

Relations

Mag. Thomas Griesmayer

Definition

- Eine relationale Datenbank ist eine Sammlung von Tabellen, die miteinander in Beziehung stehen.
- Beispiele:
 - Kunde \Leftrightarrow Produkt
 - Lehrer \Leftrightarrow Student
 - Mitarbeiter \Leftrightarrow Vorgesetzter

DVD verborgen

- Regelmäßig verborgen wir unsere DVDs.
- Ich möchte wissen, ob
 - die DVD bei mir zu Hause ist
 - die DVD verborgen ist und an wen (Vor- und Nachname)
- Weder die Rückgabe noch frühere Verleihe sollen gespeichert werden.

DVD

<u>DVDID</u>	<u>PK INT</u>
ORIGINALTITEL	VARCHAR(30)
UNTERTITEL	VARCHAR(30)
BESCHREIBUNG	VARCHAR(500)
REGIE	VARCHAR(30)
DAUER	INT
KINOSTART	DATE
GENRE	VARCHAR(30)
FSK	INT
LAND	VARCHAR(30)
BEWERTUNG	DECIMAL(4,2)
VORNAME	VARCHAR(30)
NACHNAME	VARCHAR(30)

DVD

<u>DVDID</u>	<u>PK INT</u>
ORIGINALTITEL	VARCHAR(30)
UNTERTITEL	VARCHAR(30)
BESCHREIBUNG	VARCHAR(500)
REGIE	VARCHAR(30)
DAUER	INT
KINOSTART	DATE
GENRE	VARCHAR(30)
FSK	INT
LAND	VARCHAR(30)
BEWERTUNG	DECIMAL(4,2)

PERSON

<u>PERSONID</u>	<u>PK INT</u>
VORNAME	VARCHAR(30)
NACHNAME	VARCHAR(30)

keine BEZIEHUNG
no RELATION

Properties | Data | ER Diagram | master | Databases

Table Name: ID:

Object description (comment): Type:

	Column Name	#	Type	Length	Scale	Precision	Not Null	Identity
Columns								
Unique Keys	PERSONID		int	100			[v]	[v]
Check constraints	VORNAME		varchar	30			[v]	[]
Foreign Keys	NACHNAME		varchar	30			[v]	[]
Indexes								

PERSON - Persist Changes

SQL Preview:

```
CREATE TABLE MedienBibliothek.dbo.PERSON (
  PERSONID int IDENTITY(0,1) NOT NULL,
  VORNAME varchar(30) NOT NULL,
  NACHNAME varchar(30) NOT NULL,
  CONSTRAINT PERSON_PK PRIMARY KEY (PERSONID)
);
```

Open Editor | Copy | Cancel | **Persist**

DVD

<u>DVDID</u>	<u>PK INT</u>
ORIGINALTITEL	VARCHAR(30)
UNTERTITEL	VARCHAR(30)
BESCHREIBUNG	VARCHAR(500)
REGIE	VARCHAR(30)
DAUER	INT
KINOSTART	DATE
GENRE	VARCHAR(30)
FSK	INT
LAND	VARCHAR(30)
BEWERTUNG	DECIMAL(4,2)
PERSONID	INT

PERSON

<u>PERSONID</u>	<u>PK INT</u>
VORNAME	VARCHAR(30)
NACHNAME	VARCHAR(30)

keine BEZIEHUNG
no RELATION

Table Name: DVD

Object description (comment):

Columns	Column Name
Unique Keys	DVIDID
Check constraints	ORIGINALTITE
Foreign Keys	UNTERTITEL
Indexes	BESCHREIBU
References	REGIE
Triggers	DAUERMIN
Extended Properties	KINOSTART
Statistics	GENRE
DDL	FSK
Virtual	LAND
	BEWERTUNG

Edit attribute Column1

Name: PERSONID

Properties:

Name	Value
Type	int
Length	100
Scale	
Precision	
Not Null	[]
Identity	[]
Default	
Collation	
Computed Definition	
Persisted	[]
Description	

DVD - Persist Changes

SQL Preview:

```
ALTER TABLE MedienBibliothek.dbo.DVD ADD PERSONID int NULL;
```

Open Editor Copy Cancel Persist

Fremdschlüssel

- Ein Fremdschlüssel ist ein Attribut (bzw. Attributkombination) einer Tabelle, welches auf einen Primärschlüssel einer anderen Tabelle verweist.
- Er dient als Verweis zwischen zwei Datensätzen, die inhaltlich miteinander in Verbindung stehen.

