



Datenbanken und SQL

(Woche 4 - Tag 1)

Agenda

- **Referenz-Optionen**
 - ON [UPDATE,DELETE] RESTRICT
 - Definition + Motivation
 - Beispiel
 - Syntax
 - ON DELETE CASCADE
 - Definition + Motivation
 - Beispiel
 - Syntax
 - ON UPDATE CASCADE
 - Definition + Motivation
 - Beispiel + Syntax

ON [UPDATE, DELETE] **RESTRICT**

Definition + Motivation

- Referenzoptionen sind **Constraints** der Fremdschlüsselbeziehung.
- Sie dienen der bereits angesprochenen „**Referentiellen Integrität**“.
- Entsprechend haben wir auch bereits die Default-Einstellung dieser Constraints kennen gelernt: **ON [UPDATE,DELETE] RESTRICT**
- Sie führen dazu, dass (bei aktivierter Fremdschlüsselbeziehung) kein Primärschlüsselwert eines Datensatzes verändert oder gelöscht werden kann, auf den sich der Fremdschlüssel eines anderen Datensatzes bezieht.

Beispiele mit Fehlermeldungen

Aufgabenstellung:

Aktualisieren Sie bitte die ID von Kunden **1** auf den neuen Wert 10.

Da es aktuell noch keinen Kunden mit der ID 10 gibt, ist dieser Wert also noch nicht vergeben, bzw. als neuer Wert einer Kunden-ID zulässig. Dennoch erscheint nach dem folgenden Befehl eine Fehlermeldung:

UPDATE Kunde SET Kunde_ID = 10 WHERE Kunde_ID = 1;

MySQL meldet: ⓘ

```
#1451 - Kann Eltern-Zeile nicht löschen oder aktualisieren: eine Fremdschlüsselbedingung schlägt fehl ('geld_her`.`abrechnung`, CONSTRAINT `abrechnung_ibfk_1` FOREIGN KEY (`Kunde_id`) REFERENCES `kunde` (`Kunde_ID`))
```

Die selbe Fehlermeldung erscheint im folgenden Fall:

DELETE FROM Kunde WHERE Kunde_ID = 1;

Die Erklärung für diese Fehlermeldung wird bereits durch deren Kommentar angedeutet:

In der Tabelle Abrechnung gibt es Datensätze, die sich auf den Kunden **1** beziehen. Diese Referenz ginge bei einer Änderung der ID (oder dem Löschen des Datensatzes von Kunde **1**) verloren => Die **Referentielle Integrität** der Datenbank wäre verletzt.

Beispiele ohne Fehlermeldung

Aufgabenstellung:

Aktualisieren Sie bitte die ID von Kunden **6** auf den neuen Wert 10.

Weiterhin gilt: Da es aktuell noch keinen Kunden mit der ID **10** gibt, ist dieser Wert also noch nicht vergeben, bzw. als neuer Wert einer Kunden-ID zulässig. Diesmal erscheint jedoch keine Fehlermeldung:

UPDATE Kunde **SET** Kunde_ID = **10** **WHERE** Kunde_ID = **6**;

Kunde_ID	Vorname	Nachname	Email
1	Elli	Rot	rot@xyz.de
2	Vera	Deise	deise@xyz.de
3	Witali	Myrnnow	myr@xyz.de
4	Rita	Myrnnow	myr@xyz.de
5	Eva	Hahn	ehahn@xyz.de
7	Peter	Kaufnix	nix@xyz.de
10	Gala	Nieda	gala@xyz.de

Gala Nieda hatte bisher die Kunde-ID 6. Nun lautet ihre ID 10:

Auch der folgende Code wird ohne Fehlermeldung ausgeführt:

DELETE FROM Kunde **WHERE** Kunde_ID = **10**;

Kunde_ID	Vorname	Nachname	Email
1	Elli	Rot	rot@xyz.de
2	Vera	Deise	deise@xyz.de
3	Witali	Myrnnow	myr@xyz.de
4	Rita	Myrnnow	myr@xyz.de
5	Eva	Hahn	ehahn@xyz.de
7	Peter	Kaufnix	nix@xyz.de

