

6.Šesta laboratorijska vježba

6.1. TEMA VJEŽBE

Svrha laboratorijske vježbe je korištenje tekstualnih i binarnih datoteka za čitanje podataka iz njih te zapisivanje podataka u njih, kao i serijaliziranje i deserijaliziranje objekata korištenih u aplikaciji.

6.2. ZADATAK ZA PRIPREMU

Proširiti rješenje pete laboratorijske vježbe na način da se kopira rješenje te preimenuje u naziv koji sadrži indeks „6“, umjesto „5“. 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. Klasu „NamedEntity“ iz paketa „hr.java.production.model“ proširiti novom varijablom „id“ tipa „Long“. Proširiti konstruktor kako bi primao i varijablu „id“ te za tu varijablu kreirati „get“ i „set“ metode. S parametrom „id“ potrebno je proširiti i konstruktore svih ostalih entiteta u aplikaciji koji nasljeđuju klasu „NamedEntity“.
2. Kreirati mapu „dat“ na razini projekta na istoj razini kao što je kreirana mapa „logs“.
3. Unutar mape „dat“ kreirati datoteku „categories.txt“ koja će sadržavati informacije o kategorijama formatiranih na sljedeći način, na primjer:

```
1
Food
In this category you can find all items related to food.
2
Technical equipment
In this category you can find all items from technical area.
3
```

Literature

In this category you can find books, comics etc.

Svaki zapis o kategoriji mora sadržavati tri varijable: „id“, „name“, „description“. Nakon što se pročitaju podaci iz datoteke, moraju se kreirati odgovarajući objekti klase „Category“ i koristiti u ostatku aplikacije.

Na navedenom primjeru prikazana su tri skupa podataka od po tri retka za tri kategorije označene s tri različite boje.

Na sličan način je potrebno kreirati datoteke koje će sadržavati zapise i za sve ostale entitete korištene u aplikaciji: „items.txt“, „addresses.txt“, „factories.txt“ i „stores.txt“.

U slučajevima kad je npr. potrebno povezivati artikle s tvornicama i dućanima, to je potrebno učiniti na način da se u jednu liniju postave svi identifikatori artikala odvojeni zarezom, npr. „1,2,3“. U slučaju tvornice, to može izgledati ovako:

```
1
Omnifactory
2
3,4,5,6,7,8
```

Navedeni primjer bi označavao da tvornica „Omnifactory“ ima identifikator „1“, identifikator adrese „2“, te ima tri šest artikala s identifikatorima „3“, „4“, „5“, „6“, „7“ i „8“.

4. Nakon pokretanja programa moraju se učitati podaci o objektima iz svih datoteka te izvesti ostatak programa (koji se u petoj vježbi izvršavao nakon uspješnog unošenja podataka o objektima).
5. U svim klasama koje predstavljaju entitete u aplikaciji implementirati sučelje koje omogućava serijalizaciju objekata.
6. Na kraju programa serijalizirati sve objekte klase „Factory“ i „Store“ koja imaju barem pet artikala.

NAPOMENE:

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