

## Inhalt

<b>1</b>	<b>Ein Datenbanksystem für die Bekleidungsbranche.....</b>	<b>3</b>
1.1.1	Aufbau und inhaltliche Konventionen des Skriptes .....	3
1.1.2	Allgemein zum Datenbankmanagementsystem .....	3
1.1.3	Auswahl der Aufgabenstellung.....	4
1.1.4	Programmstruktur von Microsoft Access .....	4
1.1.5	Wichtige Funktionen.....	5
1.2	Vorgehensweise des ersten Abschnittes.....	5
1.2.1	Namenkonventionen .....	5
1.3	Eigenschaften und Methoden von Objekten .....	7
1.3.1	Ästhetik vom Formular .....	7
1.3.2	Eigenschaftenblatt vom Formular .....	8
1.3.3	Eigenschaftenblatt vom Formularkopf.....	9
1.3.4	Eigenschaftenblatt vom Formulardetailbereich.....	9
1.3.5	Eigenschaftenblatt vom Registersteuerelement.....	10
1.3.6	Eigenschaftenblatt von der Seite eines Registersteuerelementes.....	11
1.3.7	Eigenschaftenblatt vom Bezeichnungsfeld.....	12
1.3.8	Eigenschaften und Methoden vom Textfeld .....	13
1.3.9	Eigenschaften und Methoden vom Kombinationsfeld.....	15
1.3.10	Eigenschaften und Methoden vom Listenfeld.....	17
1.3.11	Eigenschaften und Methoden von der Befehlsschaltfläche.....	19
1.3.12	Eigenschaften vom Bild.....	21
1.3.13	Eigenschaften vom Kontrollkästchen .....	22
1.3.14	Eigenschaften vom Optionsfeld.....	23
1.4	Tabellen- und Datenfeldeigenschaften .....	24
1.4.1	Normalisierungsformen.....	24
1.4.2	Feldeigenschaften in Abhängigkeit und deren Datentypen.....	27
1.4.3	Tabellenbeziehung für Einkaufen (Auftrag) und Verkaufen(Bestellung).....	31
1.5	Erstellung von Formularen oder Berichten .....	33
1.5.1	Einstellungen und Festlegung der Eigenschaften im Formular frmArtikel.....	33
1.5.2	Datenverarbeitung mit dem Formular frmArtikel .....	35
1.5.3	Einstellungen und Festlegung der Eigenschaften im Bericht rptArtikel.....	36

1.5.4	Datenverarbeitung mit dem Bericht rptArtikel .....	36
1.6	Erstellung und Einsatz von Abfrage .....	37
1.6.1	Abfrage der Tabelle tblStandort .....	37
1.6.2	Abfrage der Tabelle tblArtikelliste.....	38
1.7	Steuerung des Programmablaufs .....	39
1.7.1	Programmablauf z.B. zur Artikelbearbeitung .....	39
1.7.2	Codierung des Programmablaufs für das Formular frmArtikel.....	40
1.7.3	Ereignisse auf die Objekte im Formular frmArtikel.....	40
1.7.4	Ereignis im Bericht rptArtikel .....	46

## 1 Ein Datenbanksystem für die Bekleidungsbranche

### 1.1 Aufbau und Allgemeininformationen

#### 1.1.1 Aufbau und inhaltliche Konventionen des Skriptes

- ✓ Am Anfang jedes Kapitels finden Sie eine kurze Information über die allgemeine Einführung zu dem Programmbeispiel mit wichtigen Funktionen.
- ✓ Die meisten Kapitel enthalten das Programmbeispiel, mit dessen Hilfe praktische Übungen ausgeübt werden können.
- ✓ Das Ziel dieses Skriptes ist einen schnelleren Einstieg in Access zu ermöglichen.

#### Typografische Konventionen

Im Text sind drei Schriftarten

- Calibri: für den Schriftsatz des Skriptes
- Adobe Arabic: für alle zugewiesenen Namen wie Abbildungen, Tabellennamen, Formularnamen, Steuerelementen usw.
- Courier New: für Programmcodes.

angewandt.

#### Symbole



Besondere Informationen, die Ihnen weiterhelfen können.



Besondere Hinweise oder Tipps



Warnhinweise oder besondere Aufmerksamkeit

#### 1.1.2 Allgemein zum Datenbankmanagementsystem

Eine **relationale Datenbank** dient zur elektronischen Datenverwaltung in Computersystemen und beruht auf dem relationalen Datenbankmodell<sup>i</sup>. Dieses wurde 1970 von Edgar F. Codd erstmals vorgeschlagen und ist bis heute, trotz einiger Kritikpunkte, ein etablierter Standard für Datenbanken. Das zugehörige Datenbankmanagementsystem wird als **relationales Datenbankmanagementsystem** oder **RDBMS** (Relational Database Management System) bezeichnet. Zum Abfragen und Manipulieren der Daten wird überwiegend die Datenbanksprache **SQL**<sup>ii</sup> (Structured Query Language) eingesetzt.

Grundlage des Konzeptes *relationaler Datenbanken* ist die Relation, ein im mathematischen Sinn wohldefinierter Begriff. Sie stellt eine mathematische Beschreibung einer Tabelle dar.

Operationen auf diesen Relationen werden durch die *relationale Algebra*<sup>iii</sup> bestimmt. Die relationale Algebra ist somit die theoretische Grundlage von SQL.

Trotz der mathematischen, abstrakten Definition des Datenbankmodells sind relationale Datenbanken vergleichsweise einfach und flexibel zu handhaben. Dies hatte großen Einfluss auf den Erfolg dieser Datenbanktechnik.

Um praktische Aufgaben zu erstellen, wird Microsoft Access in diesem Seminar näher behandelt. **Microsoft Access** ist ein Datenbankmanagementsystem des Softwareherstellers Microsoft zur Verwaltung von Daten in Datenbanken und zur Entwicklung von Datenbankanwendungen. MS Access ist Bestandteil des Office-Professional-Pakets und unterstützt SQL (Structured Query Language).

### 1.1.3 Auswahl der Aufgabenstellung

Dieses Skript bzw. diese Aufgabenstellung soll demonstrieren, wie ein Themenbereich für die Bekleidungstechnik unter dem betriebswirtschaftlichen<sup>iv</sup> Aspekt in Access umgesetzt wird. Um z.B. die Bestandsführung<sup>v</sup> bei einer ausgedachten Firma in der Bekleidungsbranche mit verschiedenen Standorten und Sortimenten durchzuführen, werden u.a. folgende Aktivitäten gebraucht:

- Erstellen aktueller Unterlagen über die Vorräte nach Menge und Wert,
- Erstellen von Nachweisen über lagermäßige Änderungen der geführten Materialien,
- Durchführen der Inventur nach handels- bzw. steuerrechtlichen Vorschriften,
- Überwachen der zu fertigenden Mengen,
- Erstellen von Daten zur Ermittlung des Brutto- und Nettobedarfs,
- Erstellen von Daten für die Bestellabwicklung (Verkaufsabwicklung),
- Überwachen von Ausschuss, ungeplantem Mehrverbrauch, sonstigen Fehlmengen,
- Erstellen, Ändern und Löschen von Bestellmengen,
- Durchführen von Bestandskontrollen.

Bei Erreichen der Meldebestände kann z.B. Bestandsführung eine Bestellung (Verkaufen) ausgelöst werden. Man erkennt in dem hier idealisierten Modell folgende Zusammenhänge: Der Sicherheitsbestand betrage  $S$ . Dann ist der Meldebestand  $2S$  und der Höchstbestand  $3S$ . Der Lagerbestand des betrachteten Artikels pendelt zwischen den zwei Punkten Höchstbestand und Sicherheitsbestand. Bei Erreichen der Meldebestände wird die Bestellung ausgelöst. Der Bestand sinkt während der Wiederbeschaffungszeit (WBZ) bis zum Sicherheitsbestand ab. Genau hier erfolgt der Lagerzugang von  $2S$  aus der Bestellung und füllt den Bestand bis zur Obergrenze auf. Falls eine Lieferung ausfällt, reicht der Sicherheitsbestand zeitlich für eine weitere Wiederbeschaffung aus.

### 1.1.4 Programmstruktur von Microsoft Access

Eine ACCESS-Datei kann aus folgenden Bereichen bestehen:

- **Tabellen** dienen zur Sammlung der Datensätze.

- **Abfragen** dienen dazu, aus den Tabellen einer Datenbank ausgewählte Felder anzusehen und zu ändern. Man kann also Spalten angeben, die in die Abfrage aufgenommen werden sollen. Außerdem können Kriterien angegeben werden, nach denen die anzuzeigenden Datensätze ausgewählt werden. Dies erzeugt also eine Auswahl unter den Zeilen der Tabellen.
- **Formulare** dienen zur Benutzeroberfläche, um die Datensätze standardmäßig zu holen, hinzuzufügen und mit Makros oder VBA-Codes zu löschen, zu sortieren und zu suchen
- **Berichte** dienen zur Auflistung und zum Drucken von den Daten.
- **Makros** dienen zur Steuerung der Arbeitsprozesse ohne den VBA-Einsatz.
- **Module** beinhalten VBA-Syntax für die Steuerung der Steuerelemente und Datenauswertung.

### 1.1.5 Wichtige Funktionen

Diese Aufgabe wird in drei Abschnitten behandelt. Der erste Abschnitt handelt vor allem vom den Aufbau<sup>vi</sup>, der Benutzerfreundlichkeit und Robustheit der Stammdaten, der zweite Abschnitt behandelt das Datenmanagement (wie Statistiken, Auswahlkriterien, Datenverknüpfungen) und im dritten Abschnitt geht es um Erweiterbarkeit und Auswertungen. Dabei sind zu behandeln:

- Erstellen von Tabellen, Formularen, Berichten, Abfragen für Personal, Lieferanten, Kunden, Standort, Artikel, Unterkategorie, Einheiten, Einkauf, Verkauf usw.,
- Erstellung von VBA-Syntax, Makros für die Steuerung der Steuerelemente bzw. des Programmablaufs,
- Umgang mit SQL-Syntax
- Entwurf und Realisation von Datenmanagement.

## 1.2 Vorgehensweise des ersten Abschnittes

### 1.2.1 Namenkonventionen

Es gibt eine Menge Steuerelemente (Abb. A2.1), auch Controls genannt, die auf Formularen Platz finden. Damit gibt man Text und Zahlen ein, schickt Befehle los oder lässt sich Ergebnisse ausgeben. Um sie auf die Formulare einzufügen, ist es am einfachsten, wenn man die Steuerelemente-Toolbox einblendet.

Entweder wird sie schon aktiviert, wenn man ein Formular in der Bearbeitungsansicht (Objektansicht) markiert, oder aber man muss sie anklicken, damit sie angeht. Die allgemeine Logik ist immer die Gleiche. Man klickt mit der Maus auf das Objekt, das man braucht und mit gedrückter linker Maustaste, zieht man den für einen Rahmen auf dem Formular. Danach gibt man dem Teil einen vorteilhaften Namen im Eigenschaftsfenster. Auch hier bietet Microsoft wieder eine Konvention an, wie man diese benennen sollte:














<i>Control-Typ</i>	<i>Präfix</i>	<i>Symbol</i>	<i>Funktion</i>
CheckBox	chk		Kontrollkästchen
ComboBox	cbo		Auswahlliste
Command Button	cmd		Befehlsknopf
Frame	fra		Rahmen (für Optionsfelder)
Image	img		Graphikrahmen
Label	lbl		Beschriftungsfeld
ListBox	lst		Listenfeld
Multipage Page	mlp pg		Dialogunterteilung
Optionsfeld	opt		Optionsfeld
Registerkarte	tab		Dialogunterteilung
ScrollBar	(h/v)sb		Rollbalken
Spin	spn		Drehknopf zum Hörerstellen von Zahlen
TextBox	txt		Textfeld

Abb. A2.1 Steuerelemente in Office

Aus dem Grund, dass Access unter anderem eine direkte Verbindung z.B. zwischen Formularen und Berichten mit Tabellen und Abfragen herstellt, wurden die Namenskonventionen wie folgt festgelegt:

Präfix	Elemente
<b>tbl</b>	Tabelle wie z.B. für das Personal wurde <code>tblPersonal</code> festgelegt.
<b>sql</b>	Abfrage wie z.B. für die Personalliste wurde <code>sqlPersonalliste</code> festgelegt.
<b>frm</b>	Formular wie z.B. für das Personal wurde <code>frmPersonal</code> festgelegt.
<b>rpt</b>	Bericht wie z.B. für das Personal wurde <code>rptPersonal</code> festgelegt.
<b>cbo</b>	wie z.B. Auswahlliste <code>cboPersonal</code> in Formular <code>frmPersonal</code> wurde für die Auswahl eines Mitarbeiters festgelegt.
<b>cmd</b>	wie z.B. Befehlsschaltfläche <code>cmdSpeichern</code> in Formular <code>frmPersonal</code> wurde für Speichern eines Datensatz festgelegt.
<b>Register</b>	wie z.B. Registerkarte <code>RegisterPersonal</code> in Formular <code>frmPersonal</code> wurde für den Registernamen festgelegt.
<b>Lst</b>	wie z.B. Listenfeld <code>LstKategorie</code> in Formular <code>frmUnterkategorie</code> wurde für die Liste der Datensätze von der Tabelle <code>tblkategorie</code> festgelegt.
<b>txt</b>	Für alle ungebundenen Textfelder

**Alle gebundenen Textfelder, Kombinationsfelder, Optionsfelder und Kontrollfelder wurden genau mit dem gleichen Feldnamen wie in der Tabelle oder in der Abfrage festgelegt.**

**Der Name der Bezeichnungsfelder in Formularen und Berichten wurden von dem System festgelegt, d.h. sie wurden nicht geändert.**

### 1.3 Eigenschaften und Methoden von Objekten

Die Formulare, Berichte und die Steuerelemente (Abb. A2.1) in Microsoft Office besitzen sogenannte Eigenschaften und Methoden. Die Eigenschaften können manuell oder auch mithilfe von VBA festgelegt werden. In diesem Skript wurden die Eigenschaften der Objekte manuell einheitlich festgelegt. Die Gestaltung der Objekte steht jedoch jedem frei festzulegen.

#### 1.3.1 Ästhetik vom Formular

Jedes Formular im Detailbereich beinhaltet ein Multipage-Element, das aus mindestens einer Seite oder mehreren Seiten besteht. In diese werden Steuerelemente platziert, die eine Verbindung mit den Feldnamen von der entsprechenden Tabelle oder Abfrage herstellen. Im Formularkopf wie z.B. frmKategorie sind ein ungebundenes Kombinationsfeld und drei Befehlsschaltflächen. Dieses Kombinationsfeld im Formularkopf dient dazu, dass ein vorhandener Datensatz angezeigt bzw. gesucht wird.

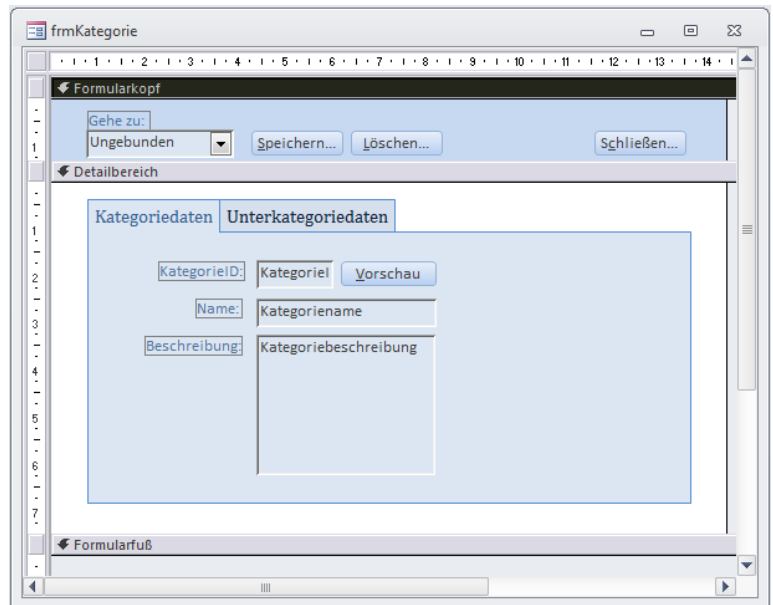


Abb. A1.3.1 das Formular frmKategorie

Die drei Befehlsschaltflächen dienen jeweils dazu, um einen neuen Datensatz zu speichern, um einen Datensatz zu löschen oder das Formular zu schließen.

### 1.3.2 Eigenschaftenblatt vom Formular

Dieses Fenster (Abb. A1.3.2) zeigt alle zur Entwurfszeit einstellbaren Eigenschaften des im Projekt-Explorer ausgewählten Objekts an. Microsoft Offline-Hilfe (zuerst auf dem Eigenschaftsfeld klicken und anschließend auf die Taste F1 drucken) hilft Ihnen jeder Zeit, etwas über die VBA-Codes und Beispiele des Eigenschaftsfeldes zu erfahren. Hier werden die wichtigsten Eigenschaften, die in diesem Programmbeispiel verwendet wurden, kurz beschrieben.

- **Datensatzquelle:** Ist eine Tabelle oder Abfrage für die Datenfelder des Formulars schon festgelegt, wird sie hier ausgewählt bzw. angegeben.
- **Beschriftung:** Es handelt es sich hier um Beschriftung des Formulars.
- **Datensatzmarkierer:** Standard ist auf Ja eingestellt. Hier wurde auf Nein eingestellt.
- **Navigationsschaltflächen:** Damit der Umgang mit dem Datensatz speichern, ändern oder löschen verdeutlicht wird, wurde diese Eigenschaft auf Nein eingestellt. Standard ist auf Ja eingestellt.
- **Mit Systemmenüfeld:** Standard ist, dass das Eigenschaftsfeld auf Ja eingestellt ist. In diesem Beispielprogramm wurde das Eigenschaftsfeld auf Nein eingestellt, weil die Formularansicht kein Systemmenü haben sollte.
- **Schließen Schaltfläche:** Die Schaltfläche Schließen in dem Formular soll nicht aktiv sein. Es wurde ein eigenes Schließen (Siehe im Abschnitt 1.7) festgelegt, um den Zustand der Datensätze besser zu kontrollieren.

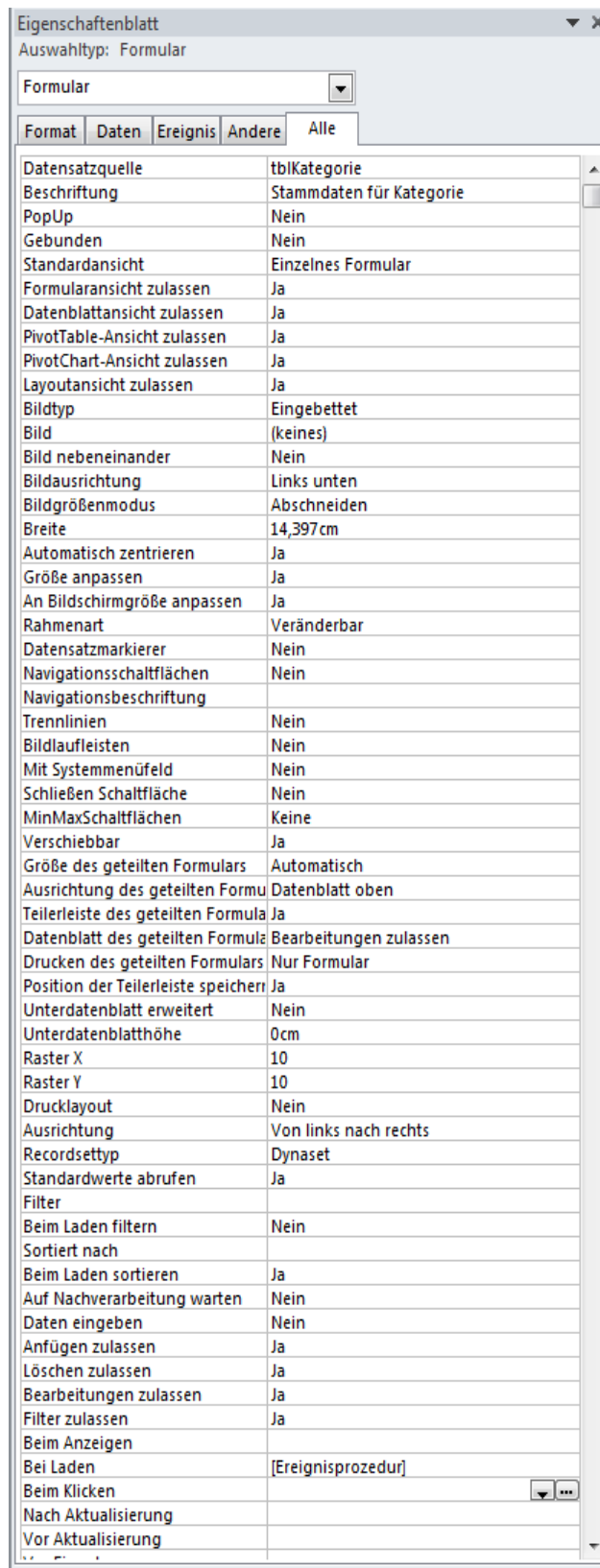


Abb. A1.3.2 Formulareigenschaft von frmKategorie

- **Andere vom System vorgegebenen Einstellungen** wurden genauso übernommen.



