LS 11: SQL-Data Definition Language (DDL) -LÖSUNG

# Situation

Nach den ersten Funktionstests haben Sie festgestellt, dass die Datenbank nicht vollständig implementiert ist. Das relationale Datenbankmodell weist neben den bereits im System hinterlegten Tabellen noch zwei weitere Tabellen auf.

Herzlichen Dank für den ausführlichen Funktionstest. Wie Sie festgestellt haben, wurden bei der Umsetzung der Datenbank die Tabellen ‚mindestbestellmenge‘ und ‚lieferer‘ nicht berücksichtigt.

Ich bitte Sie daher, die implementierte SQL-Datenbank zu erweitern und die beiden Tabellen hinzuzufügen. Berücksichtigen Sie bei der Umsetzung das relationale Datenbankmodell, um für beide Tabelle alle Attribute und alle Beziehungen zu hinterlegen.

Welches Problem/Aufgabe stellt sich dar?

Fehlende Tabellen müssen angelegt werden.

Welche Lösungsmöglichkeit(n) besteht(en)?

Sprachbereich Data Defintion Language verwenden

# Handlungsaufträge

1. Informieren Sie sich über die SQL-Befehle der ‚Data Definition Language‘ (DDL).

2. Lösen Sie die vorbereitenden Übungsaufgaben zur DDL.

3. Erstellen Sie die Tabellen ‚mindestbestellmenge‘ und ‚lieferer‘ mit deren Attributen und Beziehungen

5. Erstellen Sie die Beziehungen zu den Nachbartabellen.

Datenbank erweitern

Die aktuelle Datenbank des Online-Shops ist noch unvollständig. Damit sie vollständig wird, müssen die Tabellen ‚mindestbestellmenge‘ und ‚lieferer‘ hinzugefügt werden.

1. **Erstellen Sie die Tabellen ‚mindestbestellmenge‘ und ‚lieferer‘. Beachten Sie dabei folgende Konfiguration:**
* Datentypen ‚mindestbestellmenge‘:
*artikel\_id = integer, lieferer\_id = integer, mindestbestellmenge = double;*
* Datentypen ‚lieferer‘:
*lieferer\_id = integer, firmenname = varchar (255), strasse = varchar (50), plz\_id = varchar (11), liefervertragsbeginn = date, laufzeitjahre = integer;*
* NOT NULL-Werte ‚mindestbestellmenge‘: *artikel\_id, lieferer\_id*;
* NOT NULL-Werte ‚lieferer‘: *lieferer\_id, firmenname, strasse*;
* Belegen Sie außerdem das Attribut *‚lieferer\_id‘* als Primärschlüssel
1. **Stellen Sie die Beziehungen zu den Nachbartabellen her. Legen Sie dazu die Fremdschlüssel fest.**

artikel

**artikel\_id (PK)**

bezeichnung

**warengruppe\_id (FK)**

geschlecht

einkaufspreis

nettoverkaufspreis

mindestbestellmenge

**artikel\_id (FK)**

**lieferer\_id (FK)**

mindestbestellmenge

lieferer

**lieferer\_id (PK)**

firmenname

strasse

**plz\_id (FK)**

liefervertragsbeginn

laufzeitjahre

plz

**plz\_id (PK)**

ort

1

n

n

1

n

1

Tabelle vorhanden

Tabelle fehlt

Tabelle fehlt

Tabelle vorhanden

**Datenbanktyp InnoDB vs. MyISAM**

**ACHTUNG!** Wenn Sie mit phpmyadmin die Tabellen ‚lieferer‘ bzw. ‚mindestbestellmenge‘ erzeugen, dann werden diese vermutlich mit dem Datenbanktyp „InnoDB“ deklariert. Ist dies der Fall, so können keine Fremdschlüsselbeziehungen zu den bestehenden Tabellen (‚artikel‘ bzw. ‚plz‘) hergestellt werden, weil diese mit dem Datenbanktyp „MyISAM“ deklariert sind (detaillierte Erläuterung: <https://bit.ly/3OCDo67>).

Vorgehensweise: Wandeln Sie die Engine der entsprechenden Tabellen (‚lieferer‘ bzw. ‚mindestbestellmenge‘) mit folgendem SQL-Statement um:

**alter table mindestbestellmenge ENGINE=myisam;**

Lösung zu 1 (Tabelle ‚mindestbestellmenge‘)

CREATE TABLE mindestbestellmenge

 ( artikel\_id INTEGER NOT NULL,

 lieferer\_id INTEGER NOT NULL,

 mindestbestellmenge DOUBLE);

Lösung zu 1 (Tabelle ‚lieferer‘)

CREATE TABLE lieferer

 ( lieferer\_id INTEGER NOT NULL,

 firmenname VARCHAR (255) NOT NULL,

 strasse VARCHAR (50) NOT NULL,

 plz\_id INTEGER (11),

 liefervertragsbeginn DATE,

 laufzeitjahre INTEGER,

 CONSTRAINT PK\_lieferer PRIMARY KEY (lieferer\_id));

Lösung zu 2 (Fremdschlüssel ‚artikel\_id‘)

