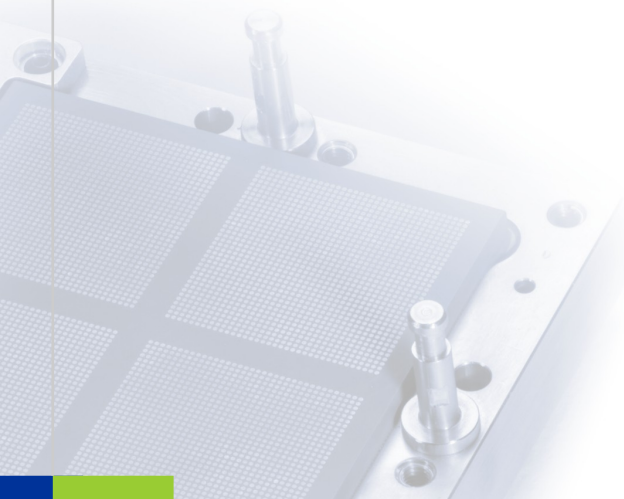


VBA (Visual Basic for Application)

Einführung

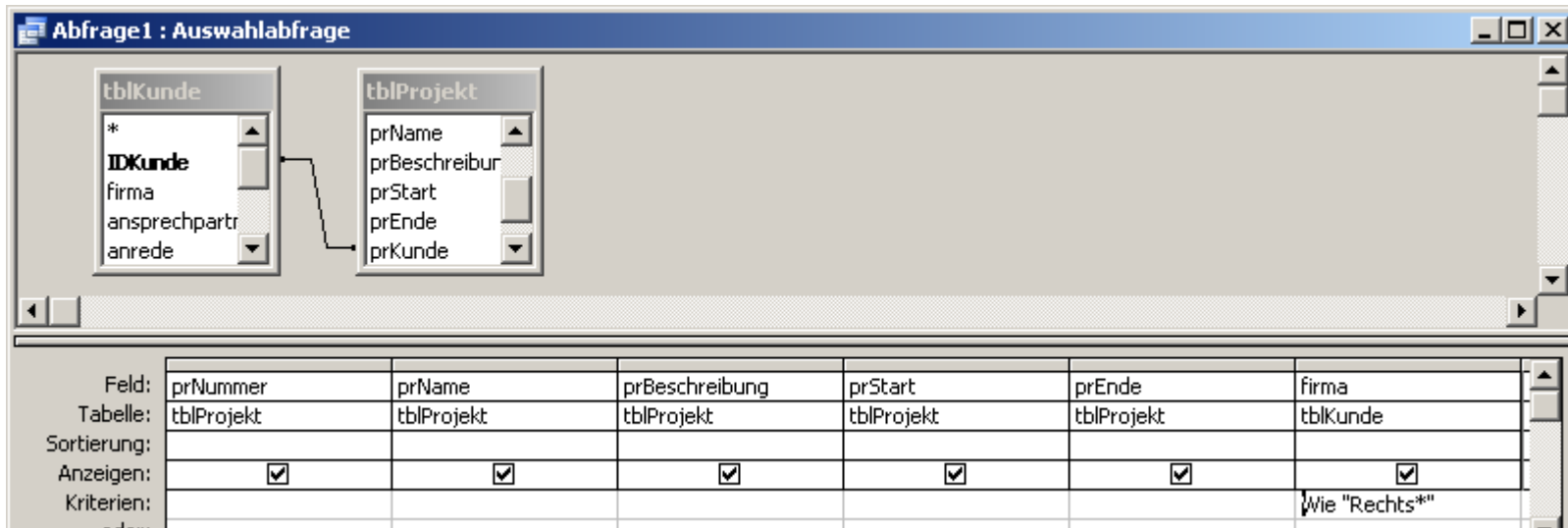


- Doberenz & Kowalski: Microsoft Office Access 2003 Programmierung; Microsoft Press
- André Minhorst: Das Access 2003-Entwicklerbuch; Addison-Wesley
- Albrecht & Nicol: Access 2003 programmieren; Addison-Wesley
- Harkins & Gunderloy: Automating Microsoft Access mit VBA; Que

- ... zu Access und VBA:
 - <http://www.donkarl.com/>
 - <http://www.achim-roehl.de/ACCESS/access.htm>
 - <http://support.microsoft.com/default.aspx/ph/2509?sid=251>
 - <http://msdn.microsoft.com/library/deu/default.asp?url=/library/DEU/modcore/html/deovrDevelopingOfficeSolutions.asp>
 - http://www.access-beispiele.tommyk-webbox.de/code/code_sonstiges.htm
 - http://www.access-o-mania.de/index.php?ind=tipps&op=section_view&idev=2
 - <http://www.mvps.org/access/index.html>
- Auflistung von Funktionen etc.
 - <http://www.ott-joachim.de/access/access-vba-funktionen.html>
- ... zu VBA, allgemein:
 - http://cis.cs.tu-berlin.de/Lehre/SS-01/Sonstiges/dbprapages/ausarbeitungen/vba/a9_vba.pdf

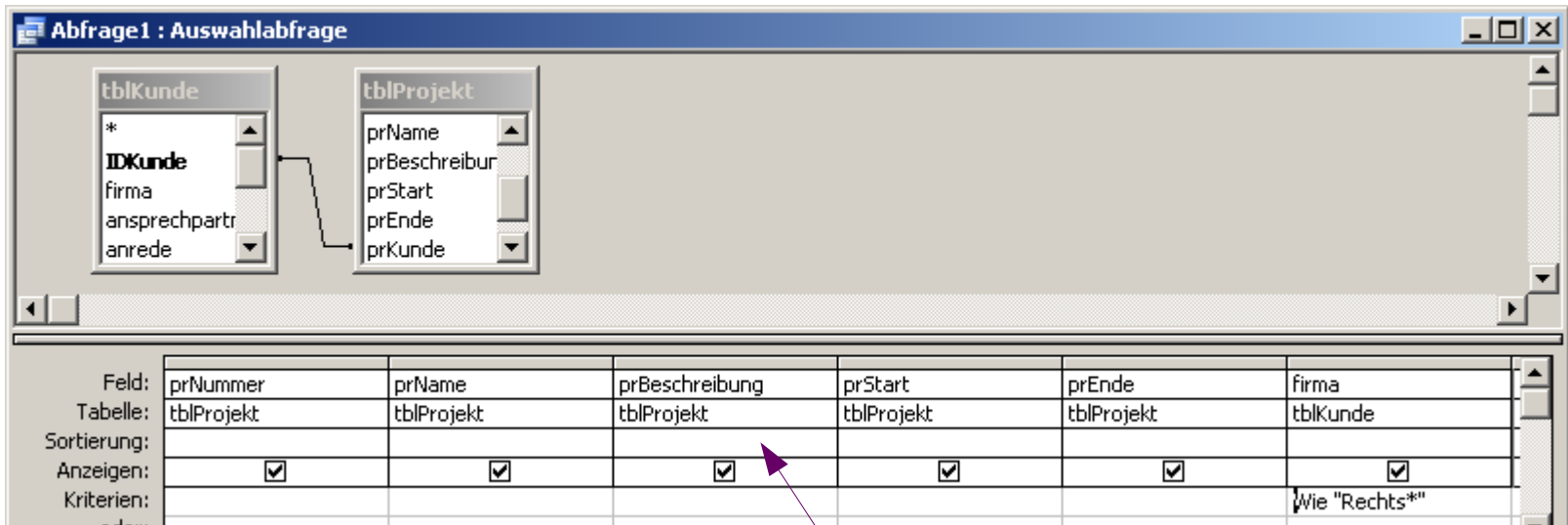
- ... ist Bestandteil der Microsoft Office-Familie.
- ... ist eine relationale Datenbank.
- ... ist ein Datenbankmanagementsystem.
 - ... verwaltet große Mengen von Daten, die in Tabellen gespeichert werden.
 - ... stellt Werkzeuge zur Auswertung, Analyse, Bearbeitung und Präsentation von Daten bereit.
 - ... kann Informationen aus verschiedenen Tabellen mit Hilfe von SQL verknüpfen.
 - ... kann mit Hilfe von Makros Arbeitsschritte automatisieren.
 - ... kann mit Hilfe von VBA programmiert werden.

