

SQL-Funktionalität für Server

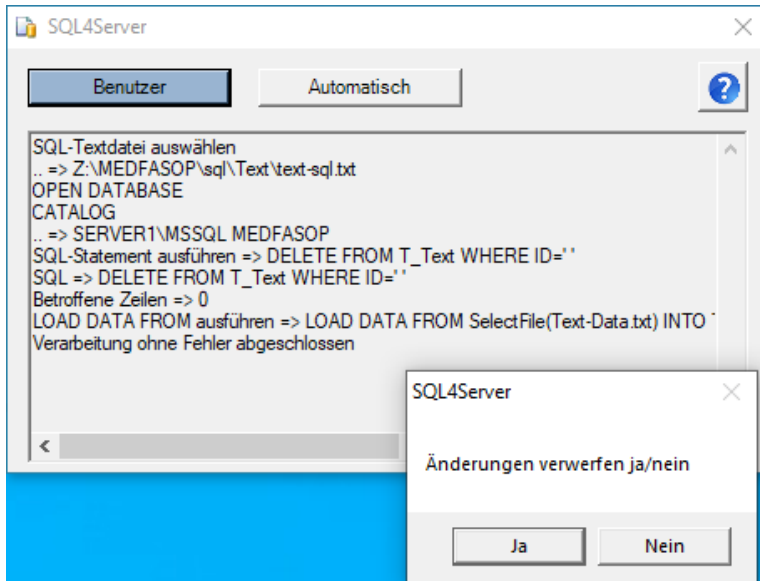
Version 2.5

INHALTSVERZEICHNIS

1	SQL aus Textdatei	3
2	SQL-Statements	4
2.1	SQL-Statements	4
2.2	Definition	5
2.3	Ergänzende Hinweise	6
3	Beispiele	7
3.1	MEDFASOP.Data.txt	7
3.2	LOAD DATA (T_ZIP).....	8
3.3	SELECT	8
3.4	TRANSFORM XML	9
3.5	SETCOL.....	9
4	Datum-Funktionen	10
4.1	DATEADD	10
4.2	GETDATE.....	10
5	Start mit Parameter.....	10


1 SQL aus Textdatei

Mit dieser Anwendung können Server-Datenbanken über SQL-Statements geändert und abgefragt werden. Sie wurde mit *Microsoft Visual Studio .NET* entwickelt und setzt voraus, dass die Statements als Textdatei mit Endung «sql.txt» vorliegen (z.B. test.sql.txt). Windows stellt dazu den Editor im Menü Zubehör zur Verfügung. Alle Statements einer Datei werden als *eine* Transaktion behandelt. Der Abschluss der Verarbeitung wird bestätigt, wobei im Modus «Benutzer» jeweils nachgefragt wird, ob Änderungen verworfen werden sollen, oder nicht. Im Modus «Automatisch» werden Änderungen im Fehlerfall verworfen, andernfalls gespeichert.



Mit «Benutzer» läuft der Vorgang standardmässig ab.

Mit «Automatisch» muss der Benutzer nur bei Handlungsalternativen eingreifen.

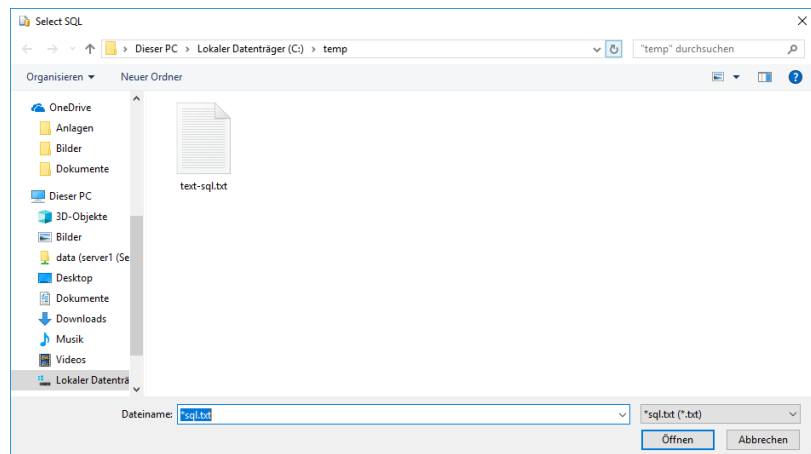
Mit  wird das Dokument der Anwendung aufgerufen.

