

# Makro-Aufgaben in Access 2010–2013

Eva Vogel (MCT)



Download free books at

Eva Vogel

# Makro-Aufgaben in Access 2010–2013

---

Makro-Aufgaben in Access 2010–2013

1. Auflage

© 2014 Eva Vogel & [bookboon.com](http://bookboon.com)

ISBN 978-87-403-XXXX-X

# Inhalt

	<b>Moin Moin – Wozu eigentlich Makros?</b>	<b>6</b>
	<b>Neuigkeiten zu Makros in Access 2010–2013</b>	<b>7</b>
<b>1</b>	<b>Der neue Makro-Generator schaut anders aus</b>	<b>8</b>
1.2	In Access 2010 abgeschaffte Makro-Aktionen	11
<b>2</b>	<b>Makros in Tabellen</b>	<b>12</b>
2.1	Makro 1: Text-Importe	13
2.2	Makro 2: Statusanzeige Datenmakro vor Änderung	15
2.3	Makro 3: Löschen eines Datensatzes vor Löschen, Datenmakro	16
2.4	Makro 4: Zeitstempel für Datensatzaktualisierungen	18
<b>3</b>	<b>Makros mit Abfragen</b>	<b>19</b>
3.1	Makro 5: Beispiel für ein Makro mit SQL-Abfrage, VBA-Code und Textexport	19
3.2	Makro 6: Bericht mit Meldungsinfo und PDF-Ausdruck	24
3.3	Makro 7: FeldSTRUKTUR per VBA und SQL ändern, hinzufügen oder löschen	25

Wenn ein

# Server

für Sie kein  
Wassersportler  
ist...


**IT-Jobs bei Lidl**  
[it-bei-lidl.com](http://it-bei-lidl.com)




3.4	Makro 8: Makro-Aktionen im Einzelschritt testen	26
3.5	Makro 9: DatenINHALTE über VBA, SQL und Makro ändern	27
3.6	Makro 10: The easy way – Exportvorgänge einfach speichern!	30
<b>4</b>	<b>Makros in Formularen</b>	<b>33</b>
4.1	Makro 11: Senden einer E-Mail mit Schaltfläche	33
4.2	Makro 12: ChangeButton (Umschaltfläche) zum Sortieren	35
4.3	Makro 13: „Maske? Welche Maske?“	36
4.4	Makro 14: Suchen in Formularen	39
<b>5</b>	<b>Makro 15: Filtern in Formularen</b>	<b>42</b>
5.1	Die Beziehungen zu den neuen Tabellen	42
5.2	Ein Kombifeld zum Anzeigen aller Orte (SQL)	44
5.3	Filter aller Kunden nach Auswahl eines Ortes (VBA)	46
5.4	Filtern über Makro mit Abfragekriterien	48
<b>6</b>	<b>Endnoten</b>	<b>50</b>

**EY**  
Building a better  
working world

**So müsste er  
aussehen: unser  
Firmenwagen  
für Einsteiger.**

[www.de.ey.com/karriere](http://www.de.ey.com/karriere)  
#BuildersWanted

„EY“ und „wir“ beziehen sich auf alle deutschen Mitgliedsunternehmen von Ernst & Young Global Limited, einer Gesellschaft mit beschränkter Haftung nach englischem Recht. ED.None.



# Moin Moin – Wozu eigentlich Makros?

Makros sind in meinen Augen „Kleine Progrämmchen“, die das Bearbeiten, Abarbeiten und nervige immer gleich bleibende Prozedere in unserem Arbeitsalltag mit MS Access vereinfachen sollen, indem sie einfach nur stur das in 10 Schritten tun, was ich mit einem Schaltflächendruck erledigt haben möchte. Und eigentlich hatte ich schon beim letzten Access-Buch erst mal ein bisschen Pause verdient und hätte mir selbst gerne einen Button gebaut wie „cmd.SchreibDuDochSelbstDiesesTutorial“ ;). Die Tatsache, dass ich deshalb mehr Bild statt Wort hier versammelt habe, verfolgt ein klares Ziel: Mehr selber machen, weniger reden!

Viele weitere Teilnehmerfragen in meinen Seminaren wiesen allerdings immer auf dieselben dringenden Fragen hin, z.B.: „Ich hab da eine Tabelle in meine Datenbank importiert. Muss ich die von Hand jedes Mal in der Struktur nur abändern oder gibt’s da ein Makro nach dem Import?“ Wenn Sie wie ich nach schnellen einfachen Lösungen suchen und sich doch schon ganz fit in der Datenbank-Erstellung mit MS Access fühlen, hilft vielleicht diese kleine Sammlung, nicht abschließend, sondern impulsgebend.

Wie immer habe ich mir eine kleine Beispiel-Datenbank ausgedacht. In dieser „KundenVerkaufProdukte.accdb“ bewegen wir uns kapitelweise in möglichen Makros zu Tabellen, Abfragen und Formularen.

Und wozu eigentlich nur 15 Makros? Ganz einfach: Dieses Buch ist nicht als Kompendium gedacht, sondern ist nur meine Sammlung einiger – also nur 15 – Makros, die Ihnen den Datenbank-Alltag vielleicht auf die eine oder andere Weise erleichtern können. Auch die VBA- oder SQL-Welt wurde hier nicht an einem Tag erbaut, so dass uns eine ausführliche Lektüre niemals erspart sein wird.

# Neuigkeiten zu Makros in Access 2010–2013

Hab ich gesagt, Makros wären einfach?

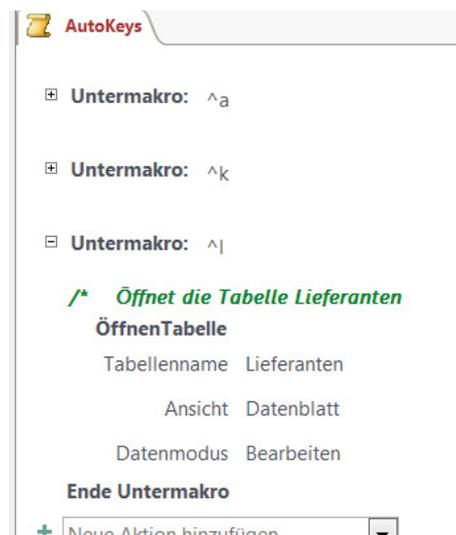
Je nach Aufgabenstellung sind sie es nur dann, wenn wir auch die anderen Sprachelemente der Automatisierung – XML, VBA und SQL – von MS Access kurz kennen lernen. Sonst haben Sie nicht wirklich viel davon. Wir unterscheiden also bei Automatisierungen durch Makros in 3 Anwendungs- / Entwicklungsbereiche:

1. **Codierung mit XML** (Menüband – Register – Backstage)
  - a) Die guten alten Start-Einstellungen von Access waren mal. Jetzt wird alles extern über XML gesteuert, wenn Sie das Menüband und die Register tatsächlich ein für alle Mal aus Ihrem Anwendungsfenster beseitigen möchten. Zum Glück hat uns Microsoft noch einen XML-Editor namens „CustomUI“ spendiert.<sup>1</sup>
2. **Programmierung mit VBA 7.0**
  - a) Die DAO wird doch wieder Standard-Verweisbibliothek, ADO bleibt mit bisherigen Benutzerberechtigungen möglich und beißt sich leider anschließend mit DAO-Befehlen. Lieber sich auf ein Objekt-Modell einigen, je nach Aufgabenpriorität!
  - b) Deaktiviert sind u.a. das ActiveX-Kalender-Steuerelement und die Snapshot-Funktion bei Berichten. Wäre ja auch Käse: Die neuen ActiveX-Datumsfelder bringen die Kalender jetzt mit und Berichte lassen sich seit Access 2007 endlich auch als pdf speichern und ausdrucken.<sup>2</sup>
3. **Abfrageskripte mit SQL** zum Auswerten, Manipulieren, Definieren und Kontrollieren
  - a) Um mit dem ADO-Objektmodell keine Überschneidungen mit DAO-Standard-Objekten zu riskieren, empfiehlt sich bei bisherigen („alten“) mdb-Dateien das ADO-CurrentProject-Connection-Objekt und die schon vorher eingerichtete DCL für Benutzer- und Gruppenberechtigungen.
  - b) Nachteil: Einmal als Admin mit Kennwort identifiziert, meldet sich die Access-Oberfläche kategorisch auch für accdb-Datenbanken mit Kennwort-Abfrage. Drum prüfe, wer sich ewig bindet. ☺

# 1 Der neue Makro-Generator schaut anders aus



Es gibt Autostart-Makros mit dem reservierten Namen „AutoExec“ oder „AutoKeys“ für selbst definierte Shortcuts, neu sind Makrostrukturen im Programm-Ablauf, z.B. mit WENN-Funktion.



Kommentare werden in grüner Schrift eingefügt

Untermakros werden als Tastaturkürzel gestartet,

SHIFT → +,

z.B. +{F10}

STRG → ^,

z.B. ^+{F12} für strg-shift-f12

1.1.1 Gleichbleibend sind folgende Makro-Aktionen

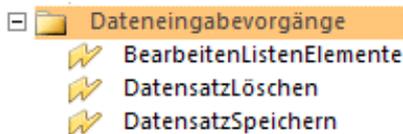
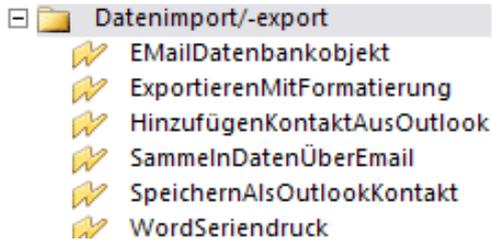
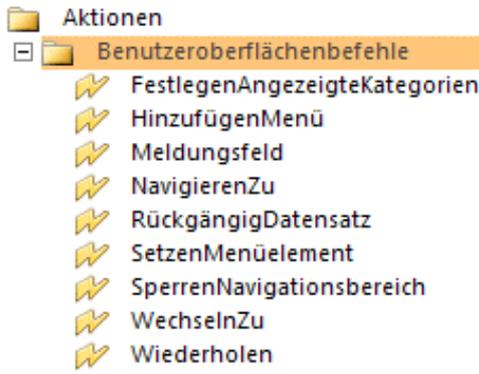
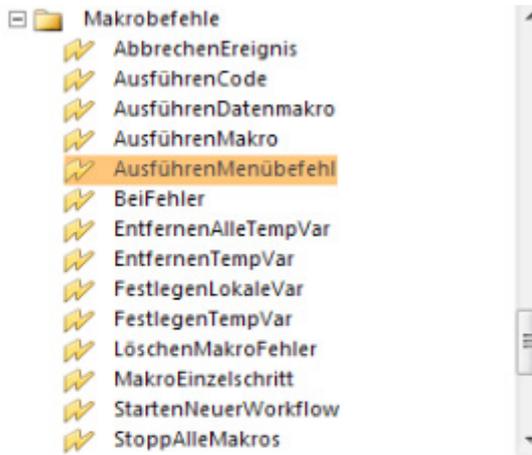
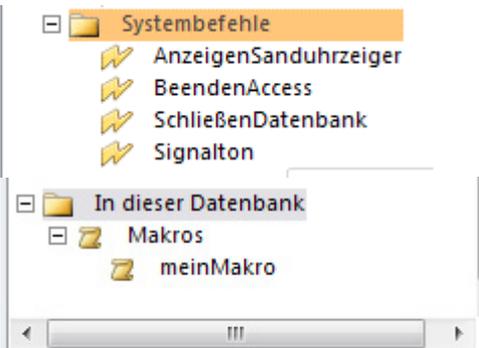
The screenshot shows the Access Macro Generator interface. On the left, under the 'Datenbankobjekte' category, the following actions are listed: AktualisierenObjekt, AuswählenObjekt, DruckenObjekt, FestlegenEigenschaft, GeheZuDatensatz, GeheZuSeite, GeheZuSteuerelement, ÖffnenBericht, ÖffnenFormular, ÖffnenTabelle, and Seitenansicht. On the right, under the 'Filter/Abfrage/Suche' category, the following actions are listed: Aktualisieren, AktualisierenDaten, AktualisierenDatensatz, AnwendenFilter, AnzeigenAlleDatensätze, FestlegenFilter, FestlegenSortiertNach, FilterSortierungEntfernen, ÖffnenAbfrage, SuchenDatensatz, SuchenNachDatensatz, and SuchenNächstenDatensatz. Below the list, a detailed description for 'SuchenNächstenDatensatz' is provided: 'Sucht den nächsten Datensatz, der den Kriterien der letzten SuchenDatensatz-Aktion oder des Dialogfelds 'Suchen' entspricht. Verwenden Sie diese Aktion, um nacheinander die Datensätze zu finden, die dieselben Kriterien erfüllen.'

