

OOP – I kolokvijum

Paket izuzeci

Tip **NemaNaMenijuException** koji je izuzetak

- Konstruktor prima referencu na restoran (Restoran) i naziv stavke (String).
- Poruka izuzetka u formatu: Stavka '{nazivStavke}' nije na meniju restorana '{nazivRestorana}'.

Tip **NedovoljnoSredstavaException** koji je izuzetak.

- Konstruktor prima referencu na kupca (Kupac) i ukupnu cenu porudžbine (double).
- Poruka izuzetka u formatu: Kupac {ime} nema dovoljno sredstava. Potrebno: {cena}, raspoloživo: {budzet}.

Paket popust

Interfejs **IPopust** koji ima sledeće ponašanje:

- Javnu metodu `dajPopust()` koja vraća popust koji je ostvaren.

Paket meni

Apstraktni tip **StavkaMenija** koji ima sledeća stanja i ponašanja:

- Privatnu String promenljivu `naziv`.
- Privatnu double promenljivu `osnovnaCena`.
- Konstruktor koji prima naziv i osnovnaCena.
- Getter za naziv i osnovnu cenu
- Javnu apstraktnu metodu `dajCenu()` koja vraća double.
- `equals(Object o)` metodu: Dve stavke su jednake ako imaju isti naziv. Poredi se samo sa drugim StavkaMenija objektima (u suprotnom vratiti false).
- `toString()` metodu koja vraća: `[naziv - osnovnaCena]`

Tip **Hrana** koji je StavkaMenija i ima sledeća stanja i ponašanja:

- Nasleđena osnovnaCena predstavlja cenu po 100 grama.
- Privatnu int promenljivu `gramaza` (u gramima).

- Konstruktor koji prima naziv, cenu za 100g i gramazu.
- Getter za gramazu
- metodu `dajCenu()` koja računa cenu proporcionalno gramaži
- `toString()` metodu u formatu: [Hrana: naziv - cenaPo100g din/100g, gramaza]

Tip **Pice** koji je StavkaMenija i ima sledeća stanja i ponašanja:

- Nasleđena osnovnaCena predstavlja cenu po litru.
- Privatnu `int` promenljivu zapremina (u mililitrima).
- Konstruktor koji prima naziv, cenu po litru i zapreminu.
- Metodu `dajCenu()` koja računa cenu proporcionalno zapremini
- `toString()` metodu u formatu: [Pice: naziv - cenaPoLitru din/L, zapremina ml]

Tip **HranaAkcija** koji je Hrana sa popustom i ima sledeća stanja i ponašanja:

- javno polje `popust` koje predstavlja popust u procentima (0 – 100)
- Konstruktor koji prima naziv, cenu po 100g, gramazu i popust
- Metodu `dajCenu()` koja računa cenu sa popustom
- Metod `dajPopust` koji vraća koliki je ostvareni popust (trebalo da bude 100din, zbog popusta je 90din, metoda vraća 10din).
- `toString()` metodu u formatu: [Hrana: naziv – cenaSaPopustom din/100g, gramaza] - `popust:{popust}`

Paket **kupci**

Tip **Kupac** koji ima sledeća stanja i ponašanja:

- Privatnu `String` promenljivu ime.
- Privatnu `double` promenljivu budzet.
- Privatnu promenljivu korpa koja predstavlja listu stavki menija.
- Konstruktor koji prima ime i budzet. Inicijalizuje korpu kao praznu listu.
- Gettere za ime, budzet i korpa.
- Javnu metodu `dodajUKorpu(Restoran restoran, String nazivStavke)`:
 - Pokušava da pronađe stavku sa datim nazivom u meniju restorana.
 - Ako restoran nema stavku, baca `NemaNaMenijuException`.

