

Programozás alapjai 3. - NHF Dokumentáció

Balogh Csaba – IP4W05

Könyvtárkezelő program

Programozói rész:

Feladat: Egy a specifikációban előzőleg leírt feladatokat ellátni tudó könyvtárkezelő alkalmazás készítése grafikus interfésszel Java nyelven. A program képes könyvek felvételére, törlésére az adatbázisból, paramétereik módosítására, a kölcsönzések nyilvántartására, a kölcsönző személyek hozzáadására, törlésére a rendszerből, az adataik módosítására, új kölcsönzések bevitelére, visszavételére, továbbá a könyvek közötti keresésre, a kölcsönzött könyvek szűrésére. A specifikációhoz képest többletfunkciókkal rendelkezik a program: kölcsönzőkkel kapcsolatos adatműveletek, könyvvisszavétel, keresés több adattag alapján, adatok módosítási lehetősége...

A dokumentációt a fájlkezelés és rövid adatszerkezetes részével kezdem, valamint egy segéd adatstruktúra jellemzésével, ezután fejtem ki az egyes osztályok legfontosabb tulajdonságait, funkcióit.

Megjegyzés: A KonyvtarFrame osztály kivételével jellemezve vannak az egyes tagfüggvények a többi osztály esetében a tárgyhonlapon kért 1-2 mondatos hosszúságban. A KonyvtarFrame tagfüggvényei konkrétan a lentebb ismertetett funkciókat valósítják meg közvetlenül, így azok ott kerülnek jellemzésre külön pontban. Továbbá az az osztályokról generált UML diagram a generátor sajátosságai miatt magában foglalja a Java beépített könyvtárainak is a viszonyrendszerét.

Fájlkezelés és fájlműveletek

A fájlba írás szerializáció (java.io.Serializable, interfész implementációja), valamint a beépített File/FileChooser... osztályok segítségével került kivitelezésre. Az adatok lementése és visszaolvasása egy az „anyamappában” található - konyvtar.libdat - nevű fájlba történik programozottan, de lehetősége van a felhasználónak az adatok mentésének kihagyására a program bezárásakor, ha a Fájlok menüpont „Csak bezárás” parancsát választják. Minden a fájlal kapcsolatos hiba esetén (StreamCorr., FileNotFoundException...) kivétel dobódik és a felhasználónak jelzi a program, hogy milyen hiba történt a fájlműveletek során. Lehetőség van a „Mentés mint” parancssal az adatok mentésének egy másik helyre történő kivitelezésére, továbbá a „Bezárás mentéssel”. Az „Importálás” fül alatt lehetőség van egy másik helyről beolvasni a szerializált adatokat.

Fa adatszerkezet a kölcsönzésekhez

A „Könyvek-Kölcsönzők” osztályok egyes példányainak kapcsolatát leíró, nyilvántartó adatszerkezet, amely tárolja, hogy az egyes kölcsönzőkhöz melyik könyv(ek) tartoznak. A könyvek visszavételekor, valamint a Könyvek Táblázatában („Összes könyv” címkével ellátott táblázat) az egyes kölcsönzött könyvekhez tartozó kölcsönző személy megadásakor, elvételekor fontos szerepet tölt be. A könyvek visszavételekor az adott könyvhöz tartozó kölcsönző személye null értékű lesz, ezzel elérve, hogy az adott kölcsönzés megszűnt.

Minden más esetben a beépített List/ArrayList listákban lesznek eltárolva az egyes osztályok példányai, a heterogén kollekció osztálypéldányai (már ahol van).

Osztályok

Delegációt (Serializable), öröklést is használok az egyes osztályok megvalósításakor. Az enumerációs adattípusok visszaadása String formájában történik a programban.

