

Ulf Neubert

dBASE lebt!

Band 1

Einführung

dBASE Programmierung unter Windows

**Für dBASE Plus und seine Vorgänger
dBASE SE, dBASE 2000, Visual dBASE**

Copyright © 2005 Ulf Neubert

Inhaltsverzeichnis

1. Einführung.....	9
1.1 Allgemeines zu dieser Buchreihe	10
1.2 Infos zum Autor	11
1.3 Syntax und Layout	12
1.4 Ihre Voraussetzungen.....	13
1.5 Etwas Grundsätzliches vorab	14
1.6 Programmlistings im Buch.....	16
1.7 Schwerpunkt dieser Ausgabe	17
2. Aller Anfang ist leicht.....	18
2.1 Installation.....	18
2.2 dBWin starten.....	20
2.3 Fenster und Hauptmenü im Überblick	21
2.4 Das dQuery-Fenster.....	22
2.5 Das Befehlsfenster	23
2.5.1 dBASE-Befehle direkt ausführen.....	24
2.5.2 Tips zum Befehlsfenster.....	29
2.5.3 Eigenschaften des Befehlsfensters	29
2.5.4 Hilfe im Befehlsfenster	30
2.6 Das Regiezentrum	31
2.6.1 Eigenschaften des Regiezentrams	32
3. Einrichten von dBWin.....	34
3.1 Allgemeine Informationen	34
3.2 Einstellungen für Länder.....	37
3.3 Einstellungen für Tabellen	42
3.4 Einstellungen zur Dateneingabe.....	49
3.5 Einstellungen für Dateien und Pfade.....	52
3.6 Einstellungen für Anwendungen.....	55
3.7 Einstellungen für Programmierung	57
3.8 Einstellungen für Quell-Aliase.....	65
3.9 Einstellungen speichern.....	67
3.10 Einstellungen der BDE.....	68

4. Eine Tabelle für Termine71

4.1 Tabellen mit dem Experten anlegen	72
4.2 Daten anzeigen lassen	75
4.3 Daten bearbeiten per Menü	78
4.4 Navigation und Anzeige per Menü	82
4.5 Suchen und ersetzen per Menü.....	84
4.6 Sortierung per Menü.....	87
4.7 Filter per Menü.....	89
4.8 Auswertungen per Menü	90

5. Eine Tabelle für Adressen92

5.1 Tabellen mit dem Designer anlegen	93
5.1.1 Datentypen und Spalten im Designer.....	94
5.1.2 Felder und einfache Indexe anlegen.....	97
5.1.3 Erweiterte Felddaten im Objekt-Inspektor ...	101
5.2 Tabellen mit Befehlen anlegen.....	103
5.3 Daten erfassen und ändern	105
5.3.1 Die Tücken leerer Tabellen	106
5.3.2 Datensätze anlegen, Felder füllen	108
5.3.3 Navigation mit Befehlen	110
5.3.4 Daten löschen und wiederbeleben.....	111
5.3.5 Daten endgültig löschen.....	115
5.4 Tabellen und Dateien löschen	116
5.5 Indexe verwalten, ändern, löschen	117
5.5.1 Indexe mit Befehlen verwalten	121
5.5.2 Kombinierte Indexe.....	125
5.5.3 Indexe per Befehl aktivieren	126
5.5.4 Nützliche Routinen für Indexe	127
5.6 Suchen und ersetzen mit Befehlen	129
5.6.1 Sequentielle Suche mit locate	129
5.6.2 Indexsuche mit seek	133
5.6.3 Suchschleifen mit scan.....	136
5.6.4 Ersetzen mit replace	137
5.7 Filter mit Befehlen	138
5.8 Auswertungen mit Befehlen.....	140
5.9 Benutzte Felder einschränken	141
5.10 Tabellen duplizieren	142
5.11 Tabellen zusammenführen	145

6. Ein Formular für Termine149

6.1	Einführung.....	149
6.2	Unterfenster im Formular-Designer	151
6.3	Tabelle per View und Browse einbinden	154
6.3.1	Feinarbeiten am Browse-Objekt.....	160
6.4	Steuerung mit Buttons	164
6.5	Datensätze anlegen und löschen.....	168
6.6	Datensätze sortieren, Indexe benutzen	172
6.7	Datensätze suchen	174
6.8	Einzelfelder als Alternative zum Browse	176
6.8.1	Tab-Reihenfolge der Objekte	182

7. Ein Formular für Adressen184

7.1	Einführung.....	184
7.2	Der Formular-Experte	185
7.3	Formular ändern, Felder platzieren	190
7.4	Eigenschaften der Felder ändern	192
7.5	Linien ins Formular einfügen.....	194
7.6	Datenbankfelder verknüpfen	196
7.7	Auf Ereignisse reagieren	201
7.8	Fortgeschrittene Ereignis-Behandlung.....	210
7.9	Steuerung mit Buttons	220
7.10	Fortgeschrittene Button-Steuerung.....	229
7.11	Datensätze im Formular anlegen	234
7.12	Datensätze im Formular löschen	237
7.12.1	Rückfragen einbauen.....	238
7.13	Datensätze im Formular suchen	241
7.13.1	Suchbegriff per Dialog eingeben	242
7.14	Sortierung ändern	252
7.14.1	Die Tücken von Radiobuttons.....	256
7.15	Ein paar letzte Feinarbeiten	262
7.15.1	Hotkeys für Eingabefelder	262
7.15.2	Suche im aktiven Index	262
7.15.3	Feintuning beim Löschen	263
7.15.4	Weitere Details für Eingabefelder.....	264
7.15.5	Tab-Reihenfolge der Objekte	264
7.15.6	Wir haben fertig	266

8. Was sonst noch wichtig ist.....267

8.1	Arbeitsbereiche.....	267
8.2	Aliasnamen.....	272
8.3	dBASE im Netzwerk.....	276
8.3.1	Installation im Netzwerk.....	276
8.3.2	Für das Netzwerk programmieren.....	278
8.4	Zeichensätze und Sprachtreiber.....	280
8.5	DOS-Befehle ausführen.....	281
8.6	Nützliche Windows-Funktionen.....	282
8.7	Drucken mit dem ?-Befehl.....	285
8.8	Verzeichnisse von dBASE Plus.....	288
8.9	Von DOS nach Windows.....	291
8.10	Grenzwerte.....	292

9. Anhang.....293

9.1	Ein Wort zum Abschied.....	293
9.2	Weitere Bücher dieser Reihe.....	294
9.3	Stichwortverzeichnis.....	295

1. Einführung

Willkommen zum ersten Band einer neuen Buchreihe. Wenn Sie diese Zeilen lesen ist es geschafft und eine lang geplante Idee wurde erfolgreich umgesetzt.

Aus der DOS-Ära stammend und danach lange für tot gehalten, hat **dBASE** in den 90er Jahren als *Visual dBASE* und *dBASE for Windows* sein verdientes Comeback erlebt. Anfangs noch als 16bit Version, später im 32bit-Gewand. Um den Jahrtausendwechsel wurde es als *dBASE 2000* angeboten und gerade als dieses Buch entsteht, heisst es zur Abwechslung *dBASE Plus*.

Fast ebenso oft wie der Name hat in den letzten Jahren auch der Eigentümer gewechselt. Diese vielen Wechsel von Name und Inhaber, die oft nicht klar erkennbare Strategie und ein zuweilen eher bescheidenes Marketing haben vermutlich ihren Teil dazu beigetragen, dass leider viele Anwender mangels Vertrauen in das Produkt und die Zukunft dBASE den Rücken gekehrt haben.

Schlimmer noch, oft habe ich es erlebt, dass auch langjährige Programmierer von dBASE unter DOS die neuen Windows-Versionen nicht kannten, ja nicht mal wussten dass es sie gibt. Ich musste bei vielen Überzeugungsarbeit leisten und mehrfach betonen „ja, dBASE gibt es auch unter Windows! dBASE lebt!“, und so entstand dann auch der Titel für diese neue Buchreihe. Vielleicht trägt er ja dazu bei, etwas vom Glanz alter Zeiten für dBASE zurück zu bekommen.

Nach so häufig geänderten Namen in so kurzer Zeit möchte ich die ständige Änderung des Textes bei jeder Neuauflage des Buchs vermeiden und bin daher so frei, im folgenden Text einfach den etwas allgemeineren Begriff

dBWin

zu verwenden, als Synonym für die gerade aktuelle Version von dBASE unter Windows. Bei Erstellung dieses Textes ist gerade **dBASE Plus V 2.5** aktuell. Damit wird auch für weitere Ausgaben dieser Buchreihe ein fester Begriff beibehalten, egal wie das Produkt in ein paar Jahren heissen wird.

Wenn im Text bezug auf ältere Versionen genommen wird, gleich ob für Windows oder DOS, so wird dies immer ausdrücklich erwähnt. Ansonsten können Sie davon ausgehen, dass die aktuelle Version gemeint ist, bzw. die Version, die bei der Texterstellung gerade aktuell war, denn Ihre Version könnte schon wieder etwas neuer sein, wenn Sie dieses Buch lesen.

1.1 Allgemeines zu dieser Buchreihe

Ich werde mich mit dieser Reihe gezielt und in jeweils kleinen und leicht verdaulichen Häppchen pro Band einzelnen Schwerpunkten widmen.

Sie bekommen kein kiloschweres, waffenscheinpflichtiges Buch mit über 1.000 Seiten vorgesetzt, sondern mehrere kleinere überschaubare Ausgaben, die sich intensiv einem ganz bestimmten Thema oder einem Themenbereich widmen. Jedes der Bücher wird in sich abgeschlossen sein. Die ersten beiden Bände, *EINFÜHRUNG* und *GRUNDLAGEN*, werden aber eine gemeinsame Basis bilden, die in den folgenden Ausgaben stets als bekannt vorausgesetzt wird.

