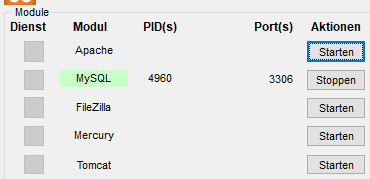
Teil 1 - Übung

# 1 Softwareinstallation

Für diese Übung benötige ich zwei Programme. Zum einen XAMPP und zum anderen SQL MySql Workbench. XAMP benötige ich in dieser Übung, um einen MySql-Datenbankserver zu installieren und einzurichten. Ist das Programm heruntergeladen, kann der SQL-Server mit dem „Starten“ Button gestartet werden.



XAMPP liefert auch phpMyAdmin zur Verwaltung von SQL Datenbanken mit. Dazu muss nur der noch Apache Webserver gestartet werden. Und unter MySQL der Button „Admin“ geklickt werden und die Webseite wird geöffnet. Jedoch bevorzuge ich MySql-Workbench, da es meiner Meinung nach intuitiver zu verwenden und übersichtlicher aufgebaut ist als phpMyAdmin. Der größte Vorteil von MySql-Workbench ist für mich jedoch, dass das Erstellen eines ER-Modells hier sehr einfach zu realisieren ist. Daher installiere ich nun MySql-Workbench und öffne das Programm.

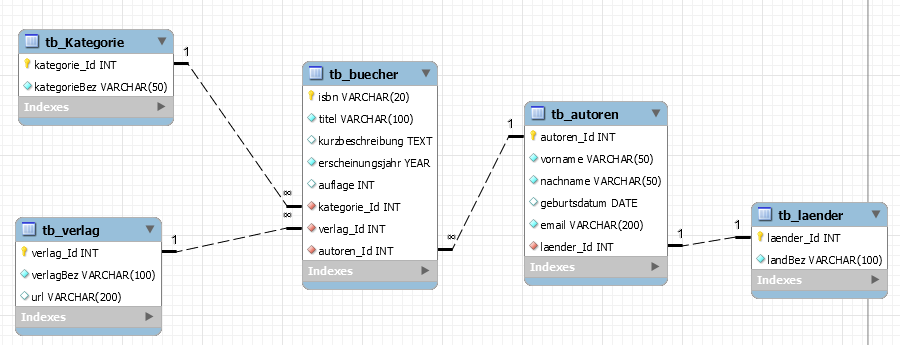
# 2 ER-Modell Bibliothek

Um ein ER-Modell zu erstellen öffne ich MySql-Workbench und wähle im Startmenü auf der linken Seite das Symbol mit den drei verbundenen Tabellen und klicke danach auf das Pluszeichen. Hier klicke ich „Add-Diagramm“.

Ich habe meine Datenbank in 5 Tabellen aufgeteilt, somit wird die 3. Normalform erreicht und es werden Redundazen und Anomalien verhindert.

Es werden nur 1:n Beziehungen benötigt, obwohl gerade zwischen autoren und bücher eine n:m Beziehung möglich wäre, jedoch wurde das ausdrücklich nicht gefordert.

Das fertige Modell sieht wie folgt aus:



Die Attribute url, auflage und kurzbeschreibung dürfen NULL sein, das heißt hier muss nicht unbedingt ein Wert enthalten sein. Die roten Rautensymbole weisen auf meine erstellten Fremdschlüssel hin. Für die kurzbeschreibung habe ich den Datentyp TEXT verwendet, da es sein kann, dass VARCHAR() hier nicht ausreicht. Außerdem kann dieses Feld den Wert NULL annehmen, es ist kein muss hier einen Wert einzugeben.

