



# Informationsblatt Informatik – Erstellung einer Datenbank mit SQLite

## Datentypen und Wertebereichsintegrität

Vor der Implementierung der Tabellen des relationalen Modells in ein Datenbanksystem wird jedem Attribut ein geeigneter Datentyp zugeordnet. Zusätzliche Angaben zum Attribut ermöglichen eine Einschränkungen des Wertebereichs und sorgen bei der Dateneingabe für Datenintegrität.

Als Datentypen stehen uns in SQLite folgende Typen zur Verfügung:

Datentyp	Beschreibung
INTEGER	ganze Zahl
REAL	Fließkommazahl
TEXT	Zeichenkette

SQLite unterstützt keine Zeit-, Datums- oder boolesche Datentypen, sondern wandelt diese in einen Ersatztyp um. Dabei sind folgende Schreibweisen für Attributwerte einzuhalten.

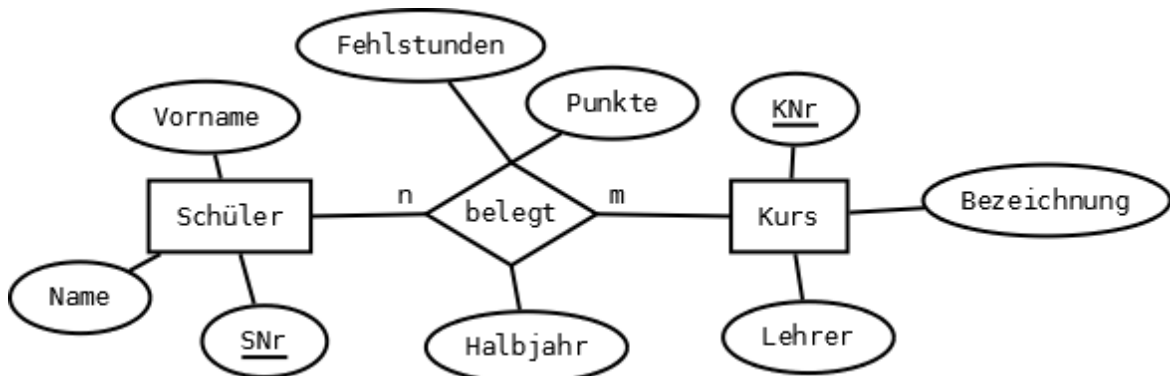
Typ	Schreibweise	verwendeter Ersatztyp
DATE	YYYY-MM-DD	TEXT
DATETIME	YYYY-MM-DD HH:MM:SS	TEXT
TIME	HH:MM:SS	TEXT
BOOLEAN	0 ⇔ FALSE, 1 ⇔ TRUE	INTEGER

Die Attribute können mit folgenden Einschränkungen zur Datenintegrität definiert werden:

Option	Beschreibung
NOT NULL	NULL-Werte sind nicht zulässig
DEFAULT	Vorgabe eines Standardwerts
PRIMARY KEY	Attribut ist Primärschlüssel (nur für einfache Schlüssel)
AUTOINCREMENT	nur in Verbindung mit PRIMARY KEY, automatische Erhöhung des Primärschlüsselattributwertes beim Hinzufügen eines Datensatzes
CHECK	Prüfkriterium für die Aufnahme des Datensatzes

## Beispiel

Gegeben sind das folgende ER-Modell und das daraus abgeleitete Relationenschema.



**Schüler**(SNr, Vorname, Name)

**Kurs**(KNr, Bezeichnung, Lehrer)

**Belegung**(Kurs.KNr, Schüler.SNr, Halbjahr, Fehlstunden, Punkte)



# Informationsblatt Informatik – Erstellung einer Datenbank mit SQLite

Die Implementation mit SQLiteStudio kann dialoggesteuert oder direkt über den SQL-Befehl durchgeführt werden. Zunächst sind die Mastertabellen (Tabellen ohne Fremdschlüssel) und im Anschluss die Detailtabellen (Tabellen mit Fremdschlüssel) zu erzeugen.

## Überlegungen und Implementation für die Tabelle Schüler

Der Schlüssel ist fortlaufend zu nummerieren, der Schülername ist zwingend erforderlich.

Attribut	Datentyp	Wertebereich	Not Null	Default
<u>SNr</u>	Integer	AUTOINCREMENT	ja	
Vorname	Text		nein	
Name	Text		ja	

```
CREATE TABLE SCHUELER (  
  SNr INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,  
  Vorname TEXT,  
  Name TEXT NOT NULL  
)
```

## Überlegungen und Implementation für die Tabelle Kurs

Der Schlüssel ist fortlaufend zu nummerieren, die Bezeichnung ist zwingend erforderlich.

Attribut	Datentyp	Wertebereich	Not Null	Default
<u>KNr</u>	Integer	AUTOINCREMENT	ja	
Bezeichnung	Text		ja	
Lehrer	Text		nein	

```
CREATE TABLE KURS (  
  KNr INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,  
  Bezeichnung TEXT NOT NULL,  
  Lehrer TEXT  
)
```

## Überlegungen und Implementation für die Tabelle Kursbelegung

Es sind die Punktwerte von 0 bis 15 möglich, Fehlstunden sind auf 0 zu initialisieren und stets positiv. Das Halbjahr kann nur die Werte 1 (default) oder 2 annehmen.

Attribut	Datentyp	Wertebereich	Not NULL	Default
<u>SNr</u>	Integer		ja	
<u>KNr</u>	Integer		ja	
<u>Halbjahr</u>	Integer	Halbjahr = 1 OR Halbjahr = 2	ja	1
Fehlstunden	Integer	Fehlstunden >= 0	nein	0
Punkte	Integer	Punkte BETWEEN 0 AND 15	nein	

Da der Schlüssel aus drei Attributen besteht, muss seine Festlegung getrennt von der Attributdefinition als zusätzliche (Zwangs-) Angabe erfolgen.

```
CREATE TABLE KURSBELEGUNG (  
  SNr INTEGER NOT NULL REFERENCES Schueler (SNr),  
  KNr INTEGER NOT NULL REFERENCES Kurs (KNr),  
  Halbjahr INTEGER NOT NULL DEFAULT 1 CHECK (Halbjahr = 1 OR Halbjahr = 2),  
  Fehlstunden INTEGER DEFAULT 0 CHECK (Fehlstunden >= 0),  
  Punkte INTEGER CHECK (Punkte BETWEEN 0 AND 15),  
  PRIMARY KEY (SNr, KNr, Halbjahr)  
)
```