

# Daten- banken in SQLiteStudio



Fortbildung im Rahmen des InformatiklehrerInnen-  
Tages MV

April 2016 – Anne Buhrand – [a.buhrand@riseup.net](mailto:a.buhrand@riseup.net)

# Darum SQLite

## Organisatorisch

- Freies, plattformübergreifend implementiertes Datenbanksystem
- Einfache Installation und Administration

## Informatisch

- Engine sehr, sehr klein (~1MB)
- Komplette DB in einer Datei, keine Serverinstallation nötig
- Max. Datenbankgröße 2TB (Access: 2GB)
- Unterstützung von Volltextsuche (fehlt in Access)

# Darum WS zu SQLiteStudio

## Organisatorisch

- Freies, plattformübergreifend implementiertes DBMS
- Einfache Installation

## Informatisch

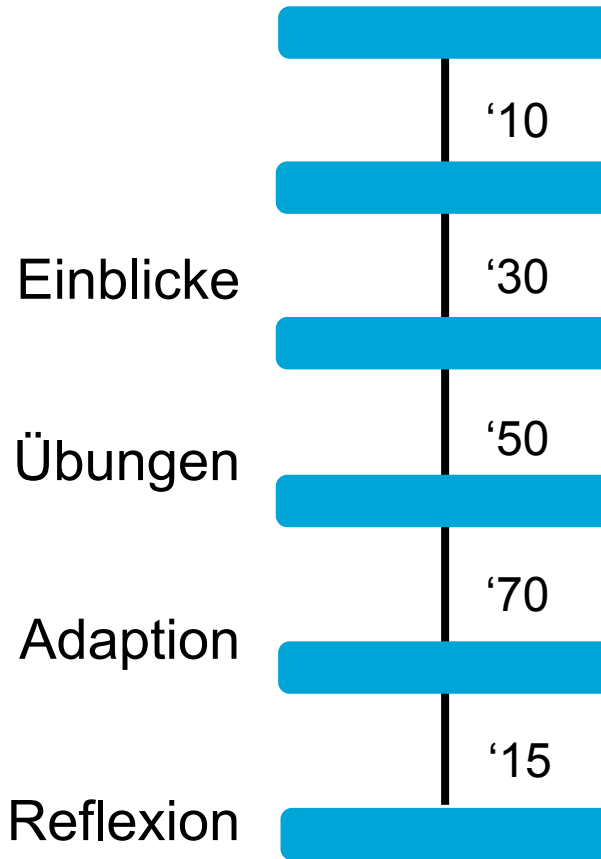
- Kontakt mit der Konsole nicht notwendig (Vor- oder Nachteil?)
- Umfangreiche GUI für SQLite
- Max. Datenbankgröße 2TB (Access: 2GB)
- Unterstützung von Volltextsuche (fehlt)

