

1. Prva laboratorijska vježba

1.1. Klase i objekti u Javi

Svrha laboratorijske vježbe je usvajanje osnovnih principa objektno orijentiranog programiranja, kao što su kreiranje vlastitih klasa te kreiranje objekata tih klasa. Osim toga se ističu principi postavljanja i dohvaćanja vrijednosti polja (varijabli) unutar klasa, te korištenje različitih modifikatora za ograničavanje pristupa tim poljima.

1.2. Zadatak za pripremu

Napisati program koji će se sastojati od nekoliko klasa spremljenih u različitim paketima. Klase simuliraju strukturu za implementaciju općenitog CRM sustava. Potrebno je kreirati sljedeće klase i pakete prema koracima:

1. Unutar Eclipse razvojnog okruženja potrebno je kreirati Java projekt koji se naziva kao Vaše prezime i sadrži redni broj vježbe (npr. "Horvat-1"). Kreiranje Java projekta opisano je u online tutorialu [1] u poglavlju 8.1.
2. Projekt pretvoriti u „Maven Project“ korištenjem opcije „Configure->Convert to Maven Project“ i na dijalogu koji se prikaže nakon toga potvrditi ponuđene parametre.
3. Kreirati paket **hr.java.vjezbe.entitet**. Kreiranje paketa unutar Eclipse razvojnog okruženja opisano je unutar poglavlja 8.2. Eclipse online tutoriala [1].
4. Kreirati klasu **Klijent** unutar paketa **hr.java.vjezbe.entitet** koja se sastoji od šest varijabli: za OIB, prezime, ime, broj telefona, e-mail adresa i datum rođenja (koji može biti tipa „LocalDate“, a ostale „String“ tipa). Napisati konstruktor za klasu **Klijent** tako da prima svih šest parametara i sprema ih u privatne varijable klase. Ograničiti pristup varijablama na način da ih je moguće dohvaćati samo korištenjem javnih "get" metoda i spremanje samo korištenjem javnih „set“ metoda. Kreiranje klase unutar Eclipse razvojnog okruženja opisano je unutar poglavlja 9.4. Eclipse online tutoriala [1], automatsko generiranje "get" i "set" metoda unutar razvojnog okruženja Eclipse prikazano je unutar poglavlja 15.
5. Kreirati klasu **Zaposlenik** unutar paketa **hr.java.vjezbe.entitet** koja se sastoji od četiri varijable: korisničko ime, ime, prezime i šifra zaposlenika. Sve varijable mogu biti „String“ tipa. Napisati konstruktor za klasu **Zaposlenik** tako da prima sva četiri parametra i sprema ih u privatne varijable klase. Ograničiti pristup varijablama na način da ih je moguće dohvaćati samo korištenjem javnih "get" metoda i spremanje samo korištenjem javnih „set“ metoda.
6. Kreirati klasu **Komunikacija** unutar paketa **hr.java.vjezbe.entitet** koja se sastoji od pet varijabli: objekta klase **Klijent**, objekta klase **Zaposlenik**, oznake vrste komunikacije, sadržaj komunikacije (posljednje dvije varijable moraju biti tipa „String“) i vrijeme komunikacije (koristiti „LocalDateTime“ tip podatka). Napisati konstruktor za klasu **Komunikacija** tako da prima svih pet parametara i sprema ih u privatne varijable klase. Ograničiti pristup varijablama na način da ih je moguće

dohvaćati samo korištenjem javnih "get" metoda i spremanje samo korištenjem javnih „set“ metoda.

7. Kreirati klasu **Usluga** unutar paketa **hr.java.vjezbe.entitet** koja se sastoji od sedam varijabli: objekta klase **Klijent**, oznake vrste usluge (String), oznake opisa usluge (String), datum usluge (LocalDate) i cijene usluge (BigDecimal) te dva statusa usluge (po pitanju je li usluga obavljena i naplaćena) koji mogu biti tipa „Boolean“. Inicijalne vrijednost „Boolean“ varijabli trebaju biti postavljene na „false“. Napisati konstruktor za klasu **Usluga** tako da prima svih sedam parametara i sprema ih u privatne varijable klase. Ograničiti pristup varijablama na način da ih je moguće dohvaćati samo korištenjem javnih "get" metoda i spremanje samo korištenjem javnih „set“ metoda.
8. Kreirati klasu **Alarm** unutar paketa **hr.java.vjezbe.entitet** koja se sastoji od četiri varijable: objekta klase **Klijent**, opis alarma (String), vrijeme alarma (LocalDateTime) o status alarma, je li aktivan ili ne (Boolean). Napisati konstruktor za klasu **Alarm** tako da prima sva četiri parametra i sprema ih u privatne varijable klase. Ograničiti pristup varijablama na način da ih je moguće dohvaćati samo korištenjem javnih "get" metoda i spremanje samo korištenjem javnih „set“ metoda.
9. Kreirati klasu **Tvrtka** unutar paketa **hr.java.vjezbe.entitet** koja se sastoji od sedam objekata: naziv tvrtke (String), OIB tvrtke (String), polja objekata klase **Klijent**, polja objekata klase **Zaposlenik**, polja objekata klase **Komunikacija**, polja objekata klase **Usluga** i polja objekata klase **Alarm**. Napisati konstruktor za klasu **Tvrtka** tako da prima samo naziv i OIB sprema ih u privatne varijable klase. Ograničiti pristup varijablama na način da ih je moguće dohvaćati samo korištenjem javnih "get" metoda i spremanje samo korištenjem javnih „set“ metoda.
10. Kreirati paket **hr.java.vjezbe.glavna**.
11. Unutar paketa **hr.java.vjezbe.glavna** kreirati klasu **Glavna** koja će se sastojati od **main** metode. Unutar nje potrebno je napisati programski kod za unošenje podataka za po dva objekta klase **Klijent**, **Zaposlenik** i jednog objekta klase **Tvrtka** te dodati klijente i zaposlenike u tvrtku. Za svaki objekt je potrebno od korisnika zatražiti unos svih parametara pojedine komponente i korištenjem konstruktora kreirati pripadajuće konstruktore. Unos podataka za pojedine objekte potrebno je izdvojiti u zasebne metode koje će se pozivati iz **main** metode, nije dozvoljeno sve imati unutar **main** metode. Svaka od tih metoda mora vraćati objekt kojeg popunjava, a primati objekt klase **Scanner**. Dozvoljeno je koristiti samo jedan objekt klase **Scanner**.
12. Unutar metode **main** klase **Glavna** potrebno je ispisati sve informacije o kreiranim objektima. Kod ispisa je informacije potrebno dohvaćati korištenjem „get“ metoda iz objekta klase **Tvrtka**.