### 1.3.3 Eigenschaftenblatt vom Formulkopf

Dieses Fenster (Abb. A1.3.3) zeigt alle zur Entwurfszeit einstellbaren Eigenschaften des im Projekt-Explorer ausgewählten Objekts an. Hier werden die wichtigsten Eigenschaften, die in diesem Programmbeispiel verwendet wurden, kurz beschrieben.

- *Name*: Der Name wurde vom System übernommen.
- *Sichtbar*: Es wurde vom System übernommen.
- *Höhe*: Die gewünschte Höhe wurde auf 1,3 cm eingestellt.
- *Hintergrundfarbe*: Die gewünschte Farbe wurde auf Text2, Heller 80% eingestellt.
- *Spezialeffekt*: Es wurde auf Flach eingestellt.
- *Andere vom System vorgegebenen Einstellungen* wurden genauso übernommen.

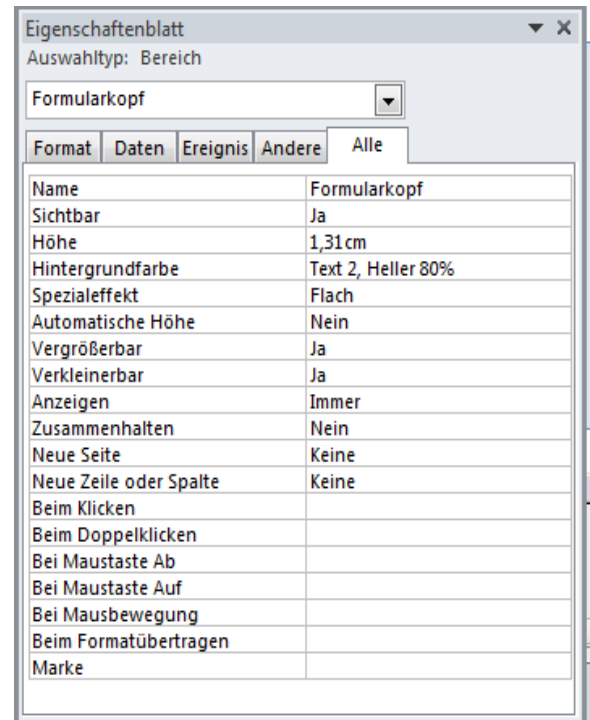


Abb. A1.3.3 Formulkopf von frmkategorie

### 1.3.4 Eigenschaftenblatt vom Formuldetailbereich

Dieses Fenster (Abb. A1.3.4) zeigt alle zur Entwurfszeit einstellbaren Eigenschaften des im Projekt-Explorer ausgewählten Objekts an. Die veränderte Eigenschaften wurden wie folgt festgelegt:

- *Name*: Der Name wurde vom System übernommen.
- *Sichtbar*: Es wurde vom System übernommen.
- *Höhe*: Die gewünschte Höhe wurde auf 7,5 cm eingestellt.
- *Hintergrundfarbe*: Die gewünschte Farbe wurde auf Hintergrund 1 eingestellt.
- *Alternative Hintergrundfarbe*: Die gewünschte Farbe wurde auf Hintergrund 1, Dunkler 5% eingestellt.
- *Spezialeffekt*: Es wurde auf Flach eingestellt.

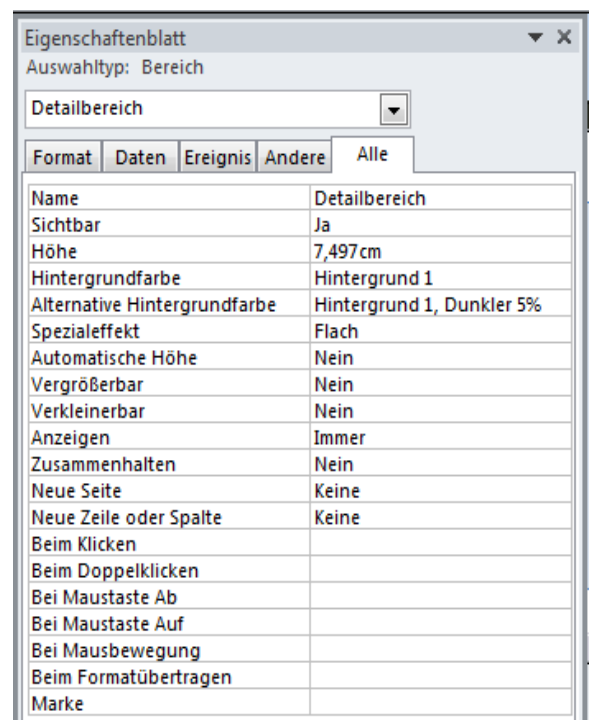


Abb. A1.3.4 Detailbereich von frmkategorie

- *Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.*

### 1.3.5 Eigenschaftenblatt vom Registersteuerelement

Die veränderte Eigenschaften des Registersteuerelementes (Abb. A1.3.5) sind vor allem Hintergrundfarbe, Rahmenart, Farbe für die Mausaktionen, Formatvorlage usw.

- **Name:** Der Name wurde in RegisterKategorie umbenannt.
- **Sichtbar:** Es wurde vom System übernommen.
- **Formatvorlage:** Hierbei ist die Möglichkeit gegeben, als Register, Schaltfläche oder Keine einzustellen. Bei allen Formularen wurde es auf Register eingestellt.
- **Breite, Höhe, Oben und Links:** Die gewünschten Maßen können Sie auch mit der Maus in der Formularansicht einstellen. Wenn Sie aber ein genaues Maß festlegen wollen, können Sie es in entsprechenden Eigenschaftsfeldern eintippen.
- **Hintergrundfarbe, Rahmenart, Rahmenfarbe, Farbe beim Daraufzeigen, Farbe für gedrückten Zustand, Vordergrundfarbe beim Daraufzeigen, Vordergrundfarbe für gedrückten Zustand, Textfarbe und Schriftfarbe:** Die Werte für diese Eigenschaftsfelder können erheblich auf die Ästhetik Ihrer Registersteuerelementen auswirken.
- *Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.*

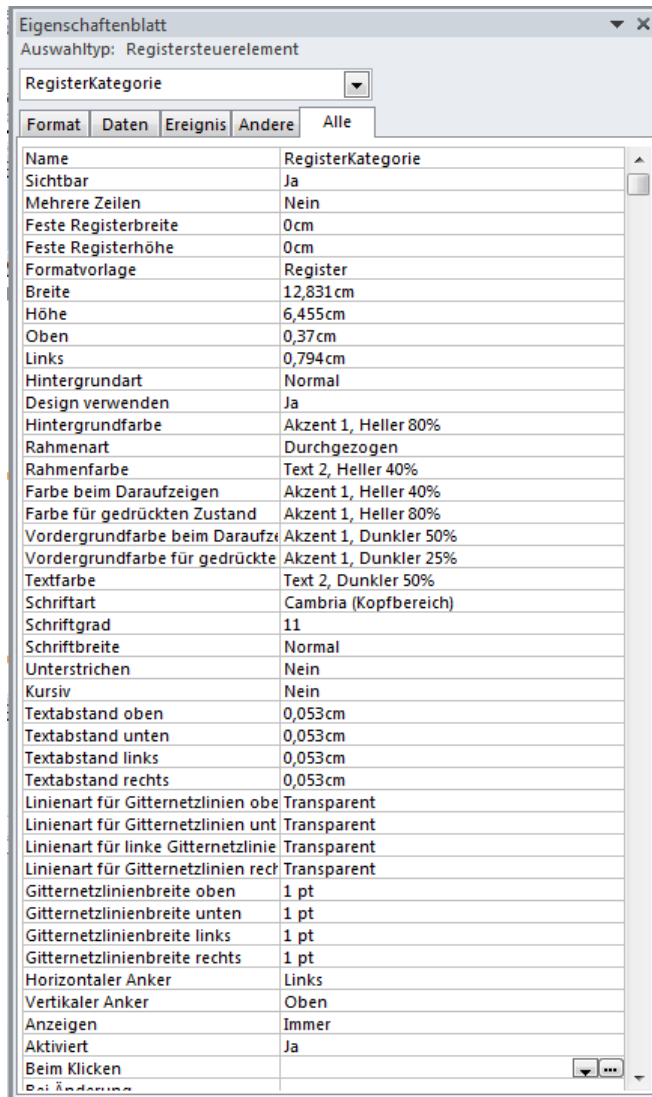


Abb. A1.3.5 Eigenschaftenblatt eines Registersteuerelementes in frmKategorie

### 1.3.6 Eigenschaftenblatt von der Seite eines Registersteuerelementes

Ein Registersteuerelement wird eingesetzt, um die Elemente eines Formulars zu unterteilen und folglich auf einem Bildschirm passend einzurichten. Für einen Benutzer werden die Aufteilungen und die Bearbeitungen der Informationen übersichtlicher. Das Eigenschaftenblatt der Seite eines Registersteuerelementes (Abb. A1.3.6) wurde nur auf die vier folgenden Werte umbenannt:

- *Name*: Damit der Benutzer sehen kann, um welche Information, es sich auf einer Seite handelt, wurde z.B. für die Daten der Artikelkategorie `Kategoriedaten` umbenannt.
- *Seitenindex*: Es wird empfohlen, die erste Seite eines Registersteuerelementes auf 0 einzustellen, falls es vom System anders angegeben wird.
- *Breite, Höhe, Oben und Links*: Die gewünschten Maßen können Sie auch mit der Maus in der Formularansicht einstellen. Wenn Sie aber ein genaues Maß festlegen wollen, können Sie es in entsprechenden Eigenschaftsfeldern eintippen.
- *SteuerelementTip-Text*: Es wird empfohlen, eine Zeichenkette für jedes Steuerelement festzulegen.
- *Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.*

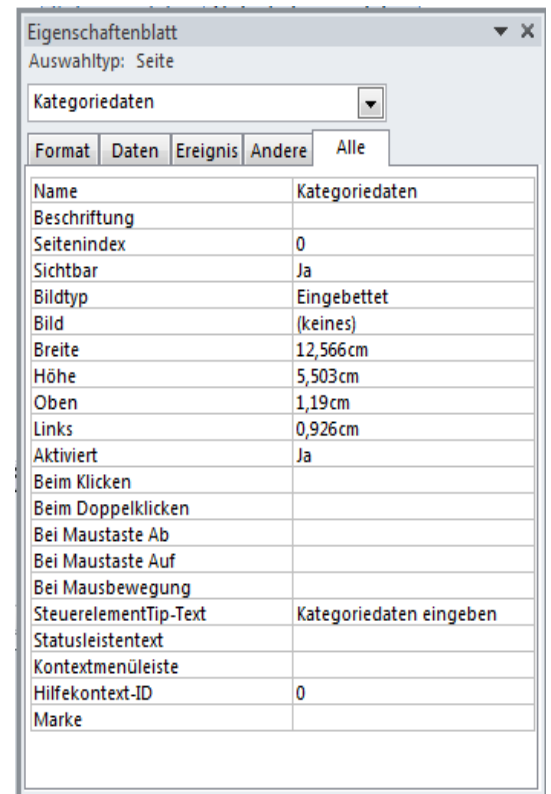


Abb. A1.3.6 Eigenschaftenblatt der Seite eines Registersteuerelementes in `frmkategorie`

### 1.3.7 Eigenschaftenblatt vom Bezeichnungsfeld

Ein Bezeichnungsfeld dient dazu, einen Anwender auf eine Auswahl oder auf eine Eingabe aufmerksam zu machen. In diesem Skript wurden die Namen der Bezeichnungsfelder nicht geändert und vom System vorgeschlagene Namen übernommen. Damit alle Bezeichnungsfelder einheitliche Eigenschaften bekommen, wurden u.a. die Beschriftungen je nach Bedarf umbenannt, und die *Hintergrundart*, *Hintergrundfarbe*, *Rahmenart*, *Rahmenbreite*, *Rahmenfarbe*, *Spezialeffekt*, *Schriftart*, *Schriftgrad*, *Textfarbe* auf die in Abb. A1.3.7 dargestellten Einstellungen umgeändert.

- *SteuerelementTip-Text*: Bei Bezeichnungsfeld wurde hier keine Angabe gemacht.
- *Andere vom System vorgegebenen Einstellungen* wurden genauso übernommen.



Abb. A1.3.7 Eigenschaftenblatt eines Bezeichnungsfeldes z.B. Gehe Zu in frmkategorie

### 1.3.8 Eigenschaften und Methoden vom Textfeld

Ein sehr wichtiges Objekt bei Microsoft-Office/Entwicklungstools ist ein Textfeld, das eine große Rolle in dem Format der Datentype (Access-Hilfe-> DataType-Eigenschaft) spielt. Daher wird dieses in diesem Skript detaillierter beschrieben. Das Register *Andere* (Abb. A1.3.8a) des Eigenschaftenblattes eines Textfeldes soll man einen Namen geben, der möglichst mit dem Feldnamen in der einbezogenen Tabelle oder Abfrage übereinstimmt (Siehe *Datensatzquelle* in diesem Abschnitt Punkt 1.3.2).

- *Name*: Der Name des Feldnamens in der Tabelle oder in der Abfrage.
- *SteuerelementTip-Text*: Das hilft den Benutzer, was er eintippen soll.
- *Vertikal*: Bei langer Zeichenkette wird auf *Ja* eingestellt.

Das Register *Daten* (Abb. A1.3.8b) des Eigenschaftenblattes eines Textfeldes beinhaltet Informationen über

- *Steuerelementinhalt*: der direkten Verbindung mit dem Feldnamen von einer Tabelle oder einer Abfrage (z.B. in diesem Programmbeispiel mit *tblKategorie*) oder auch leer. Wenn in diesem Eintrag nichts steht, wird als ein ungebundener Textfeld festgelegt.

- *Textformat*: der Auswahl als *Nur-Text* (ASCII-Zeichen) oder als *Rich-Text* (Microsoft Textformat).
- *Eingabeformat*: einer bestimmten Zeichenkette ( wie z.B. Datumformat, Artikelnummerierung, ISBN-Nummerierung usw.).
- *Standardwert*: einem Festwert für den Inhalt des Textfeldes.
- *Gültigkeitsregel*: einer eindeutigen Eingabe(wie z.B. Einstellungsdatum darf nicht kleiner oder gleich das Geburtsdatum sein).
- *Gültigkeitsmeldung*: einer Meldung, die im Falle *Gültigkeitsregel* nicht erfüllt wird.
- *Aktiviert*: dem Aktivzustand während der Laufzeit.
- *Gesperrt*: der Eingabesperrung für den Benutzer. In diesem Programmbeispiel wurden alle Feldnamen (z.B. *KategorieID*) in den Tabellen mit dem Datentyp *Autowert* als *Gesperrt* festgelegt.

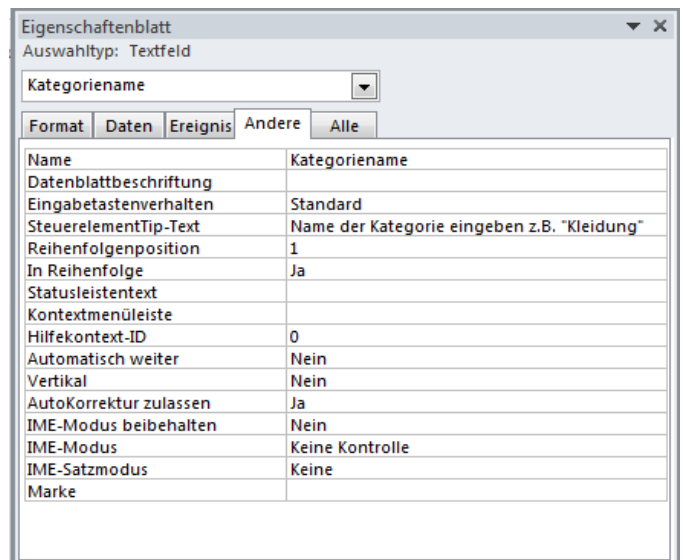


Abb. A1.3.8a Eigenschaftenblatt eines Textfeldes z.B. KategorieName in frmKategorie

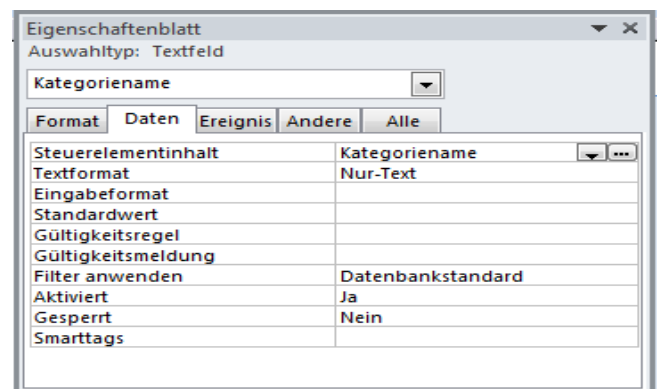


Abb. A1.3.8b Eigenschaftenblatt eines Textfeldes z.B. KategorieName in frmKategorie

Das Register *Format* (Abb. A1.3.8c) des Eigenschaftensblattes eines Textfeldes beinhaltet Informationen über

- *Format*: der Darstellung von Zahlen, Datumsangaben, Zeitangaben und Text auf dem Bildschirm.
- *Dezimalstellenanzeige*: der Anzahl von Dezimalstellen nach Komma, falls das Textfeld als Zahl festgelegt wird.
- den *Maßen*, *Farb-* und *Textattributen*. Damit alle Textfelder einheitliche Eigenschaften bekommen, wurden die *Hintergrundart*, *Hintergrundfarbe*, *Rahmenart*, *Rahmenbreite*, *Rahmenfarbe*, *Spezialeffekt*, *Schriftart*, *Schriftgrad*, *Textfarbe* auf die in Abb. A1.2.8c dargestellten Einstellungen umgeändert.
- *Andere vom System vorgegebenen Einstellungen* wurden genauso übernommen.

