

## Zadaća 3 – Turistička agencija

**Naziv projekta:** {LDAP\_korisničko\_ime}\_zadaca\_3

**Ishodišni direktorij projekta:** {LDAP\_korisničko\_ime}\_zadaca\_3

**Naziv rješenja:** {LDAP\_korisničko\_ime}\_zadaca\_3.zip

Prije predavanja projekta potrebno je napraviti Clean na projektu (obrisati sve pomoćne i izvršne datoteke kao što su .class, .jar, .txt, .csv). Zatim cijeli projekt sažeti u .zip (NE .rar) format s nazivom {LDAP\_korisničko\_ime}\_zadaca\_3.zip i predati u Moodle. Uključiti izvorni kod, popunjenu datoteku dokumentacije (u ishodišnom direktoriju projekta). Ne uključuju se datoteke s podacima. Potrebno je podesiti konfiguraciju projekta (maven) tako da se kreira Java izvršna .jar datoteka.

Rješenje treba sadržavati roditeljski projekt {LDAP\_korisničko\_ime}\_zadaca\_3 u kojem je Group ID: edu.unizg.foi.uzdiz.{LDAP\_korisničko\_ime}, Artifact ID: {LDAP\_korisničko\_ime}\_zadaca\_3, a Version: 2.0.0. Projekt sadrži dva modula s nazivima:

- {LDAP\_korisničko\_ime}\_zadaca\_3\_lib (Group Id i Version su kao i kod roditeljskog projekta, a Artifact ID odgovara nazivu modula). Ovaj modul predstavlja podsustav za učitavanje i provjeru podataka iz datoteka te treba biti realiziran uz pomoć uzorka dizajna Facade. Njegove klase ne smiju se koristiti u drugom modulu osim klase Facade. Njegove klase ne smiju koristiti klase iz drugog modula. Metode vraćaju samo ispravne podatke. Ovaj modula radi samo sintaktičku provjeru podataka (potreban broj elemenata, ispravnost tipa podatka i sl).
- {LDAP\_korisničko\_ime}\_zadaca\_3\_app (Group Id i Version su kao i kod roditeljskog projekta, a Artifact ID odgovara nazivu modula). Ovaj modul kreira aranžmane i rezervacije na temelju vraćenih podataka metoda klase koja predstavlja uzorak dizajna Facade iz prvog modula. Ovaj modul radi semantičku provjeru podataka (postoji li aranžman za rezervaciju i sl). Potrebno je definirati ovisnost na modul {LDAP\_korisničko\_ime}\_zadaca\_3\_lib.

Sve klase iz oba modula trebaju biti smještene unutar strukture paketa edu.unizg.foi.uzdiz.{LDAP\_korisničko\_ime} uz logičku organizaciju vlastitih paketa.



Struktura .zip datoteke predane zadaće treba biti sljedeća:

```
{LDAP_korisnik}_zadaca_3
    {LDAP_korisnik}_zadaca_3.pdf
    {LDAP_korisnik}_zadaca_3_app
        pom.xml
        src
            main
            ...
    {LDAP_korisnik}_zadaca_3_lib
        pom.xml
        src
            main
            ...
pom.xml
```

## Uvod

Turistička agencija planira, sastavlja i prodaje turističke aranžmane (npr. putovanja, izlete i sl). Turistički aranžmani uključuju usluge kao što su: prijevoz, smještaj, prehrana i razgledavanje. Nije nužno da jedan aranžman sadrži sve spomenute usluge. Svaki aranžman ima: oznaku, naziv, program, početni datum, završni datum, vrijeme kretanja, vrijeme povratka, cijenu, min broj putnika, maks broj putnika, broj noćenja, doplata za jednokrevetnu sobu, prijevoz, broj doručka, broj ručkova, broj večera.

Turistička agencija prikuplja rezervacije za svoje turističke aranžmane te ih obrađuje prema datumu i vremenu prijema rezervacije. Svaka rezervacija ima: ime osobe, prezime osobe, oznaku turističkog aranžmana, datum i vrijeme prijema rezervacije.

