## Adatbázis

Adatok

Adatbázis szerver: **MYSQL**

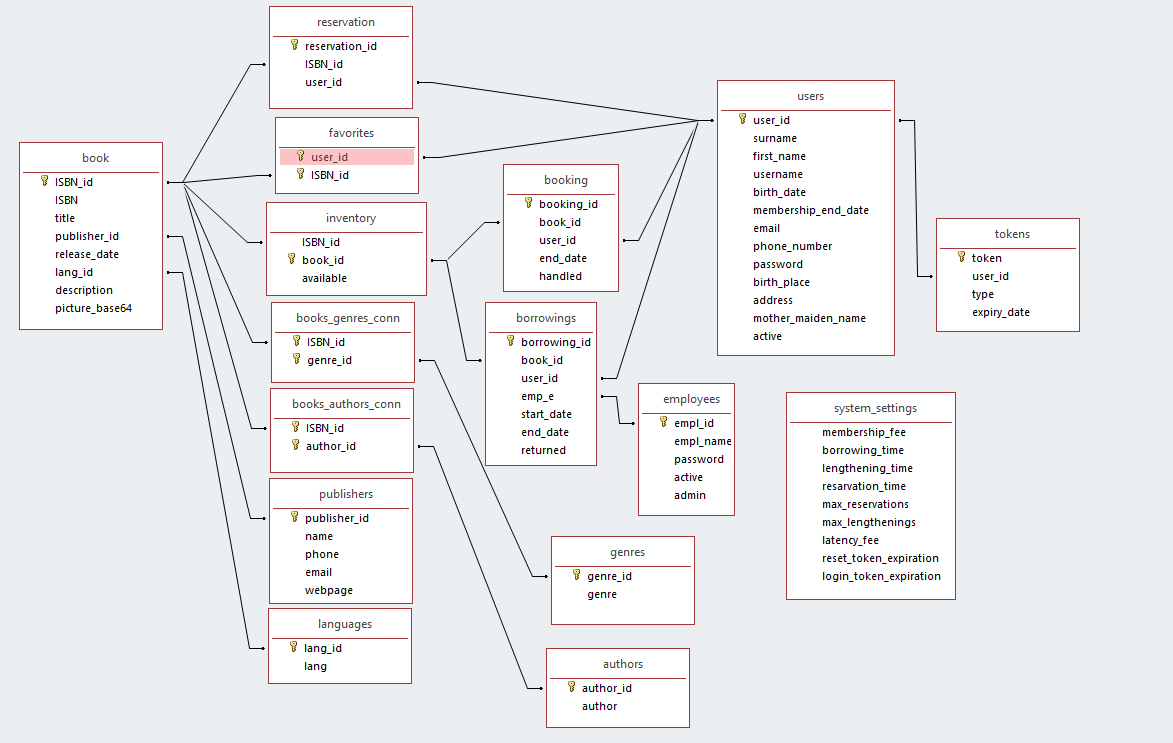
Adatbázis neve**: konyvtar**

Adatbázis illesztés: **utf8\_hungarian\_ci**

Adatbázis motor**: InnoDb**

Adatbázis létrehozó parancs:

CREATE DATABASE IF NOT EXISTS konyvtar CHARACTER SET utf8 COLLATE utf8\_hungarian\_ci;

Kapcsolatok

Táblák

A „book” tábla

A „book” tábla tárolja az egyes könyv típusok fontos adatait, amik elsősorban a weboldalon fognak megjelenni, az egyik legfontosabb és leggyakrabban használt tábla

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| ISBN\_id | INT | A könyvtípus egyedi azonosítója | primary key |
| ISBN | VARCHAR | 2007.01.01-ig megjelent könyveknél 11 karakter utána 13 |  |
| title | VARCHAR | A könyv címe |  |
| publisher\_id | INT | A könyv kiadójának azonosítója | foreign key |
| release\_date | DATE | A könyv megjelenési dátuma |  |
| lang\_id | INT | A könyv nyelvének azonosítója | foreign key |
| description | TEXT | A könyv rövid leírása |  |
| picture\_base64 | TEXT | HTML által felismert base64 formátumban vannak a képek tárolva |  |

