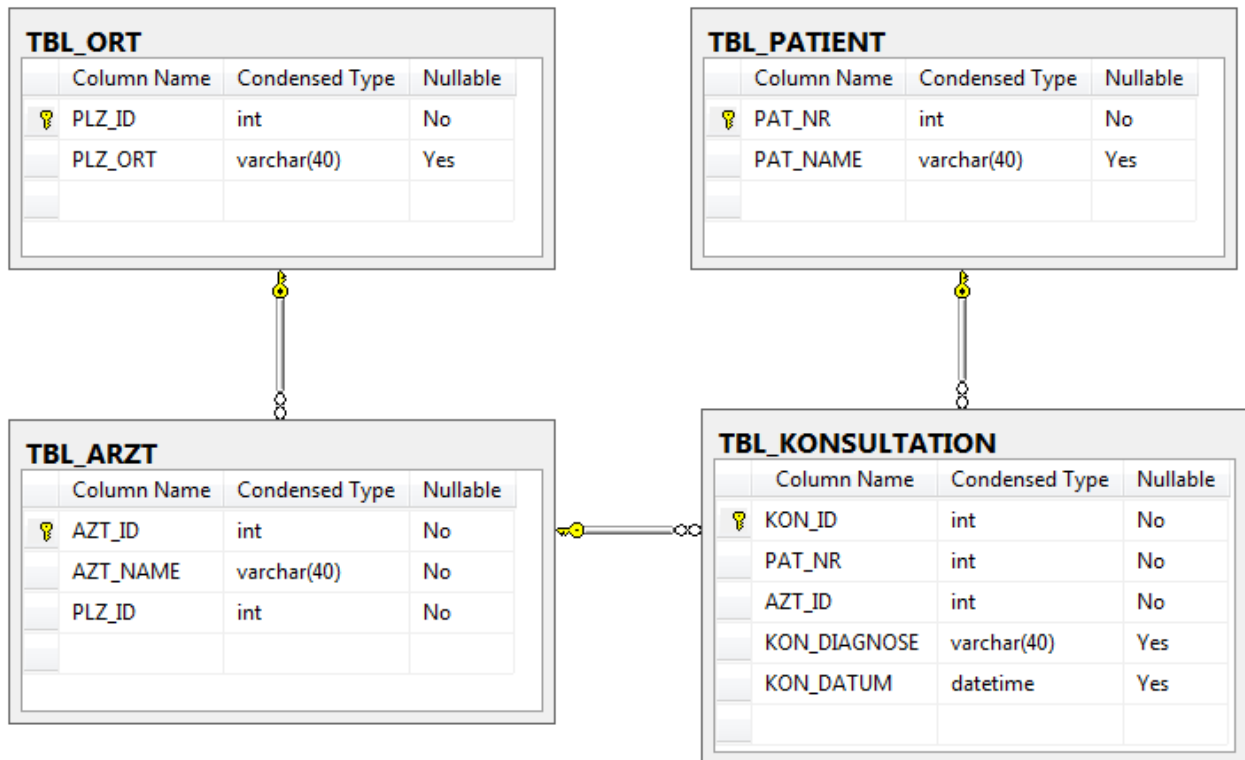


Aufgaben: Transact SQL "Trigger"

Datenmodell



Teil 1: Datenmodell implementieren (Voraussetzung)

Implementieren Sie das obige Datenmodell unter Ihrem Benutzer-Account in der SQL-Server Datenbank.

A1.1

Erstellen Sie die Textdatei CREATE_TABLES.SQL mit den CREATE TABLE Befehlen.

```
CREATE TABLE [user.]table ({column_element | table_constraint}
    [{column_element | table_constraint} ] ...)
column_element      = column datatype [column constraint]
column constraint    = column [NULL] | [NOT NULL]
table constraint     = [{UNIQUE | PRIMARY KEY}]
```

A1.2

Erstellen Sie die Beziehungen zwischen den Tabellen (referenzielle Integrität).

```
ALTER TABLE
ADD CONSTRAINT name FOREIGN KEY (attr)
REFERENCES tablename, (attr)
```

A1.3

Fügen Sie die Testdaten in die Tabellen ein. Prüfen Sie ob die referenzielle Integrität erfüllt wird.

```
INSERT INTO [user.]tabelle [ (column [,column] ...) ]
VALUES (value [,value] ...)
```

Teil 2: Triggers

Syntax:

```
CREATE TRIGGER [Besitzer.]Triggername
ON [Besitzer.]Tabellenname | Sichtname
[FOR | AFTER | INSTEAD OF] {INSERT | UPDATE | DELETE}
[WITH ENCRYPTION]
AS Sql_Anweisungen
```

A3.1

Implementieren Sie die Anforderung, dass ein Datum (KON_DATUM) einer Konsultation nicht in der Zukunft liegen darf.

a) Lösung mit Check Constraint

```
ALTER TABLE tabelle ADD CONSTRAINT ck_datum CHECK (...)
```

b) Lösung mit INSTEAD Trigger

```
CREATE TRIGGER trigger_name ON tabelle INSTEAD OF INSERT
BEGIN ... END
```

A3.2

Stellen Sie sicher, dass ein Arzt pro Tag max. 5 Konsultationen ausführen darf.

a) Lösung mit AFTER INSERT

```
CREATE TRIGGER trigger_name ON tabelle AFTER INSERT AS
BEGIN ... END
```

b) Lösung mit INSTEAD Trigger

```
CREATE TRIGGER trigger_name ON tabelle INSTEAD OF INSERT
BEGIN ... END
```

A3.3

Erstellen Sie ein Trigger welcher beim löschen eines Patienten automatisch alle zugehörigen Konsultationen mit löscht.

```
CREATE TRIGGER trigger_name ON tabelle AFTER [DELETE | INSERT | UPDATE] AS
BEGIN ... END
```

A3.4

Erstellen Sie ein Trigger welcher sämtliche Modifikationen in der Patiententabelle in einer Log-Tabelle protokolliert. Die Log-Tabelle muss vorgängig von Ihnen angelegt werden.

Attribute für Log-Tabelle (ID, Datum, Benutzernamen, Meldungstext)

```
CREATE TRIGGER trigger_name trigger ON tabelle AFTER UPDATE AS
BEGIN ... END
user_name( )
getdate()
```

A3.5

Wenn der Anwender einen Datensatz in der Patiententabelle ändert, soll automatisch in einem Feld (z.B. letzte_bearbeitung (DateTime)) das Änderungsdatum gesetzt werden. Wie lässt sich diese Anforderung lösen?