Primjer izvođenja programa:

```
UNOS PODATAKA:
UNESITE 1. KLIJENTA:
Unesite OIB klijenta:
12345678909
Unesite prezime klijenta:
Perić
```

```
Unesite ime klijenta:
Pero
Unesite broj telefona klijenta:
098777666
Unesite e-mail adresu klijenta:
pero.peric@crm.hr
Datum rođenja klijenta (dd.MM.yyyy.):
01.01.1990.
UNESITE 2. KLIJENTA:
Unesite OIB klijenta:
12345678908
Unesite prezime klijenta:
Ivić
Unesite ime klijenta:
Ivo
Unesite broj telefona klijenta:
091222333
Unesite e-mail adresu klijenta:
ivo.ivic@crm.hr
Datum rođenja klijenta (dd.MM.yyyy.):
01.01.1992.
UNESITE 1. ZAPOSLENIKA:
Unesite korisničko ime zaposlenika:
ihorvat
Unesite ime zaposlenika:
Ivan
Unesite prezime zaposlenika:
Horvat
Unesite šifru zaposlenika:
ih123
UNESITE 2. ZAPOSLENIKA:
Unesite korisničko ime zaposlenika:
imatic
Unesite ime zaposlenika:
Ivana
Unesite prezime zaposlenika:
Matić
Unesite šifru zaposlenika:
im234
Unesite naziv tvrtke:
CRM d.o.o.
Unesite OIB tvrtke:
12345678907
ISPIS PODATAKA:
Naziv tvrtke:
CRM d.o.o.
OIB tvrtke:
12345678907

OIB klijenta:
12345678909
Prezime klijenta:
Perić
Ime klijenta:
Pero
Broj telefona klijenta:
098777666
E-mail adresu klijenta:
```

```
pero.peric@crm.hr
Datum rođenja klijenta (dd.MM.yyyy.):
1990-01-01
OIB klijenta:
12345678908
Prezime klijenta:
Ivić
Ime klijenta:
Ivo
Broj telefona klijenta:
091222333
E-mail adresu klijenta:
ivo.ivic@crm.hr
Datum rođenja klijenta (dd.MM.yyyy.):
1992-01-01

Korisničko ime zaposlenika:
ihorvat
Ime zaposlenika:
Ivan
Prezime zaposlenika:
Horvat
Unesite šifru zaposlenika:
ih123
Korisničko ime zaposlenika:
imatic
Ime zaposlenika:
Ivana
Prezime zaposlenika:
Matić
Unesite šifru zaposlenika:
im234
```

NAPOMENE:

1. Osim implementacija vježbe prema uputama, dozvoljeno je uvoditi i promjene ako su opravdane i ne narušavaju koncepte objektno-orijentiranog programiranja.
2. Prilikom unosa podataka tipa **BigDecimal** koristiti decimalni zarez, a ne točku.
3. Nakon naredbi za unos numeričkih podataka (**BigDecimal** ili **Integer**) izvršiti metodu „nextLine“ nad objektom klase **Scanner** koji se koristi. Proučiti odgovore na sljedećem linku: <http://stackoverflow.com/questions/13102045/skipping-nextline-after-using-next-nextint-or-other-nextfoo-methods>.
4. U slučaju potrebe isključivanja nekih prečaca u Eclipseu (kao što je kombinacija tipki „Alt Gr + B“ ili „Alt Gr + F“ koje služe za ispisivanje zagrada) koristiti upute sa sljedećem linku: <http://stackoverflow.com/questions/12477738/how-to-disable-ctrl-alt-m-in-eclipse>.
5. Za pretvorbu objekta tipa „String“ u objekt tipa „LocalDateTime“ koristiti upute na sljedećem linku: <http://stackoverflow.com/questions/22463062/how-to-parse-format-dates-with-localdatetime-java-8>.

LITERATURA:

1. Eclipse online tutorial: <http://www.vogella.com/tutorials/Eclipse/article.html>