

## INTRODUCTION

Gary Cornell

The Visual Basic 4 for Windows 95 Handbook

Osborne McGraw-Hill  
2600 Tenth Street  
Berkeley, California 94710  
U.S.A.

For information on translations or book distributors outside the U.S.A., or to arrange bulk purchase discounts for sales promotions, premiums, or fundraisers, please contact Osborne McGraw-Hill at the above address.

**The Visual Basic 4 for Windows 95 Handbook**

Copyright \_ 1996 by McGraw-Hill, Inc. All rights reserved. Printed in the United States of America. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but they may not be reproduced for publication.

1234567890 DOC 99876

ISBN 0-07-882091-X

Information has been obtained by Osborne McGraw-Hill from sources believed to be reliable. However, because of the possibility of human or mechanical error by our sources, Osborne

McGraw-Hill, or others, Osborne McGraw-Hill does not guarantee the accuracy, adequacy, or completeness of any information and is not responsible for any errors or omissions or the results obtained from use of such information.

## Contents

<b>1</b>	<b>GETTING STARTED</b>	<b>1</b>
	Why Windows and Why Visual Basic?,	2
	What You Need to Run Visual Basic,	3
	What Does the Newest Version of Visual Basic 4 Offer You?,	4
	Setting Up Visual Basic,	5
	Running the Setup Program,	5
	Starting Visual Basic,	6
	Working with Visual Basic,	8
	Running an Existing Visual Basic Project,	10
	The Online Tutorial,	11
	Overview of How You Develop a Visual Basic Application,	13
<b>2</b>	<b>THE VISUAL BASIC ENVIRONMENT</b>	<b>17</b>
	An Overview of the Main Screen,	19
	Title Bar,	20
	Menu Bar,	21
	Toolbar,	22
	Toolbox,	22
	The Initial Form Window,	24
	Project Window,	25
	Shortcut (Pop-up) Menus,	26
	The Help System,	27
	Help Menu,	27

More on the Visual Basic Help System,	30
Using the File Menu,	34
Editing,	39
Edit Menu,	39
Using the View Menu,	41
Using the Insert Menu,	42
Using the Run Menu,	43
Using the Tools Menu,	44
Using the Add-Ins Menu,	48
<b>3 CUSTOMIZING A FORM</b>	<b>51</b>
Starting a New Project,	52
The Properties Window,	54
Common Form Properties,	58
Scale Properties,	64
Color Properties,	65
The BackColor and ForeColor Properties via the ColorPalette,	65
Making a Form Responsive,	67
The Code Window,	68
Printing a Form,	74
Typos,	74
Saving Your Work,	77
Saving from the File Menu,	77
Creating Stand-Alone Windows Programs,	79
<b>4 FIRST STEPS IN BUILDING THE USER INTERFACE</b>	<b>83</b>
The Toolbox,	85

Creating Controls,	87
Working with an Existing Control,	89
Shortcut for Creating Controls,	91
Working with Multiple Controls,	92
Locking Controls,	93
Deleting Controls,	93
Properties of Command Buttons,	93
The Caption Property,	94
The Name (Control Name) Property,	96
Other Useful Properties for Command Buttons,	97
Shortcuts for Setting Properties,	99
Simple Event Procedures for Command Buttons,	99
Other Events for Command Buttons,	102
Some Final Points on Command Buttons,	103
Access Keys,	104
Image Controls,	105
Text Boxes,	106
Standard Properties of Text Boxes,	107
Some Special Properties for Text Boxes,	107
Event Procedures for Text Boxes,	110
Labels,	110
Useful Properties for Labels,	111
Event Procedures for Labels,	112
Navigating Between Controls,	112
Access Keys for Text Boxes,	113

Message Boxes,	113
The Grid,	116
What Happens When a Visual Basic Application Runs,	118
The Display in Visual Basic,	120
The AutoRedraw Property for Forms,	121
The Refresh Method,	122
The ASCII Representation of Forms,	122
<b>5 FIRST STEPS IN PROGRAMMING</b>	<b>127</b>
Anatomy of a Visual Basic Program,	128
The Code Window,	129
The Split Bar,	131
The Object List Box,	131
The Proc(edure) List Box,	131
Statements in Visual Basic,	131
Remark Statements,	133
The End Statement,	133
Writing Code: The Editor Page on the Options Dialog Box,	134
Assignment and Property Setting,	136
Assigning to Properties,	137
Variables,	140
Variable Types,	141
Fine Points of Variables,	144
The Dim Statement for Types,	145
Requiring Declaration of Variables,	147
Changing the Defaults for Types,	148

