

# Predmetni projekti 1 i 2 iz predmeta

## Osnove programiranja

Generacija 2018/2019

### Bibliotečko poslovanje

Implementirati konzolnu aplikaciju za bibliotečko poslovanje u programskom jeziku Python (verzija 3.7) primenom proceduralnog programiranja.

Učesnici u poslovanju biblioteke su **Bibliotekar** i **Korisnik** (član biblioteke). Bibliotekar je opisan sa: jedinstvenim identifikatorom (ID); imenom, prezimenom. Korisnik je opisan sa: jedinstvenim brojem članske karte; imenom; prezimenom i podatkom o tome da li je izbrisan. Pored toga i Bibliotekar i Korisnik imaju nalog za prijavu na sistem (korisničko ime, lozinka).

**Zaduženje** je opisano ID-em knjige koja je iznajmljena, datumom iznajmljivanja, datumom vraćanja (ako je korisnik razdužen) i jedinstvenim brojem članske karte Korisnika.

**Knjiga** je opisana sa: jedinstvenim identifikatorom (ID); autorom; godinom izdavanja; ukupan broj primeraka u biblioteci; broj slobodnih primeraka. Autor je predstavljen stringom koji sadrži ime i prezime svih autora knjige.

### Implementacija funkcionalnosti sistema

**Prijava na sistem** – vrši se unosom korisničkog imena i lozinke, sve dok se ne unesu ispravni podaci ili izabere opcija “ODUSTAJEM OD PRIJAVE”. Nakon uspešne prijave ispisuje se poruka da je prijavljen Bibliotekar ili Korisnik sa navedenim imenom i prezimenom. Takođe, ako je izabrana opcija odustajanja od prijave ispisuje se poruka “ODUSTALI STE OD PRIJAVE”.

**Glavni meni za Bibliotekara.** Bibliotekaru se posle prijave pojavljuje sledeći meni:

- Unos i izmena podataka o Knjigama
- Unos podataka za Bibliotekara i Korisnika
- Izmjena podataka trenutno prijavljenog Bibliotekara
- Zaduživanje i razduživanje Korisnika
- Rashodovanje knjiga
- Brisanje Korisnika

Pri unosu i izmeni podataka o Knjigama obezbediti da se ne može uneti knjiga sa već postojećim jedinstvenim identifikatorom ID. Takođe, pri unosu i izmeni podataka za Bibliotekara obezbediti da se ne može uneti Bibliotekar sa već postojećim jedinstvenim identifikatorom (ID), a pri unosu i izmeni podataka za Korisnika obezbediti da se ne može uneti Korisnik sa već postojećim brojem članske karte. Pored toga pri unosu podataka za prijavu na sistem (korisničko ime, lozinka) obezbediti jedinstveno korisničko ime.

Funkcionalnost zaduživanja i razduživanja Korisnika je sledeća. U podmeniju obezbediti pretraživanje Korisnika po svim podacima: jedinstveni broj članske karte; ime; prezime. Rezultat pretraživanja je

spisak Korisnika (sa svim podacima) naveden po rednim brojevima (1.,2. ...). Na osnovu rednog broja selektovati traženog korisnika i prikazati sva zaduženja po rednim brojevima. Razduživanje izvršiti selekcijom odgovarajućeg rednog broja i programski dodati datum razduživanja i povećati broj slobodnih primeraka.

U slučaju zaduživanja pretražiti Knjige samo po jednom ulaznom podatku. Rezultat pretraživanja su sve knjige koje sadrže ulazni podatak u bilo kojem od podataka: jedinstvenim identifikatorom (ID); autorom; godinom izdavanja. Rezultat pretraživanja Knjiga (sa svim podacima) prikazati po rednim brojevima. Bibliotekar vrši selekciju knjige tako što u konzolu ukuca redni broj knjige. Nakon što je knjiga izabrana vrši se pretraga Korisnika, kao u prošlom pasusu. Svi korisnici čiji se podaci slažu sa unetim će biti prikazani i korisnik koji će biti zadužen se bira tako što se ukuca njegov redni broj. Kada se korisnik izabere, pravi se novo zaduženje u koje se upisuje identifikator knjige, broj članske kartice korisnika i trenutni datum.

Rashodovanje knjiga se vrši kada je knjiga u takvom stanju da nije više za korišćenje. Pri rashodovanju knjiga smanjiti ukupan broj primeraka u biblioteci za broj rashodovanih knjiga.

Brisanje korisnika se izvršava tako što su Bibliotekaru ponuđeni svi Korisnici koji trenutno nisu izbrisani. Unosom rednog broja nekog od Korisnika će tom Korisniku podatak o tome da li je izbrisan biti postavljen na True. Moguće je obrisati samo Korisnike koji trenutno nemaju zaduženja. Izbrisan Korisnik ne može da se prijavi na sistem.