Kunde10 wurde gelöscht

Erklärung:

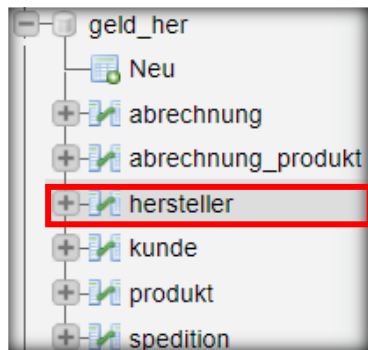
Gala Nieda hat nie die Seite von „Geld_her“ besucht.

Es gibt also keine Abrechnung, deren Fremdschlüsselwert auf Gala Nieda verweist.

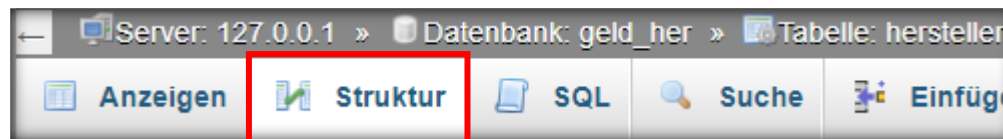
Daher kann Gala Niedas ID verändert werden, oder sogar ihr gesamter Datensatz gelöscht werden, ohne dass dadurch die Referentielle Integrität verletzt wird.

Nachweis, dass ON [UPDATE,DELETE] Default-Einstellungen sind

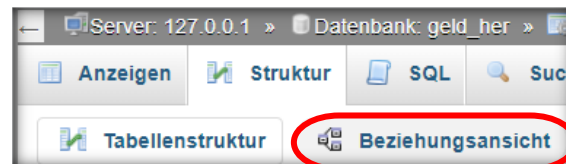
- Wir wurden soeben daran gehindert, den Primärschlüsselwert eines Datensatzes zu verändern, auf den sich der Fremdschlüssel eines anderer Datensatzes bezieht.
- Ebenso konnten wir keine Datensätze (zusammen mit ihrem Primärschlüsselwert) löschen, auf die sich Fremdschlüssel anderer Datensätze beziehen.
- Dies lag an der Default-Einstellung der Referenzoption, die wir uns durch das DBMS anzeigen lassen können:



Öffnen Sie einfach eine beliebige Tabelle, die einen Fremdschlüssel besitzt.



Betrachten Sie anschließend bitte die Struktur der ausgewählten Tabelle



Öffnen Sie nun bitte die „Beziehungsansicht“



Hier werden die aktuellen Constraints der Fremdschlüssel angezeigt

Syntax (falls man die Default-Einstellung **explizit** selbst eintragen möchte)

Um die Fremdschlüsselbeziehung zu verändern, müssen wir diese zunächst löschen.

Zur Erinnerung:

Die interne Namensvergabe eines Fremdschlüssels folgt der Form **Tabellenname_ibfk_x**

(x entspricht der Reihenfolge, mit der in der Tabelle die Fremdschlüssel eingeführt wurden.)

Im Zweifel kann der Name mittels „**SHOW CREATE TABLE** Tabellenname“ nachgeschaut werden.

Da es in der Tabelle Abrechnung nur einen Fremdschlüssel gibt, ist dessen Name klar: **Abrechnung_ibfk_1**

ALTER TABLE Abrechnung **DROP FOREIGN KEY** Abrechnung_ibfk_1;

Anschließend kann die Fremdschlüsselbeziehung neu gesetzt werden (und also z.B. mit **ON UPDATE RESTRICT**) versehen werden:

ALTER TABLE Abrechnung **ADD FOREIGN KEY**(Kunde_ID)
REFERENCES Kunde(Kunde_ID) **ON UPDATE RESTRICT**;

Bemerkungen:

Die selbe Vorgehensweise könnte auch alternativ für

ON DELETE RESTRICT vorgenommen werden.

