

3. Klasy

Zadanie 3.1 Proste klasy modelu: Ogłoszenia

Zaproponuj klasę, w której obiektach będzie się zapisywać ogłoszenia (takie jak w serwisie internetowym z ogłoszeniami). Użyj właściwych typów dla poszczególnych pól.

Najlepiej, aby klasa **Ogłoszenie** opisywała rzeczy, które posiada każde ogłoszenie, m.in. tytuł, opis, cenę, dane kontaktowe sprzedawcy, a dodatkowo stwórz klasy, które rozszerzają tę klasę („podklasy”) opisujące konkretne rodzaje ogłoszeń.

OgłoszenieSamochodowe – dziedziczy z **Ogłoszenie** i dodatkowo określa cechy sprzedawanego samochodu jak model, markę, rok produkcji, stan licznika ;), pojemność, moc i rodzaj paliwa (spróbuj użyć **enum**).

OgłoszenieMieszkaniowe – też dziedziczy z **Ogłoszenie**, a dodaje cechy sprzedawanego mieszkania / domu: miejscowość, metraż, liczba pokoi. Ewentualnie można utworzyć kolejne poziomy podklas: osobno dla nieruchomości różnego typu (mieszkanie / dom).

Własne pomysły mile widziane.

W używanym IDE wygeneruj standardowe konstruktory i metody. Dodaj inne metody według uznania.

3.1.1 Lista ogłoszeń

Warto przećwiczyć operacje na Javowych listach. Przykładowo możesz spróbować napisać programy, w których tworzysz przykładową listę ogłoszeń samochodowych, a następnie wyszukujesz:

1. Ogłoszenia o cenach z określonego przedziału.
2. Ogłoszenia dotyczące samochodów o mocy silnika powyżej podanej, a cenie poniżej podanej.

3.1.2 Odczyt ogłoszeń z pliku

Kiedy na kursie pojawiają się elementy obsługi plików tekstowych, dodaj do projektu z ogłoszeniami metody pozwalające na odczyt i zapis listy ogłoszeń z/do pliku w (uproszczonym) formacie CSV. Napisz program, który czyta ogłoszenia z pliku i wyświetla tylko ogłoszenia z określonego przedziału cenowego.

Zadanie 3.2 Stacja Paliw - wersja uproszczona

Napisz klasę `StacjaPaliw`. Każda stacja (obiekt tej klasy) obsługuje trzy rodzaje paliwa: ON, PB95, PB98. Operacje wykonuje się podając rodzaj paliwa jako `String`. Stacja pamięta ceny oraz stan poszczególnych paliw (ilość paliwa w zbiornikach).

Klasa powinna posiadać konstruktor, w którym podaje się nazwę stacji oraz trzy liczby: ceny poszczególnych paliw:

- **`StacjaPaliw`**(`String nazwa`, `double cenaON`, `double cenaPB95`, `double cenaPB98`)

Klasa powinna też posiadać publiczne metody:

- `double zatankuj`(`String paliwo`, `double ilosc`) – jeśli jest wystarczająca ilość paliwa, to odejmuje podaną ilość ze stanu zbiornika i zwraca w wyniku kwotę do zapłaty; jeśli nie ma takiej ilości paliwa, wyrzuca wyjątek
- `void dostawa`(`String paliwo`, `double ilosc`) – dodaje do stanu danego paliwa podaną ilość,
- `void ustawCene`(`String paliwo`, `double nowaCena`) – wiadomo
- `void wypiszCennik`() – wypisuje na konsolę nazwę stacji i ceny wszystkich paliw.

Dodatkowo można dodać `toString()` i gettery (ale raczej nie settery).

Przykładowe użycie w innym programie:

```
StacjaPaliw stacja = new StacjaPaliw("Stacja Ryśka", 4.90, 5.05, 5.25);
stacja.dostawa("ON", 100);
double doZapłaty = stacja.zatankuj("ON", 40);
```

Jeśli się uda - napiszcie program interaktywny działający w oparciu o `Scanner`, a dla bardzo chętnych - wersja `Swing` z okienkami.

Przypomnienie: napisy porównujemy za pomocą `equals`:

```
if("PB95".equals(paliwo)) { ....
```

albo używamy `switch`:

```
switch(paliwo) {
case "PB95":....
```

4. Przetwarzanie plików

Zadanie 4.1 *Dane skoczków narciarskich*

Korzystając z pliku CSV z danymi skoczków narciarskich (zawodnicy.csv) napisz programy, które wczytują ten plik i:

1. wypisuje najwyższego, najniższego, najcięższego i najlżejszego skoczka; gdyby kilku miało taką samą wagę lub wzrost, to wystarczy wypisać jednego z nich.
2. liczy ile łącznie ważą reprezentanci Polski (np. żeby sprawdzić czy zmieszczą się w windzie na skocznię ;)). Pozwól użytkownikowi podać kraj (niekoniecznie musi być Polska).
3. (trudniejsze) dla wszystkich (!) krajów oblicza ilu jest zawodników z tego kraju; tzn. ma się wypisać, być może w innej kolejności:
AUT – 2
FIN – 3
GER – 5
NOR – 3
POL – 3
USA – 1

To zapewne wymaga użycia kolekcji Map; jeśli jeszcze nie znasz, nie rób tych punktów

4. jak wyżej, ale liczy jeszcze dla każdego kraju średni wzrost zawodników.