Scope of Variables,	149
Sharing Values Across Procedures,	150
The Private Keyword,	151
Having Values Persist,	151
Strings,	152
ASCII/ANSI Code,	153
Fixed-Length Strings,	156
Input Boxes,	157
Numbers,	159
Operations on Numbers,	161
More on Numbers in Visual Basic,	163
Bits, Bytes, and Hexadecimal (Base 16) Numbers in Visual Basic,	165
Example Program: A Mortgage Calculator,	168
Improvements to the Mortgage Calculator,	171
Constants,	173
The Supplied Constant File,	174
Projects with Multiple Forms,	177
Writing Code for Multiple Forms,	177
How to Handle Multiple Forms at Run Time,	179
An Example Using Multiple Forms,	182
Keeping the Focus in a Form (Modality),	183
<b>6     DISPLAYING INFORMATION</b>	<b>187</b>
Displaying Information on a Form,	188
Example: Centering Text Inside a Form,	189
The Font Properties in Code,	191

Displaying Tabular Data,	192
The Tab and Spc Commands and Semicolons,	193
The Format Command,	194
Picture Boxes,	198
Working with Picture Boxes,	199
RichTextBoxes,	201
Properties for RichTextBoxes,	201
A RichTextBox Example,	203
The Printer Object,	207
Useful Properties and Methods for the Printer,	208
The Printers Collection,	210
Printing Information in a RichTextBox,	211
<b>7 CONTROLLING PROGRAM FLOW</b>	<b>213</b>
Repeating Operations,	214
Determinate Loops,	214
Indeterminate Loops,	223
Making Decisions,	233
The Block If-Then,	236
What Is It?,	238
Using the If-Then with a Message Box,	239
An Example: The Dir\$ Command,	239
Combining the If-Then with Loops,	240
Select Case,	241
Finishing Up with the If-Then,	243
An Example: The KeyPress Procedure,	246

An Example: Working with the QueryUnload Event, 247

Conditional Compilation, 248

The GoTo, 250

## **8 BUILT-IN FUNCTIONS 253**

String Functions, 254

Analyzing Strings, 255

The Like Operator and Fuzzy Searching, 266

Other String Functions, 267

The Rnd Function, 268

An Example Using the Random Number Generator, 274

Bit Twiddling, 277

Logical Operators at the Bit Level, 279

KeyUp and KeyDown, 282

Numeric Functions, 284

Date and Time Functions, 287

The Date Function, 287

Numeric Calendar Functions, 288

Financial Functions, 291

Other Financial Functions, 294

The Remaining Financial Functions, 296

The Most Useful Functions, 297

## **9 BUILDING LARGER PROJECTS: PROJECTS, PROCEDURES, AND ERROR TRAPPING 301**

User-Defined Functions and Procedures, 302

Function Procedures, 303

Advanced Uses of Functions,	308
Sub Procedures,	319
Advanced Uses of Procedures and Functions: Passing by Reference/Passing by Value,	327
Subprograms with an Optional Number of Arguments,	333
Standard (Code) Modules: Global Procedures and Global Variables,	335
Adding or Removing Code Modules,	339
Fine Points About the Scope of Procedures,	339
Accessing Windows Functions,	340
The DoEvents Function,	344
Error Trapping,	346
More on the Err Object,	350
Some General Words on Program Design,	351
<b>10 ARRAYS, GRIDS, AND RECORDS</b>	<b>357</b>
Control Arrays,	358
Adding and Removing Controls in a Control Array,	361
Example: A Square Array of Labels via Control Arrays,	364
Lists: One-Dimensional Arrays,	369
Fixed Versus Dynamic Lists,	370
Some Ways of Working with Lists,	373
Example: Improving the Built-in Random Number Generator,	375
Lists with Index Ranges,	376
The Erase Statement,	377
Arrays with More Than One Dimension,	378
Assigning Arrays to Variants: The Array Function,	379

Example: Magic Squares,	380
Using Lists and Arrays with Procedures,	384
The Grid Control,	387
General Properties of the Grid Control,	388
Properties of Selected Cells Inside Grids,	391
Events and Methods for Grid Controls,	393
Magic Squares Using a Grid Control,	394
Sorting and Searching,	396
Searching,	396
Sorting,	400
Records (User-Defined Types),	407
The With Statement,	411
Pointer Mimicking,	412
<b>11 OBJECTS</b>	<b>415</b>
Getting Started with Object-Oriented Programming,	416
Vocabulary of OOP,	418
Manipulating Objects Built into Visual Basic,	420
Manipulating Object Variables via Code,	425
Collections,	428
An Example of Using the Controls Collection with Set,	429
Building Your Own Collections,	430
The Object Browser,	433
Creating an Object in Visual Basic,	436
General Property Procedures,	438
Building Your Own Classes,	439

