

Inhalt

Vorwort	XIII
1 Der SQL Server 2017 stellt sich vor	1
1.1 SQL Server – wer ist das?	2
1.1.1 Der SQL Server im Konzert der Datenbanksysteme	2
1.1.2 Entscheidungsszenarien für Datenbanksysteme	5
1.1.3 Komponenten einer Datenbankanwendung	7
1.1.4 SQL Server – das Gesamtkonzept	10
1.2 Versionen und Editionen des SQL Servers	12
1.3 SQL Server 2017 installieren	17
1.4 Datenbanken installieren und nutzen	41
1.5 Gratis: die Express Edition	47
1.6 SQL Server Feature Pack	49
2 Die grafischen Tools des SQL Server 2017	53
2.1 Die Tools im Überblick	53
2.2 Das Management Studio	56
2.3 Das Kommandozeilentool: SQLCMD	76
2.4 Der Konfigurations-Manager	78
2.5 Das SQL Server-Installationscenter	81
2.6 Der Profiler	82
2.7 Der Datenbankoptimierungsratgeber	83
2.8 Die SQL Server Data Tools	85
2.9 Der Import/Export-Assistent	89
2.10 Der SQL Server Migration Assistant	99
2.11 SQL Operations Studio	103
3 Eine neue Datenbank erstellen	105
3.1 Erstellen einer neuen Datenbank	105
3.1.1 Bestandteile einer Datenbank	105

3.1.2	Datenbank mit dem grafischen Tool anlegen	109
3.1.3	Datenbank über eine SQL-Anweisung erstellen	116
3.1.4	Datenbank mit Filestream ausstatten	117
3.2	Tabellen in der Datenbank erstellen	122
3.2.1	Tabellenfelder definieren	123
3.2.2	Spalteneigenschaften	125
3.2.3	Constraints	128
3.2.4	Indizierung	138
3.2.5	Erste Daten erfassen	146
3.3	Datenbankdiagramme einsetzen	149
3.4	Richtlinien für Benennungsregeln einsetzen	153
3.5	Was Sie noch wissen sollten	158
3.5.1	Tabellen in anderen Dateigruppen speichern	158
3.5.2	Tabellen direkt mit DDL-Anweisungen erstellen	160
3.5.3	Gefahren der grafischen Oberfläche	161
3.5.4	Berechnete Spalten integrieren	165
3.5.5	Objekte und Datenbanken skripten	168
3.6	Tabelle mit Filestream und FileTable	171
3.6.1	Tabelle mit Filestream erstellen	172
3.6.2	Objekte in einer FileTable speichern	179
3.7	Beispieldatenbank generieren	192
3.8	Speicheroptimierte Tabellen	193
3.8.1	Datenbank mit In-Memory-Filegroup erstellen	194
3.8.2	Speicheroptimierte Tabelle anlegen	196
3.8.3	Index für speicheroptimierte Tabellen	200
3.8.4	Speichernutzung beschränken	205
4	SQL – Zugriff auf Daten	207
4.1	Einsatz des Abfrage-Designers	209
4.1.1	Die Bereiche des Abfrage-Designers	209
4.1.2	Erstellen einer Abfrage	215
4.2	Sichten für den Datenzugriff gestalten	229
4.2.1	Gründe für den Einsatz von Sichten	230
4.2.2	Erstellen einer Sicht	231
4.2.3	Daten aus einer Sicht abrufen	233
4.3	SQL-Anweisungen verwenden	235
4.3.1	Data Query Language (DQL)	236
4.3.2	Data Manipulation Language (DML)	246
4.3.3	Die MERGE-Anweisung	247
4.3.4	Den Abfrage-Designer im Abfrageeditor einsetzen	252
4.4	Abfragen mit Geodaten	254
4.4.1	Typen im Geodatenmodell	255
4.4.2	Geodaten in Tabellen speichern und verwenden	261
4.4.3	Index für räumliche Daten	278