Wesentliche Verarbeitungsschritte und Ergebnisse werden protokolliert¹.

Der File-Dialog wird situativ aufgerufen. Die Auswahl erfolgt durch Doppelklick auf die Datei. Im Kopf des Formulars ist jeweils der Kontext ersichtlich:

Select SQL: Auswahl der Datei, welche die SQL-Statements enthält.

Select Database: Auswahl der Parameter-Datei «*.Data.txt» (→ S.7)



Select Data: Auswahl der Datei welche die Load-Daten enthält.

Select XML: Auswahl der XML-Dateien welche umgeformt werden sollen².

¹ Protokoll «userid_protocol.txt» wird am Ausgangsort der SQL-Datei gespeichert (Inhalt ist nach Datum/Zeit geordnet). Die Datei sollte von Zeit zu Zeit gelöscht werden.

² Für «Select XML» ist eine Mehrfachauswahl möglich.

2 SQL-Statements

Nachfolgend sind häufig benutzte, sowie proprietäre Statements aufgeführt. Für weitergehende Informationen sei auf die Referenz des Datenbank-Herstellers verwiesen. Resultate ex *SELECT* werden im XML-Format (ADO-Funktionalität von Microsoft) geschrieben, und zwar am Ort der SQL-Datei in den Ordner *Result*. Zur Umwandlung von XML- in Textdateien steht *TRANSFORM* zur Verfügung und zum Anfügen einer Spalte mit einem vorgegebenen Wert in eine Textdatei ist es *SETCOL*.

Statements müssen immer in einer neuen Zeile beginnen und mit Semikolons abgeschlossen, oder durch eine Leerzeile voneinander getrennt werden.

Elemente in geschweiften {} Klammern sind fakultativ, solche in spitzen <> Klammern sind weiter unten definiert. Beide Klammertypen werden nie geschrieben. In SQL-Statements sind Tabellen- und Spaltennamen in eckige Klammern [] zu setzen, wenn sie mit reservierten SQL-Wörtern kollidieren. Mit .. wird die Fortsetzung des Prinzips signalisiert. Schlüsselwörter sind gross und fett geschrieben. Mit | werden Alternativen separiert. Hochgestellte Ziffern sind ab Seite 6 dokumentiert.

2.1 SQL-Statements

OPEN DATABASE <database>{;}^{1,2}

DELETE FROM <table> {**WHERE** <clause>}{;}

INSERT INTO <table> (<columns>) **VALUES** (<values>){;}

UPDATE <table> **SET** <column>=<value> {,<column>=<value> ..} {**WHERE** <clause>}{;}

SELECT <columns> **FROM** <table> {<clause>}

LOAD DATA FROM <input>^{6,10} **INTO** <table> (<columns>) **DELIMITER** <delimiter>
{**NEWID** <column>}³ {**HEADER**}⁴{;}²

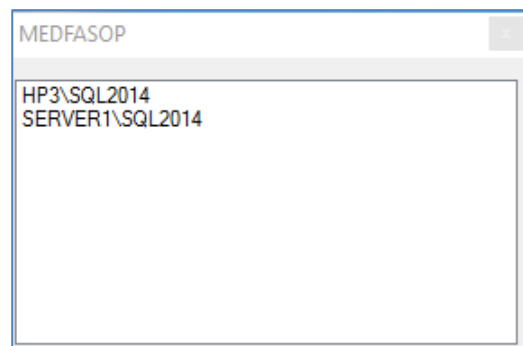
TRANSFORM XML <input>^{8,11} {**INTO** <folder>}⁹ **DELIMITER** <delimiter> {**HEADER**}⁵{;}²

SETCOL <input>¹² **DELIMITER** <delimiter> {**COLUMN** <column>}⁵ **SET** <string>{;}²

