



SQL

Structured Query Language

Teil 1_B DDL:

Constraints



DDL-Constraints

1. Definiton

Das Setzen von Constraints (deutsch „Einschränkung“) ist ein wichtiger Teil beim Anlegen von Datenbank-Tabellen. SQL-Constraints werden benötigt, um Regeln für die Daten in einer Tabelle zu beschreiben.

Sie werden eingesetzt, um den Datentyp einzuschränken, der in eine Tabelle aufgenommen werden kann. Dies gewährleistet die Genauigkeit und Zuverlässigkeit der Daten in der Tabelle. Wenn es eine Verletzung zwischen der Beschränkung und der Datenaktion gibt, wird die Aktion abgebrochen.

Constraints können auf Spalten- oder Tabellenebene definiert werden. Die Einschränkungen auf Spaltenebene gelten für eine Spalte und Einschränkungen auf Tabellenebene gelten für die gesamte Tabelle.

2. Möglichkeiten von Constraints

- **NOT NULL**
Stellt sicher, dass eine Spalte keinen NULL-Wert haben kann.
- **UNIQUE**
Stellt sicher, dass alle Werte in einer Spalte unterschiedlich sind.
- **PRIMARY KEY**
Eine Kombination aus NOT NULL und UNIQUE ist automatisch miteingeschlossen. Identifiziert jede Zeile in einer Tabelle eindeutig.
- **FOREIGN KEY**
Verhindert Aktionen, die Verknüpfungen zwischen Tabellen zerstören würden.
- **CHECK**
Stellt sicher, dass die Werte in einer Spalte eine bestimmte Bedingung erfüllen.
- **DEFAULT**
Legt einen Standardwert für eine Spalte fest, wenn kein Wert angegeben ist.
- **CREATE INDEX**
Wird verwendet, um Daten sehr schnell aus der Datenbank zu erstellen und abzurufen.

3. Beispiele für Einschränkungen

- NOT NULL / DEFAULT

```
CREATE TABLE Persons (  
  ID int NOT NULL,  
  LastName varchar(255) NOT NULL,  
  FirstName varchar(255) NOT NULL,  
  City varchar(255) DEFAULT 'Sandnes',  
  Age int  
);
```

- UNIQUE

```
CREATE TABLE Persons (  
  ID int NOT NULL UNIQUE,  
  LastName varchar(255) NOT NULL,  
  FirstName varchar(255),  
  Age int  
);
```

- PRIMARY KEY

```
CREATE TABLE Persons (  
  ID int NOT NULL,  
  LastName varchar(255) NOT NULL,  
  FirstName varchar(255),  
  Age int,  
  CONSTRAINT PK_Person PRIMARY KEY (ID,LastName)  
);
```

- **FOREIGN KEY**

Create-Statement:

```
CREATE TABLE Orders (  
    OrderID int NOT NULL,  
    OrderNumber int NOT NULL,  
    PersonID int,  
    PRIMARY KEY (OrderID),  
    CONSTRAINT FK_PersonOrder FOREIGN KEY (PersonID)  
    REFERENCES Persons(PersonID)  
);
```

Alter-Statement ADD:

```
ALTER TABLE Orders  
ADD CONSTRAINT FK_PersonOrder  
FOREIGN KEY (PersonID) REFERENCES Persons(PersonID);
```

Alter-Statement DROP:

```
ALTER TABLE Orders  
DROP CONSTRAINT FK_PersonOrder;
```

- **CHECK**

```
CREATE TABLE Persons (  
    ID int NOT NULL,  
    LastName varchar(255) NOT NULL,  
    FirstName varchar(255),  
    Age int,  
    City varchar(255),  
    CONSTRAINT CHK_Person CHECK (Age>=18 AND City='Sandnes')  
);
```