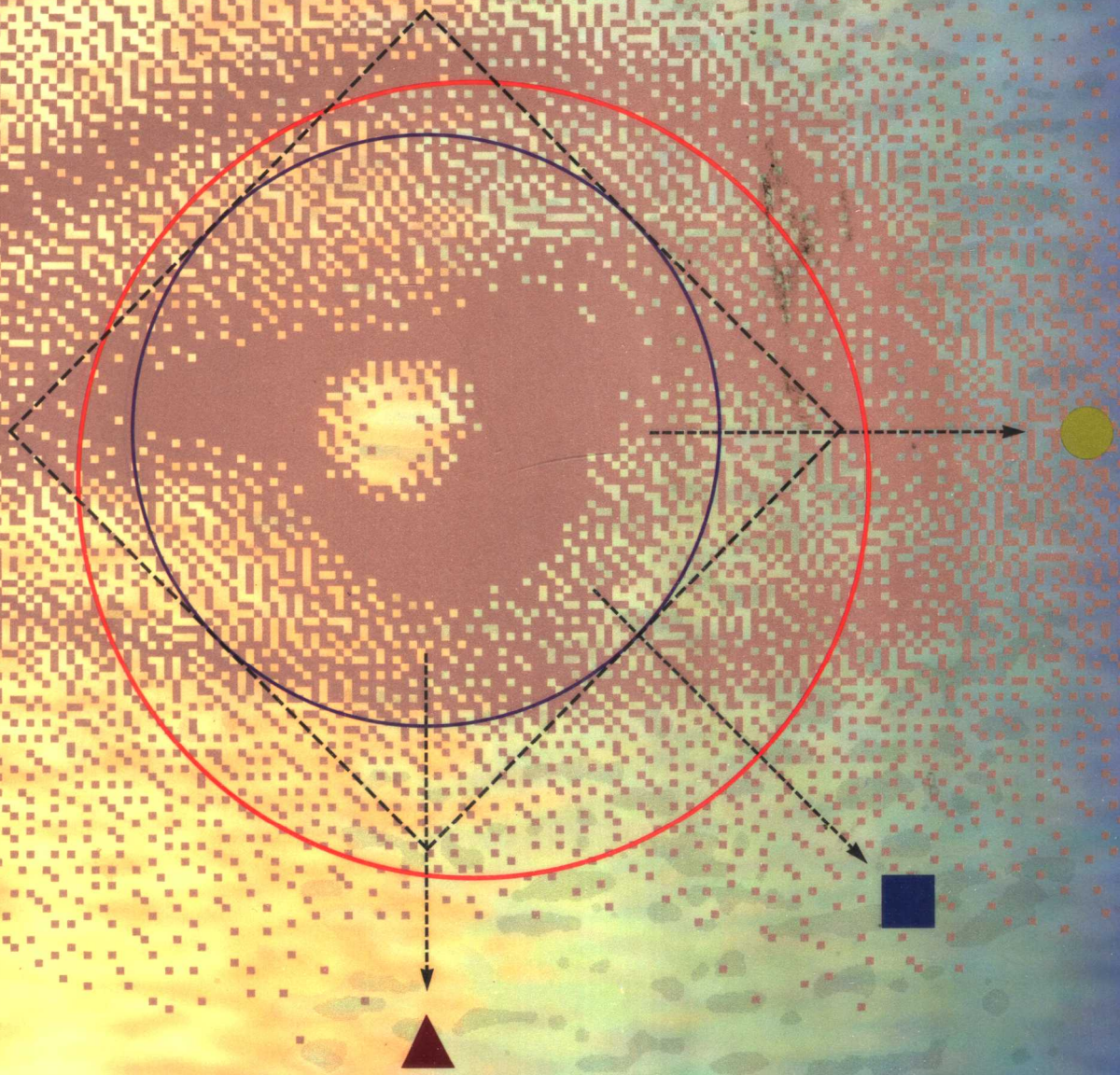
# Programming in Visual Basic

## Version 4.0



JULIA CASE BRADLEY
ANITA C. MILLSPAUGH

# Programming in Visual Basic
## Version 4.0

JULIA CASE BRADLEY

ANITA C. MILLSPAUGH

**IRWIN**

Chicago • Bogotá • Boston • Buenos Aires • Caracas
London • Madrid • Mexico City • Sydney • Toronto

© Richard D. Irwin, a Times Mirror Higher Education Group, Inc. company, 1996

*All rights reserved.* No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

**Irwin Book Team**

Publisher: *Tom Casson*
Associate editor: *Garrett Glanz*
Editorial assistant: *Carrie Berkshire*
Project editor: *Karen M. Smith*
Production supervisor: *Dina L. Genovese*
Assistant manager, graphics: *Charlene R. Perez*
Designer: *Crispin Prebys*
Compositor: *Douglas & Gayle, Ltd.*
Typeface: *11/13 Bodoni*
Printer: *Von Hoffmann Press, Inc.*

