

## 2. Druga laboratorijska vježba

### 2.1. TEMA VJEŽBE

Svrha laboratorijske vježbe je korištenje koncepata objektno orijentiranog programiranja u programskom jeziku Java, uz korištenje podklasa i nadklasa kod nasljeđivanja, korištenje modifikatora „protected“ i „super“, konstruktora u podklasama, nadjačavanje metoda, polimorfizma, apstraktnog metoda i sučelja te novih funkcionalnosti u programskom jeziku Java kao što su zapisi (engl. *records*) te zapečaćenih (engl. *sealed*) sučelja i klasa.

### 2.2. ZADATAK ZA PRIPREMU

Proširiti rješenje prve laboratorijske vježbe na način da se kopira rješenje prethodne laboratorijske vježbe te preimenuje u naziv koji sadrži indeks „2“, umjesto „1“. Osim same mape s projektom, potrebno je promijeniti i naziv projekta unutar IntelliJ-a korištenjem opcije „Refactor->Rename“. Program je potrebno proširiti na sljedeći način:

1. U paket „hr.java.restaurant.model“ dodati novu apstraktnu klasu „Entity“ koja sadrži parametar „Long id“. Sve ostale klase u projektu moraju nasljeđivati tu klasu. Potrebno je generirati konstruktor koji prima sve ulazne parametre, te „getter“ i „setter“ metode za te varijable.
2. U paket „hr.java.restaurant.model“ dodati novu apstraktnu klasu „Person“ koja sadrži dva objekta tipa „String“: „firstName“ i „lastName“. Potrebno je generirati konstruktor koji prima sve ulazne parametre, te „getter“ i „setter“ metode za te varijable. Tu klasu moraju nasljeđivati klase „Chef“, „Waiter“ i „Deliverer“.
3. U paket „hr.java.restaurant.model“ dodati novu klasu „Contract“ koja će sadržavati podatke o ugovoru koji imaju kuhari, konobari i dostavljači. Klasa „Contract“ mora sadržavati varijablu „salary“ (tipa „BigDecimal“), „startDate“ i „endDate“ tipa „LocalDate“ te vrstu ugovora koja poprima jednu od konstantnih vrijednosti koje se moraju nalaziti u klasi „Contract“: „FULL\_TIME“ ili „PART\_TIME“.
4. Klase „Chef“, „Waiter“ i „Deliverer“ moraju umjesto „Salary“ objekta sadržavati „Contract“ objekt te ga imati kao ulazni parametar konstruktora. Osim toga je potrebno generirati „getter“ i „setter“ metode za objekt klase „Contract“.
5. Za klase „Chef“, „Waiter“, „Deliverer“, „Address“ i „Category“ potrebno je koristiti „Builder pattern“ za kreiranje objekata. Objekte tih klasa potrebno je isključivo kreirati korištenjem „Builder patterna“.

6. U paket „hr.java.restaurant.model“ dodati zapis (engl. *record*) pod nazivom „Bonus“ koji predstavlja iznos bonusa na osnovnu plaću za sve klase tipa „Chef“, „Waiter“ i „Deliverer“. Taj objekt je potrebno dodati kao ulazni parametar konstruktora tih klasa te generirati „getter“ i „setter“ metode za njega.
7. U paket „hr.java.restaurant.model“ dodati zapečaćeno sučelje (engl. *sealed interface*) „Vegan“ koje označava vegansko jelo i sadržava dvije metode po želji koje su karakteristične za tu vrstu jela. Kreirati jednu klasu po želji koja nasljeđuje klasu „Meal“ i implementira sučelje „Vegan“ te joj dodati jednu varijablu koju nemaju ostala jela te po želji implementirati metode iz sučelja „Vegan“.
8. U paket „hr.java.restaurant.model“ dodati zapečaćeno sučelje (engl. *sealed interface*) „Vegetarian“ koje označava vegetarijansko jelo i sadržava dvije metode po želji koje su karakteristične za tu vrstu jela. Kreirati jednu klasu po želji koja nasljeđuje klasu „Meal“ i implementira sučelje „Vegetarian“ te joj dodati jednu varijablu koju nemaju ostala jela te po želji implementirati metode iz sučelja „Vegetarian“.
9. U paket „hr.java.restaurant.model“ dodati zapečaćeno sučelje (engl. *sealed interface*) „Meat“ koje označava mesno jelo i sadržava dvije metode po želji koje su karakteristične za tu vrstu jela. Kreirati jednu klasu po želji koja nasljeđuje klasu „Meal“ i implementira sučelje „Meat“ te joj dodati jednu varijablu koju nemaju ostala jela te po želji implementirati metode iz sučelja „Meat“.
10. Metodu „main“ u klasi „Main“ proširiti na način da se unose dodatni podaci za svakog od objekta klasa „Chef“, „Waiter“ i „Deliverer“ o ugovorima. Sve objekte navedenih klasa potrebno je smjestiti u isto polje te pronaći odrediti sljedeće zaposlenike:
  - a. Zaposlenika s najvećom plaćom
  - b. Zaposlenika s najdužim ugovorom (koji je najranije započeo)Za pronađene podatke o zaposlenicima, uključujući podatke o imenu, prezimenu, trajanju ugovora i plaći. U slučaju neispravnog unosa podatka potrebno je ispisati poruku o pogrešci te zatražiti unos ispravnog podatka.
11. Kreirati po tri vrste jela iz kategorije veganskog, vegetarijanskog i mesnog te ih također smjestiti u isto polje. U tom polju potrebno je pronaći i ispisati podatke o jelima koja imaju najveći i najmanji broj kilokalorija. Za ta pronađena jela potrebno je ispisati sve podatke, uključujući i informacije pohranjene u varijable karakteristične za svaku od jela. U slučaju neispravnog unosa podatka potrebno je ispisati poruku o pogrešci te zatražiti unos ispravnog podatka.
12. Ako je potrebno, doraditi klase na način da nijedna nema više od 200 linija koda.

### **NAPOMENE:**

1. Osim implementacija vježbe prema uputama, dozvoljeno je uvoditi i promjene ako su opravdane i ne narušavaju koncepte objektno-orijentiranog programiranja.