

Inhalt

ADO.Net-Grundlagen.....	1
1 ADO.Net-Grundlagen	2
1.1 Was ist ADO.Net?	2
1.2 Architektur von ADO.Net	3
1.3 Komponentenübersicht.....	5
1.3.1 .NET-Datenprovider	5
1.3.2 DataSet	7
1.3.3 Datenbindung.....	8
1.3.4 Steuerelemente.....	9
1.4 Zusammenspiel der ADO.Net-Komponenten.....	10
1.4.1 Beispielanwendung mit DataReader.....	10
1.4.2 Beispielanwendung mit DataAdapter	18
1.5 Zusammenfassung	25
Die Datenbanksprache SQL	27
2 Die Datenbanksprache SQL	28
2.1 Was ist SQL?.....	28
2.2 SQL im Einsatz	29
2.2.1 SQL-Editoren.....	29
2.2.2 Allgemeine SQL-Regeln.....	31
2.3 DQL – Datenabfragen mit SELECT-Anweisungen.....	33
2.3.1 Die Syntax der SELECT-Anweisung	33
2.3.2 Einfache SELECT-Anweisungen	35
2.3.3 Verknüpfen mehrerer Tabellen	37
2.3.4 Die WHERE-Bedingung.....	42
2.3.5 Einschränken der Ergebnismenge	47
2.3.6 Aggregatfunktionen	49
2.3.7 Gruppieren von Datensätzen.....	51
2.3.8 Arbeiten mit Unterabfragen	53
2.4 DML – Datenmanipulation.....	56
2.4.1 Eine neue Tabelle erstellen mit SELECT INTO	57
2.4.2 Datensätze einfügen mit INSERT	58
2.4.3 Datensätze ändern mit UPDATE	60
2.4.4 Datensätze löschen mit DELETE	62
2.5 DDL – Datendefinition.....	63
2.5.1 Generieren eines Änderungsskripts über Visual Studio.....	63

2.5.2	Skript über SQL Server Management Studio generieren.....	65
2.5.3	Verwenden von DDL im Code.....	66
2.6	Zusammenfassung.....	67
Visual Studio.NET-Überblick.....		69
3	Visual Studio.NET-Überblick	70
3.1	Einstelloptionen im Visual Studio.NET	70
3.2	Der Server-Explorer	72
3.2.1	Verbindung mit einer Datenbank herstellen	73
3.2.2	Arbeiten mit Tabellen.....	77
3.2.3	Arbeiten mit Ansichten.....	83
3.2.4	Gespeicherte Prozeduren	84
3.3	Das Datenquellenfenster.....	85
3.3.1	Welche Datenquellen kann man anlegen?	87
3.3.2	Arbeiten mit Datenquellen.....	88
3.4	Zusammenfassung.....	89
Visual Studio.NET und die Datenquellen		91
4	Visual Studio.NET und die Datenquellen.....	92
4.1	Erstellen der Beispielanwendung	92
4.2	Verwenden eines DataSets als Datenquelle.....	95
4.2.1	Datenquelle anlegen	96
4.2.2	DataSet in Formular einbinden.....	100
4.3	Verwenden einer Klasse als Datenquelle	105
4.3.1	Klasse implementieren	105
4.3.2	Klasse dem Datenquellenfenster hinzufügen.....	106
4.3.3	Klasse in Formular einbinden.....	108
4.3.4	Analyse des Ergebnisses.....	110
4.3.5	Anzeigen des Tabelleninhalts	111
4.3.6	Speichern und Laden von Objektdaten	113
4.4	Fazit.....	117
Die Connection-Klasse		119
5	Die Connection-Klasse	120
5.1	Das Beispielpogramm	121
5.2	Verbindung zur Datenbank herstellen	122
5.2.1	Verbindung zur SQL Server-Datenbank herstellen	123
5.2.2	Verbindung zu MS Access-Datenbank herstellen.....	125
5.3	Eigenschaften	126
5.3.1	DieConnectionString-Eigenschaft	126
5.3.2	Die ConnectionTimeout-Eigenschaft.....	127
5.3.3	Die State-Eigenschaft	128

