

Datenbanksysteme

1. Praktisches Projekt

Einführung

Im Laufe des praktischen Projekts sollen Sie ein System zur Verwaltung von Einträgen einer Bibliothek entwickeln. Die Entwicklung dieses Projekts soll dabei in vier Schritten ablaufen:

1. ER-Modellierung der Datenbank
2. Überführung der ER-Modellierung in ein relationales Modell
3. Implementierung der Datenbank in SQLite
4. Umsetzung eines dazugehörigen RESTful Web Services

Die einzelnen Arbeitsschritte bauen hierbei aufeinander auf. Die aktuelle Aufgabenstellung wird im letzten Abschnitt des Kapitels „Aufgabenstellung“ beschrieben.

Wichtige allgemeine Hinweise

Für die ersten zwei Schritte haben Sie jeweils einen Bearbeitungszeitraum von einer Woche, für die Implementierung der Datenbank in SQLite und die abschließende Umsetzung eines dazugehörigen RESTful Web Services haben Sie jeweils zwei Wochen Zeit. Um die praktischen Übungen zu bestehen, müssen Sie *jeden* der vier Abschnitte bestehen. Dies bedeutet, dass Sie den zweiten Schritt nur bearbeiten können, wenn Sie den ersten *bestanden* haben. Klären Sie daher kritische Punkte und Fragen *vor* der Abgabe in den Sprechstunden, da es aufgrund des Zeitplans keine Nachbearbeitungszeit geben kann. Näheres zu der Veranstaltung finden Sie auf unserer [Homepage](#).

Sie müssen alle Arbeitsschritte *alleine* und selbstständig bearbeiten. Gruppenarbeiten, auch bei nicht ganz identischen Abgaben, führen zum Ausschluss aller Beteiligten. Die Abgabe dieses dritten Teils ist bis Montag, den 22.02.2021, um 10:00 Uhr über das Ilias möglich. Achten Sie auf die **aktuelle Aufgabenstellung** am Ende des Übungsblattes.

Anwendungsszenario

Das Ziel der gesamten praktischen Übung besteht in der Implementierung eines Systems zur Verwaltung von Einträgen einer Bibliothek. Die explizite Aufgabe für dieses Blatt finden Sie weiter unten.

Es sollen folgende Sachverhalte dargestellt werden:

In unserer Datenbank sind **Nutzer** hinterlegt. Jeder Nutzer hat einen Vornamen, einen Nachnamen, ein Geburtsdatum, eine eindeutige E-Mail Adresse, sowie ein Passwort. Das Passwort und die E-Mail Adresse dienen der Authentifizierung.

Ein **Bibliothekar** ist ein Nutzer. Dieser besitzt eine einmalige Telefonnummer.

Ein **Kunde** ist ebenfalls ein Nutzer. Er besitzt ein Guthaben (Betrag in €) und kann optional vom Jahresbeitrag befreit sein.

Der Kunde kann sich beliebig viele **Exemplare** eines **Artikels** ausleihen. Dabei wird in der Datenbank zu jeder **Ausleihe** ein Zeitraum der Ausleihe mit Beginn und Ende gespeichert, sowie ein Vermerk darüber, ob die Ausleihe zurückgegeben wurde. Eine Ausleihe bezieht sich auf ein Exemplar. Der Kunde kann sich beliebig viele Exemplare ausleihen. Jede Ausleihe kann beliebig oft verlängert werden.

Ein Kunde hat genau eine **Adresse**. Jede Adresse besteht aus der Angabe zu Stadt, Postleitzahl, Straße und Hausnummer. Es können beliebig viele Kunden dieselbe Adresse teilen.

Ein Exemplar ist ein **Artikel**, der zu einem bestimmten Anschaffungspreis erworben wurde und unter genau einem **Standort** zu finden ist. Dabei unterscheiden sich die Standorte durch einen Vermerk zur Etage, sowie zum Regal (Regal Nr.).

Ein Artikel besitzt eine eindeutige ISBN Nummer, eine Bezeichnung, eine Beschreibung und ein optionales Coverbild. Zusätzlich liegt zu jedem Artikel ein Erscheinungsdatum vor.

Jeder Artikel wird in Form genau eines **Mediums** angeboten. Auf diesem Wege kann beispielsweise ein Buch als Softcover und Hardcover angeboten werden.