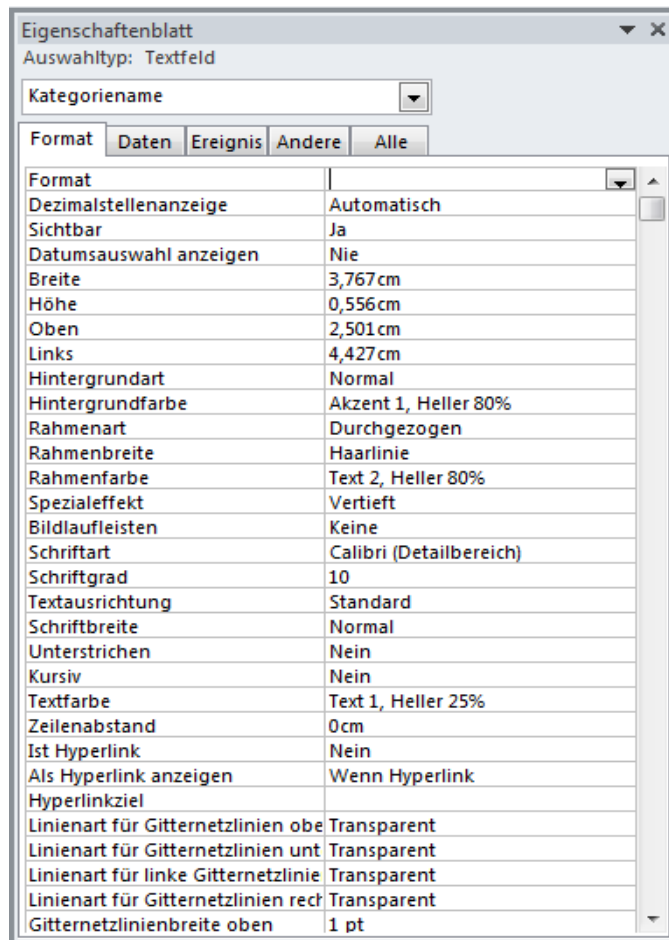


Abb. A1.3.8c Eigenschaftensblatt eines Textfeldes z.B. Kategorienname in frmkategorie

Im Register *Ereignis* (Abb. A1.3.8d) des Eigenschaftensblattes eines Textfeldes sind alle möglichen Methoden, die auf das Textfeld ausgeübt werden können, aufgelistet. Dabei ist wichtig, den Namen des Textfeldes, der im Register *Andere* gegeben wird, als Bezug zu nehmen. Standard sind diese Felder im Eigenschaftensblatt leer. Wenn Sie eine bestimmte Aufgabe mithilfe eines VBA-Codes oder Makros erfüllen wollen, müssen Sie dieses in dem Register *Ereignis* vornehmen. Es soll z.B. in der Tabelle *tblKategorie* auf mehrfachem Vorkommen eines Inhaltes für *Kategorienname* überprüft werden und zwar, wenn das Textfeld *Kategorienname* von dem Benutzer verlassen wird. (Details im Abschnitt 1.7)

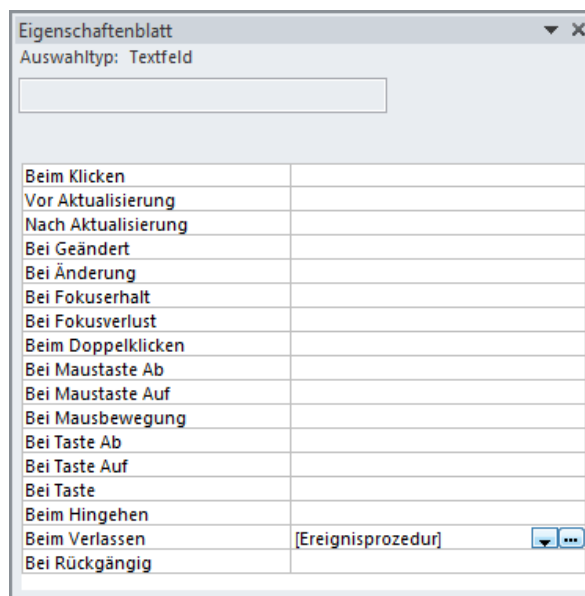


Abb. A1.3.8d Eigenschaftensblatt eines Textfeldes z.B. Kategorienname in frmkategorie



### 1.3.9 Eigenschaften und Methoden vom Kombinationsfeld

Ein weiter wichtiges Objekt bei Microsoft-Office/Entwicklungstools ist ein Kombinationsfeld, das eine große Rolle für die Bereitstellung der Datenfelder (Access-Hilfe-> DisplayControl-Eigenschaft) spielt. Das Register *Andere* (Abb. A1.3.9a) des Eigenschaftenblattes eines Kombinationsfeldes muss mit einem Namen festgelegt werden. In diesem Skript wurden die Kombinationsfelder, die ungebunden sind, mit dem Präfix *cbo* (Siehe in diesem Abschnitt Punkt 1.1 Namenkonvention) benannt. Die gebundenen Kombinationsfelder bekamen den gleichen Namen wie den in der Tabelle oder Abfrage gegebenen Feldnamen. Andere Felder wurden in diesem Register nicht verändert. Bei *SteuerelementTip-Text* wie schon erklärt, soll eine passende Zeichenkette angegeben werden.

Das Register *Daten* (Abb. A1.3.9b) des Eigenschaftenblattes eines Kombinationsfeldes beinhaltet Informationen über

- *Steuerelementinhalt*: der direkten Verbindung mit dem Feldnamen von einer Tabelle oder einer Abfrage. Wenn ein ungebundenes Kombinationsfeldes sein soll, wird kein Element zugewiesen, wie es z.B. in Abb. 1.3.9b dargestellte Feldinhalt leer steht.
- *Datensatzherkunft*: einem/mehreren Feldnamen aus einem/mehreren Tabellenblatt/Tabellenblättern oder Abfrageblatt/Abfrageblättern. (Details im Abschnitt 1.7)
- *Wertliste erben*: der geerbten Wertliste eines Kombinationsfeldes. Standard ist auf *Ja* eingestellt. Wenn Sie aber einen Feldnamen abgefragt haben, der auf mehrfaches Vorkommen des gleichen Wertes überprüft werden soll, wird das auf *Nein* eingestellt. (Details Abschnitt 1.6 Kombinationsfeld für ein Land)

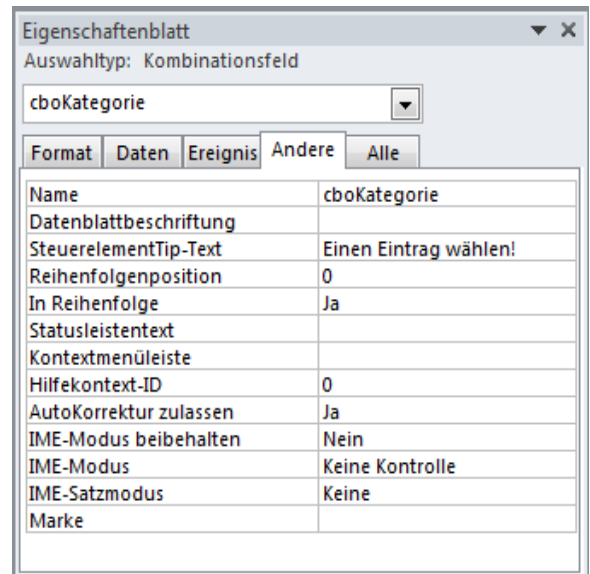


Abb. A1.3.9a Eigenschaftenblatt eines Kombinationsfeldes z.B. cboKategorie in frmkategorie

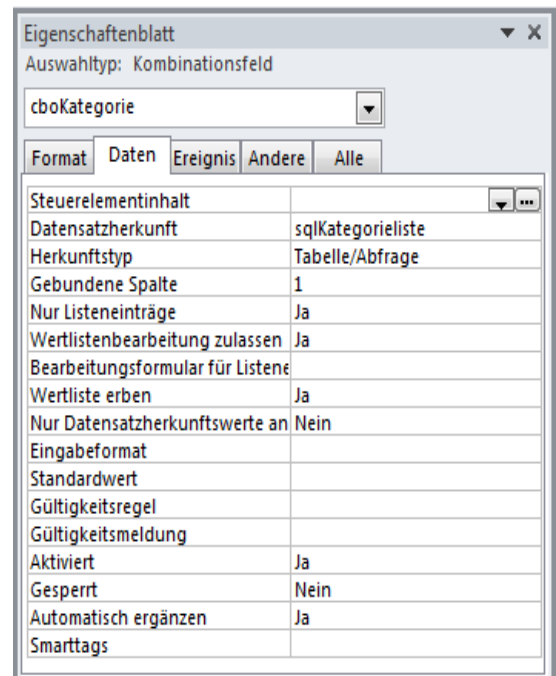


Abb. A1.3.9b Eigenschaftenblatt eines Kombinationsfeldes z.B. cboKategorienname in frmkategorie



Assistent von Access hilft Ihnen, eine schnelle und einfache Abfrage von einem Kombinationsfeld durchzuführen.

Im Register **Format** (Abb. A1.3.9c) des Eigenschaftensblattes eines Kombinationsfeldes können Sie u.a. die Ästhetik bestimmt. Darüber hinaus können Sie die Anzahl und die Breiten der abgefragten Feldnamen (Abb. A1.3.9b Datensatzherkunft) angeben, die wie folgt beschrieben werden:

- **Spaltenanzahl:** Die Anzahl der Spalten, die in einem Listenfeld oder in einem Teil des Listenfelds vorkommt, wird angegeben.
- **Spaltenbreiten:** Es ist darauf zu achten, die Spaltenbreiten für den Inhalt eines Listenfeldes anpassend anzugeben. Sollte ein Inhalt eines Listenfeldes unterdrückt bzw. nicht angezeigt werden, die Spaltenbreiten von dem auf 0cm einzugeben.
- **Listenbreite:** Hier wird die Summe der Spaltenbreiten in cm angegeben.
- Zur Bestimmung der Ästhetik (Abb. A1.3.9c) können Sie unter den Feldeintragungen (*Breite, Höhe, Oben, Links, Hintergrundart, Hintergrundfarbe, Rahmenarten, Rahmenbreite, Rahmenfarbe, Spezialeffekt, Schriftart, Schriftgrad, Textausrichtung, Schriftbreite, Textfarbe* usw.) bestimmen.
- *Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.*

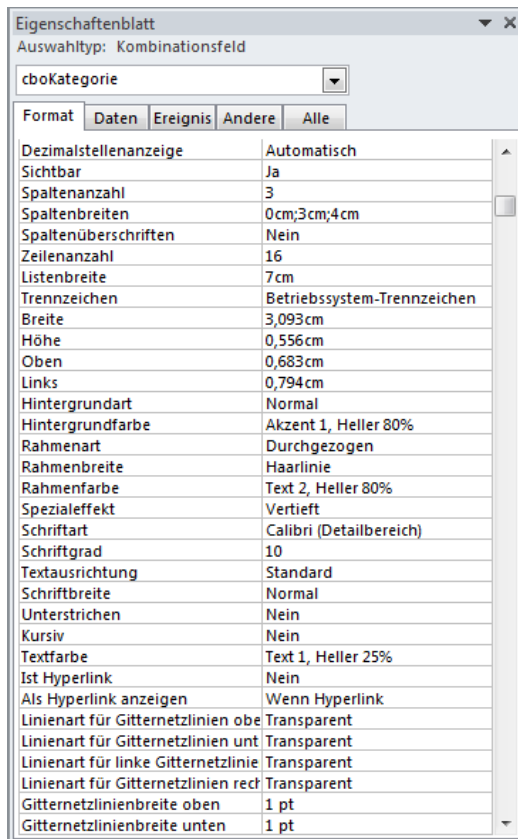


Abb. A1.3.9c Eigenschaftensblatt eines Kombinationsfeldes z.B. cboKategorie in frmKategorie

Im Register **Ereignis** (Abb. A1.3.9d) dieses Eigenschaftensblattes eines Kombinationsfeldes sind alle möglichen Methoden, die auf das ausgeübt werden können, aufgelistet. Dabei ist es wichtig, den Namen des Kombinationsfeldes, der im Register **Andere** angegeben wird, als Bezug zu nehmen. Standard sind diese Felder im Eigenschaftensblatt leer. Wenn Sie eine bestimmte Aufgabe mithilfe eines VBA-Codes oder Makros erfüllen wollen, müssen Sie in dem Register **Ereignis** vornehmen.



Es soll z.B. hier beim Klicken eines Eintrages auf diesem Kombinationsfeld `cboKategorie` den



Abb. A1.3.9d Eigenschaftensblatt eines



gewählten Datensatz aus der Tabelle `tblKategorie` holen und ihn im Formular `frmKategorie` anzeigen.  
(Details im Abschnitt 1.7)

Kombinationsfeldes z.B. `cboKategorie`  
in `frmKategorie`

### 1.3.10 Eigenschaften und Methoden vom Listenfeld

Ein weiteres Objekt bei Microsoft-Office/Entwicklungstools ist ein Listenfeld, das eine Liste ausgewählter Datenfelder (Access-Hilfe-> DisplayControl-Eigenschaft) anzeigt. Das Register `Andere` (Abb. A1.3.10a) des Eigenschaftenblattes eines Listenfeldes muss mit einem Namen festgelegt werden. In diesem Skript wurden die Listenfelder, die ungebunden sind, mit dem Präfix `Lst` (Siehe in diesem Abschnitt Punkt 1.2 Namenkonvention) benannt. Bei `SteuerelementTip-Text` wie schon öfter erklärt, soll eine passende Zeichenkette angegeben werden.

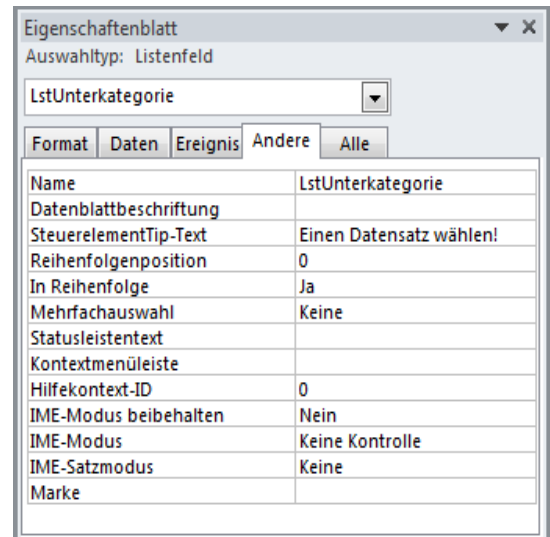


Abb. A1.3.10a Eigenschaftenblatt eines Listenfeldes z.B. `LstUnterkategorie` in `frmKategorie`

Das Register `Daten` (Abb. A1.3.10b) des Eigenschaftenblattes eines Listenfeldes beinhaltet Informationen über

- **Steuerelementinhalt:** wie bei Kombinationsfeld im Abschnitt 1.3.9.
- **Datensatzherkunft:** wie bei Kombinationsfeld im Abschnitt 1.3.9.
- **Wertliste erben:** wie bei Kombinationsfeld im Abschnitt 1.3.9

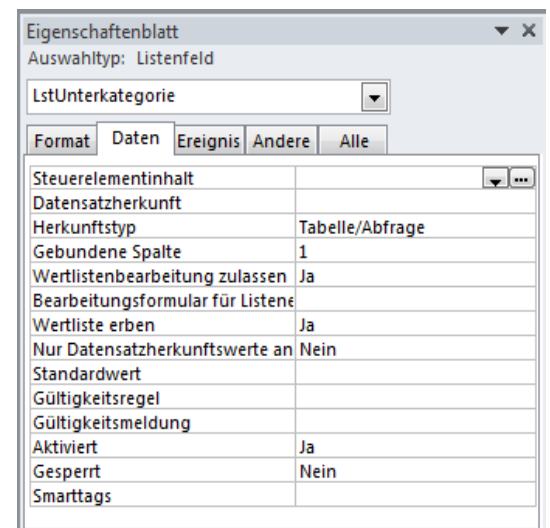


Abb. A1.3.10b Eigenschaftenblatt eines Listenfeldes z.B. `LstUnterkategorie` in `frmKategorie`



Auch hier hilft Ihnen der Assistent von Access, eine schnelle und einfache Abfrage von einem Listenfeld durchzuführen.

Im Register **Format** (Abb. A1.3.10c) des Eigenschaftenblattes eines Listenfeldes können Sie u.a. dessen Ästhetik bestimmen. Darüber hinaus können Sie die Anzahl und die Breiten der abgefragten Feldnamen (Abb. A1.3.10b Datensatzherkunft) angeben, die wie folgt beschrieben werden:

- **Spaltenanzahl:** wie bei Kombinationsfeld im Abschnitt 1.2.9.
- **Spaltenbreiten:** wie bei Kombinationsfeld im Abschnitt 1.2.9.
- **Listenbreite:** wie bei Kombinationsfeld im Abschnitt 1.2.9.
- Zur Bestimmung der Ästhetik (Abb. A1.2.10c) können Sie unter den Feldeintragen (*Breite, Höhe, Oben, Links, Hintergrundart, Hintergrundfarbe, Rahmenarten, Rahmenbreite, Rahmenfarbe, Spezialeffekt, Schriftart, Schriftgrad, Textausrichtung, Schriftbreite, Textfarbe* usw.) bestimmen.
- *Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.*

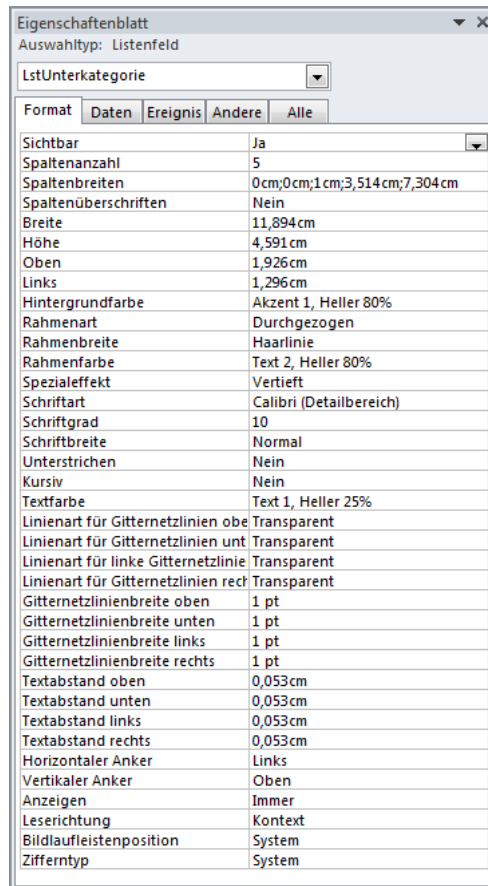



Abb. A1.3.10c Eigenschaftenblatt eines Listenfeldes z.B. LstUnterkategorie in frmkategorie

Im Register **Ereignis** (Abb. A1.3.10d) des Eigenschaftenblattes eines Listenfeldes sind alle möglichen Methoden, die auf dieses ausgeübt werden können, aufgelistet, wie bei Kombinationsfeld im Abschnitt 1.2.9 erklärt wurde.

 Es soll z.B. hier beim Klicken eines Eintrages auf diesem Listenfeld LstUnterkategorie den gewählten Datensatz aus der Tabelle tblUnterkategorie holen und ihn im Formular frmUnterkategorie anzeigen. Das Formular frmUnterkategorie soll dabei geöffnet werden. **(Details im Abschnitt 1.7)**

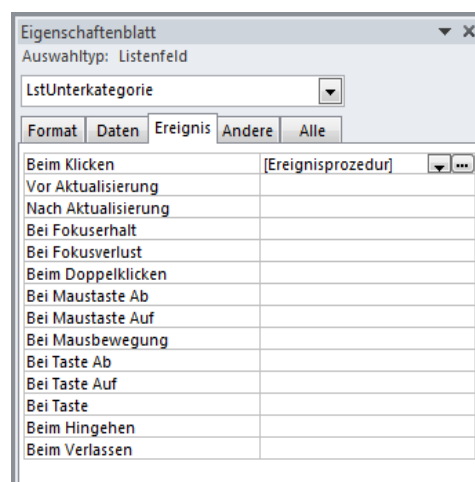


Abb. A1.3.10d Eigenschaftenblatt eines Listenfeldes z.B. LstUnterkategorie in frmkategorie

### 1.3.11 Eigenschaften und Methoden von der Befehlsschaltfläche

Ein weiteres Objekt bei Microsoft-Office/Entwicklungstools ist die Schaltfläche bzw. Befehlsschaltfläche, die die Aktionen für bestimmte Aufgaben beinhaltet. Das Register *Andere* (Abb. A1.3.11a) des Eigenschaftenblattes einer Schaltfläche muss mit einem Namen festgelegt werden. In diesem Skript wurden die Schaltflächen mit dem Präfix *cmd* (Siehe in diesem Abschnitt Punkt 1.1 Namenkonvention) benannt. Der *SteuerelementTip-Text* ist wichtig, den anzugeben, falls eine Schaltfläche keine Beschriftung enthält, oder wenn die Beschriftung keine lesbare Zeichenkette (wie ...) enthält.