Upravljanje dozvoljenim brojem rezervacija provodi se na jedan od dva moguća načina, koji se određuju opcijama:

- jdr - jedna osoba u jednom trenutku može imati samo jednu aktivnu rezervaciju turističkog aranžmana odnosno jedna osoba ne može imati aktivne rezervacije za dva ili više turistička aranžmana koji se preklapaju u svojim datumima i vremenima. U tom slučaju kronološki najranija rezervacija ima status aktivne rezervacije, a sve njene ostale rezervacije koje su u presjeku datuma i vremena mijenjaju status u odgođenu rezervaciju.
- vdr - jedna osoba u jednom trenutku može imati više rezervacija turističkog aranžmana ali ne više od  $\frac{1}{4}$  maksimalnog broja putnika odnosno jedna osoba može imati aktivne rezervacije za dva ili više turistička aranžmana koji se preklapaju u svojim datumima i vremenima.

Upravljanje dozvoljenim brojem rezervacija treba biti implementirano putem uzorka dizajna Strategy. Ukoliko kod izvršavanja programa nije pridružena ni jedna od dviju gornjih opcija, tada se primjenjuje uzorak dizajna Null Object.

Pojedina ispravna rezervacija može biti tj. imati status:

- **nova** (inicijalni status, kada rezervacija još nije došla na obradu)
- **primljena** (kada broj prijava (rezervacija) nije dostigao minimalni broj putnika na turističkom aranžmanu)
- **aktivna** (kada je broj prijava (rezervacija) dostigao minimalni broj putnika i nije prešao maksimalni broj putnika na turističkom aranžmanu)
- **na čekanju** (kada je broj prijava (rezervacija) prešao maksimalni broj putnika na turističkom aranžmanu)
- **odgođena** (kod opcije --jdr kada osoba ima drugu kronološki raniju aktivnu rezervaciju u presjeku datum i vremena aranžmana s aranžmanom te rezervacije; kod opcije --vdr kada broj rezervacija jedne osobe pređe  $\frac{1}{4}$  maksimalnog broja putnika)
- **otkazana** (kada stigne komanda za njeno otkazivanje).

Uzmimo u razmatranje jedan turistički aranžman. Potrebno je obraditi rezervacije sa statusom nova. Pojedina rezervacija prvo mijenja status u primljena. U slučaju kada je broj rezervacija sa statusom primljena dostigao minimalni broj putnika tada sve primljene rezervacije

mijenjaju status u aktivnu rezervaciju. U slučaju da je broj aktivnih rezervacija došao do maksimalnog broja putnika tada ostale rezervacije mijenjaju status na čekanju.

Što se radi ako osoba odustane (otkaže rezervaciju) koja ima:

1. primljenu rezervaciju - tada ta rezervacija mijenja status u otkazanu rezervaciju.
2. aktivnu rezervaciju - tada ta rezervacija mijenja status u otkazanu rezervaciju, uzima se prva ispravna rezervacija na čekanju za taj aranžman (ako postoji) te ona mijenja status u aktivnu rezervaciju.
3. aktivnu rezervaciju i odgođenu/e rezervaciju/e - tada ta rezervacija mijenja status u otkazanu rezervaciju, a njena sljedeća kronološki najranija odgođena rezervacija može promijeniti status u aktivnu rezervaciju ukoliko ispunjava kronološki uvjet unutar ostalih rezervacija svog aranžmana.

Ako se mijenja broj primljenih ili aktivnih rezervacija jednog aranžmana zbog dodavanja i/ili otkazivanja rezervacija tada se može desiti promjena statusa ostalih rezervacija tog aranžmana. Generalno, promjena statusa jedne rezervacije može utjecati na ostale rezervacije te osobe (na istom i ostalim aranžmanima) i na ostale rezervacije na aranžmanu te rezervacije.

Pojedini ispravni aranžman može biti tj. imati status:

- ***u pripremi*** (kada broj prijava (rezervacija) nije dostigao minimalni broj putnika na turističkom aranžmanu)
- ***aktivvan*** (kada je broj prijava (rezervacija) dostigao minimalni broj putnika i nije prešao maksimalni broj putnika na turističkom aranžmanu)
- ***popunjeno*** (kada je broj prijava (rezervacija) prešao maksimalni broj putnika na turističkom aranžmanu)
- ***otkazan*** (kada stigne komanda za njegovo otkazivanje).