Auf jeden Fall neu ist die Aktion „Seitenansicht“ des jew. aktuellen Objektes.

1.1.2 Geändert haben sich folgende Makro-Aktionen

Alt ( bis 2003)	Neu (ab 2007/2010)
Meldung	Meldungsfenster mit 6 (vbYes) oder 7 (vbNo) als VB-Konstanten und Dialogabfrage!
AusführenBefehl	AusführenMenübefehl
Ausgabeln	ExportierenMitFormatierung
SetzenWert (SetzenEigenschaft Acc2007)	„FestlegenFeld“ wechselt den Aktionsnamen je nach Datenmakro / Makro auf „FestlegenEigenschaft“

1.1.3 Es gibt etliche neue Makro-Aktionen

Neu	Auch neu
<p>Dateneingabevorgänge-BearbeitenListenelemente</p> 	
<p>Komplett neue Befehle über Menü und Navigationsleiste, usw., zum Steuern der Inhalte des Bildschirms</p> <p><b>Sperr Navigationsbereich</b> ist irreführend: Nicht der Navi-Bereich wird gesperrt, sondern die Kontext-Menüs werden teilweise deaktiviert für ein angeklicktes Objekt. Z.B. ist nur noch Entwurfsansicht und Exportieren im Kontextmenü möglich, aber nicht mehr das Umbenennen eines Objektes!</p>	
 <p><b>AusführenMenübefehl</b></p> <p>Führt einen Microsoft Access-Menübefehl aus. Der Befehl muss für die Ansicht geeignet sein, die aktuell ist, wenn das Makro den Befehl ausführt.</p>	 <p><b>In dieser Datenbank</b></p> <p>Liste der Objekte, für die es in der Datenbank Makros gibt.</p>

AusführenCode	Ausgeführt werden wie immer nur globale functions () und keine Subs in allg. Modulen.
AusführenSQL  1. Aktions-Abfragen ( in SQL mit ; am Anfang – in VBA als „STRING“ ohne ;) ) 2. Datendefinitions-Abfragen	<ul style="list-style-type: none"> <li>- <b>Tabellen-Erstellung</b> Bsp: SELECT Felder INTO TabNeu FROM Tabelle</li> <li>- <b>Anfügen von Datensätzen</b> Bsp: INSERT INTO Tabelle (Felder) VALUES (Eigsch.)</li> <li>- <b>Aktualisieren von Datensätzen</b> Bsp.: UPDATE Tab SET Ausdruck = Wert WHERE Bed</li> <li>- <b>Löschen von Datensätzen</b> Bsp: DELETE Felder FROM Tabelle CREATE TABLE – ALTER TABLE – DROP TABLE</li> </ul>

### 1.2 In Access 2010 abgeschaffte Makro-Aktionen

Echo	Warnmeldungen Ein/Aus
TransferText mit externen Spezifikationsdateien, statt dessen DoCmd.RunSavedImportExport	AusführenBefehl ist jederzeit mit dem DoCmd-Objekt in VBA einsetzbar
TransferArbeitsblatt (DoCmd.TransferSpreadsheet)	AusführenAnwendung (Shell)



# MEINE TO DO'S

- Wohnung suchen
- Mit Mama zu IKEA fahren
- Stundenplan erstellen
- Nebenjob auf Jobmensa.de finden

Entdecke jetzt deutschland's größtes Jobportal für Studenten



# 2 Makros in Tabellen

Unnötig zu sagen, dass Sie erst mal eine Datenbank modellieren müssen, bevor Sie darin Makros bauen können. Wir basteln uns wie schon in Access 2010 Basis erklärt eine neue Übungsdatenbank „KundenVerkaufProdukte.accdb“.

1. **tabKunden.xlsx** wird die Import-Datenbasis für tabKunden:

	A	B	C	D	E	F	G
1	<b>KNR</b>	<b>Zuname</b>	<b>Vorname</b>	<b>Straße</b>	<b>Ort</b>	<b>Kategorie</b>	<b>PLZ</b>
2	2000	Hohenberger	Josef	Kirchengasse 3	Hallein	1	5400
3	2001	Gruber	Markus	Waldweg 7	Golling	3	5440
4	2002	Hofer	Martin	Fichtenweg 19	Mattsee	1	5163
5	2003	Höllner	Peter	Markt 28	Golling	2	5440
6	2004	Brugger	Peter	Tannenweg 5	Hallein	1	5400

2. **tabProdukte.xlsx** wird Import-Datenbasis für tabProdukte

	A	B	C
1	PID	Pbezeichnung	Ppreis
2	1	Hose	29,9
3	2	Jacke	49
4	3	Hemd	39
5			
6			

Tab\_Kunden    **tabProdukte**

3. **tabVerkauf.xlsx**

	A	B	C	D
1	VID	Kunde	VDatum	Vbezahlt
2	1	2000	10.04.2013	WAHR
3	2	2001	12.04.2013	WAHR
4				
5				
6				

Tab\_Kunden    tabProdukte    **tabVerkauf**

4. tabVerkaufsdetails.xlsx

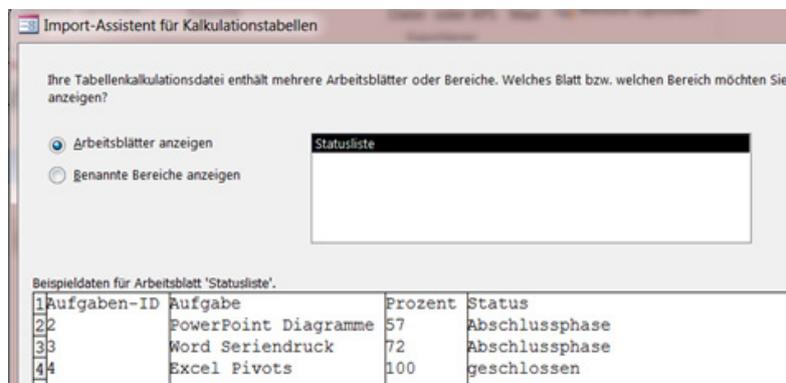
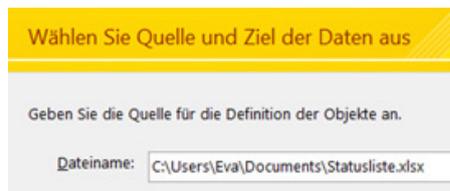
	A	B	C	D	E	F	G
1	VDID	Verkauf	Produkt	VDAnzahl			
2	1	1	1	1			
3	2	1	2	1			
4	3	2	3	1			
5							
6							

Tab\_Kunden | tabProdukte | tabVerkauf | **tabVerkaufsdetails**

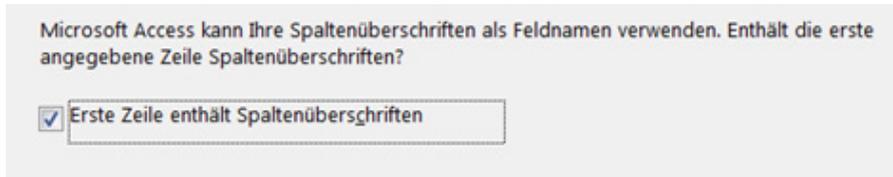
5. tabStatusliste.csv

	A	B	C	D
1	Aufgaben-ID	Aufgabe	Prozent	Status
2		2 PowerPoint Diagramme	57	
3		3 Word Seriendruck	72	
4		4 Excel Pivots	100	

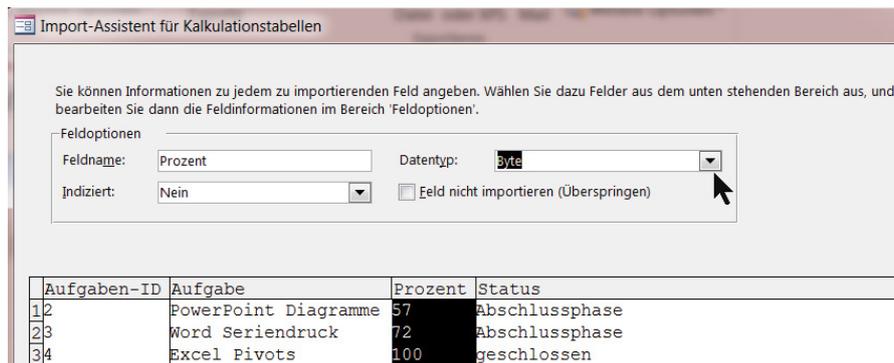
2.1 Makro 1: Text-Importe



– Weiter



– Weiter



**strategy&**

Bewirb Dich bis zum 18. Oktober 2015.

# DATA EMERGENCY

&

**7. - 9. November 2015, Berlin**

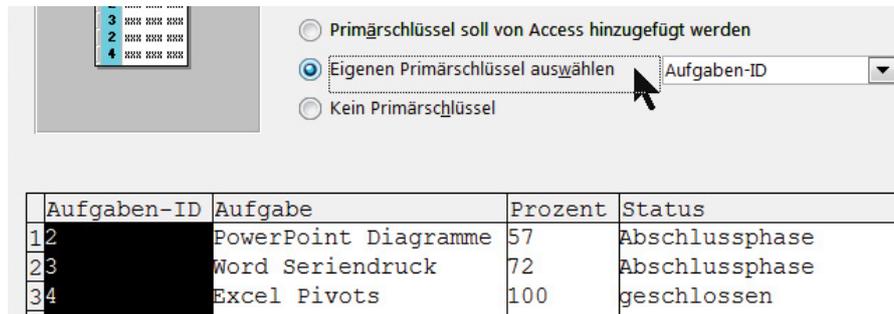
Gesundheitsbranche in der Datenkrise!  
Deine innovativen Ideen und Strategien zum Thema e-Health sind gefragt.  
Entwickle gemeinsam mit Strategy&-Beratern Hightech-Strategien für eine gesunde Zukunft.

Mehr Informationen unter [www.strategyand.pwc.com/DBTacademy](http://www.strategyand.pwc.com/DBTacademy)

© 2015 PwC. All rights reserved.  
PwC refers to the PwC network and/or one or more of its member firms, each of which is a separate legal entity. Please see [www.pwc.com/structure](http://www.pwc.com/structure) for further details.




– Datentypen – Weiter



– Weiter – Tabelle speichern – Fertig stellen – Importschritte speichern

Und das bitte nochmal, bis alle Tabellen hier in unserer Datenbank gelandet sind.

2.2 Makro 2: Statusanzeige Datenmakro vor Änderung

Wenn Status 100/50/0%, dann Aufgabe geschlossen/in Arbeit/offen.

