

# Dokumentation

## Verwendete Software:

- DB-System: MySQL/MariaDB; MySQL ist das häufigst verwendete Datenbanksystem in der Online-Entwicklung, außerdem ist MySQL (MariaDB) kostenfrei
- Dokumentations-Software: MS Word für die Dokumentation selbst, da es einen Quasi-Standard für die Textverarbeitung darstellt; PHPMyAdmin (Website am Webserver) für die Datenbankmodellierung, da in der Praxis eigene MySQL-Clients keinen Zugriff auf gehostete Datenbanken bekommen (derartige Zugriffe werden von den gängigen Providern aus Sicherheitsgründen blockiert)

## Datenbank

*Anmerkung: Dieser Teil der Dokumentation könnte auch direkt aus dem DB-Schema (das aus PHPMyAdmin exportiert werden kann) entnommen bzw. im Anhang dargestellt werden.*

### Verwendete Tabellen (Übersicht):

- `tbl_alben`: Speichert sämtliche relevanten Information zu einem Album (wie Albumtitel, Erscheinungsjahr, etc. – siehe Details weiter unten)
- `tbl_interpreten`: Speichert Informationen zum Interpreten eines Albums
- `tbl_titel`: Speichert die zu einem Album zugehörigen Titel in der jeweiligen Reihenfolge am Album ab

## Tabellen

- `tbl_alben`:
  - `IDAlbum`: `INT(10) unsigned, NOT NULL, auto-increment`; Primärschlüssel
  - `FIDInterpret`: `INT(10) unsigned, NOT NULL`; Fremdschlüssel auf `tbl_interpreten.IDInterpret`
  - `Albumtitel`: `VARCHAR(64), NOT NULL`
  - `Erscheinungsjahr`: `YEAR, NOT NULL`
- `tbl_interpreten`:

- `IDInterpret: INT(10) unsigned, NOT NULL, auto-increment;`  
Primärschlüssel
- `Name: VARCHAR(64)`
- `tbl_titel:`
  - `IDTitel: INT(10) unsigned, NOT NULL, auto-increment;` Primärschlüssel
  - `FIDAlbum: INT(10) unsigned, NOT NULL;` Fremdschlüssel auf `tbl_alben.IDAlbum`
  - `Songtitel: VARCHAR(64), NOT NULL`
  - `Reihenfolge: TINYINT(2) unsigned, NOT NULL`

## Constraints

Generell gilt: Für sämtliche verwendeten Constraints in dieser Datenbank wird festgelegt, dass bei einem `UPDATE` ein `CASCADE` zum Einsatz gebracht wird.

- `tbl_alben.FIDInterpret → tbl_interpreten.IDInterpret`: n:1-Beziehung, da ein Interpret mehrere Alben veröffentlichen kann:
  - `ON DELETE RESTRICT`: sollte der Versuch gestartet werden, einen Interpreten zu löschen, von dem bereits mind. ein Album gespeichert wurde, so soll dieser Löschversuch verhindert werden. Begründung: würde der Interpret (wie bei einem `CASCADE`) gelöscht werden, so gäbe es aus Sicht des Albums keinen Bezug zum Interpreten mehr, würde man das Löschen nicht verhindern. Auf der anderen Seite könnte bei einem Löschen auch `SET NULL` verwendet werden, was jedoch auch keinen Sinn macht, da das Album dann keine Information zum Interpreten mehr besitzt. Da angenommen wird, dass es sich um eine Datenbank zur Verwaltung der eigenen Alben handelt, macht das Löschen eines Interpreten auch keinen Sinn, da die vom Interpreten veröffentlichten Alben ja dennoch existieren.
  - `ON UPDATE CASCADE`: siehe einführende Begründung
- `tbl_titel.FIDAlbum → tbl_alben.IDAlbum`: n:1-Beziehung, da ein Album über mehrere Titel verfügen können soll:
  - `ON DELETE CASCADE`: sollte ein Album gelöscht werden (aus welchem auch immer; das kann an dieser Stelle nicht verhindert werden), so sollen alle zugehörigen Songtitel ebenfalls gelöscht werden, da bei fehlender Albuminformation ein Songtitel keinem Album mehr zugeordnet werden kann.

## ER-Diagramm

(Screenshot, Foto, Illustration aus MySQL Workbench o.ä., ...)