Za rad sa statusima rezervacija i statusima aranžmana treba koristiti utorak dizajna State. Aranžmani i rezervacije trebaju biti implementirani putem uzorka dizajna Composite tako da pojedini aranžman sadrži samo svoje rezervacije. Ne postoje posebni redovi (ili bilo koji drugi oblik kolekcija) za primljene, aktivne, na čekanju, odgođene ili otkazane rezervacije.

Sve datoteke su u UTF-8 kodnom zapisu i koriste csv (en. Comma Separated Values) format zapisa u kojem se koristi znak , za odvajanje vrijednosti pojedinih atributa/stupaca u jednom retku. U slučaju većeg broja znakova u jednom atributu ili se pojavljuje znak , u sadržaju atributa, tada se sadržaj atributa stavlja unutar znakova " i ". U svakoj datoteci prvi redak sadrži popis atributa koji se mogu nalaziti u ostalim redcima, on je informativan i preskače se kod učitavanja podataka. Ukoliko u datoteci postoji potpuno prazan redak to nije pogreška i samo ga treba preskočiti tijekom učitavanja podataka. Uloga tog retka je da vizualno odvoji retke koji čine jednu cjelinu kako bi se postigla bolja preglednost podataka. Ukoliko u datoteci postoji redak koji započinje znakom # to nije pogreška i samo ga treba preskočiti tijekom učitavanja podataka. Uloga tog retka je da pruža komentar u vezi sljedećeg retka ili više redaka. Ukoliko u datoteci postoji atribut koji ima više vrijednosti tada su one odvojene znakom ;. Ukoliko u datoteci postoji atribut koji nema vrijednost tada je atribut prazan.

**Unutar datoteka svi atributi nisu obavezni!** Npr. kod turističkih aranžmana ako je vlastiti prijevoz tada nema vremena kretanja i povratka. Ako nema noćenja tada nema doplate za jednokrevetnu sobu. Ako ima noćenja može biti da nije predviđena doplata za jednokrevetnu sobu. Ako je vlastiti prijevoz i nema drugih planiranih prijevoza u aranžmanu tada nema prijevoza. Ako nema noćenja obično nema doručka, ručka i večere, ali mogu biti.

U cijeloj zadaći potrebno je koristiti hrvatski format datuma (dd.MM.yyyy. hh:mm:ss). Više o datotekama i njihovim atributima možete pronaći u tablici 1.

Tablica 1. Nazivi datoteka, dani primjeri i dodatne informacije o njihovim atributima

Datoteka	Primjer	Dodatne informacije
Podaci o turističkim aranžmanima	DZ_3_aranzmani.csv	Oznaka, Naziv, Program, Početni datum, Završni datum, Vrijeme kretanja, Vrijeme povratka, Cijena, Min broj putnika, Maks broj putnika, Broj noćenja, Doplata za jednokrevetnu sobu, Prijevoz, Broj doručka, Broj ručkova, Broj večera
Podaci o rezervacijama turističkih aranžmana	DZ_3_rezervacije.csv	Ime, Prezime, Oznaka aranžmana, Datum i vrijeme

Priložene datoteke su samo jedan primjer sa svojim sadržajem. Nastavnici će svoje testiranje provoditi i na drugim datotekama i sadržajima. Neke od njih će se koristiti na prezentacijama zadaća. A druge će se koristiti za testiranje i bodovanje zadaća.

Predlaže se da studenti/ce pripreme svoje dodatne datoteke s kojima će testirati svoje programe. To se može uraditi tako da se priloženim datotekama dodaju novi zapisi, promijene podaci postojećim zapisima i/ili obrišu pojedini zapise. Posebno se predlaže da se pripreme datoteke u kojima će biti osim ispravnih zapisa i zapisi koji nisu ispravni u pojedinom segmentu

(npr. nema prvi informativni redak, ima ili premalo ili previše podataka/atributa u retku, neispravna vrijednost u atributu koji je brojčanog tipa i sl).