- ... ist eine Sprache zur Abfrage und Manipulation von Daten in einer relationalen Datenbank.
- ... bietet Lösungen für Fragen in der Form "Suche mir alle Zeilen in einer Tabelle, die ein bestimmtes Kriterium erfüllen" an.
- ... wird
 - ... intern für die Darstellung von Abfragen genutzt.
 - ... zur Beschreibung einer Datenquelle eines Formulars, Berichts oder Steuerelements (Controls) genutzt.
 - ... zur Bereitstellung von verschiedenen Sichten (Views) auf die Daten genutzt. Zum Beispiel: Der Sachbearbeiter, der Bestellungen von Kunden aufnimmt, bekommt die Informationen zu dem Kunden und zu den möglichen Artikeln angezeigt. Der Lagerverwalter bekommt zu jeder Ware die Anzahl der im Lager vorhandenen Artikel angezeigt und kann dementsprechend die Bestellung zusammenstellen.



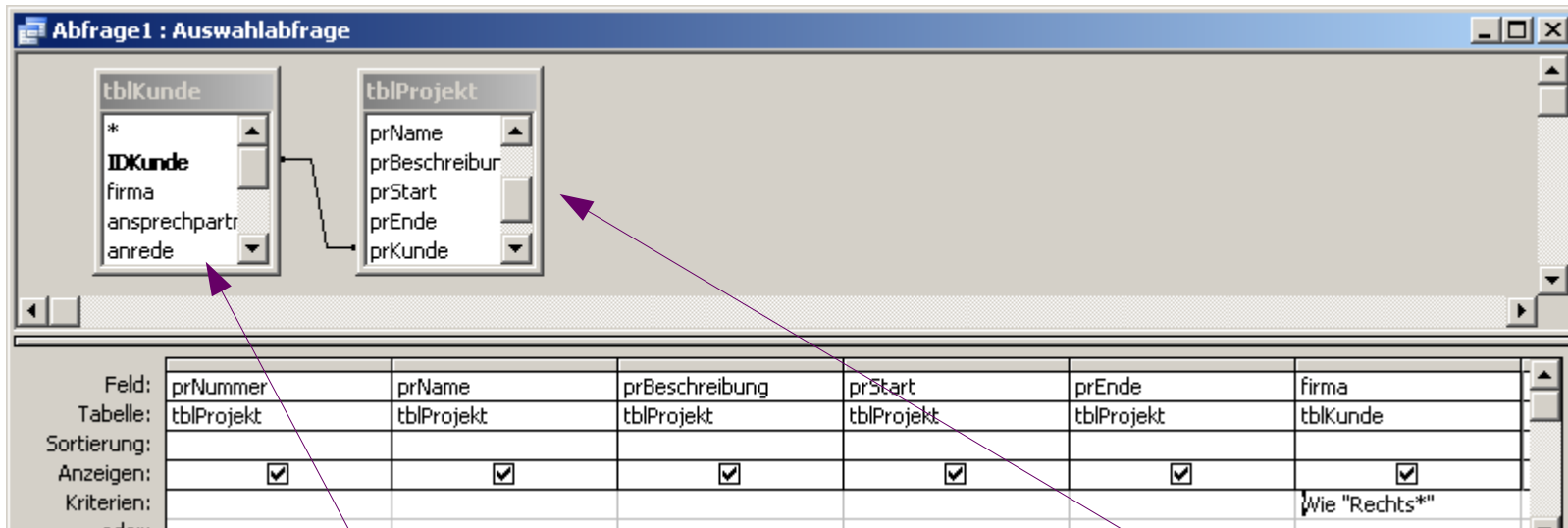
```

SELECT tblProjekt.prNummer, tblProjekt.prName, tblProjekt.prBeschreibung,
       tblProjekt.prStart, tblProjekt.prEnde, tblKunde.firma
FROM tblKunde
INNER JOIN tblProjekt ON tblKunde.IDKunde = tblProjekt.prKunde
WHERE (((tblKunde.firma) Like "Rechts*"));
    
```



Welche Daten werden angezeigt?

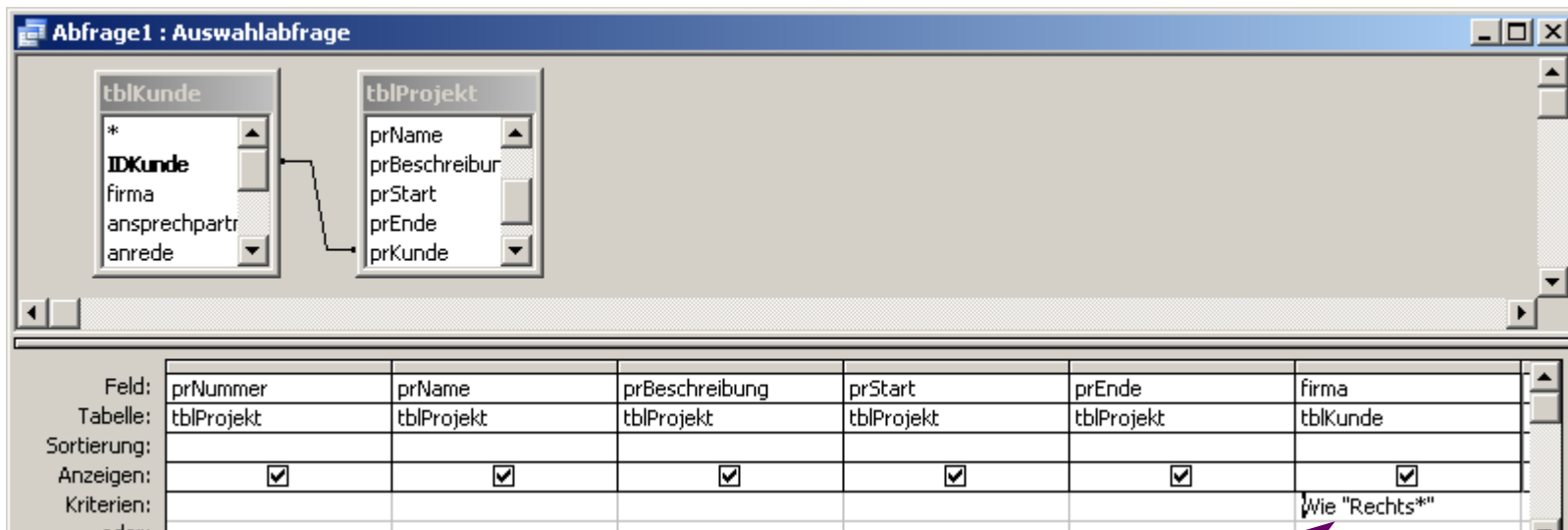
```
SELECT tblProjekt.prNummer, tblProjekt.prName, tblProjekt.prBeschreibung,
       tblProjekt.prStart, tblProjekt.prEnde, tblKunde.firma
FROM tblKunde
INNER JOIN tblProjekt ON tblKunde.IDKunde = tblProjekt.prKunde
WHERE (((tblKunde.firma) Like "Rechts*"));
```



Aus welcher Tabelle kommen die Daten?

Und mit welcher ist diese verknüpft?

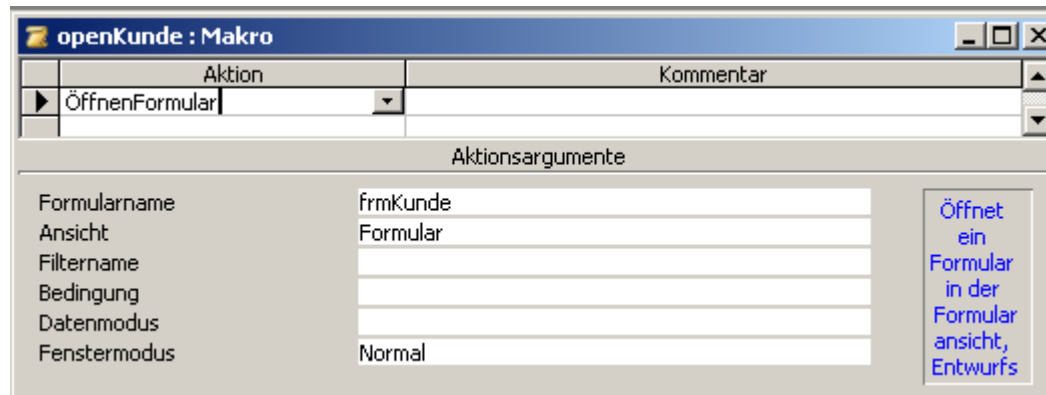
```
SELECT tblProjekt.prNummer, tblProjekt.prName, tblProjekt.prBeschreibung,  
tblProjekt.prStart, tblProjekt.prEnde, tblKunde.firma  
FROM tblKunde  
INNER JOIN tblProjekt ON tblKunde.IDKunde = tblProjekt.prKunde  
WHERE (((tblKunde.firma) Like "Rechts*"));
```



Welche Bedingungen müssen die Daten erfüllen?