Auch beide Referenz-Optionen können gemeinsam gesetzt werden:

ALTER TABLE Abrechnung **ADD FOREIGN KEY**(Kunde_ID)
REFERENCES Kunde(Kunde_ID) **ON UPDATE RESTRICT ON DELETE RESTRICT**;

ON DELETE CASCADE

Definition + Motivation

- Wenn wir dennoch einen Datensatz „X“ löschen wollen (oder z.B. aus Datenschutzgründen sogar müssen), auf den sich Datensätze „**Y1, Y2, Y3, ...**“ (per Fremdschlüssel) beziehen, so gelingt dies erst, **nach deren Löschung**.
- Falls nun aber Y1, Y2, und/oder Y3 ... selbst Datensätze sind, auf die sich ihrerseits wiederum Datensätze „**Z1, Z2 ...**“ beziehen, so müssten **zuallererst diese** gelöscht werden.
- Solche Löschvorgänge können also offensichtlich **beliebig kompliziert** werden.
- Um diese Kaskaden-artigen Löschvorgänge zu erleichtern, verwenden wir den Befehl **ON DELETE CASCADE**.

Beispiel

Aufgabenstellung:

Bei „Geld_her“ ist es regelmäßig notwendig, den **Datensatz eines Kunden zu löschen**.

Dies kann aber nur gelingen, wenn zuvor alle **Abrechnungen dieses Kunden gelöscht** wurden.

Dies wiederum verlangt jedoch, alle Einträge der Hilfstabelle zu identifizieren, die sich auf Abrechnungen des zu löschenden Kunden beziehen, da **zuerst diese Verkaufsposten gelöscht** werden müssen.

Um den Gesamt-Löschvorgang zu erleichtern, sollen die Referenzen zu einer **Kaskade** zusammengefasst werden. Wir betrachten diese Aufgabe am konkreten Beispiel des Löschvorgangs von Kunde 1:

Kunde_ID	Vorname	Nachname	Email
1	Elli	Rot	rot@xyz.de
2	Vera	Deise	deise@xyz.de
3	Witali	Myrnow	myr@xyz.de
4	Rita	Myrnow	myr@xyz.de
5	Eva	Hahn	ehahn@xyz.de
6	Gala	Nieda	gala@xyz.de
7	Peter	Kaufnix	nix@xyz.de

Abrechnung_ID	Kunde_id	Datum
1	1	2021-05-05
2	3	2021-10-07
3	2	2021-10-11
4	3	2021-10-16
5	5	2021-10-25
6	4	2021-11-03
7	3	2021-11-05
8	2	2021-11-09
9	1	2021-11-17
10	7	2022-02-14

Abrechnung_ID	Produkt_ID
1	2
1	4
1	4
1	5
2	3
2	5
3	1
3	1
3	1
3	5
4	2
4	3
5	1
5	2
6	3
6	2
6	5
7	2
8	3
9	1

Beispiel

Aufgabenstellung:

Bei „Geld_her“ ist es regelmäßig notwendig, den **Datensatz eines Kunden zu löschen**.

Dies kann aber nur gelingen, wenn zuvor alle **Abrechnungen dieses Kunden gelöscht** wurden.

Dies wiederum verlangt jedoch, alle Einträge der Hilfstabelle zu identifizieren, die sich auf Abrechnungen des zu löschenden Kunden beziehen, da **zuerst diese Verkaufsposten gelöscht** werden müssen.

Um den Gesamt-Löschvorgang zu erleichtern, sollen die Referenzen zu einer **Kaskade** zusammengefasst werden. Wir betrachten diese Aufgabe am konkreten Beispiel des Löschvorgangs von Kunde 1:

Kunde_ID	Vorname	Nachname	Email
1	Elli	Rot	rot@xyz.de
2	Vera	Deise	deise@xyz.de
3	Witali	Myrnow	myr@xyz.de
4	Rita	Myrnow	myr@xyz.de
5	Eva	Hahn	ehahn@xyz.de
6	Gala	Nieda	gala@xyz.de
7	Peter	Kaufnix	nix@xyz.de

Abrechnung_ID	Kunde_id	Datum
1	1	2021-05-05
2	3	2021-10-07
3	2	2021-10-11
4	3	2021-10-16
5	5	2021-10-25
6	4	2021-11-03
7	3	2021-11-05
8	2	2021-11-09
9	1	2021-11-17
10	7	2022-02-14

Abrechnung_ID	Produkt_ID
1	3
2	5
3	1
3	1
3	1
3	5
4	2
4	3
5	1
5	2
6	3
6	2
6	5
7	2
8	3

Beispiel

Aufgabenstellung:

Bei „Geld_her“ ist es regelmäßig notwendig, den **Datensatz eines Kunden zu löschen**.