### Opis problema

Na početku je potrebno inicijalizirati sustav turističke agencije tako da se učitaju datoteke u pravilnom redoslijedu te se kreiraju potrebni objekti za izvršavanje sustava.

Prilikom svakog učitavanja podataka iz datoteke potrebno je provjeriti ispravnost pojedinog zapisa/retka i ako nije ispravan potrebno ga je preskočiti i ispisati redni broj pogreške, sadržaj zapisa/retka i opis zašto je neispravan. Nakon što se učitaju sve potrebne datoteke potrebno je pripremiti program za izvršavanje komandi u interaktivnom načinu rada. Tijekom rada programa može se izvršiti više komandi sve dok se ne upiše komanda Q.

Svaki ispis podataka počinje ispisom komande (i njenih parametara) koja je pozvana. Ako je zadan ispis u obliku tablice tada se u sljedećem retku ispisuje naziv tablice, slijedi prazan red ili crta u širini tablice, u nastavku se ispisuje zaglavlje tablice (naslovi stupaca), slijedi crta u širini tablice. Sada su došli na red za ispis podaci. U svakom ispisu podataka potrebno je da se tekstualni podaci lijevo poravnavaju, a brojčani podaci se desno poravnavaju uz znak . (točka) za razdvajanje svake 3 znamenke cjelobrojnog dijela te ispis 2 decimalne. Tablice moraju izgledati kao tablice koje sadrže stupce i pregledni ispis podataka. Podaci u tablicama moraju biti ispisani u redoslijedu koji je zadan komandom IS.

Korisniku se daje mogućnost da izvrši sljedeće komande<sup>1</sup> za aktivnosti:

- Pregled turističkih aranžmana uz mogućnost definiranja razdoblja
  - Sintaksa:
    - ITAK [od do]
  - Primjer:
    - ITAK
    - Opis primjera:
      - Ispis podataka s turističkim aranžmanima u obliku tablice bez definiranog razdoblja (Oznaka, Naziv, Početni datum, Završni datum, Vrijeme kretanja, Vrijeme povratka, Cijena, Min broj putnika, Maks broj putnika, status).
    - Primjer:
      - ITAK 1.10.2025. 31.12.2025.
      - Opis primjera:
        - Ispis podataka s turističkim aranžmanima u obliku tablice za zadano razdoblje od 1.10.2025. do 31.12.2025. (Oznaka, Naziv, Početni datum, Završni datum, Vrijeme kretanja, Vrijeme povratka, Cijena, Min broj putnika, Maks broj putnika, status).
- Pregled turističkog aranžmana
  - Sintaksa:
    - ITAP oznaka
  - Primjer:
    - ITAP 1
    - Opis primjera:
      - Ispis podataka za odabrani turistički aranžman (Oznaka, Naziv, Program, Početni datum, Završni datum, Vrijeme kretanja, Vrijeme povratka, Cijena, Min broj putnika, Maks broj putnika, Broj noćenja, Doplata za jednokrevetnu sobu, Prijevoz, Broj doručka, Broj ručkova, Broj večera, status).

---

<sup>1</sup> Komande koje su bile zadane u 1. i 2. zadaći, a nisu realizirane/dovršene/ispravne u 3. zadaći donose negativne bodove (do 50% bodova koje nosi pojedina komanda u 1. i 2. zadaći).