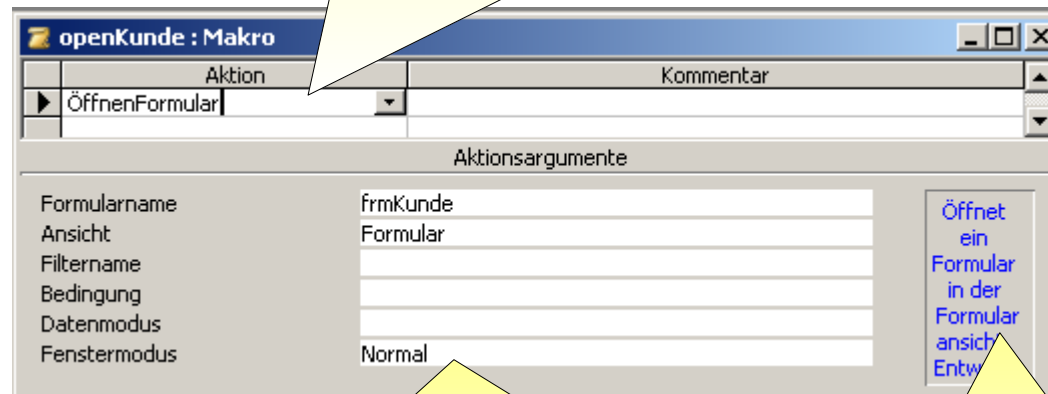
```

SELECT tblProjekt.prNummer, tblProjekt.prName, tblProjekt.prBeschreibung,
       tblProjekt.prStart, tblProjekt.prEnde, tblKunde.firma
FROM tblKunde
INNER JOIN tblProjekt ON tblKunde.IDKunde = tblProjekt.prKunde
WHERE (((tblKunde.firma) Like "Rechts*"));
  
```



- ... automatisieren häufig wiederholende Arbeitsschritte.
- ... fassen Aktionen zusammen, die dann von oben nach unten abgearbeitet werden.
- ... haben einen begrenzten Befehlsumfang.
- ... benötigen für den Start immer die gleichen Bedingungen.
- ... können beliebig oft von verschiedenen Stellen aufgerufen werden.
- ... wird intern als VBA interpretiert.

Die Liste des Kombinationsfeldes enthält alle Befehle. Mit Hilfe des schwarzen Pfeils nach unten wird die Liste geöffnet und mit einem Mausklick wird eine Aktion ausgewählt.



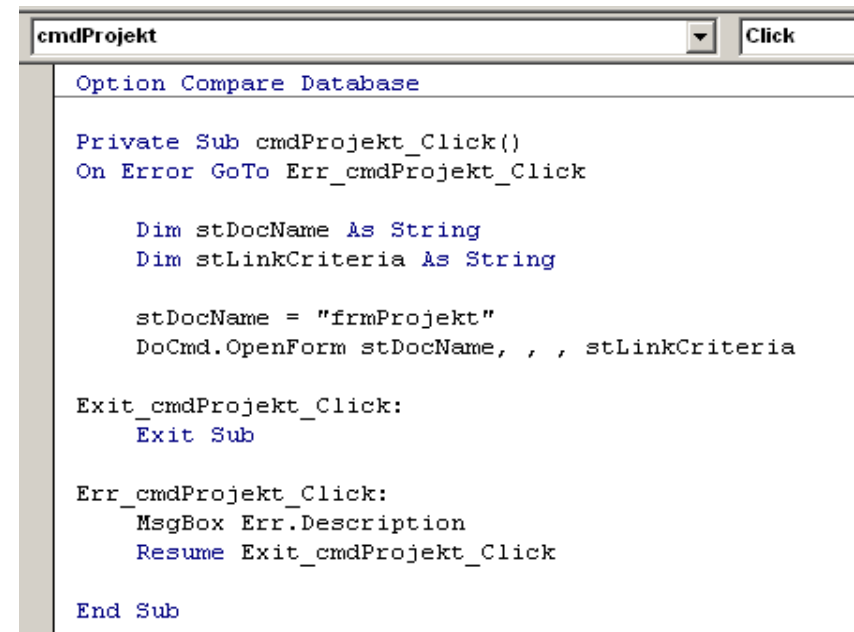
In Abhängigkeit der Aktion können Argumente gesetzt werden. Für die Aktion *ÖffnenFormular* wird ein Formularname sowie die Art der Darstellung übergeben.

In Abhängigkeit des Mauszeigers wird eine Hilfe angezeigt.

- Zuerst wird das Objekt Makros im Datenbankfenster ausgewählt.
- Mit Hilfe der Symbol-Schaltfläche Neu wird ein neues Makro angelegt.
- Pro Zeile können Sie eine Aktion auswählen.
- Stellen Sie die verschiedenen Argumente für die ausgewählte Aktion ein.
- Speichern Sie das Makro.
- Mit Hilfe des Ausrufezeichens in der Symbolleiste kann das Makro ausgeführt werden.

- Daten für die Anzeige in Formularen oder Berichten filtern.
- Daten auf Plausibilität prüfen.
- Daten in Steuerelementen neu berechnen und anzeigen.
- Daten aktualisieren und speichern.
- Daten automatisch importieren.
- Formulare und Berichte automatisiert öffnen.
- Standardeinstellung werden beim Öffnen der Datenbank festgelegt.