Dies kann aber nur gelingen, wenn zuvor alle **Abrechnungen dieses Kunden gelöscht** wurden.

Dies wiederum verlangt jedoch, alle Einträge der Hilfstabelle zu identifizieren, die sich auf Abrechnungen des zu löschenden Kunden beziehen, da **zuerst diese Verkaufsposten gelöscht** werden müssen.

Um den Gesamt-Löschvorgang zu erleichtern, sollen die Referenzen zu einer **Kaskade** zusammengefasst werden. Wir betrachten diese Aufgabe am konkreten Beispiel des Löschvorgangs von Kunde 1:

Kunde_ID	Vorname	Nachname	Email
1	Elli	Rot	rot@xyz.de
2	Vera	Deise	deise@xyz.de
3	Witali	Myrnow	myr@xyz.de
4	Rita	Myrnow	myr@xyz.de
5	Eva	Hahn	ehahn@xyz.de
6	Gala	Nieda	gala@xyz.de
7	Peter	Kaufnix	nix@xyz.de

Abrechnung_ID	Kunde_id	Datum
2	3	2021-10-07
3	2	2021-10-11
4	3	2021-10-16
5	5	2021-10-25
6	4	2021-11-03
7	3	2021-11-05
8	2	2021-11-09
10	7	2022-02-14

Abrechnung_ID	Produkt_ID
2	3
2	5
3	1
3	1
3	1
3	5
4	2
4	3
5	1
5	2
6	3
6	2
6	5
7	2
8	3

Beispiel

Aufgabenstellung:

Bei „Geld_her“ ist es regelmäßig notwendig, den **Datensatz eines Kunden zu löschen**.

Dies kann aber nur gelingen, wenn zuvor alle **Abrechnungen dieses Kunden gelöscht** wurden.

Dies wiederum verlangt jedoch, alle Einträge der Hilfstabelle zu identifizieren, die sich auf Abrechnungen des zu löschenden Kunden beziehen, da **zuerst diese Verkaufsposten gelöscht** werden müssen.

Um den Gesamt-Löschvorgang zu erleichtern, sollen die Referenzen zu einer **Kaskade** zusammengefasst werden. Wir betrachten diese Aufgabe am konkreten Beispiel des Löschvorgangs von Kunde 1:

Kunde_ID	Vorname	Nachname	Email
2	Vera	Deise	deise@xyz.de
3	Witali	Myrnow	myr@xyz.de
4	Rita	Myrnow	myr@xyz.de
5	Eva	Hahn	ehahn@xyz.de
6	Gala	Nieda	gala@xyz.de
7	Peter	Kaufnix	nix@xyz.de

Abrechnung_ID	Kunde_id	Datum
2	3	2021-10-07
3	2	2021-10-11
4	3	2021-10-16
5	5	2021-10-25
6	4	2021-11-03
7	3	2021-11-05
8	2	2021-11-09
10	7	2022-02-14

Abrechnung_ID	Produkt_ID
2	3
2	5
3	1
3	1
3	1
3	5
4	2
4	3
5	1
5	2
6	3
6	2
6	5
7	2
8	3

Syntax

Um die Fremdschlüsselbeziehungen in Abrechnung und der Hilfstabelle zu verändern, müssen wir zunächst beide löschen. Die interne Bezeichnung des Fremdschlüssels in „Abrechnung“ ist eindeutig (**Abrechnung_ibfk_1**), da die Tabelle Abrechnung nur einen einzigen Fremdschlüssel besitzt.