SQL parancs

CREATE TABLE IF NOT EXISTS book(

ISBN\_id INT AUTO\_INCREMENT PRIMARY KEY,

ISBN VARCHAR(13),

title VARCHAR(255),

publisher\_id INT,

release\_date DATE,

lang\_id INT,

description TEXT,

picture\_base64 TEXT,

FOREIGN KEY(lang\_id) REFERENCES languages(lang\_id),

FOREIGN KEY(publisher\_id) REFERENCES publishers(publisher\_id)

);

A „languages” tábla

A „languages” tábla tárolja a könyvek nyelveit

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| lang\_id | INT | A könyv nyelvének egyedi azonosítója | primary key |
| lang | VARCHAR | A könyv nyelve |  |

SQL parancs

CREATE TABLE IF NOT EXISTS languages(

lang\_id INT AUTO\_INCREMENT PRIMARY KEY,

lang VARCHAR(255)

);

A „publishers” tábla

A „publishers” tábla a könyvekhez tartozó kiadók adatait tárolja, elsősorban a különféle elérhetőségeket

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| publisher\_id | INT | A kiadó egyedi azonosítója | primary key |
| name | VARCHAR | A kiadó neve |  |
| phone | VARCHAR | A kiadó telefonszáma |  |
| email | VARCHAR | A kiadó e-mail elérhetősége |  |
| webpage | VARCHAR | A kiadó weboldalálnak címe |  |

SQL parancs

CREATE TABLE IF NOT EXISTS publishers(

publisher\_id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255),

phone VARCHAR(255),

email VARCHAR(255),

webpage VARCHAR(255)

);

A „genres” tábla

A „genres” tábla a könyvekhez tartozó kategóriákat tartalmazza

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| genre\_id | INT | A kategória egyedi azonosítója | primary key |
| genre | VARCHAR | A kategória megnevezése |  |

SQL parancs

CREATE TABLE IF NOT EXISTS genres(

genre\_id INT AUTO\_INCREMENT PRIMARY KEY,

genre VARCHAR(255)

);

A „books\_genres\_conn” tábla

A „books\_genres\_conn” egy több-több kapcsolatú kapcsolótábla, ami a könyveket kapcsolja össze a hozzá tartozó kategóriákkal

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| ISBN\_id | INT | Könyv egyedi azonosítója | foreign key |
| genre\_id | INT | Kategória egyedi azonosítója | foreign key |

SQL parancs

CREATE TABLE IF NOT EXISTS books\_genres\_conn(

ISBN\_id INT NOT NULL,

genre\_id INT NOT NULL,

FOREIGN KEY(ISBN\_id) REFERENCES book(ISBN\_id),

FOREIGN KEY(genre\_id) REFERENCES genres(genre\_id)

);

Az „authors” tábla

Az „authors” tábla a könyvekhez tartozó szerzőket tartalmazza

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| author\_id | INT | A szerző egyedi azonosítója | primary key |
| author | VARCHAR | A szerző neve |  |

SQL parancs

CREATE TABLE IF NOT EXISTS authors(

author\_id INT AUTO\_INCREMENT PRIMARY KEY,

author VARCHAR(255)

);

A „books\_authors\_conn” tábla

A „books\_authors\_conn” egy több-több kapcsolatú kapcsolótábla, ami a könyvtípusokat a hozzá tartozó szerzőkkel kapcsolja össze

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| ISBN\_id | INT | A könyv egyedi azonosítója | foreign key |
| author\_id | INT | A szerző egyedi azonosítója | foreign key |

SQL parancs

CREATE TABLE IF NOT EXISTS books\_authors\_conn(

ISBN\_id INT NOT NULL,

author\_id INT NOT NULL,

FOREIGN KEY(ISBN\_id) REFERENCES book(ISBN\_id),

FOREIGN KEY(author\_id) REFERENCES authors(author\_id)

);

Az „inventory” tábla

Az „inventory” tábla az adott könyvtípushoz tartozó tényleges könyveket tárolja egy-több kapcsolatú kapcsolótábla

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| ISBN\_id | INT | A könyvtípus egyedi azonosítója | foreign key |
| book\_id | INT | A könyv egyedi azonosítója | primary key |
| available | BOOLEAN | Elérhetőségi flag. Ha hamis, a könyv nem elérhető, le lett selejtezve |  |

SQL parancs

CREATE TABLE IF NOT EXISTS inventory(

ISBN\_id INT NOT NULL,

book\_id INT AUTO\_INCREMENT PRIMARY KEY,

available BOOLEAN,

FOREIGN KEY(ISBN\_id) REFERENCES book(ISBN\_id)

