

*Jesenji semestar, 2021/22*

*PREDMET: SE321* *OBEZBEĐENJE KVALITETA, TESTIRANJE I ODRŽAVANJE SOFTVERA*

Domaći zadatak: **14**

Ime i prezime: **Bogdan Janković**

Broj indeksa: **3920**

Datum izrade: **23.12.2021.**

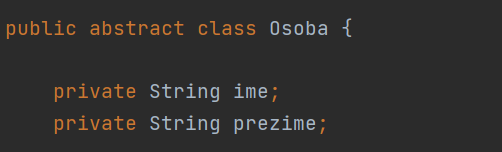
**Tekst domaćeg zadatka:** Na kod odabrane aplikaciju koju ste radili na nekom od predmeta koje ste prethodno slušali i položili primeniti neki od pristupa za postizanje višekratne upotrebljivosti:

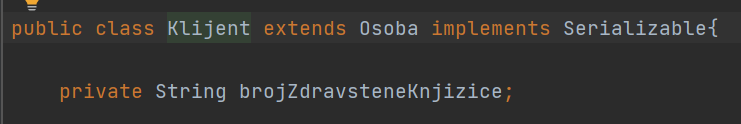
• biblioteke klasa

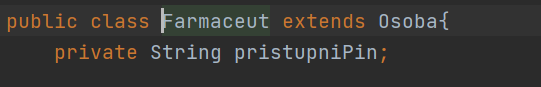
• uzorke dizajna

• rejuzibilnost klasa (npr. primenom nasleđivanja)

Primer rejuzabilnosti klasa primenom nasleđivanja:







Generalizacija je tehnika koja se koristi da bi se na jednostavan način rešavale složene

situacije. Uopštene klase sadrže atribute i operacije (metode) koje imaju druge klase, koje

su onda podklase uopštene klase. Podklase nasleđuju atribute i metode svojih klasa-roditelja

(nad klasa) te se na taj način pojednostavljuje opis klasa sistema, jer su česti slučajevi kada

više klasa imaju iste atribute i metode (operacije).

Generalizacija omogućava da se zajedničke informacije (atributi i metode) definišu i

održavaju samo na jednom mestu (klasi roditelju). Kada se treba izvršiti neka promena, onda

se promena obavi samo na jednom mestu, umesto da se vrši u raznim klasama. U Javi

(i drugim OO jezicima), generalizacija se primenjuje mehanizmom nasleđivanja koji je deo

jezika.

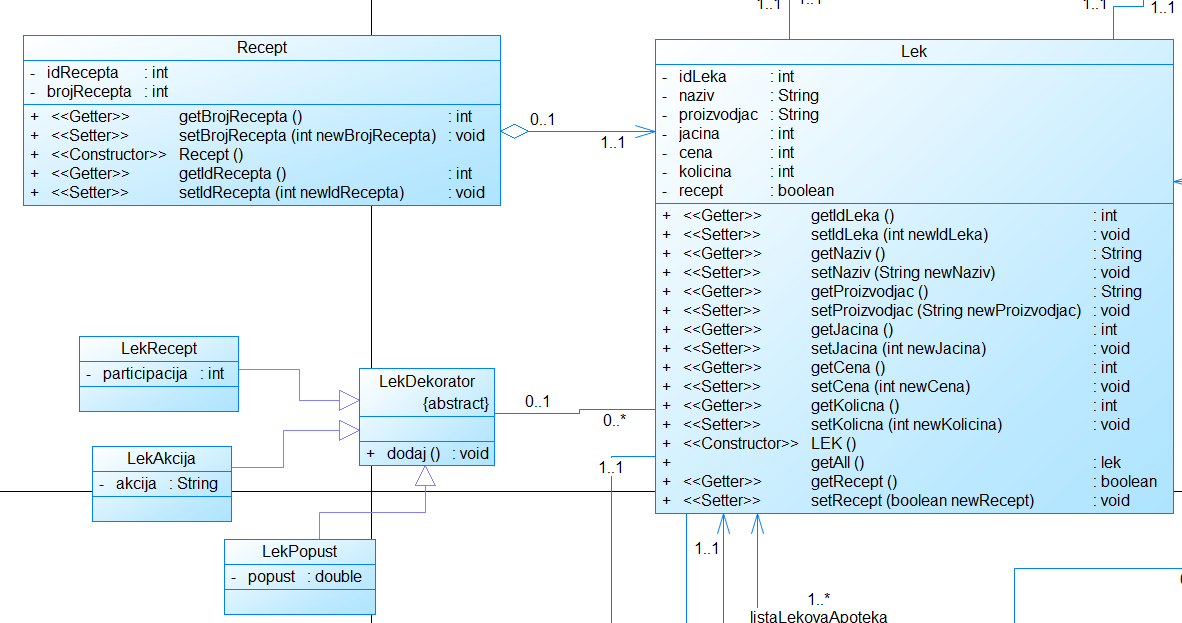
Primer primene design pattern-a:

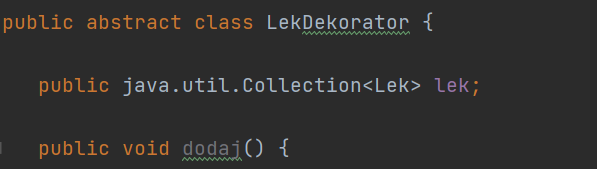
***Decorator šablon*** dozvoljava dodavanje nove funkcionalnosti postojećeg objekta bez menjanja njegove structure.

Decorator šablon dozvoljava dodavanje nove funkcionalnosti postojećeg objekta bez menjanja njegove strukture.Ovaj šablon kreira Decorator klasu omotač originalne klase i obezbeđuje dodatnu funkcionalnost bez promena potpisa metoda klase

Donosi dinamičku dodatnu odgovornost nekom objektu.Decorator objekti obezbeđuju fleksibilnu alternativu korišćenju potklasa za proširenje funkcionalnosti.

Nekada želimo da dodamo funkcionalnost samo nekim objektima, a ne celoj klasi (npr. GUI). Jedan način za ovo je primena nasleđivanja. To je nefleksibilno i statički, jer svi objekti jedne klase to dobijaju.. Drugi fleksibilniji način je da se objekat omota drugom frugim (Decorator) koja dodaje potrebnu funkcionalnost .





Decorator šablon odnosno njegova primena u ovom projektu ima više prednosti:

1. Veća fleksibilnost nego statičko nasleđivanje. Mogu se dodavati i uklanjati u vreme izvršenja. Svojstva se mogu i dva puta dodavati inkrementalno kao i dodavanje odgovornosti primenom Decorator objekata u fazi izvršenja.
2. Decorator i njegova komponenta nisu identični, te ne zavisie od identiteta objekta.
3. Mnogo malih objekata. Razlikujiu se samo sa malim inkrementalnim dodacima.