5	Transact-SQL – die Sprache zur Serverprogrammierung	283
5.1	Bestandteile und Funktionalität von Transact-SQL	285
5.1.1	Variablen und Datentypen	285
5.1.2	Benutzerdefinierte Tabellentypen	293
5.1.3	Funktionen	295
5.1.4	Kontrollstrukturen	332
5.1.5	Cursor für Datenzugriffe einsetzen	349
5.2	Transaktionen gezielt steuern	355
5.2.1	Automatische Transaktionen	356
5.2.2	Explizite und implizite Transaktionen	357
5.2.3	Benannte Transaktionen	363
5.3	SET-Optionen verwenden	364
5.4	Fehlerbehandlung in den Code einbauen	372
5.5	Sequenzen	378
5.6	Paging mit OFFSET und FETCH	381
5.7	Window-Funktionen	382
6	Gespeicherte Prozeduren, Funktionen und Trigger	385
6.1	Gespeicherte Prozeduren programmieren	386
6.1.1	Aufbau einer gespeicherten Prozedur	388
6.1.2	Erzeugen einer gespeicherten Prozedur	389
6.1.3	Einfache gespeicherte Prozeduren	399
6.1.4	Gespeicherte Prozeduren mit Eingabeparametern	402
6.1.5	Ergebnisrückgabe von Prozeduren	405
6.1.6	Cursor in gespeicherten Prozeduren nutzen	416
6.1.7	Transaktionen in Prozeduren	424
6.1.8	Table-Valued Parameter einsetzen	429
6.1.9	Systemintern kompilierte gespeicherte Prozeduren	434
6.1.10	Gespeicherte Prozeduren aus Client-Anwendungen heraus aufrufen	445
6.2	Mit Triggern automatisieren	460
6.2.1	DML-Trigger: Insert, Update, Delete	461
6.2.2	Triggerreihenfolge festlegen	481
6.2.3	INSTEAD OF-Trigger	483
6.2.4	Rekursive Trigger	486
6.2.5	Trigger löschen	498
6.2.6	Systemeigen kompilierte Trigger	499
6.2.7	DDL-Trigger	504
6.3	Benutzerdefinierte Funktionen implementieren	511
6.3.1	Skalarwertfunktionen	511
6.3.2	Inline-Funktionen	517
6.3.3	Tabellenwertfunktionen	519
6.3.4	Systemintern kompilierte benutzerdefinierte Funktionen	525

6.4	Debuggen	528
6.4.1	Voraussetzungen für das Debuggen	528
6.4.2	Debuggen einer gespeicherten Prozedur	530
6.4.3	Debuggen von Triggern	535
6.4.4	Debuggen von Funktionen	538
6.5	Praxistipps	539
6.5.1	Fehleranalyse mit ERROR_MESSAGE()	540
6.5.2	Fehler gezielt zur Ablaufsteuerung einsetzen	542
6.5.3	Fehlerprotokoll führen	545
6.5.4	Über Fehler benachrichtigen lassen	547
6.5.5	Automatisierte Importe mit BULK INSERT	551
7	SQL Server CLR-Integration	557
7.1	Mit im Boot: .NET Framework	558
7.1.1	Integration mit dem Visual Studio	560
7.2	CLR-Aktivierung	563
7.2.1	Code auf den Server bringen: Assembly	566
7.3	.NET User-Defined Functions	569
7.4	.NET Stored Procedures	577
7.4.1	Datenzugriff aus der CLR heraus	577
7.4.2	Prozeduren mit Werterückgabe	578
7.4.3	Zugriff auf externe Daten	585
7.5	.NET-Trigger	592
7.6	User-Defined Aggregates (UDA)	601
7.7	Externe Assemblys verwenden	608
7.8	CLR-Sicherheitseinstellungen	615
7.8.1	Assembly als vertrauenswürdig erklären	616
7.8.2	Assembly signieren	620
7.9	Verwalten des Servers mit SMO	628
7.10	Übrigens: Debuggen	634
7.10.1	Debuggen einer T-SQL Stored Procedure	635
7.10.2	Debuggen einer .NET-Stored Procedure	637
8	Data Tier Applications und SQL Server Data Tools	641
8.1	Datenebenenanwendungen	641
8.1.1	DAC über Management Studio erstellen	642
8.1.2	Eine DAC auf dem SQL Server bereitstellen	645
8.1.3	Aktualisieren einer DAC	647
8.1.4	Entfernen einer DAC	650
8.1.5	Von DACPAC zu BACPAC	650
8.1.6	Erstellen einer DAC mit dem Visual Studio	655
8.2	Die SQL Server Data Tools	656
8.2.1	Ein neues Datenbankprojekt erstellen	656

