Beschreibung ERM SQL

# Voraussetzungen

Um das SQL ausführen zu können ist ein MySQL oder MariaDB Datenbankmanagementsystem (DBMS) von Nöten. Außerdem darf noch keine Datenbank (DB) bzw. Schema mit dem Namen *HOCHZEITSPLANER* existieren, wobei Groß-und Kleinschreibung egal ist.

# Kurzeinführung SQL

* CREATE DATABASE: Erstellt eine neue Datenbank/Schema
* USE: Benutze die angegebene Schema
* CREATE TABLE: Erstelle eine Tabelle mit den angegebenen Attributen
* INT, TIME, VARCHAR …: Datentypen
* NOT NULL: Der Attributwert muss explizit gesetzt werden, darf also nie NULL sein
* UNIQUE: Der Attributwert darf innerhalb der Spalte nur einmal vorkommen
* UNSIGEND: Bei Datentypen mit Zahlenwerten können keine negativen Zahlen eingetragen werden. Dafür erhöht sich der positive Zahlenraum um das doppelte.
* AUTO\_INCREMENT: Der Wert dieses Attributes startet stadtartmäßig bei 1 und zählt dann nach oben. Man muss sich dadurch nicht selbst darum kümmern, sondern kann dies dem DBMS überlassen
* DEFAULT: Standartwerts des Attributs
* PRIMARY KEY: Attributmenge, die den Primärschlüssel der Tabelle bilden
* FOREIGN KEY: Fremdschlüsselbeziehung; Man gibt das Attribut an auf die der Fremdschlüssel zutrifft
  + REFERENCES: Der Fremdschlüssel mit Tabelle und Attribut
  + ON UPDATE: Aktion, die ausgeführt werden soll, wenn der Fremdschlüssel sich ändert
  + ON DELETE: Aktion, die ausgeführt werden soll, wenn der Fremdschlüssel gelöscht wird
  + CASCADE: Ändere/Lösche die Fremdschlüsselbeziehung (beim Löschen wird somit der ganze Datensatz gelöscht, also die korrespondierenden Zeilen)
  + RESTRICT: Das Ändern/Löschen wird sowohl in der Tabelle mit dem Fremdschlüssel also auch beim Fremdschlüssel verhindert.

# Erstellen des Schemas

Als erstes muss ein Schema, oder wie auch manchmal genannt eine Datenbank erstellt werden. In dieser werden dann die Tabellen erstellt.

# Setzen des aktuellen Schemas

Mit dem USE Befehl kann man erreichen dass alle nachfolgenden Befehle auf diesem Schema ausgeführt werden. Dadurch muss bei der Tabellenerstellung nicht immer das Schema mit angegeben werden.

# Erzeugen der Tabellen mit Fremdschlüsselbeziehungen

Da hier die Tabellen direkt mit den Fremdschlüsselbeziehungen erstellt werden, müssen die Tabellen in einer logischen Reihenfolge erstellt werden, sodass jede Tabelle die ein Fremdschlüssel sein wird schon erzeugt worden ist.

## Bekannte Entitäten

Zu den bekannten und schon beschrieben Entitäten gehören:

* Hochzeitsveranstaltung
* Caterer
* Aktion
* Nahrungsmittel
* Medium
* Beleg
* Ort

Die Tabellennamen sind im Plural des Entitätsnamen gehalten.

Diese Entitäten haben alle ein neues Attribut – eine ID. Diese ID wird automatisch mit dem Einfügen einer neuen Entität in die Tabelle inkrementiert. Diese ID bildet auch den Primärschlüssel.

Alle Attribute die optional sind erlauben NULL-Werte, alle andere verlangen explizit Werte.

Sofern einfache Referenzen, d.h. Kardinalitäten mit 0/1 je Seite vorhanden waren wird direkt der Fremdschlüssel erzeugt. Das Updaten der Fremdschlüsselbeziehung ist per se überall erlaubt. Beim Löschen nur, wenn das dazugehörige Attribute NULL-Werte erlaubt, ansonsten wird es verhindert.

Alle anderen Attribute, die in Listen auftauchen wurden in weitere Tabellen ausgelagert.

## Weitere Entitäten

Bei der Analyse des Lastenheftes hat sich ergeben, dass gewisse Attribute eine eigene Tabelle benötigen. Da dies unter anderem aus datenbanktechnische Gründen passiert als auch, dass man hier gewisse Attribute aus logischer Sicht auslagern muss, damit eine Speicherung möglich ist, gibt es damit Entitäten, die zwar in der Datenbank existieren, aber nicht im AKD.

Zu zweiteren gehören die Aktionsarten, die Aktionszustände und die Hilfsmittelarten. An sich lassen sich diese als einfach String abbilden, aber da man nicht benutze Zustände und Arten nicht „vergessen“ möchte, braucht man hierzu eigene Tabellen.

Außerdem brauchen die Telefonnummern und die Emailadresse eigene Tabellen, da ansonsten ein unsauberer Primärschlüssel in gewissen Tabellen, die Relationen enthalten entstehen würde.

## Relationstabellen

Alle Attribute, die Listen verlangen, brauchen im relationalen Datenbankmodell eine extra Tabelle. Diese Tabellen haben bis auf Ausnahmen nur zwei Attribute, nämlich die ID der eigentlichen Entität und die des „Listeneintrags“. Man kann natürlich auch Relationsattribute, wie z.B. die Menge bei Aktionshilfsmittel mitspeichern. Die Fremdschlüssel verweisen immer auf die Entitätstabelle. Sofern eine Entität gelöscht wird, löscht sich dieser Eintrag gleich mit, da er logischerweise nicht mehr von Nöten ist.

## Weitere Besonderheiten

Das Attribut Priorität in der Tabelle Aktionen wurde auf eine spezielle Art umgesetzt. Da es nur eine feste und nicht erweiterbare Anzahl an Prioritäten gibt, muss dafür keine extra Tabelle erzeugt werden. Um nun nicht den Prioritätstext speichern zu müssen kann auch einfach eine Zahl, die diese Priorität repräsentiert gespeichert werden. Die Applikation muss später nur noch zwischen den Zahlen und dem Text konvertieren können.

## Prorammtabellen

Diese Tabellen speichern keine Entitäten an sich, sondern für die Applikation wichtige Daten.

Die Tabelle „NichtInformierteNutzer“, ist dafür da abzuspeichern, welche Nutzer bei welchen Aktionen keine Informationen bei Aktionen erhalten wollen. Dafür speichert diese Tabelle zu einem die Person wie auch die Aktion, wobei die ID’s der beiden den Primärschlüssel bilden und entsprechende Fremdschlüssel ebenfalls vorhanden sind.