Geplante Themen sind z. B. Klassen und Objekte, Zugriff auf fremde Dateien und Datenbanken, SQL, die Feinheiten und Fallen von Windows, DDE, OLE, Web-Datenbanken, Programmierkniffe für Profis und vieles mehr. Einer der geplanten Bände wird sich intensiv mit Windows-Programmierung befassen, also dort beginnen wo andere Bücher zu dBASE bislang immer aufhörten.

Jeder Band hat seine Zielgruppe: Einsteiger, Fortgeschrittene oder Profis. Die Reihe startet mit zwei Bänden für Einsteiger und DOS-Umsteiger. Es soll aber nicht die ultimative Lösung für ein Problem angeboten werden, sondern die Bücher sind als Hilfe zur Selbsthilfe gedacht, sollen mögliche Wege aufzeigen und eine wertvolle Unterstützung für Ihre eigene Arbeit mit dBWin sein.

Diese Bücher werden auch nicht alles rund um dBASE abdecken können, sonst wären wir wieder beim 1.000-Seiten Wälzer. Sie werden also weder ein „Kompendium von A-Z“ darstellen, noch werden sie eine vollständige „Befehls-Referenz“ enthalten (die es aber bei Erfolg der Reihe evtl. später einmal separat geben könnte). Es werden Themen beleuchtet, von denen ich glaube, dass sie wichtig sind. Und das dann jeweils sehr intensiv.

Hier, liebe Leserin und lieber Leser, kommen Sie ins Spiel! Welche Themen und Schwerpunkte es künftig noch geben wird hängt nicht zuletzt auch von Ihnen, Ihren Anforderungen und Wünschen ab. Wenden Sie sich bitte direkt an mich und teilen Sie mir Ihre Wünsche mit. Je mehr Anfragen zu einem Thema eintreffen, desto grösser ist die Chance dass es später umgesetzt wird. Wünsche, Anregungen und Kritik richten Sie bitte per eMail an

dbase@ulfneubert.de

Bitte haben Sie Verständnis, dass Ihre eMails zu künftigen Themen nicht in jedem Fall individuell beantwortet werden können. Das ist zeitlich einfach nicht möglich. Aufgenommen werden Ihre Vorschläge aber auf jeden Fall.

1.2 Infos zum Autor

Ich arbeite seit vielen Jahren selbständig als Programmierer und bin Inhaber einer Software-Firma. Insbesondere mit dBWin habe ich in den letzten Jahren einige Projekte realisiert. Neben der Umstellung vieler alter DOS-Programme entwickelte ich auch ein sehr umfangreiches Programm zur Depotverwaltung für Vermögensberater und Investmentfonds-Vermittler. Es wurde konsequent von Anfang an mit dBWin entwickelt und hat noch heute treue Anwender. Es heisst *Avalonia*, ich habe es längst verkauft, betreue es aber immer noch mit.

Mit der Zeit hat sich sehr viel Erfahrung mit dBWin angesammelt. Natürlich habe ich Fehler gemacht und so manches Problem bereitete Kopfschmerzen, bevor es dann meistens doch auf die eine oder andere Art gelöst wurde.

Der Gedanke, diese Erfahrungen in Seminaren und Büchern auch anderen dBASE-Nutzern zugänglich zu machen, bestand schon länger. Letztendlich haben dann immer häufigere Nachfragen einiger Kunden nach solcher Unterstützung dazu geführt, die Idee in Form dieser Buchreihe umzusetzen.

Sie können mich und meine Firma gern im Internet besuchen:

www.cantaria.de
www.ulfneubert.de

Dort können Sie die aktuell verfügbaren und noch geplanten Ausgaben dieser Buchreihe ansehen und natürlich auch bestellen. Es ist auch geplant, die in den Büchern verwendeten Quellcodes dort zum Download anzubieten, allerdings nur bei den Ausgaben, in denen längere Programmcodes verwendet werden. Desweiteren werden Sie dort Leseproben aller Bücher dieser Reihe finden.

Die **Cantaria GmbH** führt individuelle Aufträge für Programmierung aus, ganz besonders für dBWin, aber natürlich auch anderes. Wir übernehmen komplette Projekte jeden Umfangs und ausdrücklich auch kleine Projektteile und einzelne Module, damit Sie entlastet werden. Gern unterstützen wir Sie per eMail und stehen Ihnen bei Fragen rund um dBWin helfend zur Seite. Auch eine individuelle dBWin-Schulung, bei Ihnen oder uns, ist möglich.

Allgemeine Anfragen zu Programmierleistungen, Übernahme von Projekten (oder Teilen davon), Tests und Optimierung Ihrer bereits erstellten eigenen Programme, Umsetzung von alten DOS-Programmen auf Windows sowie Anfragen zu individuellen Schulungen etc. richten Sie bitte per eMail an

info@cantaria.de

1.3 Syntax und Layout

Damit Sie stets wissen von was die Rede ist werden in diesem Buch einige gleiche Formate und Textdarstellungen für bestimmte Dinge verwendet.

Tasten und Buttons

Zu drückende Tasten wie **[Return]** oder **[Esc]** werden in eckige Klammern gesetzt und fett geschrieben. Mehrere gleichzeitig zu drückende Tasten werden mit + kombiniert, z. B. **[Alt] + [F4]** zum Beenden des Programms. Ebenso verhält es sich mit Buttons, z. B. wenn Sie auf **[Ok]** klicken sollen.

Menüpunkte, Dialogelemente, Dateinamen, Datenbankfelder etc.

Werden *kursiv* geschrieben, z. B. das Eingabefeld *Nachname* dient dazu ..., laden Sie die Datei *anfang.prg* in den Editor ..., das Feld *nummer* enthält ...

dBWin-Befehle und Programmbeispiele

Quellcodes werden mit anderer Schrift und etwas eingerückt geschrieben:

```
use adressen.dbf
? reccount()
```

Manchmal werden Befehlszeilen nummeriert. Die Nummern dienen aber nur für nachfolgende Erklärungen im Text und sind nicht von Ihnen abzutippen.

Platzhalter für Parameter oder Variablen etc. stehen in spitzen Klammern:

```
seek <suchbegriff>
select <alias>
```

Optionale Befehls Elemente werden in eckigen Klammern gesetzt:

```
set order to [indexname]
set filter to [bedingung]
```

Wahlweise Optionen werden mit einem | (Pipe) getrennt:

```
set exact on | off
```



Wichtige Punkte, Tips und Tricks, Hinweise auf Besonderheiten.



Vorsicht Falle, hier wird vor Bugs und häufigen Fehlern gewarnt.

1.7 Schwerpunkt dieser Ausgabe

Diese Ausgabe unterstützt Ihren ersten Kontakt und den Einstieg in dBWin. Egal ob Sie nun von dBASE/DOS kommen und auf Windows umgestiegen sind, oder ob die Programmierung mit dBASE noch komplett neu für Sie ist.

Wir werden zu Beginn die aktuelle dBWin-Version gemeinsam installieren, die Oberfläche näher kennenlernen, dBWin konfigurieren und uns dann mit den wichtigsten Punkten der Entwicklungsumgebung beschäftigen.

Dabei wird im Verlauf des Buches ein kleines Programm erstellt, das Ihnen einen guten ersten Eindruck von den vielfältigen Möglichkeiten, aber auch den (besonders für DOS-Entwickler) doch teils gänzlich anderen Anforderungen und Abläufen unter Windows gibt. Das Programm wird einige Basis-Elemente wie z. B. Formulare und Eingabefelder enthalten, die Sie bei der dBWin-Programmierung als zentrale Bestandteile immer wieder benötigen werden.

i Der erste Band kann und soll nur einen kleinen Ausschnitt von dBWin beleuchten. Der zweite Band *GRUNDLAGEN* wird die perfekte Ergänzung dazu. Er vertieft das hier gezeigte und enthält wertvolle Informationen, damit Sie mit dBWin effizient und sicher programmieren lernen. Überschneidungen beider Ausgaben sind minimal und auf das zum Verständnis notwendige beschränkt, da berücksichtigt werden muss, dass ein Leser evtl. nur eines der Bücher hat.

Wir werden in dieser Ausgabe weder komplexe Datenbank-Operationen noch SQL-Befehle oder ähnliches ausführen. Auch Besonderheiten wie OLE oder Web-Anwendungen sind am Anfang noch nicht relevant. Erst einmal werden wir dBWin besser kennenlernen und danach Schritt für Schritt ein kleines Programm zum Leben erwecken. Der Rest folgt später, alles zu seiner Zeit.

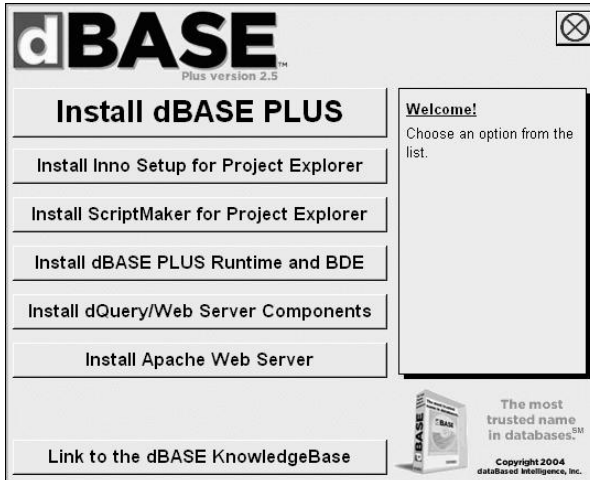
i Bei solch einem Buch ist es immer eine Gratwanderung, weder einen Teil der LeserInnen zu überfordern, noch den anderen Teil zu langweilen. Vermutlich wird mir dieser Spagat nicht immer gelingen, das ist auch schlicht unmöglich. Im Zweifelsfall werde ich aber einer genaueren Beschreibung den Vorzug geben, auch auf die Gefahr hin, einen Profi damit für einen oder zwei Absätze zu langweilen. Aber in diesem Fall haben Sie als Fortgeschrittener immerhin den grossen Vorteil, diesen Absatz einfach elegant überspringen zu können, während ein Einsteiger sich das Wissen über ein nur unzureichend beschriebenes Detail keineswegs so einfach aus der Luft saugen kann.