---

Vorabhinweise 2016/17

Themenfeld Datenbanken – SQLite mit Handreichung

# Fahrplan



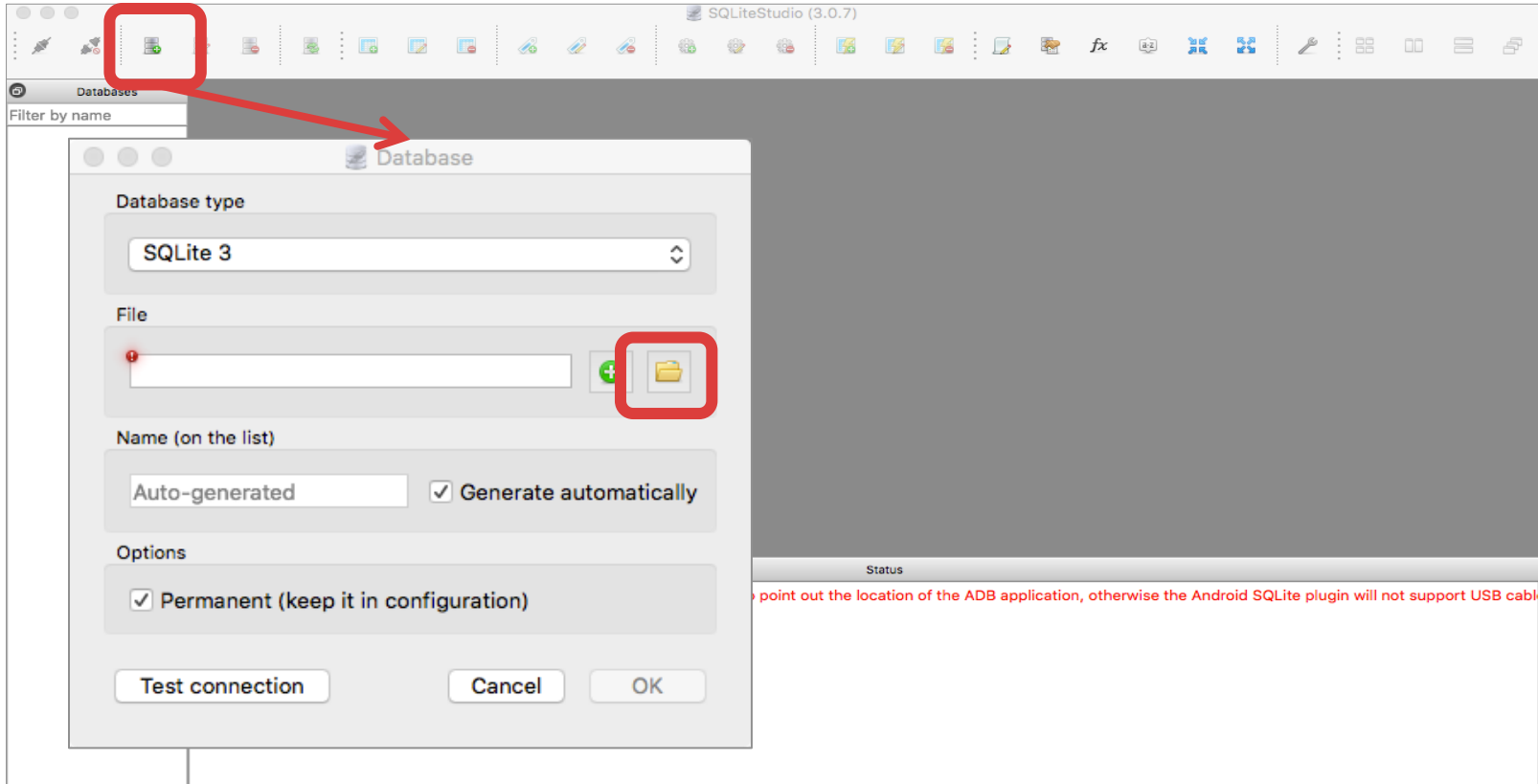
# SQLite vs „Standard“-SQL

- Datentypen NULL | INTEGER | REAL | TEXT | BLOB
- Stark an SQL orientiert

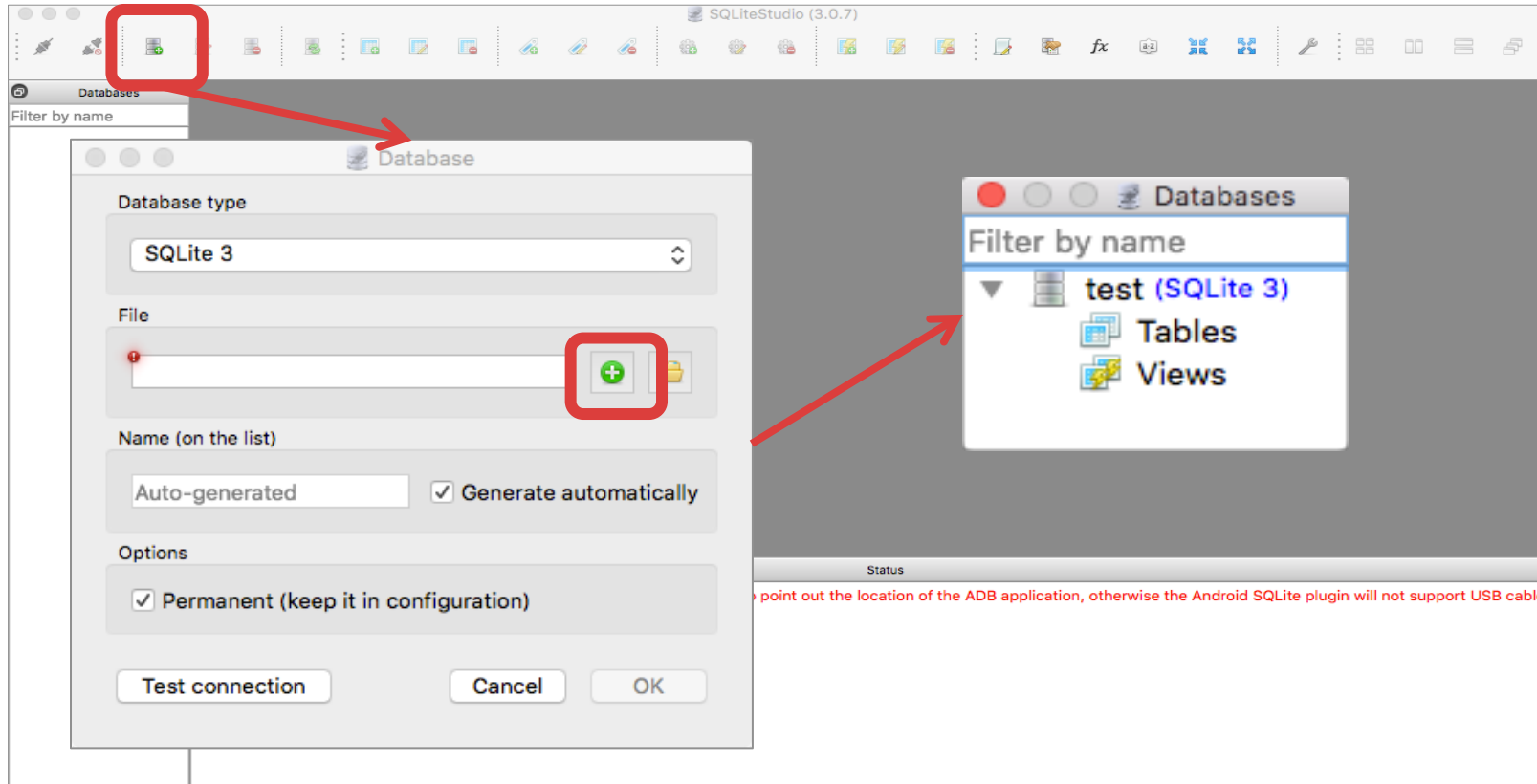
Ausnahmen:

