**Adatkezelés XML-ben**

1. **féléves beadandó feladat**

**Név:** Kiss Áron

**Neptun kód:** UHV61T

**Gyakorlatvezető:** Agárdi Anita

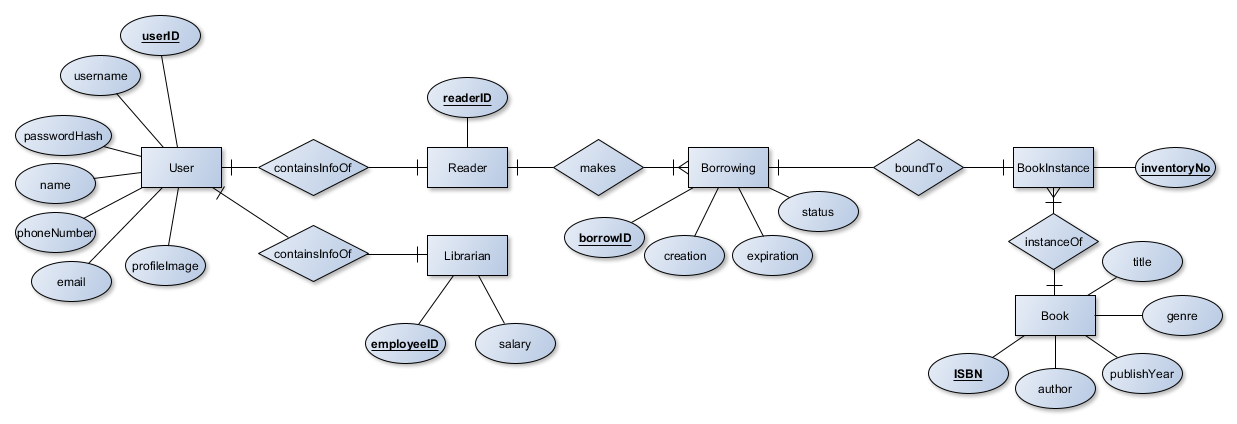
**Gyakorlat:** Szerda 10-12

**A feladat rövid leírása**

A feladat egy tetszőlegesen választott témában XML séma tervezése ER modell és XDM modell segítségével, majd a megtervezett séma megvalósítása. A sémának kb. 10 típusból kell állnia, és tartalmaznia kell attribútumot, kulcsot, idegen kulcsot, valamint származtatást (megszorítás, bővítés).

A feladat megoldására egy könyvtári kölcsönző rendszer XML sémáját valósítottam meg.

**ER modell**



Az ER modell egyedei, és azok tulajdonságai:

* **User** egyed: A rendszer felhasználóit jelképezi
  + userID: a felhasználó egyedi azonosítója
  + username: A felhasználó rendszerben használt neve
  + passwordHash: A felhasználó jelszavának hash-e
  + name: a felhasználó valódi neve
  + phoneNumber: a felhasználó telefonszáma
  + email: a felhasználó email címe
  + profileImage: a felhasználó profilképének elérési útvonala (pl. [*http://example.com/img.jpg*](http://example.com/img.jpg))
* **Reader** egyed: A könyvtár olvasóit jelképezi
  + readerID: az olvasó azonosítója
* **Librarian** egyed: A könyvtár könyvtárosait jelképezi
  + employeeID: a könyvtáros egyedi azonosítója
  + salary: a könyvtáros fizetése
* **Borrowing** egyed: Az olvasók kölcsönzéseit jelképezi
  + borrowID: a kölcsönzés egyedi azonosítója
  + creation: a kölcsönzés regisztrálásának dátuma
  + expiration: a kölcsönzés lejártának dátuma
  + status: a kölcsönzés státusza (lehetséges értékek: BORROWED, RETURNED, EXPIRED)
* **Book** egyed: A könyvtár könyveinek fő adatait tartalmazza
  + ISBN: a könyv ISBN azonosítója
  + author: a könyv szerzője
  + publishYear: a könyv kiadásának éve
  + genre: a könyv műfaja (lehetséges értékek: Scifi, Crimi, Natural, Fiction)
  + title: a könyv címe
* **BookInstance** egyed: A könyvek fizikai példányainak adatait tartalmazza
  + inventoryNumber: a könyv példányának raktári száma

Az egyedek közti kapcsolatok

**User** containsInfoOf **Reader:** 1:1 kapcsolat, mert 1 felhasználóhoz 1 olvasó, és 1 olvasóhoz 1 felhasználó tartozhat.