Creating a New Class Module, 440

An Example: A Deck of Cards Class Module, 440

## **12 THE INTERFACE REVISITED 445**

The Toolbox Revisited, 446

Frames, 448

Option (Radio) Buttons, 450

Check Boxes, 452

List and Combo Boxes, 453

Scroll Bars, 461

Timers, 465

Menus, 470

Menu Designer, 471

Sample Menu Design Window, 474

Working with Menus at Run Time, 474

Common Dialog Boxes, 478

A Simple File Viewer Example, 479

Working with Common Dialog Boxes, 481

MDI Forms, 488

The Window Menu and the Arrange Method, 489

More on How Visual Basic Displays Work: ZOrder, 490

Some Words on Windows Design, 491

Making Forms Independent of Resizing and Screen Resolution, 492

## **13 FINISHING THE INTERFACE 499**

Custom Controls in the Professional and Enterprise Editions, 501

Sheridan 3-D Custom Controls, 501

DesaWare Animated Button Control, 503

Microsoft Comm Control, 504

MicroHelp Gauge Control, 506

MicroHelp Key State Control, 508

MAPI Controls, 510

Microsoft Masked Edit Control, 510

Microsoft Multimedia Control, 512

Outline Control, 512

Microsoft Picture Clip Control, 513

OutRider Spin Button Control, 514

Sheridan Tabbed Dialog Control, 515

Windows 95 Controls, 517

ImageList Control, 517

ListView Control, 519

ProgressBar Control, 520

Slider Control, 521

StatusBar Control, 522

TabStrip Control, 524

ToolBar Control, 524

TreeView Control, 527

Toolbars and Status Bars for 16-bit Windows, 527

Help Systems, 529

Writing Help Topics, 530

Building In Context-Sensitive Help, 534

Building and Compiling the Help Files, 535

Accessing the Windows Help Engine, 536

What's This Help?, 538

## **14 TOOLS AND TECHNIQUES FOR TESTING AND DEBUGGING 541**

The Debugging Tools and What They Do, 543

Testing Programs, 544

Designing Programs to Make Testing Easier, 545

Logical Bugs, 546

The Debug Window, 547

More on Debugging in the Debug Window, 548

Single Stepping, 550

Stopping Programs Temporarily, 552

Using Breakpoints to Test Programs, 552

Final Remarks on Debugging, 558

Event-Driven Bugs and Problems, 559

Documentation and Program Style, 559

## **15 AN INTRODUCTION TO GRAPHICS 563**

Fundamentals of Graphics, 566

A Feature of the AutoRedraw Property, 567

The ClipControls Property and the Paint Event, 568

More on the Paint Event, 569

The Refresh Method, 569

Saving Pictures, 569

An Example: Simple Animation, 570

Screen Scales, 571

Custom Scales, 573

The Line and Shape Controls,	575
The Shape Control,	576
The Line Control,	579
Graphics via Code,	579
Colors,	579
Pixel Control,	580
An Example Program: "Visual Basic A-Sketch",	583
Lines and Boxes,	587
Last Point Referenced,	588
Relative Coordinates,	590
Grid Graphics,	592
DrawWidth, DrawStyle,	593
Boxes,	595
Animation and DrawMode,	603
Circles, Ellipses, and Pie Charts,	605
An Example: Pie Charts,	608
Ellipses and the Aspect Ratio,	609
Curves,	610
Pictures Without Too Many Formulas,	611
Polar Coordinates,	616
The PaintPicture Method,	620
The Graph Control,	622
Using the Graph Control,	624
<b>16 MONITORING MOUSE ACTIVITY</b>	<b>633</b>
The Mouse Event Procedures,	634