Aufgaben-ID	Aufgabe	Prozent	Status	Zum
2	PowerPoint Diagramme	34	offen	
3	Word Seriendruck	72	Abschlussphase	
4	Excel Pivots	100	geschlossen	
*	(Neu)			

Die Tabelle „Statusliste“ enthält folgende Felder:

- Aufgaben-ID: AutoWert
- Aufgabe: Text (40)
- Prozent: Zahl – Byte – Allgemeine Zahl (Kein Prozent!!)
- Status: Text (20)

Hier ist das Datenmakro im Entwurf:

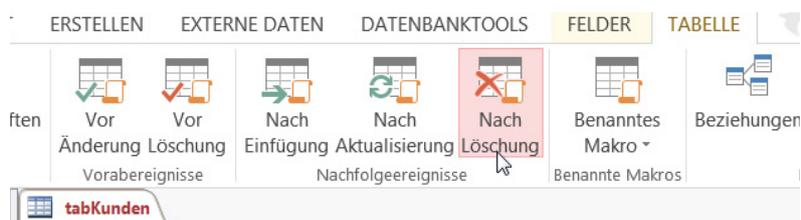


### 2.3 Makro 3: Löschen eines Datensatzes vor Löschen, Datenmakro

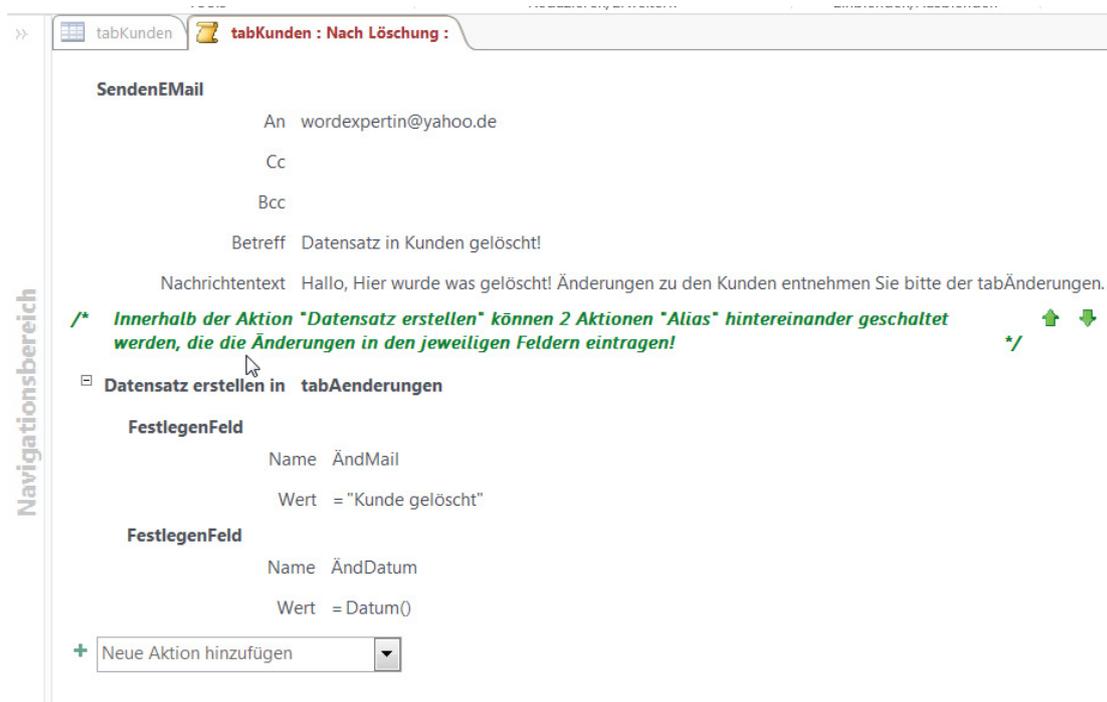
Zuerst erstellen wir uns noch eine weitere Tabelle zum Aufzeichnen von Kundenlösungen.

Feldname	Felddatentyp
ÄnderungsID	AutoWert
ÄndMail	Kurzer Text
ÄndDatum	Datum/Uhrzeit

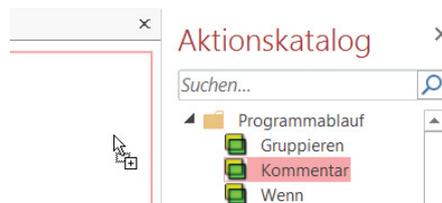
Zum Erstellen eines Datenmakros öffnen Sie die Tabelle, klicken Sie auf Register Tabellentools – TABELLE:



Wenn ein Datensatz gelöscht wurde, dann soll eine Mail an z.B. den Administrator erfolgen, dass ein Datensatz gelöscht wurde und Änderungs-/Löschdatum wird auch in „tabÄnderungen“ eingetragen.



Einfügen des Kommentars über drag&drop unter oder über die jeweilige Aktion.



## 2.4 Makro 4: Zeitstempel für Datensatzaktualisierungen

Voraussetzung: Verweisbibliothek einschalten mit „Microsoft Scripting Runtime“

```
(Allgemein)
Option Compare Database

Function ZeitstempelMeldung()

Dim fso As New Scripting.FileSystemObject
Dim erstellt As Date, zugriff As Date, änderung As Date, typ As String
Dim anwendung As String
Dim drucker As String, dateipfad As String, ausgabe As String

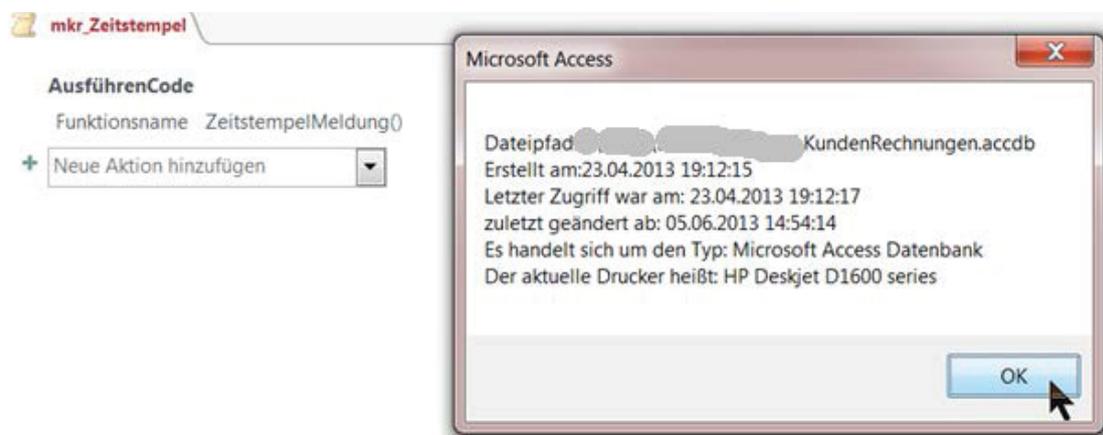
dateipfad = CurrentProject.Path & "\KundenRechnungen.accdb"
With fso.GetFile(dateipfad)
    erstellt = .DateCreated
    zugriff = .DateLastAccessed
    änderung = .DateLastModified
    typ = .Type
    anwendung = Application.Version
    drucker = Application.Printer.DeviceName

End With

ausgabe = "Dateipfad:" & dateipfad & vbCrLf & _
    "Erstellt am:" & erstellt & vbCrLf & _
    "Letzter Zugriff war am: " & zugriff & vbCrLf & _
    "zuletzt geändert ab: " & änderung & vbCrLf & _
    "Es handelt sich um den Typ: " & typ & vbCrLf & _
    "Der aktuelle Drucker heißt: " & drucker

MsgBox ausgabe

End Function
```



Makro-Aktion: AusführenCode als function

Download free eBooks at [bookboon.com](http://bookboon.com)

## 3 Makros mit Abfragen

### 3.1 Makro 5: Beispiel für ein Makro mit SQL-Abfrage, VBA-Code und Textexport

1. Zuerst bauen wir Abfrage über alle Felder einer Tabelle, z.B. unsere „Statusliste“

Aufgaben-ID	Aufgabe	Prozent	Status	Zum
2	PowerPoint Diagramme	34	offen	
3	Word Seriendruck	72	Abschlussphase	
4	Excel Pivots	100	geschlossen	
*	(Neu)			

```
SELECT Statusliste.[Aufgaben-ID], Statusliste.Aufgabe, Statusliste.Prozent, Statusliste.Status
INTO tab_Statusliste_Neu
FROM Statusliste;
```



Calling for Berlin  
Technology Advisory kennenlernen

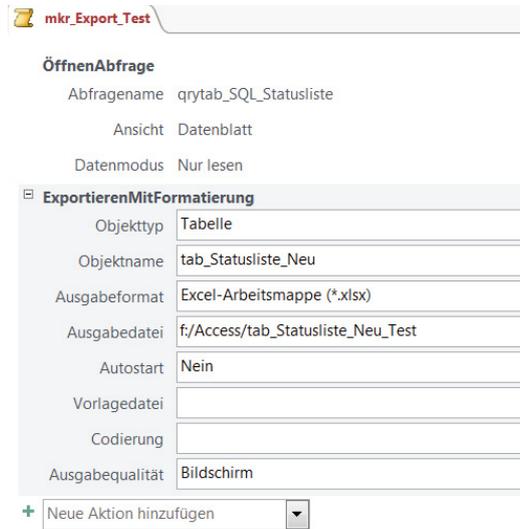
Consulting hautnah erleben  
5.–7. November 2015  
[www.deloitte.com/de/calling-for-berlin](http://www.deloitte.com/de/calling-for-berlin)

© 2015 Deloitte Consulting GmbH

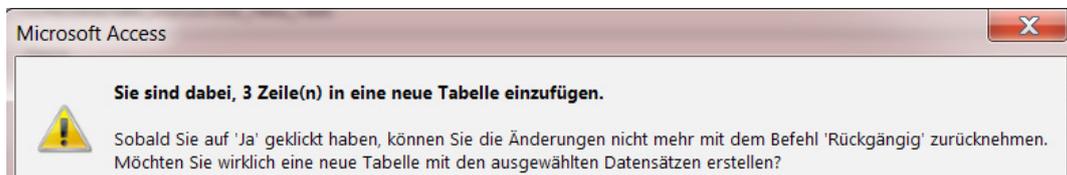




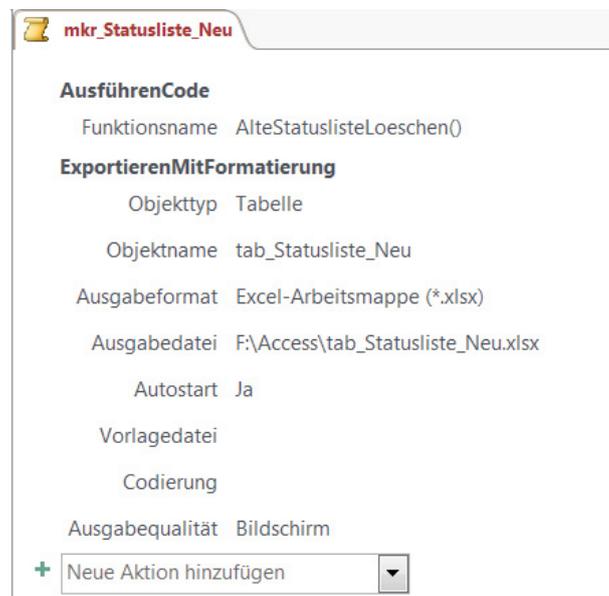
2. Erstellung eines vorläufigen Makros mit vielen Warnmeldungen „mkr\_Export\_Test“, nervig!



Und jetzt sollten ca. 4 Warnmeldungen angehen,...



## 3. Also was machen wir? Erstellung eines Makros „mkr\_Statusliste\_Neu“



**Warnmeldungen ausschalten** geht nur im alten mdb-Format, oder nur mit zweierlei:

- a. 1. Function: Das Vorhandensein der alten Statuslistentabelle wird auf true oder false gesetzt.