- ... wird für die Programmierung von Anwendungssoftware aus dem Office-Paket eingesetzt.
- ... kann in Word, Excel, Access, PowerPoint oder Outlook genutzt werden,
- ... ist an Visual Basic angelehnt.
- ... wird in einem speziellen Editor geschrieben.
- ... ermöglicht eine strukturierte, prozedurale Programmierung.
- VBA-Programme
 - ... laufen nur innerhalb einer Office-Umgebung.
 - ... wissen nicht in welcher Umgebung sie laufen.



```
cmdProjekt Click
Option Compare Database

Private Sub cmdProjekt_Click()
On Error GoTo Err_cmdProjekt_Click

    Dim stDocName As String
    Dim stLinkCriteria As String

    stDocName = "frmProjekt"
    DoCmd.OpenForm stDocName, , , stLinkCriteria

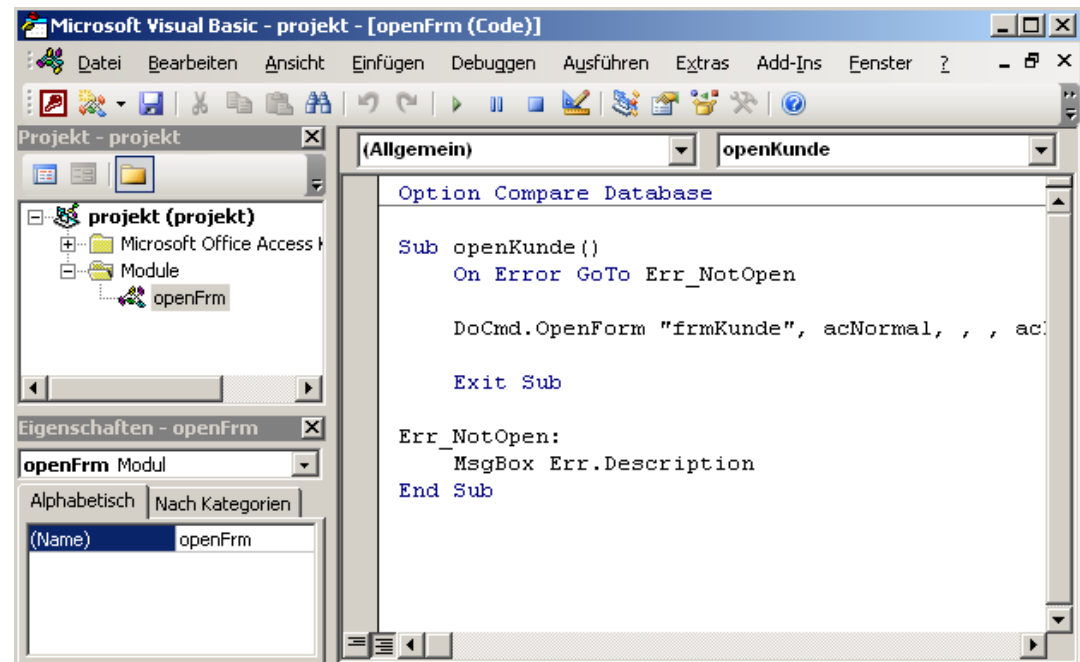
Exit_cmdProjekt_Click:
    Exit Sub

Err_cmdProjekt_Click:
    MsgBox Err.Description
    Resume Exit_cmdProjekt_Click

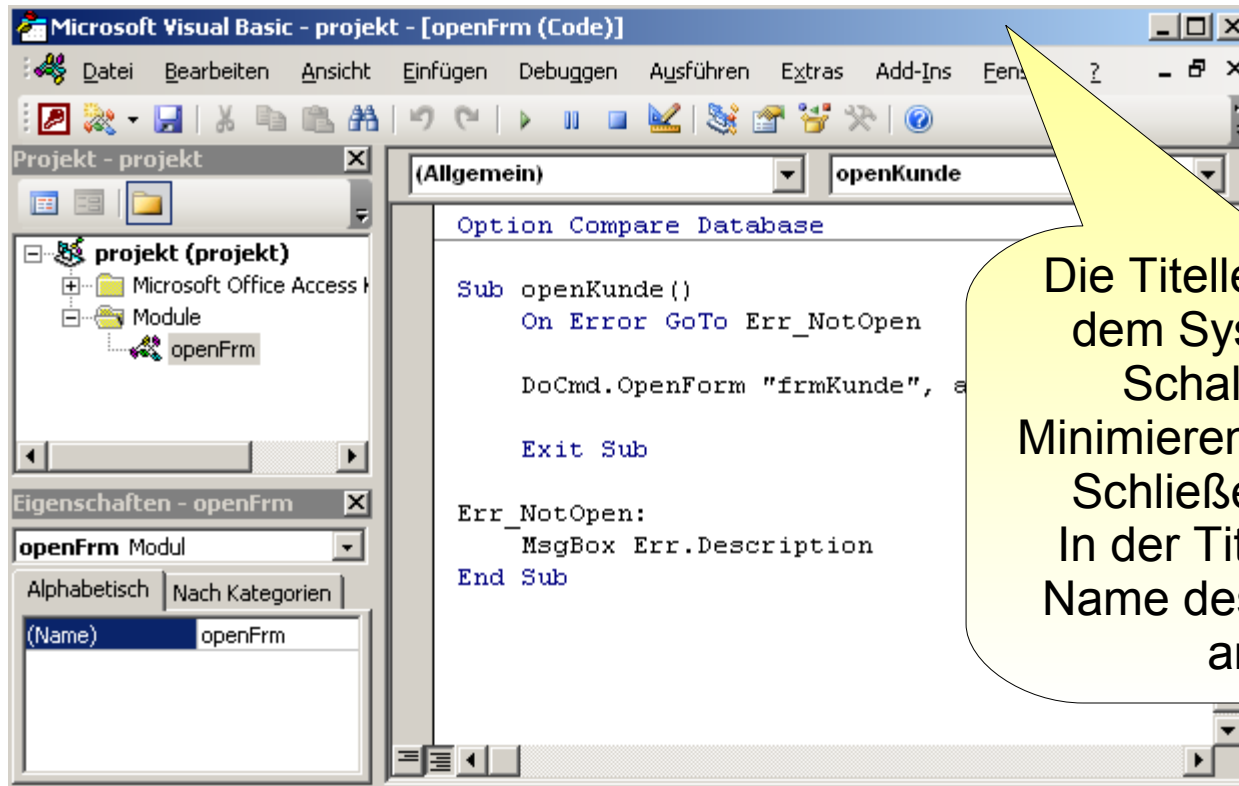
End Sub
```

- Formulare, Berichte und Steuerelemente gestalten. Zum Beispiel:
 - Fehlerhafte Eingaben werden rot markiert.
 - In Abhängigkeit einer Auswahl werden Steuerelemente ein- oder ausgeblendet.
- Es kann auf Benutzeraktionen reagiert werden. Zum Beispiel
 - Der Benutzer klickt auf eine Schaltfläche.
 - Es wird ein Ereignis (Aktion) "Mausklick" ausgelöst und der passende Bericht wird geöffnet.
- Daten
 - ... werden für die Anzeige in Formularen oder Berichten gefiltert.
 - ... werden auf Plausibilität geprüft. Falls ein Fehler auftritt, wird eine Meldung an den Benutzer ausgegeben.
 - ... in Steuerelementen werden neu berechnen oder automatisch ergänzt.
 - ... werden automatisch aktualisiert und gespeichert.
- Integration von anderen Office-Anwendungen.

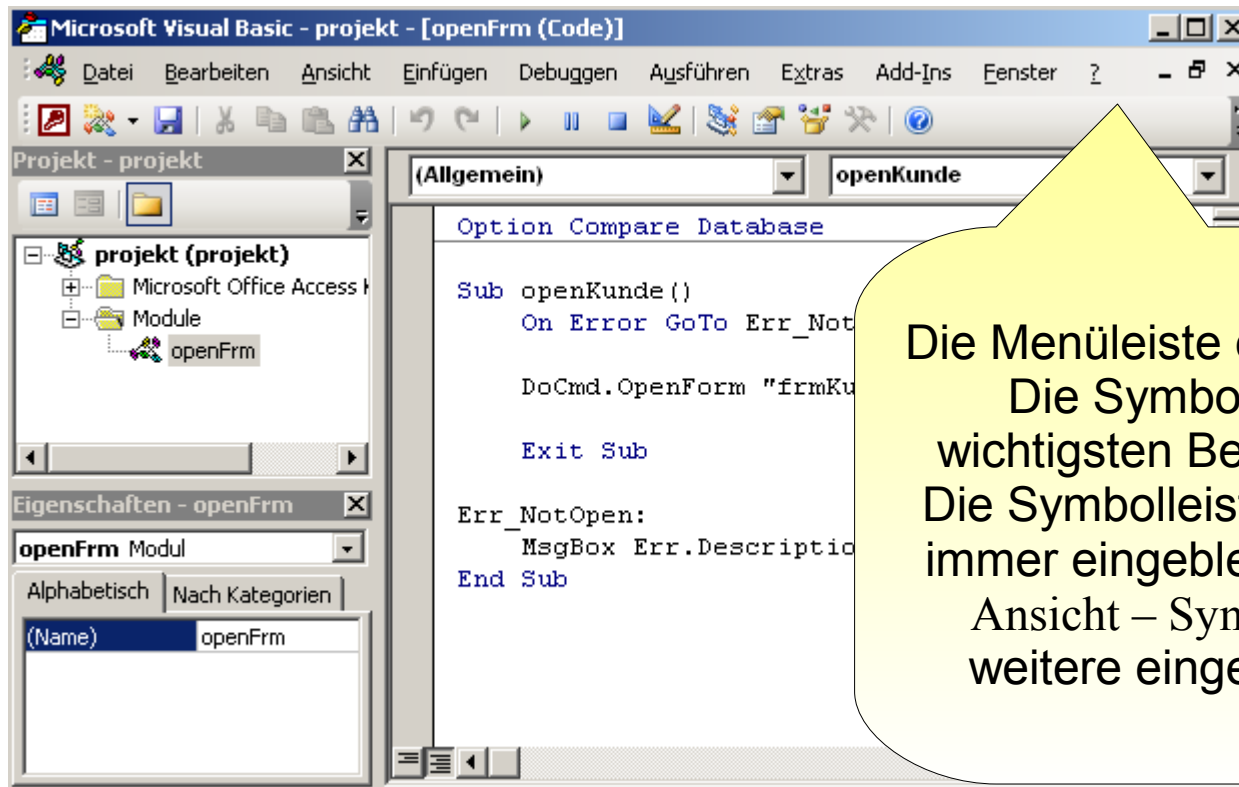
- ... ist ein eigenständiger Editor in der jeweiligen Office-Anwendung.
- ... wird in der Taskleiste angezeigt.
- ... setzt sich aus vielen verschiedenen Komponenten zusammen, die über das Menü *Ansicht* im VBA-Editor eingeblendet werden können.
- Klicken Sie auf das Objekt *Module* und anschließend auf die Schaltfläche *Entwurf* oder *Neu*, um den VBA-Editor zu öffnen.