- Pregled rezervacija za turistički aranžman
  - Sintaksa:
    - IRTA oznakaAranžmana [PA|Č|O|OD]
      - PA - primljene i aktivne rezervacije
      - Č - rezervacije na čekanju
      - O - otkazane rezervacije
      - OD - odgođene rezervacije
  - Primjer:
    - IRTA 1 PA
    - Opis primjera:
      - Ispis podataka o primljenim i aktivnim rezervacijama za turistički aranžman u obliku tablice (Ime, Prezime, Datum i vrijeme, Vrsta).
  - Primjer:
    - IRTA 1 Č
    - Opis primjera:
      - Ispis podataka o rezervacijama na čekanju za turistički aranžman u obliku tablice (Ime, Prezime, Datum i vrijeme, Vrsta).
  - Primjer:
    - IRTA 1 PAČO
    - Opis primjera:
      - Ispis podataka o primljenim i aktivnim rezervacijama, rezervacijama na čekanju i otkazanim rezervacijama za turistički aranžman u obliku tablice (Ime, Prezime, Datum i vrijeme, Vrsta, Datum i vrijeme otkaza).
  - Primjer:
    - IRTA 1 ODO
    - Opis primjera:
      - Ispis podataka o odgođenim i otkazanim rezervacijama za turistički aranžman u obliku tablice (Ime, Prezime, Datum i vrijeme, Vrsta, Datum i vrijeme otkaza).
- Pregled rezervacija za osobu
  - Sintaksa:
    - IRO ime prezime
  - Primjer:
    - IRO Lea Novak
    - Opis primjera:
      - Ispis rezervacija osobe Lea Novak u obliku tablice (Datum i vrijeme, Oznaka aranžmana, Naziv aranžmana, Vrsta).

- Otkaz rezervacije za turistički aranžman
  - Sintaksa:
    - ORTA ime prezime oznaka
  - Primjer:
    - ORTA Lea Novak 3
    - Opis primjera:
      - Otkaz rezervacije Lee Novak za turistički aranžman s oznakom 3.
      - Ispisuje se da li je uspješno obavljen otkaz rezervacije
- Dodavanje rezervacije za turistički aranžman
  - Sintaksa:
    - DRTA ime prezime oznaka datum vrijeme
  - Primjer:
    - DRTA Damir Klarić 3 21.10.2025 10:27:41
    - Opis primjera:
      - Dodana rezervacija Damira Klarića za turistički aranžman s oznakom 3 u 21.10.2025 10:27:41.
      - Ispisuje se da li je uspješno obavljeno dodavanje rezervacije
- Otkaz turističkog aranžmana
  - Sintaksa:
    - OTA oznaka
  - Primjer:
    - OTA 3
    - Opis primjera:
      - Otkaz turističkog aranžmana s oznakom 3.
      - Potrebno je otkazati sve rezervacije tog turističkog aranžmana i ažurirati rezervacije osobe čija je rezervacija otkazana
      - Ispisuje se da li je uspješno obavljen otkaz turističkog aranžmana.
- Postavljanje načina sređivanja podataka kod ispisa u obliku tablica
  - Sintaksa:
    - IP [N|S]
      - N - podaci se ispisuju u kronološkom redoslijedu (prvo stari, zatim novi). To je inicijalni način ispisa.
      - S - podaci se ispisuju u obrnutom kronološkom redoslijedu (prvo novi, zatim stari)
  - Primjer:
    - IP N
    - Opis primjera:
      - Postavljanje ispisa u kronološkom redoslijedu.

- Brisanje podataka o aranžmanima ili rezervacijama
  - Sintaksa:
    - BP [A|R]
      - A - aranžmani
      - R - rezervacije
  - Primjer:
    - BP A
    - Opis primjera:
      - Brisanje svih aranžmana (i njihovih rezervacija)
  - Primjer:
    - BP R
    - Opis primjera:
      - Brisanje svih rezervacija.
- Učitavanje podataka o aranžmanima ili rezervacijama iz datoteke
  - Sintaksa:
    - UP [A|R] nazivDatoteke
      - A - aranžmani
      - R - rezervacije
  - Primjer:
    - UP A DZ\_3\_aranzmani\_1.csv
    - Opis primjera:
      - Učitavanje novih aranžmana iz datoteke DZ\_3\_aranzmani\_1.csv.
  - Primjer:
    - UP R DZ\_3\_rezervacije\_1.csv
    - Opis primjera:
      - Učitavanje novih aranžmana iz datoteke DZ\_3\_rezervacije\_1.csv.

