typu **HashSet**, sparametryzowany na obiekty typu Integer. Następnie dodaj do podanej kolekcji, kolejno liczby dni poszczególnych miesięcy, tj. 31, 28, 31, 30 itd. Następnie wypisz wszystkie elementy podanej kolekcji.

5 Zadanie 5

Napisz program, analogiczny do zadania 4, tj. przechowujący tym razem w kolekcji **HashMap** liczbe dni miesiecy, wraz z nazwa podanego miesiaca. Następnie wypisz liczbe dni dla miesiaca marca, a następnie wypisz zawartość całej kolekcji.

6 Zadanie 6

Stwórz klasę **Osoba**, która będzie zawierała imię i nazwisko danej osoby. Następnie stwórz klasę **Samochód** zawierającą markę oraz numer rejestracyjny pojazdu. Przy pomocy mapy haszującej (**HashMap**) przypisz samochody do ich właścicieli. Przeiteruj po rekordach i wypisz właścicieli i samochody do nich należące, a następnie wypisz tylko te samochody, których numer rejestracyjny zaczyna się od „WA”. Spraw aby poniższa metoda main działała.

verbatim

```
public static void main(String[] args) {  
    Osoba kowalski = new Osoba("Jan", "Kowalski");  
    Osoba nowak = new Osoba("Adam", "Nowak");
```

```

Osoba krawczyk = new Osoba("Bartosz", "Krawczyk");
Osoba heniek = new Osoba("Kierownik", "Heniek");
Samochod skoda1 = new Samochod("WA00001", "Skoda");
Samochod skoda2 = new Samochod("SC36010", "Skoda");
Samochod mazda1 = new Samochod("WA01234", "Mazda");
Samochod mazda2 = new Samochod("DW01ASD", "Mazda");
Samochod bmw = new Samochod("WA12690", "BMW");
Samochod volvo = new Samochod("KR60606", "Volvo");
/*
* Jan Kowalski -> SKODA WA00001, BMW WA12690
* Adam Nowak -> MAZDA DW01ASD
* Bartosz Krawczyk -> VOLVO KR60606, MAZDA WA01234, SKODA SC36010
* Kierownik Heniek -> [Brak samochodów]
* Samochody, których numery rejestracyjne zaczynają się na WA:
* SKODA WA00001 * BMW WA12690 * MAZDA WA01234 * */
for (/* ... */ : mapaSamochodow.entrySet()) {
/* ...? */ System.out.println(/* ... */ + " posiada " + /* ... */ );
// np: Jan Kowalski posiada 3 pojazdy
}
System.out.println(mapaSamochodow.get(nowak).get(0)); // MAZDA DW01ASD
}

```

7 Zadanie 7

- Utwórz klasę abstrakcyjną `AAAnimal` z polami prywatnymi `name` oraz `group` i metodą `toString()`.
- Utwórz klasę abstrakcyjną `ACarnivore` dziedziczącą po `AAAnimal` z metodą abstrakcyjną `isScavenger()`;
- Utwórz klasę abstrakcyjną `AHerbavore` dziedziczącą po `AAAnimal`;
- Utwórz klasę generyczną `Enclosure<T>` extends `AAAnimal`, która pozwoli na przechowywanie zwierząt w klatkach. Zwierzęta w danej klatce będą przechowywane na liście. Utwórz 2 metody `addAnimalToEnclosure`, `getEnclosure`.
- Utwórz enum `Group`, który będzie zawierał grupy zwierząt `MAMMAL`, `BIRD`, `AMPHIBIAN`, `INSECT`
- Utwórz klasę `Worker` zawierającą pole `name` oraz metody `public void cleanAnimal(AAnimal animal)`, `public void feedAnimal(AAnimal animal)`, `toString`
- Utwórz klasę `Zoo`, która będzie zawierała 2 pola prywatne:
 - `private List<Enclosure> enclosureList = new ArrayList();`
 - `private List<Worker> workerList = new ArrayList();` oraz metody:
 - `public void showAllAnimals()`
 - `public void showAllWorkers()` Pamiętaj o konstruktorach, getterach i setterach oraz konstruktorach