- Konyvek: Az egyes könyvek adatait tároló alaposztály.
- KonyvAdat: A felhasználó által is látható programrész (táblák) szerkezetét, működését határozza meg, a modellje annak.
- KonyvPanel: Könyv adatnak bevitelére szolgáló JPanel objektum (öröklés).
- Mufaj: Enumeráció, amely tartalmazza a könyvek lehetséges műfajait és azok magyar megfelelőt visszaadni tudó függvénnyel is bír.
- KeresoPanel: A keresés funkció megvalósítására szolgáló JPanel objektum (öröklés), könyvekre vonatkozik.
- Struktúra: A kölcsönzések tárolására kitalált adatszerkezet.
- Kolcsonzo: A kölcsönzők adattagjait tároló, a kölcsönzőt megvalósító osztály.
- KolcsonzoAdat: A kölcsönzők tábla szerkezetét, modelljét megadó osztály.
- KolcsonzoPanel: A kölcsönzők adatainak bevitelére, azok módosítására szolgáló JPanel objektum (öröklés).
- KolcsonzesKezelo: A könyvek kölcsönzésekor/visszavételekor használt osztály, amelynek függvényeivel be lehet változtatni, hogy ki kölcsönözte ki az adott könyvet, vagy azt, hogy egyáltalán ki van-e kölcsönözve.
- Konyvtar: Az összes eddigi adat a Konyvtar osztályban tárolódik, ebbe történik az adatok betöltése, a könyvek/kölcsönzők hozzáadása/eltávolítása.
- KonyvtarFrame: A program maga, a főpanel. A felhasználó által is látható gombok, címkék... elhelyezéséért felel a képernyőn, valamint az egyes gombok lenyomásának (eseménykezelés) figyelésért (és ekkor a megfelelő függvények meghívásáért) is felelős. JFrame leszármazott.

A KonyvtarFrame osztály ismertetése

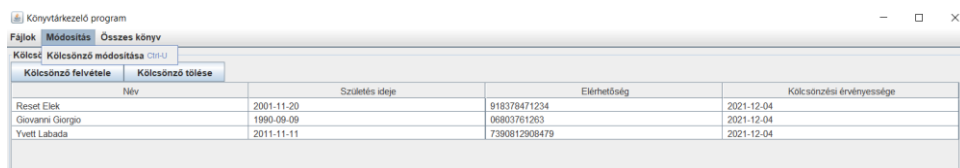
- KonyvtarFrame(): A konstruktor, amelyben az egyes adattagok inicializálódnak és a grafikus jellemzőik megadódnak az egyes komponenseknek. Ezentúl megtörténik az adatok fájlból való beolvasásáért, a kezdőablak összerakásáért, az egyes táblázatok inicializálásáért felelős függvények meghívása.
- fajlbolBeolvas(): Az adatok konkrét beolvasásáért felelős függvény.
- fajlImportAblak(): Alternatív fájl megadásáért felelős, amelyből az adatokat be lehet olvasni (Import funkció). Mindenképpen „libdat” fájlok lesznek a beolvasható fájlok. Ezután megtörténik az adatok beolvasása a létrehozott táblázat objektumokba és a későbbi megjelenítésük.
- inditMentesMint(): Ha a felhasználó egy nem az alapértelmezetten megadott – helyi mappába - szeretné menteni az adatfájlt, akkor ezen függvény meghívásakor megjeleik az a panel, amellyel kiválaszthatja, hogy hova menti el.

- hozzaadKonyv(): Az egyes könyvek adatait erre, a függvény által behozott panelre kell bevinni. Ezután láthatóvá válik az újonnan bevitt könyv és paraméterei.
- torolKonyv(): Az előző függvény párja, ugyanakkor a törlését valósítja meg egy könyvnek, kijelölését követően.
- modositKolcsonzo(): Egy adott kölcsönző adatainak módosítását valósítja meg a behozott panelen keresztül.
- inditAblak(): Az indulási ablak értékeit (magasság szélesség, maga a felbontás is így) határozza meg. Felelős még az ablak bezárásakor (X) történő automatikus mentésért, valamint az alkalmazás ablakának méretbeli változtatásáért.
- inditMenuPanel(): Az egyes funkciók megvalósításáért felelős, menüsorban található gombok, ikonok megalkotása a feladata. FÁJLOK, MÓDOSÍTÁS, ÖSSZES KÖNYV.
- inditKonyvTablazat(): Könyvek táblázatának létrehozása, egyes könyvekkel kapcsolatos funkciók/műveletek, kereső mező, valamint a kölcsönzöttek szűrésére szolgáló mező megalkotása a felelőssége
- inditKolcsonTablazat(): A kölcsönzők táblázatának, velük kapcsolatos műveletek ikonjainak létrehozása a függvény feladata.
- frissitStruk(): Ebben a függvényben a fa adatstruktúrában tárolt kölcsönzéseket érintő változások (visszavétel, kölcsönző/érintett könyv törlése) könyvelése a feladata.
- initCellaMod(): A legördülő menüben (JComboBox-ban megjelenített) könyv műfajokat vagy éppen a kölcsönzőket állítja be.