---

```
Function TabelleNeu(Tabellenname As String) As Boolean
Dim db As DAO.Database, tabelle As DAO.TableDef

Set db = CurrentDb()
Set tabelle = db.TableDefs(Tabellenname)

    For Each tabelle In db.TableDefs
        If tabelle.Name = db.TableDefs(Tabellenname).Name Then
            TabelleNeu = True
        Else
            TabelleNeu = False
        End If
    Next tabelle

End Function
```

---

- b. 2. Function: Wenn die Tabelle schon da ist, dann wird sie einfach gelöscht, wenn sie noch nicht da ist, dann wird sie mit der qrytab-Anweisung aus der Abfrage geöffnet.

```
Function AlteStatuslisteLoeschen()

    If TabelleNeu("tab_Statusliste_Neu") = True Then
        DoCmd.SetWarnings False
        DoCmd.RunSQL "DROP TABLE tab_Statusliste_Neu;"
        DoCmd.SetWarnings True
    Else
        DoCmd.SetWarnings False
        DoCmd.RunSQL "SELECT * INTO tab_Statusliste_Neu FROM
Statusliste;"
        DoCmd.SetWarnings True
    End If
End Function
```



**Mein Wissen rund um Big Data und SAP möchte ich sinnvoll einsetzen. Bin ich bei euch richtig, E.ON?**

**Lieber Herr Bennett, mit Ihren Fachkenntnissen können Sie bei uns viel bewegen.**

Bringen Sie Ihr Know-how in zukunftsweisende Projekte und Applikationen ein: Ob bei der energetischen Vernetzung von Smart Homes, der Steuerung virtueller Kraftwerke oder der Realisierung anspruchsvoller Logistik-Konzepte – der Energiesektor bietet vielfältige Herausforderungen für IT-Consultants, -Architekten und -Projektmanager. Entfalten Sie Ihre Kompetenz und geben Sie Ihrer Karriere neue Impulse.

Ihre Energie gestaltet Zukunft.

**top** ARBEITGEBER DEUTSCHLAND 2015  
CERTIFIED EXCELLENCE IN EMPLOYEE CONDITIONS

**e.on**

[www.eon-karriere.com](http://www.eon-karriere.com)



4. Ergebnis: Nur noch eine Warnmeldung erscheint, und zwar die Export-Aktion!



5. Was also tun? Na klar! Auch die Export-Aktion können wir in VBA verstecken lassen! Keine Warnmeldungen mehr da!



```
Function StatuslisteExportieren()
Dim strpfad As String

strpfad = "F:\Access\tab_Statusliste_Neu.xlsx"

    If Dir(strpfad) <> "" Then
        Kill (strpfad)
    Else
DoCmd.SetWarnings False
DoCmd.TransferSpreadsheetacExport, acSpreadsheetTypeExcel12, _
        "tab_Statusliste_Neu", strpfad, True
DoCmd.SetWarnings True
        End If
End Function
```

### 3.2 Makro 6: Bericht mit Meldungsinfo und PDF-Ausdruck

Zu diesem Makro muss ich gar nichts mehr sagen, einfach selber machen.

 **mk\_r\_KundenexportMitDrucken**

---

☐ **Wenn** Meldung("Was wollen Sie tun? Ja mit Excelexport in f:\Access\, Nein mit PDF im Mailanhang";4)=6  **Dann** 

**ExportierenMitFormatierung**

Objektyp Tabelle

Objektname tabKunden

Ausgabeformat Excel-Arbeitsmappe (\*.xlsx)

Ausgabedatei f:\Access\Kunden.xlsx

Autostart Nein

Vorlagedatei

Codierung

Ausgabequalität Bildschirm

+  

[Sonst Wenn hinzufügen](#)

☐ **Sonst**

**E-Mail-Datenbankobjekt**

Objektyp Tabelle

Objektname tabKunden

Ausgabeformat PDF-Format (\*.pdf)

An wordexpertin@yahoo.de

Cc

Bcc

Betreff test

Nachricht

Nachricht bearbeiten Ja

Vorlagedatei

**Ende Wenn**

### 3.3 Makro 7: FeldSTRUKTUR per VBA und SQL ändern, hinzufügen oder löschen

1. Erstellen Sie ein allgemeines Modul zum Einfügen folgender functions-Anweisungen:

```

[Allgemein]
FeldPerSQLAendern
Option Compare Database

Function FeldPerSQLAendern()
Dim db As DAO.Database

Set db = CurrentDb()

db.Execute "ALTER TABLE [Tab_Kunden] ALTER COLUMN PLZ Char(6) "
Set db = Nothing
End Function

Function FeldPerSQLHinzufuegen()
Dim db As DAO.Database

Set db = CurrentDb()

db.Execute "ALTER TABLE [Tab_Kunden] ADD COLUMN Sonstiges VarChar(200) "

Set db = Nothing
End Function

Function FeldPerSQLLoeschen()
Dim db As DAO.Database

Set db = CurrentDb()

db.Execute "ALTER TABLE [Tab_Kunden] DROP COLUMN Sonstiges"

Set db = Nothing
End Function

```

# 1

Ziel:

Du entwickelst unsere Zukunft.

Wir Deine.

IT-Traineeprogramm

In 18 Monaten durchläufst Du 3 verschiedene Stationen, wirst von einer Führungskraft als Mentor betreut und profitierst von einem breiten Seminarangebot. Anschließend kannst Du eine Fach- oder Führungslaufbahn einschlagen.

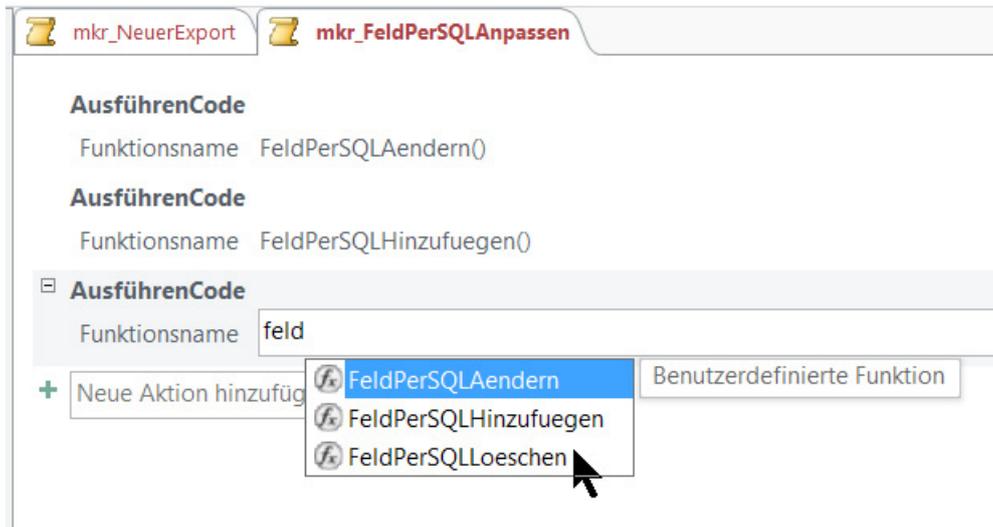
[www.perspektiven.allianz.de](http://www.perspektiven.allianz.de)

Allianz Karriere

## Allianz

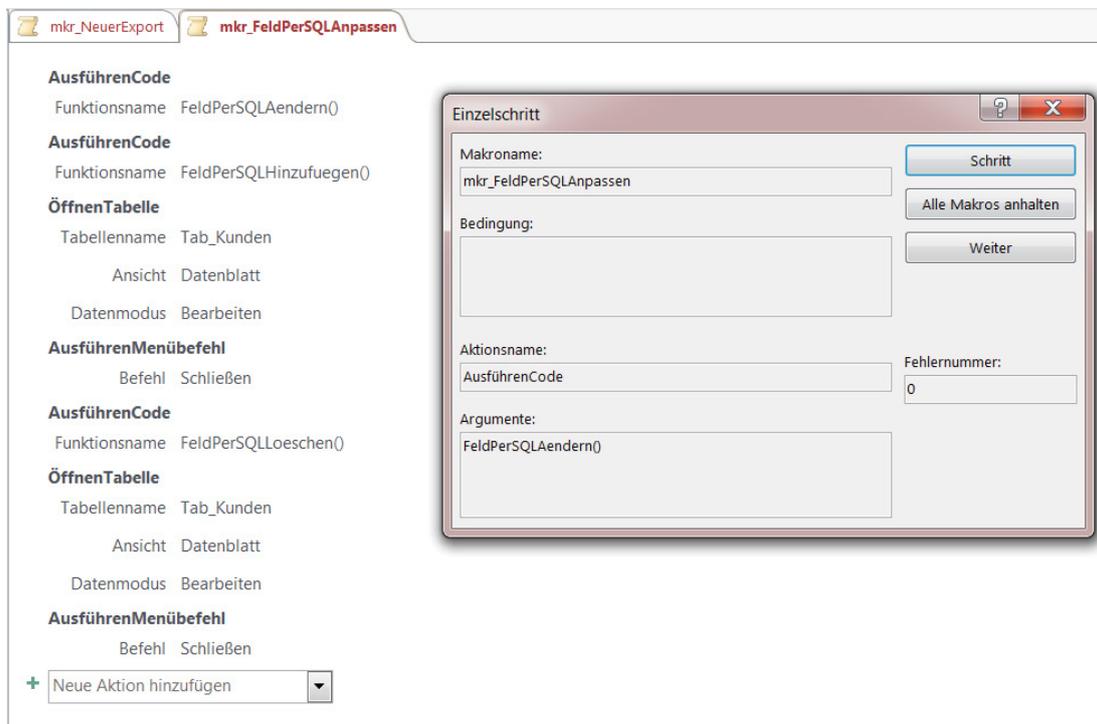


2. Schreiben Sie je ein Makro für jeden SQL-VBA-Code als functions.
3. Benutzen Sie auch die sehr praktische IntelliSense-Hilfe im Makro-Generator

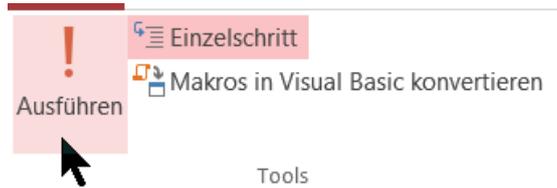


### 3.4 Makro 8: Makro-Aktionen im Einzelschritt testen

Testen Sie jede Makro-Aktion im Einzelschritt, indem Sie die „Tab\_Kunden“ öffnen und deren Strukturänderungen verfolgen.



Makrotools – Entwurf – Einzelschritt – Ausführen (F5)



### 3.5 Makro 9: DatenINHALTE über VBA, SQL und Makro ändern

PID	Pbezeichnung	Ppreis
1	Hose	29,90 €
2	Jacke	49,00 €
3	Hemd	39,00 €

Bei den Produkten gibt es eine saftige Preiserhöhung von 15%. Natürlich nur bei denen mit einem „H“ am Anfang, also Hose und Hemd. Die Ausgangstabelle „tabProdukte“:

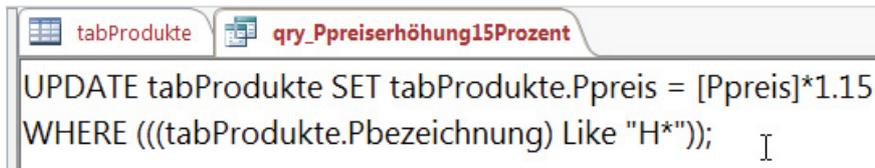
Datenmanipulation (DML)

1. Datenaktualisierungsabfrage erstellen, dann nur in der Datenblattansicht testen

Feld:	Pbezeichnung	Ppreis
Tabelle:	tabProdukte	tabProdukte
Aktualisieren:		[Ppreis]*1,15
Kriterien:	Wie "H*"	
oder:		

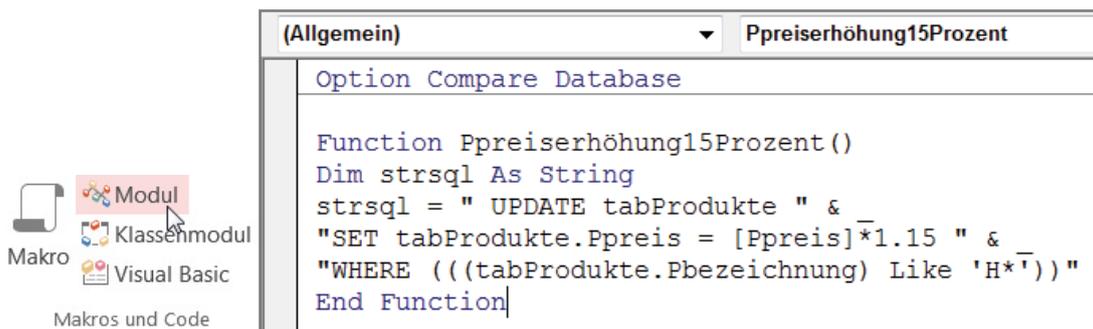
➔

2. Wechsel in die SQL-Ansicht, SQL-Code kopieren



```
UPDATE tabProdukte SET tabProdukte.Ppreis = [Ppreis]*1.15
WHERE (((tabProdukte.Pbezeichnung) Like "H*"));
```

3. Speichern in neue function und in ein neues allgemeines Modul.



```
Function Ppreiserhöhung15Prozent()
Dim strSQL As String
strSQL = " UPDATE tabProdukte " &
"SET tabProdukte.Ppreis = [Ppreis]*1.15 " &
"WHERE (((tabProdukte.Pbezeichnung) Like 'H*'))"
End Function
```

Dynamisches Abfragen über Parameter beißt sich leider, weil die VBA-Runtime im Code schon läuft, während der SQL-Code noch berechnet werden müsste. Knifflig.



