

Arbeitsaufgaben für das Buch „Keine Angst vor Microsoft Access!“

INHALT

1	Einleitung	2
2	“Herumspielen” mit einer Datenbank	3
2.1	Arbeitsaufgaben	3
2.2	Kontrollfragen	3
2.3	BeMERKENswertes (S. 21-37)	4
3	Wie funktioniert das?	5
3.1	Arbeitsaufgaben	5
3.2	Kontrollfragen	5
3.3	BeMERKENswertes (S. 42-66)	6
4	Datenmodellierung – erst mal ganz einfach.....	7
4.1	Arbeitsaufgaben	7
4.2	Kontrollfragen	7
4.3	BeMERKENswertes (S. 72-90)	8
5	Datenmodellierung – Vertiefung	9
5.1	Arbeitsaufgaben	9
5.2	Kontrollfragen	9
5.3	BeMERKENswertes (S. 83-116 und S. 125/126)	10
6	Tabellen, Beziehungen, Daten.....	11
6.1	Arbeitsaufgaben	11
6.2	Kontrollfragen	11
6.3	BeMERKENswertes (S. 128-138)	12
7	Ein Stammdatenformular	13
7.1	Arbeitsaufgaben	13
7.2	Kontrollfragen	13
7.3	BeMERKENswertes (S. 139-149)	14
8	Ein Formular mit Unterformular	15
8.1	Arbeitsaufgaben	15
8.2	Kontrollfragen	15
8.3	BeMERKENswertes (S. 149-155)	16
9	“Einzelzuordnung”	17
9.1	Arbeitsaufgaben	17
9.2	Kontrollfragen	17
9.3	BeMERKENswertes (S. 155-160)	17
10	Das Startformular	18
10.1	Arbeitsaufgaben	18
10.2	Kontrollfragen	18
10.3	BeMERKENswertes (S. 160-161)	18
11	Wie geht’s weiter?	19

1 Einleitung

Ich habe alles, was ich zum Thema "Datenbankentwicklung mit MS Access" sagen kann, in meinem Buch geschrieben. Wer sich also gründlich mit der Thematik beschäftigen will (oder muss ;-), der sollte das Buch von vorne bis hinten durcharbeiten. DurchARBEITEN! Nicht durchLESEN!

Für alle anderen, die nicht so viel Zeit haben – z.B. Studenten, die auch noch andere Vorlesungen besuchen müssen – ist dieses Dokument gedacht. Es führt Sie anhand von Arbeitsaufgaben durch die wichtigsten ersten Abschnitte des Buches.

Ich gehe dabei in einer Schrittfolge vor, die sich in der Lehre immer wieder bewährt hat und nach der auch das Buch aufgebaut ist:

1. Ich lasse Sie zunächst mit einem fertigen Produkt (in diesem Fall mit einer Access-Datenbank) „herumspielen“, damit Sie wissen, wo es hin gehen soll.
2. Dann werfen wir einen Blick hinter die Kulissen und entdecken gemeinsam, warum das so funktioniert, wie es funktioniert.
3. Als nächstes lernen Sie, selber ein einfaches Produkt (in diesem Fall eine Access-Datenbank) herzustellen.
4. Dann zeige ich Ihnen ein paar Tricks und Kniffe der Profis und sage Ihnen abschließend, in welcher Richtung Sie selber weiter lernen können.

ACHTUNG: Sie sollten die hier angegebenen Seiten und Kapitel wirklich durchARBEITEN – nicht nur durchLESEN! Sämtliche Erläuterungen im Buch erfolgen anhand von Beispieldatenbanken. Sie sollten also beim Arbeiten mit dem Buch unbedingt Ihren Rechner laufen haben und alle Beispiele dort unmittelbar beim Lesen nachvollziehen!

Die Seitenangaben und Kapitelnummern beziehen sich auf die vierte Auflage des Buches. Wenn Sie eine andere Auflage haben, müssen Sie ein wenig hin- und her blättern ;-)

Die Unterkapitel haben im Buch leider keine Nummer. Eigentlich müsste es z.B. heißen „1.1. Formulare benutzen“ und „1.1.1. Das Formular frmMitglieder“. Innerhalb des Buches sind die Unterkapitel aber nur an der etwas kleineren Schriftart zu erkennen. Ich bitte Sie daher, sich am Inhaltsverzeichnis zu orientieren. Dort sind die Unterkapitel deutlicher an der Einrückung zu erkennen.