Ezen túl a TableModelListener osztály implementációja is megtörténik (delegáció) a két táblázatra nézve külön-külön. Emiatt az adatok változása nem csak elkönyvelődik, de érzékelhetővé is válik a felhasználó számára.

Az alábbi műveleteket lehet végrehajtani a megjelölt („”) gombokkal:

- „Kölcsönző felvétele” - A kölcsönző felvétele panel felugrik és a felhasználó beviheti az egyes adattagokat: Név, Születési idő, Telefonszám. Ezt követően az OK gombra kattintva elmentésre kerülnek az adatok a táblázatban és a felhasználó előtt is megjelennek a bevitt adatok.
- „Kölcsönző törlése” - A kijelölt cella kitörlése történik meg, amit azonnal látni is fog a felhasználó. Fontos, hogy ekkor megszűnik a az adott kölcsönzőhöz tartozó összes aktív kölcsönzés és ezek a könyvek újra kikölcsönözhetővé válnak
- „Kölcsönző módosítása”- A Módosítás menüpont alatt van. - A kölcsönző adatainak módosítására szolgáló panel ugrik fel, ahol a felhasználó felviheti az általa helyesnek vélt adatokat. CTRL + U billentyűkombinációval is elérhető, vagy az érintett cellára (sorra) való dupla kattintással is.



Név	Születési ideje	Elérhetőség	Kölcsönzési érvényessége
Róbert Elek	2001.11.20	918378471234	2021.12.04
Giovanni Giorgio	1990.09.09	06803761263	2021.12.04
Yvett Labada	2011.11.11	7380812908479	2021.12.04

Könyvműveletek

„Összes könyv” menüpont alatt vagy a könyvek táblázatban található gombok segítségével érhetőek el:

- „Új könyv felvétele”: Megjelenik a felhasználónak az adatbeviteli panel, amellyel a könyvek egyes jellemzőit be kell vinni. Szerző, címe, műfaj, nyelve, megjelenés éve, kölcsönzési státusz. Helytelen adatok megadásakor (üresen hagyott dobozka esetén) hibaizenetet kap a felhasználó és újra próbálkozhat. Azt, hogy ki kölcsönözte ki csak később tudja majd beállítani, az adatok bevitelét és az OK gombra kattintást követően. A CANCEL gombbal kiléphet a panelből, tehát ezzel félbeszakítja az adatok bevitelét.
- „Könyv törlése”: A kijelölt könyvet lehet ezzel a funkcióval eltávolítani a könyvtárból. Megerősítést kér majd a program a törlés szándékáról, akkor is ha ki van kölcsönözve a könyv, vagy ha nincs. Ha kivan kölcsönözve egy konkrét kölcsönző által a könyv, akkor az adott kölcsönzés megszűnik a személyre nézve (már nem nála lesz nyilvántartva a könyv helye).
- „Visszavétel”: Ezzel a funkcióval akkor élhet a felhasználó, ha egy olyan könyvet jelöl ki (rákattint az adott sorra), amihez be van jegyezve egy kölcsönző (a kölcsönző nevének van egy név). A visszavétellel megszűnik a könyv kölcsönzése, újra kölcsönözhetővé válik.
- „Rendezés” - Mint külön menüpont nincsen, de ha a felhasználó az adott oszlop nevére kattint, akkor a megfelelő rendezés szerint (lexikografikus, abc, növekvő, csökkenő...) jeleníti meg az adatokat, ami hasznos lehet.
- „Kölcsönzöttek szűrése” - A nyilvántartott könyvek közül csak azokat listázza ki, amelyek ki vannak kölcsönözve, tehát a kölcsönző nevének szerepel egy megfelelő név. Ekkor az említett könyv cellái ki vannak emelve (félkövér típusú betűkkel), ezzel is jelezve a kölcsönzési státuszukat.