);

A „users” tábla

A „users” tábla tárolja az olvasók/felhasználók összes személyes adatát. Szintén egy gyakran használt tábla

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| user\_id | INT | Felhasználó egyedi azonosítója | primary key |
| surname | VARCHAR | Vezetékné |  |
| firstname | VARCHAR | Keresztnév |  |
| username | VARCHAR | Felhasználónév, automatikusan generált (vezeték- és keresztév első két betűje + 3 random generált szám) |  |
| birth\_date | DATE | Születési dátum |  |
| membership\_end\_date | DATE | A felhasználó tagságának lejárati dátuma |  |
| email | VARCHAR | Felhasználó e-mail címe |  |
| phone\_number | INT | Felhasználó telefonszáma |  |
| password | VARCHAR | Felhasználó jelszava, a MYSQL PASSWORD nevű függvényével lett titkosítva |  |
| birth\_place | VARCHAR | Születési hely |  |
| address | VARCHAR | Jelenlegi lakcíme |  |
| mother\_maiden\_name | VARCHAR | Anyja lánykori neve |  |
| active | BOOLEAN | Deaktivált felhasználók úgy viselkednek mintha törölve lett volna a fiók |  |

SQL parancs

CREATE TABLE IF NOT EXISTS users (

user\_id INT AUTO\_INCREMENT PRIMARY KEY,

surname VARCHAR(255),

first\_name VARCHAR(255),

username VARCHAR(7),

birth\_date DATE,

membership\_end\_date DATE,

email VARCHAR(255),

phone\_number VARCHAR(255),

password VARCHAR(255),

birth\_place VARCHAR(255),

address VARCHAR(255),

mother\_maiden\_name VARCHAR(255),

active BOOLEAN

);

A „reservation” tábla

A „reservation” tábla tárolja az aktuális előjegyzéseket, több-több kapcsolatú kapcsolótáblában, néhány művelet könnyítése érdekeben kapott egy egyedi azonosítót is.

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| reservation\_id | INT | Az előjegyzés egyedi azonosítója | primary key |
| user\_id | INT | A felhasználó egyedi azonosítója | foreign key |
| ISBN\_id | INT | A könyvtípus egyedi azonosítója | foreign key |

SQL parancs

CREATE TABLE IF NOT EXISTS reservation(

reservation\_id INT AUTO\_INCREMENT PRIMARY KEY,

ISBN\_id INT NOT NULL,

user\_id INT NOT NULL,

FOREIGN KEY(ISBN\_id) REFERENCES book(ISBN\_id),

FOREIGN KEY(user\_id) REFERENCES users(user\_id)

);

A „shelf” tábla

A „shelf” tábla, a kedvencek funkció. Tárolja a felhasználók kedvenc könyveit. Több-több kapcsolatú kapcsolótábla.

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| user\_id | INT | A felhasználó egyedi azonosítója | foreign key |
| ISBN\_id | INT | A könyvtípus egyedi azonosítója | foreign key |

SQL parancs

CREATE TABLE IF NOT EXISTS shelf(

user\_id INT,

ISBN\_id INT,

PRIMARY KEY(user\_id, ISBN\_id),

FOREIGN KEY(user\_id) REFERENCES users(user\_id),

FOREIGN KEY(ISBN\_id) REFERENCES book(ISBN\_id)

);

Az „employees” tábla

Az „employees” tábla tárolja az összes alkalmazott adatait

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| empl\_id | INT | Az alkalmazott egyedi azonosítója | primary key |
| empl\_name | VARCHAR | Az alkalmazott neve |  |
| empl\_uname | VARCHAR | Az alkalmazott felhasználóneve automatikusan generált, mint a felhasználók nevei |  |
| password | VARCHAR | Az alkalmazott jelszava az asztali alkalmazáshoz, a MYSQL PASSWORD függvényével titkosítva |  |
| active | BOOLEAN | Deaktivált fiókot nem lehet újra aktiválni, a deaktivált alkalmazottak nem tudnak bejelentkezni |  |
| admin | BOOLEAN | Admin flag, admin alkalmazottból csak egy létezhet és nem lehet deaktiválni |  |