Wenn ich im Folgenden schreibe „Bitte arbeiten Sie im 1. Kapitel das Unterkapitel *Formulare benutzen* durch“, dann meine ich dass Sie den Text unmittelbar unter der Überschrift „Formulare benutzen“ des 1. Kapitels durcharbeiten sollen – nicht die Unterkapitel des Unterkapitels. Wenn Sie diese auch durcharbeiten sollen, dann weise ich Sie ausdrücklich darauf hin!

Dieses Dokument hat – wie alle meine Dokumente – eine Versionsnummer. Bevor Sie es durcharbeiten, sehen Sie also bitte unter

<http://www.andreasstern.de/buch/download.php>

nach, ob es vielleicht schon eine neuere Version gibt.

2 “Herumspielen” mit einer Datenbank

2.1 Arbeitsaufgaben

Als Erstes sollen Sie einmal mit einer fertigen Datenbank „herumspielen“, um ein Gefühl dafür zu bekommen, worum es geht; und auch um zu sehen, wo wir am Ende hin wollen.

Bitte arbeiten Sie im 1. Kapitel die folgenden Unterkapitel durch:

- Technische Voraussetzungen (S. 20)
- Formulare benutzen (S. 27)
 - Das Formular frmMitglieder (S. 29)
- Benutzerfreundlichkeit (S. 41)

2.2 Kontrollfragen

Nach der Erledigung der Arbeitsaufgaben sollten Sie in der Lage sein, folgende Fragen zu beantworten:

- a) Was sind Programm-Bibliotheken?
- b) Wie stellt man die Verweise auf Bibliotheken richtig ein?
- c) Was enthält ein Startformular und was kann man damit machen?
- d) Was ist der Unterschied zwischen „Formular“ und „Bericht“?
- e) Welche Steuerelemente gibt es auf einem Formular und was kann man damit machen?
- f) Was ist beim Löschen zu beachten?
- g) Welche Möglichkeiten gibt es, das Löschen zu programmieren?
- h) Was ist eine Standardschaltfläche?
- i) Wie kann ein Formular ohne Maus bedient werden?
- j) Welche Grundsätze gibt es bei der Formulargestaltung?
- k) Wie kann man Fehlbedienungen im Formular verhindern?
- l) Wann verhält sich ein Formular erwartungskonform?
- m) Welche Schaltflächen hat ein Formular üblicherweise?
- n) Was kann man tun, um ein Formular benutzerfreundlich zu gestalten?

2.3 BeMERKENswertes (S. 21-37)

1. Verweise einstellen
1. DB Verein
2. frmStart: Buttons öffnen Formulare
3. frmMitglieder:
 - Listenfeld
 - Unterformulare
 - Kombinationsfelder
 - neue Mitgliedschaft zuweisen
 - speichern
 - löschen, Warnung, Standardschaltfläche
 - neuen Beitragssatz eingeben
 - Trainer bzw. Mannschaft zuweisen mit Kombinationsfeld
 - Trainer bzw. Mannschaft wieder löschen
4. Vergleich mit frmMannschaften: gleiches Layout
5. frmMitglieder: von Feld zu Feld mit Tabs
6. Merkmale guter Software
7. Layout-Überlegungen
8. frmMitglieder:
 - Neu / Speichern / Löschen („erwartungskonform“)
 - Muss-Daten
 - Löschproblematik
9. Buttons: Hilfe / Startformular / Schließen
10. Übung: Probieren Sie andere Formulare aus!

3 Wie funktioniert das?

3.1 Arbeitsaufgaben

Jetzt wollen wir einen Blick hinter die Kulissen werfen und herauskriegen, warum das alles so funktioniert.

Bitte arbeiten Sie im 1. Kapitel die folgenden Unterkapitel durch:

- Tabellen, Felder und Primärschlüssel (S. 42)
- Beziehungen und Fremdschlüssel (S. 50)
- Formulare entwerfen (S. 55)
 - Eigenschaften von Formularelementen (S. 55)
 - Datensatzquelle (S. 57)
 - Unterformulare (S. 59)
 - Datensatzherkunft und Steuerelementinhalt (Textfelder, Kombinationsfelder, Listenfelder) (S. 62)

3.2 Kontrollfragen

Nach der Erledigung der Arbeitsaufgaben sollten Sie in der Lage sein, folgende Fragen zu beantworten:

- a) Was ist ein Feld? Was ist ein Primärschlüssel? Was ist ein Fremdschlüssel?
- b) Warum kann der Nachname einer Person nicht Primärschlüssel sein?
- c) Was macht man mit dem Navigationsbereich?
- d) Wie wird ein Primärschlüssel in Access definiert?
- e) Wie werden die Namen von Tabellen, Feldern, Primär- und Fremdschlüsseln gebildet?
- f) Worin besteht der Unterschied zwischen dem Primärschlüssel mit_id und der Mitgliedsnummer mit_nummer?
- g) Sind Primärschlüssel in einer Tabellenspalte, in einer Tabelle oder in einer Datenbank eindeutig?
- h) Was sehen Sie in der Entwurfsansicht einer Tabelle?
- i) Welche Felddatentypen gibt es?
- j) Was wird durch die Festlegung der Felddatentypen verhindert?
- k) Was ist das Besondere an dem Felddatentyp Autowert?
- l) Was ist ein Nachschlagefeld? Wie wird es definiert?
- m) Wie werden Beziehungen in Access dargestellt?
- n) Was bedeutet „1“ und „∞“ in einer Beziehung? Wie liest man eine Beziehung?
- o) Was ist eine „Zwischentabelle“? Welche Spalten enthält sie?
- p) Was versteht man unter der „Datensatzquelle“ eines Formulars?
- q) Was versteht man unter „Datensatzherkunft“ und „Steuerelementinhalt“ bei Text-, Listen- und Kombinationsfeldern?
- r) Wie hängt ein Unterformular mit dem übergeordneten Formular zusammen?
- s) Was bezeichnet die „Gebundene Spalte“ eines Listen- bzw. Kombinationsfeldes?
- t) Was sehen Sie auf der Registerkarte „Ereignis“ des Eigenschaftenblattes?

3.3 BeMERKENswertes (S. 42-66)

1. DB Verein: Navigationsbereich nach Objekttyp ordnen
2. Tabelle tblMitglied:
 - Primärschlüssel „_id“, Fremdschlüssel „_id_f“
 - Definition Primärschlüssel
 - Namensschema: „tbl“, „man_“, „_id“, „_id_f“
 - „_id“ <-> „nummer“
 - fehlende Primärschlüssel durch Löschen
3. Entwurfsansicht der Tabelle:
 - „Felder“
 - Datentyp
 - Feldgröße, Standardwert, Eingabeerforderlich
 - Autowert
 - Nachschlagefelder
4. tblMannschaft, tblTrainer: Definition „Fremdschlüssel“
5. „Beziehungsgeflecht“
 - lesen der Beziehungen
 - Zwischentabellen tblMit_Training, tblMit_Wet, ...
 - hervorgegangen aus m:n-Beziehungen
6. frmMannschaften: Entwurfsansicht
7. Rechtsklick / Eigenschaften, rechts andocken
8. Datensatzquelle
 - des Formulars
 - Schaltfläche mit den drei Punkten
 - Bedienung des Abfrage-Generators (Drag&Drop der Spalten, Tabelle hinzufügen)
 - Vorgehensweise komplett
9. frmMannschaften: Unterformular „Mitglieder“
 - Datensatzquelle
 - Bezug zum übergeordneten Formular?
 - Formular-Eigenschaften / Unterformular-Eigenschaften
 - Verknüpfen von (Fremdschlüssel) / Verknüpfen nach (Primärschlüssel)
10. frmMannschaften: Datensatzherkunft und Steuerelementinhalt
 - Textfelder: Daten / Steuerelementinhalt
 - Kombinationsfelder: tblTrainer.trainer_id <-> tblMannschaft.trainer_id_f, Spaltenbreiten, Gebundene Spalte
 - Listenfelder: „Ungebunden“ = kein Steuerelementinhalt, stattdessen: Ereignis / Nach Aktualisierung
11. Übung: Sehen Sie sich die Datensatzquelle, die Datensatzherkunft und den Steuerelementinhalt in den Formularen „Einzelwettkämpfe“ (frmEWettkampfe) und „Mannschaftswettkämpfe“ (frmMWettkampfe) an und versuchen Sie darauf aufbauend zu verstehen, wie die Formulare genau funktionieren!

4 Datenmodellierung – erst mal ganz einfach

4.1 Arbeitsaufgaben

Nachdem ich Sie jetzt zu der Erkenntnis geführt habe, dass wir um eine saubere Datenmodellierung nicht herumkommen, will ich Ihnen zeigen, wie man das macht.

