

Feladat

A könyvtár nyilvántartja a könyvtárba beiratkozott személyeket, és a kikölcsönözhető könyveit. Könyvtári tag az a személy lehet (ismerjük a nevét), aki beiratkozik a könyvtárba.

Egy könyvtári tag egy alkalommal legfeljebb öt, a könyvtárban meglévő könyvet kölcsönözhet ki. Egy könyvnek ismert a címe, szerzője, kiadója, ISBN száma, az oldalszáma, és van egy könyvtári azonosítója, miután a könyvtárba kerül. Az egyszerre kikölcsönzött könyveket több részletben is vissza lehet hozni, így egy kölcsönzés eseményhez tartozó könyvek listája folyamatosan csökkenhet.

Egy könyv kölcsönzési pótdíja a kölcsönzés lejáratí idejétől számított napok számától függ, de az egy napi pótdíj a könyv műfajától (természettudományi, szépirodalmi, ifjúsági) és példányszámát jellemző kategóriától (ritkaság, sok példány) függ.

napi pótdíj	ritkaság	sok példány
természettudományi	100	20
szépirodalmi	50	10
ifjúsági	30	10

- Tegye lehetővé, hogy a könyvtár beszerezhesen egy könyvet, egy új személy be tudjon iratkozni, egy tag kikölcsönözhesse az általa kért könyvek közül azokat, amelyek jelenleg elérhetők, és bármikor visszahozhasson egy kikölcsönzött könyvet.
- Mennyi pótdíjat kell fizetnie egy tagnak a vissza nem hozott könyvei után?

Készítsen használati eset diagramot! Ebben jelenjenek meg használati esetként a később bevezetett fontosabb metódusok. Adjon meg a fenti feladathoz egy olyan objektum diagramot, amely mutat öt könyvet, két könyvtári tagot, hozzájuk kapcsolható három kölcsönzési tevékenységet, ahol az egyik kölcsönzés egyszerre két könyvet is tartalmaz.

Rajzolja fel a feladat osztály diagramját (először csak a konstruktorokkal)! Azoknak a privát/védett adattagoknak a láthatóságát, amelyekhez getter-t is, és setter-t is kell készíteni, jelölheti publikusnak. (A triviális getter/setter-eket később sem kell beírni a modellbe.)

Készítse el egy könyv objektum állapotgépét! Különböztesse meg a „könyvtárban”, és a „kikölcsönözve” állapotokat. Az állapot-átmeneteket megvalósító tevékenységeket majd a könyv osztály metódusaiként definiálhatja.

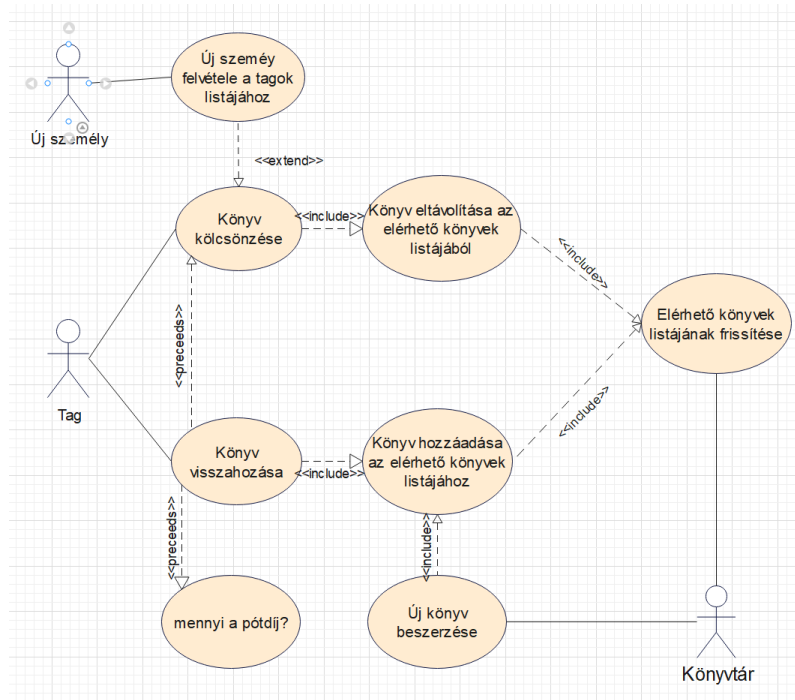
Egészítse ki az osztálydiagramot az objektum-kapcsolatokat létrehozó metódusokkal, valamint a feladat kérdéseit megválaszoló metódusokkal. A metódusok leírásában a félév első felében bevezetett végrehajtható specifikációs jelöléseket használja. Azoknak a konstruktoroknak a törzsét, amelyek kizárólag az adattagok inicializálását végzik, nem kell feltüntetni. Ilyenkor a konstruktor paraméterlistája helyén elég felsorolni az inicializálandó adattagok neveit. Az összes közvetlen (tehát nem szerepnév) adattag felsorolása helyett elég „...”-ot írni.