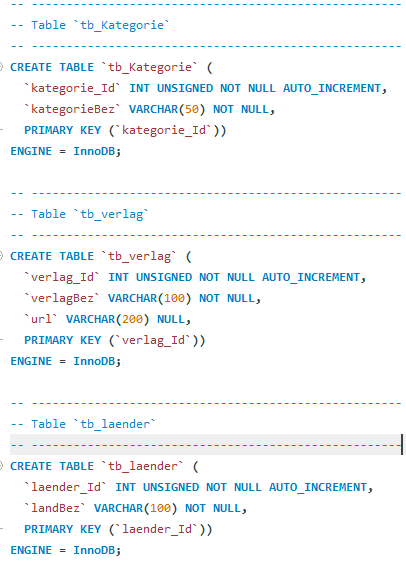
# 3 Datenbankerstellung

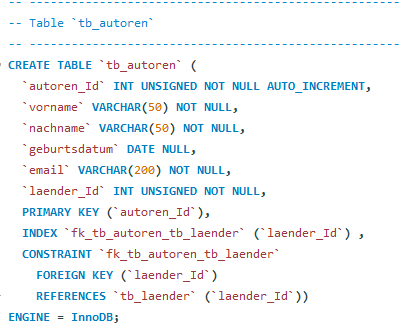
Zu aller erst erstelle ich die Datenbank bibliothek. Das erfolgt mit folgendem Befehl:

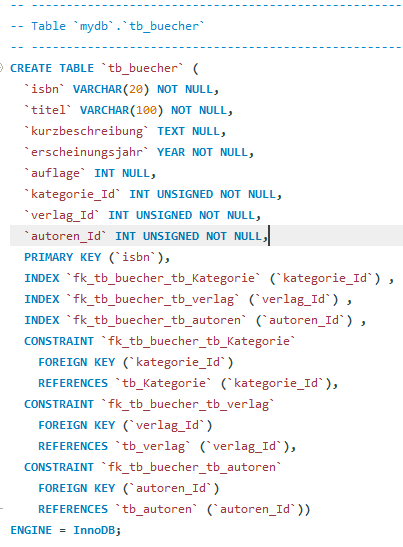


Ich wähle die neu erstellte Datenbank aus, indem ich einem Rechtklick darauf mache und „Set as default Schema “ klicke.

Nun können die einzelnen Tabellen erstellt werden. Dazu sind folgende SQL Befehle nötig:







CREATE TABLE [Tabellenname] ()🡪 erstellt Tabelle mit Namen und Attributen

UNSIGNED 🡪 Beispiel kategorie\_Id darf nicht negativ sein

PRIMARY KEY 🡪 setzt den Primärschlüssel

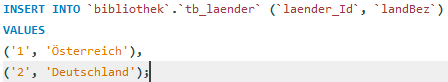
CONSTRAINT 🡪 Name für erstellten Fremdschlüssel

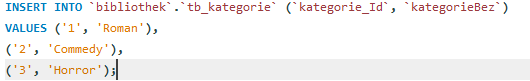
FOREIGN KEY 🡪 Fremdschlüssel angeben

REFERENCES 🡪 Primärschlüssel der verknüpften Tabelle angeben

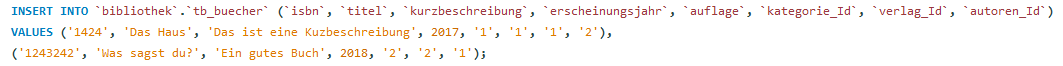
Als nächsten Schnitt befülle ich die Tabellen mit Werten, die Befehle dazu sehen wie folgt aus:











# 4 Datenbank Tests/Dokumentation

**Testfall 1: Foreign Keys**

Als erstes werde ich überprüfen, ob meine zuvor erstellten Foreign Key auch richtig funktionieren. Dazu versuche ich in tb\_buecher ein neues Buch mit einer Kategorie, welche in der tb\_kategorien nicht enthalten ist, einzutragen.

Mit folgendem Befehl füge ich einen neuen Datensatz ein:



Es erscheint ein Fehler:



Der Test war erfolgreich. Es ist nicht möglich eine Kategorie zu verwenden, welche nicht in der tb\_kategorien bereits vorhanden ist. Der Fremdschlüssel wurde richtig gesetzt.

**Testfall 2: Auto Increment**

Als nächstes werde ich in der Tabelle tb\_leander das von mir zuvor gesetzte Auto Increment aus den Primärschlüssel laender\_Id testen. Füge ich einen Datensatz ohne laender\_Id ein, bzw einen leeren String, so muss trotzdem die nächste ID eingefügt werden.

Das passiert mit folgendem Befehl. Ich gebe keine laender\_Id ein.



Das Ergebnis sieht wie folgt aus:



Der Test war erfolgreich, die ID wurde trotzdem automatisch erhöht.