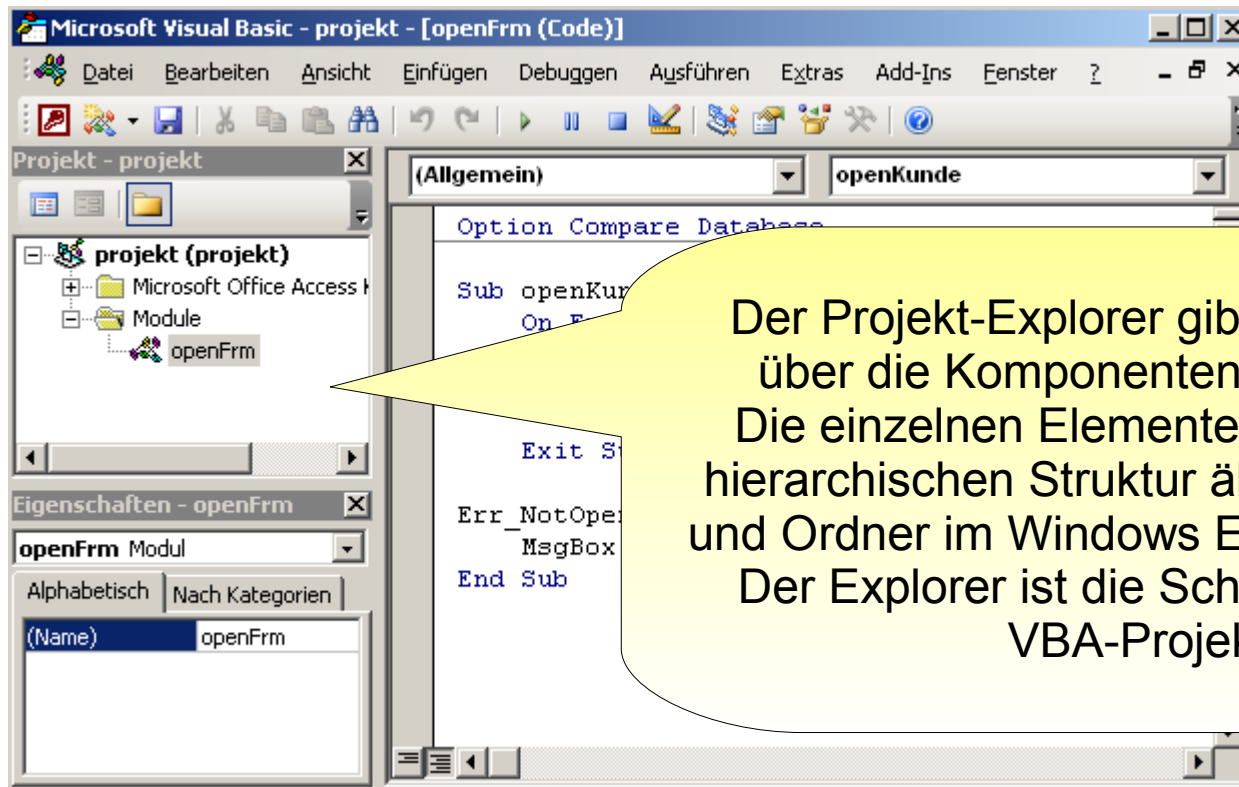
- Klicken Sie auf das Objekt **Module** im Datenbankfenster. Wählen Sie ein vorhandenes Modul aus und klicken auf die Schaltfläche **Entwurf** am oberen Rand des Datenbankfensters. Klicken Sie auf die Schaltfläche **Neu**, um den VBA-Editor zu öffnen und gleichzeitig ein neues Modul anzulegen.
- Öffnen Sie ein Formular oder Bericht in der Entwurfsansicht. Markieren Sie ein Objekt aus und öffnen das zugehörige **Eigenschaftenfenster**. Wechseln Sie auf die Registerkarte **Ereignis** und wählen das gewünschte Ereignis aus. Im dazugehörigen **Kombinationsfeld** wird der Eintrag **Ereignisprozedur** ausgewählt. Es wird automatisch der VBA-Editor geöffnet und der Rahmen der Prozedur erstellt. Wenn Sie eine Schaltfläche oder Liste mit Hilfe des Assistenten erstellen, wird VBA-Code automatisch generiert.
- Wählen Sie ein vorhandenes Makro aus. Klicken Sie auf **Speichern** unter im dazugehörigen **Kontextmenü**. Geben Sie eine Bezeichnung ein und speichern es als Modul. Sobald die Konvertierung des Makros abgeschlossen ist, wird automatisch der VBA-Editor geöffnet.



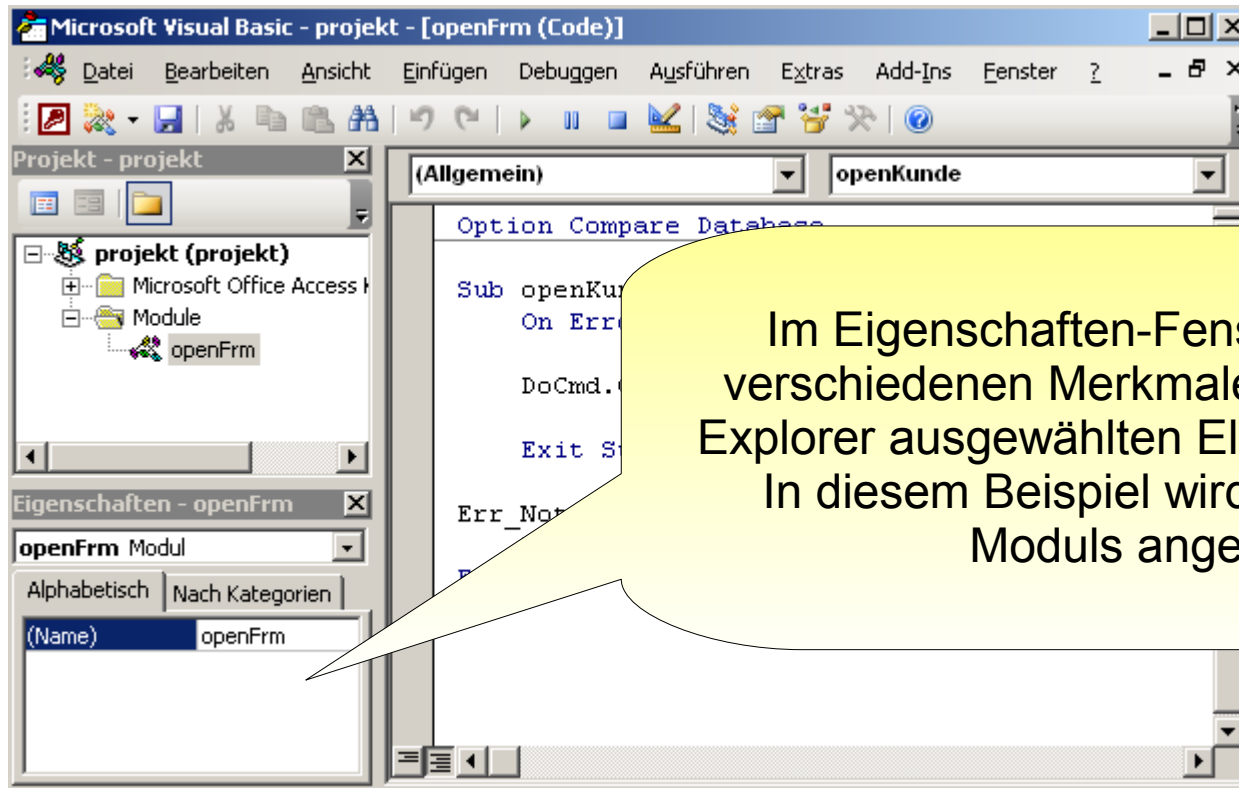
Die Titelleiste besteht aus dem Systemmenü, den Schaltflächen zum Minimieren, Maximieren und Schließen des Editors. In der Titelleiste wird der Name des aktiven Moduls angezeigt.