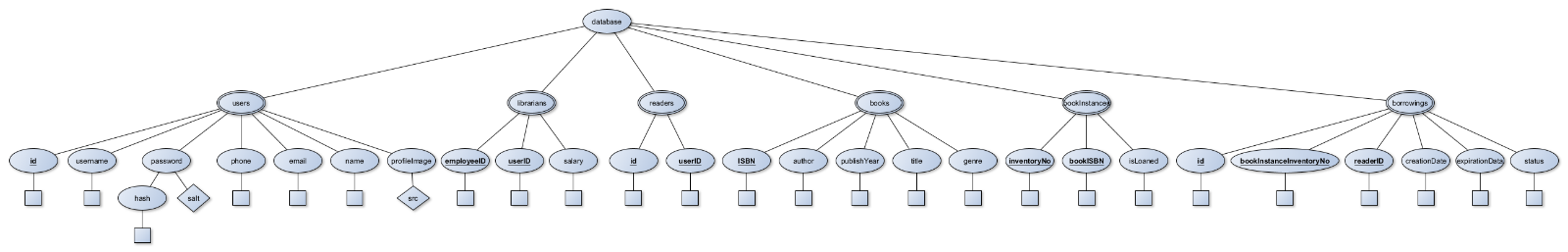
**User** containsInfoOf **Librarian**: 1:1 kapcsolat, mert 1 felhasználóhoz 1 könyvtáros, és 1 könyvtároshoz 1 felhasználó tartozhat.

**Reader** makes **Borrowing**: 1:N kapcsolat, mert 1 olvasónak lehet több kölcsönzése, de 1 kölcsönzés 1 olvasóhoz tartozhat.

**Borrowing** boundTo **BookInstance**: 1:N kapcsolat, mert 1 kölcsönzéshez 1 könyv példány kapcsolódhat, de 1 könyvpéldány tartozhat több kölcsönzéshez is.

**BookInstance** instanceOf **Book**: 1:N kapcsolat, mert 1 könyv példány 1 könyvhöz kapcsolódhat, de 1 könyvnek lehet több példánya is.

**XDM modell**

****

*Ha az XDM modell nem olvasható,* [*teljes méretben itt érhető el.*](https://i.imgur.com/Op03bq2.png)

A fent megadott egyedeket egy „database” nevű csomópont kapcsolja össze gyökérelemként. A felhasználókhoz tartozó profilkép tulajdonság üreselem lesz, 1 attribútummal, ami a profilkép elérési útvonalát tartalmazza. A jelszó külön típus lesz, ami attribútumként tartalmazza a salt értéket, és egy „hash” nevű text-only elemként a jelszó hash-t.

Az XDM modell további részei egyértelműen adódnak az ER modellből.

**XML séma**

A sémában a User típus a personInformation kibővítése lesz, megszorítást a „genre” értékeknél, és a kölcsönzések státuszánál alkalmazok.

A fenti információk alapján létrehozott XMLSchema definíció:

<?xml version="1.0" encoding="UTF-8"?>  
<!-- Kiss Áron, UHV61T -->  
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" >  
 <xs:complexType name="passwordType">  
 <xs:sequence>  
 <xs:element name="hash" type="xs:string"></xs:element>  
 </xs:sequence>  
 <xs:attribute name="salt" type="xs:string"></xs:attribute>  
 </xs:complexType>  
   
 <!-- empty element -->  
 <xs:complexType name="profileImage">  
 <xs:attribute name="src" type="xs:string"></xs:attribute>  
 </xs:complexType>  
   
 <xs:complexType name="personInformation">  
 <xs:sequence>  
 <xs:element name="name" type="xs:string"></xs:element>  
 <xs:element name="email" type="xs:string"></xs:element>  
 <xs:element name="mobileNumber" type="xs:string"></xs:element>  
 <xs:element name="image" type="profileImage"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:complexType name="userType" id="username">  
 <xs:complexContent>  
 <!-- extension -->  
 <xs:extension base="personInformation">  
 <xs:sequence>  
 <xs:element name="id" type="xs:ID"></xs:element>  
 <xs:element name="username" type="xs:string"></xs:element>  
 <xs:element name="password" type="passwordType"></xs:element>  
 <xs:element name="info" type="personInformation"></xs:element>  
 </xs:sequence>  
 </xs:extension>  
 </xs:complexContent>  
 </xs:complexType>  
   
 <xs:complexType name="librarianType">  
 <xs:sequence>  
 <xs:element name="employeeID" type="xs:ID"></xs:element>  
 <xs:element name="salary" type="xs:int"></xs:element>  
 <xs:element name="userID" type="xs:IDREF"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:complexType name="readerType">  
 <xs:sequence>  
 <xs:element name="id" type="xs:ID"></xs:element>  
 <xs:element name="userID" type="xs:IDREF"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:simpleType name="genreType" final="restriction">  
 <!-- restriction -->  
 <xs:restriction base="xs:string">  
 <xs:enumeration value="Scifi"></xs:enumeration>  
 <xs:enumeration value="Crimi"></xs:enumeration>  
 <xs:enumeration value="Natural"></xs:enumeration>  
 <xs:enumeration value="Fiction"></xs:enumeration>  
 </xs:restriction>  
 </xs:simpleType>  
   