- Ispis statističkih podataka za aranžmane uz mogućnost definiranja razdoblja
  - Sintaksa:
    - ITAS [od do]
    - Primjer:
      - ITAS
      - Opis primjera:
        - Ispis statističkih podataka o turističkim aranžmanima u obliku tablice bez definiranog razdoblja (ukupan broj rezervacija, broj aktivnih rezervacija, broj rezervacija na čekanju, broj odgođenih rezervacija, broj otkazanih rezervacija, ukupan prihod aranžmana = broj aktivnih rezervacija \* cijena aranžmana).
    - Primjer:
      - ITAS 1.10.2025. 31.12.2025.
      - Opis primjera:
        - Ispis statističkih podataka o turističkim aranžmanima u obliku tablice za zadano razdoblje od 1.10.2025. do 31.12.2025. (ukupan broj rezervacija, broj aktivnih rezervacija, broj rezervacija na čekanju, broj odgođenih rezervacija, broj otkazanih rezervacija, ukupan prihod aranžmana).

- Pretraživanje podataka u aranžmanima i rezervacijama primjenom uzorka dizajna Visitor
  - Sintaksa:
    - PPTAR [A|R] riječ
      - A - aranžmani
      - R - rezervacije
    - Primjer:
      - PPTAR A skijanje
      - Opis primjera:
        - Ispis naziva svih turističkih aranžmana koji u nazivu i/ili opisu programa sadrže traženu riječ (kao cijelu riječ ili dio riječi). Razlikuju se male i velika slova.
    - Primjer:
      - PPTAR R Lea
      - Opis primjera:
        - Ispis podataka o rezervacijama u kojima se nalazi osoba koja u imenu i/ili prezimenu sadrži riječ. Razlikuju se male i velika slova.
- Pohranjivanje odabranog turističkog aranžmana i njegovih rezervacija u spremište primjenom uzorka dizajna Memento i Command
  - Sintaksa:
    - PSTAR oznaka
    - Primjer:
      - PSTAR 3
      - Opis primjera:
        - Spremanje turističkog aranžmana s oznakom 3 kao i svih njegovih rezervacija.
- Vraćanje iz spremišta zadnje spremlijenog stanja turističkog aranžmana i njegovih rezervacija primjenom uzorka dizajna Memento i Command
  - Sintaksa:
    - VSTAR oznaka
    - Primjer:
      - VSTAR 3
      - Opis primjera:
        - Vraćanje turističkog aranžmana s oznakom 3 kao i svih njegovih rezervacija.

- Preplata za informacije o promjenama turističkog aranžmana i/ili njegovih rezervacija
  - Sintaksa:
    - PTAR ime prezime oznaka
    - Primjer:
      - PTAR Lea Novak 3
      - Opis primjera:
        - Osoba Lea Novak preplatila se za informacije o turističkom aranžmanu s oznakom 3 te njegovim rezervacijama. Kod promjene prvo se prikazuju podaci o osobi, zatim oznaka turističkog aranžmana i na kraju opis nastale promjene na turističkom aranžmanu i/ili njegovim rezervacijama.
  - Uklanjanje preplate za informacije o promjenama turističkog aranžmana i/ili njegovih rezervacija
    - Sintaksa:
      - UPTAR [ime prezime oznaka] | [oznaka]
      - Primjer:
        - UPTAR Lea Novak 3
        - Opis primjera:
          - Uklada se preplata osobe Lea Novak za informacije o turističkom aranžmanu s oznakom 3 te njegovim rezervacijama.
      - Primjer:
        - UPTAR 3
        - Opis primjera:
          - Uklidaju se preplate svih osoba za informacije o turističkom aranžmanu s oznakom 3 te njegovim rezervacijama.
    - Prekid rada programa
      - Sintaksa:
        - Q

Potrebno je dodati vlastitu funkcionalnost (aktivnost/komandu) projektu tako da se koristi GOF uzorak Decorator. To znači da je u dokumentaciju projekta **potrebno dodati opis funkcionalnosti** koja nije zadana u opisu zadaće i koja će se realizirati zadanim GOF uzorkom.

Potrebno je dodati vlastitu funkcionalnost (ne smije biti nova komanda) projektu tako da se koristi GOF uzorak Chain-of-Responsibility. To znači da je u dokumentaciju projekta **potrebno dodati opis funkcionalnosti** koja nije zadana u opisu zadaće i koja će se realizirati zadanim GOF uzorkom.