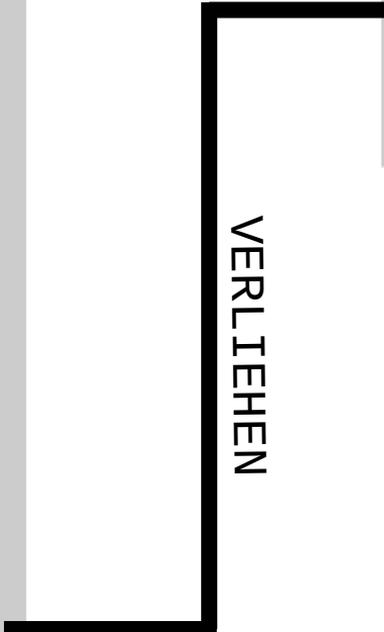
DVD

<u>DVDID</u>	PK	INT
ORIGINALTITEL		VARCHAR(30)
UNTERTITEL		VARCHAR(30)
BESCHREIBUNG		VARCHAR(500)
REGIE		VARCHAR(30)
DAUER		INT
KINOSTART		DATE
GENRE		VARCHAR(30)
FSK		INT
LAND		VARCHAR(30)
BEWERTUNG		DECIMAL(4,2)
PERSONID	FK	INT

PERSON

<u>PERSONID</u>	PK	INT
VORNAME		VARCHAR(30)
NACHNAME		VARCHAR(30)

VERLIEHEN



*<master> Script-5 DVD X *<master> Script-6 PERSON

Properties Data ER

Table Name:
 Object description (comment)

Columns
Unique Keys
Check constraints
Foreign Keys
Indexes
References
Triggers
Extended Properties
Statistics
DDL
Virtual

Create foreign key | Create foreign key for table "DVD"

Table:

Container: dbo ▼

Reference table:

- DVD
- PERSON**

Unique Key: PERSON_PK (Primary Key) ▼

Columns:

Column	Column Type	Ref Column	Ref Column Type
123 PERSONID	int	123 PERSONID	int

On Delete: No Action ▼ On Update: No Action ▼

Cancel OK

DVD - Persist Changes

SQL Preview:

```
ALTER TABLE MedienBibliothek.dbo.DVD ADD CONSTRAINT DVD_FK FOREIGN KEY (PERSONID) REFERENCES MedienBibliothek.dbo.PERSON(PERSONID);
```

Open Editor

Copy

Cancel

Persist

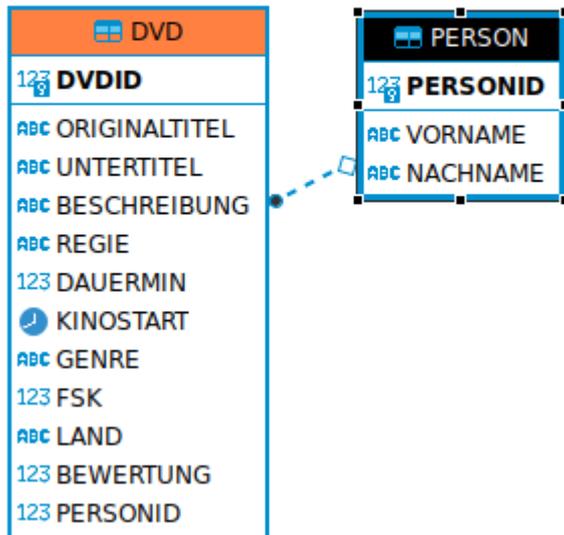
*<master> Script-5

*<master> Script-6

Properties

Data

ER Diagram



Confirm

Check results

- ✓ Import source
- ✓ Input file(s)
- ✓ Tables mapping
- ✓ Data load settings
- ✓ Confirm

Objects

Source Container	Source	Target Container	Target
 /home/thomas/Desk	 PERSON.csv	 master	 MedienBibliothek.dbo

Source settings

CSV settings:

Extension: csv,tsv,txt
 Encoding: utf-8
 Column delimiter: ;
 Header position: top
 Quote char: "
 Escape char: \
 Date/time format: yyyy-MM-dd[HH:mm:ss[.SS]
 Trim whitespaces: false
 Sample rows count: 100
 Minimum column length: 50

Target settings

Database settings:

Open new connection(s): Yes
 Use transactions: Yes
 Do Commit after row insert: 10000
 Use multi-row Insert: No
 Multi-row insert batch size: 500
 Skip bind values: No
 Disable batches: No
 Ignore duplicate rows: No
 Method for duplicate key case:
 Transfer auto-generated columns:
 Disable referential integrity: No
 Use bulk load: No
 Truncate before load: No

Save task



	PERSONID	VORNAME	NACHNAME
1	0	Thomas	Griesmayer
2	1	Susi	Griesmayer
3	2	Max	Berger
4	3	Alexandra	Meier

< Back

Proceed

Cancel

Next >

DVD | Enter a SQL expression to filter results (use Ctrl+Space)

	123 DVDID	ABC ORIGINALTITEL	ABC UNTERTITEL	123 PERSONID	ABC BESCHREIBUNG
1	0	Casablanca		0	Er schaut ihr wieder in die Augen
2	1	James Bond 007	Octopussy	[NULL]	Im 13. Bond-Abenteuer zeigt F
3	2	Der Pate		0	Francis Ford Coppolas legendä
4	3	Taxi Driver		1	Die Wiederaufführung dieses M
5	4	Stirb langsam		[NULL]	Gangster besetzen ein Hochha

Refresh Save Cancel Export data 200 5 Inserted: 0 / Deleted: 0 / Updated: 3

4	3	Taxi Driver		456	Die
---	---	-------------	--	-----	-----

Data error

 Error synchronizing data with database

Reason:
SQL Error [547] [23000]: The UPDATE statement conflicted with the FOREIGN KEY constraint "DVD_FK". The conflict occurred in database "MedienBibliothek", table "dbo.PERSON", column 'PERSONID'.

<< Details OK