

Schritt 6: VBA-Code ergänzen

Jetzt können Sie sich zufrieden zurücklehnen und stolz behaupten, das erste Access-Formular erstellt zu haben. Die Freude währt aber nur kurz, denn beim sorgfältigen Ausprobieren aller Funktionen stellt sich heraus, dass das Formular so noch nicht 100%ig brauchbar ist. Die Übeltäter sind die beiden Schaltflächen *speichern* und *löschen*.

Probieren Sie es einmal selbst aus: Wenn Sie den Namen eines Mitarbeiters ändern und auf *speichern* klicken, erscheint der geänderte Name nicht sofort im Listefeld. Das ist erst der Fall, wenn Sie im Listefeld auf den Namen eines anderen Mitarbeiters klicken.

Wenn Sie einen neuen Mitarbeiter hinzufügen und auf *speichern* klicken, erscheint der neue Name überhaupt nicht im Listefeld, sondern erst, wenn Sie das Formular schließen und erneut öffnen.

Um das erfolgreiche Löschen auszuprobieren, sollten Sie zunächst einen neuen Mitarbeiter anlegen und ihn dann wieder löschen, denn fast alle in der Beispieldatenbank vorhandenen Mitarbeiter können nicht gelöscht werden (siehe die Erläuterung dazu im Abschnitt »Schritt 4: Schaltflächen hinzufügen« weiter oben).

Beim Löschen passiert dann nämlich Folgendes: Der gelöschte Mitarbeiter verschwindet nicht aus dem Listefeld. Aber wenn Sie nach dem Löschen einen anderen Mitarbeiter in der Liste auswählen, erscheint anstelle des gelöschten Namens der Text *#Gelöscht*.



Wichtig Diese Effekte sind kein Hinweis auf »falsche Daten«. Es ist alles völlig korrekt. Aber ein Formular, das in der beschriebenen Weise reagiert, verhält sich nicht erwartungsgemäß und verwirrt damit den Benutzer.

Die Ursache für dieses Verhalten habe ich bereits in Kapitel 1 erläutert. Das Formular ist Bestandteil der Benutzeroberfläche und als solches fordert es Daten per SQL beim DBMS an und stellt sie auf dem Bildschirm dar. Die Anforderung von Daten beim DBMS erfolgt einmal beim Öffnen des Formulars und dann wieder jedes Mal, wenn im Listefeld ein neuer Mitarbeiter ausgewählt wird. Sie erfolgt aber offenbar **nicht** beim Klicken auf die Schaltflächen *speichern* und *löschen*. Um das zu ändern, müssen wir in den Code eingreifen, der beim Klick auf diese Schaltflächen ausgeführt wird.

Beim Stichwort »in den Code eingreifen« ergibt sich ein neues Problem:

Leider ist es mit der 2007er-Version von Access eingeführt worden, dass die Schaltflächen-Assistenten keinen VBA-Code mehr generieren. Stattdessen werden Sie auf der Registerkarte *Ereignis* des Eigenschaftenblattes den Eintrag [*Eingebettetes Makro*] finden. Das geht natürlich auch – aber: Ich halte es nicht für sinnvoll, einen Teil des Programmcodes in VBA zu schreiben und einen anderen Teil als Makro. Das mag für fortgeschrittene Programmierer Vorteile haben, für Anfänger ist das einfach nur verwirrend.



Hinweis Das war auch der Grund für meinen auf den ersten Blick anachronistischen Rat ganz am Anfang dieses Kapitels, zunächst mit dem 2003er-Datenbankformat zu arbeiten. Dann erzeugen die Assistenten nämlich VBA-Code statt Makros!

Wenn Sie Ihre Access-Anwendung bereits mit eingebetteten Makros »verseucht« haben, rate ich Ihnen, sie durch selbst geschriebenen VBA-Code zu ersetzen. Wie das geht, steht in Kapitel 8 im Abschnitt »Standardlösungen / VBA-Code in Formularentwürfen ergänzen«.

So – was machen wir jetzt? Ohne eine kleine Änderung an dem VBA-Code, den der Assistent generiert hat, wird Ihre Anwendung beim Speichern und Löschen nicht richtig funktionieren. Das Thema »VBA« kommt aber erst weiter hinten im Buch. Daher gebe ich Ihnen jetzt eine Art Kochrezept, nach dem Sie vorgehen können, ohne wirklich verstehen zu müssen, was Sie da tun.

Machen Sie einfach folgendes:

1. Öffnen Sie das Formular in der Entwurfsansicht, klicken Sie mit der rechten Maustaste auf die Befehlsschaltfläche *speichern* und wählen Sie *Eigenschaften*.
2. Auf dem Eigenschaftenblatt wählen Sie das Register *Ereignis*. Dort sehen Sie in der Zeile *Beim Klicken* den Eintrag [*Ereignisprozedur*]. Dahinter verbirgt sich der vom Assistenten generierte VBA-Code, den wir jetzt um einen Befehl ergänzen wollen. (Wenn Sie in Ihrer eigenen Datenbank statt des Eintrages [*Ereignisprozedur*] den Eintrag [*Eingebettetes Makro*] sehen, so liegt das daran, dass Sie leider meinen Rat nicht befolgt haben, die Datenbank im 2003er Format anzulegen. Dann geht es hier leider nicht weiter und Sie müssen nochmal vor vorne beginnen.)
3. Klicken Sie rechts neben dem Eintrag [*Ereignisprozedur*] auf die Schaltfläche mit den drei Punkten. Daraufhin öffnet sich das VBA-Fenster. Die Einfügemarke steht unterhalb der Zeile *Private Sub cmdSpeichern_Click()*.
4. Unmittelbar hinter der Codezeile *DoCmd.RunCommand acCmdSaveRecord* fügen Sie die folgende Zeile ein: *Liste5.Requery*. ACHTUNG: Die Liste wird bei Ihnen wahrscheinlich eine andere Nummer als 5 haben. Welche das ist, bekommen Sie heraus, wenn Sie im VBA-Fenster etwas hoch oder herunter scrollen und nach der Zeile *Private Sub Liste5_AfterUpdate()* suchen. Dort steht bei Ihnen dann evtl. *Private Sub Liste17_AfterUpdate()* – oder irgendeine andere Zahl. Dann müssen Sie natürlich *Liste17.Requery* statt *Liste5.Requery* schreiben.
5. Schließen Sie das VBA-Fenster. Das Formular ist nach wie vor in der Entwurfsansicht geöffnet. Klicken Sie mit der linken Maustaste auf die Schaltfläche *löschen*.
6. Klicken Sie auf dem Eigenschaftenblatt wieder auf die Schaltfläche mit den drei Punkten rechts neben dem Eintrag [*Ereignisprozedur*] in der Zeile *Beim Klicken*. Das VBA-Fenster öffnet sich wieder.
7. Fügen Sie unmittelbar hinter der Code-Zeile *DoCmd.RunCommand acCmdDeleteRecord* wieder die Zeile *Liste5.Requery* ein. ACHTUNG: Wegen der Nummer der Liste ist wieder der Hinweis aus Punkt 4 zu beachten!
8. Schließen Sie das VBA-Fenster und öffnen Sie das Formular in der Formularansicht. Die Schaltflächen *speichern* und *löschen* sollten jetzt ordnungsgemäß funktionieren!



Hinweis Wenn Sie bei dieser Aktion etwas falsch machen und VBA Ihnen eine Fehlermeldung anzeigt, so können Sie im Kapitel 5 im Abschnitt »Fehler finden und korrigieren« nachlesen, was zu tun ist.