

Microsoft SQL

- JOIN Tabelle in andere Tabelle auslagern
- SELECT Befehle
 - Uhrzeit im Select-Befehl modifizieren
- Administration
 - Transaktionsprotokolle verkleinern

JOIN Tabelle in andere Tabelle auslagern

Wenn du zwei Tabellen oder mehr mit **JOIN** miteinander verknüpfen möchtest, kannst du die Tabelle auch in eine andere Tabelle auslagern. (WICHTIG: Die Tabelle wird bei dem Vorgang erstellt! Du kannst also keine schon existierende Tabelle verwenden.)

```
SELECT p.Stadtname, p.PLZ, c.Country INTO tblLaender FROM tblPlace p  
INNER JOIN tblCountry c  
ON p.ID = c.ID;
```

Du erhältst nun die Ausgabe der **JOIN** Tabelle in eine andere Tabelle. Du kannst diese jetzt wie gewohnt in SQL weiterverwenden. Die alten Tabellen könntest du dann aus theoretischer Sicht löschen.

SELECT Befehle

Uhrzeit im Select-Befehl modifizieren

Einleitung

In diesem Artikel beschreibe ich kurz, wie wir im `SELECT` **Befehl** das Datum modifizieren können, ohne den Datenbestand in der Datenbank zu ändern. So können wir das Datum anpassen damit richtig auf z.B. einer Visualisierung angezeigt werden kann.

Uhrzeit und Datum anpassen

Um die Uhrzeit anpassen müssen wir im ersten `SELECT` **Teil** dort den Befehl `DATEADD` mit einbeziehen. Als Parameter geben wir hier mit an welchen Teil unserer Uhrzeit wir anpassen wollen, wie der Wert verändert werden soll und in welcher SQL-Spalte sich die **Uhrzeit** und das **Datum** befindet.

sql_select_date_1.png

Der Befehl könnte dann wie folgt aussehen:

```
DATEADD(hh, -2, timestamp) AS Uhrzeit
```

Administration

Transaktionsprotokolle verkleinern

Einleitung

In diesem Artikel erläutere ich kurz, wie wir auf unserem **Microsoft SQL Server** die **Transaktionsprotokolle (LOG-Dateien)** verkleinern können. Dadurch können wir belegten Speicherplatz auf unserem Datenbank-Server wieder freigeben.

LOG-Dateien verkleinern

Über Microsoft SQL Management Studio

Um die LOG-Dateien vom SQL-Server zu verkleinern, müssen wir im ersten Schritt das **Microsoft SQL Management Studio** öffnen. Dort wählen wir die Datenbank aus und öffnen das Kontextmenü und wählen unter **Tasks** den Punkt **Verkleinern**, und dort den Punkt **Dateien** aus.

Es öffnet sich ein Fenster, hier wählen wir unter **Dateityp** Protokoll aus und überprüfen, ob der Haken bei "**Nicht verwendeten Speicherplatz freigeben**" gesetzt ist.

Wenn wir auf **OK** drücken, wird die LOG-Datei verkleinert.

Über SQL-Befehl

Wahlweise können wir das auch über einen SQL-Befehl durchführen. Dazu müssen wir im Vorwege den Namen der LOG-Datei heraussuchen und dann den folgenden Befehl auf der entsprechenden Datenbank durchführen:

```
# Syntax
DBCC SHRINKFILE ([NAMEN DER LOG-Datei], 1);
GO

# BEISPIEL
DBCC SHRINKFILE ([DATENBANK_log], 1);
GO
```

Jetzt sollte hier auch die LOG-Datei verkleinert wurden sein.

Info: Es kann sein das der Verkleinerungsvorgang öfter durchgeführt werden muss um die gewünschte Größe zu erreichen.