5.3.4	Weitere Eigenschaften	128
5.4	Methoden.....	131
5.4.1	Die Open-Methode	132
5.4.2	Die Close-Methode.....	132
5.4.3	Die BeginTransaction-Methode.....	132
5.4.4	Die CreateCommand-Methode	133
5.4.5	Weitere Methoden	134
5.5	Verbindungspooling.....	136
5.6	Metadaten.....	138
5.6.1	Metadatenauflistungen.....	138
5.6.2	Metadatenauflistung laden.....	139
5.6.3	Metadaten zu den Metadaten	140
5.7	Zusammenfassung.....	140
Die Command-Klasse		143
6	Die Command-Klasse	144
6.1	Objektmodell des Commands.....	144
6.2	Das Beispielprogramm	145
6.3	Erzeugen des Command-Objekts	147
6.4	Command ausführen.....	149
6.5	Parameter.....	153
6.5.1	Parameter im SqlCommand	153
6.5.2	Parameter im OleDbCommand-Objekt.....	153
6.5.3	Direction-Eigenschaft der Parameter	154
6.5.4	Parameter bei Verwendung des DbCommand	157
6.5.5	Parameter durch String-Variablen ersetzen.....	159
6.6	Eigenschaften	160
6.6.1	CommandType-Eigenschaft	160
6.6.2	CommandText-Eigenschaft	161
6.6.3	Weitere Eigenschaften	161
6.7	Methoden zur Ausführung des Commands.....	163
6.7.1	ExecuteScalar-Methode	163
6.7.2	ExecuteNonQuery-Methode	164
6.7.3	ExecuteReader-Methode.....	165
6.7.4	ExecuteXMLReader-Methode.....	166
6.8	Asynchrone Ausführung des SqlCommands	167
6.8.1	Der „async“-Parameter im ConnectionString	168
6.8.2	Rückmeldung.....	168
6.8.3	Anwendungsbeispiel.....	169
6.9	Weitere Methoden des Commands	171
6.10	Anwendungsbeispiele für die Command-Klasse	173
6.10.1	Das Formular in der Beispielanwendung.....	173
6.10.2	Neuen Datensatz anlegen.....	175
6.10.3	Datenänderungen speichern	176
6.10.4	Datensatz löschen	177

6.11	Zusammenfassung	180
Die DataReader-Klasse		181
7	Die DataReader-Klasse	182
7.1	Verwendung	182
7.2	Objektmodell des DataReaders	184
7.3	Das Beispielprogramm	184
7.4	Erzeugen des DataReaders	186
7.5	Verwendung des DataReaders	187
7.6	Eigenschaften	188
7.6.1	Item-Eigenschaft bzw. Indexer	188
7.6.2	Weitere Eigenschaften	189
7.7	Methoden	191
7.7.1	Die Read-Methode	191
7.7.2	Die NextResult-Methode	192
7.7.3	Get-Methoden zum Auslesen der Spaltenwerte	194
7.7.4	Methoden zum Auslesen von Metadaten	196
7.7.5	Die GetSchemaTable-Methode	198
7.7.6	Weitere Methoden	200
7.8	Zusammenfassung	201
Die DataAdapter-Klasse		203
8	Die DataAdapter-Klasse	204
8.1	Lokaler Datenspeicher und DataAdapter	204
8.2	DataAdapter innerhalb des .Net Data Providers	205
8.3	Objektmodell des DataAdapters	208
8.4	Das Beispielprogramm	209
8.5	Erzeugen des DataAdapters	211
8.5.1	Erzeugen des Adapters über den Konstruktor	212
8.5.2	DbDataFactory und DbDataAdapter	213
8.6	Verwendung des DataAdapter-Objekts	215
8.6.1	DataAdapter und Commands anlegen	216
8.6.2	Daten laden und anzeigen	218
8.6.3	Änderungen in Datenbank zurückspeichern	219
8.7	CommandBuilder und DataAdapter	220
8.7.1	CommandBuilder verwenden	220
8.7.2	ConflictOption-Eigenschaft des CommandBuilders	222
8.8	Eigenschaften des DataAdapters	225
8.8.1	AcceptChanges...-Eigenschaften	225
8.8.2	Command-Eigenschaften	226
8.8.3	ContinueUpdateOnError-Eigenschaft	226
8.8.4	FillLoadOption-Eigenschaft	228
8.8.5	MissingSchemaAction-Eigenschaft	229
8.8.6	ReturnProviderSpecificTypes-Eigenschaft	230