So, und nun sitzen Sie bitte bequem, wir fangen an ...

2. Aller Anfang ist leicht

2.1 Installation

Legen Sie die CD mit dBWin ein (in Ihre CD-ROM Laufwerk, nicht in Essig und Öl, das greift die Datenschicht an ...). Sie erhalten einen Startbildschirm, der je nach Version Ihres eigenen dBWin etwa so aussieht:



Die Installation beginnt

Da wir uns fürs erste nur mit der Installation des Hauptprogramms befassen klicken Sie bitte den entsprechenden Punkt [**Install dBASE Plus**] an.

Die anderen angebotenen Programme und Module werden Sie am Anfang noch nicht brauchen, daher können Sie alles andere erst einmal ignorieren. Belasten Sie sich bitte nicht unnötig mit Dingen, mit denen Sie zu Beginn noch nicht viel anfangen können, die Welt ist schon kompliziert genug.

Nun folgen Sie einfach den weiteren Anweisungen des Setup-Programms. Sie haben das alle schon so oft bei so vielen Programmen gesehen, dass es keinen Sinn macht, hier nochmal die Installations-Dialoge wiederzugeben. Die vom Setup-Programm vorgeschlagenen Pfade können Sie beibehalten oder ändern, wenn Sie z. B. eine Abneigung gegen zu lange Pfadnamen haben oder wenn Sie es gewohnt sind selbst zu bestimmen, wo ein Programm installiert wird. Ich setze soviel Windows-Kenntnisse voraus, dass Sie wissen was zu tun ist.

Wenn Sie nach einer Seriennummer gefragt werden, geben Sie diese bitte ein. Sie finden die Serien- oder Lizenznummer entweder auf der CD aufgeklebt, oder auf der Rechnung des Lieferanten bzw. auf einem separat beiliegenden Blatt. Achten Sie bei Eingabe der Zeichen auf Gross- und Kleinschreibung und heben Sie das Blatt mit der Nummer zusammen mit der Original-CD auf.

Das Verzeichnis für die Installation wird vorgegeben. Sie können es natürlich ändern, das ist Ihre Entscheidung. Ist schon eine ältere Version installiert wird evtl. deren Verzeichnis vorgeschlagen und so die alte Version überschrieben. Meist können Sie dies auch bedenkenlos ausführen, wobei es aber ratsam ist, das alte Verzeichnis erst komplett mit allen Unterverzeichnissen zu kopieren um im Problemfall einen einfachen Weg zurück zu haben. Sie wissen ja, wenn Sie ein Backup haben klappt im Regelfall alles, schief geht's immer nur dann, wenn Sie mal keine Vorsorge für den Fall der Fälle getroffen haben ...

Die BDE wird ebenfalls installiert, am besten in ein eigenes Verzeichnis. Auch das geht weitgehend automatisch und ist eine Windows-übliche Installation.

i **BDE** steht für **Borland Database Engine**. Die Firma *Borland*, bekannt durch die Programmierwerkzeuge *Turbo-Pascal* und *Delphi*, war einer der früheren dBASE-Eigentümer und maßgeblich für die Entwicklung der BDE verantwortlich. Diese Datenbank-Schnittstelle von Borland dient u. a. dazu, dass Sie mit dBASE auch auf andere Datenbanken zugreifen können. Fürs erste brauchen Sie sich darüber den Kopf nicht zerbrechen, es genügt wenn Sie die BDE installieren, wir werden sie später gemeinsam konfigurieren.

Wenn Sie nach Ressourcen für Fremdsprachen gefragt werden, so benötigen Sie hier im Regelfall nur Deutsch. Nur wenn Sie jetzt schon sicher wissen, dass Sie später einmal Ihre Programme auch in andere Sprachen übersetzen wollen können Sie diese Sprachen, sofern verfügbar, hier gleich mit angeben.

Wundern Sie sich bitte nicht, wenn Teile der Installation oder der im Verlauf angezeigten Bilder in Englisch sind obwohl Sie eine dt. Version erworben haben. Nicht immer wird alles vollständig übersetzt und gerade beim Setup wird leider immer mal wieder gern etwas geschlampt. Wohl in der falschen Annahme, dass das ja eh niemand beachtet. Anscheinend begreift niemand, dass gerade der erste Eindruck eines neuen Produkts so überaus wichtig ist.

Nachdem die Installation abgeschlossen ist kann es nötig sein, dass Sie Ihren PC neu starten. Wenn dem so ist werden Sie vom Setup-Programm darauf hingewiesen und Sie sollten das auch tun. Erfolgt kein weiterer Hinweis können Sie davon ausgehen, dass ein Neustart von Windows nicht nötig ist.

2.2 dBWin starten

Bei der Installation wurde eine Programmgruppe im Windows-Startmenü eingerichtet, sowie evtl. auch ein Programm-Symbol (auch engl. *Icon* oder eingedeutscht *Ikone* genannt) auf Ihrem Desktop angelegt. Ob das gemacht wurde hängt von der Version bzw. den Details des Setup-Programms, sowie ggf. auch von Ihren Angaben während der Installation ab.

Sie werden also ein Symbol wie z. B. eines dieser hier vorfinden



*Eine Auswahl möglicher Symbole zum Starten von dBWin,
links das alte Visual dBASE, mittig dBASE 2000, rechts dBASE Plus*

Das Symbol ist entweder direkt auf Ihrem Desktop oder zumindest innerhalb der angelegten Programmgruppe bei *Start - Programme - dBASE ...* zu finden.

Je nach Version Ihres dBWin können das Symbol und die Textbezeichnung anders ausfallen. Oben sehen Sie daher die bei Erstellung dieses Buches aktuellen und am weitesten verbreiteten Symbole beispielhaft dargestellt. Wenn Sie allerdings noch mit *Visual dBASE* arbeiten sollten Sie sich ein Update überlegen, diese Version ist inzwischen nicht mehr zeitgemäss.

Durch einen Klick (oder Doppelklick, je nach dem wie Sie Ihr Windows konfiguriert haben) auf das Symbol starten Sie Ihr dBWin zum ersten mal.

Es öffnet sich das Hauptfenster mit einer Reihe von Unterfenstern (engl. Client-Windows oder auch MDI-Windows genannt), wobei sich einer ungeschriebenen Regel folgend meist das Unterfenster in den Vordergrund drängt, das am seltensten benötigt wird. Aber keine Sorge, das ist erstens völlig „normal“ und zweitens werden wir dieses Verhalten bald abstellen.

i MDI steht für **M**ultiple **D**ocument **I**nterface, was vereinfacht gesagt soviel bedeutet wie dass Sie innerhalb eines Programmfensters mehrere Dokumente in jeweils eigenen Unterfenstern öffnen und bearbeiten können. Sie kennen das aus Ihrer Textverarbeitung, auch dort können Sie ja mehrere Briefe und andere Texte gleichzeitig innerhalb des Programms bearbeiten. Genauso verhält es sich mit dBWin und seinen zahlreichen Unterfenstern.

3. Einrichten von dBWin

3.1 Allgemeine Informationen

Wenn Sie in eine neue Wohnung einziehen werden Sie zu Beginn erst einmal alles so einrichten, wie es sinnvoll und praktisch ist, bzw. wie es Ihnen gefällt.

Ebenso verhält es sich mit einem neuen Programmierwerkzeug, nur dass Sie hier nicht tapezieren, keine Lampen anschliessen und keine schweren Möbel durch die Gegend schieben müssen. Es genügt, den Dialog zur Konfiguration aufzurufen und ein paar Einstellungen zu tätigen. Das geht meist schnell und mit einer neuen Hausordnung brauchen Sie sich auch nicht befassen. Dennoch gibt es auch hier ein paar wichtige Dinge am Anfang zu beachten.

dBWin enthält zwei wesentliche Dialoge zur Konfiguration. Den ersten haben Sie ein paar Seiten zurück bereits kennengelernt und gesehen, wie Sie das Befehlsfenster und später den Editor nach Ihren Vorlieben einstellen können. Daher kommen wir nun gleich zum zweiten Dialog, der viel umfangreicher ist.

Auch diesen erreichen Sie über das Menü *Eigenschaften*, der Unterpunkt heisst *Eigenschaften des Desktop*. Als Alternative können Sie aber auch

```
set
```

direkt in das Befehlsfenster eingeben, beide Wege führen zum Ziel.

Dabei ist der *set*-Befehl ohne jeden Zusatz nur die einfachste Variante einer längeren Liste von *set*-Befehlen mit weiteren Zusätzen. Mit dem *set*-Befehl können Sie fast alle Einstellungen des Konfigurations-Dialogs von dBWin auch programmgesteuert ändern. Diese vielfältigen Möglichkeiten mögen Sie für den Anfang vielleicht abschrecken. Aber keine Angst, erstens ist nahezu alles sehr einfach zu verstehen und zweitens werden Sie die meisten dieser Einstellungen nur einmal vornehmen und dann kaum jemals wieder ändern.

Aber welchen Sinn macht es, viele wichtige Grundeinstellungen sowohl per Dialog als auch direkt im Programm ändern zu können, und welche der beiden Möglichkeiten ist vorzuziehen? Sie sollten beides anwenden, und ich werde in der folgenden Erklärung des Dialogs auch die dazugehörigen Varianten des *set*-Befehls erläutern. Dabei werden Sie in lockerer Folge neue dBASE-Befehle kennenlernen und sich Schritt für Schritt mit dBWin vertraut machen.

3.2 Einstellungen für Länder

Beginnen wir nun also mit unserem Bummel durch den Konfigurations-Dialog und der Erläuterung der jeweils dazu gehörenden *set*-Befehle.