8.2.2	Datenbankobjekte erstellen	658
8.2.3	Datenbankprojekt bereitstellen	663
8.2.4	Schemavergleich	666
8.2.5	Datenbank in ein Datenbankprojekt importieren	671
8.2.6	Ersatz für das Management Studio?	674
9	Client-Server-Datenbank verwalten	677
9.1	Anfügen und Trennen von Datenbanken	677
9.1.1	Trennen einer Datenbank	678
9.1.2	Anfügen einer Datenbank	681
9.1.3	Option „Automatisch schließen“	687
9.2	Datenbank sichern	688
9.2.1	Sicherungsvarianten	688
9.2.2	Sicherungsziele	690
9.2.3	Sicherung mit dem Management Studio	693
9.2.4	Sicherung über TRANSACT-SQL	699
9.2.5	Zeitgesteuerte Sicherung mit dem SQL Server-Agent	703
9.2.6	Zeitgesteuerte Sicherung mit der Express Edition	707
9.2.7	Datenbank wiederherstellen	711
9.2.8	Einsatz der Zeitachse beim Wiederherstellen	715
9.2.9	Wiederherstellung über Transact-SQL	720
9.2.10	Desaster Recovery	721
9.2.11	Recovery mit FILESTREAM	728
9.3	Datenänderungen protokollieren	731
9.3.1	Change Data Capture	731
9.3.2	Temporale Tabellen	737
9.4	Mit mehreren Instanzen arbeiten	763
9.4.1	Standardinstanzen und benannte Instanzen	764
9.4.2	Zugriff auf Instanzen steuern	766
10	Sicherheit und Zugriffsberechtigungen	771
10.1	Authentifizierungsmodi - Anmeldungen und Benutzer	771
10.1.1	Windows-Authentifizierung	773
10.1.2	Gemischter Modus	774
10.1.3	Anmeldung und Benutzer	774
10.2	Berechtigungen	776
10.3	Rollen	777
10.3.1	Serverrollen	777
10.3.2	Datenbankrollen	780
10.3.3	Anwendungsrollen	781
10.4	Anmeldeinformationen (Credentials)	782
10.5	Schema	784
10.6	Verwaltung im Management Studio	786

10.6.1	Serveranmeldung hinzufügen	786
10.6.2	Schema anlegen	792
10.6.3	Datenbankbenutzer hinzufügen	792
10.6.4	Rollen in einer Datenbank anlegen	796
10.7	Berechtigungen vergeben	798
10.7.1	Berechtigungen auf Datenbankebene	798
10.7.2	Berechtigungen auf Serverebene	806
10.8	Lösungen mit T-SQL	807
10.8.1	Sicherheitsobjekte anlegen	808
10.8.2	Generische Skripte	814
10.9	Contained Databases	814
10.10	Administratorzugriff wiederherstellen	822
10.11	Indirekte Zugriffe verwalten	826
10.11.1	Datenzugriffe über Sichten	827
10.11.2	Sicherheit mit Prozeduren erhöhen	828
10.12	Sicherheit auf Zeilenebene	835
10.12.1	Bestandteile von Row Level Security (RLS)	836
10.12.2	Sicherheitsfunktion erstellen	839
10.12.3	Security Policy definieren	842
10.12.4	Ändern von Sicherheitsrichtlinien	845
10.13	Zugriff auf andere Server	848
10.13.1	SQL Server als Verbindungsserver	850
10.13.2	Verbindungsserver mit Fremdprodukten	857
10.14	Daten verschlüsseln mit Always Encrypted	861
10.14.1	Voraussetzungen für Always Encrypted	862
10.14.2	Konfiguration von Always Encrypted	863
10.14.3	Vorhandene Daten verschlüsseln	870
10.14.4	Abfragen von verschlüsselten Daten	873
10.14.5	Erstellen von Tabellen mit verschlüsselten Spalten	877
10.14.6	Einfügen von Daten mit Verschlüsselung	880
10.14.7	Treibereinsatz am Client	883
11	Erweiterte Funktionalitäten	887
11.1	Datenbank-E-Mail	887
11.1.1	Einrichten von Datenbank-E-Mail	888
11.1.2	E-Mails aus der Anwendung heraus versenden	895
11.1.3	Varianten des E-Mail-Versands	897
11.1.4	Konfiguration über Systemprozeduren	904
11.1.5	Mailbenachrichtigung für Agent-Aufträge	910
11.2	Integration Services	918
11.2.1	Datenabgleich mit IS	919
11.2.2	Pakete ausführen und auf den Server bringen	942
11.2.3	SSIS-Projekte auf den Server bringen	944

12	SQL Server 2017 auf Linux	951
12.1	Installation des SQL Servers	953
12.2	Kommandozeilentools installieren	958
12.2.1	SQLCmd mit ODBC	958
12.2.2	mssql-cli	962
12.3	Server-Agent ergänzen	966
12.4	Integration Services	967
12.5	Serverdienst starten	968
12.6	Updates installieren	969
12.7	Weitere Konfiguration	971
12.8	Windows-Authentifizierung	976
12.9	Linux auch am Client: SQL Operations Studio	984
A	Anhang	989
A.1	Die Tabellen der Datenbank WAWI	989
Index	1001