SQL parancs

CREATE TABLE IF NOT EXISTS employees(

empl\_id INT AUTO\_INCREMENT PRIMARY KEY,

empl\_name VARCHAR(255),

empl\_uname VARCHAR(5),

password VARCHAR(255),

active BOOLEAN,

admin BOOLEAN

);

A „borrowings” tábla

A „borrowings” tábla tárolja az összes aktív és már lejárt/befejezett könyv kivételt. Több-több kapcsolatú kapcsolótábla, műveletek egyszerűsítése érdekében saját egyedi azonosítót is kapott

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| borrowing\_id | INT | Kivétel egyedi azonosítója | primary key |
| book\_id | INT | Könyv egyedi azonosítója | foreign key |
| user\_id | INT | Felhasználó egyedi azonosítója | foreign key |
| empl\_e | INT | Alkalmazott egyedi azonosítója. Amíg nincs visszahozva, azt az alkalmazottat tárolja, aki kiadta, ha már vissza lett véve, akkor azt tárolja, aki visszavette | foreign key |
| start\_date | DATE | Kivétel kezdetének dátuma |  |
| end\_date | DATE | Vége dátum, amíg nincs visszahozva a várt dátumot tárolja, visszahozást követően a visszavétel dátumát |  |
| returned | BOOL | Vissza lett-e már hozva flag |  |

SQL parancs

CREATE TABLE IF NOT EXISTS borrowings(

borrowing\_id INT AUTO\_INCREMENT PRIMARY KEY,

book\_id INT NOT NULL,

user\_id INT NOT NULL,

empl\_e INT NOT NULL,

start\_date DATE,

end\_date DATE,

returned TINYINT(1),

FOREIGN KEY(book\_id) REFERENCES inventory(book\_id),

FOREIGN KEY(user\_id) REFERENCES users(user\_id),

FOREIGN KEY(empl\_e) REFERENCES employees(empl\_id)

);

A „booking” tábla

A „booking” táblában található az összes aktív könyv foglalás, ha egy foglalás lejár automatikusan kitörlődik a táblából. Több-több kapcsolatú kapcsolótábla, néhány művelet végett kapott saját egyedi azonosítót.

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| booking\_id | INT | Foglalás saját azonosítója | primary key |
| user\_id | INT | Felhasználó saját azonosítója | foreign key |
| book\_id | INT | Könyv saját azonosítója | foreign key |
| end\_date | DATE | Foglalás lejárati dátuma |  |
| handled | INT | Állapot flag, három állapot 1 <- új foglalás, a könyvet félre kell tenni, 0 <- a könyvvel nincs semmi tennivaló, -1 <- foglalás törölve lett a könyvet vissza kell tenni a helyére |  |

SQL parancs

CREATE TABLE IF NOT EXISTS booking(

booking\_id INT AUTO\_INCREMENT PRIMARY KEY,

book\_id INT NOT NULL,

user\_id INT NOT NULL,

end\_date DATE,

handled TINYINT DEFAULT 0,

FOREIGN KEY(book\_id) REFERENCES inventory(book\_id),

FOREIGN KEY(user\_id) REFERENCES users(user\_id)

);

A „system\_settings” tábla

A „system\_settigns” tábla a könyvtár rendszeréhez tartozó alapbeállításokat tárolja nem kapcsolódik más táblához szimplán egységes tárolás végett létezik

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| membership\_fee | INT | Tagság díja, mértékegység: forint |  |
| borrowing\_time | INT | Kivétel ideje, mértékegység: nap |  |
| lengthening\_time | INT | Hosszabbítás ideje, mértékegység: nap |  |
| booking\_time | INT | Foglalás ideje, mértékegység: nap |  |
| max\_reservations | INT | Maximum előjegyzések száma per felhasználó, mértékegység: darab |  |
| max\_lengthenings | INT | Maximum kivétel hosszábbítás per felhasználó, mértékegység: darab |  |
| latency\_fee | INT | Késedelemi bírság per késett nap, mértékegység: forint |  |
| reset\_token\_expiration | INT | Reset token élettartama, mértékegység: óra |  |
| login\_token\_expiration | INT | Login token élettartama, mértékegység: nap |  |

SQL parancs

CREATE TABLE IF NOT EXISTS system\_settings (

membership\_fee INT,

borrowing\_time INT,

lengthening\_time INT,

booking\_time INT,

max\_reservations INT,

max\_lengthenings INT,

latency\_fee INT,

reset\_token\_expiration INT,

login\_token\_expiration INT

);

A „tokens” tábla

A „tokens” tábla emlékezz rám és elfelejtett jelszó funkciókhoz tárol tokeneket.