Das Register *Daten* (Abb. A1.3.11b) des Eigenschaftenblattes einer Schaltfläche beinhaltet nur den Status der Schaltfläche, ob es mit *Ja* aktiviert oder mit *Nein* deaktiviert eingestellt werden soll.

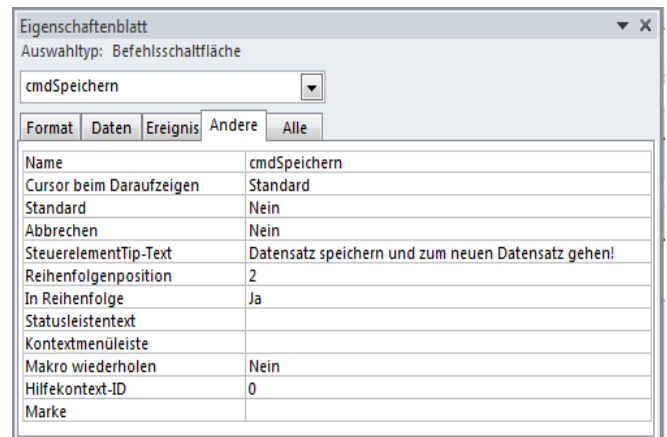


Abb. A1.3.11a Eigenschaftenblatt einer Schaltfläche  
Z.B. cmdSpeichern in frmkategorie

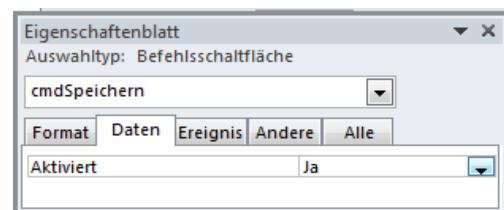



Abb. A1.3.11b Eigenschaftenblatt einer Schaltfläche  
Z.B. cmdSpeichern in frmkategorie

Im Register **Format** (Abb. A1.3.11c) des Eigenschaftensblattes einer Schaltfläche können Sie u.a. deren Ästhetik bestimmen.

Zur Bestimmung der Ästhetik können Sie die Eingaben, wie es in Abschnitt 1.3.5 erklärt wurde, bestimmen.

 Geben Sie in der Beschriftung ein kaufmännisches Und-Zeichen (&) vor dem Zeichen ein, das zur Zugriffstaste werden soll. Das Zeichen wird dann unterstrichen dargestellt. Beim Bewegen im Formular drücken Sie ALT und gleichzeitig das unterstrichene Zeichen, um den Fokus auf dieses Steuerelement zu verschieben. Geben Sie zwei kaufmännische Und-Zeichen (&&) in der Einstellung der Beschriftung eines Steuerelements ein, wenn Sie das kaufmännische Und-Zeichen (&) selbst im Beschriftungstext anzeigen möchten. Wenn Sie z. B. Speichern & Schließen anzeigen möchten, müssen Sie Speichern && Schließen im Caption-Eigenschaftensfeld eingeben<sup>vii</sup>.

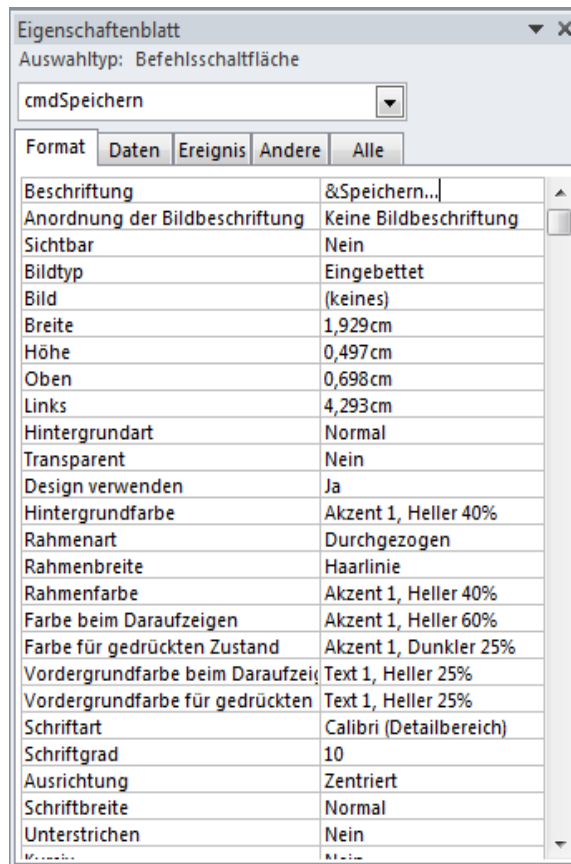



Abb. A1.3.11c Eigenschaftensblatt einer Schaltfläche z.B. cmdSpeichern in frmkategorie

Im Register **Ereignis** (Abb. A1.3.11d) des Eigenschaftensblattes einer Schaltfläche sind alle möglichen Methoden, die auf die ausgeübt werden können, aufgelistet, wie bei Listenfeld im Abschnitt 1.3.10 erklärt wurde.

 Die aktuellen Daten sollen hier beim Klicken auf die Schaltfläche in die Tabelle tblkategorie abgelegt werden. Die VBA-Codes sollen noch den Datensatz auf das mehrfache Vorkommen überprüfen. (Details im Abschnitt 1.7)

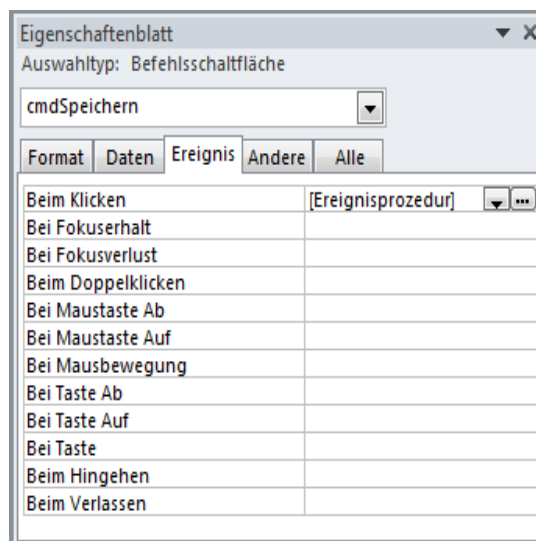



Abb. A1.3.11d Eigenschaftensblatt einer Schaltfläche z.B. cmdSpeichern in frmkategorie

 Auch hier hilft Ihnen Assistent von Access, eine schnelle und einfache Einstellung für eine Schaltfläche durchzuführen.

### 1.3.12 Eigenschaften vom Bild

Das Bild-Objekt bei Microsoft-Office/Entwicklungstools wurde in diesem Programmbeispiel häufig benutzt, um z.B. Bilder für einen Mitarbeiter im Formular frmPersonal, oder auch für eine Artikel im Formular frmArtikel anzuzeigen. Es gibt zwar die Möglichkeit in Office mit dem Datentyp Anlage einfach und komfortabel Bilder einzubeziehen und anzuzeigen, aber erfahrungsgemäß wird die Speicherkapazität von der Access-Datei sehr hoch. In diesem Skript wurden die Bilder mit dem Präfix img (Siehe in diesem Abschnitt Punkt 1.2 Namenkonvention) benannt. Damit die Bilder ganz und zentriert angezeigt werden, sollen Sie die Optionen bei den *Größenanpassung* und *Bildausrichtung* jeweils auf *Zoomen* und *Mitte* einstellen. Zur Bestimmung der Ästhetik (Abb. A1.3.12) achten Sie auf die Einstellungen bei *Breite*, *Höhe*, *Oben*, *Links*, *Hintergrundart*, *Hintergrundfarbe*, *Rahmenarten*, *Rahmenbreite*, *Rahmenfarbe*, *Spezialeffekt*.

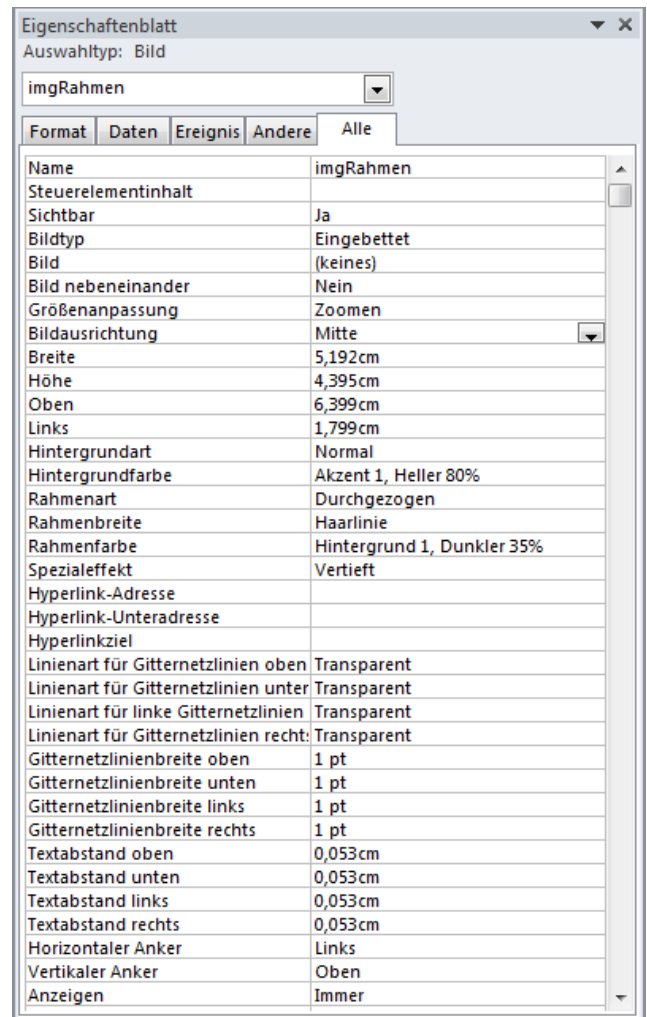


Abb. A1.3.12 Eigenschaftenblatt eines Bildes z.B. imgRahmen in frmArtikel

### 1.3.13 Eigenschaften vom Kontrollkästchen

Das Kontrollkästchen-Objekt bei Microsoft-Office/Entwicklungstools wurde in diesem Programmbeispiel wie bei dem Artikel benutzt, um z.B. zu prüfen, ob ein bestimmter Artikel aus dem Formular frmArtikel ausgelaufen ist oder nicht. In diesem Skript wurden Kontrollkästchen mit dem gleichen Namen wie dem Feldnamen in der Tabelle oder Abfrage benannt. Zur Bestimmung der Ästhetik (Abb. A1.3.13) achten Sie auf die Einstellungen bei *Breite, Höhe, Oben, Links, Rahmenart, Rahmenbreite, Rahmenfarbe, Spezialeffekt*.

Wenn im Laufe des Programmes ein Kontrollkästchen aktiviert oder deaktiviert werden soll, setzen Sie die Einstellung für *Aktiviert* auf *Nein* bei diesem Eigenschaftenblattes. Standard ist dies auf *Ja* eingestellt.

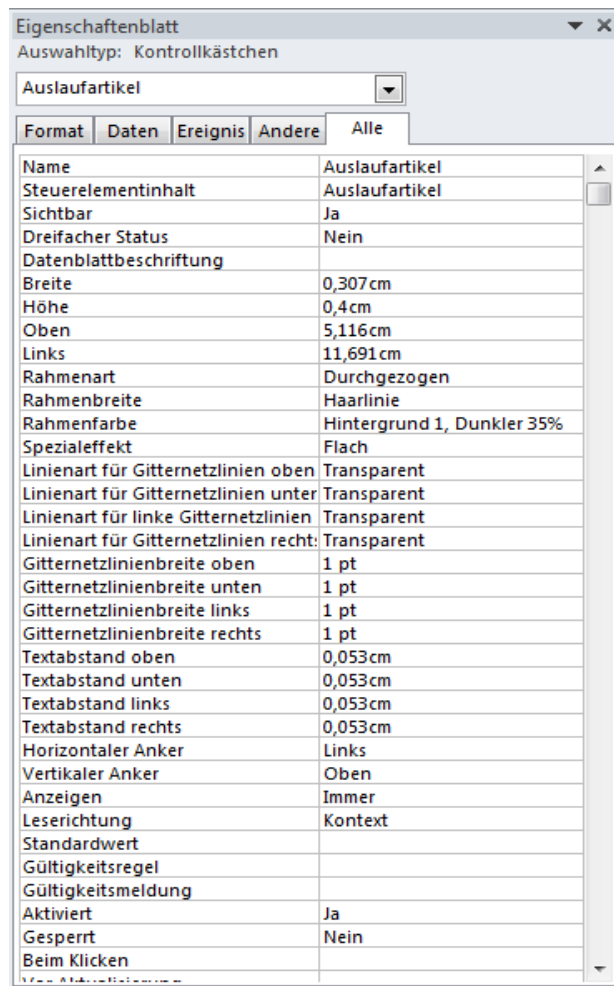


Abb. A1.3.13 Eigenschaftenblatt eines Kontrollkästchens z.B. Auslaufartikel in frmArtikel

### 1.3.14 Eigenschaften vom Optionsfeld

Das Optionsfeld-Objekt bei Microsoft-Office/Entwicklungstools wurde in diesem Programmbeispiel wie bei dem Einkaufen (Auftrag) benutzt, um z.B. einen Auftrag im Formular `frmEinkaufen` zu betätigen. Dabei wird ein Auftrag in jedem Zustand (Status) verfolgt. In diesem Skript wurden die Optionsfelder mit dem Präfix `opt` (Siehe in diesem Abschnitt Punkt 1.1 Namenkonvention) benannt. Zur Bestimmung der Ästhetik (Abb. A1.3.14) achten Sie auf die Einstellungen bei *Breite*, *Höhe*, *Oben*, *Links*, *Rahmenart*, *Rahmenbreite*, *Rahmenfarbe*, *Spezialeffekt*. Wenn im Laufen des Programmes ein Optionsfeld aktiviert oder deaktiviert werden soll, setzen Sie seine Eigenschaft gleich auf `True` oder auf `False`. Wenn Sie ein Ereignis auf das Optionsfeld ausüben wollen, wie in diesem Fall `Beim Klicken`, dann schreiben Sie den gewünschten VBA-Codes unter dem Namen des Optionsfeldes. (Details im Abschnitt 1.7)

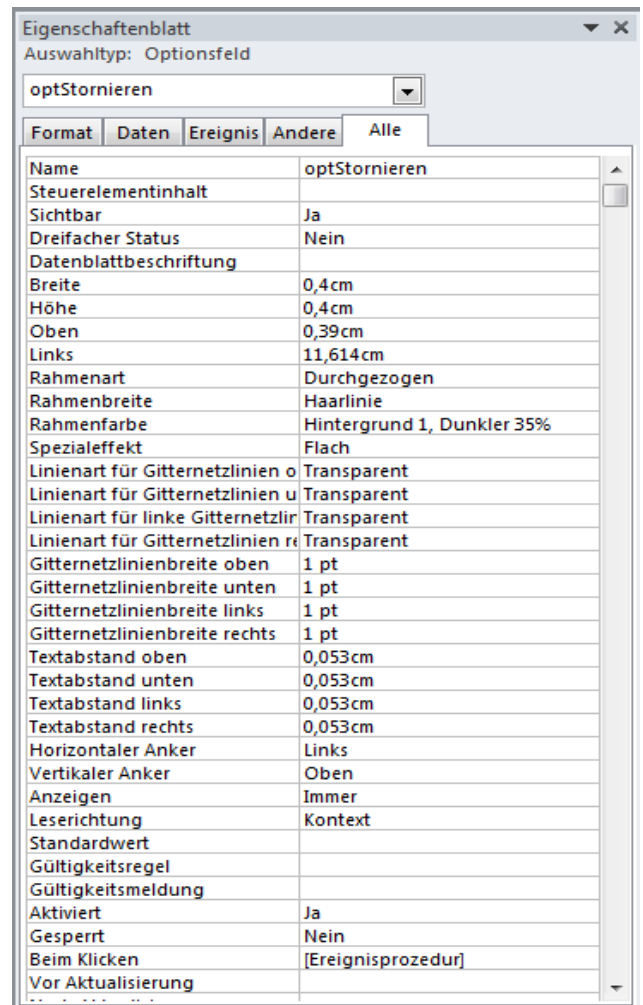


Abb. A1.3.14 Eigenschaftenblatt eines Optionsfeldes z.B. `optOffen` in `frmEinkaufen`



## 1.4 Tabellen- und Datenfeldeigenschaften

### 1.4.1 Normalisierungsformen

In der Praxis ist es ja leider nicht damit getan, irgendwelche Daten in unterschiedliche Tabellen zu packen. Zuerst muss man wissen, welche Daten denn in welche Tabelle gehören ... und das gehorcht logischen „Spielregeln“.

Es gibt 5 - 6 sog. **Normalisierungsformen**<sup>viii</sup>; in DBs sind aber meist nur **drei** üblich und auch ausreichend. Die sind aber in sehr abstrakter Form verfasst und es ist schwierig, etwas damit anzufangen.

Die **Umsetzung** der Regeln ist dann eine ganz andere Sache, die Zeit braucht. Zeit, die man sich unbedingt nehmen sollte. Setzen Sie sich bei einer DB niemals unter Zeitdruck (bzw. lassen Sie sich nicht von anderen unter Zeitdruck setzen). Es rächt sich sonst später bitter!!



**Erste Normalform->Wissenschaftliche Aussage:** Eine Relation R ist in erster Normalform, wenn alle Attribute elementar(atomar) sind.

**Umgangssprachlich (möglicherweise etwas ungenau):** Alle Felder einer Tabelle (wie tblStandort, tblKunden, tblLieferanten, tblKategorie usw.) müssen so sein, dass sie nicht mehr unterteilt werden können. Das unten genannte Beispiel entspricht der ersten Normalform, weil die Attribute alle elementar sind.

Die Relation Einkaufen

ArtikelID	Artikelcode	Kategorie-ID	Kategorienname	Unter-kategorieID	Unterkategorie-name	Unterkategorie-beschreibung
1	HO-BW-089	2	Damenkleidung	1	Hosen	Größe 34-38
2	RO-BW-100	2	Damenkleidung	3	Röcke	Größe 36-46
3	HO-BW-302	3	Herrenkleidung	2	Hosen	Größe 40-48

enthält nur atomare Werte. In jeder Spalte ist nur eine Information enthalten.

Wenn man diese Relation jedoch wie folgt verändert,

ArtikelID	Artikelcode	Kategorie-ID	Kategorienname	Unter-kategorieID	Unterkategorie-name	Unterkategorie-beschreibung
1	HO-BW-089	2,3	Damenkleidung, Herrenkleidung	1	Hosen	Größe 34-38
2	RO-BW-100	2	Damenkleidung	3	Röcke	Größe 36-46
3	HO-BW-302	3,2	Herrenkleidung, Damenkleidung	2	Hosen	Größe 40-48

dann ist die **erste Normalform** verletzt, da die Spalten KategorieID und Kategorienname ggf. mehrere Position und Bezeichnungen enthalten.



**Zweite Normalform->Wissenschaftliche Aussage:** Eine Relation R ist in zweiter Normalform, wenn die erste Normalform vorliegt und alle Nichtschlüsselattribute von jedem Schlüsselkandidaten voll funktional abhängig sind.

Ein **Schlüssel** dient in einer Relationalen Datenbank dazu, eine Tupel (eine Sammlung mit einer beliebigen Anzahl n von Attributen) einer Relation eindeutig zu identifizieren. Anschaulich kann man sich eine Relation als Tabelle vorstellen. Der Schlüssel einer solchen Tabelle ist dann



eine Gruppe von Spalten, die so ausgewählt wird, dass jede Zeile in dieser Gruppe eine einmalige Wertekombination besitzt.