Használjon tervezési mintákat, és mutasson rá, hogy hol melyiket alkalmazta.

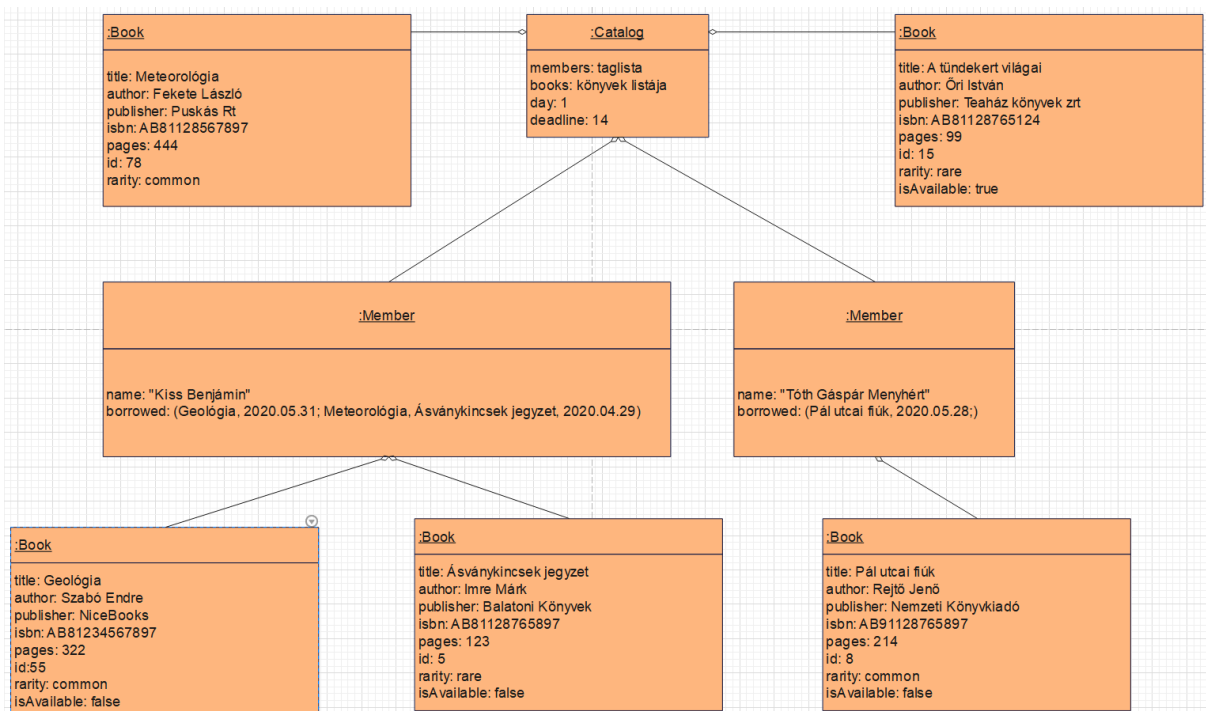
Implementálja a modellt! Szerkesszen olyan szöveges állományt, amelyből fel lehet populálni egy könyvtár könyveit, könyvtári tagjait, néhány kölcsönzést és könyv visszahozást. Számoljuk ki egy tag pótdíját! Készítsen teszteseteket, néhánynak rajzolja fel a szekvencia diagramját, és hozzon létre ezek kipróbálására automatikusan tesztkörnyezetet

