

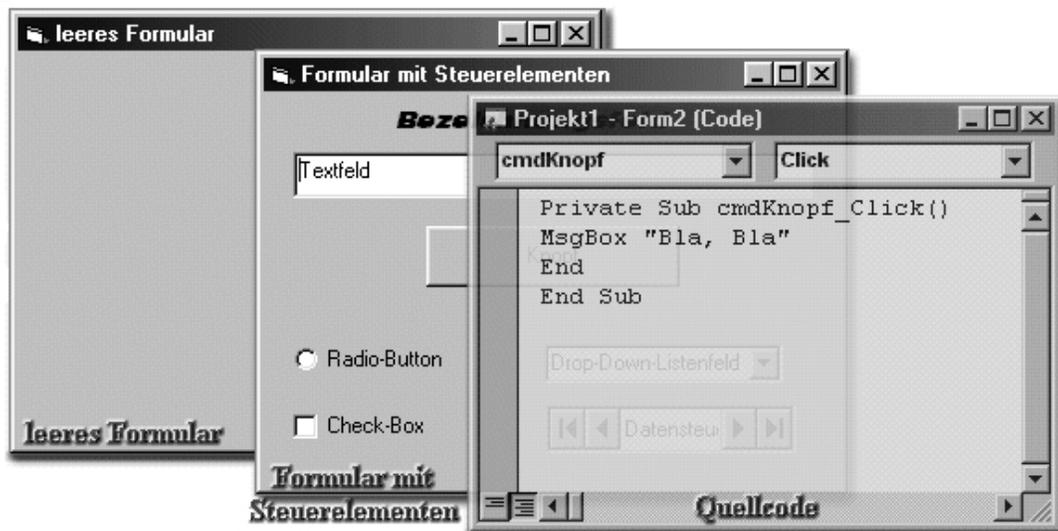
# Visual Basic Einführung

## Einführung

Visual Basic ist zwar nicht so verbreitet (und auch nicht so professionell) wie C++ und dessen Entwicklungsumgebung Visual C++, aber es ist ebenfalls eine vielverwendete und vor allem einfacher zu erlernende Programmiersprache als C++. Visual Basic müsste korrekt Visual BASIC heißen, denn der Name setzt sich aus 'Visual', das für die Fähigkeit visuell programmieren zu können und 'BASIC', was eine Programmiersprache ist. BASIC ist die Abkürzung für **B**eginners **A**ll-purpose **S**ymbolic **I**nstruction **C**ode.

## Die Entwicklung vom Visual Basic

Bei Visual Basic werden leere Fenster für einen erzeugt (*siehe Bild* leeres Formular), in die man nur noch die gewünschten Elemente einfügen muss (Buttons, Textfelder, usw.) (*siehe Bild* Formular mit Steuerelementen). Die Elemente, die einsetzbar sind, sind nicht nur auf die von Visual Basic mitgelieferten beschränkt, sondern da es sich um ActiveX-Steuerelemente handelt, vielseitig erweiterbar. So sind ActiveX-Steuerelemente bekanntlicher Weise auch im Internet erhältlich (z.B. [jcom](http://jcom)). Zum Schluss muss man den einzelnen Elementen noch sagen was sie zu tun haben (Fenster öffnen, rechnen oder wie im Beispiel auf dem Bild eine Meldung mit der Aufschrift "Bla, Bla" als Meldung ausgeben). Dies geschieht über die Eingabe von Quellcode der Programmiersprache Basic (*siehe Bild*). Die Programmiersprache Basic ist sehr leicht erlernbar. Die Entwicklung von Programmen mit Visual Basic ist garnicht so schwer:



## Visual Basic im Detail

Solltest du Visual Basic noch nicht besitzen und dich nun entschieden haben, dir eine solche Entwicklungsumgebung zu kaufen, dann musst du für Visual Basic hast du die Auswahl zwischen VB 6 und VB.NET. VB.NET ist eine weiterentwickelt Version von Microsoft und benötigt die .NET-Runtime-Environmentdateien. Ich selbst arbeite noch Visual Basic 6 und alle Workshops basieren aus VB6. Aber meines Wissens sollten die Programme auch mit der neuen Version von Visual Basic funktionieren. Für VB 6 wird vielfach noch verkauft, bei eBay wird man jedoch ganz bestimmt fündig. Für VB .NET Standart sind ganze 128 € (*Stand: 03/2003 BASIS1*) fällig.