**Umgangssprachlich (möglicherweise etwas ungenau):**

Alle nicht primären Attribute (nicht Teil des Schlüssels) sind vom ganzen Schlüssel abhängig, nicht von nur einem Teil des Schlüssels (Genau das ist die „voll funktionale Abhängigkeit“: Ein Attribut ist vom ganzen Schlüssel abhängig). Die unten genannten Beispiele (tblArtikel, tblEinkaufen usw.) entsprechen der zweiten Normalform, weil z.B. für eine Unterkategorie viele Artikel existieren.

Die Relation Einkaufen(Auftrag)

ArtikelID (PK)	Artikelcode	Kategorie-ID	Kategoriename	Unter-kategorieID (PK)	Unter-kategorienam e	Unterkategorie-beschreibung
1	HO-BW-089	2	Damenkleidung	1	Hosen	Größe 34-38
2	RO-BW-100	2	Damenkleidung	3	Röcke	Größe 36-46
3	HO-BW-302	3	Herrenkleidung	2	Hosen	Größe 40-48

mit dem zusammengesetzten Primärschlüssel (UnterkategorieID, ArtikelID) soll vereinfacht ein Artikelgeschehen abbilden. Es sind folgende volle funktionale Abhängigkeiten gegeben:

1. UnterkategorieID -> Unterkategoriename, KategorieID, Kategoriename
2. ArtikelID -> Bezeichnung
3. UnterkategorieID, ArtikelID -> Unterkategoriebeschreibung
4. KategorieID -> Kategoriename

Die ersten beiden funktionalen Abhängigkeiten hängen nur partiell vom Primärschlüssel ab. Die dritte Abhängigkeit ist voll vom gesamten Primärschlüssel abhängig. Die vierte Abhängigkeit (KategorieID -> Kategoriename) hingegen wird erst für **die Dritte-Normalform** relevant, da es eine transitive Abhängigkeit ist.

Bei der Überführung in die zweite Normalform entstehen folgende Relationen:

**Unterkategorie**

KategorieID	Kategoriename	Unter-kategorieID (PK)	Unter-kategoriename	Unterkategorie-beschreibung
2	Damenkleidung	1	Hosen	Größe 34-38
2	Damenkleidung	3	Röcke	Größe 36-46
3	Herrenkleidung	2	Hosen	Größe 40-48

**Artikel**

ArtikelID(PK)	Artikelcode
1	HO-BW-089
2	RO-BW-100
3	HO-BW-302

**Einkaufen**

ArtikelID (PK, FK)	Artikelcode	KategorieID	Kategorienname	UnterkategorieID (PK, FK)	Unterkategorie- beschreibung
1	HO-BW-089	2	Damenkleidung	1	Größe 34-38
2	RO-BW-100	2	Damenkleidung	3	Größe 36-46
3	HO-BW-302	3	Herrenkleidung	2	Größe 40-48

**PK** = Primärschlüssel (primary key), **FK** = Fremdschlüssel (foreign key)

Die Anzahl der Relationen, die bei einer Überführung in die zweite Normalform entstehen können, hängt von der Anzahl der **n** der Primärschlüsselattribute ab. Dies ist die Potenzmenge einer **n** – Elementen von Teilmenge, also  $2^n - 1$ . Bei einer Relation mit drei Primärschlüsselattributen (**n=3**) können daher bis zu 7 neue Relationen entstehen, bei einer Relation mit 2 Primärschlüsselattributen sind es bis zu 3 neue Relationen, die so entstehen können.



**Dritte Normalform->Wissenschaftliche Aussage:** Eine Relation R ist in dritter Normalform, wenn die erste Normalform und die zweite Normalform vorliegen und alle Nichtschlüsselattribute von jedem Schlüsselkandidaten voll funktional abhängig sind.

Die Überführung in die dritte Normalform erfolgt ähnlich wie die Überführung in die zweite Normalform (2NF), mit dem Unterschied, dass bei der 3NF auch transitive funktionale Abhängigkeiten im Nichtschlüsselattributbereich aufgelöst werden. Dieser Zusammenhang ist in der untenstehenden Erklärung dargestellt. Falls eine Relation nur ein einziges Nichtschlüsselattribut besitzt, können keine transitiven funktionalen Abhängigkeiten auftreten.

**Die Ausgabetable**

Unter- kategorieID	Unterkategorie name	Unterkategorie- beschreibung	Kategorie-ID	Kategorienname
1	Hosen	Größe 34-38	2	Damenkleidung
2	Hosen	Größe 40-48	3	Herrenkleidung
3	Röcke	Größe 36-46	2	Damenkleidung

mit dem Primärschlüssel **UnterkategorieID** hat eine transitive Abhängigkeit: **UnterkategorieID** -> **KategorieID** -> **Kategorienname**

**Überführung in dritter Normalform**

Unter- kategorieID	Unterkategorie name	Unterkategorie- beschreibung	Kategorie-ID
1	Hosen	Größe 34-38	2
2	Hosen	Größe 40-48	3
3	Röcke	Größe 36-46	2

**Kategorie**

Kategorie-ID	Kategorienname
2	Damenkleidung
3	Herrenkleidung

### 1.4.2 Feldeigenschaften in Abhängigkeit und deren Datentypen

Um alle Datentypen in Access-Hilfe anzusehen, können Sie `DataType` in das Feld „Suchende Wörter eingeben“ eingeben, dann erhalten Sie die folgende Grafik (Abb. A1.4.2a):

Einstellung	Datentyp	Größe
Text	(Standardeinstellung) Text oder Kombination von Text und Zahlen sowie Zahlen, die keine Berechnungen erfordern, wie beispielsweise Rufnummern.	Bis zu 255 Zeichen oder eine von der Eigenschaft <b>FieldSize</b> festgelegte Länge (je nachdem, welcher der beiden Werte kleiner ist). Microsoft Access reserviert keinen Platz für nicht verwendete Teile eines Textfelds.
Memo	Sehr langer Text oder Kombination aus Text und Zahlen.	Bis zu 63.999 Zeichen. (Wenn das Memo-Feld mit DAO bearbeitet wird und nur Text und Zahlen [keine binären Daten] darin gespeichert werden, dann ist die Größe des Memo-Felds durch die Größe der Datenbank beschränkt.)
Zahl	Numerische Daten, die in mathematischen Berechnungen verwendet werden. Weitere Informationen zur Einstellung eines bestimmten Zahlentyps finden Sie im Thema der Eigenschaft <b>Feldgröße</b> .	1, 2, 4, oder 8 Byte (16 Byte, wenn die <b>FieldSize</b> -Eigenschaft auf die Replikations-ID festgelegt ist).
Datum/Uhrzeit	Datums- und Zeitwerte für die Jahre 100 bis 9999.	8 Byte.
Währung	Währungswerte und numerische Daten, die in mathematischen Berechnungen verwendet werden und eine bis vier Dezimalstellen enthalten. Auf bis zu 15 Stellen links, und bis zu 4 Stellen rechts vom Dezimaltrennzeichen genau.	8 Byte.
AutoWert	Eine eindeutige fortlaufende (durch 1 erhöhte) Nummer oder durch Microsoft Access bei jedem Hinzufügen eines neuen Datensatzes zu einer Tabelle zugewiesene zufällige Zahl. AutoWert-Felder können nicht aktualisiert werden. Weitere Informationen finden Sie unter dem Thema zur <b>NewValues</b> -Eigenschaft.	4 Byte (16 Byte, wenn die <b>FieldSize</b> -Eigenschaft auf Replikations-ID festgelegt ist).
Ja/Nein	Ja-/Nein-Werte und -Felder, die nur einen von zwei Werten enthalten (Ja/Nein, True/False oder An/Aus).	1 Bit.
OLE-Objekt	Ein Objekt (wie z. B. eine Microsoft Excel-Tabelle, ein Microsoft Word-Dokument, Grafiken, Klänge oder andere binäre Daten), das mit einer Microsoft Access-Tabelle <b>verknüpft</b> oder darin <b>eingebettet</b> ist.	Bis zu 1 Gigabyte (durch verfügbaren Speicherplatz begrenzt)
Hyperlink	Text oder Kombinationen aus Text und Zahlen, die als Text abgespeichert und als <b>Hyperlinkadresse</b> verwendet werden. Eine Hyperlinkadresse besteht aus bis zu vier Teilen: <i>Anzuzeigender Text</i> – der in einem Feld oder Steuerelement angezeigte Text. <i>Adresse</i> . Der Pfad zu einer Datei ( <b>UNC-Pfad</b> ) oder Seite ( <b>URL</b> ). <i>Unteradresse</i> . Eine Stelle innerhalb einer Datei oder Seite. <i>Infofeld</i> . Text, der als Quickinfo eingeblendet wird.	Jeder der Teile eines <b>Hyperlink</b> -Datentyps kann bis zu 2.048 Zeichen enthalten.
Anlage	Jeder unterstützte Dateityp	Sie können den Datensätzen in der Datenbank Bilder, Arbeitsblattdateien, Dokumente, Diagramme und andere unterstützte Dateitypen anfügen, so wie Sie Dateien an E-Mail-Nachrichten anfügen. Sie können angefügte Dateien auch anzeigen und bearbeiten, in Abhängigkeit davon, wie der Datenbankentwickler das Anlagefeld einrichtet. Anlagefelder bieten größere Flexibilität als Felder für OLE-Objekte, und der Speicherplatz wird effizienter verwendet, da kein Bitmapbild der ursprünglichen Datei erstellt wird.
Nachschlage-Assistent	Erstellt ein Feld, mit dem Sie mit einem Listenfeld oder Kombinationsfeld einen Wert aus einer anderen Tabelle oder aus einer Werteliste auswählen können. Durch Klicken auf diese Option wird der Nachschlage-Assistent gestartet, der ein <b>Nachschlagefeld</b> erstellt. Nach dem Beenden des Assistenten wird von Microsoft Access basierend auf den im Assistenten ausgewählten Werten der Datentyp festgelegt.	Die gleiche Größe wie das Feld <b>Primärschlüssel</b> , das zum Nachschlagen verwendet wird, typischerweise 4 Bytes.

Abb. A1.4.2a `DataType`-Eigenschaft in Access

Die Tabelle `tblArtikel` beinhaltet fast alle Felddatentypen, die hier in diesem Programmbeispiel angewandt wurden. Daher wird jeder Feldname in den unteren Abbildungen beschrieben. Zur Bestimmung der Feldeigenschaften (Abb. A1.4.2b) achten Sie auf die Einstellungen

- **Feldgröße**: Speichert zwischen 2.147.483.648 und 2.147.483.647 Datensätze.
- **Neue Werte**: Inkrement oder Zufall
- **Indiziert**: Ja (ohne Duplikat), Ja (mit Duplikat) oder Nein



Die Namenkonventionen für die Schlüsselattribute wurden mit dem

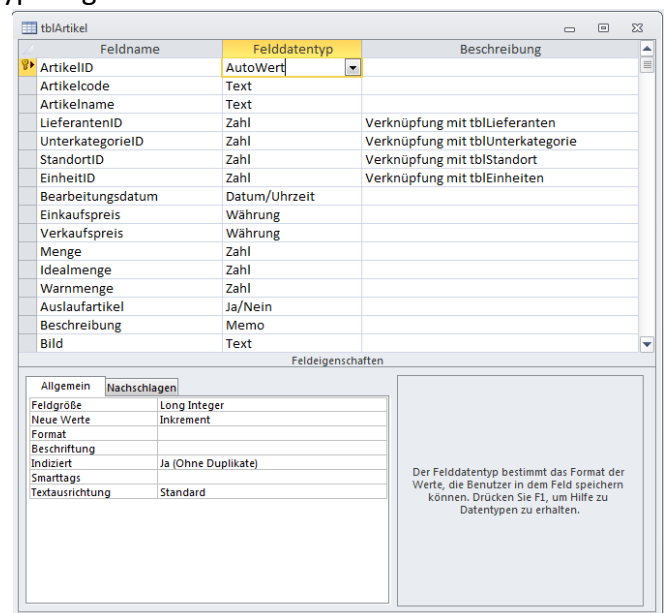


Abb. A1.4.2b Feldeigenschaften von `AutoWert`

Suffix ID (wie z.B. ArtikelID) vorgegeben.  
 Zur Bestimmung der Feldeigenschaften (Abb. A1.4.2c) achten Sie auf die Einstellungen

- **Feldgröße:** reserviert bis maximal 255 Zeichen für die Inhalte. Unnötige Zeichenreservierung vergrößert die Kapazität der Datenbank und folglich sie verlangsamt die Ausführungsprozessen.
- **Eingabe erforderlich:** Ja oder Nein. Wenn Sie Ja wählen, wird eine Eingabe verlangt, ansonsten wird eine Fehlermeldung angezeigt.
- **Indiziert:** Ja (ohne Duplikat), Ja(mit Duplikat) oder Nein
- **Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.**

Der Feldname LieferantenID wird als Zahl festgelegt, weil er mit dem Schlüsselattribut LieferantenID der Tabelle tblLieferanten verknüpft werden soll. Zur Bestimmung der Feldeigenschaften (Abb. A1.4.2d) können Sie u.a.

- **Feldgröße:** wie bei der Abb. A1.4.2b.
- **Eingabe erforderlich:** Wie bei der Abb. A1.4.2c
- **Indiziert:** Wie bei der Abb. A1.4.2c
- **Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.**

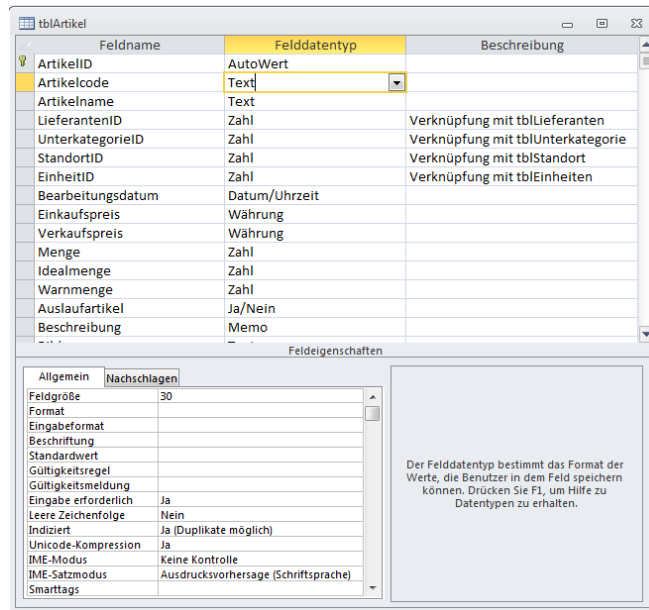


Abb. A1.4.2c Feldeigenschaften von Text

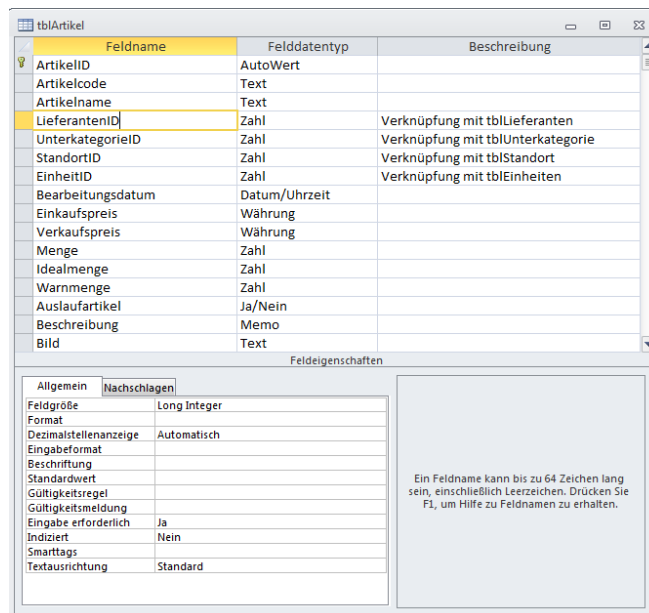


Abb. A1.4.2d Feldeigenschaften von Zahl

Der Feldname `Bearbeitungsdatum` wird als Datum/Uhrzeit festgelegt. Zur Bestimmung der Feldeigenschaften (Abb. A1.4.2e) können Sie u.a.

- **Format:** Datumformat festlegen. (Weitere Details unter Format-Eigenschaft - Datentyp "Datum/Uhrzeit" in Microsoft-Hilfe)
- **Eingabeformat:** Darstellungsformat festlegen. (Weitere Details unter „TextBox.InputMask“ in Microsoft-Hilfe)
- **Standardwert:** Funktionen in Ausdrucks-Generator eingeben.
- **Gültigkeitsregel:** Funktionen in Ausdrucks-Generator eingeben.
- **Gültigkeitsmeldung:** bei einem Fehler eine Meldung ausgeben.
- **Indiziert:** Wie bei der Abb. A1.4.2c
- **Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.**

Der Feldname `Verkaufspreis` wird als Währung festgelegt. Zur Bestimmung der Feldeigenschaften (Abb. A1.4.2f) können Sie u.a.

- **Format:** Währungsformat festlegen. (Weitere Details unter Format-Eigenschaft - Datentyp "Währung" in Microsoft-Hilfe)
- **Eingabeformat:** Darstellungsformat festlegen. (Weitere Details unter „TextBox.InputMask“ in Microsoft-Hilfe)
- **Standardwert:** Funktionen in Ausdrucks-Generator eingeben.
- **Gültigkeitsregel:** Funktionen in Ausdrucks-Generator eingeben.
- **Gültigkeitsmeldung:** bei einem Fehler eine Meldung ausgeben.
- **Indiziert:** Wie bei der Abb. A1.4.2c
- **Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.**

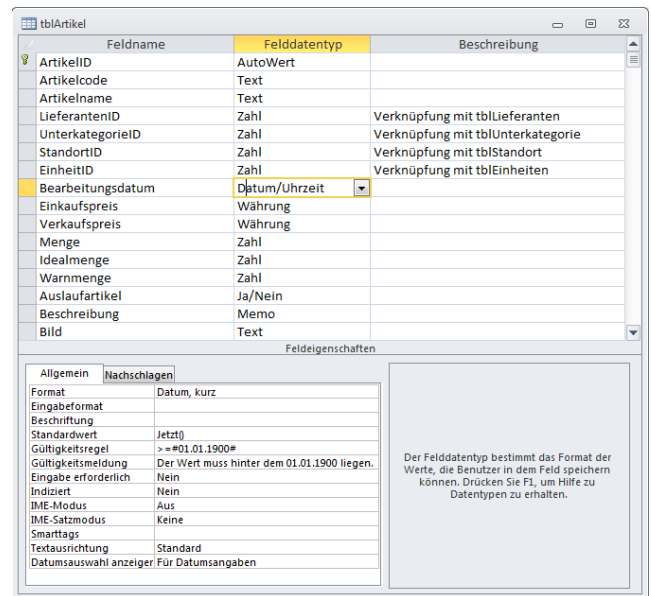


Abb. A1.4.2e Feldeigenschaften von Datum/Uhrzeit

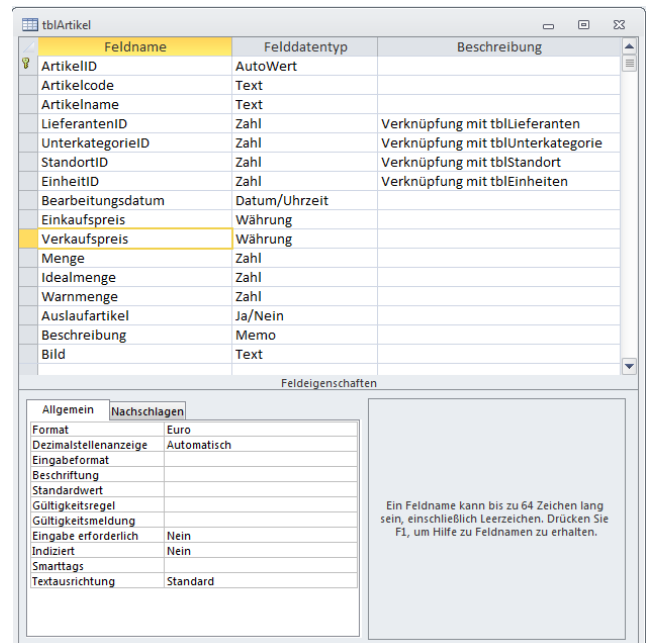


Abb. A1.4.2f Feldeigenschaften von Währung

Der Feldname `Auslaufartikel` wird als Wahr/Falsch festgelegt. Zur Bestimmung der Feldeigenschaften (Abb. A1.4.2g) können Sie u.a.