Nakon izvršenja bilo koje od navedenih stavki u meniju treba ponovo ponuditi izbor svih funkcionalnosti sve dok ne odabere stavku Izlazak iz aplikacije.

**Glavni meni za Korisnika.** Korisniku se posle prijave pojavljuje sledeći meni:

- Pregled zaduženih Knjiga
- Pretraživanje knjiga
- Izmena podataka trenutno prijavljenog Korisnika

Pri pregledu zaduženih Knjiga Korisniku se pojavljuje spisak svih zaduženih knjiga.

Implementirati pretraživanje knjiga po autoru i naslovu. Rezultat pretraživanja je spisak knjiga i informacija da li ima slobodnih primeraka. Rezultat pretraživanja nije case sensitive što podrazumeva da će se prilikom unosa naziva knjige npr. „tvrđica“, „TVRDICA“, „TvrDiCa“ ... pojaviti knjiga „Tvrđica“ jer pretraga ne treba da uzima u obzir veličinu slova. Takođe treba prikazivati i knjige koje samo delimično sadrže unesenu reč ( npr. ako se unese „drin“ prikazati knjigu „Na Drini ćuprija“ ).

**Izlazak iz aplikacije.** Nakon izvršenja bilo koje od navedenih operacija korisniku treba ponovo ponuditi izbor svih funkcionalnosti sve dok nakon izvršene operacije ne odabere prekid programa.

Napomene:

- Pri pokretanju aplikacije potrebno je jednom učitati sve podatke iz fajlova, a kasnije im pristupati u memoriji programa. Nakon svake izmene, potrebno je tu izmenu i sačuvati u datotekama sa kojima je data izmena povezana.
- Struktura projekta bi trebalo da se sastoji od foldera:
  - src – u ovom folderu se nalaze svi Python-ovi moduli
  - data – u ovom folderu se nalaze sve tekstualne datoteke u kojima se čuvaju podaci

- Implementaciju uraditi sa najmanje četiri modula. Dva modula se odnose na učesnike (**Bibliotekar** i **Korisnik**) i jedan modul na **Knjige**, dok će četvrti modul biti glavni modul (main) iz koga će se pozivati funkcije iz prethodna tri modula. Student može imati i više modula ukoliko uvidi da je tako lakše struktuirati projekat.
- Prilikom unosa podataka, nijedan od unesenih podataka ne sme biti prazan. Tako da se korisniku mora tražiti da unese neki podataka sve dok se ne unese ispravno.
- Za svaku funkciju potrebno je napisati kratak komentar u kome će se opisati koji zadatak navedena funkcija obavlja.
- Menije je potrebno prikazati kao listu stavki, gde će se stavka odabrati unosom rednog broja stavke. Obratiti pažnju da Bibliotekar i Korisnik nemaju isti meni. Ukoliko se napravi pogrešan odabir stavke, prikazati odgovarajuću poruku i ponovo ispisati isti meni, sve dok se ne odabere ispravna opcija.
- Prilikom predaje projekta, student je dužan da projekat spakuje u zip fajl, koji će nositi naziv OP\_X\_Y.

X – oznaka studijskog programa (SII ili IT)

Y – broj indeksa

## Zahtevi za Projekat 1

Za Projekat 1 ne zahteva se korišćenje fajlova za skladištenje podataka. Sve potrebne podatke za testiranje programa inicijalizovati unutar programa. Za drugi deo projekta koristiće se fajlovi. Dakle, bez korišćenja fajlova za Projekat 1 treba implementirati najmanje sledeće funkcionalnosti:

**Prijava na sistem.** Za testiranje prijave učesnika (Bibliotekar i Korisnik) podatke o nalogu (korisničko ime i lozinka) u odgovarajućem obliku dati u programskom kodu.

**Glavni meni za Bibliotekara.** Unos podataka testirati tako što se uneti podaci neće sačuvati u fajlove nego dodavati u odgovarajuće kolekcije i ispisivanje svih elemenata kolekcije (Knjiga, Bibliotekar, Korisnik).

## Zahtevi za Projekat 2

Projekat 2 podrazumeva kompletno implementiranu aplikaciju koja bi radila sa tekstualnim fajlovima u JSON formatu. Ako je student već uradio Projekat 1 onda mi ostaje da još uradi sledeće:

- Da izbací deo koda koji se odnosi na podatke i da podatke unosi u fajlove kao i da podatke čita iz tih fajlova.
- Doda implementaciju funkcionalnosti: **Glavni meni za Korisnika.**

## **Odbrana Projekata**

Prvi termin za odbranu Projekta 1 i/ili Projekta 2 biće organizovan u XII nedelji nastave – kolokvijumska nedelja. Sledeći termin za odbranu projekata biće u XV nedelji nastave. Posle toga u svakom ispitnom roku biće organizovana odbrana projekata. Postupak za odbranu projekata biće posebno objavljen.