The MouseUp/MouseDown Events,	636
Dragging and Dropping Operations,	641
<b>17 WORKING WITH FILES</b>	<b>653</b>
File Commands,	655
Example: How to Reset the Logged Drive,	656
The Shell Function,	657
Command-Line Information,	658
File-Handling Functions,	660
File System Controls,	662
File List Boxes,	663
Directory List Boxes,	665
Drive List Boxes,	667
Tying All the File Controls Together,	668
Sequential Files,	668
Reading Back Information from a File,	673
Adding to an Existing File,	677
General Sequential Files,	679
Sending Special Characters to a Sequential File,	682
Making Changes Inside a Sequential File,	683
The RichTextBox Control and File Handling,	686
Making a File Program Robust: Error Trapping,	687
Random-Access Files,	689
Headers and Indexes for Random-Access Files,	693
Binary Files,	694
Using Binary Access in More General (Non-Text) Situations,	697

Final Remarks on Binary File Handling,	699
Sharing Files,	702
General Form of the Open Command,	703
Adding Licensing Strings,	704
Keeping File Information Secret,	706
More Complicated Ciphers,	707
A More Secure Cipher,	711
<b>18 COMMUNICATING WITH OTHER WINDOWS APPLICATIONS</b>	<b>715</b>
The Clipboard,	717
Selecting Text in Visual Basic,	718
Clipboard Formats and Graphics Transfers,	720
Clipboard Example Program,	721
Activating Windows Applications,	723
Sending Keystrokes to an Application,	724
Dynamic Data Exchange (DDE),	727
Creating DDE Links at Design Time,	728
DDE Properties,	730
DDE Events,	731
DDE Methods,	734
OLE 2,	735
Using OLE,	736
Creating OLE Objects,	737
Using OLE 2 at Design Time,	740
Paste Special,	742
OLE Properties,	742

OLE Automation,	745
Using OLE Automation,	745
OLE Servers,	748
Creating an Out-Of-Process OLE Server,	748
Creating an In-Process OLE Server,	751
<b>19 RECURSION</b>	<b>753</b>
Getting Started with Recursion,	755
Recursive Functions,	757
Simple Recursive Procedures,	761
An Example--The Tower of Hanoi,	762
Recursive Sorts,	768
Merge Sort,	769
Quicksort,	773
Making Sorts Stable,	775
Binary Trees,	776
Tree Sort,	782
Fractals,	784
Other Fractal Curves,	787
When Not to Use Recursion,	793
<b>20 A SURVEY OF DATABASE FEATURES</b>	<b>797</b>
Some General Words on Modern Databases,	799
The Data Manager,	801
Using the Data Manager to Examine Existing Databases,	802
Adding Information to a Database,	805
Using the Data Manager to Create New Databases,	807

Other Operations You Can Do with the Data Manager,	812
Using the Data Control,	814
Seeing the Data Control at Work,	814
Programming with the Data Control,	820
The Field Object,	821
Other Useful Methods and Events for the Data Control,	822
Closing a Recordset or Database,	825
Setting Properties via Code,	826
Monitoring Changes to the Database,	828
Transaction Control,	830
Structured Query Language (SQL) Basics,	831
More on SELECT Statements,	832
Finding Records Using SQL,	833
Modifying a Table's Data Through SQL,	833
Database Objects,	834
<b>21 A CHECKBOOK MANAGEMENT PROGRAM</b>	<b>839</b>
Designing the Program,	841
The User Interface,	843
The Startup Form and the Set Directory Form,	844
The Picture Box Used for Account Statistics Forms,	847
The Accounts Form,	848
The EStats (Edit Statistics) Form,	849
The Forms for Entering Checks and Deposits,	850
The Reports Form,	852
The Results Form,	854

The General Code for the Program,	854	
The General Procedures Module,	855	
Code for the Finance Form,	862	
Code for the SetDir Form,	865	
Code for the Accounts Form,	867	
Code for the EStats Form,	869	
Code for the Checks Form,	872	
Code for the Deposits Form,	875	
Code for the Reports Form,	878	
Code for the Results Form,	884	
How to "Objectify" This Program,	885	
<b>A</b>		
<b>DISTRIBUTING YOUR APPLICATION: THE SETUP WIZARD</b>		<b>887</b>
What a Visual Basic Executable Needs to Run,	888	
The Setup Wizard,	889	
Using the Wizard,	891	
OLE Automation Setup Problems,	895	
Database Setup Problems,	895	
<b>B</b>		
<b>MICROSOFT'S TOP 25 VISUAL BASIC TECHNICAL SUPPORT QUESTIONS</b>		<b>899</b>
Can I get help without paying Microsoft support charges?,	900	
Will the 16-bit and 32-bit versions of Visual Basic 4.0 run on Windows 95?,	903	
Why can't I load a Visual BasicX control in the 32-bit version of Visual Basic?,	903	
How can I write an application for 16-bit and 32-bit systems at the same time?,	904	
Can a Visual Basic 4.0 32-bit application run on Win32s?,	905	
Can I create DLL files with Visual Basic 4.0?,	905	
Does this mean that I can use a Visual Basic-created DLL from other languages?,	906	