## Sind Sie bereit für IBM?

Lieben Sie Herausforderungen?

Möchten Sie innovative Lösungen für führende Unternehmen entwickeln?

Wollen Sie dem weltweit größten Beratungsunternehmen angehören?

**Entdecken Sie Ihre vielfältigen Karrieremöglichkeiten.** IBM ist auf der Suche nach den besten und hellsten Köpfen. Nach Menschen, die Möglichkeiten entdecken, wo andere nur Probleme sehen. Nach Mitarbeitern, die auch Mitgestalter sein wollen. Wir suchen diese Menschen aus dem Anspruch heraus, die Welt täglich ein bisschen besser zu machen. Sie sind ideengetrieben, zukunftsorientiert und möchten schon heute an den Lösungen von morgen arbeiten? Dann sollten wir uns kennenlernen!

Machen wir den Planeten ein bisschen smarter.  
[ibm.com/start/de](http://ibm.com/start/de)

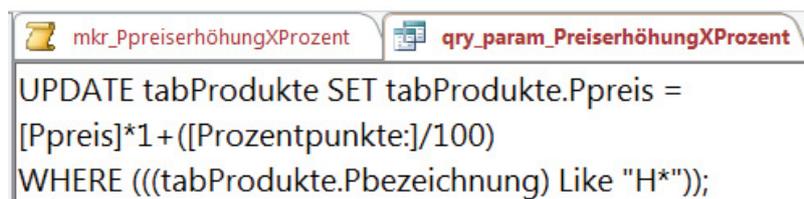
Alle Bezeichnungen, die in der männlichen Sprachform verwendet werden, schließen sowohl Frauen als auch Männer ein. IBM schafft ein offenes und tolerantes Arbeitsklima und ist stolz darauf, ein Arbeitgeber zu sein, der für Chancengleichheit steht. IBM, das IBM Logo und ibm.com sind Marken oder einget. Marken der International Business Machines Corp. in den Vereinigten Staaten und/oder anderen Ländern. Andere Namen von Firmen, Produkten und Dienstleistungen können Marken oder eingetr. Marken ihrer jeweiligen Inhaber sein. © 2010 IBM Corp. Alle Rechte vorbehalten.



Die Aktualisierungsabfrage mit dem Parameter in der Entwurfsansicht:

Feld:	Ppreis	Pbezeichnung
Tabelle:	tabProdukte	tabProdukte
Aktualisieren:	[Ppreis]*1+([Prozentpunkte:]/100)	
Kriterien:		Wie "H*"

Und in der SQL-Ansicht:



The screenshot shows two tabs: 'mkr\_PpreiserhöhungXProzent' and 'qry\_param\_PreiserhöhungXProzent'. The active query tab contains the following SQL code:

```
UPDATE tabProdukte SET tabProdukte.Ppreis =
[Ppreis]*1+([Prozentpunkte:]/100)
WHERE (((tabProdukte.Pbezeichnung) Like "H*"));
```

meine Lösung für den 2. function-Code könnte so aussehen:

```
Function PpreiserhöhungXProzent()
Dim strSQL As String, db As DAO.Database

strSQL = "UPDATE tabProdukte SET " & _
"tabProdukte.Ppreis = [Ppreis]*1+([Prozentpunkte:]/100) " & _
"WHERE (((tabProdukte.Pbezeichnung) Like 'H*'))"

On Error GoTo fehler
Set db = CurrentDb()
db.Execute (strSQL)
Set db = Nothing
Exit Function

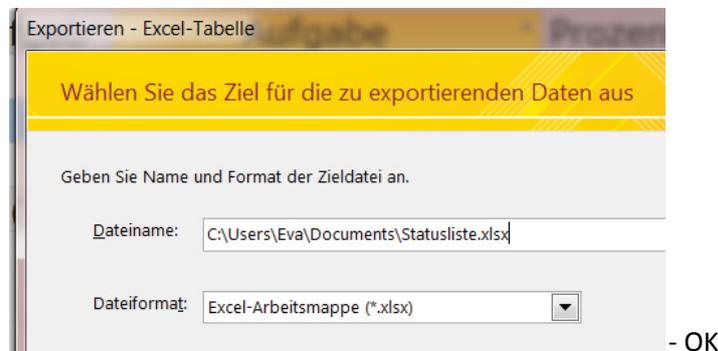
fehler:
'MsgBox Err.Number & vbCrLf & Err.Description
If Err.Number = 3061 Then
DoCmd.SetWarnings False
DoCmd.RunSQL (strSQL)
DoCmd.SetWarnings True
End If
Exit Function
|
End Function
```

### 3.6 Makro 10: The easy way – Exportvorgänge einfach speichern!

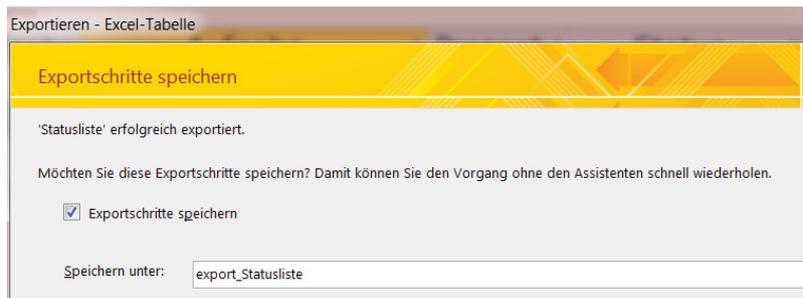
In Makro 4 haben wir bereits die Tabelle samt PLZ (Text-6 Zeichen) geändert. Jetzt sollten wir mal die „Statusliste“ mit Exportspezifikation zum Export vorbereiten. Register Externe Daten – Exportieren – Typ anklicken.



Speicherort auswählen



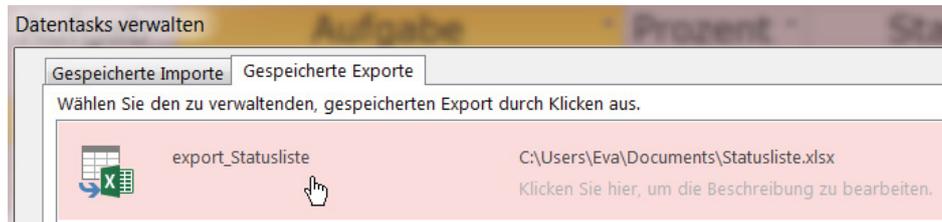
Kästchen „Exportschritte speichern“ anhängen: z.B. „**export-Statusliste**“ – Export speichern



Gespeicherte Exporte immer einsehbar unter Register „Externe Daten“ – Exportieren



Sowohl der Export-Name als auch der Speicherort lassen sich bearbeiten.



### 3.6.1 Eine Tabelle nach Excel mit TransferText exportieren

```
Sub TabelleNachExcelTransferieren()
```

```
Docmd.transferspreadsheet acExport, acspreadsheetTypeExcel12, "Statusliste", _  
Application.currentproject.path & "\Statusliste.xlsx"
```

## JETZT BEWERBUNG AUFPOLIEREN.

Bereiten Sie sich optimal auf den Bewerbungsprozess vor und geben Sie Ihrem Profil den letzten Schliff. Nutzen Sie unsere Tipps, Persönlichkeitstests und kostenlosen E-Books zu Studium, Business und Karriere.





VORWEG GEHEN



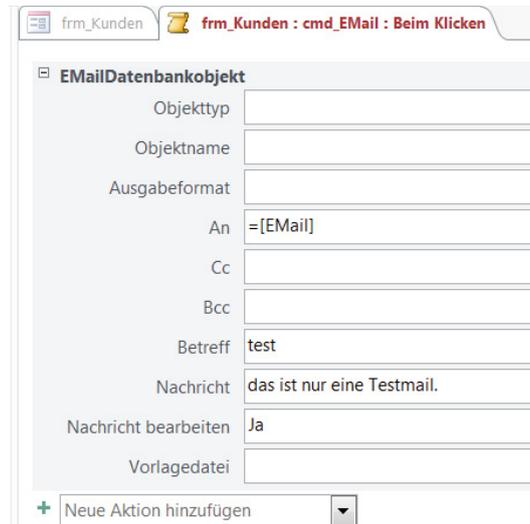
### 3.6.2 Export nach Excel mit Exportspezifikation (mdb oder accdb)

```
Sub Export_tabKundenAusfuehren()  
Dim excapp As Excel.Application, wb As Workbook, ws As Worksheet  
Dim umgebung As String  
umgebung = Environ("userprofile")  
    On Error GoTo fehler  
        DoCmd.RunSavedImportExport "Export_tabKunden"  
    Set excapp = GetObject(, "Excel.Application")  
anfang:  
    Set excapp = CreateObject("Excel.Application")  
    Set wb = excapp.Workbooks.Open (umgebung & "/Documents/tabKunden.xlsx")  
  
    excapp.Visible = True  
    wb.Activate  
    Set ws = wb.Worksheets(1)  
    ws.Rangee("A1").Select  
  
    MsgBox "ExcelDatei ist offen!"  
excapp.Quit  
  
Set ws = Nothing  
Set wb = Nothing  
Set excapp = Nothing  
Exit Sub  
  
fehler:  
    If Err.Number = 429 And Err.Number = 0 Then  
        Err.Number = 0  
    End If  
    GoTo anfang  
Exit Sub  
  
End Sub
```

# 4 Makros in Formularen

## 4.1 Makro 11: Senden einer E-Mail mit Schaltfläche

Versuch mit eingebettetem Makro über Schaltfläche: **E-Mail-Datenbankobjekt**



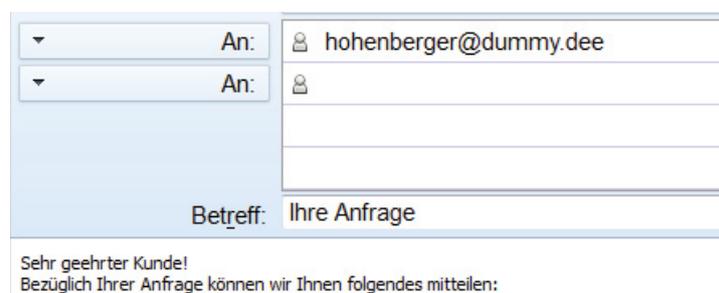
Besser: Schaltfläche – Ereignis: Beim Klicken – Code-Generator

```
Private Sub cmd_Email_Click()

If Me.Email <> "" Then
DoCmd.SendObject , , , Me.Email
End If

End Sub
```

**Besser: Application.FollowHyperlink (mit Betreff und Mailtext!)**



```
Private Sub cmd_Mail_Click()  
Dim strBetreff As String, strBody As String  
  
strBetreff = "Ihre Anfrage"  
MsgBox strBetreff  
strBody = "Sehr geehrter Kunde! %0ABezüglich Ihrer Anfrage können wir  
Ihnen folgendes mitteilen:%0A"  
    If Me.Mail <> "" Then  
        'DoCmd.SendObject , , , Me.Mail, , , "Hallo!"  
        Application.FollowHyperlink "mailto:" & Me.Mail & "?subject=" &  
            strBetreff & "&body=" & strBody  
    End If  
End Sub
```



© 2013 Accenture.  
All rights reserved.

be > your degree

Bring your talent and passion to a global organization at the forefront of business, technology and innovation. Discover how great you can be.