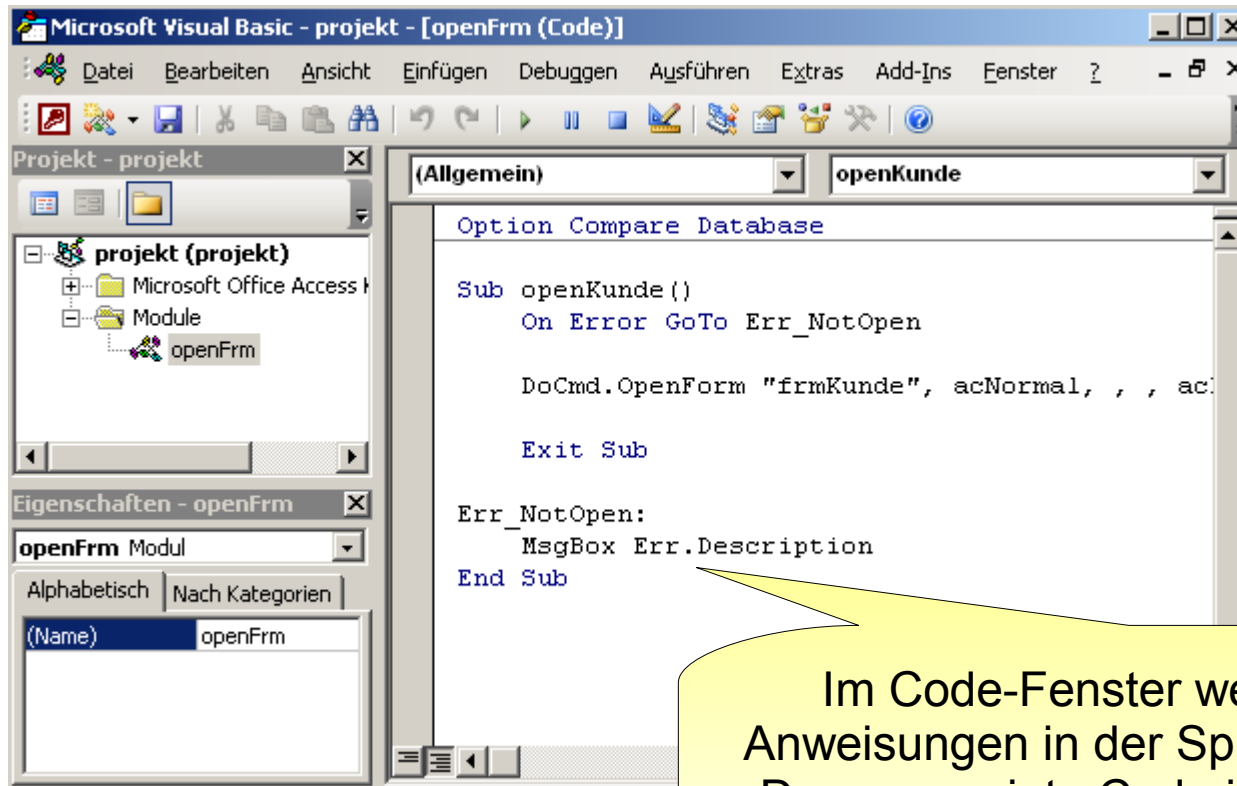
Die Menüleiste enthält alle Befehle.
Die Symbolleiste stellt die wichtigsten Befehle als Icon dar.
Die Symbolleiste Voreinstellung ist immer eingeblendet. Mit Hilfe von Ansicht – Symbolleiste können weitere eingeblendet werden.



Der Projekt-Explorer gibt einen Überblick über die Komponenten eines Projekts. Die einzelnen Elemente werden in einer hierarchischen Struktur ähnlich wie Dateien und Ordner im Windows Explorer dargestellt. Der Explorer ist die Schaltzentrale eines VBA-Projekts.



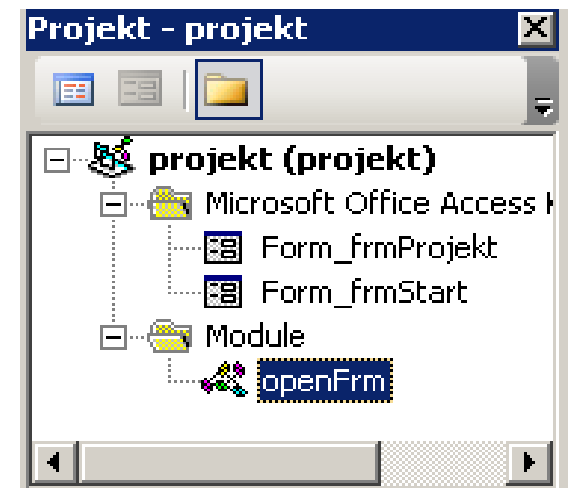
Im Eigenschaften-Fenster werden die verschiedenen Merkmale des, im Projekt-Explorer ausgewählten Elements angezeigt. In diesem Beispiel wird der Name des Moduls angezeigt.



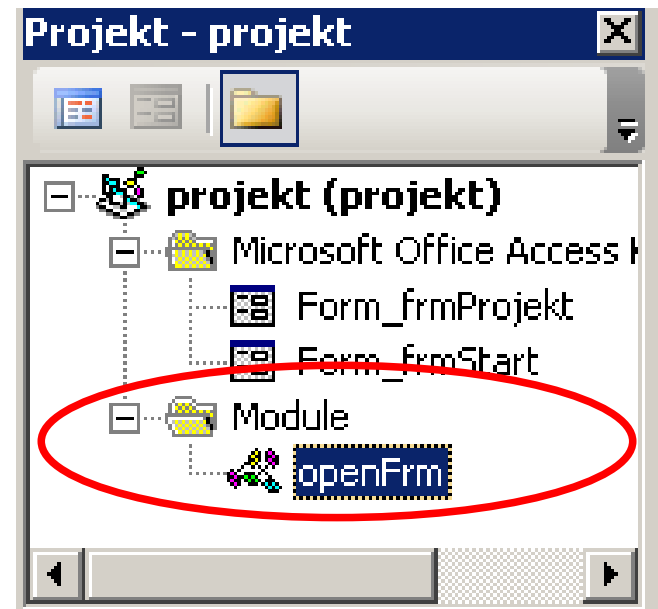
Im Code-Fenster werden Befehle und Anweisungen in der Sprache VBA angezeigt. Der angezeigte Code ist immer mit, dem im Projekt-Explorer ausgewählten Element verbunden.

- ... sind der äußere Rahmen für ein VBA-Programm.
- ... werden automatisch von einer Office-Anwendung angelegt.
- ... können nicht vom Entwickler angelegt werden.
- ... werden gemeinsam mit der Datenbank in einer Datei gespeichert.

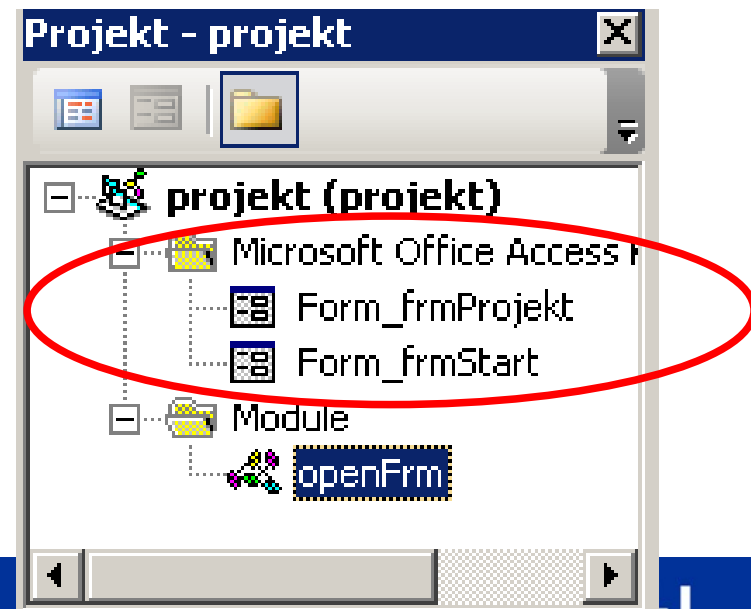
- ... ist frei platzierbar.
- ... enthält alle zu dem Projekt gehörenden Elemente.
- In der Titelleiste
 - ... wird der Name des Projekts angezeigt.
 - ... befindet sich am rechten Rand eine Schaltfläche zum Schließen des Explorers.
- In der Symbolleiste werden folgende Funktionalitäten von links nach rechts in Form von Icons angeboten:
 - Es wird der Code des ausgewählten Elements angezeigt.
 - Das dazugehörige Objekt (Formular oder Bericht) wird angezeigt.
 - Die Elemente werden hierarchisch mit Hilfe von Ordnern gegliedert oder in einer Ebene dargestellt..