Will Visual Basic 4.0 exist for other platforms?, 906

Why are there references to Macintosh in the documentation and on-line help?, 906

Does the 32-bit version of Visual Basic provide statements and functions for multitasking?, 907

Does Visual Basic 4.0 use the new controls of Windows 95?, 907

Do I need to change my Visual Basic 3.0 applications to compile with the 32-bit version of Visual Basic 4.0? What about under the 16-bit version?, 907

Do I need to change my Visual Basic 3.0 application to compile with the 16-bit version of Visual Basic 4.0?, 908

When I load an existing application into Visual Basic 4.0, it says it's going to "upgrade" my controls. Is it actually changing my controls, and should I back them up?, 908

Why are the distributables so large?, 909

I am having problems connecting to my database using open database connectivity (ODBC) and Microsoft Visual Basic for Windows. What am I doing wrong?, 909

.INI File Settings, 909

ODBC and Driver DLLs, 911

Server Information Needed to Connect to a Data Source, 912

Information Specific to Microsoft and Sybase SQL Servers, 912

How do I scroll a form when Visual Basic forms are limited to screen size?, 913

Summary, 914

Step-by-Step Example, 914

How do I create a floating toolbar in Visual Basic?, 916

Summary, 916

Creating a Floating Toolbar, 916

How do I create a modeless dialog box or form in Visual Basic?, 929

Summary, 929

Step-by-Step Example, 930

How do I bind data controls across multiple forms in Visual Basic 4.0?, 931

Summary, 931

Step-by-Step Example, 931

What are the differences between exclamation mark (!) and dot (.) syntax?, 933

Summary, 933

More Information, 933

How do I determine whether Visual Basic 4.0 App was launched as an OLE server?, 934

Summary, 934

More Information, 935

Step-by-Step Example, 935

How do I determine when a shelled 32-bit process has terminated?, 936

Summary, 936

More Information, 936

Step-by-Step Example, 937

How do I determine when a shelled 16-bit process has terminated?, 939

How do I launch a Win32 application from Visual Basic?, 940

Summary, 940

More Information, 941

How to Launch Win32 Applications from Visual Basic, 941

Step-by-Step Example, 942

How do I create and use a minimal OLE Automation server?, 944

Summary,	944
Step-by-Step Example,	945
How do I optimize OLE calls in Visual Basic 4.0?,	947
Summary,	947
More Information,	947
How do I register OLE Controls?,	949
Summary,	949
More Information,	950
How do I create scrollable viewports in Visual Basic?,	950
Summary,	950
More Information,	951
How can I get Windows 95 Logo certification for my application?,	953
<b>C VISUAL BASIC ON THE INTERNET</b>	<b>955</b>
A. Preface,	957
B. General Visual Basic Questions,	957
C. Common Visual Basic Programming Questions,	957
D. Advanced Visual Basic Programming Issues,	959
E. Calling the Windows API and DLLs in General,	960
F. Visual Basic and Databases,	961
G. Distributing Visual Basic Applications,	961
H. Miscellaneous Tips and Information,	962
I. Visual Basic for Applications (VBA),	962
A. Preface,	962

A1. Where to get the VB/Win FAQ,	963
A2. Kudos & comments,	965
B. General Visual Basic Questions,	966
B1. Does VB/Win make standalone .EXE files?,	966
B3. Where can I get updated VB and other Microsoft files?,	966
B4. Help! I am lost on ftp.microsoft.com, what do I do?,	967
B5. Where can I get good up-to-date information about VB?,	967
B6. Are there any WWW pages for VB?,	970
B7. Are there any examples of commercial applications built using Visual Basic?,	971
C. Common Visual Basic Programming Questions,	971
C5. How do I make a text box not beep but do something else when I hit the ENTER key?,	971
C7. How do I get the TAB key to be treated like a normal character?,	972
C11. Is there any way to pass a variable to a form apart from using global variables?,	972
C12. How should dates be implemented so they work with other language and country formats?,	973
C14. How do I dial a phone number without using the MSCOMM VBX?,	973
C17. How do I get my application on top?,	974
C21. How do I make my applications screen resolution-independent?,	974
C23. Why doesn't "my string" & Chr\$(13) do what I want?,	976
C26. How do I force a file dialog box to reread the current disk?,	976
C27. How do I get the number of free bytes on a disk?,	976
D. Advanced Visual Basic Programming Issues,	977
D2. How do I access C-style strings?,	977
D4. Any tips for speeding up VB?,	977
D5. How do I speed up control property access?,	978