Visit [accenture.com/bookboon](http://accenture.com/bookboon)

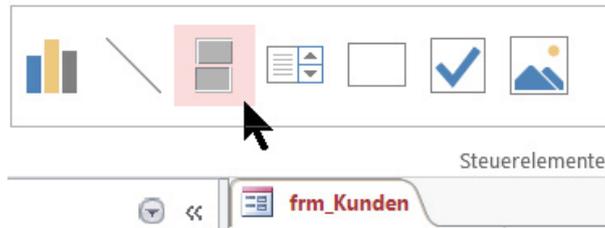
Be greater than.  
consulting | technology | outsourcing

accenture  
High performance. Delivered.



## 4.2 Makro 12: ChangeButton (Umschaltfläche) zum Sortieren

Formularentwurfstools – Steuerelemente – Umschaltfläche



The image shows the Access Macro Editor for the 'frm\_Kunden' form. The macro is named 'frm\_Kunden : chb\_AufAb : Beim Klicken'. The macro steps are as follows:

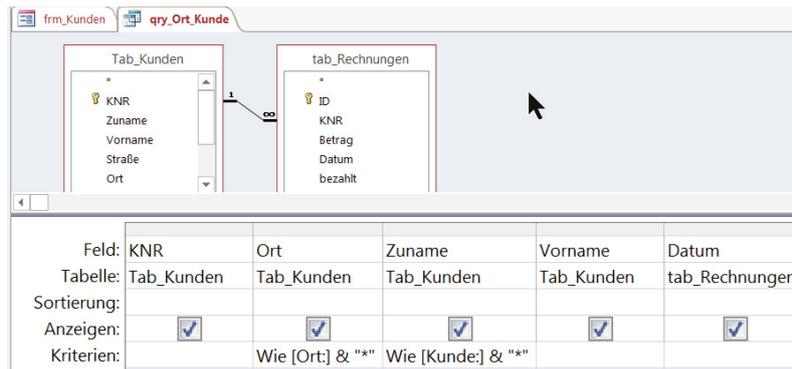
- GeheZuSteuerelement**
  - Steuerelementname = [cbo\_sortieren]
- Wenn** [chb\_AufAb].[Beschriftung] = "Aufsteigend" **Dann**
  - AusführenMenübefehl**
    - Befehl: SortierenAufsteigend
  - FestlegenEigenschaft**
    - Steuerelementname: chb\_AufAb
    - Eigenschaft: Beschriftung
    - Wert: Absteigend
- Sonst**
  - AusführenMenübefehl**
    - Befehl: SortierenAbsteigend
  - FestlegenEigenschaft**
    - Steuerelementname: chb\_AufAb
    - Eigenschaft: Beschriftung
    - Wert: Aufsteigend
- Ende Wenn**

At the bottom, there is a button labeled '+ Neue Aktion hinzufügen' with a dropdown arrow.

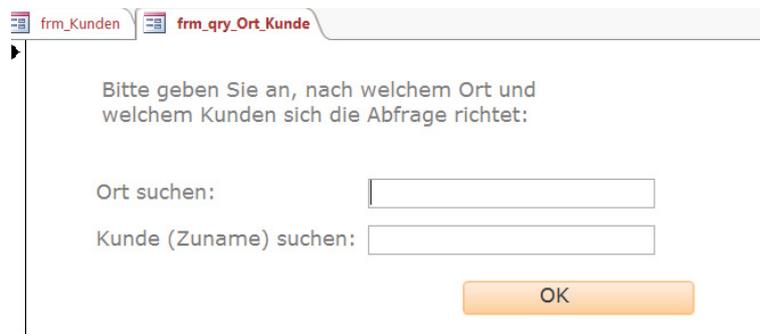
### 4.3 Makro 13: „Maske? Welche Maske?“

3 neue Objekte brauchen wir: Zuerst Formular aus einer Tabelle, dann Parameter-Abfrage erstellen, zuletzt Abfrageformular erstellen. Vorgehen: Das Ausgangs-Formular öffnet das Abfrageformular, die Felder des Abfrageformulars lösen (Klick: OK) die Parameter-Abfrage aus.

1. Wir erstellen uns eine Parameter-Abfrage, für Ort und Zuname der gesuchten Kunden.

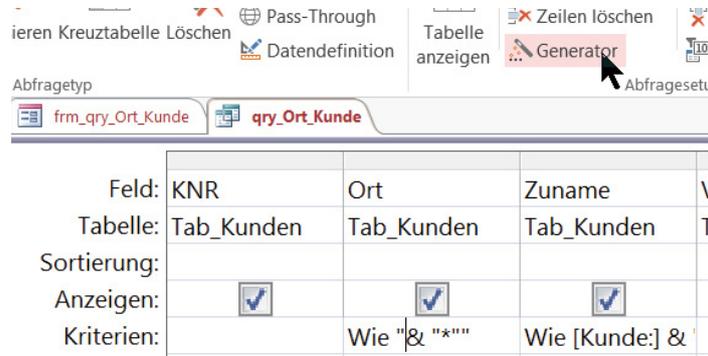


2. Jetzt basteln wir uns das Formular zur Eingabe der Abfragekriterien.



txt\_OrtSuchen, txt\_ZunameSuchen, cmd\_OK

3. Dann wechseln wir in den Entwurf der qry\_Kunde\_Ort und tauschen die Parameter aus durch den Pfad von Formular!Datenfeld, einfach über den Ausdrucks-Generator zusammenklicken!



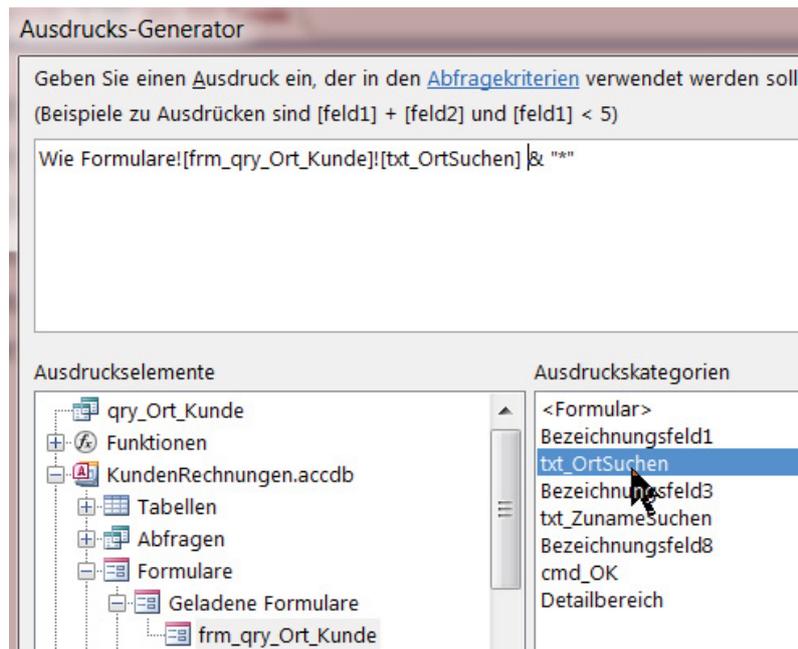
**McKinsey & Company**

**Start your engines.**

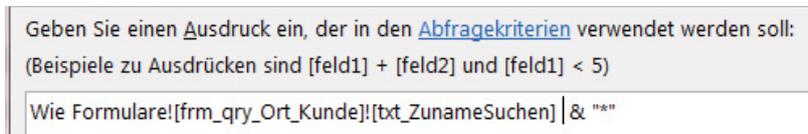
McKinsey sucht Ingenieure. Nutzen Sie Ihr Potenzial und starten Sie durch.

Mehr auf [mckinsey.de/ingenieure](http://mckinsey.de/ingenieure)

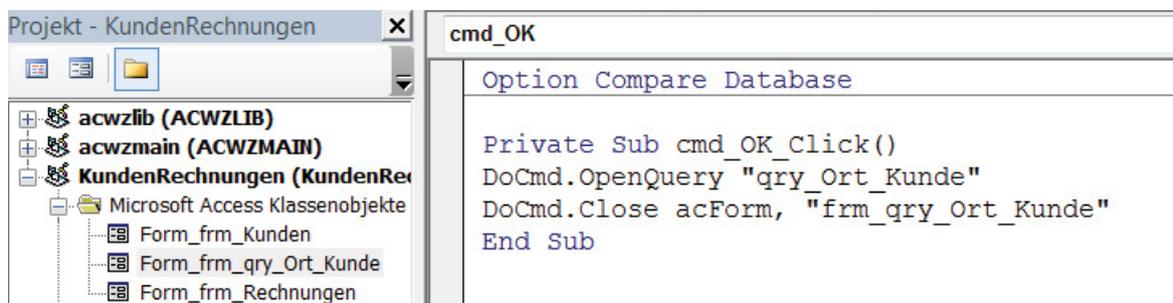




- Bestätigen mit OK, dann das Gleiche nochmal mit dem Kunden-Kriterium austauschen:



- OK, Abfrage speichern und schließen.
- Abfrageformular auch speichern, hier im Entwurf folgenden Code eingeben:



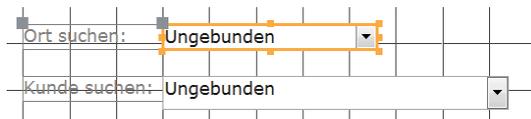
Ort suchen:

Kunde (Zuname) suchen:

7. Über frm\_Kunden“ wird das Abfrageformular mit einer Schaltfläche gestartet. Wir geben Kriterien ein, z.B. „H“ wie „Hallein“ und mit OK wird die Abfrage „qry\_Ort\_Kunde“ geöffnet. Gleich danach wird das Auswahlformular „frm\_qry\_Ort\_Kunde“ geschlossen.

KNR	Ort	Zuname	Vorname	Datum
2000	Hallein	Hohenberger	Josef	17.10.2005
2000	Hallein	Hohenberger	Josef	18.09.2005

#### 4.4 Makro 14: Suchen in Formularen



Erstellen wir uns zuerst wieder im Formulkopf einen Button zur Orts-Suche und einen für die Kundensuche. Interessant ist, dass ja in einem Ort mehrere Kunden wohnen. Das zweite Ergebnis soll als auf die Ergebnisse des ersten Kombifeldes aufbauen.

#### Eigenschaftenblatt

Auswahltyp: Kombinationsfeld

cboOrtSuchen

Format    Daten    Ereignis    Andere    Alle

Format	
Dezimalstellenanzeige	Automatisch
Sichtbar	Ja
Spaltenanzahl	1
Spaltenbreiten	3cm
Spaltenüberschriften	Nein

Man beachte die Spaltenanzahl beim 1. Kombifeld mit 1 Spalte zum 1. Suchbegriff nach Ort und beim 2. Kombifeld mit 2 Spalten, nämlich für Kundennummer und Kundennamen.