- **Format:** Ja/Nein festlegen. (Weitere Details unter Format-Eigenschaft - Datentyp "Ja/Nein" in Microsoft-Hilfe)
- **Standardwert:** Funktionen in Ausdrucks-Generator eingeben.
- **Gültigkeitsregel:** Funktionen in Ausdrucks-Generator eingeben.
- **Gültigkeitsmeldung:** bei einem Fehler eine Meldung ausgeben.
- **Indiziert:** Wie bei der Abb. A1.4.2c
- **Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.**

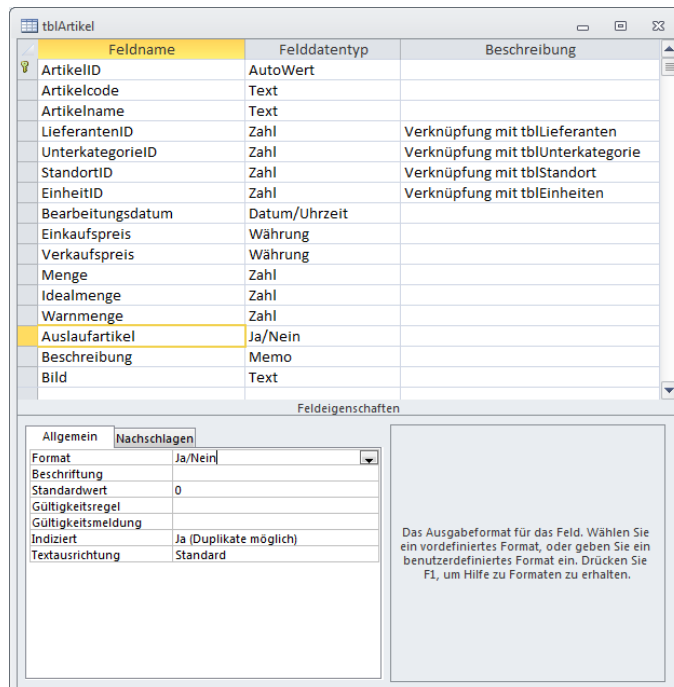


Abb. A1.4.2g Feldeigenschaften von Ja/Nein

Der Feldname `Memo` wird für einen langen Text festgelegt. Zur Bestimmung der Feldeigenschaften (Abb. A1.4.2h) können Sie u.a.

- **Format:** ein benutzerdefiniertes Format festlegen. (Weitere Details unter Format-Eigenschaft - Datentyp "Memo" in Microsoft-Hilfe)
- **Standardwert, Gültigkeitsregel, Gültigkeitsmeldung, Eingabe erforderlich, Leere Zeichenfolge, Indiziert:** wie bei der Abb. A1.4.2a-c festlegen.
- **Andere vom System vorgegebenen Einstellungen wurden genauso übernommen.**

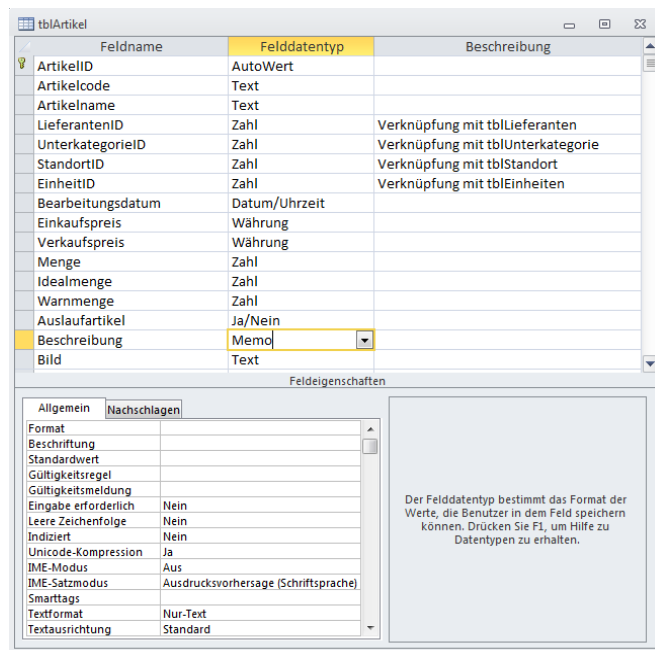


Abb. A1.4.2h Feldeigenschaften von Memo

### 1.4.3 Tabellenbeziehung für Einkaufen (Auftrag) und Verkaufen (Bestellung)

Das Schema zum Einkaufen (Abb. A1.4.3a) von Artikeln zeigt die Beziehungen aller notwendigen Tabellen, die bei einem Auftrag zum Einsatz kommen können. Eine eindeutige Beziehung muss bei der Konzeption eines Datenbanksystems berücksichtigt werden. In unserem Programmbeispiel sind Mitarbeiter der Unternehmen in verschiedenen Standorten beschäftigt. Also daher werden zwei Tabellen gebraucht, um die Personen eindeutig von jeweiligen Standorten zu identifizieren. Zur Datenerfassung soll jede Tabelle bis auf die Tabelle `tblEinkaufsstatus` ein Formular und je nach Bedarf einen Bericht besitzen. In diesem Programmbeispiel wurde auf Makros verzichtet und alle Steuerungen mit VBA-Codes (Details im Abschnitt 1.7) erreicht. Ein fast gleiches Schema ergibt auch beim Beziehungsschema für den Artikelverkauf (Abb. A1.4.3b). Der Status eines Auftrages (Einkauf) oder einer Bestellung (verkauf) hängt davon ab, wie das überwacht und ausgewertet werden soll. Ein- und Verkaufsstatus wurden in diesem Programmbeispiel wie folgt zusammengefasst:

tblEinkaufsstatus		
Ein	Status	Zum Hinzufügen klicken
0	Offen	
1	Genehmigt	
2	Gekauft	
3	Stornieren	
4	Kein Bestand	
*		

tblVerkaufsstatus		
Verk	Status	Zum Hinzufügen klicke
0	Neu	
1	Verkauft	
2	Versendet	
3	Stornieren	
4	kein Bestand	
*		



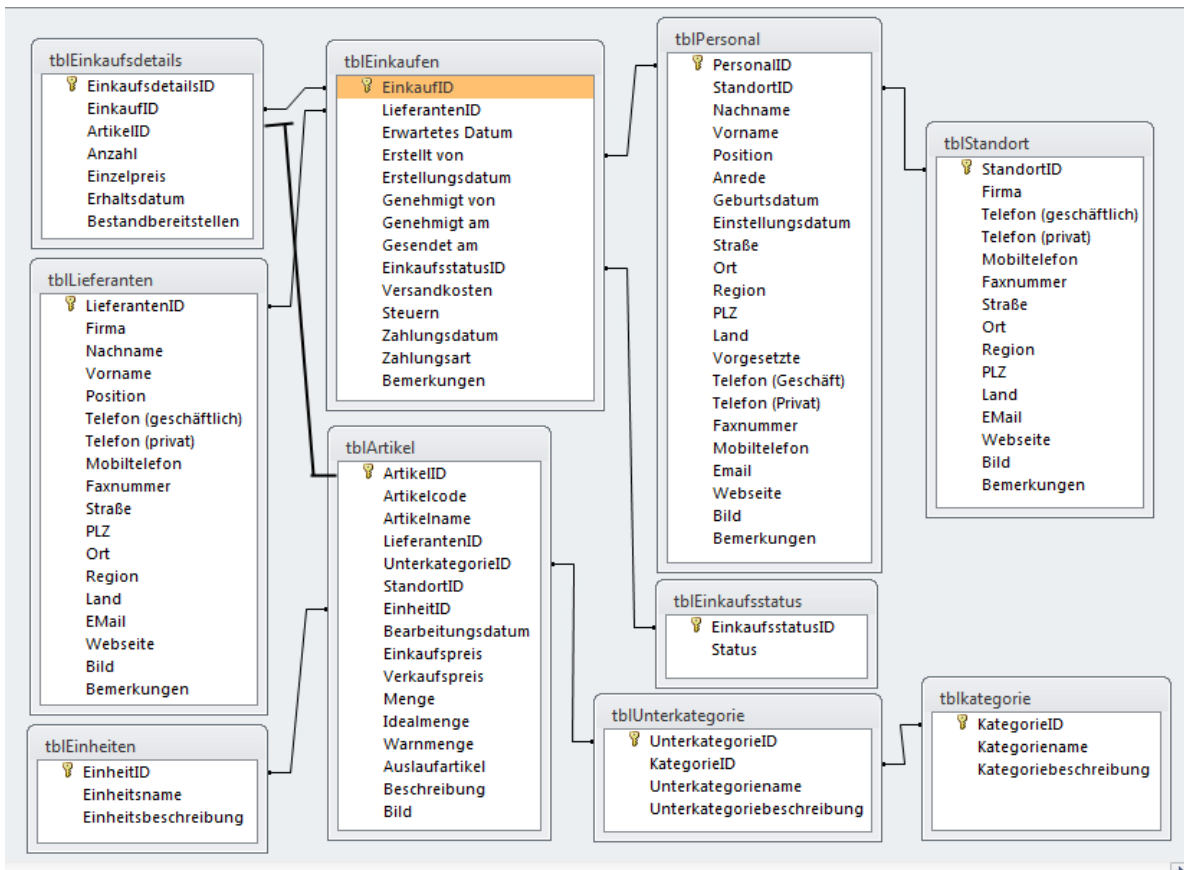


Abb. A1.4.3a Das Beziehungsschema für den Artikeleinkauf

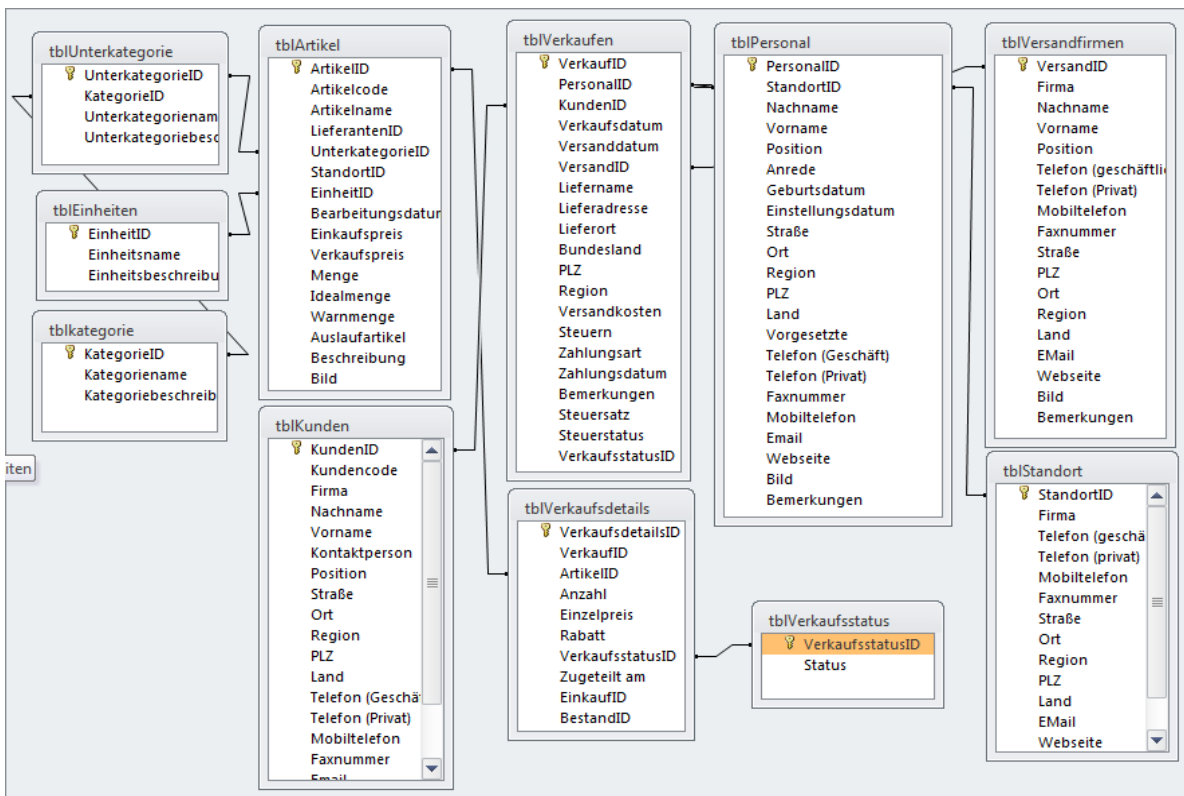


Abb. A1.4.3b Das Beziehungsschema für den Artikelverkauf



## 1.5 Erstellung von Formularen oder Berichten

Die Erstellung von Formularen und Berichten stellt Access gleiches Prinzip. In diesem Programmbeispiel wurden alle Formulare mit gleicher Ästhetik, wie sie schon in Abschnitt 1.3 erläutert wurde, angewandt. Das Formular frmArtikel ist stellvertretend für alle anderen Formulare, die zu der Verwaltung der Stammdaten (frmPersonal, frmLieferanten, frmKunden, frmVerandfirmen, frmStandort, usw.) dienen.

### 1.5.1 Einstellungen und Festlegung der Eigenschaften im Formular frmArtikel

Das Formular frmArtikel ist mit der Tabelle tblArtikel verknüpft. Damit können alle Feldnamen von der Tabelle mit dem Formular verbunden werden. Sollte man einen Inhalt von den Textfeldern, Kontrollkästchen, Optionsfeldern oder Kombinationsfeldern in dem Formular verändern, werden diese in der Tabelle genauso verändert, soweit deren Steuerelementinhalt auf dem Feldnamen der Tabelle verwiesen werden. In der folgenden Auflistung werden die wichtigen Merkmale der

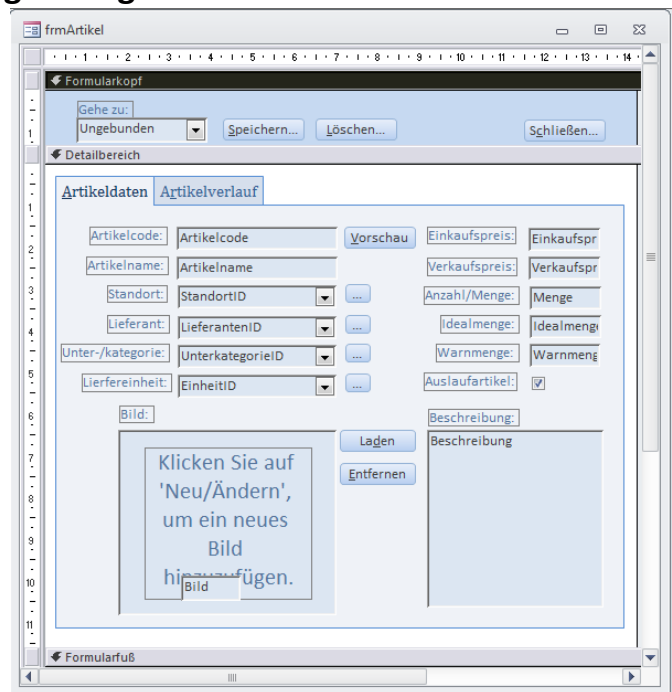


Abb. A1.5.1 Entwurfsansicht von frmArtikel

Steuerelementen für das Formular frmArtikel zusammengestellt:

#### a) Angewandte Schaltflächen

Pos.	Steuer- element	Name	Beschriftung	Aktionen
1	Schaltfläche	cmdSpeichern	&Speichern...	Beim Klicken
2	Schaltfläche	cmdLoeschen	&Löschen...	Beim Klicken
3	Schaltfläche	cmdSchliessen	S&chließen...	Beim Klicken
4	Schaltfläche	cmdAnzeigen	&Vorschau	Beim Klicken
5	Schaltfläche	cmdstandortoeffnen	...	Beim Klicken
6	Schaltfläche	cmdLieferantenoeffnen	...	Beim Klicken
7	Schaltfläche	cmdUnterkategorieoeffnen	...	Beim Klicken
8	Schaltfläche	cmdEinheitoeffnen	...	Beim Klicken
9	Schaltfläche	cmdladen	La&den	Beim Klicken
10	Schaltfläche	cmdEntfernen	&Entfernen	Beim Klicken

b) Angewandte Textfelder und das Bild

Pos.	Steuerelement	Name	Inhalt	Besonderes	Aktionen
1	Textfeld	Artikelcode	Artikelcode		Beim Verlassen
2	Textfeld	Einkaufspreis	Einkaufspreis	EURO Format	
3	Textfeld	Artikelname	Artikelname		
4	Textfeld	Verkaufspreis	Verkaufspreis	EURO Format	
5	Textfeld	Menge	Menge		
6	Textfeld	Idealmenge	Idealmenge		
7	Textfeld	Warnmenge	Warnmenge		
8	Textfeld	txtBildpfad	Bild		Nach Aktualisierung
9	Textfeld	Beschreibung	Beschreibung	Bildlaufleiste->Vertikal	
10	Bild	imgRahmen			
11	Kontrollkästchen	Auslaufartikel	Auslaufartikel		

c) Angewandte Kombinationsfelder

Pos.	Name	Inhalt	Datensatz-herkunft	Spalten-anzahl	Spaltenbreiten in cm	Aktionen
1	cboArtikel	-	sqlArtikelliste	5	0;4;3;3;0	Beim Klicken
2	StandortID	StandortID	sqlStandortliste	4	0;4;3;3	
3	LieferantenID	LieferantenID	sqlLieferantenliste	4	0;4;3;3;3;0;0	
4	UnterkategorieID	UnterkategorieID	sqlUnterkategorie-fuerKategorie	5	0;4;0;3;5	
5	EinheitID	EinheitID	sqlEinheitenliste	3	0;4;5	

### 1.5.2 Datenverarbeitung mit dem Formular frmArtikel

Beim Start des Formulars, beim Speichern oder beim Löschen eines neuen Datensatzes soll das Layout von den Formularen bzw. von dem Formular gemäß frmArtikel (Abb. 1.5.2a) aussehen. Alle Felder sollen leer sein und die Schaltflächen Speichern und Löschen ausgeblendet sein. Sollte ein verfügbarer Datensatz gesucht bzw. aufgelistet werden, muss dies über dem Kombinationsfeld cboArtikel geschehen.

Sollte ein neuer Datensatz eingegeben werden, müssen die Schaltflächen, Speichern und Löschen eingeblendet werden.

Die Bilder müssen entweder im JPG- oder BMP-Format in einem Unterordner gespeichert sein, wo die Access-Datei sich befindet. Sie können über die Schaltfläche Laden geholt über das Steuerelement imgRahmen angezeigt werden. Sollte ein vorhandenes Bild nicht mehr angezeigt werden, kann das über die Schaltfläche Entfernen ausgeblendet werden. Diese Methode wurde verwendet, damit die Access-Datei möglichst kleine Speicherkapazität verbraucht und die Datenbank schneller auf die Datensätze agiert. Die Details zu VBA-Codes finden Sie im Abschnitt 1.7.

The screenshot shows the 'Stammdaten für Artikel' form. At the top, there is a 'Gehe zu:' dropdown menu and a 'Schließen...' button. Below this, there are two tabs: 'Artikeldaten' and 'Artikelverlauf'. The 'Artikeldaten' tab is active. The form contains several input fields: 'Artikelcode', 'Artikelname', 'Standort', 'Lieferant', 'Unter-/kategorie', and 'Liefereinheit'. To the right of these fields are 'Vorschau' and 'Laden' buttons. Further right are fields for 'Einkaufspreis', 'Verkaufspreis', 'Anzahl/Menge', 'Idealmenge', 'Warnmenge', and 'Auslaufartikel'. At the bottom, there is a 'Bild:' area with 'Laden' and 'Entfernen' buttons, and a 'Beschreibung:' area.