Szerkezet

|  |  |  |  |
| --- | --- | --- | --- |
| Mező neve | Mező típusa | Leírás | Kulcs/index |
| token | VARCHAR | A generált egyedi token | primary key |
| user\_id | INT | A felhasználó egyedi azonosítója | foreign key |
| type | VARCHAR | A token típusa remember\_me/reset |  |
| exipry\_date | DATE | A token lejárati dátuma |  |

SQL parancs

CREATE TABLE IF NOT EXISTS tokens(

token VARCHAR(11) PRIMARY KEY,

user\_id INT NOT NULL,

expiry\_date DATETIME,

type VARCHAR(255),

FOREIGN KEY(user\_id) REFERENCES users(user\_id)

);

Eljárások

A projekt összes adatbázissal kapcsolatos lekérdezése és parancsa eljárásokkal (stored procedure) van megvalósítva. Mivel minden lekérdezés, hozzáadás, törlés és módosítás procedure-ként van tárolva így, minden átlátható, könnyen módosítható és tesztelhető, illetve újra használható akár egy másik eljáráson belül.

Egy másik előnye az eljárásoknak, hogy egy php file segítségével egy Sql injection biztos php metódust lehet belőle csinálni. Az adatbázissal való kommunikációt kizárólag ilyen metódusokkal sikerült megoldani. Az alábbi kódrészlet a projektben megvalósított metódusokra példa

function GetBorrowedBooks($user\_id){

        require "databaseConnect.php";

        $query = "CALL getBorrowedBooks(?);";

        $stmt = $conn->prepare($query);

        $stmt->bind\_param("i", $user\_id);

        $stmt->execute();

        $results = $stmt->get\_result();

        $conn->close();

        return $results->fetch\_all(MYSQLI\_ASSOC);