**Times Mirror**
**Higher Education Group**

**Library of Congress Cataloging-in-Publication Data**

Bradley, Julia Case.
Programming in Visual Basic, version 4.0 / Julia Case Bradley, Anita C. Millspaugh
p. cm.
Includes index.
ISBN 0-697-20273-9
1. BASIC (Computer program language) 2. Microsoft Visual BASIC.
I. Millspaugh, A. C. (Anita C.) II. Title.
QA76.73.B3B723 1996
005.26'2—dc20 95-42175

*Printed in the United States of America*
3 4 5 6 7 8 9 0 VH 3 2 1 0 9 8 7 6

# PREFACE

As the world turns to graphical user interfaces, computer programming languages are changing to accommodate the shift. Visual Basic 4.0 has been designed to allow the programmer to develop applications that run under Windows without the complexity generally associated with Windows programming. With very little effort, the programmer can design a screen that holds standard Windows elements such as command buttons, check boxes, option buttons, text boxes, and list boxes. Each of these Windows objects operates as expected, producing a "standard" Windows user interface.

Visual Basic is easy to learn, which makes it an excellent tool for understanding elementary programming concepts. In addition, it has evolved into such a powerful and popular product that skilled Visual Basic programmers are in demand in the job market.

## ABOUT THIS TEXT

This textbook is intended for use in an introductory programming course, which assumes no prior knowledge of computer programming. However, many of the later chapters are appropriate for an advanced-level course. The later chapters are also appropriate for professional programmers who are learning a new language to upgrade their skills.

It is assumed that the student is familiar with the Windows operating environment.

## APPROACH

The authors have successfully taught Visual Basic to beginning and advanced programming students for several years. This text is designed to incorporate the basic concepts of programming, problem solving, programming logic, as well as the design techniques of an event-driven language.

The student will learn to deal with a visual interface while acquiring the skills of programming such as loops, decisions, and data management.

Chapter topics are presented in a sequence that allows the programmer to learn how to deal with a visual interface while acquiring important programming skills such as loops, decisions, and data management.

The later chapters may be used in various sequences to accommodate the needs of beginning and advanced level courses, as well as a shorter quarter system or a semester-long course. For a shorter course, the professor may choose to skip the chapter on data files and cover only the first of the two database chapters.

## CHAPTER ORGANIZATION

Each chapter begins with identifiable objectives and a brief overview. There are numerous coding examples as well as a hands-on project with guidance for the coding. Thought-provoking feedback questions are interspersed throughout the chapter to allow the students time to reflect on the current topic and their understanding of the details. The end of the chapter items include a chapter review, questions, programming exercises, and two case studies. The case studies provide a continuing theme exercise that may be used throughout the course.

The first chapter walks the student through the creation of a first Visual Basic project, incorporating command buttons and labels. The programming environment is introduced along with the concepts of objects and their related properties, methods, and events.

Chapter 2 continues coverage of controls, including option buttons, check boxes, frames, and images. It also covers some of the finer points of using the environment and working with keyboard access keys, multiple controls, and alignment. The color constants are used at this point, to lead the novice programmer into chapter 3, which introduces variables and constants. The text includes a naming convention to make the scope and data type of a variable or constant easier to determine from the coding syntax. Standards also provide for the use of Option Explicit to force the declaration of all variables and constants.

Chapter 4 introduces the relational and logical operators and their use with the If statement. Control arrays and message boxes are also covered. Chapter 5 provides the programmer with the knowledge for creating sub functions and sub procedures. The Case structure is provided as an alternative for the If statement. Menus and multiple forms are first discussed in chapter 6.

Chapter 7 incorporates list boxes and combo boxes into the projects, providing the opportunity to discuss looping procedures and printing lists of information. The list concept leads logically into the use of variable arrays and grids in chapter 8, and to writing the information to disk in chapter 9. Chapter 9 covers both sequential and random files, but it may be covered in portions.

Chapters 10 and 11 deal with the use of Visual Basic as a front end for database programming. The projects display and update tables created by a database application such as Microsoft Access. Chapter 11 includes writing queries in SQL.

试读结束，需要全本PDF请购买 www.ertongbook.com

The drag-and-drop feature of Windows programming is introduced in chapter 12. This chapter normally brings great enthusiasm from students as they learn to deal with the source and target objects. The chapter examples and assignments provide a blend of practical and just-for-fun applications. This is also true of chapter 13, which provides an introduction to the graphics methods and graphic controls.