Bitte arbeiten Sie im 2. Kapitel die folgenden Unterkapitel durch:

- Ein Unternehmensdatenmodell (S. 72)
- ... und seine Erstellung (S. 77)
 - Überall Bemerkungen (S. 78)
 - Preise – wofür? (S. 79)
 - Kunden ohne Aufträge? (S.80)
 - Kein Material und keine Leute? (S.80)
 - Keine Fremdschlüssel? (S. 81)
- Die strukturierte Szenariobeschreibung (S. 88)
- Die Überführung in das physische Modell (S. 90)

4.2 Kontrollfragen

Nach der Erledigung der Arbeitsaufgaben sollten Sie in der Lage sein, folgende Fragen zu beantworten:

- a) Wie werden Objekte und Beziehungen im logischen Modell dargestellt?
- b) Wie liest man ein logisches Modell?
- c) Welche Kardinalitäten gibt es?
- d) Welche ganz einfache Regel gilt für die Erstellung des logischen Modells?
- e) Was ist der Unterschied zwischen „Entität“ (oder „Objekttyp“) und „Tabelle“?
- f) Aus was für Sätzen sollte eine strukturierte Szenariobeschreibung bestehen?
- g) Warum sollten Kardinalitäten i.a. vom Typ (0,1) oder (0,n) statt (1,1) oder (1,n) sein?
- h) Warum enthält das logische Modell keine Fremdschlüssel?
- i) Zeigt uns das logische Modell die späteren Tabellen?
- j) Nach welchen beiden Regeln erfolgt die Überführung des logischen Modells ins physische Modell?
- k) Was bedeutet die Aussage „Das logische Modell ist datenbankunabhängig – das physische nicht.“?
- l) Was bedeuten die Pfeile im physischen Modell?
- m) Warum kann das physische Modell mehr „Rechtecke“ enthalten als das logische Modell?

4.3 BeMERKENswertes (S. 72-90)

1. DB Firma: Beispiel und Symbolik
2. Lesen eines Datenmodells
3. Kardinalitäten
4. verbale Beschreibung: Substantive / Verben -> Entitäten / Relationen
5. strukturierte Szenariobeschreibung
6. überall Bemerkungen
7. Preis / Preiseinheit
8. Kunden ohne Aufträge? Kein Material und keine Leute?
9. logisches und physisches Modell, „Entität“ / „Tabelle“
10. zwei Regeln für die Überführung logisches -> physisches Modell
11. Übung: Sehen Sie sich das logische Datenmodell für die Datenbank „Firma“ (S. 87) bzw. „Verein“ (S. 93) bzw. „Verleih“ (S. 98) an und sprechen Sie die Sätze für die Beziehungen laut aus: „Zu EINEM ... gehört / gehören ...“!
12. Übung: Überführen Sie die logischen Datenmodelle für die Datenbanken „Firma“ (S. 87) bzw. „Verein“ (S. 93) bzw. „Verleih“ (S. 98) jeweils ins physische Modell!

5 Datenmodellierung – Vertiefung

5.1 Arbeitsaufgaben

Hier kommen noch ein paar spezielle Themen und Vertiefungen zur Datenmodellierung.

Bitte arbeiten Sie im 2. Kapitel die folgenden Unterkapitel durch:

- Anschrift1, Anschrift2 (S. 83)
 - Typ (S. 84)
 - Status (S. 84)
- Mengen und Zeiten in »Zwischentabellen« (S. 86)
- Modellbesonderheiten und -erweiterungen (S. 103)
 - 1:1-Beziehungen (S. 103)
 - Dreifach-Beziehungen (S. 107)
 - Rollen in Beziehungen (S. 108)
 - Typ, Art, Status (S. 110)
- ADaMo (S. 125)

5.2 Kontrollfragen

Nach der Erledigung der Arbeitsaufgaben sollten Sie in der Lage sein, folgende Fragen zu beantworten:

- a) Warum sind zwei Spalten „Anschrift1“ und „Anschrift2“ in einer Tabelle schlecht?
- b) Wie sieht die bessere Lösung aus?
- c) Wofür kann man „Typ“- und „Status“-Tabellen benutzen?
- d) Warum gehören manche Mengen und Zeiten nicht in die „normalen“ Tabellen, sondern in die Zwischentabellen?
- e) Wie können Sie feststellen, ob eine Eigenschaft in eine Tabelle oder in eine Zwischentabelle gehört?
- f) Welchen Vorteil habe ich, wenn ich die Zwischentabellen schon im logischen Modell einführe?
- g) Welche Konsequenzen ergeben sich daraus, dass zwei Objekte „unterscheidbar“ sind oder nicht?
- h) Warum braucht man i.a. keine 1:1-Beziehungen?
- i) Wie wird eine 1:1-Beziehung in Access korrekt realisiert?
- j) Wie wird eine Dreifach-Beziehung im logischen Modell realisiert?
- k) Wofür braucht man „Rollen“ in Beziehungen und wie werden sie im logischen Modell realisiert?
- l) Was macht „ADaMo“?

5.3 BeMERKENswertes (S. 83-116 und S. 125/126)

1. DB Firma: Anschrift1, Anschrift2 -> Tabelle mit Kontaktangaben
2. Kontaktyp
3. Auftrags-Status statt tblAnfrage + tblAuftrag
4. Kauf_Mat, Kauf_Mit: Mengen und Zeiten in Zwischentabellen
5. Umwandlung einer m:n-Beziehung in zwei 1:n-Beziehungen
6. Material / Materialart
7. 1:1-Beziehungen
 - Rechnung – Auftrag
 - meistens nicht erforderlich
 - PersonGeheim – PersonOeffentlich
 - Achtung: Fremdschlüssel „ohne Duplikate“!
8. Dreifachbeziehungen
 - VerleihDB: Buch – Person – Mahnung
 - zuerst Zwischentabelle, dann Beziehung zur Zwischentabelle
 - Satz mit drei Entitäten
9. Rollen in Beziehungen
 - FirmenDB: Kunde – Auftrag (Besteller, Warenempfänger, Rechnungsempfänger)
 - mehrere 1:n-Beziehungen oder eine Zwischentabelle
10. Typ, Art, Status
 - in allen drei Beispiel-Datenbanken
 - Typ als Primärschlüssel / künstlicher Primärschlüssel
11. ADaMo
 - vorgegebenes Beispiel nachvollziehen
 - eigenes Beispiel konstruieren
12. Übung: Erstellen Sie ein eigenes Datenmodell – im ersten Ansatz ohne die hier behandelten Modellbesonderheiten. Überlegen Sie dann nochmal genauer, ob Ihr Modell evtl. einige der Besonderheiten erforderlich macht und bauen Sie diese ins Modell ein!

6 Tabellen, Beziehungen, Daten

6.1 Arbeitsaufgaben

Wenn Sie ein gutes, wohlüberlegtes Datenmodell haben, können Sie endlich Access starten und damit beginnen, Tabellen anzulegen und Daten einzugeben.

Bitte arbeiten Sie im 3. Kapitel die folgenden Unterkapitel durch:

- Tabellen anlegen (S. 128)
- Beziehungen definieren (S. 132)
 - Referentielle Integrität (S. 132)
- Kardinalitäten (S. 134)
- Löschweitergabe (S. 134)
- Testdaten eingeben (S. 135)
 - Auf die Reihenfolge achten (S. 136)
 - Aus Fehlern lernen (S. 137)
 - Nachschlagefelder (S. 138)

6.2 Kontrollfragen

Nach der Erledigung der Arbeitsaufgaben sollten Sie in der Lage sein, folgende Fragen zu beantworten:

- a) Warum sollte die Option „Beim Schließen komprimieren“ aktiviert werden?
- b) Was ist zu tun, um in Access eine Tabelle anzulegen?
- c) Warum haben Fremdschlüssel nicht den Datentyp „Autowert“? Welchen Datentyp hat er dann?
- d) Wie wird ein Muss-Feld in Access korrekt realisiert?
- e) Wie wird in Access eine Beziehung zwischen zwei Tabellen definiert?
- f) Was bedeutet „referentielle Integrität“ und wie schaltet man sie ein?
- g) Inwiefern stellt das Beziehungsfenster in Access eine Mischung aus logischem und physischem Datenmodell dar?
- h) Was macht die Löschweitergabe, wie stellt man sie ein und warum ist sie gefährlich?
- i) Worauf muss bei der Eingabe von Testdaten geachtet werden? Warum?
- j) Warum sind Nachschlagefelder schlecht?