}

Eljárások felsorolása

|  |  |  |  |
| --- | --- | --- | --- |
| Név | Bemenet | Kimenet | Leírás |
| listBooksFiltered | title, genres, authors,  release date,  language, ISBN, offseter, limiter | ISBN\_id, title, picture base64, genres, authors, publisher, release date, language, ISBN, description | Kilistázza a filtereknek megfelelő könyvek adatait, oldalakra bontva |
| getBookByISBN | ISBN | ISBN\_id, title, picture base64, genres, authors, publisher, release date, language, ISBN, description | Kilistázza egy adott ISBN kódú könyv összes adatát |
| getGenres |  | genres | Kilistázza az összes adatbázisban létező kategóriát |
| getLangs |  | languages | Kilistázza az összes adatbázisban létező nyelvet |
| getIdFromISBN | ISBN | ISBN\_id | A megadott ISBN kódot ISBN\_id-ra fordítja |
| getBookById | ISBN\_id | ISBN\_id, title, picture base64, genres, authors, publisher, release date, language, ISBN, description | Kilistázza egy ISBN\_id alapján megadott könyv összes adatát |
| getBookedAndReservedBooks | user\_id | title, authors, state, reservation id/booking id | Kilistázza egy adott felhasználó összes foglalását és előjegyzését |
| cancelReservation | reservation\_id, user\_id |  | Törli egy adott felhasználó előjegyzésést |
| cancelBooking | booking\_id, user\_id |  | Módosítja az adott foglalás handled értékét és ha van előjegyzés az adott könyvre akkor a legrégebbit teszi új foglalásnak |
| removeBooking | booking\_id |  | Törli az adott foglalást |
| getBorrowedBooks | user\_id | book\_id, auhtors, ISBN, title, end date, start date, meghosszabítható, fizetendő | Kilistázza egy felhasználó a jelenleg kivett könyveinek adatait pl.: a késésből jövő fizetendőt és hosszabbítható-e |
| getPreviouslyBorrowedBooks | user\_id | book\_id, authors, ISBN, title, picture base64, end date, start date, | Kilistázza egy felhasználó már korábban kivett, de már visszahozott könyveinek néhány adatát |
| deleteLateBookings |  | output | Törli a lejárt foglalásokat illetve a lejárt tagsággal rendelkező felhasználók foglalásait és új foglalást add hozzá ha van előjegyezve output mezőben a törölt book\_id és user\_id , ha lett hozzáadva új foglalás a hozzá tartozó user\_id, végül a törlés okát |
| deleteLateReservations |  | output | Törli azon felhasználók előjegyzéseit, akiknek a tagsága lejárt, output mezőben a törölt előjegyzések adatai user\_id, ISBN\_id |
| loginUser | username, password | result | Ellenőrzi a felhasználó bejelentkezési adatait és a megfelelő választ adja rá |
| getUserId | username | user\_id | A username-hez tartozó user\_id-t adja vissza |
| getEmailData | user\_id | email, surname, first name | Egy user mail küldéshez szükséges adatainak kiírása |
| updateUserDetails | user\_id, surname, first name, birth place, birth date, address, email, phone number, mothers maiden name(mmn) |  | Módosítja egy felhasználó adatait |
| updateBookDetails | ISBN\_id, title, release date, language, publisher, authors, genres, description, picture base64 |  | Módosítja egy könyv adatait |
| getUser | user\_id | surname, first name, username, birth place, birth date, email, phone number, address, mothers maiden name(mmn), membership end date, active | Adott user\_id alapján megadja a felhasználó összes adatát |
| getEmployee | empl\_id | name, username, admin | Adott empl\_id alapján megadja a felhasználó összes adatát |
| checkAvailability | ISBN\_id, user\_id | status, available | Egy adott könyv-felhasználó kombinációra adja meg, hogy előjegyzés vagy foglalás lenne ->status, illetve egyáltalán lehetséges-e ->available |
| addReservationOrBooking | ISBN\_id, user\_id | selected | Vagy az előjegyzés vagy a foglalás táblába rakja a könyv-felhasználó párost és visszaadja melyikbe tette |
| extendReturnDate | book\_id, user\_id |  | Meghosszabítja az adott kivétel lejárati dátumát |
| extendable | book\_id, user\_id | result | Visszaadja, hogy a megadott kivétel hosszabítható-e |
| deleteExpiredTokens |  |  | Törli a lejárt tokeneket a táblából |
| generateToken | user\_id, token, type | result | Generál egy új tokent a táblába a megadott adatok alapján és visszaküldi, hogy sikerült-e |
| changePassword | password, user\_id |  | Megváltoztatja az adott felhasználó jelszavát |
| changeEmplPass | empl\_id, old pasword, new password | result | Megváltoztatja az adott alkalmazott jelszavát és visszaküldi, hogy sikerült-e |
| deleteToken | token |  | Törli az adott tokent-t az adatbázisból |
| addBookType | ISBN, title, genres, authors, publisher, release date, lang, description, picture base64 | state | Hozzáad egy új könyvet ha már létezik a könyv azt jelzi |
| loginEmployee | username, password | info | Az alkalmazott bejelentkezési adatait nézi, ha jó az empl\_id-t ad vissza |
| getUsers | searchTerm | user\_id, surname, first name | Kilistázza a keresésnek megfelelő felhasználók néhány adatát |
| getEmployees |  | empl\_id, name, username | Kilistázza az aktív alkalmazottak adatait |
