

Programowanie obiektowe w Java

mgr inż. Adam Zalewski

Zestaw nr 7

Zadania:

1. Utwórz klasę **Magazyn**, która przechowuje - **produkty**, oraz ich **ilość**. Dodaj konstruktor, który ustawia wartości pól w obiekcie oraz metodę, która wyświetla asortyment magazynu. Zmodyfikuj poprzedni kod, aby działał z nową klasą.
2. Utwórz klasę **Adres** z polami **ulica**, **numerDomu**, **numerMieszkania** (**opcjonalnie**), **miasto**, **kodPocztowy**. Zdefiniuj dwa konstruktory: jeden z numerem mieszkania i jeden bez. Zdefiniuj metodę **pokaz**, która wypisuje w pierwszej linii kod pocztowy i miasto, a w następnej linii pozostałe informacje oraz metodę **przed**, która sprawdza czy dany adres występuje przed innym (porównuj tylko kody pocztowe). Dodaj adres do klasy **Klient**.
3. Zmodyfikuj klasę **KoszykZakupowy**, w taki sposób, że przechowuje ona listę produktów w postaci **produkty**, oraz ich **ilość**. Dodaj do klasy **Sklep** pola **nazwaSklepu**, **dataPowstania** oraz **magazynSklepu**. Odpowiednio zaktualizuj konstruktor. Data powstania nie może być datą z przyszłości.
4. Zastąp wszystkie metody wyświetlające informację o klasach metodą **toString**. Zaimplementuj polimorfizm dynamiczny. (+1 pkt)
5. Nadpisz metodę **getClass**, która zwraca napis „Cześć jestem klasą <nazwa klasy>”
6. Nadpisz metodę **hashCode**, która zwraca hash w podanych klasach:
Produkt: hash dla nazwy i ceny
Platnosc: hash dla kwoty i statusu płatności
Klient: hash dla imienia, nazwiska oraz adresu
7. Ustaw modyfikatory dostępu do pól w klasach na prywatne.
8. Napisz metody typu **getter** i **setter** dla wszystkich pól we wszystkich klasach. Waliduj dane wejściowe w setterach (np. czy nie są null).
9. Napisz wyjątki w **setterach** kiedy wartość nie jest poprawna np.: **throw new IllegalArgumentException(<komunikat>)**
10. Nadpisz metodę **equals** dla podanych klas. Dwie klasy są równe kiedy:
Produkt: posiada identyczną nazwę i cenę
Platnosc: posiada identyczną kwotę i status płatności
Klient: posiada identyczne imię, nazwisko oraz adres (+1 pkt)