8.8.7	TableMappings-Eigenschaft	230
8.8.8	MissingMappingAction-Eigenschaft	232
8.8.9	UpdateBatchSize-Eigenschaft.....	234
8.9	Methoden des DataAdapters.....	234
8.9.1	Die Fill-Methode	234
8.9.2	Die FillSchema-Methode	237
8.9.3	GetFillParameters-Methode.....	238
8.9.4	Die Update-Methode.....	238
8.10	Ereignisse	242
8.10.1	RowUpdating-Ereignis	242
8.10.2	RowUpdated-Ereignis.....	245
8.10.3	FillError-Ereignis.....	246
8.11	Zusammenfassung	246
Die Transaction-Klasse		247
9	Die Transaction-Klasse.....	248
9.1	Was ist eine Transaktion?.....	248
9.2	Objektmodell der Transaction	250
9.3	Das Beispielprogramm	251
9.4	Erzeugen des Transaction-Objekts	253
9.5	Verwendung der Transaction-Klasse.....	253
9.5.1	Verwendung mit Commands	253
9.5.2	Verwendung mit einem DataAdapter-Objekt	256
9.6	Eigenschaften	259
9.6.1	Die Connection-Eigenschaft	259
9.6.2	Die IsolationLevel-Eigenschaft.....	259
9.7	Methoden.....	263
9.7.1	Die Commit-Methode	263
9.7.2	Die Rollback-Methode.....	264
9.7.3	Weitere Methoden	264
9.8	Zusammenfassung	265
Die SqlBulkCopy-Klasse		267
10	Die SqlBulkCopy-Klasse.....	268
10.1	Prinzip der SqlBulkCopy-Klasse.....	268
10.2	Erzeugen des SqlBulkCopy-Objekts	269
10.3	Die SqlBulkCopyOptions	270
10.4	Eigenschaften	271
10.4.1	BatchSize-Eigenschaft	271
10.4.2	BulkCopyTimeOut-Eigenschaft.....	271
10.4.3	ColumnMappings-Eigenschaft	272
10.4.4	DestinationTableName-Eigenschaft	272
10.4.5	NotifyAfter-Eigenschaft	273
10.5	WriteToServer-Methode.....	274

10.6	Das Beispielprogramm.....	275
10.6.1	Benutzeroberfläche.....	275
10.6.2	Programmaufbau	277
10.7	Verwendung der SqlBulkCopy-Klasse.....	279
10.7.1	Import einer DataTable ohne Transaktion	279
10.7.2	Import eines DataRow-Arrays mit interner Transaktion.....	281
10.7.3	Import eines DataReaders mit externer Transaktion.....	283
10.7.4	Vergleich mit dem DataAdapter	285
10.8	Zusammenfassung	286
Die DataTable-Klasse		287
11	Die DataTable-Klasse.....	288
11.1	Objektmodell der DataTable	288
11.2	Das Beispielprogramm.....	290
11.3	Erzeugen eines DataTable-Objekts	292
11.4	Verwendung des DataTable-Objekts.....	292
11.5	Rows-Auflistung	294
11.5.1	Verwendung der Rows-Auflistung	295
11.5.2	Die Count-Eigenschaft	296
11.5.3	Die Add-Methode	296
11.5.4	Die Contains-Methode.....	298
11.5.5	Die Find-Methode	299
11.5.6	Die Remove-Methode.....	300
11.5.7	Die Clear-Methode	302
11.6	Columns-Auflistung.....	302
11.6.1	Die Contains-Methode.....	303
11.6.2	Die Add- und Remove-Methode.....	303
11.6.3	Die Clear-Methode	305
11.6.4	Die IndexOf-Methode.....	305
11.7	Eigenschaften der DataTable.....	306
11.7.1	Auflistungen	306
11.7.2	Die PrimaryKey-Eigenschaft.....	308
11.7.3	DataSet	309
11.7.4	Die DefaultView-Eigenschaft.....	310
11.7.5	Weitere Eigenschaften.....	311
11.8	Methoden der DataTable	314
11.8.1	NewRow-Methode	314
11.8.2	ImportRow-Methode	315
11.8.3	LoadDataRow-Methode	315
11.8.4	ReadXml-, WriteXml-Methode.....	316
11.8.5	Accept- und RejectChanges-Methode	319
11.8.6	Compute-Methode.....	319
11.8.7	Select-Methode.....	321
11.8.8	Weitere Methoden	322
11.9	Ereignisse der DataTable.....	325