#### Eigenschaftenblatt

Auswahltyp: Kombinationsfeld

cboKundeSuchen

Format    Daten    Ereignis    Andere    Alle

Format	
Dezimalstellenanzeige	Automatisch
Sichtbar	Ja
Spaltenanzahl	2
Spaltenbreiten	0cm;3cm
Spaltenüberschriften	Nein

**Zusatz-Tipp:**

Damit das Kombifeld beim ersten Anzeigen (Current) des Formulars nicht komplett weiß bleibt, empfiehlt sich der Kniff mit der UNION-Abfrage für nicht den ersten sondern den nullten Datensatz innerhalb der Ortsliste. Der nullte Eintrag („Bitte Ort auswählen!“) kann von uns über die UNION-Anweisung in einer Spalte beliebig gefüllt werden.

```

Form
Option Compare Database
Option Explicit
Private Sub Form_Current()
Dim strSQL As String

Me.cboOrtSuchen.RowSource = ""
Me.cboKundeSuchen.RowSource = ""
strSQL = "select 'Bitte Ort auswählen' as Ort from [tab_Kunden] " & _
"UNION select distinct Ort from [tab_Kunden]"

Me.cboOrtSuchen.RowSource = strSQL
Me.cboOrtSuchen.Value = "Bitte Ort auswählen"
Me.cboKundeSuchen.Value = 0
End Sub
    
```

Ort suchen:

Kunde suchen:

**IELTS** UNIVERSITY OF CAMBRIDGE **TOEFL iBT**

**GEWINNE EINEN  
SPRACHKURS IN MIAMI MIT  
EXAMENSVORBEREITUNG**

Bereite Dich mit EF Sprachreisen auf ein international anerkanntes Sprachzertifikat wie TOEFL, Cambridge oder IELTS vor.

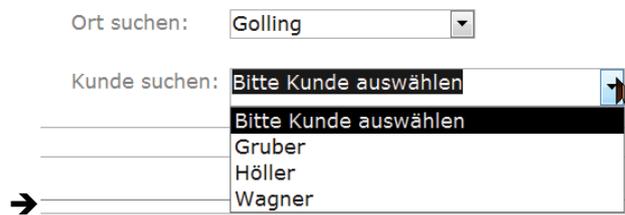
[www.ef.com/bookboon](http://www.ef.com/bookboon)

**JETZT TEILNEHMEN!**

Education First



Sobald wir jetzt einen Ort ausgewählt haben, wird die Formularanzeige im 2. Feld aktiv, indem die RowSource (Datensatzquelle) sich nach dem Ort richtet.



Und weil's so schön war mit dem UNION-Kniff, hab ich das nochmal über das Ereignis „Nach Aktualisierung“ im cbo-Feld nochmal eingebaut.

Schließlich wird über die FindFirst-Methode die tatsächliche Kundennummer (KNR) des ausgesuchten Kundennamens ermittelt.

```
Private Sub cboOrtSuchen_AfterUpdate()
Dim strSQL As String

Me.cboKundeSuchen.RowSource = ""
strSQL = "select 0 as KNR, 'Bitte Kunde auswählen' as Zuname FROM [tab_Kunden] " & _
"UNION SELECT KNR, Zuname from [tab_Kunden] where Ort =" & Me.cboOrtSuchen & ""

Me.cboKundeSuchen.RowSource = strSQL
Me.cboKundeSuchen.Value = 0

End Sub
```

---

```
Private Sub cboKundeSuchen_AfterUpdate()

Me.Recordset.FindFirst "KNR=" & Me.cboKundeSuchen.Column(0)

End Sub
```

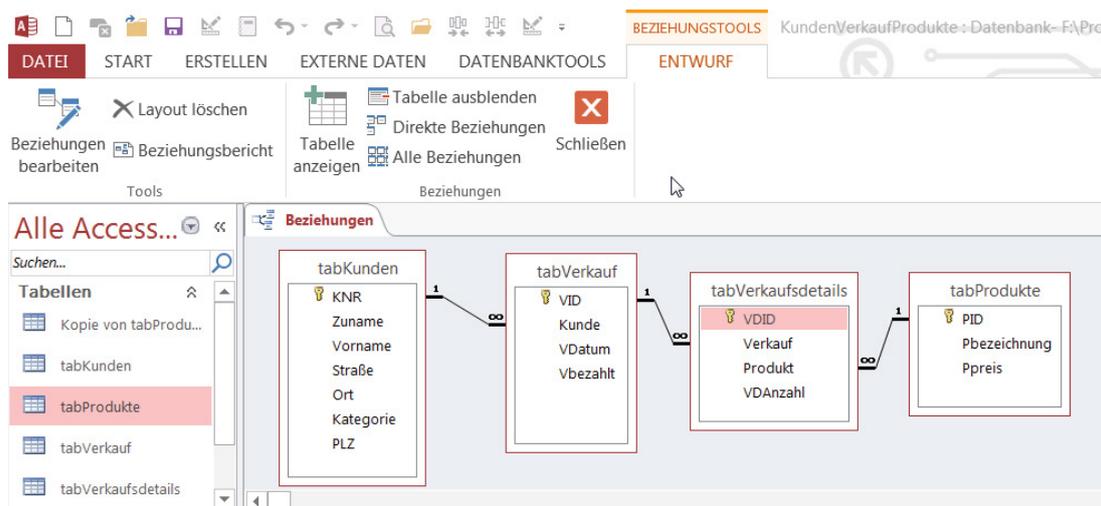
Also haben wir hier 3 Prozeduren eingebaut:

1. Die Formularanzeige zum Einfüttern der Ortsliste im 1. Kombifeld
2. Das Einfüttern des 2. Kombifeldes für die Kundenliste
3. Das Ermitteln des gesuchten Kunden-Datensatzes anhand seiner Kundennummer.

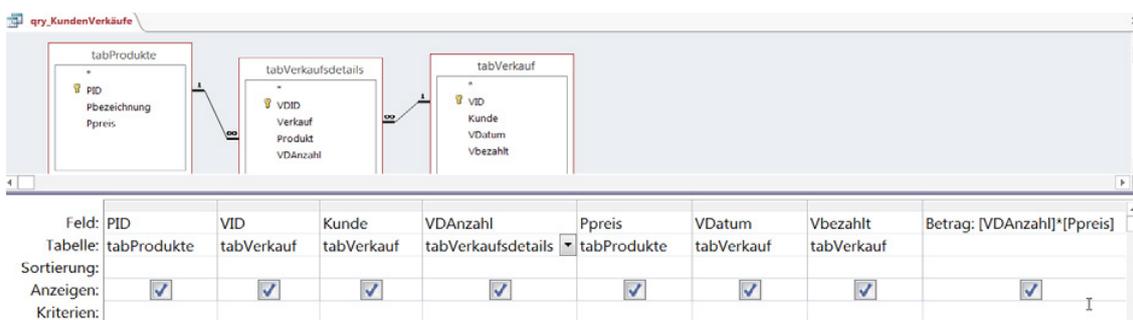
# 5 Makro 15: Filtern in Formularen

## 5.1 Die Beziehungen zu den neuen Tabellen

Alle neuen Tabellen müssen miteinander wieder in Relation gebracht werden. Klicken Sie auf Register Datenbanktools – Beziehungen



Ich brauche für jeden Kunden (Hauptformular) alle Verkäufe (Unterformular), nebst Verkaufsdetails und Produktpreis. Also erstellen wir zuerst eine Abfrage qryKundenVerkäufe.

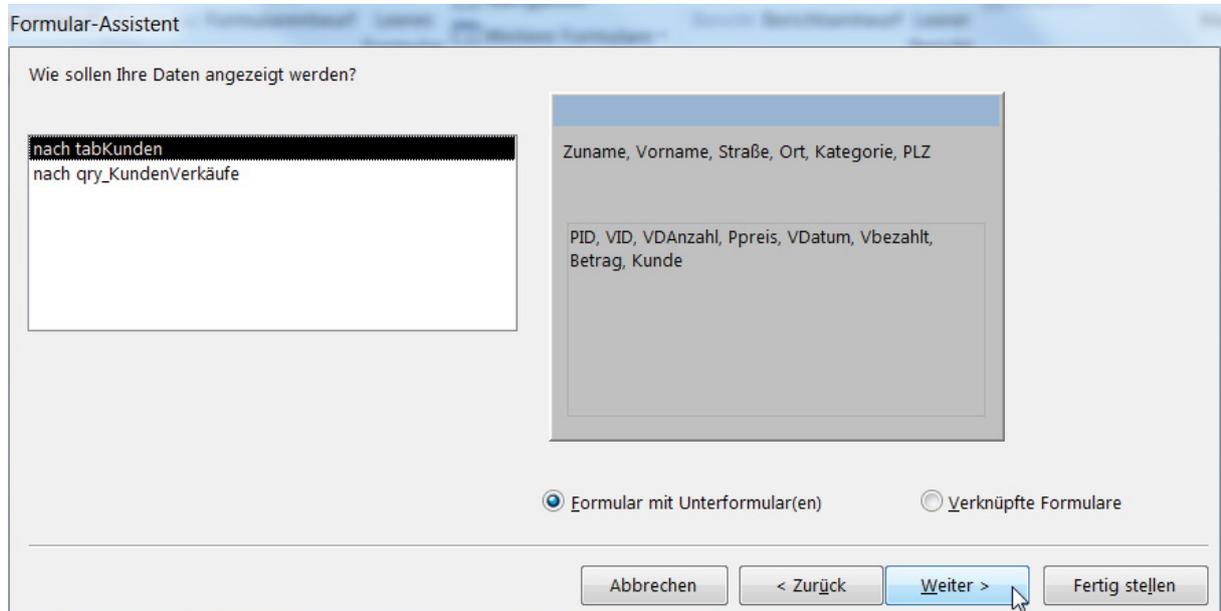


Ergebnis in Datenblatt-Ansicht:

PID	VID	Kunde	VDAnzahl	Ppreis	VDatum	Vbezahlt	Betrag
1	1	2000	1	30,95 €	10.04.2013	Ja	30,95 €
2	1	2000	1	49,00 €	10.04.2013	Ja	49,00 €
3	2	2001	1	40,05 €	12.04.2013	Ja	40,05 €
*							

Dann erstellen wir aus tabKunden(Tabelle) und der qryKundenVerkäufe (Abfrage) alle Felder außer KNR von tabKunden.

Starten des Assistenten über Reg. Erstellen – Formulare – FormularAssistent



**START UP – MEHR ALS EIN  
TRAINEE-PROGRAMM.  
JETZT BEWERBEN!**

Die Antwort auf fast alles.  
Antworten auf Ihre Karrierefragen finden  
Sie hier: [www.telekom.com/absolventen](http://www.telekom.com/absolventen)

Jetzt bewerben!

**T . . .**

**ERLEBEN, WAS VERBINDET.**



Ergebnis:

The screenshot shows the 'frmKunden' form with the following data:

Zuname	Hohenberger
Vorname	Josef
Straße	Kirchengasse 3
Ort	Hallein
Kategorie	1
PLZ	5400

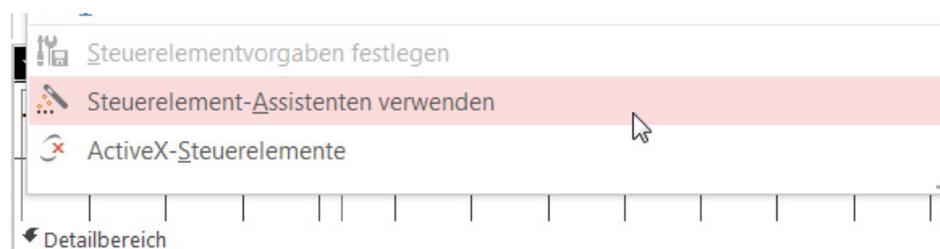
  

PID	VID	VDAn	Ppreis	VDatum	Vbezahlt	Betrag	Kunde
1	1	1	30,95 €	10.04.2013	Ja	30,95 €	2000
2	1	1	49,00 €	10.04.2013	Ja	49,00 €	2000

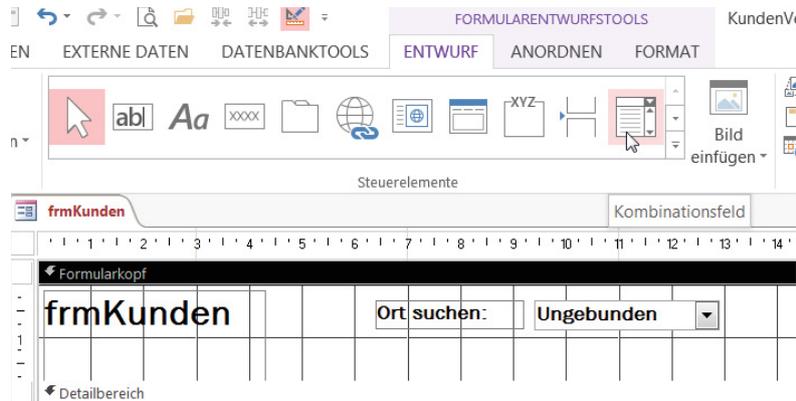
Speichern wir also das frmKunden als Hauptformular und das darin eingebettete ufrmKundenVerkäufe als Unterformular.