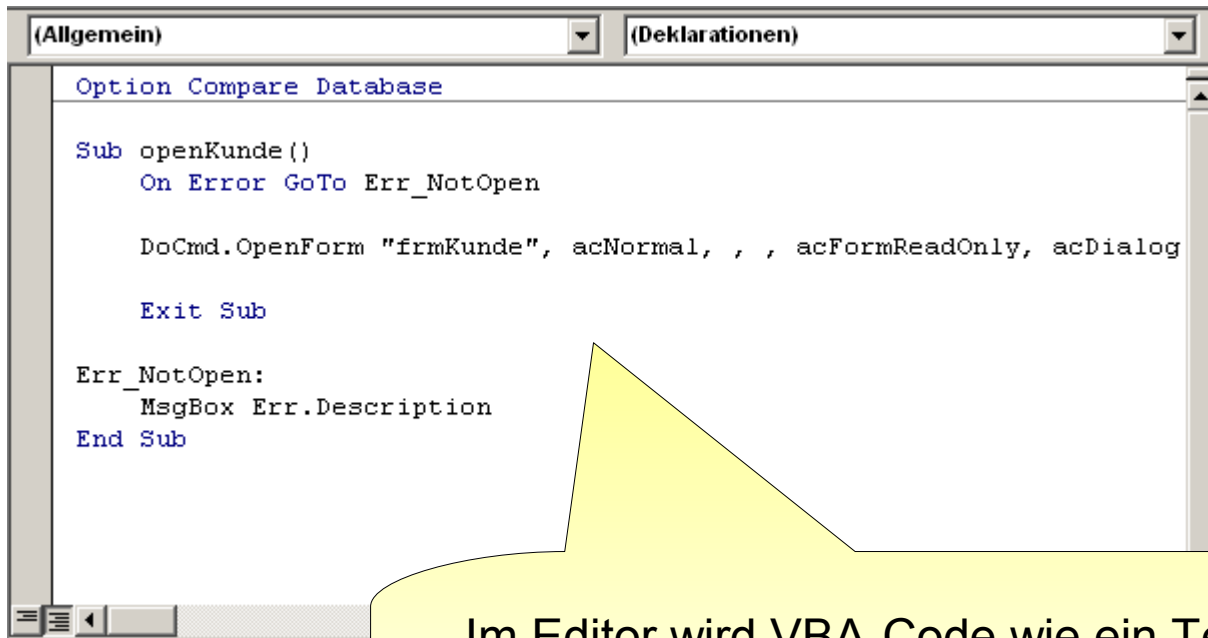
- ... enthalten Code, der nicht an ein Formular, Bericht oder Steuerelement gebunden ist.
- ... sind nicht an ein bestimmtes Objekt gebunden.
- ... enthalten häufig Code, der innerhalb des gesamten Projekts von vielen Elementen genutzt wird.
- ... werden über das Objekt Module und dem Symbol Neu im Datenbankfenster erstellt.



- ... sind mit einem bestimmten Formular oder Bericht verbunden.
 - Sobald eine Ereignisprozedur erstellt wird, wird automatisch, falls nicht vorhanden, ein Klassenmodule für das entsprechende Formulare oder Bericht angelegt.
 - In diesen Klassenmodulen sammeln sich Code, der in irgendeiner Form mit dem Formular / Bericht oder den darin enthaltenen Steuerelementen verbunden ist.
- ... können für die Definition von eigenen Objekten genutzt werden.
 - Diese Module werden mit Hilfe des Menüs *Einfügen - Klassenmodule* im VBA-Editor erstellt.



- ... wird immer in einer bestimmten Sprache geschrieben. In Office-Programmen wird die Sprache VBA genutzt.
- ... wird in einer bestimmten Syntax geschrieben. Die Sprache "Deutsch" besitzt genauso wie VBA eine eigene Grammatik.
- ... besteht aus einer bestimmten Anzahl von Befehlen.
- ... ist wie eine Fremdsprache für Menschen. Sobald Sie man die Sprache erlernt hat, kann man sie verstehen und lesen.



```
(Allgemein) (Deklarationen)
Option Compare Database

Sub openKunde ()
    On Error GoTo Err_NotOpen

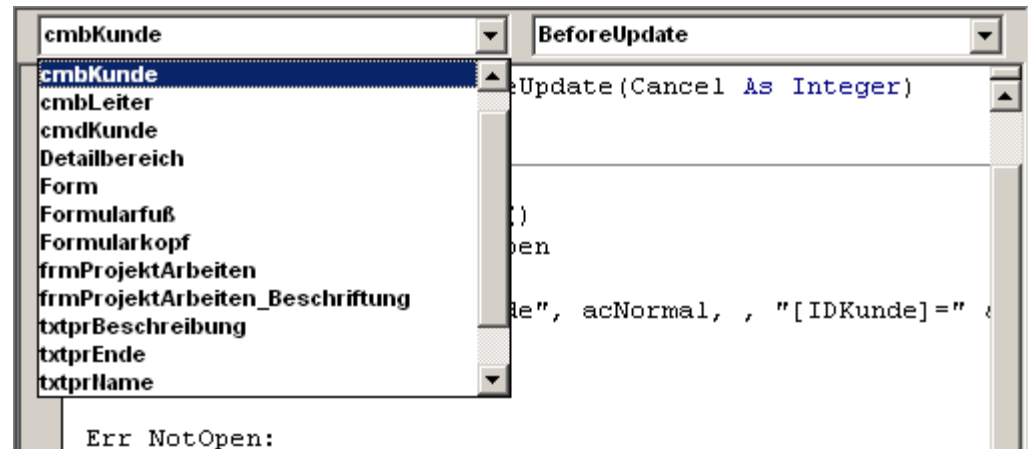
    DoCmd.OpenForm "frmKunde", acNormal, , , acFormReadOnly, acDialog

    Exit Sub

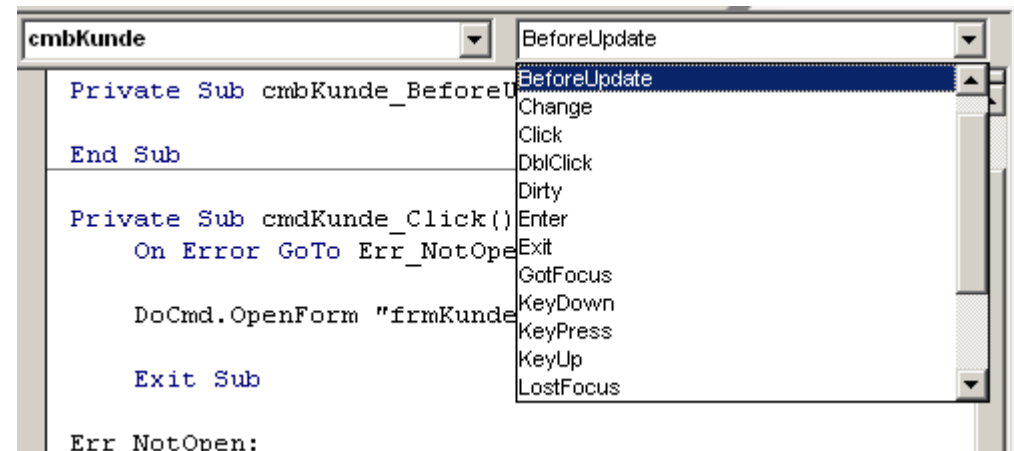
Err_NotOpen:
    MsgBox Err.Description
End Sub
```

Im Editor wird VBA-Code wie ein Text in Word geschrieben.
Die Einrückungen dienen der besseren Lesbarkeit.

- Am linken, oberen Rand des Code-Fenster befindet sich ein Kombinationsfeld, welches alle Objekte auflistet, die mit dem Modul verbunden sind.
- Mögliche Elemente in diesem Kombinationsfeld:
 - Der Eintrag Allgemein ist immer vorhanden. Hier werden alle Prozeduren etc. angezeigt, die nicht einem bestimmten Objekt zugeordnet werden können.
 - Falls das Modul an ein Formular oder Bericht gebunden ist, wird das Formular / Bericht sowie deren Steuerelemente aufgelistet.