11.9.1	Das RowChanging-Ereignis.....	326
11.9.2	Weitere Ereignisse	327
11.10	Anwendungsfälle für die DataTable	328
11.10.1	Kalkulierte Spalten erzeugen	328
11.10.2	AutoIncrement-Spalten erzeugen.....	332
11.10.3	Datenzeilen auslesen.....	332
11.10.4	Spaltenwerte auslesen	334
11.10.5	DataTable an Steuerelement in Formularen binden	335
11.11	Zusammenfassung	341
Die DataRow-Klasse		343
12	Die DataRow-Klasse.....	344
12.1	Objektmodell der DataRow	344
12.2	Das Beispielprogramm	346
12.3	Erzeugen eines DataRow-Objekts	347
12.4	Spaltenwerte auslesen und setzen	348
12.5	Versionen eines DataRow-Objekts.....	350
12.6	Eigenschaften	353
12.6.1	Die RowState-Eigenschaft	353
12.6.2	Weitere Eigenschaften	354
12.7	Methoden.....	356
12.7.1	Die AcceptChanges- und RejectChanges-Methode	356
12.7.2	Die BeginEdit-, CancelEdit-, EndEdit-Methode.....	357
12.7.3	Die HasVersion-Methode	358
12.7.4	Die GetChildRows- und GetParentRow-Methode.....	359
12.7.5	Weitere Methoden	361
12.8	Versionen und die DataTable-Ereignisse.....	365
12.8.1	ColumnChanged-Ereignis.....	365
12.8.2	Weitere Ereignisse	371
12.9	Zusammenfassung	372
Die DataView-Klasse		373
13	Die DataView-Klasse	374
13.1	Objektmodell der DataView	374
13.2	Das Beispielprogramm	375
13.3	Erzeugen eines DataView-Objekts	377
13.4	Daten lesen und schreiben	377
13.5	Eigenschaften	379
13.5.1	Die AllowDelete-, AllowEdit- und AllowNew-Eigenschaft	379
13.5.2	Die RowFilter-Eigenschaft	382
13.5.3	Die RowStateFilter-Eigenschaft	383
13.5.4	Die Sort-Eigenschaft.....	385
13.5.5	Weitere Eigenschaften	386
13.6	Methoden.....	388

13.6.1	Die AddNew-Methode	388
13.6.2	Die Find- und FindRows-Methode	390
13.6.3	Delete-Methode	391
13.6.4	ToTable-Methode	392
13.7	ListChanged-Ereignis	393
13.8	Die DataRowView-Klasse	394
13.8.1	Objektmodell der DataRowView	394
13.8.2	Eigenschaften der DataRowView	394
13.8.3	Methoden der DataRowView	396
13.9	Datenanbindung des DataView-Objekts	397
13.10	Datenanbindung des DataRowView-Objekts	401
13.11	Zusammenfassung	402
Die DataSet-Klasse		403
14	Die DataSet-Klasse	404
14.1	Objektmodell der DataSet-Klasse	405
14.2	Das Beispielprogramm	406
14.3	Erzeugen eines DataSet-Objekts	408
14.4	Eigenschaften des DataSets	409
14.5	Methoden des DataSets	412
14.6	Tabellen hinzufügen	417
14.6.1	Add-Methode der Tables-Auflistung	418
14.6.2	Fill-Methode des DataAdapters	419
14.7	Relationen	419
14.7.1	Was sind Relationen?	420
14.7.2	Objektmodell der DataRelation-Klasse	421
14.7.3	Eigenschaften der DataRelation-Klasse	423
14.7.4	Relationen erzeugen	424
14.7.5	Über Relationen auf Daten zugreifen	428
14.7.6	Objektmodell der ForeignKeyConstraint-Klasse	430
14.7.7	Eigenschaften der ForeignKeyConstraint-Klasse	430
14.8	DataSet mit Daten füllen	432
14.8.1	Über DataAdapter	432
14.8.2	ReadXml-Methode	433
14.9	Daten speichern	435
14.9.1	Über DataAdapter	435
14.9.2	WriteXml-Methode	436
14.10	Das abgeleitete Schema	438
14.10.1	Die Nested-Eigenschaft der DataRelation-Klasse	439
14.10.2	Das Schema	443
14.11	Serialisierung im Binärformat	445
14.12	Zusammenfassung	448