**Testfall 3: Unsigned**

Diesen Test führe ich ebenfalls in der tb\_laender durch. Ich versuche eine negative laender\_ID einzugeben. Das darf nicht funktionieren, da ich dieses Feld auf unsigned gesetzt habe. Es dürfen nur positive Werte zugelassen werden.

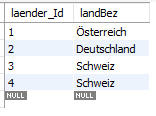
Der Befehl dazu:



Es erscheint eine Warnmeldung, dass der Wert außerhalb des Wertebereichs liegt.



Der Wert „-1“ wurde ersetzt durch die nächste folgende ID. Das ist 4.



Testfall OK

**Testfall 4:** NULL

In der Tabelle tb\_buecher habe ich das Feld titel auf NOT NULL gesetzt. Das heißt dieses Feld darf nicht den Wert NULL annehmen. Das werde ich nun testen, indem ich dem titel den Wert NULL zuweise:

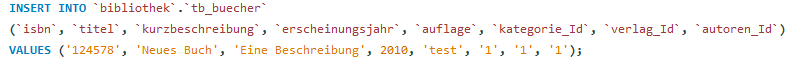


Es erscheint eine Fehlermeldung, dass die Spalte titel nicht NULL sei darf. Test OK



**Testfall 5 : Integer Feld**

Nun versuche ich in ein Integer Feld einen String einzugeben. Das teste ich wieder in der tb\_buecher mit der Spalte auflage.



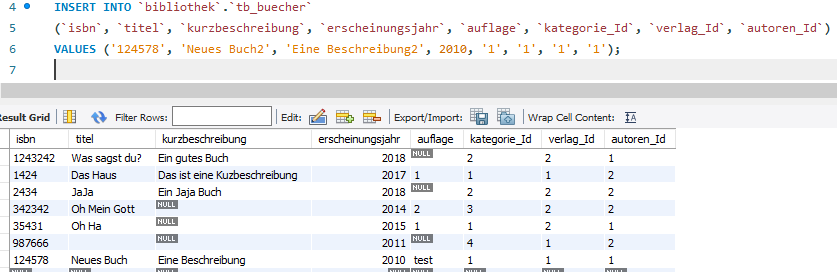
In das Feld „auflage“ habe ich „test“ eingegeben. Es erscheint folgende Meldung:



Der String wurde durch den Wert 0 ersetzt. Test OK

**Testfall 6 : Primary Key**

Getestet wird der Primary Key ISBN in der Tabelle tb\_buecher. Es darf nicht möglich sein eine ISBN, welche bereits vorhanden ist, nochmals einzutragen.



ES erscheint die Meldung:



Der neue Record wird nicht hinzugefügt. Test OK

Testfall 7: Mit Fremdschlüssel verknüpften Eintrag löschen. Ich versuche eine Kategorie, welche in der tb\_buecher verwendet wird, in der tb\_kategorien herauszulöschen. Das darf nicht möglich sein. Da ich einen solchen Eintrag erst löschen kann, wenn er in keiner anderen Tabelle mehr verwendet wird.

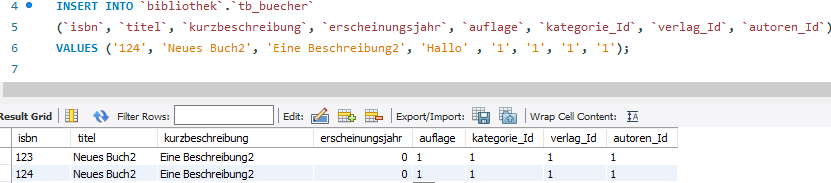


Es ist nicht möglich, Test ist OK.



**Testfall 7: Feld YEAR**

In der Tabelle tb\_buecher ist das Feld erscheinungsdatum als Datentyp Year hinterlegt. Es sollte nicht möglich sein hier einen String einzugeben.



Es wird eine 0 eingetragen. Test OK

**Testfall 8: Feld Date**

Das Feld geburtsdatum in der Tabelle tb\_autoren besitzt den Datentyp Date. Füge ich hier einen String en, so wird der Standardwert gesetzt 🡪 0000-00-00 Test OK

