



Záverečný test

Zadanie



Ústav informatiky
Prírodovedecká fakulta
UPJŠ v Košiciach

Dvakrát meraj (rozmyšľaj), raz rež (programuj)

Dôležité pravidlá a informácie (viac na stránke predmetu):

- čas na riešenie úloh je **240 minút**,
- nie je dovolená žiadna (elektronická aj neelektronická) komunikácia s kýmkoľvek okrem dozoru
- v prípade akýchkoľvek problémov alebo z dôvodu ohodnotenia riešenia kontaktujte dozor,
- riešenia je možné nechať si ohodnotiť aj priebežne,
- **funkčnosť každej metódy musí byť preukázaná spustením na vami vytvorenom testovacom vstupe, nespustiteľné metódy neumožňujú zisk príslušných bodov,**
- všetky inštančné premenné musia byť neverejné.

Autobazár

Motivácia: Väčšina ľudí skôr či neskôr uvažuje o kúpe vlastného auta. Jednou z možností je kúpa ojazdeného auta. Tieto autá sa najčastejšie predávajú cez autobazáre. Autobazár *PAZCars* je jedným z nich. Keďže počet áut v ponuke a počet predajov im neustále rastie, rozhodli sa využívať na evidenciu vlastný informačný systém. Autobazár si eviduje informácie o všetkých autách, ktoré prijme na predaj. Pri prijatí auta do autobazáru sa zaevidujú technické údaje o aute, očakávaná cena a dátum prijatia auta (ďalšie reálne zapisované údaje nebudeme kvôli zjednodušeniu uvažovať). Umiestnenie auta v autobazári nie je bezplatné. Autobazár si účtuje parkovné. Navyše pri predaji auta si účtuje províziu za predaj. Parkovné aj províziu platí (pôvodný) majiteľ auta. Auto odchádza z autobazáru v dvoch prípadoch: auto sa predá alebo sa majiteľ rozhodne auto z ponuky stiahnuť. Autobazár si vtedy v evidencii poznačí tento dátum a v prípade, že išlo o predaj aj reálnu predajnú cenu. Predajná cena môže byť iná (nižšia) ako cena v ponuke – kvôli možnosti zjednávať cenu.



Pohľad analytika: Pri implementácii budeme potrebovať:

- triedu `Auto`, ktorá reprezentuje záznam o aute, jeho majiteľovi a prípadnom predaji,
- triedu `Autobazar`, ktorá bude uchovávať zoznam áut.

Zadanie: V balíku `sk.upjs.finalTerm` vytvorte triedu `Auto` obsahujúcu dátové položky prístupné cez `getter` (a podľa uváženia aj modifikovateľné cez `setter`):

- **znacka** (Honda, Kia, Lexus, Mitsubishi, Peugeot, Suzuki, Toyota, Volkswagen, ...),
- **model** (Civic, Optima, IS 250, Pajero, 208, SX4, Corolla, Golf, ...),
- **motor** (vo formáte objem motora v litroch a typ motora napr.: 1.5 i-VTEC Turbo, 1.7 CRDi, 2.5 V6, 2.6 i, 1.2 PureTech, 1.6 VVT, 1.6 Valvematic, 2.0 TDI, ...),
- **typ** (sedan, liftback, hatchback, kombi, SUV, crossover, ...),
- **rok** (rok výroby),
- **najazdeneKm** (najazdené kilometre podľa odometra),
- **cena** (cena auta, za ktorú by chcel majiteľ auto predať),

- **spz** (prázdna v prípade neprihláseného vozidla alebo v tvare `XXYYYYXX`, kde X je písmeno a Y je číslica – vlastné ŠPZ kvôli zjednodušeniu neuvažujeme; pripomeňme, že prvé 2 písmená reprezentujú okres, v ktorom je auto zaregistrované),
- **datumPrijatia** (dátum v tvare `dd.mm.rrrr`, kedy bolo auto dovezené do autobazáru),
- **datumOdovzdania** (dátum v tvare `dd.mm.rrrr`, kedy bolo auto odvezené z autobazáru – auto bolo predané alebo majiteľ stiahol auto z predaja),
- **predajnaCena** (cena, za ktorú sa auto predalo; v prípade, ak bolo auto stiahnuté z predaja majiteľom, je ako predajná cena uvedená hodnota 0).

Upozornenie: Zadané pre triedu `Auto` predpisuje dátové položky prístupné cez `gettre`. Aké privátne inštančné premenné použijete na uloženie týchto dátových položiek je na vašom rozhodnutí.

Ďalej vytvorte triedu `sk.upjs.finalTerm.Autobazar`, ktorá bude uchovávať zoznam áut.

Konštruktory a pridávanie požiadaviek (3 body dokopy – povinné):

- **public** `Auto(String znacka, String model, String motor, String typ, int rok, int najazdeneKm, int cena, String spz, String datumPrijatia)` – vytvorí záznam o aute, ktoré je aktuálne v ponuke autobazáru,
- **public** `Auto(String znacka, String model, String motor, String typ, int rok, int najazdeneKm, int cena, String spz, String datumPrijatia, String datumOdovzdania, int predajnaCena)` – vytvorí záznam o aute, ktoré sa už nenachádza v autobazári (bolo predané alebo stiahnuté z predaja),
- **public void** `pridaj(Auto auto)` – inštančná metóda v triede `Autobazar`, ktorá pridá záznam o aute do autobazáru.