### Was bekommt man?

Mit Visual Basic 6 bekommt man eine Entwicklungsumgebung, die visuell arbeitet, 32bit-Programme erzeugt, ADO (active data objects) beinhaltet, DHTML beherrscht, die durch ActiveX-Steuerelemente beliebig erweiterbar ist, usw.

Solltest du dir Visual Basic dann gekauft haben, stehst du da. Bei Visual Basic liegt kein Handbuch bei, nur die msdn-Bibliothek, die einen mit Informationen überschüttet. Für Anfänger, die noch nichts mit Visual Basic gemacht haben, gibt es kaum Informationen.

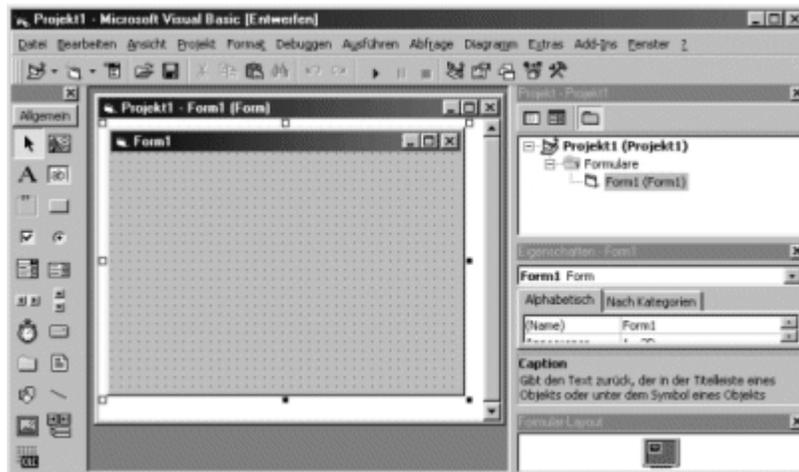
### Programmierungsumgebung

Unter der Programmierungsumgebung versteht man die Oberfläche von VB (Visual Basic) in der die Programme entworfen werden (siehe Bild unten). Da man die ganze Zeit in ihr und mit ihr arbeitet, sollte man sich etwas mit der IDE auskennen.

### Die IDE

Die IDE (*integrated development environment*: Integrierte Entwicklungsumgebung = Programmierungsumgebung) besteht aus einem großen Fenster mit vielen Symbolleisten und andockten Fenstern und einem darin integrierten Projektfenster, in dem die Form zu sehen ist, die man gerade bearbeitet. Was auf dieser Form zu sehen ist, sieht auch im späteren Programm so aus. Die vielen Symbolleisten enthalten Werkzeuge mit denen Objekt auf dem Form platziert werden können und mit denen man dann deren Eigenschaften festlegen kann. Außerdem sind die Fenster zur Verwaltung der einzelnen Fenster des Programms zuständig.

Zu den Symbolleisten und Fenstern nun mehr.



### Werkzeugleiste

Es gibt neben der Microsoft-typischen Symbolleiste am oberen Rand des Fensters noch eine weitere Symbolleiste. Diese Werkzeugleiste genannte Symbolleiste befindet sich ganz links und beherbergt verfügbaren Steuerelemente mit denen die Form bestückt wird. Für jedes geladene Steuerelement gibt es in der Symbolleiste ein kleines Symbol. Will man das Steuerelement auf der Form platzieren, so klickt man das Symbol an und zieht es dann auf der Form auf. Also Symbol anklicken, mit Mauszeiger zur Form gehen, Maustaste klicken und gedrückt halten und ein Quadrat aufziehen, dann wieder los lassen und fertig ist das Steuerelement. Ist die Größe nicht perfekt oder ist die Position falsch, so kann man das Steuerelement auch noch verän-

dern. Zum Verschieben auf das Steuerelement klicken, Maustaste gedrückt halten und an die korrekte Position verschieben. Zum Verkleinern Steuerelement anklicken, dann an der rechten unteren Ecke anklicken und bei gedrückter Maustaste in die richtige Größe verkleinern.