Datenbindung.....	451
15 Datenbindung	452
15.1 Arten der Datenbindung	452
15.2 Klassen der Datenanbindung	453
15.3 Die Synchronisierung	456
15.4 Die BindingSource Klasse.....	459
15.4.1 BindingSource erzeugen	460
15.4.2 Datenbindungen anlegen.....	460
15.4.3 Die DataSource- und List-Eigenschaft.....	463
15.4.4 Die Position-Eigenschaft	464
15.4.5 Die Current-Eigenschaft	465
15.4.6 Weitere Eigenschaften	466
15.4.7 Methoden der BindingSource	467
15.4.8 Ereignisse der BindingSource	469
15.5 Die Binding-Klasse	472
15.5.1 Anlegen der Bindung	472
15.5.2 Eigenschaften der Binding-Klasse	473
15.5.3 Die ReadValue- und WriteValue-Methode.....	478
15.5.4 Das Format- und Parse-Ereignis	478
15.6 Datenquellen.....	479
15.7 Das Beispielprogramm	484
15.8 Standardprobleme.....	486
15.8.1 Angezeigte Daten und Funktionen.....	487
15.8.2 Entwurf des lokalen Datenspeichers	490
15.8.3 Lokale Datenbank erzeugen und füllen.....	492
15.8.4 Datenbindung.....	496
15.8.5 Datennavigation und Synchronisation	503
15.8.6 Filtern der lokalen Daten	504
15.8.7 Vorgabewerte setzen.....	507
15.8.8 Änderungen verwerfen	509
15.8.9 Änderungen speichern	514
15.9 Zusammenfassung	519
Typisiertes DataSet.....	521
16 Typisiertes DataSet	522
16.1 Was ist ein typisiertes DataSet?.....	522
16.1.1 Logischer Aufbau	522
16.1.2 Typisierte Klassen	523
16.1.3 Vor- und Nachteile des typisierten DataSets.....	525
16.2 Erweiterungen der DataSet-Klassen	528
16.2.1 Erweiterung der DataSet-Klasse	528
16.2.2 Erweiterungen der DataTable-Klasse	528
16.2.3 Erweiterungen der DataRow-Klasse	530
16.3 Der TableAdapter	531

16.4	Das Beispielprogramm	534
16.5	Anlegen des typisierten DataSets	536
16.5.1	Typisiertes DataSet über Datenquellen-Fenster anlegen	537
16.5.2	Typisiertes DataSet direkt dem Projekt hinzufügen.....	539
16.5.3	Dateien des typisierten DataSets.....	540
16.6	Entwerfen des DataSets mit dem Editor.....	541
16.7	Modifizieren der TableAdapter	545
16.7.1	Der Abfrage-Generator.....	545
16.7.2	Konfigurieren des TableAdapters	547
16.8	Verwenden des typisierten DataSets im Formular	549
16.8.1	Instanzen der Objekte für die Bindung anlegen	550
16.8.2	Binden der BindingSources an die Datenquelle.....	551
16.8.3	Fehler im Formulardesigner	553
16.8.4	TableAdapter hinzufügen und Daten laden	554
16.8.5	Steuerelemente an BindingSources binden.....	556
16.9	Nachträgliche Änderungen am DataSet.....	561
16.9.1	Änderungen an der Datenbank	561
16.9.2	Änderungen am typisierten DataSet	563
16.10	Ein Blick auf den Programmcode.....	564
16.11	Fazit.....	567
XML und XML-Schemata.....		569
17	XML und XML-Schemata.....	570
17.1	Das Beispielformular.....	570
17.2	XML-Grundlagen.....	572
17.3	XML-Syntax	575
17.4	XML-Elemente und Typen	578
17.4.1	XML-Deklaration	578
17.4.2	Element.....	578
17.4.3	Attribute	579
17.4.4	Typen.....	581
17.4.5	Benannte und unbenannte Typen.....	582
17.4.6	Namespace.....	585
17.5	DataSet verwendet XML-Daten	587
17.5.1	Der Schema-Editor	587
17.5.2	Anlegen des Schemas über den DataSet-Editor.....	588
17.5.3	Laden des Schemas und der XML-Datei	591
17.5.4	Benutzereinstellungen in XML-Datei speichern.....	593
17.5.5	Lesen und schreiben der Benutzereinstellungen	594
17.5.6	Komplexe Benutzereinstellung.....	596
17.6	XML-Daten verwenden ein DataSet	600
17.7	Datenaustausch über DiffGrams.....	607
17.7.1	DiffGram-Format	608
17.7.2	DataSet als DiffGram speichern	612
17.7.3	DataSet von DiffGram füllen.....	613