- In Abhängigkeit des ausgewählten Objekts (linkes Kombinationsfeld) werden im rechten, oberen Kombinationsfeld alle dazugehörigen Prozeduren angezeigt.
- Folgende Elemente können vorhanden sein:
 - Der Eintrag *Deklaration* ist immer vorhanden. Hier werden alle Definitionen angezeigt, die nicht einem bestimmten Objekt zugeordnet werden. Hier können auch Definitionen abgelegt werden, die alle zu einem Modul gehörenden Objekte betreffen.
 - Falls Sie ein Formular, Bericht oder Steuerelement als Objekt ausgewählt haben, werden hier alle möglichen Ereignisse angezeigt.



- Wähle das Objekt Modul im Datebankfenster.
 - Klicke auf die Schaltfläche Neu.
- oder
- Menü Extras – Makro – Visual Basic Editor (Alt+F11) in Access.
 - Menü Einfügen – Modul im VBA-Editor.
-
- Setzen Sie die Einfügemarke in die nächste freie Zeile.
 - Geben Sie zum Beispiel den nachfolgenden Code ein, um ein bestimmtes Formular zu öffnen.

```
Sub openFrmKunde()  
    DoCmd.OpenForm "frmKunde", acNormal,,acFormReadOnly, acDialog  
End Sub
```

```
Sub openFrmKunde()
```

```
DoCmd.OpenForm "frmKunde", acNormal,,acFormReadOnly, acDialog
```

```
End Sub
```

Jeder Code wird in eine Prozedur eingeschlossen. Eine Prozedur sammelt alle Zeilen Code, die zu einem bestimmten Thema gehören.

In VBA beginnt eine Prozedur mit dem Schlüsselwort *Sub*, gefolgt von einer frei wählbaren Bezeichnung und den runden Klammern. Innerhalb der runden Klammer können der Prozedur Werte übergeben werden.

Eine Prozedur endet mit dem Schlüsselwort *End Sub*.

```
Sub openFrmKunde()
```

```
DoCmd.OpenForm "frmKunde", acNormal,,acFormReadOnly, acDialog
```

```
End Sub
```

DoCmd ist ein Objekt in VBA, welches fast alle Makro-Befehle nachbildet. Hier wird der Makro-Befehl *OpenForm* (ÖffnenFormular) genutzt.

Die Aktion wird mit einem Punkt mit dem Objekt verbunden. Alle in VBA definierten Funktionen / Aktionen besitzen einen englischsprachigen Namen.

```
Sub openFrmKunde()
```

```
DoCmd.OpenForm "frmKunde", acNormal,,acFormReadOnly, acDialog
```

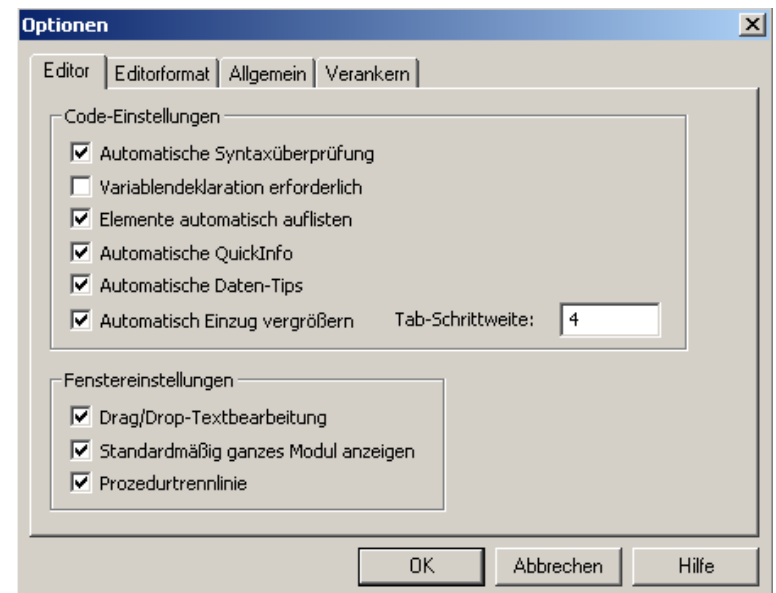
```
End Sub
```

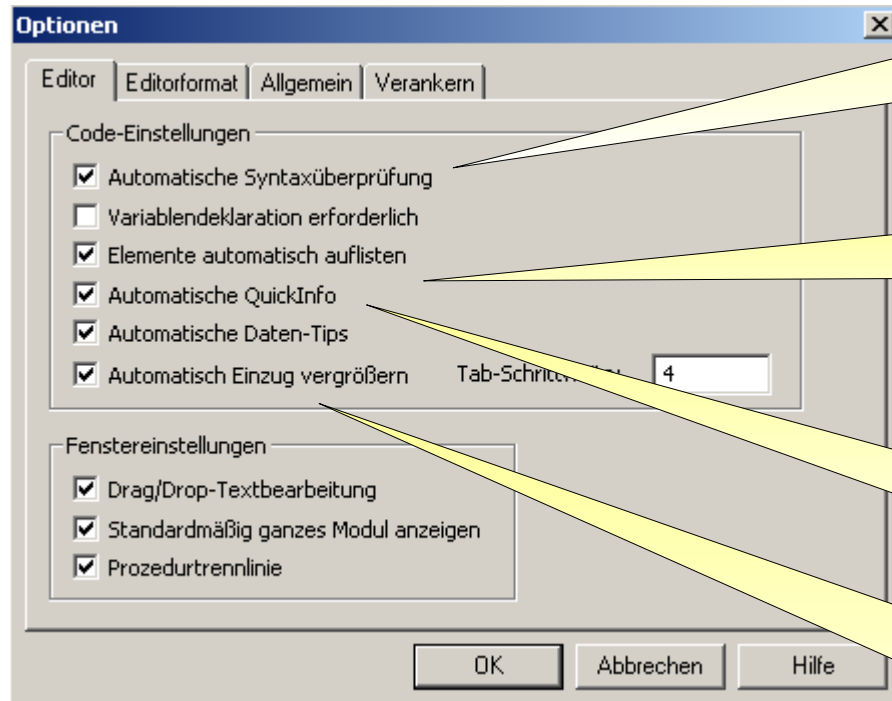
Für die Aktion können wie bei einem Makro bestimmte Werte festgelegt werden. In diesem Beispiel werden für die Aktion *OpenForm* der Name des Formulars *frmKunde*, die Öffnungsart (*acDialog*, Dialogfenster), die Ansicht (*acNormal*, Formularansicht) übergeben. Die Daten in diesem Formular dürfen nur gelesen (*acFormReadOnly*) werden.

- Klicken Sie auf das Menü Datei - ... speichern.
- Setzen Sie den Mauszeiger in die Prozedur, die Sie laufen lassen möchten.
- Drücken Sie <F5>.
- Das angegebene Formular wird geöffnet.

- Schließen-Schaltfläche in der Titelleiste des Editors.
- ... oder Datei – Schließen und zurück zu Microsoft Office Access.
- Falls Code nicht gespeichert ist, wird eine Warnmeldung ausgegeben.

- Klicken Sie auf Extras – Optionen, um
 - ... mit Hilfe der Registerkarte Editor die Darstellung und Eingabe von Code zu beeinflussen.
 - ... mit Hilfe der Registerkarte Editorformat die Schriftart und Schriftfarbe für die verschiedenen Elemente einzustellen.
 - ... mit Hilfe der Registerkarte Allgemein die Art der Fehlerüberprüfung festzulegen.
 - ... mit Hilfe der Registerkarte Verankern die Ausrichtung der einzelnen Elemente des Editors untereinander festzulegen.





Fehler in der Syntax eines VBA Codes werden farblich hervorgehoben.

Sinnvolle Ergänzungen zu Objekten etc. werden angeboten.

Gelbe Erklär Fenster werden bei vordefinierten Funktionen etc. angeboten.

Code-Zeilen werden eingerückt. Einrückungen erhöhen die Lesbarkeit von Code.

- Extras – Verweise listet eine Vielzahl von Objektbibliotheken auf.
- Objektbibliotheken enthalten Informationen zu den verschiedensten Modellen. Zum Beispiel die Objektbibliothek "Word" enthält Informationen zu dem Aufbau und Arbeitsweise von Elementen in Word.
- Mit Hilfe der Microsoft Access 11.0 Object Library werden alle Elemente von Access VBA bekannt gemacht.