Chapter 14, the final chapter, covers a variety of topics that bridge from Visual Basic into other applications. These include the Windows API, DLLs, OLE, and Visual Basic for Applications.

## ACKNOWLEDGMENTS

We would like to express our appreciation to the many people who have contributed to the successful completion of this text. Most especially, we thank our students at Mt. San Antonio College who helped class test the material and greatly influenced the manuscript. A special thank you goes to Theresa Berry for her thorough technical review, constructive criticism, and many valuable suggestions. Thanks to Tricia for her work on the index. And most important, we are grateful to Dennis, Andy, and Eric for their support and understanding through the long days and busy phone lines.

## THE AUTHORS

We have had fun teaching and writing about Visual Basic. We hope that this feeling is evident as you read this book and that you will enjoy learning or teaching this outstanding programming language.

**Julia Case Bradley**
**Anita C. Millspaugh**

# TO THE STUDENT

The best way to learn to program in Visual Basic is to do it. If you enter and run the example projects, you will be on your way to writing Windows applications. Just reading the example projects is like trying to learn a foreign language or mathematics by reading alone. Enter the projects, look up your questions in Visual Basic's excellent Help, and make those projects *run*.

## FORMAT USED FOR VISUAL BASIC STATEMENTS

Visual Basic statements and functions are shown in `this font`. Any values you must supply are in *italics*. Optional items are in [square brackets]. Braces and a vertical bar indicate that you must choose one or the other value {one | other}.

Example:

```
Open "FileName" For {Input|Output|Append|Random} As #FileNumber [Len=RecLength]
```

As you work your way through this textbook, note that you may see a subset of the available options for a Basic statement. Generally, the options that are included will reflect those covered in the chapter. If you want to see the complete format for any statement, refer to online Help.

**J.C.B.**
**A.C.M.**

# CONTENTS

试读结束，需要全本PDF请购买 www.ertongbook.com

# INTRODUCTION TO VISUAL BASIC

**AT THE COMPLETION OF THIS CHAPTER, YOU WILL BE ABLE TO**

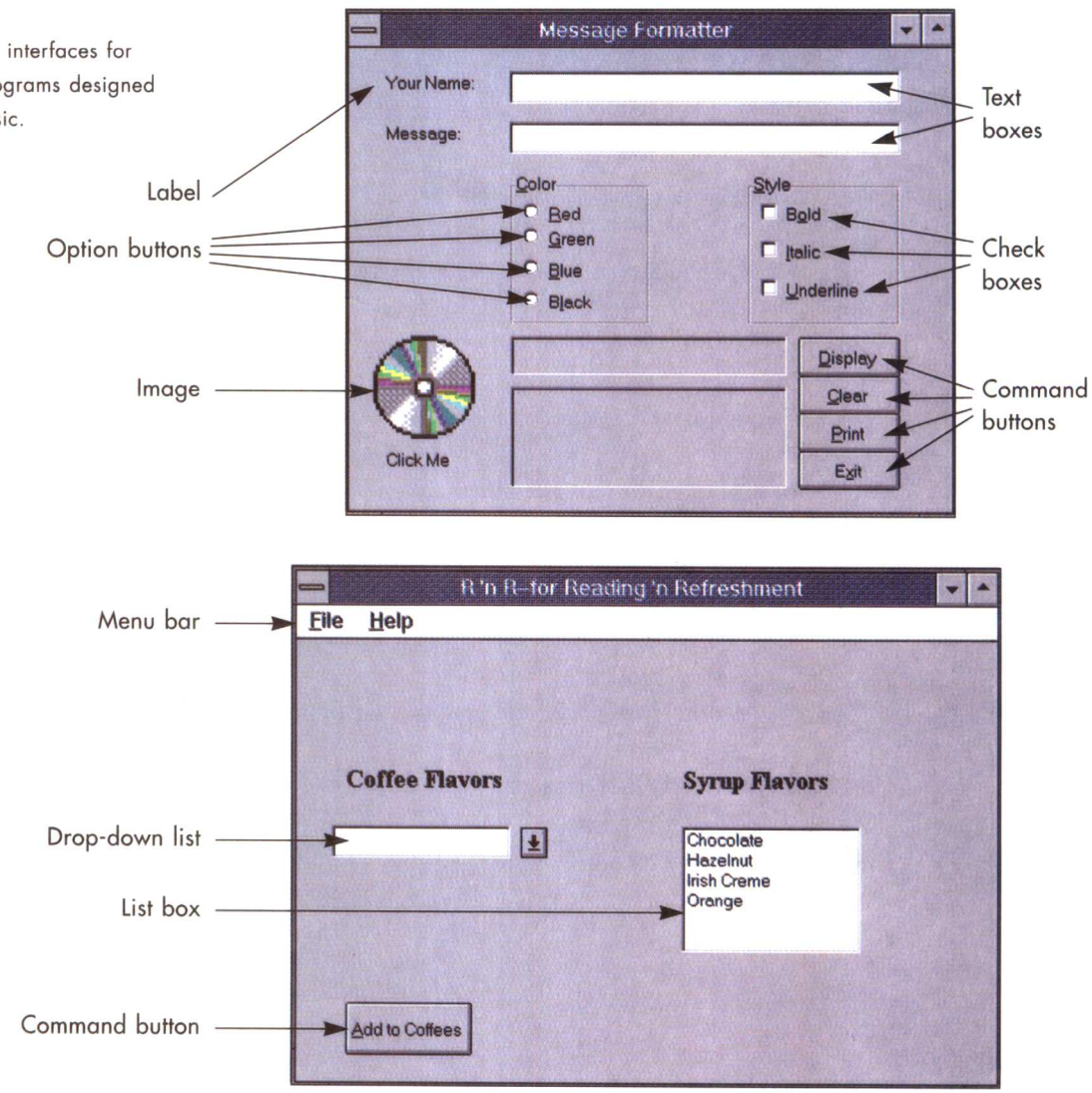
1. Describe the process of visual program design and development.
2. Explain the term *event-driven programming*.
3. Explain the concepts of objects, properties, and methods.
4. List and describe the three steps for writing a Visual Basic project.
5. Describe the various files that make up a Visual Basic project.
6. Identify the elements in the Visual Basic environment.
7. Explain the differences among design time, run time, and break time.
8. Write, run, save, print, and modify your first Visual Basic project.
9. Identify compile errors, run-time errors, and logic errors.
10. Look up Visual Basic topics in Help.