- „Keresés” - Keresés a szerző nevében bepipálva kell legyen ha a szerző nevében is szeretnénk keresni. - Alapvetően a kereső mező a nyilvántartott könyvek közötti keresésre szolgál.

Fájlok menüpont használata

- „Importálás”: - CTRL + O gyorsbillentyűvel elérhető -
- „Adatok mentése”: - CTRL + S gyorsbillentyűvel elérhető - A bevitt adatok az alapértelmezett mentési helyre (szerializációs útvonal - “könyvtar.libdat”) mentésre kerülnek, itt természetesen nincs kilépés a programból a mentést követően. Ez akkor is megtörténik automatikusan, ha az ablakot az X-el zárja be (jobb felső sarok), így nincs adatvesztés.
- „Mentés mint”: Meg lehet ezzel a funkcióval adni egy másik mentési helyet és másik mentési nevet, de az adatfájl továbbra is .libdat kiterjesztésű marad. A felhasználó a felugró panelben kiválaszthatja - éppen mint a fájlkezelőben -, hogy hova szeretné menteni az adatokat és milyen néven fusson a fájl.
- „Csak bezárás”: Mentés nélküli bezárása a programnak, ekkor nem menti el az esetleg módosításokat/bevitt/törölt adatokat, a műveletek nem kerülnek könyvelésre.
- „Bezárás mentéssel”: - CTRL + E gyorsbillentyűvel elérhető - Az adatok/módosítások/elvégzett műveletek a célmappába, cél fájlba (amely alapértelmezetten van megadva - “könyvtar.libdat”) mentésre kerülnek, és bezáródik a program ablaka.

Az egész program ablakának méretét lehet változtatni, növelni-csökkenteni a kurzorral, ha a felhasználó a program valamelyik sarka/pereme mentén próbálja a kurzort lenyomva tartott bal egérgomb mellett mozgatni. Hasonló módon lehet a két táblázat egymáshoz képest látott méretét változtatni a vastagon szedett, vízszintes elválasztóvonal mozgásával.

JUnit tesztek eredményei

Sikeresek voltak a lefutott JUnit tesztek, amelyek a legfőbb funkciókat tesztelték:

- KonyvtarTest az alábbiakat tesztelte le:
 - Konyvtar() helyes működése, osztály sikeres létrehozása.
 - Fájlból való beolvasása az adatoknak.
 - Mentés.
 - Változók sikeres inicializálása.
 - Az egyes adatbázisban levő kölcsönzők adatainak helyessége.
 - Az adatok fájlba való írásának helyessége és azt követő megfelelő visszaolvasása. Lehet látni, hogy a bevitt adatok sikeresen lementődtek a megadott fájlba és onnan visszaolvasódtak (a lekérdezett adattagok helyesek voltak). Egy könyv került bevitelre a táblázatba a képen látható paraméterekkel.

```

1 package konyvtar;
2
3 import static org.junit.jupiter.api.Assertions.*;
4
5 import org.junit.Before;
6 import org.junit.jupiter.api.Test;
7
8 import junit.framework.Assert;
9
10 class KonyvtarTest {
11
12     Konyvtar konyvtar;
13
14     @SuppressWarnings("deprecation")
15     @Test
16     void beolvas() {
17         this.konyvtar = new Konyvtar();
18         this.konyvtar.initializalValtozok("konyvtar.libdat");
19         this.konyvtar = Konyvtar.fajlbolBeolvas(konyvtar);
20         Assert.assertEquals(3, konyvtar.getKonyvek().size());
21         Assert.assertEquals("Balogh Árpád", konyvtar.getKonyvek().get(0).getKonyvSzerzo());
22     }
23
24 }
25