6.3 BeMERKENswertes (S. 128-138)

1. physisches Datenmodell DB Firma als Grundlage
2. leere DB anlegen
 - im 2003er Format, damit keine eingebetteten Makros erzeugt werden!!!
3. erste Tabelle
 - Primärschlüssel
 - Feldgröße / Standardwert / Eingabe erforderlich / leere Zeichenfolge
 - Fremdschlüsselfelder
 - Entwurfsansicht / Datenblattansicht der Tabelle
4. Beziehungen
 - drag&drop
 - referentielle Integrität
 - Kardinalitäten
 - Löschweitergabe
5. Testdaten eingeben
 - Reihenfolge
 - Primärschlüssel / Fremdschlüssel
 - zum Üben absichtlich Fehler machen
 - keine Nachschlagefelder benutzen
6. Übung: Realisieren Sie die physisches Datenmodelle „Verein“ und/oder „Verleih“ und/oder „Firma“ als Access-DB! Sie können dabei auch ADaMo benutzen!
7. Übung: Erstellen Sie mit Access eine eigene Datenbank (unter Benutzung Ihres physischen Modells)! Sie können dabei auch ADaMo benutzen!

7 Ein Stammdatenformular

7.1 Arbeitsaufgaben

Anfänger wollen immer am liebsten damit anfangen: Formulare. Das geht aber erst jetzt – nachdem Sie viele, viele Vorbereitungen getroffen haben!

Bitte arbeiten Sie im 3. Kapitel die folgenden Unterkapitel durch:

- Ein einfaches Stammdatenformular (S. 138)
 - Schritt 1: Mit dem Formular-Assistenten Formulare erzeugen (S. 139)
 - Schritt 2: Text- und Bezeichnungsfelder anordnen (S. 140)
 - Schritt 3: Listenfeld hinzufügen (S. 142)
 - Schritt 4: Schaltflächen hinzufügen (S. 145)
 - Schritt 5: Standardbedienelemente entfernen (S. 146)
 - Schritt 6: VBA-Code ergänzen (S.147; ergänzend hierzu arbeiten Sie bitte im Kapitel 8 das Unterkapitel „Standardlösungen / VBA-Code in Formularentwürfen ergänzen“ durch! S. 326)
 - Schritt 7: Layout gestalten (S. 148)

7.2 Kontrollfragen

Nach der Erledigung der Arbeitsaufgaben sollten Sie in der Lage sein, folgende Fragen zu beantworten:

- a) Was sind Stammdaten?
- b) Welche Daten zeigt ein einfaches Stammdatenformular?
- c) Welche Steuerelemente enthält ein einfaches Stammdatenformular typischerweise?
- d) Wobei unterstützt Sie der Formular-Assistent?
- e) Wofür brauchen Sie im Stammdatenformular ein Listenfeld?
- f) Worauf müssen Sie beim Einfügen des Listenfeldes mit dem Assistenten achten?
- g) Was müssen Sie tun, damit die Daten im Listenfeld alphabetisch sortiert aufgelistet werden?
- h) Was ist der „Abfrage-Generator“?
- i) Welche Standardbedienelemente hat ein Formular, wie können Sie sie entfernen und warum sollten Sie das tun?
- j) Wie sollten die Standard-Steuerelemente auf dem Formular angeordnet werden?
- k) Warum kommen Sie nicht ganz ohne Änderungen am VBA-Code aus? Was ist konkret zu tun?
- l) Was besagt der Grundsatz „Erst richtig, dann schön!“?

7.3 BeMERKENswertes (S. 139-149)

1. weiter mit der DB Firma
2. Optionen einstellen
3. Schritt 1: Mit dem Formular-Assistenten Formulare erzeugen
4. Schritt 2: Text- und Bezeichnungsfelder anordnen
 - Verbindung zwischen den Textfeldern aufheben
 - Formularkopf und –fuß
 - Ausrichtung am Raster
5. Schritt 3: Listenfeld hinzufügen
 - Steuerelement-Assistenten einschalten!
 - dritte Option!
 - alphabetische Sortierung herstellen
 - Listennamen löschen
6. Schritt 4: Schaltflächen hinzufügen
 - Namenskonvention beachten!
7. Schritt 5: Standardbedienelemente entfernen
8. Schritt 6: VBA-Code ergänzen
 - Probleme beim Speichern und Löschen
 - siehe auch im Kapitel 8 das Unterkapitel „Standardlösungen / VBA-Code in Formularentwürfen ergänzen“
9. Schritt 7: Layout gestalten
10. Übung: Entwickeln Sie Stammdatenformulare für einige Tabellen aus den Beispieldatenbanken!
11. Übung: Entwickeln Sie Stammdatenformulare in Ihrer eigenen Datenbank!