Die interne Bezeichnung des Fremdschlüssels „Abrechnung_ID“ in der **Hilfstabelle** werden wir hingegen mittels **SHOW CREATE TABLE** nachschlagen, da die Hilfstabelle 2 Fremdschlüsselbeziehungen besitzt:

SHOW CREATE TABLE Abrechnung_Produkt;

Table	Create Table
Abrechnung_Produkt	<pre>CREATE TABLE `abrechnung_produkt` (`Abrechnung_ID` int(11) NOT NULL, `Produkt_ID` int(11) NOT NULL, KEY `Abrechnung_ID` (`Abrechnung_ID`), KEY `Produkt_ID` (`Produkt_ID`), CONSTRAINT <u>abrechnung_produkt_ibfk_1</u> FOREIGN KEY (`Abrechnung_ID`) REFERENCES `abrechnung` (`Abrechnung_ID`), CONSTRAINT `abrechnung_produkt_ibfk_2` FOREIGN KEY (`Produkt_ID`) REFERENCES `produkt` (`Produkt_id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4</pre> <p>Die gesuchte Nummer</p>

Syntax

Nachdem wir nun die internen Bezeichnungen beider Fremdschlüsselbeziehungen kennen, können wir diese löschen:

```
ALTER TABLE Abrechnung_Produkt DROP FOREIGN KEY Abrechnung_Produkt_ibfk_1;  
ALTER TABLE Abrechnung DROP FOREIGN KEY Abrechnung_ibfk_1;
```

Anschließend können wir die Fremdschlüsselbeziehungen samt „Kaskadierung“ einführen:

```
ALTER TABLE Abrechnung_Produkt ADD FOREIGN KEY(Abrechnung_ID)  
REFERENCES Abrechnung(Abrechnung_ID) ON DELETE CASCADE;
```

```
ALTER TABLE Abrechnung ADD FOREIGN KEY(Kunde_ID)  
REFERENCES Kunde(Kunde_ID) ON DELETE CASCADE;
```


Syntax

Nun können wir – ohne Verletzung der Referentiellen Integrität – den Kunden 1 löschen:

DELETE FROM Kunde WHERE Kunde_ID=1;

Kunde_ID	Vorname	Nachname	Email
2	Vera	Deise	deise@xyz.de
3	Witali	Myrnow	myr@xyz.de
4	Rita	Myrnow	myr@xyz.de
5	Eva	Hahn	ehahn@xyz.de
6	Gala	Nieda	gala@xyz.de
7	Peter	Kaufnix	nix@xyz.de

Abrechnung_ID	Kunde_id	Datum
2	3	2021-10-07
3	2	2021-10-11
4	3	2021-10-16
5	5	2021-10-25
6	4	2021-11-03
7	3	2021-11-05
8	2	2021-11-09
10	7	2022-02-14

Abrechnung_ID	Produkt_ID
2	3
2	5
3	1
3	1
3	1
3	5
4	2
4	3
5	1
5	2
6	3
6	2
6	5
7	2
8	3

ON UPDATE CASCADE

Definition + Motivation

- Falls wir den **Primärschlüsselwert eines Datensatzes ändern** wollen (üblicherweise allerdings nicht empfohlen) können wir auch dies mittels Kaskadierung bequem und ohne Verletzung der Referentiellen Integrität durchführen.
- Anders als beim Kaskaden-förmigen Löschvorgang, wird die notwendige „**Kaskadierungstiefe**“ bei einem UPDATE allerdings in der Regel geringer sein. Siehe hierzu das folgende Beispiel.

Beispiel + Syntax

Aufgabenstellung:

Wir wollen die Kunde_ID von Kunden 1 auf den Wert 10 ändern. (Dieser Wert ist wie bereits erwähnt noch „frei“.)

Dann müssen aber auch die Fremdschlüsselwerte aller Abrechnungen geändert werden, die sich bisher auf Kunden 1 bezogen, und die sich nun zukünftig auf Kunden 10 beziehen sollen.