D6. How much gain in performance will I get if I write my number crunching routines in C instead of Visual Basic?,	979
D9. How do you change the system menu (on the Control-Menu Box)?,	980
D14. How do I implement Undo?,	980
D23. How do I set the Windows wallpaper at run time?,	981
E. Calling the Windows API and DLLs in General,	982
E3. What about DLL calls that require callbacks?,	982
E4. Tips for calling DLLs (such as the Windows API),	982
G. Distributing Visual Basic Applications,	983
G2. Are there restrictions on what I can distribute with my VB program?,	983
G5. Can I distribute my app without vbrunXXX.dll?,	983
G7. Where do I install VBXs and DLLs?,	983
H. Miscellaneous Tips and Information,	984
H1. Multiple identifiers after the DIM statement can be confusing.,	984
H2. "Clean up" your project before final EXE compilation.,	985
H3. Multiple END statements can be dangerous; or, The program that refused to terminate.,	985
<b>D THIRD-PARTY TOOLS AND OTHER RESOURCES</b>	<b>987</b>
Resources,	988
Tools,	989
Design and Debugging Tools,	989
Tools for Speed and Convenience,	990
Custom Controls,	991
<b>INDEX</b>	<b>993</b>

**Acknowledgments**

One of the best parts of writing a book is when the author gets to thank those who have helped him or her, for rarely (and certainly not in this case) is a book by one author truly produced alone. First and foremost, I have to thank the team at Osborne/McGraw-Hill. Their patience, dedication, help, cheerfulness--you name it--went way beyond the call of duty. To Wendy Rinaldi, Mark Karmendy, and Jirka Vcelak... Thanks! Jonathan Zuck and especially David Jezak combined for a very useful tech review. Next, I have to thank those people at Microsoft (whose names I unfortunately don't know) who created Visual Basic and have now made it even better in version 4. It's a very useful upgrade with a lot of potential for its users. Finally, thanks to all my friends who put up with my strange ways and my occasionally short temper for lo, so many months.

## **Introduction**

When Visual Basic 1.0 was released, Bill Gates, Chairman and CEO of Microsoft, described it as "awesome." Steve Gibson in *Infoworld* said Visual Basic is a "stunning new miracle" and will "dramatically change the way people feel about and use [Microsoft] Windows." Stewart Alsop was quoted in the *New York Times* as saying Visual Basic is "the perfect programming environment for the 1990s."

So what is all the hype about? Exactly what is Visual Basic and what can it do for you? Well, Bill Gates describes Visual Basic as an "easy yet powerful tool for developing Windows applications in Basic." This may not seem like enough to justify all the hoopla until you realize that Microsoft Windows is used by millions of people, and that developing a Microsoft Windows application formerly required an expert C programmer supplied with about 20 pounds worth of documentation for the needed C compiler and the essential add-ons. As Charles Petzold (author of one of the standard books on Windows programming in C) put it in the *New York Times*: "For those of us who make our living explaining the complexities of Windows programming to programmers, Visual Basic poses a real threat to our livelihood."

Visual Basic 2.0 was faster, more powerful and even easier to use than Visual Basic 1.0. Visual Basic 3 added a simple way to control the most powerful databases available. Visual Basic 4 adds support for 32-bit development and begins the process of turning Visual Basic into a fully object oriented programming language. Moreover, unlike earlier versions of Visual Basic, the newest version allows you to seamlessly add in third part tools to make your development easier. So welcome to the latest and best version of Visual Basic, the programming tool of the 1990s!

## **About This Book**

This tutorial is a comprehensive, hands-on guide to all the ins and outs of Visual Basic 4 programming--one that doesn't assume you've programmed before. (However, people familiar with earlier versions of Visual Basic or another structured programming language will, of course, have an easier time.) You'll start at the beginning and quickly move along the road to mastery. Soon you'll be writing sophisticated Windows programs that take full advantage of Visual Basic's exciting and powerful event-driven nature. Finish this book and you will be in a position to write commercial-quality Visual Basic programs!

I've tried hard to stress the new ways of thinking needed to master Visual Basic programming, so even experts in more traditional programming languages can benefit from this book. I've taken this approach because trying to force Visual Basic into the framework of older programming languages is ultimately self-defeating--you can't take advantage of its power if you continue to think within an older paradigm.