- Export mittels **Into Outfile** nicht vorgesehen
- **Right** und **Outer Join** nicht implementiert, nur LEFT JOIN
- **Alter Table** nicht komplett umgesetzt – Außer:
  - RENAME TABLE
  - ADD COLUMN
- **Views** sind nur lesend möglich

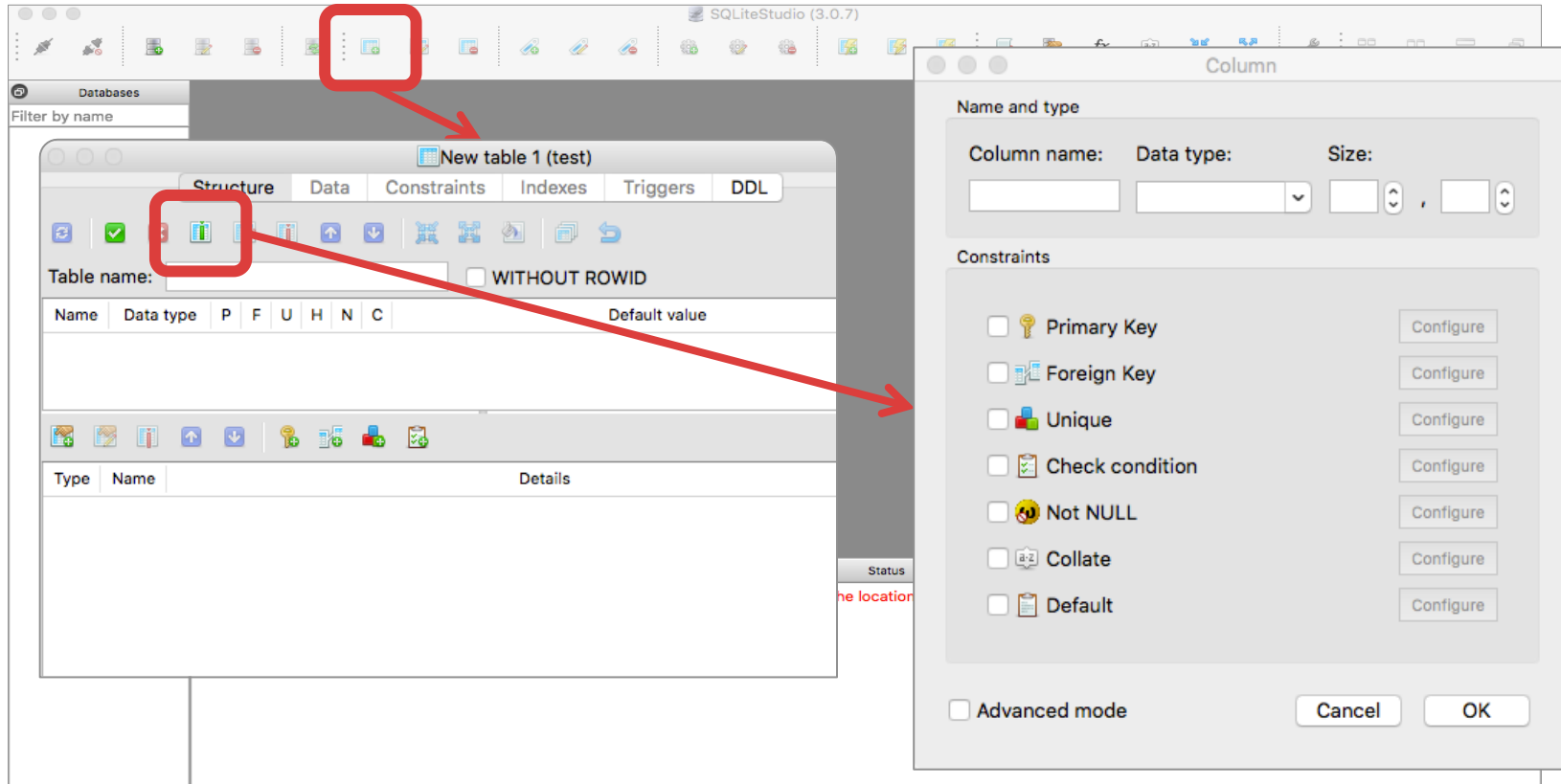
# Öffnen einer bestehenden Datenbank (\* .db)