## 5.2 Ein Kombifeld zum Anzeigen aller Orte (SQL)

Durch STRG + ENTER können wir das neue frmKunden auch gleich im Entwurf öffnen. Im Formularkopf fügen wir ein Kombinationsfeld ohne Schaltflächen-Assistent ein.



Formularentwurfstools – Entwurf – Steuerelemente – Kombinationsfeld.



### Machen Sie die Zukunft sichtbar

**Kleine Chips, große Wirkung:** Heute schon sorgt in rund der Hälfte aller Pässe und Ausweise weltweit ein Infineon Sicherheitscontroller für den Schutz ihrer Daten. Gleichzeitig sind unsere Halbleiterlösungen der Schlüssel zur Sicherheit von übermorgen. So machen wir die Zukunft sichtbar.

**Was wir dafür brauchen?** Ihre Leidenschaft, Kompetenz und frische Ideen. Kommen Sie zu uns ins Team! Freuen Sie sich auf Raum für Kreativität und Praxiserfahrung mit neuester Technologie. Egal ob Praktikum, Studienjob oder Abschlussarbeit: Bei uns nehmen Sie Ihre Zukunft in die Hand.

**Für Studierende und Absolventen (w/m):**

- > Ingenieurwissenschaften
- > Naturwissenschaften
- > Informatik
- > Wirtschaftswissenschaften



[www.infineon.com/karriere](http://www.infineon.com/karriere)



charta der vielfalt



Eigenschaften (im Entwurf irgendwo klicken auf die Taste F4), Datensatzherkunft eintippen:



**SELECT Kunden.Ort FROM Kunden GROUP BY Kunden.Ort;**

Ergebnis in der Formularansicht:



### 5.3 Filter aller Kunden nach Auswahl eines Ortes (VBA)



## Alternative 1: Filtern mit VBA-Filtern

```
Private Sub cboOrtSuchen_AfterUpdate()
```

```
DoCmd.ShowAllRecords
```

```
Me.Filter = "Ort=" & Me!cboOrtSuchen & ""
```

```
Me.FilterOn = True
```

```
End Sub
```

## Alternative 2: Filtern mit der Makro/VBA-Aktion DoCmd.ApplyFilter

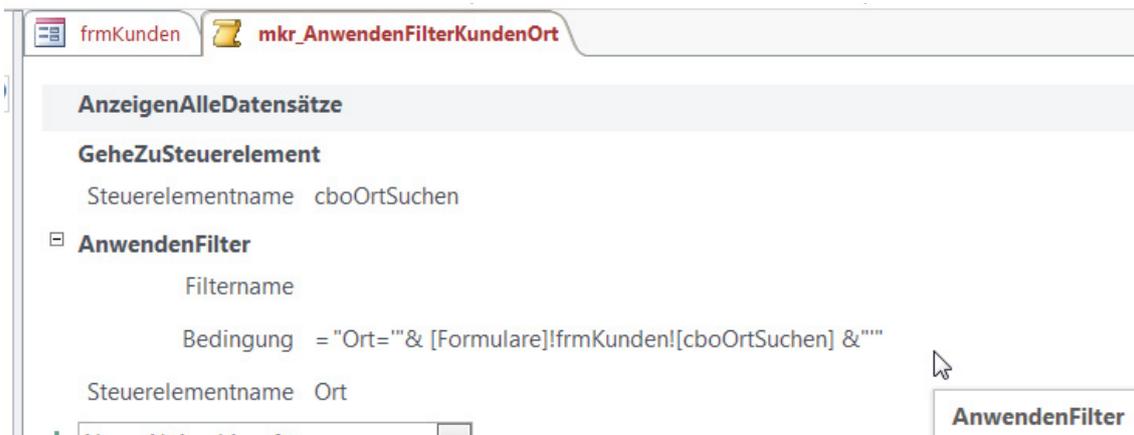
```
Private Sub cboOrtSuchen_AfterUpdate()
```

```
DoCmd.ShowAllRecords
```

```
DoCmd.ApplyFilter , "Ort=" & Me.cboOrtSuchen & ""
```

```
End Sub
```

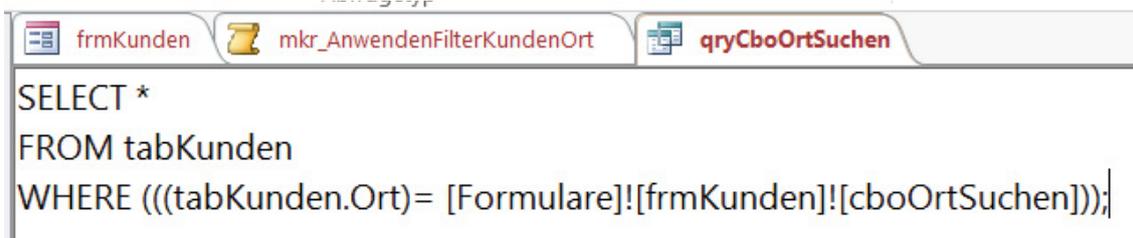
## Alternative 3: Filtern ohne VBA, mit Makro-Aktion AnwendenFilter



## 5.4 Filtern über Makro mit Abfragekriterien

Etwas Anderes ist es, einfach statt dem „AnwendenFilter“ erst mal alle Filterkriterien in einer neuen Abfrage zu versammeln.

In der SQL-Ansicht könnte das so aussehen:



```
SELECT *
FROM tabKunden
WHERE (((tabKunden.Ort)= [Formulare]![frmKunden]![cboOrtSuchen]));
```

Dann brauche ich eine Schaltfläche (command button = cmd) zum Ausführen der Abfrage mittels Kombifeld. Freuen wir uns auch jedes Mal über die neue IntelliSense-Hilfe zum Eintippen über TAB-Taste der vorgeschlagenen Objekt-Namen. Noch nie war das Zusammenklicken eines Objektpfades so einfach wie heute!



**SIEMENS**

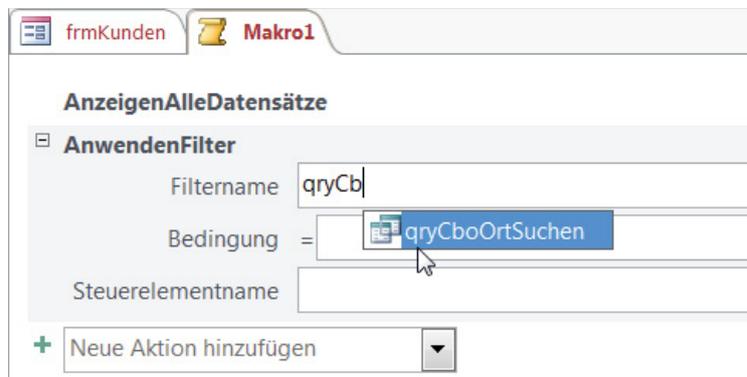
EIGENVERANTWORTUNG  
KREATIVE TEAMPLAYER  
NEUGIERDE  
OFFENHEIT  
INNOVATION ERFINDERGEIST  
ENGAGEMENT  
PERSPEKTIVEN CHANCEN  
ENTSCHLOSSENHEIT  
WELTWEITE MÖGLICHKEITEN  
WORK-LIFE-BALANCE

Verwirklichen, worauf es ankommt –  
mit einer Karriere bei Siemens.

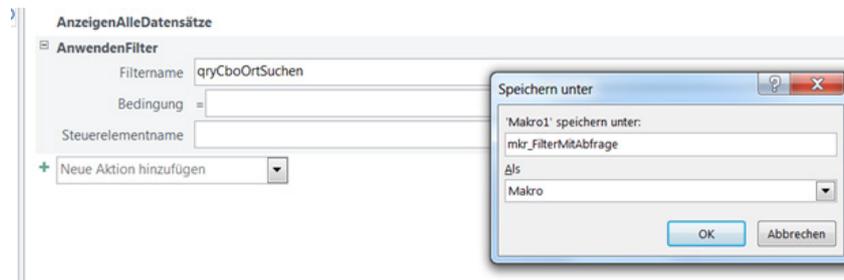
siemens.de/karriere



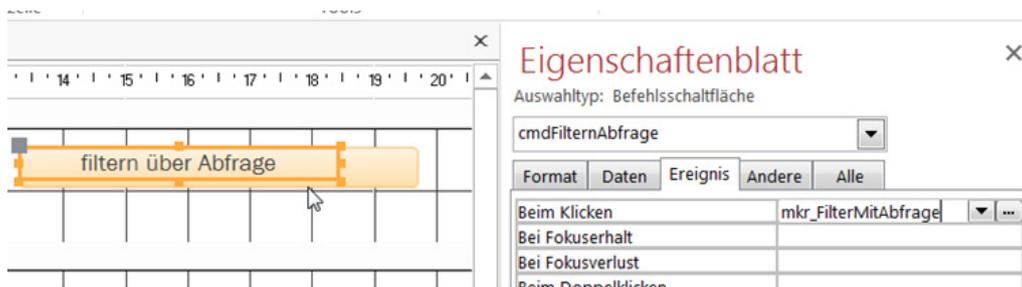
2 Makroaktionen sind dabei immer empfehlenswert: Erst mal im Formular alle Datensätze anzeigen lassen, dann den Filter anwenden.



Ach ja, ein schnelles Speichern des neuen Makros ist mit F12 jederzeit möglich, für alle Objekte.



Dann schließen wir unser neues Makro, bauen uns wieder im Formularentwurf eine Schaltfläche ein, suchen das Makro über das Ereignis „Beim Klicken“ aus und haben das auch geschafft.



Fürs weitere Rumbasteln und Makrobauen wünsche ich viel Spaß!

Freundliche Grüße von

Eva Vogel  
(MCT)  
Nürnberg

[www.vogel-trainings.de](http://www.vogel-trainings.de)

Download free eBooks at [bookboon.com](http://bookboon.com)

## 6 Endnoten

- 1 Andreas Thehos bei Youtube: „XML-Oberflächengestaltung mit Access“
- 2 Lorenz Hölscher: „Richtig einsteigen: Access 2010 VBA-Programmierung“, MS Press/O'Reilly, Köln, 2010



Jonas von Malottki Finance Accounting IT Solutions, Deutschland (Stuttgart)  
Hortense Denise Kirby HR Business Partner, USA (Dallas/Fort Worth)  
Yu Chang Engineering Support Office, China (Peking)

Fünf Kontinente. Jede Menge Platz zur persönlichen Entfaltung. Das sind wir.

Hier geht es für Sie weiter: [www.career.daimler.com](http://www.career.daimler.com)

**DAIMLER**

Die Daimler AG ist eines der erfolgreichsten Automobilunternehmen der Welt. Zum Markenportfolio gehören Mercedes-Benz, smart, Freightliner, Western Star, BharatBenz, Fuso, Setra, Thomas Built Buses sowie die Mercedes-Benz Bank, Mercedes-Benz Financial und Truck Financial.