Man kann zusätzlich Steuerelemente über "**Projekt/Komponenten**" tun, dort findest du eine Vielzahl von Steuerelementen, welche die am häufigsten verwendeten Steuerelemente sind erfährst du in dem dazu gehörigen Workshop. Aus dem Internet heruntergeladene Steuerelemente findest du ebenfalls hier. Einfach ein Häkchen neben dem Steuerelement machen und es wird danach in der Werkzeugleiste angezeigt. Entfernt wird es wieder in dem du das Häkchen wieder entfernst.

Es reicht meist nicht die Steuerelemente einfach nur dort zu platzieren, fast immer sind noch zusätzliche Eigenschaften wie Namen oder Beschriftung zu verändern. Dies geht in dem du auf das Steuerelement klickst und in dem Eigenschaftenfenster die Eigenschaften bearbeitest.

Fast alle Steuerelemente sollen auf Aktionen des Benutzers reagieren, was passieren soll, legst du fest in dem du Ereignisprozeduren in Visual Basic schreibst. Einen Ereignisprozedurrohling erhältst du, wenn du auf das Steuerelement doppelklickst. Dann gelangst du in das Codefenster wo du den Quellcode eingeben musst.

#### **Projekt-Fenster**

Das Projektfenster habe ich bereits vorhin vorgestellt, in ihm siehst du wie das Fenster der Anwendung später aussehen wird. Auf dem Formular werden die Steuerelemente platziert. Die Größe des Formulars wird wie die Steuerelemente verändert. Wie bei den Steuerelementen werden auch die Eigenschaften für das Formular in dem Eigenschaftenfenster vorgenommen.

#### **Projekt-Explorer**

Im Projekt-Explorer, der sich rechts oben befindet, siehst du die Ressourcen des gerade aktiven Projekts befinden. Meist sind das Formulardateien, manchmal auch Module in die du für alle Formulare gültigen Code ausgelagert hast. Die Dateien die dein Projekt umfasst werden hier in einer Verzeichnisbaumstruktur angezeigt. Wenn du das Formular wechseln willst, so kannst du dies über dieses Fenster tun. Doppelklicke einfach auf den Eintrag und schon erscheint das gewünschte Formular. An der oberen linken Ecke befindet sich ein Umschalter zwischen dem Design- und dem Quellcodemodus des Formulars. Wenn du in den Quellcodemodus wechseln willst, so klicke auf das Symbol, wenn du wieder zurück in den Designmodus willst, klicke auf das andere Symbol. Der jeweils gewählte Modus wird darauf hin im Vordergrund eingeblendet.

#### **Eigenschaftsdialog**

Der Eigenschaftsdialog ist neben dem Projektfenster, dem Quellcodefenster und der Werkzeugleiste einer der wichtigsten Bestandteile der IDE. Alle Steuerelemente haben Eigenschaften. Eine der Aufgaben des Programmierers ist es die anzupassen, so muss man manchmal die Beschriftung oder den Namen ändern. Dies alles ist im Eigenschaftendialog möglich.

Die Eigenschaften sind tabellarisch aufgelistet, neben dem Eigenschaftsname steht immer die aktuelle Einstellung dazu. Für Anfänger steht auch noch mal unter dem Eigenschaftsfenster eine kurze Beschreibung der Eigenschaft.

Um eine Eigenschaft zu ändern, klickt man auf die Einstellung daneben, daraufhin kann man bei manchen Eigenschaften in einer Liste die richtige Einstellung auswählen oder man muss die neue Eigenschafteneinstellung eingeben.

Will man die Eigenschaften eines anderen Steuerelements bearbeiten, so kann man dieses in der Liste über der Eigenschaftsanzeige.

#### **Formularlayout**

Im Formularlayout schließlich kann man sehen wo das Fenster bei der späteren Ausführung angezeigt werden wird. Bei Bedarf kann man die Position den eigenen Wünschen anpassen.