Terv

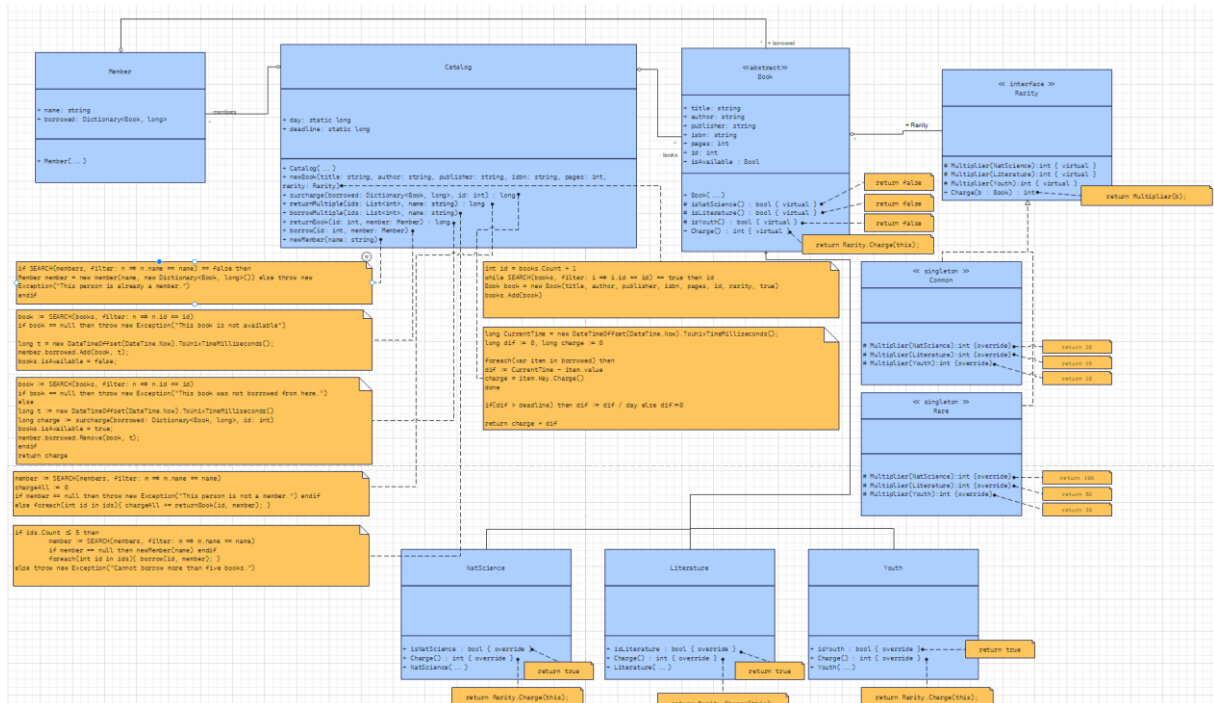
A feladat megoldásának fő mozzanatai a könyvtári könyvek kölcsönzése és visszahozása, és ennek vezetése katalógusban. Ezek változtatnak a könyvek és a tagok listáján.



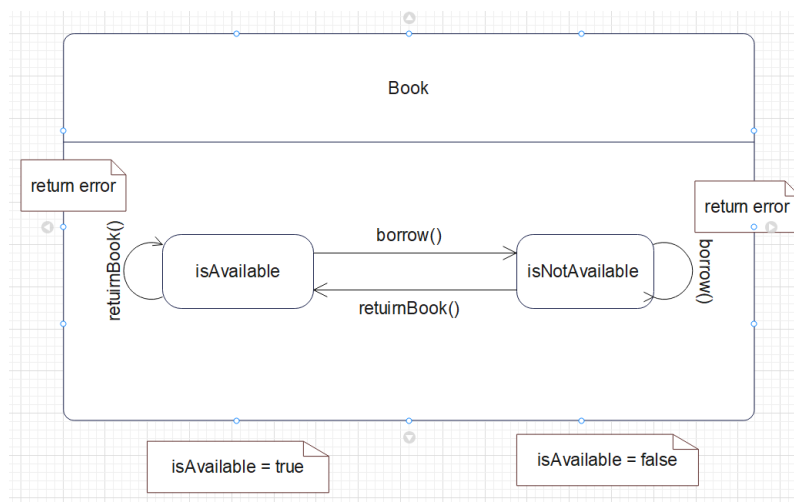
Eközben kommunikáció történik a tagok és a könyvek közt, akár csak a katalógussal. Ennek következtében lesznek elérhetőek vagy nem elérhetőek, vagy változik azok száma.



Az egyes feladatrészek megoldásának specifikációját és az osztályok közti kapcsolatokat az alábbi ábra szemlélteti. A **filter** egy olyan függvény, amely kiválogatja az adott feltételnek megfelelő elemeket.



A Könyvek listájában minden könyv rendelkezik egy elérhetőségi igaz/hamis (isAvailable) változóval, ami jellemzi az adott azonosítóval rendelkező könyvet kikölcsönözték, vagy a könyvtárban van. Ha egy olyan könyvet szeretnénk kikölcsönözni (borrow()), ami nem elérhető, akkor hibát kapunk, más esetben az isAvailable = false lesz. Ugyanez igaz akkor is, ha egy olyan azonosítóval rendelkező könyvet szeretnénk visszahozni (returnBook()) ami amúgy is a könyvtárban van, de ha nincs, az isAvailable = true lesz.



Tesztelési terv

A teszteléshez fehérdobozos módszert alkalmaztam a rendszerteszteléshez. Két tesztelő osztály van, az elsőben a bemeneti fájl kerül feldolgozásra, itt ezen állapot alapján teszteli a kód, hogy az értékek megfelelően lettek-e kiszámítva.

A másodikban a feladatoknak megfelelő metódusok tesztesei futnak, ami összefügg az objektumdiagram tartalmával is, a program tesztesei a gyakorlatban ezek alapján lesz megoldva.