17.8	SQL Server und FOR XML-Abfragen	614
17.8.1	Anzeigen der XML-Ausgabe	614
17.8.2	FOR XML RAW	616
17.8.3	FOR XML AUTO	617
17.8.4	FOR XML EXPLICIT	622
17.8.5	FOR XML PATH	625
17.8.6	Verwendung von FOR XML mit XMLReader	629
17.8.7	Verwendung von FOR XML mit DataSets	632
17.9	Zusammenfassung	636
Die Webdienste		637
18	Die Webdienste	638
18.1	Prinzip des Webdienstes	638
18.2	Die Infrastruktur für Webdienste	641
18.2.1	Webdienstverzeichnisse	641
18.2.2	Homepage des Dienstanbieters	643
18.2.3	Web Service Description Language	644
18.3	Webdienst erstellen	644
18.3.1	Neues Projekt anlegen	644
18.3.2	Funktionalität implementieren	647
18.3.3	Testen des Webdienstes	653
18.4	Webdienstclient erstellen	657
18.4.1	Projekt erstellen	657
18.4.2	Funktionalität implementieren	660
18.4.3	Testen des Webdienstclients	664
18.5	Asynchroner Methodenaufruf	665
18.5.1	Grundlagen	666
18.5.2	Der asynchrone Methodenaufruf	666
18.6	Datenübertragung im Binärformat	670
18.7	Zusammenfassung	674
ASP-Grundlagen		675
19	ASP-Grundlagen	676
19.1	Einführung in ASP.NET	676
19.2	Installation von ASP.NET	679
19.2.1	Software-Anforderungen	680
19.2.2	Installation des IIS	680
19.2.3	ASP.NET dem IIS bekannt machen	686
19.2.4	Beispielanwendungen	690
19.3	Die erste ASP.NET-Anwendung	693
19.3.1	Erstellen einer neuen Webanwendung	694
19.3.2	Hinzufügen der Controls	698
19.3.3	Dokumentgliederung	703
19.3.4	Hinzufügen des Codes für die Buttons	705

19.4 Der Namespace System.Web706

19.5 Das GridView-Control709

19.5.1 Erstellen der Anwendung709

19.5.2 Hinzufügen einer Datenquelle711

19.5.3 Anbinden des GridViews an eine Access-Datenbank.....712

19.5.4 Blättern, Sortieren und Layout715

19.5.5 Ändern von Datensätzen.....718

19.6 Zusammenfassung.....721

Sicherheit von Webapplikationen..... 723

20 Sicherheit von Webapplikationen 724

20.1 Grundbegriffe724

20.2 Ein einfaches Beispiel728

20.2.1 Die fertige Anwendung728

20.2.2 Erstellen einer neuen Anwendung729

20.2.3 Erstellen der Benutzeroberfläche.....729

20.2.4 Der Code732

20.3 Wahl der Sicherheitsstrategie736

20.3.1 Änderung der ASPNET-Berechtigungen.....737

20.3.2 Änderung der ACLs.....739

20.3.3 Impersonation742

20.3.4 Zusammenfassung747

ASP.NET und ADO.NET 749

21 ASP.NET und ADO.NET 750

21.1 Die fertige Anwendung750

21.2 Erstellen der Reservierungsseite.....752

21.2.1 Vorbereitende Maßnahmen752

21.2.2 Hinzufügen der Controls753

21.2.3 Programmcode.....756

21.3 Erstellen der Ergebnisseite760

21.3.1 Hinzufügen einer neuen Web Form760

21.3.2 Erstellen der Benutzeroberfläche.....761

21.3.3 Programmatischer Wechsel zur Ergebnisseite763

21.3.4 Code zur Durchführung einer Reservierung764

21.3.5 Code zur Ausgabe der Belegungsdaten767

21.4 Zusammenfassung.....772

Index 773