To make this book even more useful there are extensive discussions of important topics left out of most other introductory books. There's a whole chapter on objects, including a non-trivial example of building your own objects with Visual Basic. There's a whole chapter on recursion. There are methods of keeping file information secret and ways to add protection to programs you may want to distribute. It also includes an extensive discussion of sorting and searching techniques and lots of tips and tricks. My goal was to make this book a "one stop resource." I even have an appendix on how to use the Internet to get further information about Visual Basic.

Unlike many of the books out there, I not only want to introduce you to a topic but to take you the one step further that leads you to mastery.

The main body of this book ends with a professional quality checkbook management program. This was chosen because, in my opinion, nothing shows off the power of Visual Basic better than a program like this. This program took less than a week to design and implement. It is easy to imagine an entire finance program capable of generating millions of dollars in sales written in only a few weeks using Visual Basic. (This is not simply my imagining what could happen--substantial parts of Quicken are written in Visual Basic, and UnInstaller, the most successful Windows product to come along in quite a while, was written mostly in Visual Basic.)

Finally, since most users of Visual Basic have (or eventually will have) the higher-end versions ("Visual Basic Professional Edition" or "Visual Basic Enterprise Edition," I cover many of the special features of these higher-end products into the text itself rather than relegating them to an appendix.

## **How This Book Is Organized**

This book can be used in a variety of ways depending on your background and needs. People familiar with structured programming techniques can skim the complete discussions of the programming constructs such as loops and Sub and Function procedures in Visual Basic. These appear in Chapters 5 through 11. Beginners will want to work through this material more carefully.

Here are short descriptions of the chapters:

Chapters 1 and 2 introduce you to Visual Basic and help you become familiar with the Visual Basic environment.

Chapters 3 and 4 start you right off with the notion of a customizable window (called a *form*) that is the heart of every Microsoft Windows (and thus, Visual Basic) application. You'll see how to add the Basic controls--such as command buttons, text boxes, and labels--to your forms.

Chapters 5 through 11 discuss the core programming techniques needed to release Visual Basic's powers. You'll see how to take full advantage of Visual Basic 4's various data types and control structures, as well as its many built-in functions. You also learn how to add your own functions to Visual Basic 4. You'll see how to use Visual Basic's implementation of part of the object oriented programming paradigm--including how to build your own objects with Visual Basic. You'll see how to sort and search through data, use the grid control, and use modular programming techniques to make your programs more flexible, powerful, and easier to debug.

Chapters 12 and 13 show you how to finish building the user interface. They take you through most of the rest of the controls you can add to your forms. You'll see how to add list boxes, radio (option) buttons, check boxes, scroll bars, and all the other controls that Microsoft Windows users expect in their Windows applications--and that make Windows applications so much easier to use than their counterparts running under DOS. We cover both the standard controls and those found in the Professional and Enterprise editions (including the Windows 95 specific controls.) Moreover, we indicate workarounds if you want to use a Professional edition feature and only have the standard edition whenever possible.)

Chapter 14 shows you the powerful debugging tools in Visual Basic 4. You'll learn how to isolate bugs (programming errors) and then eradicate them.

Chapter 15 introduces you to the world of graphics. Since Microsoft Windows is a graphically based environment, the powers of Visual Basic in this arena are pretty spectacular.

Chapter 16 shows you how to analyze the way a user is manipulating his or her mouse.

Chapter 17 shows you how to handle files in Visual Basic, including sophisticated methods for encrypting (that is, keeping the contents of files safe from casual probes). We especially emphasize the changes in file handling techniques between Visual Basic 4 and earlier versions of Visual Basic.

Chapter 18 introduces you to the world of dynamic data exchange (DDE), object linking and embedding (OLE), and OLE automation. DDE lets you automate the transfer of information between Windows applications--and you'll be able to have Visual Basic coordinate the transfers! OLE lets you embed other Windows applications within a Visual Basic application so users can take full advantage of many of their other Windows applications. OLE automation lets you control another Windows application using *its* native language.

Chapter 19 is an extensive treatment of *recursion*. Recursion is one of the most powerful programming tools available, and it's too often slighted in introductory books. In addition to powerful methods for sorting data, this chapter gives you a short introduction to recursive graphics, or *fractals*. Fractals are one of the most powerful tools in graphics--for example, they were used in the Genesis sequence in *Star Trek II: The Wrath of Khan*.

Chapter 20 is an introduction to the Data manager and other data access features added to Visual Basic 4. You'll see how to control the most powerful PC databases with only a few lines of code using the new data control.