Länderspezifische Einstellungen

Für *Zahlenwerte* können Sie sowohl das (Tausender-)Trennzeichen, meist wird das ein Punkt sein, als auch das *Dezimalzeichen*, meist ein Komma, einstellen. Die beiden entsprechenden *set*-Befehle dazu lauten

```
set separator to "."          && Tausenderpunkt
set point to ","             && Dezimalzeichen
```

während (wie bei vielen der *set*-Befehle) der Befehl ohne Zusatz, also

```
set separator to
set point to
```

die in der *Windows-Systemsteuerung* eingestellten Vorgaben aktiviert.

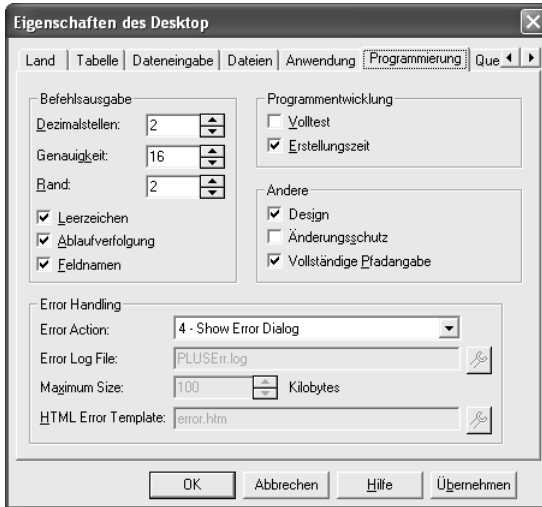
Die dazu gehörenden Abfragen der gerade aktiven Einstellung sind

```
? set ( "separator" )
? set ( "point" )
```

Die Abfrage via *set (<befehlswort>)* werde ich künftig nicht mehr bei jedem Befehl mit angeben, denn das Prinzip ist ja in allen Fällen dasselbe.

3.7 Einstellungen für Programmierung

Na endlich, jetzt wird es wieder interessant, es geht um Programmierung.



Einstellungen zur Programmierung

Der Bereich *Befehlsausgabe* enthält einige sehr wichtige Einstellungen. Diese beziehen sich, wie die Bezeichnung bereits vermuten lässt, auf die Ausgabe von Texten und Zahlen, beispielsweise im Ausgabebereich des Befehlsfensters oder (in eingeschränktem Umfang) auch auf dem Drucker.

i Diese Einstellungen haben im Regelfall keinen Einfluss darauf, wie Sie Ihre Daten in eigenen Formularen und Berichten anzeigen und drucken. Denn Sie werden später alle Feinheiten für Ein- und Ausgaben, wie z. B. die Anzahl der angezeigten Vor- und Nachkommastellen und noch vieles mehr, direkt im Formular bzw. direkt in den Objekten-Eigenschaften einstellen.

Beginnen wir mit den *Dezimalstellen*, die festlegen, auf wieviele Stellen nach dem Komma dBWin auszugebende Zahlen bei der Anzeige rundet.

Hierfür gibt es auch wieder einen *set*-Befehl, so dass Sie diese wichtige Option bei Bedarf jederzeit innerhalb eines Programms ändern können.

```
set decimals to <zahl>      && 0 bis 18
```


4. Eine Tabelle für Termine

So, jetzt haben Sie die Konfiguration kennengelernt und dabei eine Reihe nützlicher Dinge zu dBWin erfahren. Vielleicht haben Sie nicht alles bis ins letzte Detail verstanden, aber das macht nichts, das kommt mit der Praxis.

Damit Sie in diese Praxis Stück für Stück hineinwachsen, werden wir jetzt gemeinsam ein Programm mit dBWin entwickeln. Genauer gesagt werden es zwei Programme sein: eine Adressen-Verwaltung und ein Termin-Planer. Zwei Bereiche also, die sich zum einen sehr schön miteinander verknüpfen lassen, die aber andererseits auch unabhängig voneinander nutzbar sind.

Natürlich kann im Rahmen dieses Buches kein Mega-Projekt entstehen, das es mit professionellen Programmen dieser Art aufnehmen kann. Wir werden aber eine Basis schaffen, die Sie bei Gefallen gern selbst beliebig und nach Ihren eigenen Wünschen und Vorstellungen ausbauen und verfeinern können.

Daher gleich zu Beginn der ausdrückliche Hinweis: es steht Ihnen frei, ob Sie sich exakt an die hier aufgezeigten Vorgaben halten, oder davon abweichen. Sie können z. B. in den zu erstellenden Datenbanken gleich von Anfang an weitere Felder einfügen und diese dann im Programm, in den Dialogen und Eingabemasken mit berücksichtigen. Tun Sie es bitte, wenn Ihnen danach ist. Seien Sie flexibel und neugierig, spielen Sie. Halten Sie sich nicht einfach nur buchstabengetreu an das Buch, sondern sind Sie kreativ, experimentieren Sie!

Natürlich, wenn Sie das Beispiel 1:1 übernehmen wollen und kein Interesse haben, dem Beispielprogramm durch eigene Ideen eine individuelle Note zu verpassen, so ist das ebenfalls Ihr gutes Recht. Letztendlich ist es Ihre Sache und Sie wissen selbst am besten, welcher Weg für Sie der Beste sein dürfte. Ich kann und will Ihnen nur Anregungen geben, alles andere liegt bei Ihnen.

i Dieses beispielhafte Programm wird natürlich nicht alle Feinheiten von dBWin oder alle Möglichkeiten der Windows-Programmierung enthalten. Ich sagte in der Einführung ja bereits, das hier wird kein 1.000 Seiten Wälzer ... Es ist eine Einführung, die Ihnen einige Grundlagen vermittelt, auf denen Sie dann aufbauen können. Nicht mehr, aber auch nicht weniger soll es sein. Auf dieser Basis können Sie dann Ihre weitere Arbeit mit dBWin aufbauen. Und es wird zugleich auch die Grundlage für alle weiteren Bücher dieser Reihe sein.

4.1 Tabellen mit dem Experten anlegen

Beginnen wir mit der geplanten Termin-Datenbank. Sie lässt sich bequem im Regiezentrum neu anlegen. Aktivieren Sie bitte das Regiezentrum, wechseln Sie dort auf die Seite *Tabellen* und doppelklicken Sie die Auswahl *Neu*. Als Alternative geht auch der Weg über das Menü *Datei - Neu - Tabelle*.

Jetzt haben Sie die Auswahl, sich von einem *Experten* beraten zu lassen oder doch lieber zum *Designer* zu greifen. Wir werden unsere erste Datenbank mit dem *Experten* erstellen, denn dieser hält für den gewünschten Zweck schon eine recht brauchbare Vorlage bereit. In der später noch anzulegenden zweiten Tabelle greifen wir dann auf den *Designer* zurück, so lernen Sie beide kennen.



Experte oder Designer, Sie haben die Wahl

i Die Wahl zwischen Designer und Experte haben Sie nur, wenn es in den *Einstellungen des Desktop* (Menü *Eigenschaften*) auf der Seite *Anwendung* aktiviert ist. Holen Sie dies bitte nach, wenn Sie an dieser Stelle bei sich keine Wahl zwischen Experte und Designer haben, da Sie den Experten benötigen.

Der Experte für Tabellen ist eine Sammlung von Datenbank-Vorlagen, die Sie für häufige Anwendungsfälle als Grundlage verwenden können. Dazu gehören auch Adressen, Aufträge, Rechnungen, Projekte, ToDo-Listen und so weiter

Wenn Sie eine Datenbank für solch einen Zweck entwerfen wollen kann es hilfreich sein, wenn Sie sich einer solchen Vorlage bedienen. Mit ihr haben Sie schnell einen grossen Teil der nötigen Felder und brauchen nur noch die für Ihren individuellen Bedarf fehlenden Felder hinzufügen. Evtl. müssen Sie noch die Längen und Eigenschaften der vordefinierten Felder etwas anpassen.

Klicken Sie bitte auf *Experte*. Sie sehen einen Dialog, der Ihnen diverse Vorlagen für häufig vorkommende Anwendungsfälle vorschlägt.



Der Experte hält viele Tabellen-Vorlagen bereit

Wählen Sie bitte zuerst in der linken Liste *Tabellen* die passende Vorlage. Für unser Beispielprogramm ist die Tabelle *ToDo-Projektplanung* gut geeignet. In der mittleren Liste sehen Sie die Felder dieser Vorlage, und unter den Listen den Datentyp und die Feldbreite des gerade angewählten Feldes.

Ganz rechts ist die Liste der von Ihnen bereits ausgewählten Felder. Mit den Schaltern zwischen rechter und mittlerer Liste können Sie einzelne oder auch alle Felder Ihrer Auswahl hinzufügen oder wieder aus der Auswahl entfernen.

Für unser Beispiel übernehmen Sie bitte die oben in der Abbildung gezeigten Felder. Die beiden übrigen Felder bleiben in der mittleren Liste, die brauchen wir nicht, bzw. ich werde sie für das Beispiel nicht verwenden. Wenn Sie sie trotzdem mit einfügen wollen tun Sie es ruhig, das ist Ihre Entscheidung.

Wie Kleidung von der Stange passen diese Vorlagen nicht immer optimal und das eine oder andere für Sie wichtige Feld fehlt vielleicht. Das ist in der Praxis aber kein Problem, denn erstens können Sie Felder verschiedener Vorlagen auswählen und zu einer Tabelle zusammenfügen. Ändern Sie dazu einfach die Tabellen-Vorlage in der linken Spalte, die bisherige Feldauswahl bleibt davon unberührt. Und zweitens können Sie die so erstellte Tabelle bei Bedarf später mit dem *Tabellen-Designer* noch gezielt an Ihre Vorstellungen anpassen.

Klicken Sie nach Auswahl aller relevanten Felder auf **[Weiter]**. Es erscheint ein neuer Dialog, in dem Sie zwischen Anzeige der Tabelle in einem einfachen Browse-Fenster oder der Bearbeitung der Tabellen-Struktur wählen können.

