LS4-2: SQL – Data Manipulation Language (DML) - LOESUNG

# Situation

In der letzten Woche konnte nun das Datenbankmodell vollständig auf dem Server umgesetzt werden. Jedoch fehlen in den neu hinzugekommen Tabellen noch die entsprechenden Daten.

Liebe Kolleginnen und Kollegen – bald ist es geschafft. Die Datenbank für den Online-Shop der Höllental-Senfmühle ist fast fertig. Die letzten Tabellen wurden inzwischen hinzugefügt. Allerdings fehlen für die beiden Tabellen ‚mindestbestellmenge‘ und ‚lieferer‘ noch die entsprechenden Datensätze. Ich bitte Sie, das umgehend nachzuholen und die Daten zu vervollständigen.

Außerdem hat uns der Geschäftsführer noch eine Mitteilung über ein paar Datenänderungen zu kommen lassen. Ich leite Ihnen diese Information weiter, dann können Sie die Daten bei dieser Gelegenheit auch noch anpassen. Danke!

Welches Problem/Aufgabe stellt sich dar?

Welche Lösungsmöglichkeit(n) besteht(en)?

# Handlungsaufträge

1. Informieren Sie sich über die SQL-Befehle der ‚Data Manipulation Language‘ (DML) und ergänzen Sie den Infotext.

2. Bearbeiten Sie die Datenbank der Höllental-Senfmühle und fügen Sie die fehlenden Datensätze hinzu.

Datensätze bearbeiten

**AA**

**AA**

In der Datenbank des Online-Shops fehlen einerseits noch Daten. Andererseits sollen bestehende Daten bearbeitet werden, weil sich diese geändert haben. Bearbeiten Sie diesbezüglich die folgenden Aufgabenstellungen.

1. **Mindestbestellmengen und Liefererinformationen einpflegen**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Tabelle lieferer* | | | | | |
| **# lieferer\_id** | **firmenname** | **strasse** | **# plz\_id** | **liefervertragsbeginn** | **laufzeitjahre** |
| 1 | Neumann KG | Marienstrasse 24 | 92224 | 01.01.2020 | 8 |
| 2 | SUN-Senf GmbH | Gradestrasse 60 | 12347 | 24.05.2020 | 12 |
| […] |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| *Tabelle mindestbestellmenge* | | |
| **# artikel\_id** | **# lieferer\_id** | **mindestbestellmenge** |
| 1 | 1 | 25 |
| 1 | 9 | 18 |
| […] |  |  |

1. **Passwortänderungen einpflegen**

* Passwortänderung für Dietlinde Neumann: altes PW = jw3PQ612, neues PW = 78rifp0!FE
* Passwortänderung für Jens-Uwe Heine: altes PW = FgJ7d8X3, neues PW = hd!&93KM

1. **Kundeninformationen löschen**

Kundin Grit Schreiner hat Ihren Account gelöscht – bitte Datenbank bereinigen

Kunde Wolf Janssen hat seinen Account gelöscht – bitte Datenbank bereinigen

Lösung zu 1 (Tabelle ‚lieferer‘)

INSERT INTO `lieferer` (`lieferer\_id`, `firmenname`, `strasse`, `plz\_id`, `liefervertragsbeginn`, `laufzeitjahre`) VALUES (1, 'Neumann KG', 'Marienstrasse 24', '92224', '2020-01-01', 8);

INSERT INTO `lieferer` (`lieferer\_id`, `firmenname`, `strasse`, `plz\_id`, `liefervertragsbeginn`, `laufzeitjahre`) VALUES (2, 'SUN-Senf GmbH', 'Gradestrasse 60', '12347', '2020-05-24', 12);

Lösung zu 1 (Tabelle ‚mindestbestellmenge‘)

INSERT INTO `mindestbestellmenge` (`artikel\_id`, `lieferer\_id`, `mindestbestellmenge`) VALUES (1, 1, 25);

INSERT INTO `mindestbestellmenge` (`artikel\_id`, `lieferer\_id`, `mindestbestellmenge`) VALUES (3, 1, 20);

Lösung zu 2 (Passwortänderung)

UPDATE login SET passwort = ‘78rifp0!FE‘

WHERE login\_id = 12;

UPDATE login SET passwort = ‘hd!&93KM ‘

WHERE login\_id = 34;

Lösung zu 2 (Kundendaten löschen)

DELETE FROM kunden

WHERE kunde\_id = 18;

DELETE FROM kunden

WHERE kunde\_id = 36;

SQL – Data Manipulation Language (DML)

# Datensätze einfügen: INSERT

Mit der INSERT-Anweisung werden Datensätze oder Teile von Datensätze in eine bestehende Tabelle eingefügt. Kurz: die Tabelle wird mit Werten gefüllt.

Syntax Datensatz einfügen:

INSERT INTO tabellenname [(spaltenname1, spaltenname2, …)]

VALUES (wert1, wert2, …);

Anmerkung: Spaltennamen sind optional, wenn alle Spalten der Reihe gefüllt werden sollen!