8 Ein Formular mit Unterformular

8.1 Arbeitsaufgaben

Ein Formular soll im Allgemeinen Daten aus mehreren Tabellen zusammen auf den Bildschirm bringen. Dazu kommen wir jetzt.

Bitte arbeiten Sie im 3. Kapitel die folgenden Unterkapitel durch:

- Ein Formular mit Unterformular (S. 149)
 - Schritt 1: Mit dem Formular-Assistenten Formulare erzeugen (S. 149)
 - Schritt 2: Text- und Bezeichnungsfelder anordnen (S. 152)
 - Schritt 3: Listenfeld hinzufügen (S. 152)
 - Schritt 4: Schaltflächen hinzufügen (S. 153)
 - Schritt 5: Standardbedienelemente entfernen (S. 153)
 - Schritt 6: VBA-Code ergänzen (S. 154; ergänzend hierzu arbeiten Sie bitte im Kapitel 8 das Unterkapitel „Standardlösungen / VBA-Code in Formularentwürfen ergänzen“ durch! S. 326)
 - Schritt 7: Layout gestalten (S. 154)

8.2 Kontrollfragen

Nach der Erledigung der Arbeitsaufgaben sollten Sie in der Lage sein, folgende Fragen zu beantworten:

- a) Was bewirkt die Aktivierung der Option „Überlappende Fenster“?
- b) Woran erkennt Access automatisch, dass ein Unterformular benötigt wird und wie äußert sich das?
- c) Welchen Namen sollte das Unterformular haben? Warum sollte es nicht den von Access vorgeschlagenen Namen haben?
- d) Welches Standard-Bedienelement wird im Unterformular nicht entfernt? Wofür brauchen wir es?
- e) Warum programmieren wir einen Löschbutton im Unterformular?
- f) Was ist der Unterschied zwischen der Liste im Listenfeld und der Liste im Unterformular?

8.3 BeMERKENswertes (S. 149-155)

1. Ziel: Formular mit Kunden und dazugehörigen Kontakten
2. Schritt 1: Mit dem Formular-Assistenten Formulare erzeugen
 - Optionen einstellen
 - „nach Kunde“ oder „nach Kontakt“?
 - „mit Unterformular“ / „verknüpfte Formulare“
 - Namenskonvention: „frmKunden“ und „frmKunden_ufoKontakte“!!
 - abwechselnde Färbung der Liste im Unterformular
3. Schritt 2: Text- und Bezeichnungsfelder anordnen
4. Schritt 3: Listenfeld hinzufügen
5. Schritt 4: Schaltflächen hinzufügen
6. Schritt 5: Standardbedienelemente entfernen
7. Schritt 6: VBA-Code ergänzen
 - Probleme beim Speichern und Löschen
 - siehe auch im Kapitel 8 das Unterkapitel „Standardlösungen / VBA-Code in Formularentwürfen ergänzen“
8. Schritt 7: Layout gestalten
 - Bedienelemente des Unterformulars: Schreibstift und Sternchen
 - Kombinationsfeld für die Kontaktart
9. Übung: Entwickeln Sie Formulare mit Unterformularen für einige Tabellen aus den Beispieldatenbanken!
10. Übung: Entwickeln Sie Formulare mit Unterformularen in Ihrer eigenen Datenbank!

9 “Einzelzuordnung”

9.1 Arbeitsaufgaben

Unter dieser merkwürdigen Überschrift geht es darum, einem Objekt ein anderes Objekt zuzuordnen; also z.B. einem Auftrag einen Kunden zuordnen.

Bitte arbeiten Sie im 3. Kapitel die folgenden Unterkapitel durch:

- Einzelzuordnung (S. 155)
 - Schritte 1 bis 3 (S. 155)
 - Schritt 4: Aktive Bedienelemente hinzufügen (S. 156)
 - Schritte 5 bis 7 (S. 158)

9.2 Kontrollfragen

Nach der Erledigung der Arbeitsaufgaben sollten Sie in der Lage sein, folgende Fragen zu beantworten:

- a) Was bewirkt die Aktivierung der Option „Überlappende Fenster“?
- b) Was ist eine „Einzelzuordnung“?
- c) Wie funktioniert eine Einzelzuordnung mittels Kombinationsfeld ganz genau? Welche Rolle spielen dabei die Datensatzherkunft und der Steuerelementinhalt des Kombinationsfeldes?