# Anlegen einer Datenbank



# Anlegen einer Datenbanktabelle





# Anlegen einer Datenbanktabelle

Column

Name and type

Column name: Data type: Size:

Constraints

- ☐ Primary Key Configure
- ☐ Foreign Key Configure
- ☐ Unique Configure
- ☐ Check condition Configure
- ☐ Not NULL Configure
- ☐ Collate Configure
- ☐ Default Configure

☐ Advanced mode Cancel OK

Remember, Remember

Datentypen in SQLite

- NULL
- INTEGER (insb. für ID)
- REAL
- TEXT
- BLOB

Alle anderen Datentypen werden intern in einen dieser DT konvertiert

# Anlegen einer Datenbanktabelle

The screenshot shows the SQLiteStudio 3.0.7 interface. The 'New table 1 (test)' dialog is open, with the 'Structure' tab selected. The table name is 'Person'. The columns are defined as follows:

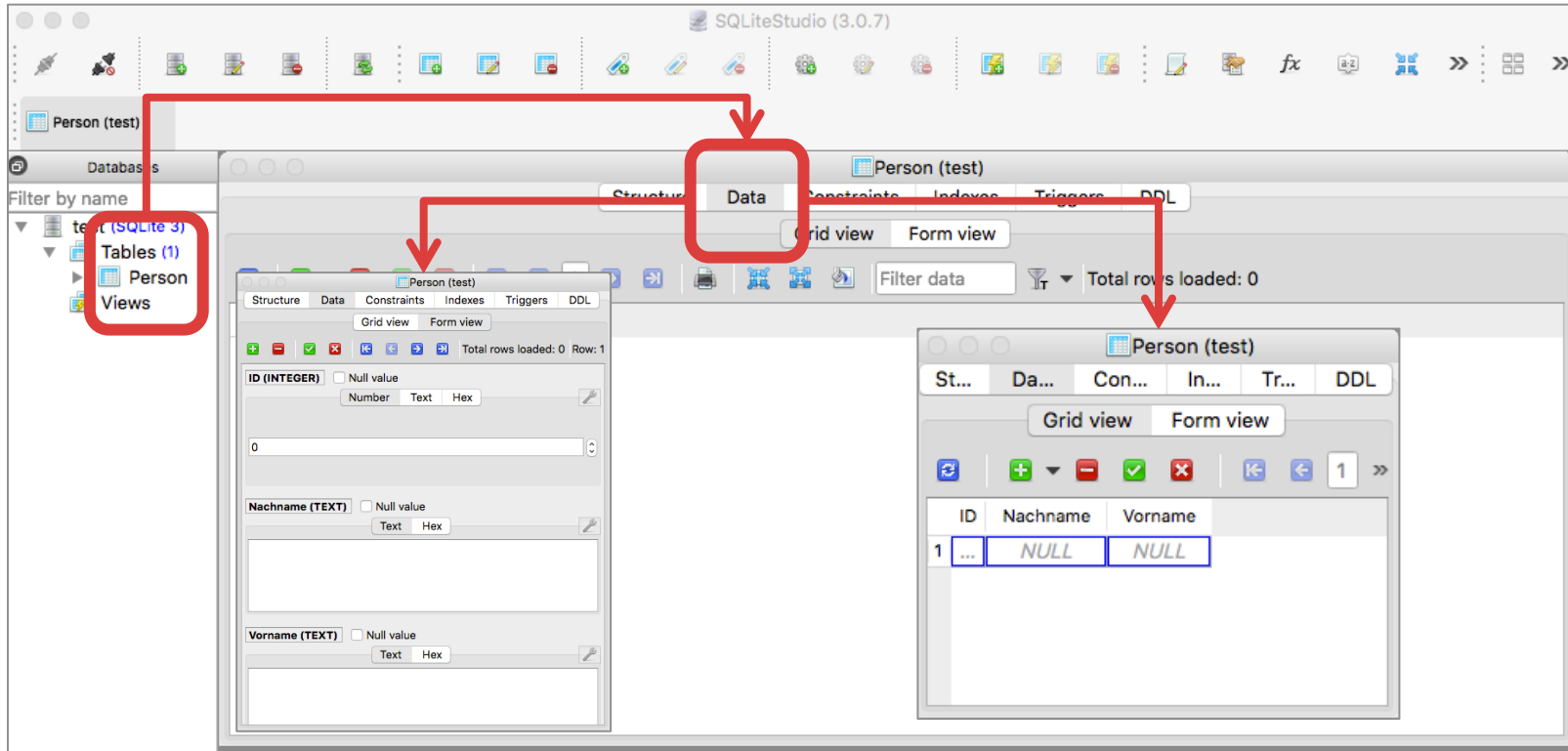
	Name	Data type	P	F	U	H	N	C	Default value
1	ID	INTEGER							NULL
2	Nachname	TEXT							NULL
3	Vorname	TEXT							NULL