5. Eine Tabelle für Adressen

Nach diesem ersten kleinen Exkurs in die Welt von dBWin können wir uns nun ganz entspannt dem Anlegen einer etwas grösseren Datenbank widmen. Diesmal geht es um eine kleine Adressen-Verwaltung und damit auch wieder um ein Thema, das Sie bei Bedarf selbst sinnvoll anwenden und nach Ihren eigenen Vorstellungen und Wünschen individuell erweitern können.

Aber diesmal verwenden wir den Tabellen-Designer und die Funktionen zur Bearbeitung der Daten werden nicht per Menü sondern mit Befehlen gesteuert. Es geht also deutlich mehr ins Eingemachte und Sie werden weniger mit der Maus als mit der Tastatur arbeiten und viele dBWin-Befehle kennenlernen.

Wenn Sie das vorherige Kapitel durchgearbeitet haben sind Sie vielleicht wie ich zu der Überzeugung gelangt, dass Experten und Menüs zwar den Einstieg erleichtern und für sehr simple Aufgaben gerade noch geeignet sind. Aber wenn es etwas mehr sein soll, und das ist wohl die Regel, dann ist auch mehr Arbeit, mehr Wissen, mehr Erfahrung und mehr Engagement nötig.

Die Arbeit an der Tabelle der Termine ist aber noch lange nicht abgeschlossen. In einem späteren Kapitel werden wir gemeinsam dazu ein Formular erzeugen, wie natürlich auch noch für die Adressen. Wenn Sie dieses Termin-Formular jetzt sofort und noch vor der Entwicklung der Adressen-Datenbank erstellen wollen, so können Sie das natürlich auch tun. Springen Sie dann einfach ein Kapitel weiter und machen danach hier wieder weiter.

Ich habe das Anlegen der Adressen-Tabelle und deren Bearbeitung bewusst gleich an den Anschluss der Termin-Tabelle gesetzt, damit Sie einen besseren und direkteren Vergleich beider Methoden und Vorgehensweisen erhalten. Wenn Ihnen eine andere Reihenfolge lieber ist, bitteschön, von mir aus gern.

i Auch für das zweite Beispiel gilt wieder: ich möchte Ihnen Anregungen und hilfreiche Tips geben. Ob Sie das alles genau so umsetzen, oder ob Sie es mit eigenen Ideen und Vorstellungen verfeinern und variieren, liegt bei Ihnen. Ich kann Sie nur nochmals dazu ermuntern, sich nicht einfach nur blind an die Vorgaben zu halten, sondern von Anfang an auch eigene Gedanken, Vorlieben und Ziele aktiv mit einzubeziehen. Schliesslich lernen Sie das hier gezeigte nicht nur zum Zeitvertreib, schon garnicht für mich, sondern für sich selbst und für Ihre zukünftigen Pläne mit dBWin, wohin diese auch führen mögen.

5.1.1 Datentypen und Spalten im Designer


Bevor wir damit beginnen, die einzelnen Felder für unsere Adressen zu definieren, zuerst eine Erklärung der Spalten im Tabellen-Designer.

Feld

Dies ist lediglich ein Zähler, der jedem Feld eine eindeutige fortlaufende Nummer gibt. Sie können und brauchen diese Nummern nicht ändern.

Name

Der Name des Feldes. Bis 10 Zeichen sind erlaubt und es wird nicht zwischen Groß- und Kleinbuchstaben unterschieden. Das erste Zeichen muss immer ein Buchstabe sein, danach können Zahlen und auch der Bindestrich verwendet werden. Umlaute sind erlaubt, weitere Sonderzeichen nicht. Sie können zwar andere Sonderzeichen eingeben, aber dBWin wird sich später mit der Meldung *Invalid Character ...* darüber beschweren. Verwenden Sie klare und eindeutige Namen, die den Zweck des Felds beschreiben. Auch wenn 10 Zeichen etwas knapp sind, mit etwas Phantasie finden sich immer aussagekräftige Namen.

 Ich rate von Umlauten im Namen grundsätzlich ab. Auch wenn dBWin das erlaubt, Sie wissen nicht wie es in späteren Situationen, wenn Sie mal mit anderen Programmen auf die Tabellen zugreifen, vielleicht Probleme macht. Auch Buchstaben wie z. B. é, è, â etc. sind für Feldnamen völlig ungeeignet.

Typ, Länge und Dezimal

Die Infos fasse ich in der Erklärung zusammen, da Länge und Dezimalstellen vom Feldtyp abhängen. Beim Datenbank-Level 5, auf das ich mich hier im ersten Band einmal beschränken werde, gibt es die folgenden Feldtypen:

- **Zeichen**, alphanumerisch, für beliebige Texte

Länge ab 1 bis max. 254, keine Dezimalstellen. Für längere Texte als 254 müssen sogenannte *Memo*-Felder verwendet werden (kommt gleich noch).

- **numerisch**, für Zahlen aller Art, egal ob mit oder ohne Dezimalanteil.

Länge ab 1 bis max. 20, max. 18 Dezimalstellen (max. 2 weniger als Länge). Die Angabe der Länge bezieht sich auf die Zahlen, ohne Dezimalzeichen und ohne Tausender-Trennzeichen. Die Länge ist auch immer die Gesamtlänge aller Zahlen, also Vor- und Nachkommastellen. Bei einem numerischen Feld der Länge 14 und 4 Dezimalstellen bleiben also noch 10 Vorkommastellen.

- **logisch**, kennt nur zwei Zustände: ja/nein bzw. true/false, Länge ist immer 1, es gibt keine Dezimalstellen. Bei der Bearbeitung werden logische Felder oft als Zeichen angezeigt: J und T (ja, true) bzw. N und F (nein, false).

6. Ein Formular für Termine

6.1 Einführung

Datenerfassung und Pflege mittels *edit* oder *browse* wie in den ersten Kapiteln gezeigt ist für den Anfang ja ganz nett. Aber von einem modernen Programm erwarten Sie sicher etwas mehr als diese doch sehr simplen Grundfunktionen.

Dazu gibt es in dBWin die Formulare. Sie sind nichts anderes als von Ihnen selbst gestaltete Fenster zur Anzeige, Erfassung und Änderung von Daten jeglicher Art. Diese Formulare enthalten Eingabefelder, Bilder, Buttons und was eben sonst noch so alles zu einem typischen Windows-Fenster gehört.

Grundsätzlich bietet Ihnen dBWin zwei Varianten, über Formulare auf Daten in Tabellen zuzugreifen. Die erste ist einfach. Über eine *View*, eine „Sicht“ auf eine Tabelle wird die Verbindung vom Formular zu den Daten hergestellt. Im Formular können Sie die Daten per *Browse*-Objekt oder auch über spezielle Eingabefelder für jedes Datenbankfeld sichtbar machen, und die Steuerung geht mit den bekannten xBase-Befehle wie *seek*, *replace*, *skip*, *go to* usw.

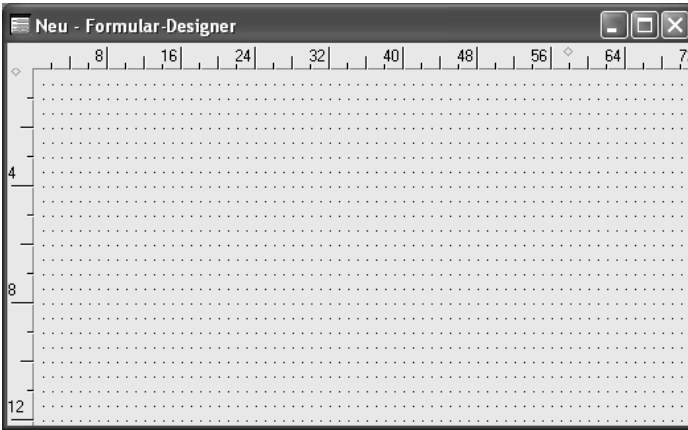
Die zweite Variante dagegen ist deutlich anspruchsvoller. Hier gibt es statt der *View* eine *Query*, eine Abfrage mittels SQL-Befehl. Das Ergebnis der Abfrage wird in einem Objekt namens *Rowset* abgelegt, ein „Satz von Zeilen“, der die einzelnen Datensätze enthält. Auch hier können Sie wieder über Befehle auf die Datensätze und einzelne Felder zugreifen, nur ist das Konzept und die zu verwendende Syntax weitaus komplizierter. Mit *use* und *skip* können Sie bei einem *Rowset* keinen Blumentopf gewinnen, da wird deutlich mehr verlangt.

Welche Variante Sie lieber und häufiger benutzen ist Ihre Entscheidung. Aber bevor Sie diese Entscheidung treffen können müssen Sie beides kennenlernen.

Ich beginne mit einem Beispiel der einfachen Version mit einer *View* auf die vorhin angelegte Tabelle der Termine. Das Beispiel im nächsten Kapitel wird dann für die Adressen die komplexere *Query/Rowset*-Version benutzen. So lernen Sie beide kennen, sehen wo Unterschiede und auch Gemeinsamkeiten sind und bekommen eine gute Basis, sich je nach Programm und Projekt mal für die eine und mal für die andere Variante zu entscheiden. Beherrschen müssen Sie sowieso beide, wenn Sie dBWin-Profi sein wollen.