```

```

7
8 import junit.framework.Assert;
9
10 class KonyvtarTest {
11
12     Konyvtar konyvtar;
13
14     @SuppressWarnings("deprecation")
15     @Test
16     void beolvas() {
17         this.konyvtar = new Konyvtar();
18         this.konyvtar.initializalValtozok("konyvtar.libdat");
19         this.konyvtar = Konyvtar.fajlbolBeolvas(konyvtar);
20         Assert.assertEquals(3, konyvtar.getKonyvek().size());
21         Assert.assertEquals("Balogh Árpád", konyvtar.getKonyvek().get(0).getKonyvSzerzo());
22     }
23
24     @Test
25     void beviszadat() {
26         this.konyvtar = new Konyvtar();
27         this.konyvtar.initializalValtozok("konyvtar.libdat");
28         this.konyvtar = Konyvtar.fajlbolBeolvas(konyvtar);
29         this.konyvtar.addKolcsonzo("Hunor Monor", "1999-03-09", "6306541347");
30         this.konyvtar.mentes();
31         this.konyvtar = new Konyvtar();
32         this.konyvtar.initializalValtozok("konyvtar.libdat");
33         this.konyvtar = Konyvtar.fajlbolBeolvas(konyvtar);
34         assertEquals("1999-03-09", konyvtar.getKolcsonzok().get(4).getSzulido().toString());
35     }
36 }
37

```

- Konyvtarkezelot:
 - Teszteli az egyes kölcsönzőkkel kapcsolatos műveletek helyes működését, pl.: kölcsönző eltávolítása a rendszerből.
 - A könyvek kikölcsönzését ellenőrző tesztet is tartalmaz, amelyben megnézi, hogy sikeresen lett-e bejegyezve a kölcsönzési státusz és az adott könyvhöz tartozó kölcsönző személye, előzetesen a kölcsönzési checkbox (pipa) true-ra lett állítva (isSelected()).
 - A könyvtárhoz való könyv hozzáadását is teszteli, sikeresen. Megadott paraméterekkel rendelkező könyvet ad hozzá a könyvek listához, aminek (lista) a méretét lekérdezi és ezt követően megnézi, hogy stimmel-e a szám.
 - A törlés funkciót oly módon teszteli, hogy létrehozza a könyvtár objektumot, hozzáad egy könyvet - a könyvek számát is ellenőrzi -, ezután kiveszi a listából azt az egy könyvet (törli).

```

33 konyvtar.initializalValtozok(konyvtar.libdat);
34 konyvtar.addKonyv("Winston Churchill", "Kubai szivarom", "1930", Mufaj.MISCELLANEOUS, "magyarul", false);
35 Assert.assertEquals(1, konyvtar.getKonyvek().size());
36 konyvtar.eltavolit(konyvtar.getKonyvek().get(0));
37 Assert.assertEquals(0, konyvtar.getKonyvek().size());
38
39 }
40
41 @Test
42 public void kikolcsonoz() {
43     konyvtar = new Konyvtar();
44     konyvtar.initializalValtozok("konyvtar.libdat");
45     Kolcsonzo kolcsonzo = new Kolcsonzo("06603214590", "Patta Nóra", LocalDate.of(2000, 11, 11));
46     Assert.assertEquals(0, kolcsonzo.getKolcsonzottKonyvek().size());
47     Konyvek konyv = new Konyvek("Winston Churchill", "Kubai szivarom", 1930, Mufaj.MISCELLANEOUS, "magyarul", true);
48     this.konyvtar.getKonyvekadatai().addKonyv(konyv);
49     this.konyvtar.getKonyvekadatai().setValueAt(kolcsonzo, 0, 6);
50     //Assert.assertEquals(kolcsonzo.getKolnev(), this.konyvtar.getKonyvekadatai().getValueAt(0, 6));
51     //Assert.assertEquals(1, kolcsonzo.getKolcsonzottKonyvek().size());
52     Assert.assertSame(konyv.getKikolcsonzi().getKolnev(), kolcsonzo.getKolnev());
53 }
54
55 @Test
56 public void kolcsonzoEltavolit() {
57     konyvtar = new Konyvtar();
58     konyvtar.initializalValtozok("konyvtar.libdat");
59     Kolcsonzo kolcsonzo = new Kolcsonzo("06603214590", "Patta Nóra", LocalDate.of(2000, 11, 11));
60     this.konyvtar.addKolcsonzo("Patta Nóra", LocalDate.of(2000, 11, 11).toString(), "06603214590");
61     Assert.assertEquals(1, konyvtar.getKolcsonzok().size());
62     this.konyvtar.eltavolit(konyvtar.getKolcsonzok().get(0));
63     Assert.assertEquals(0, konyvtar.getKolcsonzok().size());
64 }
65

```