Beispiele:

Ausgangssituation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tabelle Kunde* |  |  |  |  |
| **# K\_Nr** | **Nachname** | **Vorname** | **Strasse** | **Wohnort** |
|  |  |  |  |  |

Alternative 1: Alle Werte eines Datensatzes sind verfügbar

INSERT INTO Kunde

VALUES (1, ‘Schmidt‘, ‘Sarah‘, ‘Maxstrasse‘, ‘Fürth‘);

Alternative 1: Ergebnis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tabelle Kunde* |  |  |  |  |
| **# K\_Nr** | **Nachname** | **Vorname** | **Strasse** | **Wohnort** |
| 1 | Schmidt | Sarah | Maxstrasse | Fürth |

Alternative 2: Teile eines Datensatzes sind verfügbar

INSERT INTO Kunde (K\_Nr, Vorname, Wohnort)

VALUES (2, ‘Amira‘, ‘Nürnberg‘);

Alternative 2: Ergebnis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tabelle Kunde* |  |  |  |  |
| **# K\_Nr** | **Nachname** | **Vorname** | **Strasse** | **Wohnort** |
| 1 | Schmidt | Sarah | Maxstrasse | Fürth |
| 2 |  | Amira |  | Nürnberg |

# Datensätze einfügen: UPDATE

Mit der UPDATE-Anweisung werden Datenwerte einzelner oder mehrerer Reihen geändert.

Syntax Datensatz bearbeiten:

UPDATE tabellenname

SET spaltenname1 = neuer\_Wert1 [, spaltenname2 = neuer\_Wert2, …]

[WHERE bedingung];

Anmerkung: WHERE-Bedingung ist optional! Sollte aber gut überlegt sein, damit es eindeutig ist d. h., dass wirklich nur das geändert wird, was geändert werden soll!

Beispiele:

Ausgangssituation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tabelle Kunde* |  |  |  |  |
| **# K\_Nr** | **Nachname** | **Vorname** | **Strasse** | **Wohnort** |
| 1 | Schmidt | Sarah | Maxstrasse | Fürth |
| 2 | Castelli | Amira | Brückenstrasse | Nürnberg |

Alternative 1: Attributwerte in bestimmten Datensätze ändern ⭢ Bedingung muss formuliert werden:

UPDATE Kunde SET Nachname = ‘Schmidt-Reis‘

WHERE K\_Nr = 1;

Alternative 1: Ergebnis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tabelle Kunde* |  |  |  |  |
| **# K\_Nr** | **Nachname** | **Vorname** | **Strasse** | **Wohnort** |
| 1 | *Schmidt-Reis* | Sarah | Maxstrasse | Fürth |
| 2 | Castelli | Amira | Brückenstrasse | Nürnberg |

Alternative 2: Attributwerte in allen Datensätzen ändern ⭢ keine Bedingung angeben:

UPDATE Kunde SET Strasse = ‘Schulstrasse‘;

Alternative 2: Ergebnis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tabelle Kunde* |  |  |  |  |
| **# K\_Nr** | **Nachname** | **Vorname** | **Strasse** | **Wohnort** |
| 1 | Schmidt-Reis | Sarah | *Schulstrasse* | Fürth |
| 2 | Castelli | Amira | *Schulstrasse* | Nürnberg |

# Datensätze löschen: DELETE

Mit der DELETE-Anweisung werden Datenwerte einzelner oder mehrerer Reihen gelöscht.

Syntax Datensatz löschen:

DELETE FROM tabellenname

[WHERE bedingung];

Anmerkung: WHERE-Bedingung ist optional! Sollte aber gut überlegt sein, damit es eindeutig ist d. h., dass wirklich nur das gelöscht wird, was geändert werden soll!

Beispiele:

Ausgangssituation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tabelle Kunde* |  |  |  |  |
| **# K\_Nr** | **Nachname** | **Vorname** | **Strasse** | **Wohnort** |
| 1 | Schmidt-Reis | Sarah | Schulstrasse | Fürth |
| 2 | Castelli | Amira | Schulstrasse | Nürnberg |

Alternative 1: Einzelnen Datensatz löschen ⭢ Bedingung muss formuliert werden:

DELETE FROM Kunde

WHERE K\_Nr = 1;

Alternative 1: Ergebnis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tabelle Kunde* |  |  |  |  |
| **# K\_Nr** | **Nachname** | **Vorname** | **Strasse** | **Wohnort** |
| 2 | Castelli | Amira | Schulstrasse | Nürnberg |

Alternative 2: Alle Datensätze löschen ⭢ keine Bedingung angeben:

DELETE FROM Kunde;

Alternative 2: Ergebnis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tabelle Kunde* |  |  |  |  |
| **# K\_Nr** | **Nachname** | **Vorname** | **Strasse** | **Wohnort** |