2.2 Definition	
<catalog>: <string> ¹³	z.B: medfasop
<clause>	Klausel nach SQL-Syntax
<column>: {[]column name[]}	Spaltenname
<columns>: <column> {,<column> ..}	Für LOAD wird die Auflistung von den Input-Daten bestimmt. Die Spalte von NEWID darf nicht in der Auflistung stehen.
<comment>: -- [text] --! [text] --? [text]	(2 x Bindestrich). Ergänzt mit ! wird der text angezeigt. Mit ? wird der Text auch angezeigt, aber mit der Möglichkeit, die Verarbeitung abzubrechen.
<comparison>: < > <= >= =	Vergleichsoperatoren
<database>: <input> <catalog>	«*.Data.txt» für <input> (→ S.7)
<date>	Datum ⁷ als Konstante oder Funktion (→ S.10)
<date time>	DatumZeit nach SQL-Syntax
<delimiter>: BLANK COMMA TABULATOR SEMICOLON QUOTE <string>	Leer Komma Tabulator Strichpunkt Hochkomma Zeichenkette
<expression>	Ausdruck nach SQL-Syntax
<file>	Dateiname mit/ohne Jokerzeichen zur Vorauswahl
<folder>	Pfad und Name des Ordners.
<fullpath>	Pfad und Dateiname
<function>	Funktion nach SQL-Syntax
<input>: SELECTFILE({<file>}) GETFILE(<fullpath>)	SelectFile() startet den File-Dialog von Windows.
<number>	Numerische Werte nach SQL-Syntax.
<string>	1 bis n Zeichen
<table>: {[]table name[]}	Tabellenname
<UDL>	Eine UDL-Datei (Microsoft Datenlink) wird folgendermassen erstellt: (a) eine leere Datei BEISPIEL.txt erstellen und danach txt durch udl ersetzen. (b) Doppelklick auf BEISPIEL.udl und es öffnet sich ein Fenster, wo die Eigenschaften eingestellt werden können.
<value>: '<string>' <number> <date time> <function> NEWID() ³	Text-Daten sind zwischen Hochkommata zu setzen.
<values>: <value> {,<value> ..}	Werte in Reihenfolge der aufgeführten Spalten

2.3 Ergänzende Hinweise

- ¹ Statement muss vor allen anderen Statements stehen und darf nur einmal vorkommen.
- ² Statement vollständig proprietär (kein SQL-Statement).
- ³ Funktion NEWID() wird einer Spalte zugeordnet. Sie darf nicht in der Auflistung <columns> stehen.
- ⁴ HEADER überliest erste Zeile von <input>.
- ⁵ Erstellt Kopfzeile (Spaltennamen).
- ⁶ In Textspalten wird HEX „01“ in HEX „0D0A“ (neue Zeile) umgewandelt.
- ⁷ Format Länderspezifisch: DD.MM.YYYY {HH:MM{:SS}} | MM/DD/YYYY {HH:MM{:SS}}
- ⁸ XML-Datei, welche mit ADO-Funktionalität von Microsoft erstellt wurde. Andere Dateien werden übergangen.
- ⁹ Ohne Angabe wird Datei im Input-Ordner erstellt.
- ¹⁰ DatumZeit-Spalten dürfen Lokal-, wie auch US-Format haben.
- ¹¹ In Textspalten wird HEX(0D0A) (neue Zeile) in HEX(01) umgewandelt.
- ¹² TXT-Datei. Andere Dateien werden übergangen.
- ¹³ Ein Fenster mit den verfügbaren SQL-Server-Instanzen öffnet sich. Im Kopf ist der gesuchte Katalog aufgeführt. Ein Mausklick selektiert die gewünschte Instanz.



3 Beispiele

3.1 MEDFASOP.Data.txt³

Z:\MEDFASOP\MEDFASOP.Data.txt

```
UDLCONN=\\server1\data\MEDFASOP\MEDFASOP.UDL
SQLCONN=Integrated Security=SSPI;Persist Security Info=False;Initial Catalog=MEDFASOP;Data Source=SERVER1\SQL2014
DBTEMP=\\server1\data\temp
DBLANG=german
DBDOC=MEDFASOP.Client
DOCUMENT=\\server1\data\MEDFASOP\DOCUMENT
USERPATH=\\server1\data\MEDFASOP\USER
HTMLPATH=\\server1\data\MEDFASOP\HTML
```

