**Database Documentation**

**Import Instructions:**

HeidiSQL installs with MariaDB automacially.  
In HeidiSQL choose a session (new or existing) -> Advanced tab -> Startup script -> EfiLibrary1.sql file.

The sql file can be found in teams under database folder.

(Full table creation, inserting queries and table dropping queries are in the bottom of the document.)

**Relation Schema:**

Graphical user interface, application, Word

Description automatically generated

**Tables:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | library\_user | | | |
| **Definition** | A user of the software | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | id |  | integer | PK |
|  | username | Username to log in | varchar(50) | NOT NULL |
| Type length depends on encryption | passw | Password to log in | varchar(150) | NOT NULL |
|  | administrator | Determines admin status | Bit(1) |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | book | | | |
| **Definition** | A book that’s been registered to the library | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | id |  | integer | PK |
|  | library\_user | id of the user who owns the book | integer | FK |
|  | topic | Books topic taken from Topic | varchar(50) | FK |
|  | title | Name of book | varchar(250) | NOT NULL |
|  | author | Name of author | varchar(250) | NOT NULL |
|  | isbn | The books ISBN | varchar(20) | NOT NULL |
|  | location | Location of book | varchar(20) | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | borrowing | | | |
| **Definition** | A borrow card that shows if a book is borrowed, by who and until when. | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | id |  | integer | PK |
|  | library\_user | id of the user who is borrowing the book | integer | FK |
|  | book | id of the book that’s being borrowed | integer | FK |
|  | dueDate | Date of return | date | NOTNULL |
|  | borrowDate | Date of borrow | date | NOT NULL |
|  | returned | Is the borrow completed or not | bit(1) | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | recommendation | | | |
| **Definition** | A recommendation that is given to a book by a user | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | id |  | integer | PK |
|  | book | id of the book that’s being rated | integer | FK |
|  | library\_user | id of user who is doing the recommendation | integer | FK |
|  | recommendation | Is the book recommended or not | bit(1) | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | reservation | | | |
| **Definition** | A reservation of a book made by a user | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | id |  | integer | PK |
|  | library\_user | id of the user who is doing the reserving | integer | FK |
|  | book | id of book that is being reserved | integer | FK |
|  | date | The beginning date of the reservation | date | NOT NULL |
|  | duration | The duration of the reservation in days(?) | integer |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | book\_list | | | |
| **Definition** | A list of books made by a user | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | id |  | integer | PK |
|  | library\_user | id of the user that owns the list | integer | FK |
|  | name | Name of the list | varchar(20) |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | topic | | | |
| **Definition** | Contains topics for books | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | topic | Topic names for books | varchar(50) | PK |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | book\_list\_entry | | | |
| **Definition** | Helper table that binds books to different lists | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | id |  | integer | PK |
|  | book | id of a book that is being binded | integer | FK |
|  | list | id of a list the book is in | integer | FK |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | keyword | | | |
| **Definition** | Table contains a list of keywords that books can be labelled with. Useful for keyword searches | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | keyword | The keyword to label books with | varchar(30) | PK |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | book\_keyword | | | |
| **Definition** | Helper table that binds keywords to books | | | |
| **Properties** | **Name** | **Description** | **Type** | **PK/FK/NOT NULL** |
|  | id |  | integer | PK |
|  | book | id of a book taken from Book table | integer | FK |
|  | keyword | keyword taken from Keyword table | Varchar(30) | FK |

CREATE TABLE Queries

**CREATE** **TABLE** book (

id **INTEGER** **PRIMARY** **KEY**,

library\_user **INTEGER** **NOT** **NULL**,

topic **VARCHAR**(50) **NOT** **NULL**,

title **VARCHAR**(250) **NOT** **NULL**,

author **VARCHAR**(250) **NOT** **NULL**,

isbn **VARCHAR**(20) **NOT** **NULL**,

location **VARCHAR**(20) **NOT** **NULL**

);

**CREATE** **TABLE** library\_user (

id **INTEGER** **PRIMARY** **KEY**,

username **VARCHAR**(50) **NOT** **NULL**,

passw **VARCHAR**(150) **NOT** **NULL**,

administrator **BOOLEAN**

);

**ALTER** **TABLE** library\_user **ADD** **CONSTRAINT** UQ\_libary\_user\_username **UNIQUE**(username);

**ALTER** **TABLE** book **ADD** **CONSTRAINT** FK\_book\_library\_user **FOREIGN** **KEY**(library\_user) **REFERENCES** library\_user(id);

INSERT INTO Queries

**INSERT** **INTO** library\_user **VALUES** (

1, "mikkomallikas", "1234", **TRUE**

);

**INSERT** **INTO** book **VALUES** (

1, 1, "JS", "JS For Dummies", "JS Teacher", "12345-67-89", "MERICA"

);

DROP TABLE Queries

**DROP TABLE** book;

**DROP TABLE** library\_user;