

Zadanie 1.

1. Stworzyć klasę Food, która posiada metodę abstrakcyjną void prepare() (w klasach nadpisujących niech wypisują pole name)
2. Klasa Food powinna przyjmować i ustawiać następujące pole w konstruktorze:
 - protected final String name
3. Stworzyć klasy Nudle, Cabbage, Beef dziedziczące po klasie Food, konstruktor bezparametrowy z wewnętrznym przekazaniem nazwy do konstruktora nadrzędnego
4. Stworzyć klasę Chef, która jest parametryzowana przez klasę rozszerzającą Food
5. Klasa Chef powinna posiadać metodę void prepareMeal(T foodToPrepare) wywołującą metodę prepare() na obiekcie foodToPrepare
6. Stworzyć klasę ~~GenericChefTest~~ GenericChefRunner (pamiętamy dlaczego ? (□■_■)) z psvm
7. W metodzie main stworzyć tylu kucharzy ile jest klas, które dziedziczą po Food, każdy z odpowiednim typem generycznym (Chef<Nudle> itd.)
8. Stwórz obiekty jedzenia (Cabbage, Nudle, Beef)
9. Na każdym kucharzu wywołać metodę prepareMeal z różnymi rodzajami jedzenia, kiedy zadziała, kiedy nie ?
10. Na jakich obiektach mogą operować kucharze ?
11. Stwórz obiekt Top-chefa (Chef<Food>), jakie przyjmuje argumenty ?

Zadanie 2.

1. Napisz klasę Pair przyjmującą 2 typy generyczne Pair<T,R>, która przechowuje wartości poszczególnych typów, wykorzystaj je w konstruktorze, oraz zwróci przy metodach getFirst() (typ T) oraz getSecond() (typ R)
2. Napisz analogiczną klasę Triple wykorzystującą 3 typy generyczne (np. T, R, S), zwracającą analogicznie za pomocą metod getFirst(), getSecond(), getThird() obiekty typów T, R, S
3. Stwórz klasę FiveElements, która wykorzysta dwa powyższe typy do zwrócenia obiektów pięciu typów: getFirst(), getSecond(), getThird(), getFourth(), getFifth()
4. Stwórz obiekty poszczególnych typów (punkty 1-3), wyświetl pobrane wartości w konsoli, niech: a. Pair będzie typu String i Integer, b. Triple typu Double, Float, Character, c. FiveElements typu String, Integer, Double, Float, Character
5. Czy istnieje możliwość przechowywania pięciu typów bez użycia klasy FiveElements, wykorzystując jedynie Pair oraz Triple ?