- Ako je pronađe, dodaje je u korpu.
- Javnu metodu `dajVrednostKorpe()` koja računa ukupno cenu korpe
- Javnu metodu `isprazni korpu` koja prazni korpu kupca (brise sve reference iz liste)
- Setter za budžet (potreban za ažuriranje stanja nakon kupovine).
- `toString()` metodu u formatu: Kupac: {ime}, Budžet: {budzet}

Tip **StudentKupac** koji nasleđuje **Kupac** i implementira **IPopust**:

- Staticku vrednost popusta od 10%
- Konstruktor koji prima ime i budzet
- Javnu metodu `dajVrednostKorpe()` koja računa ukupnu cenu korpe sa popustom (popust se obracunava za sve artikle koji nisu na akciji).
- metodu `dajPopust()` koja vraća ukupno ostvareni popust. Racunati samo popust koji je student ostvario, ignorisati popuste koji ostvaren na proizvodima na akciji
- Prepisano `toString()` metodu u formatu: Student: {ime}, Budžet: {budzet}

Paket restoran

Tip **Restoran** koji ima sledeća stanja i ponašanja:

- Privatnu String promenljivu naziv.
- Privatnu promenljivu meni koja predstavlja listu `StavkiMenija`.
- Konstruktor koji prima naziv. Inicijalizuje meni kao praznu listu.
- Javnu metodu `dodajNaMeni(StavkaMenija s)` koja dodaje stavku na meni.
- Javnu metodu `nadjiStavku(String naziv)` koja vraća referencu na stavku iz menija koja ima isti naziv, ili null ako ne postoji.
- `toString()` metodu koja ispisuje naziv restorana i sve stavke menija, svaku u novom redu.
- Getter za naziv (zbog ispisa u gresci)

Paket porudzbine

Nabrojivi tip (enum) **StatusPorudzbine**:

- KREIRANA, PRIHVACENA, OTKAZANA

Tip **Porudzbina** koji ima sledeća stanja i ponašanja:

- Privatnu int promenljivu id.
- Privatnu Kupac referencu.
- Privatnu double promenljivu ukupnaCena koja predstavlja ukupnu cenu porudzbine;
- Privatnu StatusPorudzbine promenljivu status (podrazumevano KREIRANA).
- Konstruktor koji prima kupca. Automatski dodeljuje jedinstveni id, postavlja status i ukupnu cenu porudzbine.
- Javnu metodu `dajUkupnuCenu()` koja vraca ukupnu cenu porudzbine
- Javnu metodu `potvrdi()` koja:
 - Proverava da li kupac ima dovoljno budžeta. Ako nema, postavlja status porudzbine na `OTKAZANA` i baca `NedovoljnoSredstavaException`.
 - Ako ima, umanjuje budžet kupca za iznos cene porudzbine i postavlja status porudžbine na `PRIHVACENA`.
 - Uklanja proizvode iz korpe Kupca
- Prepisanu `toString()` metodu u formatu: `[ID: {id} | Kupac: {ime} | Status: {status} | Cena: {ukupnaCena}]`
- Getter za status

Glavna klasa (Main.java)

U main metodi klase `Test` potrebno je:

- Napraviti 1 restoran.
- Dodati na meni restorana bar 5 stavki (npr. 2 hrane, 2 pića i 1 hrana na akciji).
- Ispisati meni restorana.
- Napraviti listu od 3 kupca:
 - 1 StudentKupac.
 - 2 obična Kupac-a (sa različitim budžetima).
- Simulirati kupovinu:
 - Svaki kupac neka doda po 2-3 stavke u korpu.
 - Jednom namerno pokušati dodati stavku koja ne postoji na meniju i obraditi `NemaNaMenijuException`.
- Za svakog kupca:
 - Kreirati porudžbinu.
 - Ispisati detalje porudžbine pre potvrde.
 - Pokušati potvrditi porudžbinu.

- Jednom namerno izazvati `NedovoljnoSredstavaException` (npr. kupac sa malim budžetom) i obraditi ga.
- Za sve uspešno potvrđene porudžbine, ispisati njihov finalni status i ažurirano stanje budžeta kupaca kako bi se video efekat kupovine i popusta.