 <xs:complexType name="bookType">  
 <xs:sequence>  
 <xs:element name="ISBN" type="xs:ID"></xs:element>  
 <xs:element name="author" type="xs:string"></xs:element>  
 <xs:element name="title" type="xs:string"></xs:element>  
 <xs:element name="publishYear" type="xs:int"></xs:element>  
 <xs:element name="genre" type="genreType"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:complexType name="bookInstanceType">  
 <xs:sequence>  
 <xs:element name="inventoryNo" type="xs:long"></xs:element>  
 <xs:element name="bookISBN" type="xs:IDREF"></xs:element>  
 <xs:element name="isLoaned" type="xs:boolean"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:simpleType name="borrowStatusType" final="restriction">  
 <xs:restriction base="xs:string">  
 <xs:enumeration value="BORROWED"></xs:enumeration>  
 <xs:enumeration value="RETURNED"></xs:enumeration>  
 <xs:enumeration value="EXPIRED"></xs:enumeration>  
 </xs:restriction>  
 </xs:simpleType>  
   
 <xs:complexType name="borrowingType">  
 <xs:sequence>  
 <xs:element name="id" type="xs:ID"></xs:element>  
 <xs:element name="creationDate" type="xs:date"></xs:element>  
 <xs:element name="expirationDate" type="xs:date"></xs:element>  
 <xs:element name="status" type="borrowStatusType"></xs:element>  
 <xs:element name="readerID" type="xs:IDREF"></xs:element>  
 <xs:element name="bookInstanceInventoryNo" type="xs:IDREF"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
   
 <xs:element name="database">  
 <xs:complexType>  
 <xs:sequence>  
 <xs:element name="users">  
 <xs:complexType>  
 <xs:sequence minOccurs="0" maxOccurs="unbounded">  
 <xs:element name="user" type="userType" ></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
   
 <xs:element name="librarians">  
 <xs:complexType>  
 <xs:sequence minOccurs="0" maxOccurs="unbounded">  
 <xs:element name="librarian" type="librarianType"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
   
 <xs:element name="readers">  
 <xs:complexType>  
 <xs:sequence minOccurs="0" maxOccurs="unbounded">  
 <xs:element name="reader" type="readerType"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
   
 <xs:element name="books">  
 <xs:complexType>  
 <xs:sequence minOccurs="0" maxOccurs="unbounded">  
 <xs:element name="book" type="bookType"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
   
 <xs:element name="bookInstances">  
 <xs:complexType>  
 <xs:sequence minOccurs="0" maxOccurs="unbounded">  
 <xs:element name="bookInstance" type="bookInstanceType"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
   
 <xs:element name="borrowings">  
 <xs:complexType>  
 <xs:sequence minOccurs="0" maxOccurs="unbounded">  
 <xs:element name="borrowing" type="borrowingType"></xs:element>  
 </xs:sequence>  
 </xs:complexType>  
 </xs:element>  
 </xs:sequence>  
 </xs:complexType>  
 <xs:key name="userKey">  
 <xs:selector xpath="userType/id"></xs:selector>  
 <xs:field xpath="."></xs:field>  
 </xs:key>  
 <xs:key name="bookKey">  
 <xs:selector xpath="bookType/ISBN"></xs:selector>  
 <xs:field xpath="."></xs:field>  
 </xs:key>  
 <xs:key name="bookInstanceKey">  
 <xs:selector xpath="bookInstanceType/inventoryNo"></xs:selector>  
 <xs:field xpath="."></xs:field>  
 </xs:key>  
 <xs:key name="borrowingKey">  
 <xs:selector xpath="borrowingType/id"></xs:selector>  
 <xs:field xpath="."></xs:field>  
 </xs:key>  
 <xs:keyref refer="bookKey" name="instanceToBookRef">  
 <xs:selector xpath="bookInstanceType/bookISBN"></xs:selector>  
 <xs:field xpath="."></xs:field>  
 </xs:keyref>  
 <xs:keyref refer="bookInstanceKey" name="borrowingToInstanceRef">  
 <xs:selector xpath="borrowingType/bookInstanceInventoryNo"></xs:selector>  
 <xs:field xpath="."></xs:field>  
 </xs:keyref>  
 <xs:keyref refer="userKey" name="readerToUserRef">  
 <xs:selector xpath="readerType/userID"></xs:selector>  
 <xs:field xpath="."></xs:field>  
 </xs:keyref>  
 <xs:keyref refer="userKey" name="librarianToUser">  
 <xs:selector xpath="librarianType/userID"></xs:selector>  
 <xs:field xpath="."></xs:field>  
 </xs:keyref>  
 </xs:element>  
</xs:schema>