9.3 BeMERKENswertes (S. 155-160)

1. Einzelzuordnung
 - zu einem Kundenauftrag gehört ein Kunde
2. Schritte 1 bis 3
3. Schritt 4: Aktive Bedienelemente hinzufügen
 - „Zuordnen“ = Eintragen eines Primärschlüsselwertes in ein Fremdschlüsselfeld (kun_id aus tblKunde -> kun_id_f in tblKundenauftrag)
 - Kombinationsfeld anlegen
 - „Wert speichern in“
 - Kombinationsfeld zur Anzeige und zur Zuordnung -> Absicherung!
4. Schritte 5 bis 7
5. nochmal Kundenformular mit UFO Kontakte ansehen: dort Kontaktart als Einzelzuordnung!
6. Übung: Entwickeln Sie Formulare mit Einzelzuordnung für einige Tabellen aus den Beispieldatenbanken!
7. Übung: Entwickeln Sie Formulare mit Einzelzuordnung in Ihrer eigenen Datenbank!

10 Das Startformular

10.1 Arbeitsaufgaben

Jetzt ist die Datenbank-Anwendung fast fertig – sie muss nur noch benutzerfreundlich gestartet werden.

Bitte arbeiten Sie im 3. Kapitel die folgenden Unterkapitel durch:

- Das Startformular (S. 160)

10.2 Kontrollfragen

Nach der Erledigung der Arbeitsaufgaben sollten Sie in der Lage sein, folgende Fragen zu beantworten:

- a) Was macht ein Startformular und welche Steuerelemente enthält es?
- b) Warum braucht das Startformular keine Standard-Bedienelemente?
- c) Warum brauchen Sie für den Entwurf des Startformulars keinen Formular-Assistenten?
- d) Welche Angaben sollte das Startformular enthalten?
- e) Was ist ein Makro? Was ist ein Autoexec-Makro?
- f) Wofür benutzen wir es beim Thema „Startformular“?
- g) Wie kann man Startformular auch ohne Autoexec-Makro automatisch beim Öffnen der Datenbank anzeigen lassen?
- h) Wenn wir alles richtig gemacht haben – was passiert nach dem Doppelklick auf die fertige Datenbank-Datei? Warum?

10.3 BeMERKENswertes (S. 160-161)

1. Startformular
 - leeres Formular ohne Assistenten
 - Titel, Kontaktangaben, Version
 - Buttons
2. Autoexec-Makro
 - Makro als abgespeicherte Befehlsfolge
 - Makro-Befehl "ÖffnenFormular"
3. Startformular ohne Makro öffnen
4. Übung: Entwickeln Sie ein Startformular in Ihrer eigenen Datenbank!

11 Wie geht's weiter?

Wenn Sie mit Ihrer eigenen Datenbank-Anwendung nicht so 100%ig zurecht kommen, wie Sie sich das wünschen, können Sie sich gerne mit konkreten Fragen an mich wenden. Auf „Geht das jetzt so?“ kann man immer schlecht antworten ...

Bevor Sie mir Ihre Access-Datei mailen, tun Sie bitte zweierlei:

Erstens: komprimieren

Access 2010/2013: Datei / Optionen / Aktuelle Datenbank / Anwendungsoptionen / Beim Schließen komprimieren

Access 2007: Office-Schaltfläche / Access-Optionen / Aktuelle Datenbank / Anwendungsoptionen / Beim Schließen komprimieren

Access 2003: Extras/Optionen/Allgemein/Beim Schließen komprimieren

Zweitens: packen (WinZip, 7zip, WinRar, ...)

Wenn Sie die Aufgaben bis hierher abgearbeitet haben, können Sie schon eine ganze Menge. Die schlechte Nachricht ist: Es fehlt aber auch noch eine ganze Menge - in erster Linie die VBA-Programmierung. Jetzt sollten Sie sich also daran machen, das Buch komplett durchzuarbeiten! Was dann immer noch fehlt, steht am Ende des 10. Kapitels.

Ich wünsche Ihnen viel Erfolg und auch etwas Spaß dabei!

A*