ALTER TABLE mindestbestellmenge ADD

 CONSTRAINT FK\_mbmart FOREIGN KEY (artikel\_id)

 REFERENCES artikel (artikel\_id);

Lösung zu 2 (Fremdschlüssel ‚lieferer\_id‘)

ALTER TABLE mindestbestellmenge ADD

 CONSTRAINT FK\_mbmlief FOREIGN KEY (lieferer\_id)

 REFERENCES lieferer (lieferer\_id);

Lösung zu 2 (Fremdschlüssel ‚plz\_id‘)

ALTER TABLE lieferer ADD

 CONSTRAINT FK\_artplz FOREIGN KEY (plz\_id)

 REFERENCES plz (plz\_id);

SQL – Data Definition Language

1. **Befehl CREATE DATABASE:**

CREATE DATABASE dbLEIT-Systems;

Ergebnis des Befehls:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **dbLEIT-Systems** |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Erläuterung des Befehls:

Erstellung Datenbank „dbLeit-Systems“

1. **Befehl CREATE TABLE:**

CREATE TABLE tblMitarbeiter

 ( M\_Nr INTEGER NOT NULL,

 M\_Name VARCHAR (20) NOT NULL);

Ergebnis des Befehls:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **dbLEIT-Systems** |  |  |  |  |
|  | **tblMitarbeiter** |  |  |  |  |
|  | M\_Nr | M\_Name |  |  |  |

Erläuterung des Befehls:

Erstellung Tabelle tblMitarbeiter mit den Spalten

M\_Nr (Ganzzahl) und M\_Name (Zeichenkette mit Begrenzung auf 20 Zeichen) > Beide dürfen nicht frei gelassen werden

1. **Befehl CONSTRAINT <Regelname> PRIMARY KEY:**

CREATE TABLE tblAbteilung

 ( A\_Nr INTEGER NOT NULL,

 A\_Bezeichnung VARCHAR (20) NOT NULL,

 A\_Leitung VARCHAR (20) NOT NULL,

 CONSTRAINT PK\_Abteilung PRIMARY KEY (A\_Nr));

Ergebnis des Befehls:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **dbLEIT-Systems** |  |  |  |  |
|  | **tblAbteilung** |  |  |  |  |
|  | # A\_Nr | A\_Bezeichnung | A\_Leitung |  |  |

Erläuterung des Befehls:

Erstellung Tabelle mit Spalten

Festlegen von A\_Nr als Primärschlüssel unter dem Namen der Regel „Prim\_S\_tblAbteilung“

1. **Befehl ALTER TABLE <Tabellenname> ADD COLUMN:**

ALTER TABLE tblAbteilung ADD COLUMN A\_Raum VARCHAR (20);

Ergebnis des Befehls:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **dbLEIT-Systems** |  |  |  |  |
|  | **tblAbteilung** |  |  |  |  |
|  | # A\_Nr | A\_Bezeichnung | A\_Leitung | A\_Raum |  |

Erläuterung des Befehls:

Abänderung Tabelle:

Hinzufügen einer Spalte „A\_Raum“

1. **Befehl ALTER TABLE <Tabellenname> DROP COLUMN:**

ALTER TABLE tblAbteilung DROP COLUMN A\_Leitung;

Ergebnis des Befehls:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **dbLEIT-Systems** |  |  |  |  |
|  | **tblAbteilung** |  |  |  |  |
|  | # A\_Nr | A\_Bezeichnung | A\_Raum |  |  |

Erläuterung des Befehls:

Abänderung Tabelle:

Löschen der Spalte „A\_Leitung“

1. **Befehl ALTER TABLE <Tabellenname> CONSTRAINT <Regelname> PRIMARY KEY:**
2. ALTER TABLE tblMitarbeiter ADD
3. CONSTRAINT PK\_Mitarbeiter PRIMARY KEY (A\_Nr);

Ergebnis des Befehls:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **dbLEIT-Systems** |  |  |  |  |
|  | **tblMitarbeiter** |  |  |  |  |
|  | # M\_Nr | M\_Name |  |  |  |

Erläuterung des Befehls:

Abänderung Tabelle Mitarbeiter:

Hinzufügen eines Primärschlüssels

1. **Ausgangssituation**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **dbLEIT-Systems** |  |  |  |  |
|  | **tblMitarbeiter** |  |  |  |  |
|  | # M\_Nr | M\_Name |  |  |  |
|  |  |  |  |  |  |
|  | tblAbteilung |  |  |  |  |
|  | # A\_Nr | A\_Bezeichnung | A\_Raum |  |  |

Befehl DROP TABLE:

DROP TABLE tblMitarbeiter;

Ergebnis des Befehls:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **dbLEIT-Systems** |  |  |  |  |
|  | **tblAbteilung** |  |  |  |  |
|  | # A\_Nr | A\_Bezeichnung | A\_Raum |  |  |

Erläuterung des Befehls:

Löschen der Tabelle „tblMitarbeiter“

1. **Befehl DROP DATABASE:**

DROP DATABASE dbLEIT-Systems;

Ergebnis des Befehls:

Erläuterung des Befehls:

Löschen der gesamten Datenbank „dblLEIT-Systems“