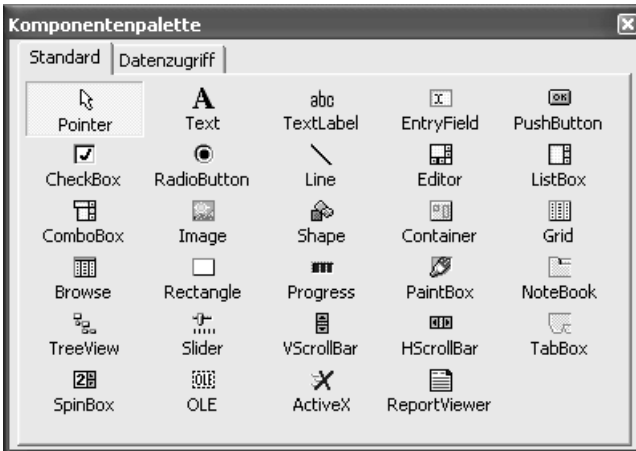
6.2 Unterfenster im Formular-Designer

Innerhalb Ihres dBWin-Hauptfensters gibt es jetzt weitere neue Unterfenster. Zuerst das leere Formular, das wir noch gemeinsam gestalten werden:



Ein neues leeres Formular wartet auf die Gestaltung

Und auch die sog. *Komponenten-Palette* hat sich parallel dazu geöffnet:



Die Komponenten-Palette mit den Objekten

i Wenn Sie die Komponenten-Palette nicht sehen öffnen Sie sie bitte über das Menü *Ansicht - Werkzeugfenster - Komponentenpalette*. Sollten sich auch noch die Fenster *Feldpalette* und *Format* geöffnet haben, so schliessen Sie die bitte, die werden wir für den Anfang nicht benötigen und da irritieren sie nur.

7. Ein Formular für Adressen

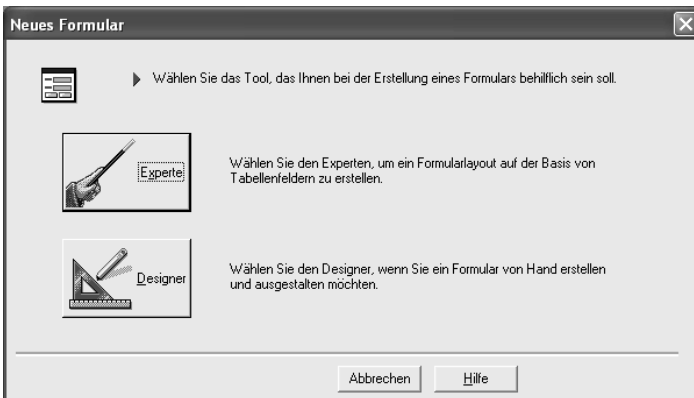
7.1 Einführung

Nachdem das Formular für die Termine doch eher leichte Kost war, vor allem wenn Sie dBase schon von früheren Versionen oder dBASE von DOS kennen, geht es in diesem Kapitel schon ein wenig mehr zur Sache.

Wir werden gemeinsam für die Tabelle mit den Adressen ein neues Formular erstellen. Dabei lernen Sie einen vollkommen anderen Weg kennen, Daten aus Tabellen in ein Formular einzubinden. Auch wenn es Ihnen nach dem letzten Kapitel am Anfang erst einmal sehr einfach vorkommt, weil Sie vieles schon kennen, dieser neue Weg wird steiler. Aber keine Angst, er ist gut begehbar.

Einiges wird sich zwangsläufig wiederholen, weil gewisse Routinearbeiten in jedem Formular anfallen. Aber sie werden schon bald weitere Eigenschaften, Ereignisse und Methoden von Formularen und Objekten kennenlernen und auch anwenden, damit Langeweile erst gar nicht aufkommt. Und natürlich werde ich auch wieder ein paar Fallen und Gemeinheiten für Sie einbauen ...

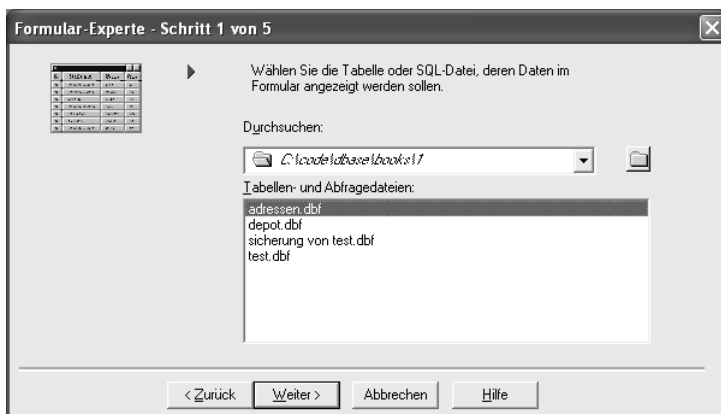
Legen Sie bitte ein neues Formular an, Sie wissen ja inzwischen wie das geht. Ob Sie das per Menü, per Tastatur oder mit der Maus machen ist völlig egal. Danach bietet sich Ihnen wieder die folgende Auswahl an, bei der Sie sich diesmal bitte für den Besserwisser, pardon für den *Experten* entscheiden.



Auswahl zwischen Experte und Designer für Formulare

Ein Klick auf den Button für den Experten führt Sie in mehreren Schritten durch einen Dialog, in dem Sie die Details zum Formular angeben können.

7.2 Der Formular-Experte



Schritt 1, Auswahl der Tabelle für das Formular

Mit dem Formular-Experten erstellen Sie Formulare, denen eine schon fertige Tabelle zugrunde liegt. Damit der Experte die Tabelle mit dem Formular verknüpfen kann müssen Sie ihm natürlich erst einmal mitteilen, welche es denn bitte sein soll. Wählen Sie die zuvor angelegte Tabelle *adressen.dbf* aus und klicken Sie anschliessend auf **[Weiter]**.

i Der Experte bietet Ihnen hier alle Tabellen zur Auswahl an, die er in den beiden Suchpfaden des Regiezentrums findet. Ist die gesuchte Tabelle nicht darunter wechseln Sie direkt hier in diesem Dialog in das Verzeichnis, in dem sich die gewünschte Tabelle befindet. Oft ist es aber sinnvoll, schon vorher die Suchpfade auf die Verzeichnisse einzustellen, in denen sich die Dateien Ihres aktuellen Projekts befinden, bzw. gleich alle Dateien eines Projekts in einem eigenen Verzeichnis zu bündeln. Details hängen aber auch vom Projekt ab.

Im nächsten Schritt können Sie all die Felder aus der Tabelle auswählen, die Sie im Formular anzeigen und bearbeiten wollen. In unserem Beispiel sind dies alle Felder bis auf *ID*, ein internes Feld, das nicht angezeigt werden soll.

Dazu können Sie entweder alle Felder bis auf *ID* einzeln mit [**>**] aus der Liste der *vorhandenen Felder* in die Liste der *ausgewählten Felder* übernehmen.

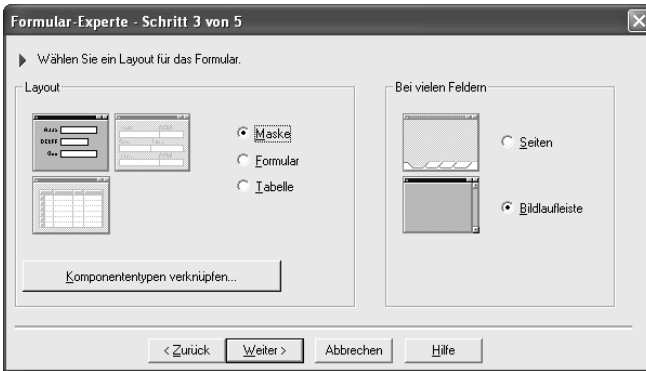
Oder Sie übernehmen gleich alle Felder auf einmal mit [**>>**] um anschliessend das Feld *ID* wieder mit [**<**] in die linke Liste zurück zu befördern. Sie kennen dieses Vorgehen vermutlich aus anderen Programmen. Wenn nicht spielen Sie ruhig ein wenig damit, um das sehr einfache Prinzip besser kennenzulernen.

Das gewünschte Ergebnis sieht nach Auswahl der richtigen Felder so aus:



Schritt 2, Auswahl der Felder für das Formular

Ein erneuter Klick auf **[Weiter]** bringt Sie dem Ziel wieder ein Stück näher. Nun wählen Sie das gewünschte Layout (hier *Maske*) und geben an, ob bei vielen Feldern für eine Seite die Felder entweder auf mehrere Fensterseiten aufgeteilt werden sollen, oder das Fenster mit einer Bildlaufleiste (Scrollbar) von oben nach unten verschiebbar sein soll. Um unser Formular nicht gleich zu Beginn unnötig komplex werden zu lassen wählen Sie bitte *Bildlaufleiste*.



Schritt 3, Auswahl des Layouts

i Sie können das Layout auch später jederzeit ändern und anpassen. Die Auswahl hier dient nur dazu, den ersten Aufbau des Formulars festzulegen, damit dBWin das erste grobe Layout autom. erstellen kann. Das von dBWin erstellte Layout kann danach von Ihnen beliebig oft bearbeitet werden. Und darum werden Sie meist auch nicht herum kommen, denn diese automatisch erstellten Formulare sind zwar eine gute Ausgangsbasis, mehr aber auch nicht.

8. Was sonst noch wichtig ist

8.1 Arbeitsbereiche

Ein Arbeitsbereich ist ein geschützter Bereich im Speicher zur Verwaltung der Daten einer Tabelle. Er schützt die Daten vor Verwechslungen mit Inhalten anderer Tabellen, die Sie vielleicht zur selben Zeit auch noch geöffnet haben.

Das Ganze ist erfreulich unbürokratisch und um die interne Verwaltung der Arbeitsbereiche brauchen Sie sich i.d.R. keine Gedanken machen, das regelt dBWin für Sie. Alles was Sie tun müssen, ist beim öffnen einer Tabelle mit dem Befehl *use* anzugeben, in welchem Arbeitsbereich dies geschehen soll. Aber selbst wenn Sie das nicht tun ist es nicht weiter tragisch. Dann wird die zu öffnende Tabelle einfach im gerade aktiven Arbeitsbereich geöffnet.

Es gibt 255 verschiedene Arbeitsbereiche, die ab 1 bis 255 nummeriert sind. In jedem dieser 255 Arbeitsbereiche kann eine Tabelle geöffnet werden, so dass im Extremfall bis 255 verschiedene Tabellen gleichzeitig offen sein können. Auch ist es möglich, dass Sie 255 mal nebeneinander dieselbe Tabelle öffnen!

i Pro Arbeitsbereich kann immer nur eine Tabelle geöffnet werden. Wenn Sie in einem Arbeitsbereich, in dem bereits eine Tabelle geöffnet wurde, eine andere Tabelle öffnen, wird die vorherige Tabelle dabei autom. geschlossen. Beim Start von dBWin ist autom. immer der Arbeitsbereich Nr. 1 aktiviert.