³ data = Freigabename

3.2 LOAD DATA (T_ZIP)

ZIP-SQL.txt

```
OPEN DATABASE GETFILE(Z:\MEDFASOP\MEDFASOP.Data.txt);

DELETE FROM T_ZIP WHERE (Country='CH' OR Country='FL');

LOAD DATA FROM SelectFile(ZIP-Data.txt) INTO T_ZIP (country,zip,city,state) DELIMITER TABULATOR NEWID TID HEADER;
```

ZIP-Data.txt (Auszug)⁴

Country	ZIP	City	State
CH	1000	Lausanne 25	VD
CH	1000	Lausanne 26	VD
CH	1008	Prilly	VD
CH	1009	Pully	VD

3.3 SELECT

SELECT-SQL.txt

```
OPEN DATABASE GETFILE(Z:\MEDFASOP\MEDFASOP.Data.txt);

SELECT T_Patient.*, T_Consultation.Consultation
FROM T_Patient INNER JOIN T_Consultation ON T_Patient.PID = T_Consultation.PID;
```

C:\ED\Projekte\SQL\Result\Admin 20050124 091235.xml (Auszug, ohne Schemadaten)

```
<rs:data>
<z:row PID="200501068073835" FirstName="Beat" LastName="Heil" DateOfBirth="05.12.1951" Sex="2" Language="3" Insurer="200501068002975"
  AddrLine2="Chasernweg 20" Country="CH" ZIP="8370" City="Sirnach" ForSearch="HEIL BEAT CHASERNWEG 20 SIRNACH" UpdateTime="2005-01-23T20:10:20"
  UpdateUser="200501068018855" Consultation="0" Diagnose="" />
<z:row PID="200501068109396" FirstName="Marianne" LastName="Bürki" DateOfBirth="05.12.2004" Sex="1" Language="0" Insurer="0" Country="CH" ZIP="8001"
  City="Zürich" ForSearch="BUERKI MARIANNE ZUERICH" UpdateTime="2005-01-11T10:31:57" UpdateUser="200501068018855"
  Consultation="0" Diagnose="" Messungen="" Gewicht (kg)="" Grösse (cm)="" BMI="" />
<z:row PID="200501068109396" FirstName="Marianne" LastName="Bürki" DateOfBirth="05.12.2004" Sex="1" Language="0" Insurer="0" Country="CH" ZIP="8001"
  City="Zürich" ForSearch="BUERKI MARIANNE ZUERICH" UpdateTime="2005-01-11T10:31:57" UpdateUser="200501068018855"
  Consultation="0" Diagnose="" Messungen="" Gewicht (kg)="" Grösse (cm)="" BMI="" />
<z:row PID="200501068109396" FirstName="Marianne" LastName="Bürki" DateOfBirth="05.12.2004" Sex="1" Language="0" Insurer="0" Country="CH" ZIP="8001"
  City="Zürich" ForSearch="BUERKI MARIANNE ZUERICH" UpdateTime="2005-01-11T10:31:57" UpdateUser="200501068018855"
  Consultation="0" Diagnose="" Messungen="" Gewicht (kg)="" Grösse (cm)="" BMI="" />
<z:row PID="200501096238096" FirstName="Peter" LastName="Neumann" DateOfBirth="05.01.1995" Sex="2" Language="3" Insurer="0" Country="CH" ZIP="8000"
  City="Zürich" ForSearch="NEUMANN PETER ZUERICH" UpdateTime="2005-01-12T11:54:52" UpdateUser="200501068018855"
  Consultation="0" Diagnose="" Messungen="" Gewicht (kg)="" Grösse (cm)="" BMI="" />
<z:row PID="200501096238096" FirstName="Peter" LastName="Neumann" DateOfBirth="05.01.1995" Sex="2" Language="3" Insurer="0" Country="CH" ZIP="8000"
  City="Zürich" ForSearch="NEUMANN PETER ZUERICH" UpdateTime="2005-01-12T11:54:52" UpdateUser="200501068018855"
  Consultation="0" Diagnose="" Messungen="" Gewicht (kg)="" Grösse (cm)="" BMI="" />
<z:row PID="200501096238096" FirstName="Peter" LastName="Neumann" DateOfBirth="05.01.1995" Sex="2" Language="3" Insurer="0" Country="CH" ZIP="8000"
  City="Zürich" ForSearch="NEUMANN PETER ZUERICH" UpdateTime="2005-01-12T11:54:52" UpdateUser="200501068018855"
  Consultation="0" Diagnose="" Messungen="" Gewicht (kg)="" Grösse (cm)="" BMI="" />
<z:row PID="200501096238096" FirstName="Peter" LastName="Neumann" DateOfBirth="05.01.1995" Sex="2" Language="3" Insurer="0" Country="CH" ZIP="8000"
  City="Zürich" ForSearch="NEUMANN PETER ZUERICH" UpdateTime="2005-01-12T11:54:52" UpdateUser="200501068018855"
  Consultation="0" Diagnose="" Messungen="" Gewicht (kg)="" Grösse (cm)="" BMI="" />
<z:row PID="200501135448187" FirstName="Max" LastName="Meier" DateOfBirth="05.12.1951" Sex="2" Language="0" Allergies="aaaaaaaa" Insurer="0"
  Country="CH" ZIP="8340" City="Hinwil" ForSearch="MEIER MAX HINWIL" UpdateTime="2005-01-23T20:13:12" UpdateUser="200501068018855"
  Consultation="0" Diagnose="" />
</rs:data>
```