The 'Queries to be executed' window shows the following SQL statement:

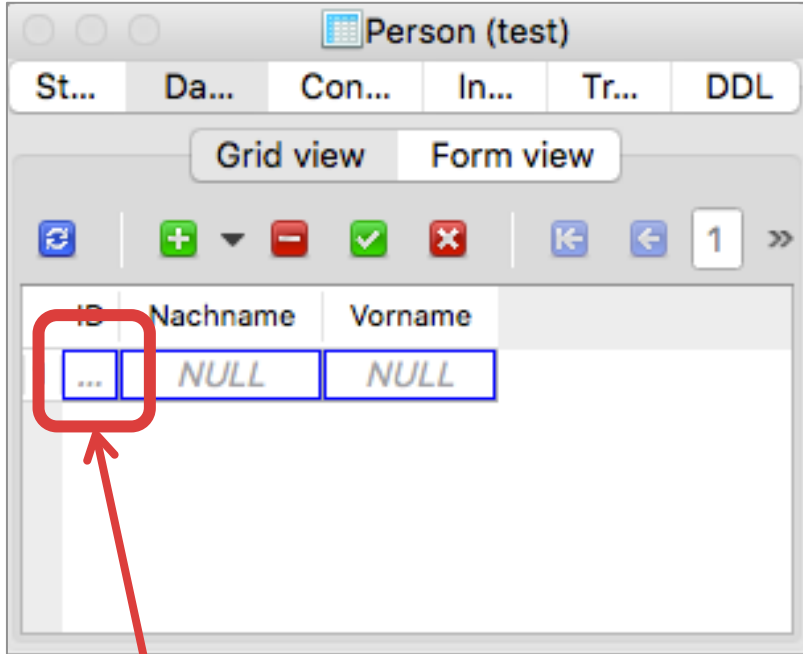
```
CREATE TABLE Person (  
  ID      INTEGER PRIMARY KEY AUTOINCREMENT,  
  Nachname TEXT,  
  Vorname  TEXT  
);
```

A red box highlights the 'Structure' tab in the 'New table 1 (test)' dialog, and a red arrow points from it to the 'CREATE TABLE' SQL statement in the 'Queries to be executed' window.