#### **Wichtige Funktionen der restlichen Menü- und Symbolleisten**

Wie bei allen Microsoft-Programmen gibt es die Symbolleiste und die Menüleiste. Manche Symbole daraus werden dir aus anderen Programmen bekannt vorkommen. Die restlichen möchte ich nun kurz aufzählen.

Mit dem ersten Symbol kann man dem Arbeitsbereich ein neues Projekt hinzufügen. Dies ist jedoch nicht zu empfehlen. Mit dem Symbol daneben werden dem Projekt neue Formulare, Module und weitere Ressourcen hinzugefügt. In der Standardeinstellung werden Formulare hinzugefügt, möchtest du etwas anderes hinzufügen, so klicke auf den kleinen Pfeil daneben, nun sollte eine Liste mit weiteren Möglichkeiten erscheinen. Mit dem dritten wird der Editor für Menüs für das Formular aktiviert.

Die nächsten acht Symbole sollten dir bekannt sein (Öffnen, Speichern, Ausschneiden, Kopieren, Einfügen, Suchen, Rückgängig, Wiederherstellen).

Das nächste Symbol startet das aktuelle Programm, mit dem nächsten Symbol wird das Programm angehalten und mit dem letzten wieder beendet.

Die nächste Symbolgruppe blendet die Fenster Projekt-Explorer, Eigenschaftsfenster und Formularlayoutfenster wieder ein, falls sie ausgeblendet wurden. Das nächste Symbol blendet den Objektkatalog ein, in dem alle Eigenschaften aller zur Zeit verfügbaren Objekte (wenn Steuerelemente hinzugefügt werden, werden es mehr) aufgelistet und beschrieben werden. Diese Auflistung ist jedoch nur für Fortgeschrittene interessant. Mit dem letzten Symbol wird schließlich die Werkzeugleiste wieder eingeblendet.

Die restlichen Angaben danach zeigen die Größe während dem aufziehen eines Steuerelements an.

#### **Programme ausführen und weitergeben**

Wie erwähnt wird mit dem Pfeil in der Symbolleiste oder unter *Ausführen/Starten* das Programm ausgeführt, jedoch ist dafür immer die Entwicklungsumgebung von VB nötig.

Willst du das Programm an Freunde weitergeben oder im Internet veröffentlichen, so muss du eine ausführbare exe-Datei erstellen. Unter *Datei* findest du den Menüpunkt "Projekt.exe erstellen" (Projekt steht für den Projektnamen), damit lässt sich die angesprochene exe-Datei erstellen. Neben der Tatsache, dass das Programm noch auch ohne Visual Basic lauffähig ist, wird auch nur noch eine Datei weitergegeben und nicht die vielen Ressourcen aus denen das Projekt besteht. Außerdem ist die Datei kleiner als die Quellcodedateien.

Noch einen Schritt weiter kannst du mit dem von VB mitgelieferten "Weitergabe- und Verpackungs"-Assistenten gehen. Dieser erstellt nicht nur eine ausführbare Datei, sondern schreibt auch noch ein Setup-Programm mit dem dein Programm auch auf Systemen die noch keine VB-Runtime-Dateien installiert haben laufen. Denn VB-Programme sind nicht ohne die Runtime-Dateien lauffähig. Leider umfasst die VB-Runtime mehr als 1,5 MB, so dass sich VB-Programme (zumindestens für Modemuser) nur bedingt für das Internet eignen.

### Die Hilfe

Auf der msdn-CD-ROM findest du zu jedem Element eine meist ausführliche Beschreibung. Gehe dafür, wenn du in der msdn-Hilfe bist (In VB über Hilfe aufrufbar), auf den Registerreiter "Suchen" am linken Bildschirmrand. Es werden dann die Treffer aufgelistet. Auch als Profi wirst du noch mit der msdn arbeiten. Sie wird für dich eine Art Nachschlagewerk für Funktionen und Eigenschaften werden, denn kein Programmierer hat alle Funktionen im Kopf!

**Diesen und viele andere Workshops gibt es auf  
[www.abbyter.de](http://www.abbyter.de)**