Arbeitsbereiche sind relevant, wenn Sie Tabellen mit *use <tabelle>* öffnen. Sie können entweder den gewünschten Arbeitsbereich direkt beim Befehl mit angeben, oder den Arbeitsbereich vorher mit einem eigenen Befehl aktivieren.

So aktivieren Sie einen bestimmten Arbeitsbereich:

```
select [(] <arbeitsbereich> [)]
```

wobei es keine Rolle spielt, ob in dem zu aktivierenden Bereich auch eine Tabelle geöffnet ist oder nicht. Arbeitsbereiche werden auf engl. *workarea* genannt, daher lautet der Befehl, um den aktiven Arbeitsbereich abzufragen

```
? workarea()           && zeigt aktiven Arbeitsbereich
```

Zahlreiche Befehle für mit *use* geöffnete Tabellen können optional einen bestimmten Arbeitsbereich verwenden, oder benutzen den gerade aktiven.

8.2 Aliasnamen

Jede mit `use <tabelle>` geöffnete Tabelle hat auch einen sog. *Aliasnamen*. Wenn Sie den nicht explizit beim öffnen auf einen bestimmten Namen setzen, nimmt dBWin meistens den Dateinamen als Alias. Wenn das nicht mehr geht, weil der Aliasname schon verwendet wird, nimmt dBWin den Arbeitsbereich (siehe vorheriges Kapitel) und setzt der Nummer einen Unterstrich `_` voran.

Sie können den Alias einer offenen Tabelle nicht mehr ändern, aber abfragen.

```
? alias()
```

ermittelt den Aliasnamen des aktiven Arbeitsbereichs. Dagegen ermittelt

```
? alias ( <arbeitsbereich> )
```

den Aliasnamen des als Parameter übergebenen Arbeitsbereichs.

```
use adressen in (1)           && Klammern optional
use termine in (2)
? alias ( 1 )                 && Ausgabe: ADRESSEN
? alias ( 2 )                 && Ausgabe: TERMINE
use in (1)                    && schliesst Bereich 1
use in (2)                    && schliesst Bereich 2
```

Für den Arbeitsbereich 1 wurde der Alias anhand des Dateinamens verwendet, für den Arbeitsbereich 2 ebenfalls. Völlig problemlos, beide heissen ja anders.

```
use adressen in (1)
use adressen in (2)
? alias ( 1 )                 && Ausgabe: ADRESSEN
? alias ( 2 )                 && Ausgabe: _2
close all                     && schliesst alle Bereiche
```

Für den Arbeitsbereich 1 wurde der Alias anhand des Dateinamens verwendet, für den Arbeitsbereich ging das nicht mehr, da hier nochmals dieselbe Tabelle geöffnet wurde. Also hat dBWin für den Bereich 2 den Alias `_2` generiert.

```
use adressen in (1) alias erster
use termine in (2) alias zweiter
? alias ( 1 )                 && Ausgabe: ERSTER
? alias ( 2 )                 && Ausgabe: ZWEITER
```

Hier haben Sie beim öffnen explizit einen bestimmten Aliasnamen vergeben, und der wird für die Tabelle in diesem Arbeitsbereich verwendet. Das ist im Grunde die übliche und häufigste Art, Aliasnamen zu verwenden. Denn dann haben Sie es fest in der Hand, welcher Aliasname vergeben wird.

8.3 *dB*ASE im Netzwerk

Prinzipiell sind *dB*Win und die damit erstellten Programme gut im Netzwerk einzusetzen. Das war ja nicht immer so, aber diese Probleme sind gelöst. Was bleibt ist die Frage, wie man Netzwerk-Programme und Setups richtig angeht.

Grundsätzlich gilt es beim Thema Netzwerk die zwei Bereiche *Installation* und damit Einsatz des Programms, und *Programmierung* zu unterscheiden.

Übrigens, wenn ich in diesem Abschnitt von einem *Server* rede, so muss das nicht unbedingt ein „richtiger“ Server sein, der „nur“ als Server verwendet wird. Auch ein Arbeitsplatz, der einen Teil seiner Festplatte anderen PCs im Netz zur Verfügung stellt, kann als Server (genauer als Peer-Server) dienen.

8.3.1 *Installation* im Netzwerk

Das hier gesagte gilt sowohl für die *Installation* von *dB*Win selbst, als auch für die von Ihnen damit erstellten und vertriebenen Programme. Die sich stellende Frage ist immer dieselbe: wohin mit dem Programm und wohin mit den Daten.

Grundsätzlich sind hierfür mehrere Lösungen denkbar, alle sind nach meiner Erfahrung mit *dB*Win praxistauglich, mit individuellen Vor- und Nachteilen. Welcher Weg für Sie besser ist hängt mal wieder von einigen Details und von den konkreten Umständen bei Ihnen ab. Da ich die nicht kenne hier einfach mal ein paar allg. erläuterte Varianten, wie sie oft in der Praxis vorkommen.

Programm und Daten auf dem Server:

Die häufigste Lösung, am schnellsten installiert und am einfachsten gepflegt.

Vorteile: genau das, schnelle Einrichtung und bequemes Handling. Updates müssen nur einmal installiert werden, die Daten liegen zentral am Server und sind auch in die (hoffentlich regelmässige) Datensicherung mit einbezogen.

Nachteile: ggf. schlechtere Performance durch Zugriffe über's Netz. Nutzer der Arbeitsplätze brauchen die nötigen Rechte auf dem Server, um Daten und Dateien anlegen, ändern und löschen (wegen temporärer Dateien) zu dürfen.

Programm auf dem Server, Daten auf dem lokalen Arbeits-PC:

Eher selten anzutreffen, und nur dann wirklich sinnvoll, wenn ein Anwender im Netz mit eigenen und speziell für ihn aufbereiteten Daten arbeitet, auf die andere Anwender keinen Zugriff haben sollen.

Vorteile: schneller und relativ sicherer Datenzugriff von lokaler Festplatte.

Nachteile: lokale Daten sind ggf. nicht gut aufgehoben, evtl. kein Backup.

9. Anhang

9.1 Ein Wort zum Abschied

Liebe Leserin, lieber Leser, ich möchte mich an dieser Stelle herzlichst für Ihr Interesse, Ihre Geduld und Ihr grosses Engagement bedanken, das Sie gezeigt haben, wenn Sie das Buch bis hierhin durchgearbeitet haben.

Sie werden sicher davon profitiert haben. Der eine vielleicht etwas mehr, die andere evtl. etwas weniger, da hängt von den jeweiligen Vorkenntnissen ab. Und davon, wie Sie die gezeigten Beispiele in der Praxis mitgemacht haben.

Wie bereits erwähnt ist dies nur ein Buch einer Serie zu dBWin. Infos zu weiteren geplanten, in Arbeit befindlichen oder bereits erhältlichen Themen finden Sie auf der Website des Autors, die auf den ersten Seiten vorgestellt wurde. Ein Blick dorthin lohnt und Sie bleiben immer auf dem Laufenden. Weitere Bände für Einsteiger, Fortgeschrittene und Profis sind geplant.

Wenn Sie Fragen, Anmerkungen, Lob und Tadel oder sonstiges zu diesem Buch loswerden wollen, dann tun Sie es bitte. Wir alle lernen aus Fehlern, profitieren von Anregungen und Meinungen von anderen, und es gibt wohl kaum etwas, das man nicht immer noch ein bisschen besser machen könnte.

Auf meiner privaten Website finden Sie auch einen kleinen Fragebogen zum Download. Sie würden mir sehr helfen, wenn Sie ihn ausfüllen. Vielen Dank.

Sie finden im Anhang des Buchs keine Web-Links zu dBASE unter Windows. Dazu ist das Internet zu vielen und zu schnellen Änderungen unterworfen. Die Gefahr, dass viele der hier abgedruckten Links nicht mehr funktionieren ist einfach zu gross. Es würde Sie und mich nur ärgern, wenn ich hier viele Links aufführe, von denen dann nach ein paar Monaten sowieso die Hälfte nicht mehr funktionieren. Aber dazu gibt es ja die Suchmaschinen ...

Abschliessend noch der Hinweis, dass es bei künftigen Versionen von dBWin, die sich stark von der für dieses Buch verwendeten Version unterscheiden, vermutlich eine Ergänzung oder je nach Bedarf auch eine Neuauflage dieses Bandes geben wird. Kleinere Ergänzungen und Korrekturen können Sie bei Bedarf aber auch auf meiner Website finden.

Nun bleibt mir nur noch eines: *Ihnen viel Erfolg und natürlich auch viel Freude bei der Programmierung mit dBASE unter Windows zu wünschen!*

9.2 Weitere Bücher dieser Reihe

Dieses Buch ist nur eines einer ganzen Reihe von neuen Büchern zu dBWin. Bei Fertigstellung dieser Ausgabe sind die folgenden Titel verfügbar/geplant:

Band 1 - Einführung

(Einsteiger, DOS-Umsteiger)

Installation - Konfiguration von dBWin und BDE - kurze Einführungen zu Befehlsfenster und Regiezentrum - Tabellen anlegen und verwalten - Daten erfassen, ändern und löschen - Formulare erstellen - Beispielprogramme u. a.

Band 2 - Grundlagen

(Einsteiger, DOS-Umsteiger)

Weitere Details zu Befehlsfenster, Regiezentrum und Projekt-Explorer - Quelltext-Editor - Projekte verwalten - EXE-Programme - INI-Dateien - Basiswissen Programmierung - Variablen, Geltungsbereiche, Datentypen - Defines und Präprozessor - Formular-Komponenten - globales Objekt _app - Parameter an Programme - Datenbankfelder - Indizes - Tips und Tricks u. a.