Práca so súborami (povinné):

V triede `Auto`:

- **public static** `Auto zoStringu(String popis)` – statická metóda, ktorá vráti referenciu na novovytvorený objekt triedy `Auto`. Parameter je `String` v tvare `"znacka \t model \t motor \t typ \t rok \t najazdeneKm \t cena \t spz \t datumPrijatia"`, resp. `"znacka \t model \t motor \t typ \t rok \t najazdeneKm \t cena \t spz \t datumPrijatia \t datumOdovzdania \t predajnaCena"`, ak auto už nie je v ponuke autobazáru (3 body);
Poznámka: Znak `\t` je neviditeľný znak tabulátora. Scanner-u môžete povedať, že oddeľovač má byť tabulátor zavolaním jeho metódy `useDelimiter("\t")`.
- **public** `String toString()` – vráti reťazec reprezentujúci záznam o aute vo vhodnom tvare (1 bod).

V triede `Autobazar`:

- **public static** `Autobazar zoSuboru(String nazovSuboru)` – statická metóda, ktorá z uvedeného súboru prečíta zoznam záznamov o autách, pričom v každom riadku bude záznam iba jedného auta (4 body).
- **public void** `uloz(String nazovSuboru)` – uloží všetky záznamy o autách z autobazáru do súboru v tvare, ktorý vie spracovať metóda `zoSuboru(String nazovSuboru)` (3 body).
- **public** `String toString()` – vráti reťazec vhodne reprezentujúci obsah autobazáru (1 bod).

Inštančné metódy triedy `Autobazar`:

Ak niektorá z metód nevie vrátiť referenciu na objekt s požadovanými vlastnosťami, metóda nech vráti `null`.

- **public** `List<Auto> vratAutaVPredaji()` – vráti zoznam áut ponúkaných na predaj (1 bod).

- **public int** vratZinkasovanuProviziu() – vráti celkovú zinkasovanú províziu zo všetkých predaných áut. Sprostredkovateľská provízia je 3,8 % z predajnej ceny auta minimálne však 100 € (3 body).
- **public** List<String> vratVsetkyZnacky() – vráti zoznam značiek áut autobazáru, pričom každá značka je v zozname len raz (3 body).
- **public** Auto vratNajstarsieAuto() – vráti najstaršie auto, aké kedy autobazár má alebo mal v ponuke (3 body).
- **public** Autobazar vratAutobazarTypuAut(String typ) – vráti referenciu na novovytvorený Autobazár obsahujúci len záznamy zadaného typu áut (3 body).
- **public** List<Auto> autaSMaxObjemom(**double** maxObjemMotora) – vráti zoznam áut v predaji, s objemom motora menším alebo rovným ako je zadaná hodnota. Objem motora je prvé číslo v popise motora (4 body).
- **public** Map<String, Integer> autaPodlaOkresov() – pre každý okres (KE, KS, RV, PO, BJ, ...) určíme počet áut, ktoré boli v autobazári z daného okresu (6 bodov).
- **public** Map<String, Integer> priemernaCenaTypuAut() – pre každý typ auta vráti jeho priemernú cenu zaokrúhlenú na celé eurá (6 bodov).
- **public double** priemernyPercentualnyPoklesCeny(String značka) – pre danú značku aut spočíta priemerný percentuálny pokles predajnej ceny voči predpokladanej cene auta. Príklad: Predpokladaná cena auta je 5000€ a autobazár ho predá za 4500€ - to čini 10% pokles ceny auta (6 bodov).
- **public** List<Auto> vytvorPonuku() – vytvorí ponuku áut, pričom pre každý pár značka a model vyberieme z predávaných áut ľubovoľného reprezentanta (4 body), resp. ako reprezentanta vyberie najnovšie auto vzhľadom na rok výroby (+3 body).
- **public** Auto najdlhsieNepredaneAuto() – vráti auto v ponuke, ktoré stojí v autobazári najdlhšie (6 bodov).
- **public** List<String> topNajziadanejsieAuta(**int** n, **int** rok) – vráti referenciu na zoznam reťazcov (pozostávajúcich zo značky a modelu auta) n áut, ktoré boli najžiadanejšie (najviac predané) za daný rok (7 bodov).
- **public int** vyplatenePeniaze(Auto auto) – vráti sumu, ktorá bola vyplatená majiteľovi auta. Táto suma pozostáva z predajnej ceny auta, z ktorej je odpočítaná provízia (podľa úlohy vratZinkasovanuProviziu) a cena parkovného. Cena parkovného je 30€/mesiac, ale prvé 2 mesiace auto parkuje zadarmo. Pre jednoduchosť parkovné nezáleží na dni v mesiaci a platí sa za každý začatý mesiac (10 bodov).

Triedenie a komparátor (dokopy 4 body):

Vytvorte triedu CenovyKomparator implementujúcu `java.util.Comparator<Auto>` s metódou (2 body):

- **public int** compare(Auto auto1, Auto auto2) – porovná záznamy áut podľa ich ceny, pričom v prípade rovnakej ceny preferuje auto s menším kilometrovým nájazdom.

V triede Autobazar implementujte inštančnú metódu (2 body):

- **public** List<Auto> vratAutaVPredajiPodlaCeny() – vráti zoznam áut v predaji usporiadaný podľa ceny (počnúc najlacnejším), pričom v prípade rovnakej ceny preferujeme autá s menším nájazdom.

Výnimky (4 body)

Vytvorte nekontrolovanú výnimku `NeplatnaSPZException` a vhodne ju použite aspoň v jednej metóde. Neprázdny reťazec predstavuje platnú ŠPZ, ak spĺňa predpísaný formát.