⁴ Das aktuelle Verzeichnis enthält z.Z. mehr als 2200 Einträge, welche von der Post stammen (<http://www.post.ch/SiteOnLine/DE/Accueil/1,1727,5811-0,00.html> – PLZ light).

3.4 TRANSFORM XML

Transform-SQL.txt

```
TRANSFORM XML SelectFile(*.xml) DELIMITER TABULATOR HEADER;
```

C:\ED\Projekte\SQL\Result\Admin 20050124 091235.txt (Spalten rechts abgeschnitten)

PID	APID	FirstName	LastName	DateOfBirth	Sex	Language	Allergies
20050246157788			Marianne	Bürki	12.03.1995	1 3	

3.5 SETCOL

SETCOL-SQL.txt

```
SETCOL SelectFile(*.txt) DELIMITER @@ COLUMN BEREICH SET 1;
```

Datei "kapitel.txt" vor ATTACH

```
KNR@@GVON@@GBIS
00@@01.01.2001@@31.12.2199
00.01@@01.01.2001@@31.12.2199
00.01.01@@01.01.2001@@31.12.2199
00.01.02@@01.01.2001@@31.12.2017
00.01.02.01@@01.01.2001@@31.12.2017
00.01.02.02@@01.01.2001@@31.12.2017
```

Datei "kapitel.txt" nach ATTACH

```
KNR@@GVON@@GBIS@@BEREICH
00@@01.01.2001@@31.12.2199@@1
00.01@@01.01.2001@@31.12.2199@@1
00.01.01@@01.01.2001@@31.12.2199@@1
00.01.02@@01.01.2001@@31.12.2017@@1
00.01.02.01@@01.01.2001@@31.12.2017@@1
00.01.02.02@@01.01.2001@@31.12.2017@@1
```

4 Datum-Funktionen

4.1 DATEADD

DATEADD(datepart, number, date) AS DATETIME

Liefert im Funktionswert DATETIME zurück, welches zuvor um «number» erhöht, bzw. reduziert worden ist.

datepart	Ausdruck bestimmt das zu addierende Zeitintervall.
dd	Tag
dw	Wochentag
dy	Tag des Jahres
hh	Stunde
mi	Minute
mm	Monat
ms	Millisekunde
qq	Quartal
ss	Sekunde
wk	Woche
yy	Jahr
number	Bestimmt die Anzahl der zu addierenden Intervalle. Kann positiv (für ein zukünftiges Datum) oder negativ (für ein vergangenes Datum) sein.
date	Kann vom Typ DATETIME oder ein konstanter Datum-/Zeit-Ausdruck (in Hochkommas) sein, zu dem das Intervall hinzuaddiert wird.

4.2 GETDATE

GETDATE() AS DATETIME

Liefert im Funktionswert System-Datum und -Zeit.

5 Start mit Parameter

Das Programm kann mit einem Parameter, welcher die SQL-Datei mit Pfadangabe enthalten muss, aufgerufen werden und arbeitet in diesem Fall im automatischen Modus.