Jeder Artikel kann beliebig viele andere Artikel empfehlen, zeitgleich können zu einem Artikel beliebig viele Artikel empfohlen sein.

Jeder Artikel wurde von mindestens einem **Autor** verfasst. Ein Autor besitzt einen Vornamen und einen Nachnamen.

Jeder Artikel gehört zu beliebig vielen **Genres** (z. B.: Kriminalroman).

Eine detaillierte Auflistung der geforderten Funktionalität des endgültigen Programms folgt auf dem letzten Aufgabenblatt. Die Benutzer des Systems sollen unter anderem folgende Aktionen durchführen können:

- Ein neuer Nutzer soll die Möglichkeit haben sich zu registrieren.
- Ein Bibliothekar soll die Möglichkeit haben sich zu registrieren.
- Ein Bibliothekar soll Artikel anlegen und modifizieren können.
- Es soll möglich sein, nach Artikeln zu suchen.
- Es soll möglich sein, Artikel nach ihren Attributen filtern zu können.

Aufgabenstellung (3.Teil)

Geben Sie gegebenenfalls eine **korrigierte** Version Ihres Relationenmodells als PDF ab. Setzen Sie dieses dann in SQLite um und vermerken Sie die Version (empfohlen 3.16.2). Legen Sie für *jede* Tabelle mindestens 2 Beispieleinträge an und geben Sie abschließend die entstandene SQLite Datenbank (database.db) und die Statements zur Erzeugung und Füllung der Datenbank als reine Textdateien (schema.sql, data.sql) in einem ZIP-Archiv ab. Beachten Sie bitte hierbei, dass die Daten bereits beim Einfügen in die Datenbank kontrolliert werden müssen (z.B. Wertebereich) und nicht erst später in der Anwendung. Falls Sie vor der Abgabe Ihre Datenbank auf einem unserer Rechner testen möchten, benötigen Sie Zugangsdaten. Diese bekommen Sie in Ihrer Sprechstunde bei Ihrem Tutor.

Geben Sie weiterhin Anfragen in einer reinen Textdatei (queries.sql) an, die Folgendes auf Ihrer Datenbank beantworten:

- Geben Sie genau die Artikel an, zu denen mehr als 5 Exemplare existieren.
- Geben Sie genau die 3 Kunden mit dem höchsten Guthaben in aufsteigender Reihenfolge an.
- Geben Sie den Vornamen und die E-Mail-Adresse genau der Kunden an, die den Nachnamen „Musk“ tragen und bisher keinen Kriminalroman ausgeliehen haben.

Achten Sie bitte darauf, dass Ihre Datenbank für die Anfragen passende Einträge enthält und eine nicht leere Ergebnismenge liefert. Alle reinen Textdateien müssen vom SQLite-Konsolenprogramm gelesen werden können. Für kritische Entscheidungen nutzen Sie bitte das vorhandene Textfeld beim Einreichen der Abgabe. Hinweise und Mindestanforderungen zu den Wertebereichen finden Sie unter folgendem Link: [Hinweise](#)

Halten Sie sich bei der Einreichung Ihrer Abgabe an das bereits vom ersten Übungsblatt bekannte Format. Die Abgabe soll ein als ZIP-Archiv komprimierter Ordner mit Namen dbs-propra-ws2021-<vorname>-<nachname>, wobei <vorname> Ihr Vorname und <nachname> Ihr Nachname ist, und folgender Ordnerstruktur sein:

- phase1 (Ordner mit Inhalt aus Phase 1)
- phase2 (Ordner mit Inhalt aus Phase 2)
- phase3/rm-<vorname>-<nachname>_korrigiert.pdf (korrigiertes Relationenschema)

- phase3/README.adoc (kritische Entscheidungen und die verwendete SQLite-Version (mindestens 3.16.2)))
- phase3/schema.sql (nur SQLite-DDL-Anweisungen einschließlich Trigger)
- phase3/data.sql (nur SQLite-DML-Anweisungen)
- phase3/database.db (SQLite-Datenbank)
- phase3/queries.sql (nur SQLite-SELECT-Anweisungen für Anfragen aus der Aufgabenstellung in entsprechender Reihenfolge)