Adatbázis

Adatok

Adatbázis szerver: **MYSQL**

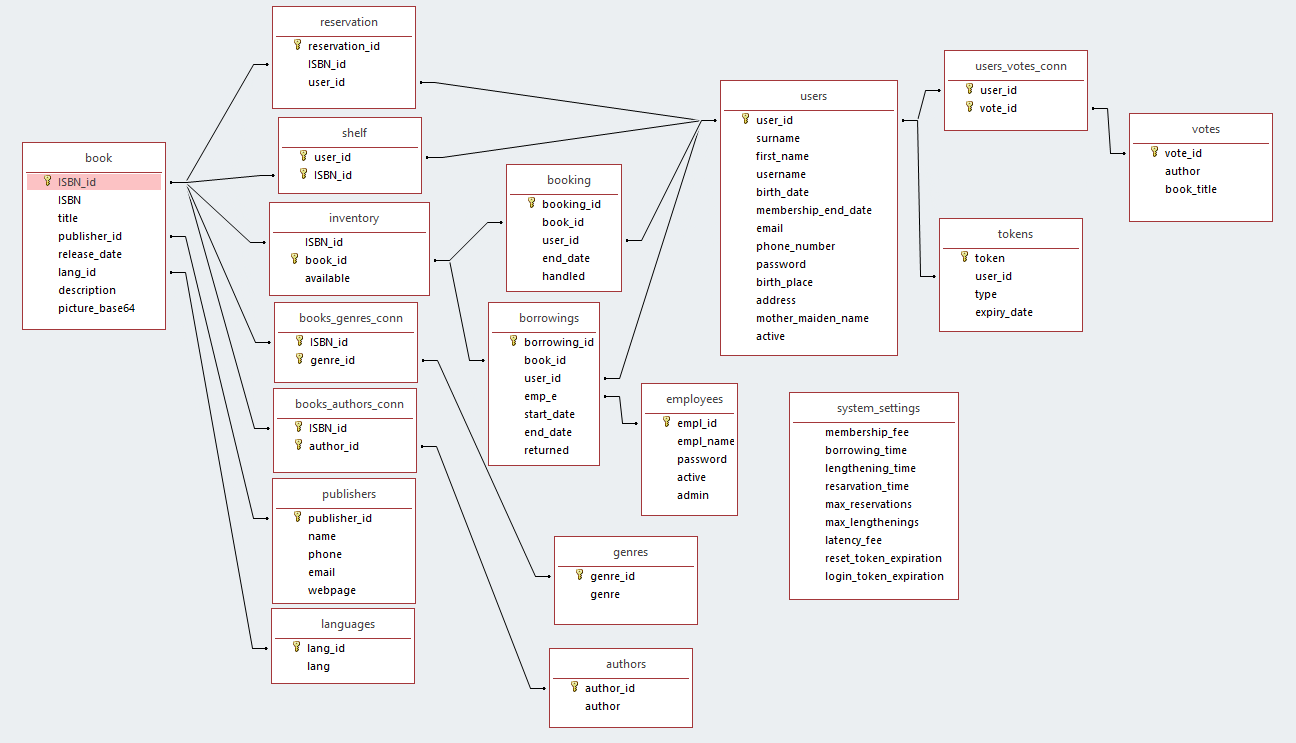
Adatbázis neve**: konyvtar**

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)

);