Abb. A1.5.2a Öffnen von frmArtikel

The screenshot shows the 'Stammdaten für Artikel' form with data entered. At the top, there is a 'Gehe zu:' dropdown menu showing 'HO-BW-089-301' and buttons for 'Speichern...', 'Löschen...', and 'Schließen...'. Below this, there are two tabs: 'Artikeldaten' and 'Artikelverlauf'. The 'Artikeldaten' tab is active. The form contains several input fields: 'Artikelcode' (HO-BW-089-301), 'Artikelname' (Hose1), 'Standort' (BTMtec Hamburg), 'Lieferant' (s.Oliver Bernd Freier G), 'Unter-/kategorie' (Damenkleidung), and 'Liefereinheit' (Stück). To the right of these fields are 'Vorschau' and 'Laden' buttons. Further right are fields for 'Einkaufspreis' (28,00 €), 'Verkaufspreis' (43,00 €), 'Anzahl/Menge' (20), 'Idealmenge' (25), 'Warnmenge' (5), and 'Auslaufartikel'. At the bottom, there is a 'Bild:' area showing a preview of a pair of trousers, with 'Laden' and 'Entfernen' buttons, and a 'Beschreibung:' area.

Abb. A1.5.2b Öffnen von frmArtikel

### 1.5.3 Einstellungen und Festlegung der Eigenschaften im Bericht rptArtikel

Der Bericht rptArtikel ist mit der Tabelle tblArtikel verknüpft. Damit können alle Feldnamen von der Tabelle mit dem Bericht verbunden werden. D.h. alle Daten werden eins zu eins von der Tabelle in den Bericht übertragen. Wie bei den Formularen können alle Steuerelemente in einem Bericht genauso festgelegt werden. Der Pfad für das Bild wurde hier unsichtbar festgelegt.

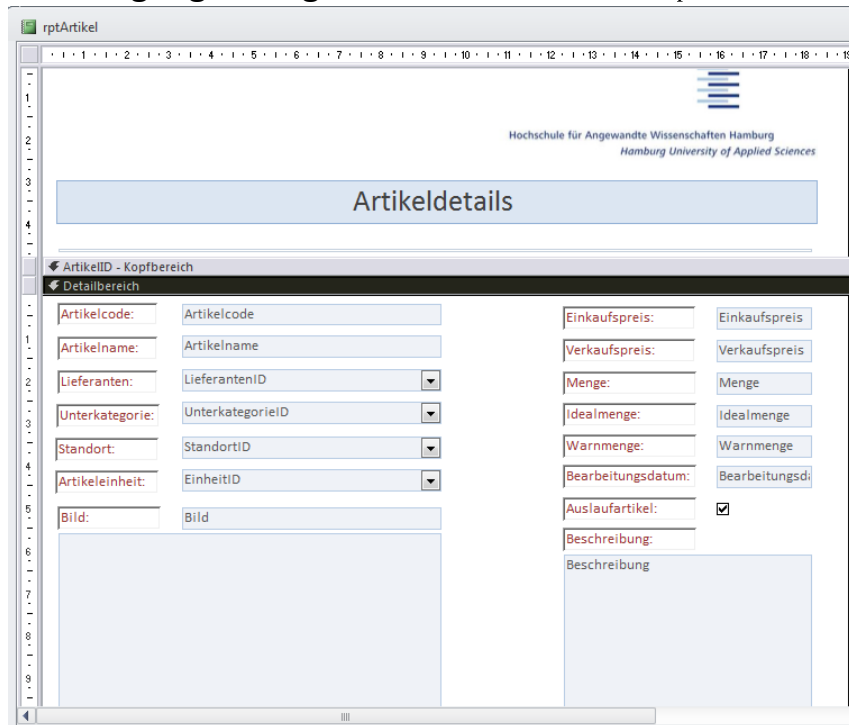


Abb. A1.5.3 Entwurfsansicht von rptArtikel

### 1.5.4 Datenverarbeitung mit dem Bericht rptArtikel

Beim Start des Berichtes, je nachdem wie die Daten strukturiert sind, werden die Daten von der Tabelle in den Bericht übertragen. In diesem Report soll ein vollständiger Datensatz mit dem entsprechenden Bild auf einer DIN-A4 Seite erscheinen. Damit das Bild gleichzeitig mit dem Datensatz angezeigt werden kann, muss der Pfad des Bildes im Detailbereich des Berichtes über die Aktion Beim Formatieren aufgerufen werden. Die Details zu VBA-Codes finden Sie im Abschnitt 1.7.

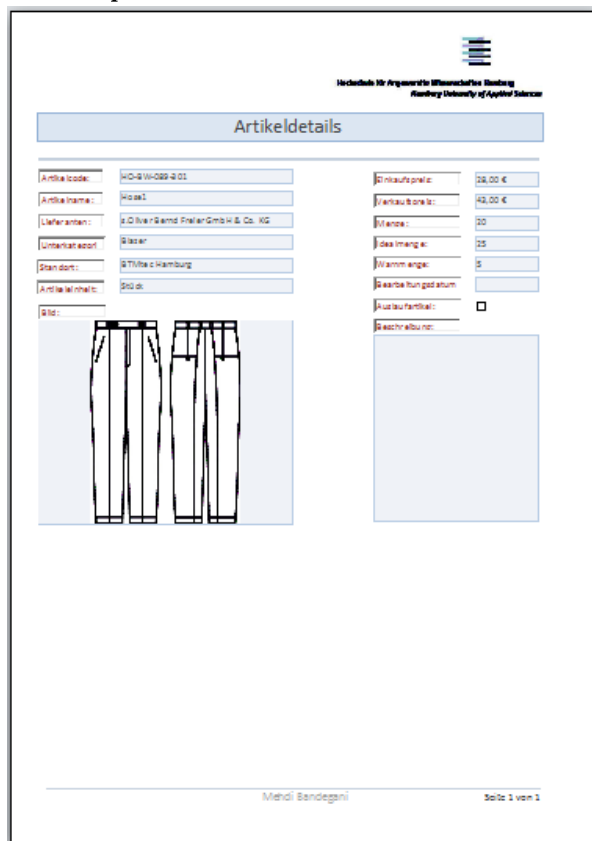


Abb. A1.5.4 Öffnen von rptArtikel

## 1.6 Erstellung und Einsatz von Abfrage

Eine Abfrage dient dazu, aus den Tabellen einer Datenbank ausgewählte Felder anzusehen und gegebenenfalls zu ändern. Man kann also Spalten angeben, die in die Abfrage aufgenommen werden sollen. Außerdem können Kriterien angegeben werden, nach denen die anzuzeigenden Datensätze ausgewählt werden. Dies erzeugt also eine Auswahl unter den Zeilen der Tabellen. Eine Abfrage stellt somit einen Ausschnitt aus der Datenbank dar, der entweder direkt als Datenblatt betrachtet oder als Grundlage für einen Bericht oder ein Formular benutzt werden kann.

Wenn Sie in diesem Datenbanksystem die Artikelreihe, die ausgelaufen worden sind, auflisten wollen, können Sie mit einer Abfrage nach dem Schlüsselattribut `ArtikelID` suchen, die dabei Auslaufartikel auf `Ja` aktiviert.

Wie es im Abschnitt 1.5.1 Punkt c aufgelistet wurden, wurden 5 Abfragen für die Kombinationsfelder zusammengestellt. Sie wurden ohne SQL-Anweisungen über die Datenbankobjekte erstellt.

Wie alle Datenbankobjekte, werden in Microsoft Access auch neue Abfragen mit *Abfrageentwurf* angelegt. Das sollte auch hier wieder über die *Entwurfsansicht* geschehen. Bei vorhandenen Abfragen kommt man mit *Entwurf* wieder zur Entwurfsansicht und mit *Öffnen* zur Datenblattansicht. Außerdem gibt es in einer geöffneten Abfrage einen Button **SQL**, mit dem man zur SQL-Ansicht gelangt. Beim Erstellen einer Abfrage öffnet sich das Fenster *Tabelle anzeigen*, mit dem man die als Datenquelle(n) benötigten Tabellen oder Abfragen hinzufügen kann. Dieses Fenster kann man natürlich auch nachträglich öffnen.

### 1.6.1 Abfrage der Tabelle tblStandort

Im oberen Teil des Entwurfssfensters sieht man die mit *Tabelle anzeigen* ausgewählten Tabellen und Abfragen. Klickt man in die Zeile *Feld*, öffnet sich eine Auswahlliste mit allen Feldnamen, die in den Datenquellen vorhanden sind. Wir wählen jetzt einmal `StandortID` aus.

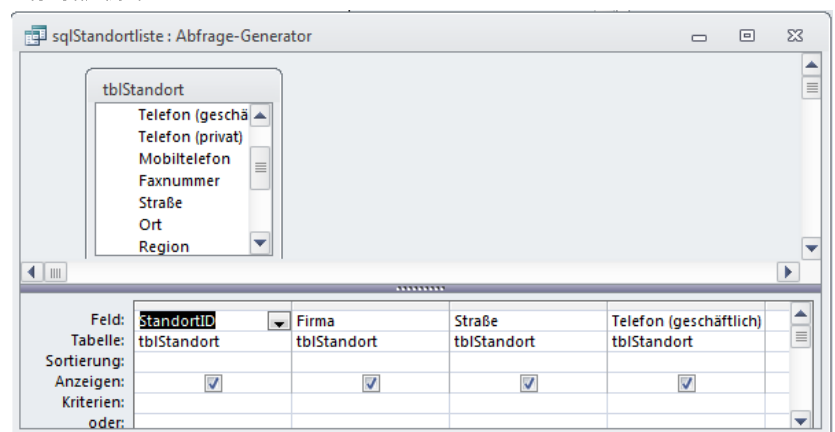


Abb. A1.6.1 Entwurfsansicht von `sqlStandortliste`



Man könnte auch aus der Feldliste im oberen Teil des Fensters einen Feldnamen herunterziehen.

Die Reihenfolge der Spalten in der Abfrage ist also abhängig von der Reihenfolge, in der sie in der Entwurfsansicht angeordnet sind. Als Ergebnis der Abfrage (Abb. A1.6.1a) wird die Auflistung (Abb. A1.6.1b) geliefert.

StandortID	Firma	Straße	Telefon (geschäftlich)
1	BTMtec Adi-Dassler-Strasse	Adi-Dassler-Strasse 1	+49 (0) 9132 84-0
2	BTMtect International Marketing	Hoogoorddreef 9a	+31 (0) 20 573 4573
3	BTMtec Europa	Hoogoorddreef 9a	+31 (0) 20 573 4573
4	BTMtec Hamburg	Hamburger Straße 24	+49(40)6963456
*	(Neu)		

Abb. A1.6.1b Öffnen von sqlStandortliste

### 1.6.2 Abfrage der Tabelle tblArtikelliste

In dieser Abfrage (Abb. 1.6.2a) sollen alle Artikel mit den dazugehörigen Lieferanten angezeigt werden. Bei den Artikeln sollen die Felder der Tabelle tblArtikel (ArtikelID, Artikelcode und Artikelname) und der Tabelle tblLieferanten (Firma und LieferantenID) abgefragt werden.

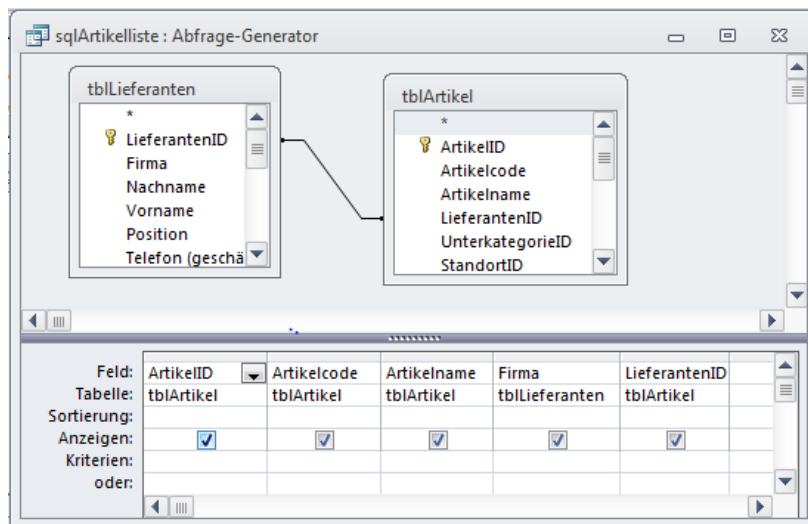


Abb. A1.6.2a Entwurfsansicht von sqlArtikelliste

Als Ergebnis der Abfrage (Abb. A1.6.2a) wird die Auflistung (Abb. A1.6.2b) geliefert.

ArtikelID	Artikelcode	Artikelname	Firma	LieferantenID
1	HO-BW-089-301	Hose1	s.Oliver Bernd F	3
2	HM-WO-026-902	Hemd1	ESCADA SE	1
*	(Neu)			

Abb. A1.6.2b Öffnen von sqlArtikelliste

## 1.7 Steuerung des Programmablaufs

Visual Basic for Applications ist eine Makrosprache, die die Steuerung von Programmabläufen (Literatur: Fußnote VI Seiten 56-101) in Microsoft Office ermöglicht. Vor der Entwicklung und Implementierung des Codes wird empfohlen, einen Programmablauf (Abb. A1.7.1) zu jedem Modul zu erstellen. Dieser verschafft Ihnen eine bessere Übersicht und ermöglicht folglich auch in fehlerhaften Zuständen schnell den Ablauf festzulegen bzw. zu ändern.

### 1.7.1 Programmablauf z.B. zur Artikelbearbeitung

- 1) Wenn das Formular `frmArtikel` gestartet wird, tritt das Ereignis `Private Sub Form_Load()` ein. In diesem Fall soll der letzte Datensatz der Tabelle `tblArtikel` geholt werden und in diesem Formular angezeigt werden. Der letzte Satz ist freilich leer. Außerdem sollen die Schaltflächen `Speichern` und `Löschen` ausgeblendet werden.
- 2) Wenn der Anwender einen vorhandenen Datensatz über das Kombinationsfeld `cboArtikel` sucht und wählt, dann soll der Datensatz aus der Tabelle geholt und in dem Formular angezeigt werden. In diesem Fall sollen die Schaltflächen `Speichern` und `Löschen` eingeblendet werden.
- 3) Wenn der Anwender den Punkt 2) nicht ausführt, sondern einen Eintrag im Feld `Artikelcode` vornimmt, soll das Programm als einen neuen Datensatz erkannt werden. In diesem Fall sollen auch die Schaltflächen `Speichern` und `Löschen` eingeblendet werden.
- 4) Bei den Punkten 2) und 3) soll der Anwender mit den Schaltflächen `Speichern` und `Löschen` ermöglichen, jeweils einen Datensatz über das Formular in der Tabelle abzulegen oder aus der Tabelle zu entfernen. Das entsprechende Bild zu dem Datensatz soll auch im Formular angezeigt bzw. entfernt werden.
- 5) Mit den Schaltflächen `Laden` und `Entfernen` soll ermöglicht werden, den Pfad des Bildes jeweils in die Tabelle abzulegen bzw. aus der Tabelle zu entfernen und gleichzeitig das Bild im Formular anzuzeigen bzw. zu entfernen.

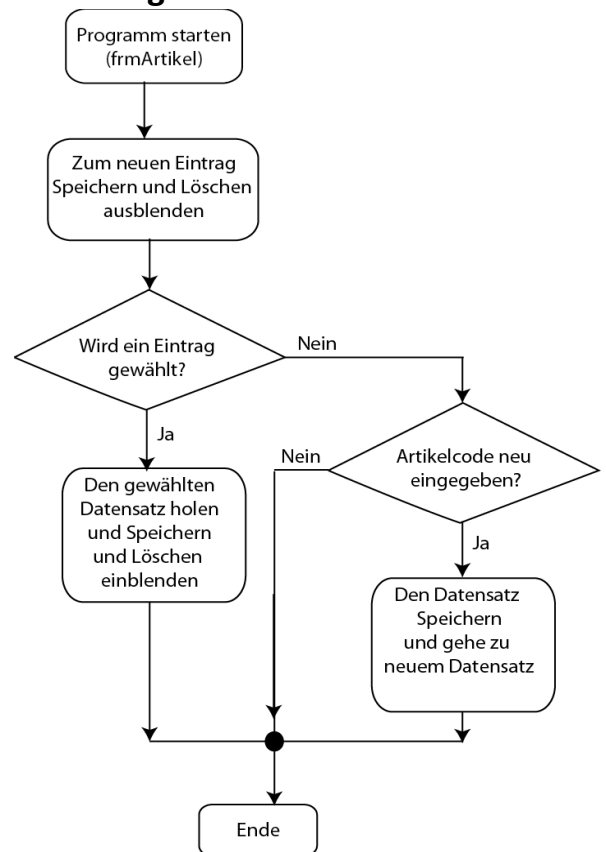


Abb. A1.7.1 Programmablauf für `frmArtikel`

### 1.7.2 Codierung des Programmablaufs für das Formular frmArtikel

Die Codes für die Steuerelemente eines Formulars werden in der Ansicht bzw. Modulansicht erstellt. Wenn ein Steuerelement ein Ereignis bekommen soll, bezieht es sich auf das Modul des Formulars. Für ein Ereignis auf einem Steuerelement klicken Sie im Eigenschaftenblatt auf Code Generator, darauf können die Anweisung für das Ereignis eingetippt werden. Damit die Anweisungen für ein Steuerelement kontrollierbar werden, wurde dabei die folgende Struktur festgelegt:

```
Private Sub Objektname_Ereignis()
On Error GoTo Zu_Fehler

    Anweisungen

Exit_Fehler:
Exit Sub
Zu_Fehler:
    MsgBox Err.Description, vbInformation, "Fehler"
    Resume Exit_Fehler
End Sub
```

### 1.7.3 Ereignisse auf die Objekte im Formular frmArtikel

Die globale Deklarationen lauten:

```
Option Compare Database
Option Explicit
Dim Path As String
```

Und die Ereignisse, die u.a. im Abschnitt 1.5.1 dargestellt wurden, lauten:

#### 1) Das Ereignis *Beim Anzeigen* für das Formular frmArtikel

```
Private Sub Form_Current()
    Dim res As Boolean
    Dim fName As String
    Path = CurrentProject.Path
    On Error Resume Next
        lblMeldung.Visible = False

    If Not IsNull(Me!Bild) Then
        res = IsRelative(Me!Bild)
        fName = Me![txtBildpfad]
        If (res = True) Then
            fName = Path & fName
        End If
        Me![imgRahmen].Picture = fName
        Me![imgRahmen].Visible = True
        Me.PaintPalette = Me![imgRahmen].ObjectPalette
        If (Me![imgRahmen].Picture <> fName) Then
            Me![imgRahmen].Visible = False
            lblMeldung.Caption = "Bild wurde nicht gefunden."
            lblMeldung.Visible = True
        End If
    Else
        Me![imgRahmen].Visible = True
        lblMeldung.Visible = True
        lblMeldung.Caption = "Klicken Sie auf 'Laden', _
            um ein neues Bild hinzuzufügen."
```



```

        End If
    End Sub

```

## 2) Das Ereignis *Beim Laden* für das Formular frmArtikel

```

Private Sub Form_Load()
On Error GoTo Zu_Fehler
    DoCmd.GoToRecord , , acNewRec
    Me.cmdSpeichern.Visible = False
    Me.cmdLoeschen.Visible = False
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Err.Description, vbInformation, "Fehler"
    Resume Exit_Fehler
End Sub

```

## 3) Das Ereignis *Nach Aktualisierung* für das Formular frmArtikel

```

Private Sub Form_AfterUpdate()
    Me("Artikelcode").Requery
    On Error Resume Next
    Rem showErrorMessage und IsRelative müssen existieren!
    showErrorMessage
    Me![imgRahmen].Visible = True
    If (IsRelative(Me!txtBildpfad) = True) Then
        Me![imgRahmen].Picture = Path & Me![txtBildpfad]
    Else
        Me![imgRahmen].Picture = Me![txtBildpfad]
    End If
End Sub