CHAPTER 1

## WRITING WINDOWS APPLICATIONS WITH VISUAL BASIC

Using this text, you will learn to write computer programs that run in the Microsoft Windows environment. Your projects will look and act like standard Windows programs. Visual Basic (VB) provides the tools you need to create windows with familiar elements like menus, text boxes, command buttons, option buttons, check boxes, list boxes, and scroll bars. Figure 1.1 shows some sample Windows user interfaces.

**Figure 1.1**
Graphical user interfaces for application programs designed with Visual Basic.




## THE WINDOWS GRAPHICAL USER INTERFACE

Microsoft Windows uses a **graphical user interface**, or **GUI** (pronounced *gooey*). The Windows GUI defines how the various elements look and function. As a Visual Basic programmer, you have available a toolbox of these elements. You will create new windows, called ***forms***.

Then you will use the toolbox to add the various elements, called ***controls***. The projects that you will write follow a relatively new type of programming, called *event-driven programming*.

# PROGRAMMING LANGUAGES— PROCEDURAL, OBJECT ORIENTED, AND EVENT DRIVEN

There are literally hundreds of programming languages. Each was developed to solve a particular type of problem. Most traditional languages, such as BASIC, C, COBOL, FORTRAN, PL/I, and Pascal are considered *procedural* languages. That is, the program specifies the sequence of all operations, step-by-step. Program logic determines the next instruction to execute, in response to conditions and user requests.

The newer programming languages, such as C++ and Visual Basic, use a different approach: ***object-oriented programming*** (OOP) and ***event-driven programming***. Microsoft refers to Visual Basic as an event-driven programming language, which has many (but not all) elements of an object-oriented language such as C++. Each release of Visual Basic moves it a little closer to a true object-oriented language.

In the event-driven model, programs are no longer procedural; they do not follow a sequential logic. You, as the programmer, do not take control and determine the sequence of execution. Instead, the user can press keys and click on various buttons and boxes in a window. Each user action can cause an *event* to occur, which triggers a Basic procedure that you have written. For example, the user clicks on a command button labeled Calculate. The clicking causes the button's click event to occur, and the program automatically jumps to a procedure you have written to do the calculation.

## THE OBJECT MODEL

In Visual Basic you will work with **objects**, which have **properties** and **methods**.

### OBJECTS

Think of an object as a thing, or a noun. Examples of objects are forms and controls. Forms are the windows and dialog boxes you place on the screen; controls are the elements you place inside a form, such as text boxes, command buttons, and list boxes.

### PROPERTIES

Properties tell something about an object, such as its name, color, size, location, or how it will behave. You can think of properties as adjectives that describe objects.

When you refer to a property, you first name the object, then a period, then name the property. For example, refer to the Caption property of a form called Form1 as Form1.Caption (say "form1 dot caption").

试读结束，需要全本PDF请购买 www.ertongbook.com