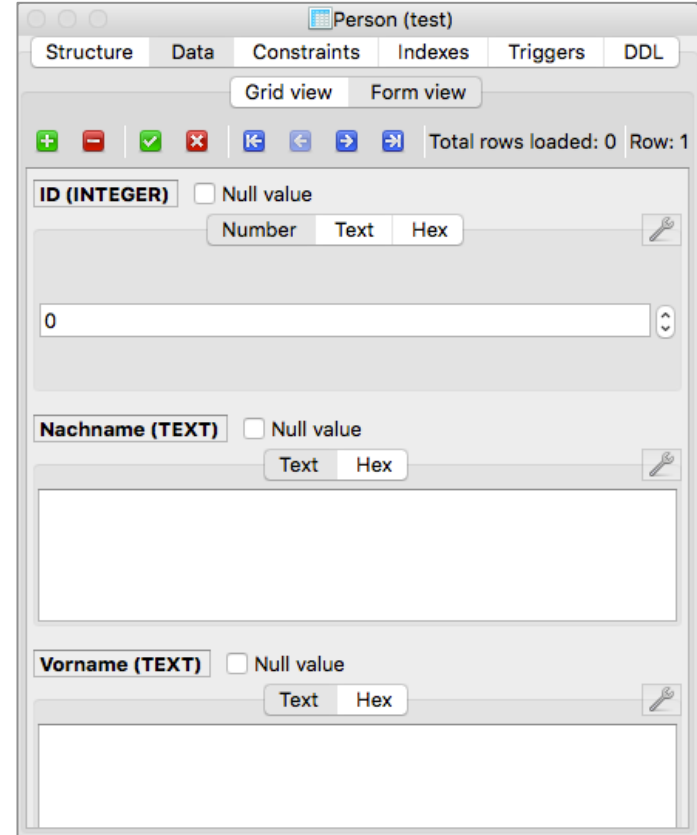
# Befüllen einer Datenbanktabelle



# Befüllen einer Datenbanktabelle



Obacht: Felder mit AutoIncrement unbearbeitet lassen



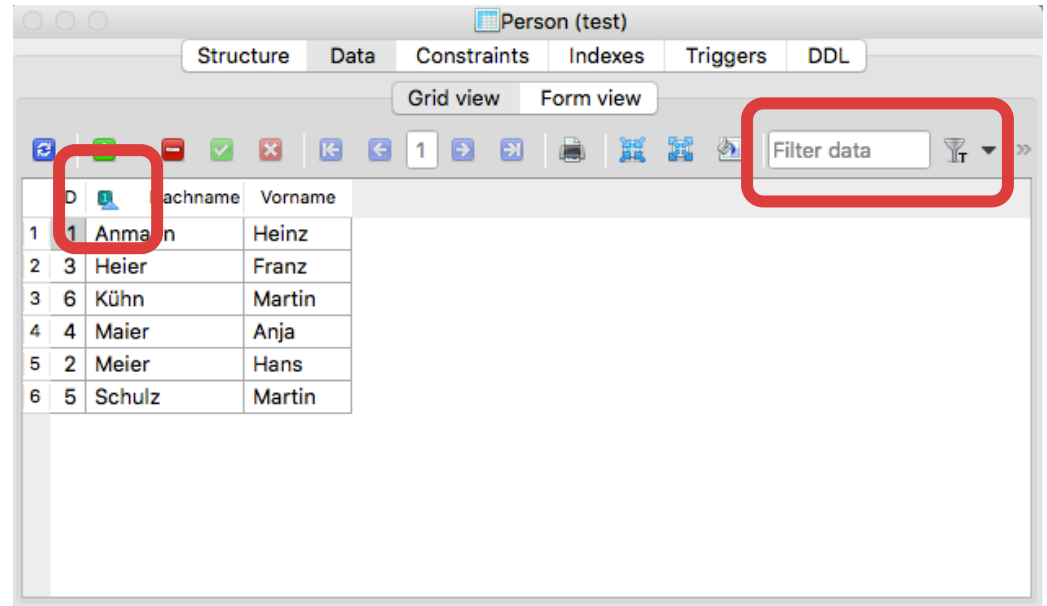