Potrebno je napraviti program tj. aplikaciju za komandni/linijski mod/terminal u operacijskom sustavu putem kojeg će se izvršiti opisane akcije. Program se NEĆE izvršavati putem razvojnog alata (IDE). Aplikacija NE smije biti s grafičkim korisničkim sučeljem. **Kod izvršavanja programa NE smiju se ispisivati nepotrebni podaci (ostaci od testiranja i sl).**

Kod izvršavanja programa upisuju se argumente/opcije i nazivi datoteka s podacima (u jednoj liniji). Npr:

```
> java -jar  
/home/UzDiz/DZ_3/dkermek_zadaca_3/dkermek_zadaca_3_app/target/dkermek_  
zadaca_3_app.jar --jdr
```

```
> java -jar  
/home/UzDiz/DZ_3/dkermek_zadaca_3/dkermek_zadaca_3_app/target/dkermek_  
zadaca_3_app.jar  
--ta DZ_3_aranzmani.csv --vdr
```

```
> java -jar  
/home/UzDiz/DZ_3/dkermek_zadaca_3/dkermek_zadaca_3_app/target/dkermek_  
zadaca_3_app.jar  
--ta DZ_3_aranzmani.csv --rta DZ_3_rezervacije.csv --jdr
```

Argumenti/opcije i nazivi datoteka nisu obavezni kod pokretanja programa. Ako su upisane nepoznate opcije tada treba javiti informaciju o tome i prekinuti rad programa. Redoslijed opcija je proizvoljan kod unosa komandne linije tako da treba voditi brigu kod obrade primljenih podataka. Vrijednosti za argumente/opcije su proizvoljne tako da će tijekom prezentacije zadaća biti neke druge, a treće kod testiranja i bodovanja zadaća. Nazivi datoteka mogu biti drugačiji nego što su ovdje prikazani.

Za izvršavanje programa prvo je potrebno postaviti da je važeći direktorij/mapa onaj na kojem se nalaze podaci (npr.: cd /home/UzDiz/DZ\_3/podaci). To će biti mjesto s kojeg se izvršava program tako da nazivi datoteka s podacima ne smiju sadržavati putanju. Zbog toga se kod izvršavanja programa mora nalaziti putanja do njegove izvršne verzije (npr.: /home/UzDiz/DZ\_3/dkermek\_zadaca\_3/target/dkermek\_zadaca\_3.jar).

```
> java -jar  
/home/UzDiz/DZ_3/dkermek_zadaca_3/dkermek_zadaca_3_app/target/dkermek_  
zadaca_3_app.jar  
--rta DZ_3_rezervacije.csv --ta DZ_3_aranzmani.csv --vdr
```

U ishodišnom direktoriјu roditeljskog projekta treba priložiti datoteku dokumentacije {LDAP\_korisničko\_ime}\_zadaca\_3.pdf kako je opisano u dokumentima „Preporuke u vezi zadaća“ i „Opći model ocjenjivanja zadaća“.

**Napomena:** smiju se koristiti svi uzorci dizajna koji su obrađeni na nastavi. Ne smiju se koristiti ugrađene osobine odabranog programskega jezika za realizaciju funkcionalnosti pojedinih uzoraka dizajna. Ne smiju se koristite dodatne biblioteke/knjžnice klase osim {LDAP\_korisnik}\_zadaca\_3.lib koja je zadana u opisu zadaće.

Programski kod (klase, atributi i metode klase, varijable, konstante, komentari i sl.) pišu se na **hrvatskom jeziku**. Metode u klasama NE smiju imati više od **30 linija programskog koda**, u što se ne broji definiranje metode, njenih argumenata i lokalnih varijabli. U jednoj liniji može biti jedna instrukcija. Linija ne može imati **više od 120 znakova**. Ne broji se linija u kojoj je samo vitičasta zagrada ili je prazna linija. Ne broje se linije u kojima se nalazi komentar osim u slučaju da se u komentaru nalazi programski kod. Prekoračenja spomenutih ograničenja donose **penalizacijske bodove**.