```

## 4) Das Ereignis *Beim Klicken* für das Kombinationsfeld cboArtikel

```

Private Sub cboArtikel_Click()
On Error GoTo Zu_Fehler
    Dim SuchID As Long, SuchCtl As String
    SuchID = Me!cboArtikel.ListIndex + 1
    SuchCtl = Me!cboArtikel.Column(1)
    If DLookup("[ArtikelID]", "tblArtikel", "Artikelcode LIKE '" & SuchCtl
    & "'") Then
        DoCmd.GoToRecord acDataForm, "frmArtikel", acGoTo, SuchID
        DoCmd.FindRecord SuchID, acEntire, , acSearchAll, , acCurrent
        Me.cmdSpeichern.Visible = True
        Me.cmdLoeschen.Visible = True
    Else
        MsgBox "Der Datensatz existiert nicht."
        Exit Sub
    End If
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Err.Description, vbInformation, "Fehler"
    Resume Exit_Fehler
End Sub

```

## 5) Das Ereignis *Beim Klicken* für die Schaltfläche cmdSpeichern

```

Private Sub cmdSpeichern_Click()
On Error GoTo Zu_Fehler
On Error Resume Next
    If (Forms!frmArtikel.Dirty) Then
        DoCmd.RunCommand acCmdSaveRecord
        DoCmd.Requery
    End If
    DoCmd.GoToRecord acDataForm, "frmArtikel", acNewRec
    Me.cmdSpeichern.Visible = False
    Me.cmdLoeschen.Visible = False
    On Error GoTo 0
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Error$
    Resume Exit_Fehler
End Sub

```

#### 6) Das Ereignis *Beim Klicken* für die Schaltfläche cmdLoeschen

```

Private Sub cmdLoeschen_Click()
On Error GoTo Zu_Fehler
    DoCmd.RunCommand acCmdDeleteRecord
    DoCmd.Requery
    Me.cmdSpeichern.Visible = False
    Me.cmdLoeschen.Visible = False
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Error$
    Resume Exit_Fehler
End Sub

```

#### 7) Das Ereignis *Beim Klicken* für die Schaltfläche cmdSchliessen

```

Private Sub cmdSchliessen_Click()
On Error GoTo Zu_Fehler
On Error Resume Next
    If (Forms!frmArtikel.Dirty) Then
        DoCmd.RunCommand acCmdSaveRecord
    End If
    On Error GoTo 0
    DoCmd.Close acForm, "frmArtikel"
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Error$
    Resume Exit_Fehler
End Sub

```

#### 8) Das Ereignis *Beim Verlassen* für das Textfeld Artikelcode

```

Private Sub Artikelcode_Exit(Cancel As Integer)
On Error GoTo Zu_Fehler
    Dim SuchCtl As String
    SuchCtl = Me.Artikelcode.Value
    If DLookup("[ArtikelID]", "tblArtikel", "Artikelcode LIKE '" & SuchCtl & "'") Then
        Dim rst As DAO.Recordset

```

```

Set rst = Me.Recordset
rst.FindFirst "[Artikelcode] = '" & Me!Artikelcode & "'"
If rst.NoMatch Then
    MsgBox "Der Datensatz wurde nicht gefunden."
End If
Else
    Rem die Befehlsschaltflächen einblenden!
    Me.cmdSpeichern.Visible = True
    Me.cmdLoeschen.Visible = True
End If
Exit_Fehler:
Exit Sub
Zu_Fehler:
MsgBox Err.Description, vbInformation, "Fehler"
Resume Exit_Fehler
End Sub

```

### 9) Das Ereignis *Beim Klicken* für die Schaltfläche cmdstandortoeffnen

```

Private Sub cmdstandortoeffnen_Click()
On Error GoTo Zu_Fehler
Dim SuchCtl As String
If Me.StandortID.Value Then
    SuchCtl = "[StandortID] = " & Me.[StandortID]
    DoCmd.OpenForm "frmStandort", acNormal, , SuchCtl
    DoCmd.GoToRecord , , acGoTo
Else
    DoCmd.OpenForm "frmStandort", acNormal
End If
Exit_Fehler:
Exit Sub
Zu_Fehler:
MsgBox Err.Description, vbInformation, "Fehler"
Resume Exit_Fehler
End Sub

```

### 10) Das Ereignis *Beim Klicken* für die Schaltfläche cmdLieferantenoeffnen

```

Private Sub cmdLieferantenoeffnen_Click()
On Error GoTo Zu_Fehler
Dim SuchCtl As String
If Me.LieferantenID.Value Then
    SuchCtl = "[LieferantenID] = " & Me.[LieferantenID]
    DoCmd.OpenForm "frmLieferanten", acNormal, , SuchCtl
    DoCmd.GoToRecord , , acGoTo
Else
    DoCmd.OpenForm "frmLieferanten", acNormal
End If
Exit_Fehler:
Exit Sub
Zu_Fehler:
MsgBox Err.Description, vbInformation, "Fehler"
Resume Exit_Fehler
End Sub

```

### 11) Das Ereignis *Beim Klicken* für die Schaltfläche cmdUnterkategorieoeffnen

```

Private Sub cmdUnterkategorieoeffnen_Click()

```

```

On Error GoTo Zu_Fehler
    Dim SuchCtl As String
    If Me.UnterkategorieID.Value Then
        SuchCtl = "[UnterkategorieID] = " & Me.[UnterkategorieID]
        DoCmd.OpenForm "frmUnterkategorie", acNormal, , SuchCtl
        DoCmd.GoToRecord , , acGoTo
    Else
        DoCmd.OpenForm "frmUnterkategorie", acNormal
    End If
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Err.Description, vbInformation, "Fehler"
    Resume Exit_Fehler
End Sub

```

### 12) Das Ereignis *Beim Klicken* für die Schaltfläche cmdUnterkategorieoeffnen

```

Private Sub cmdUnterkategorieoeffnen_Click()
On Error GoTo Zu_Fehler
    Dim SuchCtl As String
    If Me.UnterkategorieID.Value Then
        SuchCtl = "[UnterkategorieID] = " & Me.[UnterkategorieID]
        DoCmd.OpenForm "frmUnterkategorie", acNormal, , SuchCtl
        DoCmd.GoToRecord , , acGoTo
    Else
        DoCmd.OpenForm "frmUnterkategorie", acNormal
    End If
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Err.Description, vbInformation, "Fehler"
    Resume Exit_Fehler
End Sub

```

### 13) Das Ereignis *Beim Klicken* für die Schaltfläche cmdEinheitoeffnen

```

Private Sub cmdEinheitoeffnen_Click()
On Error GoTo Zu_Fehler
    Dim SuchCtl As String
    If Me.EinheitID.Value Then
        SuchCtl = "[EinheitID] = " & Me.[EinheitID]
        DoCmd.OpenForm "frmEinheiten", acNormal, , SuchCtl
        DoCmd.GoToRecord , , acGoTo
    Else
        DoCmd.OpenForm "frmEinheiten", acNormal
    End If
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Err.Description, vbInformation, "Fehler"
    Resume Exit_Fehler
End Sub

```

### 14) Das Ereignis *Beim Klicken* für die Schaltfläche cmdladen

```

Private Sub cmdLaden_Click()
    Dim strDatei As String

```

```

Dim strPfadname As String
Dim result As Integer
With Application.FileDialog(3)
    .Title = "Artikelbild auswählen"
    .Filters.Add "Alle Dateien", "*.*"
    .Filters.Add "JPEG-Dateien", "*.jpg"
    .Filters.Add "Bitmaps", "*.bmp"
    .FilterIndex = 3
    .AllowMultiSelect = False
    .InitialFileName = CurrentProject.Path
    result = .Show
    If (result <> 0) Then
        strPfadname = Trim(.SelectedItem.Item(1))
        Me![txtBildpfad].Visible = True
        Me![txtBildpfad].SetFocus
        strDatei = Right(strPfadname, Len(strPfadname) - Len(CurrentProject.Path))
        Me![txtBildpfad].Text = strDatei
        Me![Artikelcode].SetFocus
        Me![txtBildpfad].Visible = False
    End If
End With
End Sub

```

#### 15) Das Ereignis *Beim Klicken* für die Schaltfläche cmdEntfernen

```

Private Sub cmdEntfernen_Click()
    Me![txtBildpfad] = ""
    Me![imgRahmen].Visible = False
    lblMeldung.Visible = True
End Sub

```

#### 16) Das Ereignis *Beim Klicken* für die Schaltfläche cmdAnzeigen

```

Private Sub cmdAnzeigen_Click()
On Error GoTo Zu_Fehler
    Dim stDocName As String
    Dim LinkCriteria As String
    LinkCriteria = "[ArtikelID] = " & Me.[ArtikelID]
    stDocName = "rptArtikel"
    DoCmd.OpenReport stDocName, acPreview, , LinkCriteria
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Err.Description, vbInformation, "Fehler"
    Resume Exit_Fehler
End Sub

```

#### 17) Das Ereignis *Nach Aktualisierung* für das Textfeld txtBildpfad

```

Private Sub txtBildpfad_AfterUpdate()
    On Error Resume Next
    showErrorMessage
    Me![imgRahmen].Visible = True
    If (IsRelative(Me!txtBildpfad) = True) Then
        Me![imgRahmen].Picture = Path & Me![txtBildpfad]
    Else
        Me![imgRahmen].Picture = Me![txtBildpfad]
    End If
End Sub

```

```
End Sub
```

Die beiden Hilfsfunktionen für das Bild lauten:

```
Sub showErrorMessage()
    If Not IsNull(Me!Bild) Then
        lblMeldung.Visible = False
    Else
        lblMeldung.Visible = True
    End If
End Sub

-----
Function IsRelative(fName As String) As Boolean
    IsRelative = (InStr(1, fName, ":") = 0) And (InStr(1, fName, "\\") = 0)
End Function
```

### 1.7.4 Ereignis im Bericht rptArtikel

Wie schon im Abschnitt 1.5.4 erläutert wurde, wird ein Bild mit dem Datensatz gleichzeitig in einem Bericht angezeigt, wenn man in das Ereignis `Beim Formatieren` im Detailbereich des Berichtes explizit anweist.

Das Ereignis *Beim Formatieren* für den Detailbereich des Berichtes rptArtikel

```
Option Compare Database
Dim Path As String

-----
Private Sub Detailbereich_Format(Cancel As Integer, FormatCount As Integer)
    Dim res As Boolean
    Dim fName As String
    Path = CurrentProject.Path
    On Error Resume Next

        If Not IsNull(Me!Bild) Then

            res = IsRelative(Me!Bild)
            fName = Me![txtBildpfad]
            If (res = True) Then
                fName = Path & fName
            End If

            Me![imgRahmen].Picture = fName
            Me![imgRahmen].Visible = True
            Me.PaintPalette = Me![imgRahmen].ObjectPalette
            If (Me![imgRahmen].Picture <> fName) Then
                Me![imgRahmen].Visible = False
            End If
        Else
            Me![imgRahmen].Visible = True
        End If
End Sub

-----
Function IsRelative(fName As String) As Boolean
    IsRelative = (InStr(1, fName, ":") = 0) And (InStr(1, fName, "\\") = 0)
End Function
```

### 1.7.5 Ereignisse auf die Objekte im Formular frmUnterkategorie

Die globalen Deklarationen lauten:

```
Option Compare Database
Option Explicit
```

Und die Ereignisse, die u.a. im Abschnitt 1.5.1 dargestellt wurden, lauten:

#### 1) Das Ereignis *Beim Laden* für das Formular frmUnterkategorie

```
Private Sub Form_Load()
On Error GoTo Zu_Fehler
    Rem DoCmd.GoToRecord , , acNewRec
    Rem beim Laden die Befehlsschaltflächen unsichtbar machen
    Me.cmdSpeichern.Visible = False
    Me.cmdLoeschen.Visible = False
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Err.Description, vbInformation, "Fehler"
    Resume Exit_Fehler

End Sub
```

#### 2) Das Ereignis *Beim Klicken* für das Kombinationsfeld cboUnterkategorie

```
Private Sub cboUnterkategorie_Click()
On Error GoTo Zu_Fehler
    Dim SuchID As Long, SuchCtl As String
    SuchID = Me!cboUnterkategorie.ListIndex + 1
    SuchCtl = Me!cboUnterkategorie.Column(1)
    If DLookup("[UnterkategorieID]", "tblUnterkategorie", _ &
        "Unterkategorienname LIKE '" & SuchCtl & "'") Then
        DoCmd.GoToRecord acDataForm, "frmUnterkategorie", acGoTo, SuchID
        DoCmd.FindRecord SuchID, acEntire, , acSearchAll, , acCurrent
        Rem die Befehlsschaltflächen sichtbar machen
        Me.cmdSpeichern.Visible = True
        Me.cmdLoeschen.Visible = True
        Rem Dieser Teil wird erst eingetragen, nachdem Artikel zusammengestellt wird.
        Dim strSQL As String
        strSQL = "SELECT DISTINCT * " &
            "FROM sqlArtikelfuerUnterkategorie " &
            "WHERE Unterkategorienname= '" & Me!cboUnterkategorie.Column(1) & "' " &
            "ORDER BY [Unterkategorienname] asc"
        Debug.Print strSQL
        Me!LstArtikel.RowSource = strSQL
    Else
        MsgBox "Der Datensatz existiert nicht."
        Exit Sub
    End If
Exit_Fehler:
    Exit Sub
Zu_Fehler:
    MsgBox Err.Description, vbInformation, "Fehler"
    Resume Exit_Fehler

End Sub
```

#### 3) Das Ereignis *Beim Klicken* für die Schaltfläche cmdSpeichern

```
Private Sub cmdSpeichern_Click()
On Error GoTo Zu_Fehler
    If (Forms!frmUnterkategorie.Dirty) Then
        DoCmd.RunCommand acCmdSaveRecord
        DoCmd.Requery
    End If
```



```

DoCmd.GoToRecord acDataForm, "frmUnterkategorie", acNewRec
Rem die Befehlsschaltflächen sichtbar machen
Me.cmdSpeichern.Visible = False
Me.cmdLoeschen.Visible = False
Exit_Fehler:
Exit Sub
Zu_Fehler:
MsgBox Err.Description, vbInformation, "Fehler"
Resume Exit_Fehler
End Sub

```

#### 4) Das Ereignis *Beim Klicken* für die Schaltfläche cmdLoeschen

```

Private Sub cmdLoeschen_Click()
On Error GoTo Zu_Fehler
DoCmd.RunCommand acCmdDeleteRecord
DoCmd.Requery
Rem die Befehlsschaltflächen sichtbar machen
Me.cmdSpeichern.Visible = False
Me.cmdLoeschen.Visible = False
Exit_Fehler:
Exit Sub
Zu_Fehler:
MsgBox Err.Description, vbInformation, "Fehler"
Resume Exit_Fehler
End Sub

```

#### 5) Das Ereignis *Beim Klicken* für die Schaltfläche cmdSchliessen

```

Private Sub cmdSchliessen_Click()
On Error GoTo Zu_Fehler
If (Forms!frmUnterkategorie.Dirty) Then
Dim Ergebnis As Integer
Ergebnis = MsgBox("Wollen Sie den Datensatz speichern?", vbYesNo +
vbCritical + vbDefaultButton2, "Datensatz löschen")
If Ergebnis = vbYes Then
DoCmd.RunCommand acCmdSaveRecord
End If
End If
DoCmd.Close acForm, "frmUnterkategorie"
Exit_Fehler:
Exit Sub
Zu_Fehler:
MsgBox Err.Description, vbInformation, "Fehler"
Resume Exit_Fehler
End Sub

```

#### 6) Das Ereignis *Beim Verlassen* für das Textfeld Unterkategoriename

```

Private Sub Unterkategoriename_Exit(Cancel As Integer)
On Error GoTo Zu_Fehler
Dim SuchCtl As String
SuchCtl = Me.Unterkategoriename.Value
If DLookup("[UnterkategorieID]", "tblUnterkategorie",
"Unterkategoriename LIKE '" & SuchCtl & "'") Then
Dim rst As DAO.Recordset
Set rst = Me.Recordset
rst.FindFirst "[Unterkategoriename] = '" & Me!Unterkategoriename &

```

```

'''
    If rst.NoMatch Then
        MsgBox "Der Datensatz wurde nicht gefunden."
        Exit Sub
    End If
Else
    Rem die Befehlsschaltflächen sichtbar machen
    Me.cmdSpeichern.Visible = True
    Me.cmdLoeschen.Visible = True
End If
Exit_Fehler:
Exit Sub
Zu_Fehler:
MsgBox Err.Description, vbInformation, "Fehler"
Resume Exit_Fehler
End Sub

```

### 7) Das Ereignis *Beim Klicken* für die Schaltfläche cmdkategorieoeffnen

```

Private Sub cmdkategorieoeffnen_Click()
On Error GoTo Zu_Fehler
    Dim SuchCtl As String
    If Me.KategorieID.Value Then
        SuchCtl = "[KategorieID] = " & Me.[KategorieID]
        DoCmd.OpenForm "frmkategorie", acNormal, , SuchCtl
        DoCmd.GoToRecord , , acGoTo
    Else
        DoCmd.OpenForm "frmkategorie", acNormal
    End If
Exit_Fehler:
Exit Sub
Zu_Fehler:
MsgBox Err.Description, vbInformation, "Fehler"
Resume Exit_Fehler
End Sub

```

### 8) Das Ereignis *Beim Klicken* für die Schaltfläche cmdAnzeigen

```

Private Sub cmdAnzeigen_Click()
On Error GoTo Zu_Fehler
    Dim SuchCtl As String
    If Me.KategorieID.Value Then
        SuchCtl = "[KategorieID] = " & Me.[KategorieID]
        DoCmd.OpenReport "rptUnterkategorie", acPreview, , SuchCtl

    Else
        MsgBox "Wählen Sie einen Kategorie aus!"
        Exit Sub
    End If
Exit_Fehler:
Exit Sub
Zu_Fehler:
MsgBox Err.Description, vbInformation, "Fehler"
Resume Exit_Fehler
End Sub

```

9) Das Ereignis *Beim Klicken* für die Artikelliste LstArtikel

```
Private Sub LstArtikel_Click()  
On Error GoTo Zu_Fehler  
    Dim LinkCriteria As String  
    LinkCriteria = "[ArtikelID] = " & Me!LstArtikel.Column(2)  
    DoCmd.OpenForm "frmArtikel", acNormal, , LinkCriteria  
Exit_Fehler:  
    Exit Sub  
Zu_Fehler:  
    MsgBox Err.Description, vbInformation, "Fehler"  
    Resume Exit_Fehler  
End Sub Exit_Fehler  
End Sub
```

---

<sup>i</sup> Datenbankmodell: <http://de.wikipedia.org/wiki/Datenbankmodell>

<sup>ii</sup> Einführung in SQL: [http://de.wikibooks.org/wiki/Einf%C3%BChrung\\_in\\_SQL:\\_Inhaltsverzeichnis](http://de.wikibooks.org/wiki/Einf%C3%BChrung_in_SQL:_Inhaltsverzeichnis)

<sup>iii</sup> Bedeutung: [http://de.wikipedia.org/wiki/Relationale\\_Algebra](http://de.wikipedia.org/wiki/Relationale_Algebra)

<sup>iv</sup> Wirtschaftslexikon: <http://www.wirtschaftslexikon24.net/i/index-p.htm>

<sup>v</sup> Bestandführung: <http://www.wirtschaftslexikon24.net/d/bestandsfuehrung/bestandsfuehrung.htm>

<sup>vi</sup> Vorgehensweise der strukturierten Programmierung, Seite 14, Access 2010 , Herdt –Verlag von Ricardo

Hernández García

<sup>vii</sup> Access-Entwicklerreferenz : CommandButton.Caption-Eigenschaft (Beschriftung) mit Taste F1 aufrufen

<sup>viii</sup> Datenbank Online Lexikon: [http://wikis.gm.fh-koeln.de/wiki\\_db/Datenbanken/Erste-Normalform](http://wikis.gm.fh-koeln.de/wiki_db/Datenbanken/Erste-Normalform)