# Daten sortieren und filtern

## Sortieren

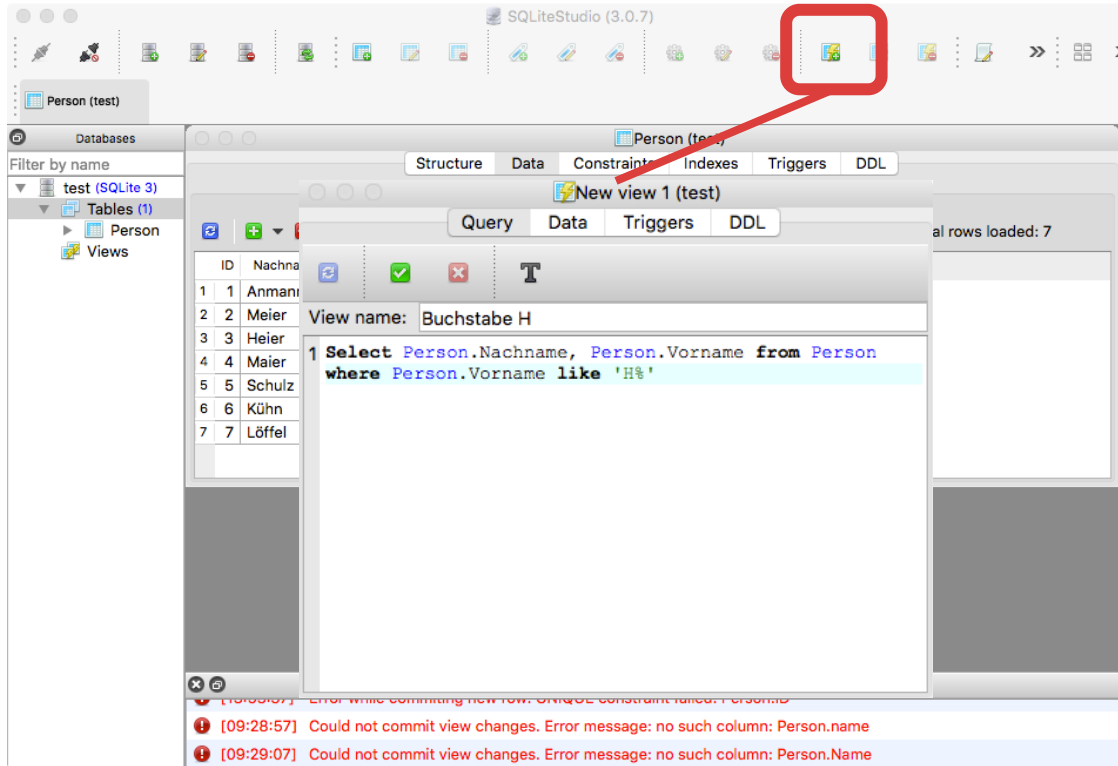
- Aufsteigend/Absteigend

## Filterkriterien

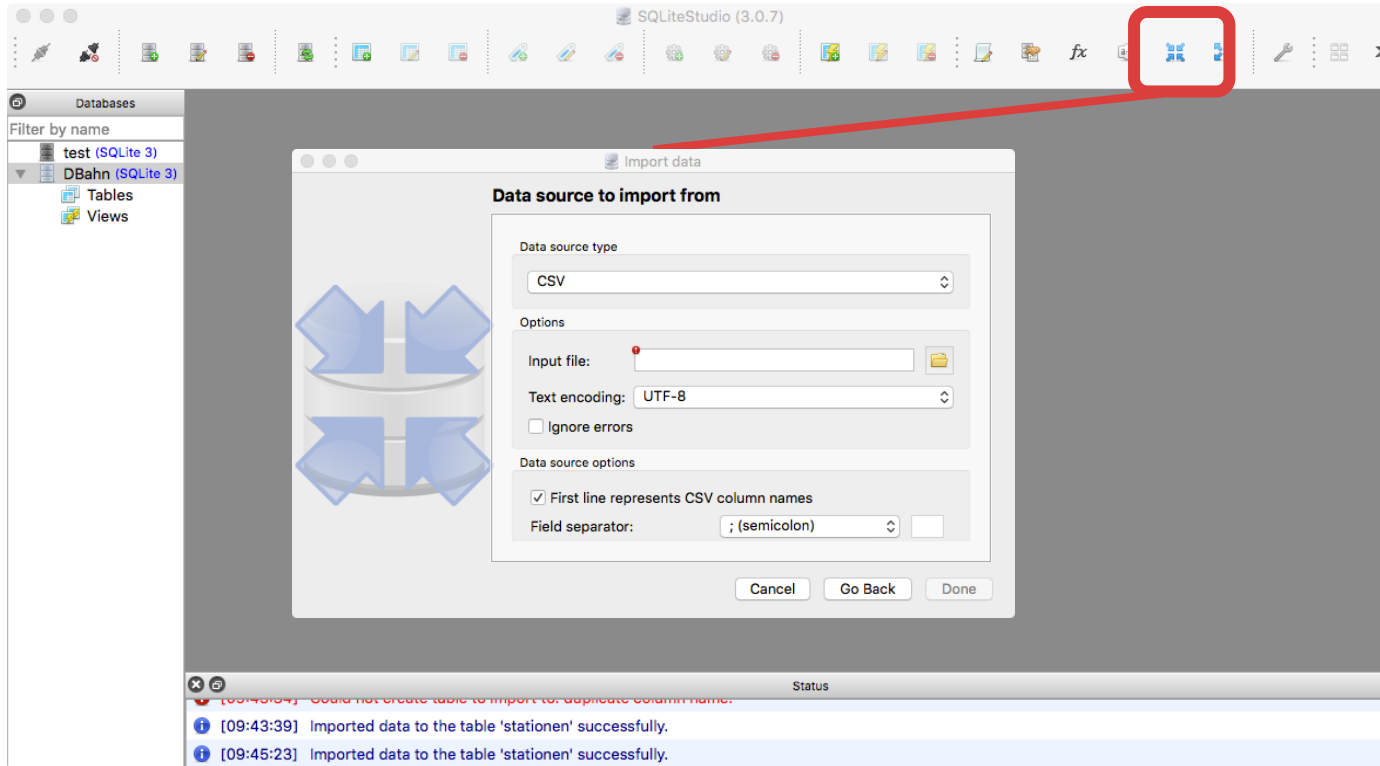
- Text
- Regulärer Ausdruck
- SQL-Ausdruck