| getBooks | searchTerm | booki\_id, ISBN, cím, available, status | Kilistázza a keresésnek megfelelő könyvek néhány adatát, illetve, hogy jelenleg mi a státuszuk (elérhető/kivéve/foglalva) |
| getPublishers | searchTerm | publisher\_id, name, phone number, email, website | Kilistázza a keresésnek megfelelő kiadók összes adatát |
| updatePublishers | publisher\_id, name, phone number, email, website |  | Módosítja egy adott kiadó adatait |
| deletePublisher | publisher\_id | state | Törli az adott kiadót, ha már nincs egy könyv kiadójaként sem megadva és visszaküldi, hogy sikeres volt-e |
| borrowInfo | book\_id, state | title, ISBN, name, username, user\_id | A state-től függően ad vissza adatokat egy új kivétellelhez („booking” -> előre le volt foglalva, minden adatot visszaad, „book\_id” -> csak a könyv adatait, „user\_id” -> csak a felhasználó adatait adja vissza) |
| borrowBook | user\_id, book\_id, empl\_id | state | Véglegesít egy kivételt és visszajelez, hogy sikeres-e |
| returnBook | user\_id, book\_id, empl\_id | new | Visszavesz egy kiadott könyvet és ha volt rá előjegyzés, akkor a legrégebbit foglalásra váltja |
| returnInfo | book\_id | title, ISBN, name, user\_id, end date, late | Könyv visszavételéhez szükséges adatokat adja vissza |
| addPublisher | name, phone number, email, website | state | Hozzáad egy új kiadót az adatbázishoz és visszaküldi, hogy sikeres volt-e a művelet |
| addUser | surname, first name, username, birth date, birth place, email, phone number, address, mothers maiden name(mmn) | state | Hozzáad egy új felhasználót, és visszaküldi, hogy sikeres volt-e a művelet |
| addEmployee | name, username, jelszó |  | Hozzáad egy új alkalmazottat |
| renewMembership | user\_id | end\_date | Meghosszabítja az adott felhasználó tagságát egy évvel és visszaadja az új lejárati dátumot |
| getSystemSettings |  | membeship fee, borrowing time, lengthening time, booking time, max reservations, max lengthenings, latency fee, reset token expiraiton, login token expiration | Kilistázza az összes rendszer beállítást |
| deactiveUser | user\_id |  | Deaktivál egy felhasználót |
| deactiveEmpl | empl\_id |  | Deaktivál egy alkalmazottat |
| checkPermissions | user\_id | reservation, booking | Megadja, hogy az adott felhasználó tud-e előjegyezni és foglalni |
| getBookStates | search | ISBN\_id, ISBN, title, state | Kilistázza a keresésnek megfelelő könyvek néhány adatát és, hogy foglalható-e, vagy csak előjegyezhető |
| getUsernames | name | username | A felhasználónév generálásához kilistázza a már létező, hasonló felhasználóneveket |
| getEmplUsernames | name | username | Az alkalmazott felhasználónév generálásához kilistázza a már létező, hasonló alkalmazotti felhasználóneveket |
| getHistory | book\_id, user\_id | result | Kivételnél megmondja, hogy a felhasználó olvasta-e már az adott könyvet, ha igen azt is megmondja hány napja olvasta |
| getAuthors | ISBN\_id, inverse | author\_id, author | Kilistázza egy könyv szerzőit, ha az inverse igaz akkor azokat a szerzőket listázza ki, amik nem tartoznak a könyvhöz |
| getCategories | ISBN\_id, inverse | genre\_id, genre | Kilistázza egy könyv kategóriáit, ha az inverse igaz akkor azokat a kategóriákat listázza ki, amik nem tartoznak a könyvhöz |
| deactivateBook | book\_id |  | Deaktiválja az adott könyvet |
| addToInventory | ISBN\_id | book\_id | Hozzáad egy adott típusú könyvet a raktárhoz és megadja az azonosítóját |
| getReservations | searchTerm | reservation\_id, username, title | Kilistázza a keresésnek megfelelő előjegyzéseket |
| getBookings | searchTerm | booking\_id, username, title, end date handled | Kilistázza a keresési feltételnek megfelelő foglalások összes adatát |
| getBorrowings | searchTerm, returnedToo | borrowing\_id, username, title | Kilistázzat a keresési feltételnek megfelelő kivételeknek adatait, ha returnedToo igaz akkor a visszavitt kivételek között is keres |
| updateSystemSettings | membeship fee, borrowing time, lengthening time, booking time, max reservations, max lengthenings, latency fee, reset token expiraiton, login token expiration |  | Megváltoztatja a rendszer beállításokat |
| getBooksFromFavorites | user\_id | title, ISBN\_id, authors, ISBN, picture base64 | Kilistázza az adott felhsználó kedvenceit |
| addBooksToFavorites | user\_id, ISBN\_id |  | Hozzáad egy könyvet a kedvencekhez |
| removeBooksFromFavorites | user\_id, ISBN\_id |  | Töröl egy könyvet a kedvencek közül |
| isItOnFavorites | user\_id, ISBN\_id | result | Megnézi, hogy az adott könyv a kedvencek között van-e |
| getSimilarBooks | ISBN\_id | ISBN, title, picture base64, authors, genres | A megadott könyvhöz hasonló könyveket listáz ki |
| getReservables | searchTerm | ISBN\_id, ISBN, title, state | Kilistázza a keresésnek megfelelő csak előjegyezhető könyveket |
| checkForBookings |  | result | Nézi, hogy jött-e új foglalás |
| changeHandled | booking\_id |  | Megváltoztatja az adott foglaláson a handled értéket |