Chapter 21 is the checkbook manager. It's a long program, but by this stage nothing in it will be foreign to you.

Appendix A shows you how to use the new Setup Wizard that makes distributing your files a breeze.

Appendix B surveys what resources are available for Visual Basic on the Internet.

Appendix C contains, courtesy of Microsoft, answers to what they regard as the most frequently asked questions about Visual Basic. We added this appendix in the hope that it will save you hours of waiting on the line for Microsoft's support services to help you.

Appendix D opens up to you the exciting world of third-party tools and resources.

## Conventions Used in This Book

Keys are set in small capital letters in the text. For example, keys such as CTRL and HOME appear as shown here. Arrow and other direction keys are spelled out and also appear in small capital letters. For example, if you need to press the right arrow key, you'll see, "Press RIGHT ARROW."

When you need to use a combination of keys to activate a menu item, the first two keys will be separated by a plus sign and the entire key combination will appear in small capital letters. For example, "Press CTRL A+B" indicates that you should hold down the key marked "Ctrl" on your keyboard while holding down the "A" and the "B." On the other hand, ALT F, P means press the "Alt" key, then the "F" key, and then the "P" key--you don't have to hold down the "Alt" key.

DOS commands, filenames, and file extensions appear in full capital letters: COMMAND.COM, .TXT, and so on. Microsoft Windows 95 is referred to just as Windows most of the time. Earlier versions of Windows are called Windows 3.X when they need to be referred to. Keywords in Visual Basic appear with the first letter of each word capitalized: for example, Print, DeBug, FontSize, and so on.

The syntax for a command in Visual Basic is set in ordinary type except that items the programmer can change appear in italics. For example, the Name command used to rename a file appears as

Name *OldFileName* As *NewFileName*

Also, programs are set in a monospaced font, as shown here:

```
PRIVATE SUB FORM_CLICK()  
  PRINT "HELLO WORLD!"  
END SUB
```

and lines of code that should appear on one line when you program, but which for typographic reasons need to be on more than one line in this book, will be outdented or an underscore will be used. (Visual Basic 4 added a line continuation character--the underscore--which you can use whenever you are not inside quotes.) Here are some examples:

---

PRINT "THIS LINE IS MUCH TOO LONG TO FIT ON ONE LINE  
BUT IS OUTDENTED TO INDICATE IT SHOULD BE ENTERED ON ONE LINE"

M\$ = "THIS LINE IS ALSO TOO LONG BUT IT HAS AN " + \_  
"UNDERSCORE SO IT WILL BE REGARDED AS ONE LINE BY VB"

Finally, menu choices are indicated by writing the menu name, followed by a bar (|), followed by the menu item; for example, "choose File|Open" means open the File menu and then choose the Open option.

## Disk Order Form

There are a couple hundred pages of code in this book. You may not want to enter it all. The companion disk contains all the code in *The Visual Basic 4 for Windows Handbook* with more than two lines. It also includes the form files for the figures and the more interesting projects. The disk also includes the commented source code for the version of the checkbook manager given in the text, and also one that uses the more object oriented features of Visual Basic. In addition, the disk contains around fifty bonus procedures with full source code. Here's a sample of what the bonus procedures include:

### *Business procedures for...*

Calculating internal rate of return

Analyzing a loan or annuity

Determining straight-line, sum-of-digits, or double-declining balance depreciations

### *Procedures for working with arrays...*

Sorting, insertion, and searching

### *Mathematics procedures for...*

Numerical integration

Working with vectors

Matrix algebra

To order your disk, send your name, address, along with a check or money order for **\$25.00** for the first copy (**\$3.00** for each additional copy--if you have purchased a companion disk to an earlier version of this book the price is only **\$10.00.**) to

Books on Science  
128 Moulton Road  
Storrs, CT 06268

------(cut here)-----

Please rush one companion disk for *The Visual Basic 4 for Windows Handbook* to

Name \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_  
          \_ ZIP \_\_\_\_\_

First copy @ \$25.00 \_\_\_\_\_

Additional copies at \$3.00 each \_\_\_\_\_

Upgrade at \$10.00 \_\_\_\_\_

Total enclosed is: \_\_\_\_\_

(Orders from outside the United States should use a check issued on a bank with an American branch please. Connecticut residents please add the appropriate sales tax.)

We offer a 30-day, money-back guarantee.

This is solely the offering of the author. Osborne/McGraw-Hill takes no responsibility for the fulfillment of this offer.