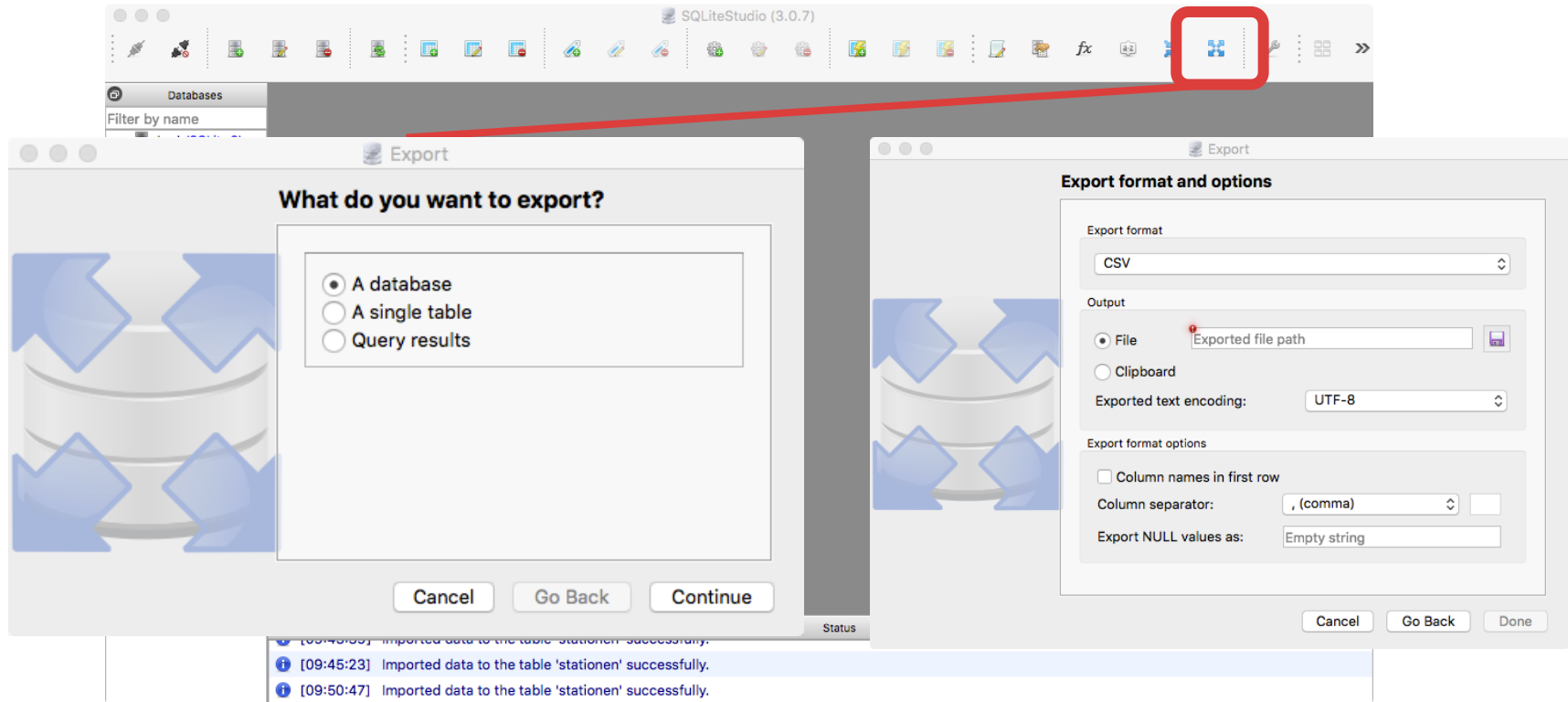
# Views definieren und nutzen



# Import von CSV-Dateien



# Export von Datenbanken, Tabellen und Views



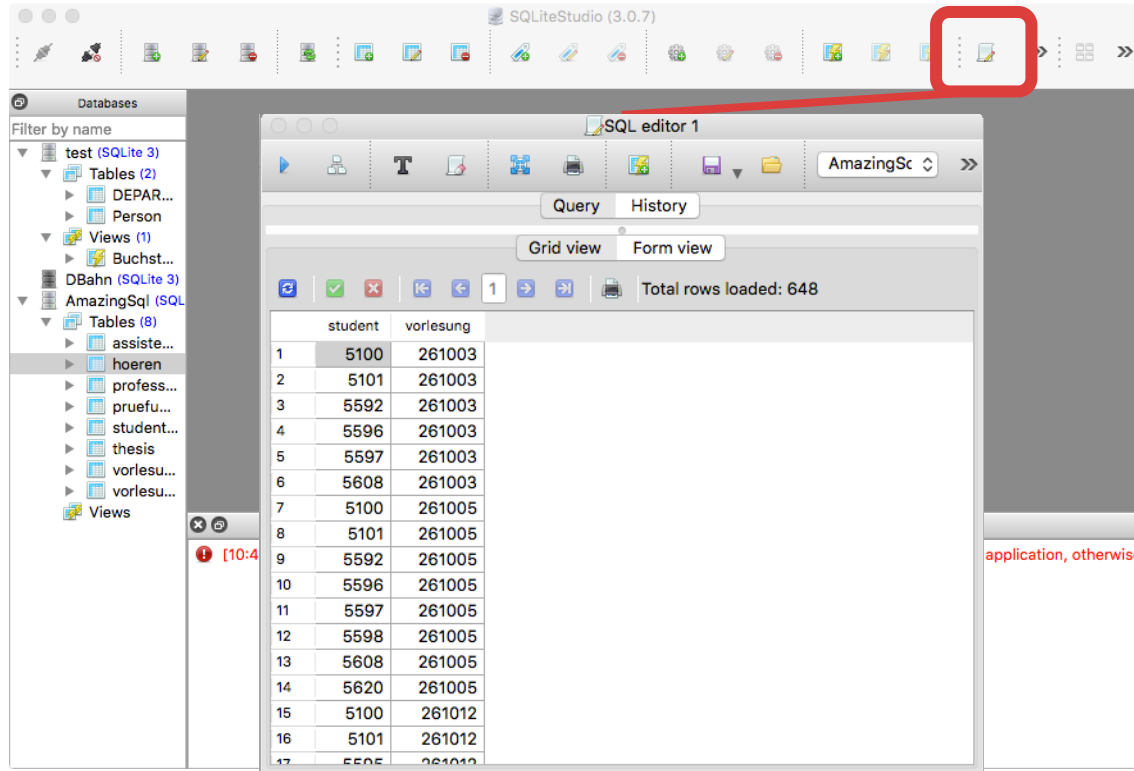


# Export von Datenbanken, Tabellen und Views

## Formate

- CSV
- PDF (*Erzeugt reproduzierbaren Absturz in OSX 10.11.4*)
- HTML
- SQL
- XML
- JSON

# Manuelle SQLite-Abfragen



# Übungsphase

Material:

AmazingSQL.db (Entnommen aus dem Tool Amazing SQL)