Band 3 - Klassen und Objekte

(Fortgeschrittene)

Basisklassen - eigene Objekte - objektorientierte Programmierung - Vererbung (in Planung, irgendwann 2006, bitte auf der Website des Autors informieren).

i Insbesondere die Bände 1 und 2 ergänzen sich in idealer Weise. Es gibt nur sehr wenig Überschneidungen und es empfiehlt sich, dBWin mit Band 1 kennenzulernen und danach mit Band 2 tiefer in diese Materie einzusteigen. Das in den beiden Bänden 1 und 2 vermittelte Wissen wird in allen folgenden Ausgaben, die dann Schwerpunktthemen haben, als bekannt vorausgesetzt.

Weitere Ausgaben sind geplant, bzw. teilweise bereits in Arbeit. Mögliche Themen sind u. a. Schnittstellen zu anderen Programmen (Excel, Word u.a.), Datenformate, SQL, OLE, DEO, DLL, Programmierung der Windows-API, Debugging und Fehlermanagement, Profitips aller Art, Drucken und Reports, Internet-Anwendungen, Umstieg von DOS, Windows-Spezialitäten etc.

Wenn Sie das hier lesen gibt es vielleicht bereits weitere Ausgaben. Einfach mal ab und an auf meiner Homepage vorbeischaun (in jedem Buch auf den ersten Seiten zu finden), oder bei (Internet-)Buchhändlern stöbern.

Bis auf weiteres werden die Bücher als „Books on Demand“ herausgegeben. Das hält Aufwand und Kosten im Rahmen und ermöglicht dennoch eine gute Verbreitung über die üblichen Vertriebswege. Alle Bände sind sowohl direkt bei mir, als auch bei allen wichtigen Internet-Händlern und natürlich auch in (fast) jedem „normalen“ Buchladen zu beziehen. Falls Ihr lokaler Buchhändler keine „Books on Demand“ bestellen will kaufen Sie künftig eben woanders ...

9.3 Stichwortverzeichnis

?	24, 285 u.v.a.
!	281
&	127
==	45
->	273
, &&, //, / ... */ (Kommentarzeichen)	261, 262
.and., .or., .not.	139
_app	63, 66, 76, 287
_porientation	287
A	
alias	65, 157, 272
and, .and.	139
ansi()	280
append	48, 108, 145
Arbeitsbereiche	267
Arbeitssitzungen	90
ASCII-Dateien exportieren	143
autoCenter (Objekteigenschaft)	246
automem	109, 146
average	140
B	
BDE (Erklärung)	19, 25, 68, 290
Befehlsfenster	23, 27
Betriebssystem ermitteln	284
bof()	106
browse (Befehl)	78, 81
Browse (Formular-Komponente)	155
build	60
C	
calculate	140, 279
Cantaria GmbH	11
cd	52
cdow()	207
chooseprinter()	282, 285
chr()	167, 230, 238
class::	217, 244
class ... endclass	206, 214, 243
clear	29, 109
close()	28, 246, 250
close printer	286
Codeblock	166, 249
continue	131
copy file	142, 176, 258
count	113, 140, 279
create form	188
... session	54
... table	103
ctod()	181
CUA-Eingabe	49, 77

D

dataLink (Objekteigenschaft)	178, 198
date()	141, 180
Datentypen (in Tabellen)	94
Datumstrennzeichen	38
dBASE-Versionen	9
default (Objekteigenschaft)	250
define	248
delete	111, 116, 121, 171
Dezimalzeichen	37
diskspace()	284
display	67, 292
do	238, 244
do ... until/while Schleifen	131
dos, DOS-Befehle, DOS-Umstieg	281, 291
dow()	207
dQuery	22, 56
drucken, Drucker	285

E

edit	47, 78, 102
Entryfield (Formular-Komponente)	174, 177, 247
Ereignisse (in Formularen)	166, 173, 180, 201, 232, 263
Editor (Formular-Komponente)	177
Editor (von dBWin, oder externer)	53, 212
Eigenschaften des Befehlsfensters	29
... des Formular-Designers	191
... des Regiezentrams	32
eMail des Autors	10
empty()	146, 251
eof()	46, 106
exit	134
exklusiv (öffnen von Tabellen)	96, 278

F

fields (Objekteigenschaft)	160, 198
file()	116, 146
Filter	138
flush	109
fontBold (Objekteigenschaft)	251
for ... next	27
found()	46, 129, 132, 268
function (Objekteigenschaft)	174, 193, 249
Function	214
Funktionszeiger	249

G

getcolor()	283
getdirectory()	282
getenv()	284
getfile()	282
getfont()	283
go bottom, go top	107, 110, 268
go record	110, 138, 170, 271
group (Objekteigenschaft)	250

H

help.....	30
height (Objekteigenschaft)	28, 193
Hotkeys	166, 182, 235, 262

I, J

Icon (Symbole von dBWin).....	20
id().....	279
if ... else ... endif.....	27, 36, 127, 146
index on.....	121, 125, 139
Indexe verwalten.....	96, 117
Installation	18, 276
Jahrhundert (Einstellung beim Datum).....	38, 51

K

Knowledgebase.....	290
Kommentare.....	261, 262
Komponententypen (in Formularen)	48, 151, 194
Konvertierung (alter Programme und Reports).....	288

L

Langdriver, language driver (BDE).....	69
Layout (Browse- und Edit-Fenster).....	75, 105
left (Objekteigenschaft)	28, 193
left().....	119
Level (für Datenbanken).....	70, 101
Line (Formular-Komponente)	177, 194
list structure.....	99
Lizenznummer	19
local.....	146
LocalShare (BDE).....	278
locate	129, 269
lock()	279
Löschmarken.....	44, 81, 90, 111
ltrim()	119

M

maxLength (Objekteigenschaft).....	180
MDI-Fenster (Formulare und Dialoge)	20, 246
memory().....	284
Methoden	213
modify file.....	62
... form.....	172, 190
... structure	117
msgbox()	238

N

ndx (Endung für ältere Indexdateien).....	96, 124
net use (DOS-Befehl)	277
network().....	279
Netzwerk	42, 276
new	28, 242, 248
nosave	79
not, .not	139
noupdate.....	79
NULL.....	48, 215

O

Objekt-Inspektor	101, 152, 192
oem()	280
on error, on neterror	64
onChange	205, 216, 254
onClick	166, 225
onGotFocus	257
onLeftDbClick	180
onLostFocus	257
onNavigate	209, 218
onOpen	173, 255
open()	28, 246
or, .or	139
os()	284

P

pack	44, 115, 278
picture (Objekteigenschaft)	180, 193, 249
private	146
Procedure	214
property	248
Pushbutton (Formular-Komponente)	164, 220

Q

Quell-Aliase	65
Query	196, 228

R

Radiobutton (Formular-Komponente)	252
rangeMax, rangeMin (Objekteigenschaften)	181, 201
readmodal()	242, 246, 250
recall	114
reccount()	106, 114, 268
recno()	46, 106, 170, 271
resize()	292
refresh()	169
Regiezentrum	31
reindex	122, 292
release, release()	28, 242
replace	108, 137, 279
Ressourcen	222
return	146
rlock()	279
round()	58
Rowset	196, 225
rtrim()	119
run, run()	281
Runtime	224, 290

S

scan ... endscan	136
SDF-Dateien exportieren	143
seek, seek()	46, 133, 175, 268
select()	146, 267
selectAll (Objekteigenschaft)	264

set	34
... alternate	54
... autonullfields	48
... autosave	44, 80, 236
... bell	50
... blocksize	43
... century	38
... confirm	49
... console	24, 167
... coverage	61
... cuaenter	49
... currency	40
... date	38, 40
... dbtype	43
... decimals	57
... deleted	44, 81, 90, 111
... design	62
... development	61
... directory	52
... editor	53
... encryption	44
... epoch	51
... error	64
... escape	50
... exact	45, 132
... exclusive	42
... fields	141
... filter	138
... fullpath	62
... headings	60
... iblock	43
... ldcheck	41, 280
... lock	42
... margin	59
... mark	38
... near	46, 135
... order	46, 133, 139, 269
... path	52
... point	35
... precision	59
... printer	285
... procedure	65, 244
... refresh	43
... reprocess	43
... safety	62
... separator	37
... space	59
... talk	60
... typeahead	50
set()	36, 40
SetFocus	180
setto()	40
shared	278
show object	169

Signalton	50
skip	110, 268
sort	144
speedtip (Objekteigenschaft)	224, 264
Speicher abfragen	284
Spinbox (Formular-Komponente)	177, 201
Sprachtreiber	41, 69, 280
SQL	56, 81, 197
Standard-Dialoge von Windows	282
Startbildschirm (abschalten)	56
Statuszeile	76
step (Objekteigenschaft)	201
store	109, 146
str()	26, 125
style	286
sum	140, 279
Systemdatum ändern	40
T	
Tab-Reihenfolge in Formularen	182, 264
Tabellen-Designer	93
Tabellen-Experte	72
tag(), tagcount()	127
Tasten (zur Navigation durch Datensätze)	82
Tastaturpuffer	50
Tausender-Trennzeichen	37
text (Objekteigenschaft)	28, 166, 192, 204
Text (Formular-Komponente)	177, 202
this	207, 217, 248
Tooltip	224, 264
top (Objekteigenschaft)	28, 193
U	
Umgebungsvariablen	284
unlock	279
upBitmap (Objekteigenschaft)	221
upper()	46, 119
use	46, 78, 278
V	
validdrive()	284
value (Objekteigenschaft)	174, 199, 207
version()	25
Verzeichnis (aktuelles einstellen)	32, 52
View (Formular-Komponente)	154
visible (Objekteigenschaft)	193
W	
width (Objekteigenschaft)	28, 193
with ... endwith	248
workarea()	146, 267
X, Y, Z	
zap	111
Zeichenketten vergleichen	44
Zeichensatz (für Tabellen)	41, 69, 280