Eine weiterführende Kaskadierung wird allerdings (im Unterschied zu ON DELETE CASCADE) nicht notwendig sein, da sich die Einträge der Hilfstabelle nicht auf die Kunde-ID, sondern auf die unveränderte(!) Abrechnungs-ID der Tabelle „Abrechnung“ beziehen. Es reicht also der folgende Code:

ALTER TABLE Abrechnung **DROP FOREIGN KEY** Abrechnung_ibfk_1;


ALTER TABLE Abrechnung **ADD FOREIGN KEY**(Kunde_ID)
REFERENCES Kunde(Kunde_ID) **ON UPDATE CASCADE**;

UPDATE Kunde **SET** Kunde_ID=10 **WHERE** Kunde_ID=1;

Kunde_ID	Vorname	Nachname	Email
2	Vera	Deise	deise@xyz.de
3	Witali	Myrnov	myr@xyz.de
4	Rita	Myrnov	myr@xyz.de
5	Eva	Hahn	ehahn@xyz.de
6	Gala	Nieda	gala@xyz.de
7	Peter	Kaufnix	nix@xyz.de
10	Elli	Rot	rot@xyz.de

Abrechnung_ID	Kunde_Id	Datum
1	10	2021-05-05
2	3	2021-10-07
3	2	2021-10-11
4	3	2021-10-16
5	5	2021-10-25
6	4	2021-11-03
7	3	2021-11-05
8	2	2021-11-09
9	10	2021-11-17
10	7	2022-02-14

Gemeinsame Übung („Live-Coding“) -> A_04_01_01



Aufgabe_04_01_01

Formulieren Sie bitte entsprechende SQL-Anweisungen für folgende Aufgabstellungen:

- Der Besitzer von „Geld_her“ wollte eine Spedition löschen, erhielt allerdings eine Fehlermeldung. Ändern Sie bitte die Struktur der Datenbank, damit dies zukünftig ohne Verletzung der Referentiellen Integrität möglich ist.
- Alle Kunden (Kunde-ID und Vorname), alle Produkte (Produkt-ID und Produktnamen) und alle Hersteller (Hersteller-ID und Herstellername) sollen unter den Unterschriften „ID“ und „Entitäten-Name“ ausgegeben werden.
- Für alle Kunden (Vorname und Nachname) sollen alle Speditionen (Name) ausgegeben werden, die ihn bereits mindestens 1-mal belieferten. Es sollen aber auch Kunden ausgegeben werden, die bisher noch nichts bestellten, so wie auch Speditionen aufgeführt werden sollen, die bisher noch kein Produkt von „Geld_her“ ausgeliefert haben.


Zur Erläuterung der Aufgabenstellung zeigen wir die gewünschte Ausgabetable:

Vorname	Nachname	Spedition_name
Eli	Rut	Speedvan GmbH
Eli	Rut	Rockett,ologic AG
Eli	Rut	Turbo Transport
Vera	Deise	Speedvan GmbH
Vera	Deise	Turbo Transport
Vera	Deise	Rockett,ologic AG
Wital	Mymow	Rockett,ologic AG
Wital	Mymow	Turbo Transport
Wital	Mymow	Speedvan GmbH
Rita	Mymow	Rockett,ologic AG
Rita	Mymow	Speedvan GmbH
Rita	Mymow	Turbo Transport
Eva	Hahn	Speedvan GmbH
Gala	Nieds	NULL
Peter	Kaufnix	NULL
NULL	NULL	Parkhrur
NULL	NULL	Krieghrur
NULL	NULL	Ganzal Lenn

WBS TRAINING AG
Lorenzweg 5
D-12099 Berlin
Amtsgericht Berlin HBR 68531
Sitz der Gesellschaft: Berlin

Vorstand:
Heinrich Krombholz,
Joachim Giese
Aufsichtsrat (Vorsitz): Dr. Daniel Stadler
USt-IDNr.: DE 209 768 248

GLS Gemeinschaftsbank eG
IBAN: DE18 4306 0967 1146 1814 00
BIC: GENODEM3GLS



Vielen Dank für Ihre Aufmerksamkeit!

