Beginning Programming FOR DUMMIES®

4TH EDITION

Wallace Wang

BICENTENNIAL
1807
WILEY
2007
BICENTENNIAL

Wiley Publishing, Inc.

Beginning Programming FOR DUMMIES®

4TH EDITION

Wallace Wang

BICENTENNIAL 1807 WILEY 2007 BICENTENNIAL

Wiley Publishing, Inc.

Beginning Programming For Dummies®, 4th Edition

Published by
Wiley Publishing, Inc.
111 River Street
Hoboken, NJ 07030-5774
www.wiley.com

Published by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

Trademarks: Wiley, the Wiley Publishing logo, For Dummies, the Dummies Man logo, A Reference for the Rest of Us!, The Dummies Way, Dummies Daily, The Fun and Easy Way, Dummies.com, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. Wiley Publishing, Inc., is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 800-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002.

For technical support, please visit www.wiley.com/techsupport.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Library of Congress Control Number: 2006932692

ISBN-13: 978-0-470-08870-8

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

4O/RW/RQ/QW/IN

WILEY

About the Author

Wallace Wang is one of many carbon-based life forms currently populating the planet Earth. He began his working career by going to college and getting a “good” job — only to find that a college education never guaranteed you a “good” job and most “good” jobs actually stink.

So faced with the prospect of spending the rest of his waking life in a caffeine-induced stupor coupled with shots of alcohol and gambling to dull the frustration of a dead-end job, he decided to pursue one of many dreams by becoming a writer. First starting out writing magazine articles for a local San Diego computer magazine, he soon graduated to writing for national publications and book publishers as well.

After nearly 20 years of writing full-time, he’s still pursuing a variety of different dreams besides continuing book and magazine writing. He’s still bouncing around comedy clubs around Michigan, Las Vegas, and San Diego, performing stand-up comedy to anyone sober enough to listen.

He’s also branched away from computer books by teaming up with coauthors to write non-computer related books, most notably *Breaking Into Acting For Dummies* with Larry Garrison. (Thanks go to Ben Affleck for carrying around a copy and getting his picture taken with the book, which appeared in many major magazines including *People Magazine* and *The New York Post.*)

In his latest mad venture to avoid having to work in an ordinary 9–5 job, he’s also teamed up with three other comedians (Rick Gene, Wes Sample, and Justin Davis) to create, produce, and host a radio show called “Keeping It Weird,” currently (at least at the time of this writing) airing on 103.7 Free FM in San Diego.

By the time you read this, the author may be off pursuing something entirely different. Whatever he may be doing at the time, it should at least make for interesting stories to tell his grandchildren about one day.

Dedication

This book is dedicated to all the wonderful people I've met along the path of life, including . . .

Cassandra (my wife), Jordan (my son), and Bo, Scraps, Tasha, and Nuit (our cats).

Lily Carnie, the only person I know who can truly see both sides of the story.

All the friendly folks I've met while performing at the Riviera Comedy Club, located at the Riviera Hotel & Casino in Las Vegas: Steve Schirripa (who also appears on the HBO show, *The Sopranos*, which you can read about at `www.hbo.com/sopranos`), Don Learned, Bob Zany, Gerry Bednob, Bruce Clark, Darrell Joyce, and Kip Addotta. The next time you're visiting Las Vegas, drop by the Riviera and watch a comedy show. Then dump some money in a slot machine on the way out to ensure that the Riviera Hotel & Casino continues making enough money to keep its comedy club open.

Thanks also go to Roger Feeny at the Ann Arbor Comedy Showcase in Ann Arbor, Michigan; Russ Rivas at Laff's in Albuquerque, New Mexico; Pat Wilson at Mesquite, Nevada; and Joe Jarred at Primm and Pahrump, Nevada for running some of the friendliest comedy clubs around the country.

Final thanks must also go to Leo (the man, the myth, the legend) Fontaine, Chris (the Zooman) Clobber, Rick Gene, Wes Sample, Justin Davis, and Dante (who gets excited just to see his name in a book).

Author's Acknowledgments

If it wasn't for Bill Gladstone at Waterside Productions, I might still be staring off into space in a cubicle somewhere, working in a dead-end job, wondering what could have been.

Additional thanks go to Allen Wyatt for making sure that everything in this book is accurate, along with Rebecca Senninger and Virginia Sanders for making the process of writing a book always painless, easy, and often fun.

Final acknowledgements go to Cassandra (my wife) for putting up with multiple computers that (from her point of view) seem to spontaneously appear and disappear around the house at random. Each time a computer disappears, a more advanced model appears that promises more speed and hard drive space, but still never seems to have more speed or as much room as the previous computer model that it replaced.

Publisher's Acknowledgments

We're proud of this book; please send us your comments through our online registration form located at www.dummies.com/register/.

Some of the people who helped bring this book to market include the following:

Acquisitions, Editorial, and Media Development

Project Editor: Rebecca Senninger

(Previous Edition: Andrea C. Boucher)

Acquisitions Editor: Katie Feltman

Copy Editor: Virginia Sanders

Technical Editor: Allen Wyatt

Editorial Manager: Leah Cameron

Media Development Specialists: Angela Denny, Kate Jenkins, Steven Kudirka, Kit Malone

Media Development Coordinator: Laura Atkinson

Media Project Supervisor: Laura Moss

Media Development Manager: Laura VanWinkle

Editorial Assistant: Amanda Foxworth

Sr. Editorial Assistant: Cherie Case

Cartoons: Rich Tennant (www.the5thwave.com)

Composition Services

Project Coordinator: Michael Kruzil

Layout and Graphics: Claudia Bell, Carl Byers, Brooke Graczyk, Denny Hager, Joyce Haughey, Stephanie D. Jumper, Barbara Moore, Barry Offringa, Laura Pence, Alicia South

Proofreaders: Laura Albert, Techbooks

Indexer: Steve Rath

Anniversary Logo Design: Richard Pacifico

Publishing and Editorial for Technology Dummies

Richard Swadley, Vice President and Executive Group Publisher

Andy Cummings, Vice President and Publisher

Mary Bednarek, Executive Acquisitions Director

Mary C. Corder, Editorial Director

Publishing for Consumer Dummies

Diane Graves Steele, Vice President and Publisher

Joyce Pepple, Acquisitions Director

Composition Services

Gerry Fahey, Vice President of Production Services

Debbie Stailey, Director of Composition Services

Contents at a Glance

Table of Contents

Chapter 11: Dividing a Program into Subprograms 157

Chapter 12: Storing Stuff in Arrays 175

Chapter 13: Playing with Object-Oriented Programming 195

Introduction

Anyone can learn to program a computer. Computer programming doesn't require a high IQ or an innate proficiency in advanced mathematics. Computer programming just requires a desire to learn and the patience never to give up.

Programming is a skill like rock climbing, tap dancing, and pole vaulting. Some people are naturally better than others, but anyone can get better with regular practice. That's why so many kids become programming wizards at such an early age. These kids aren't necessarily brilliant; they're just willing to put in the time to learn a new skill, and they're not afraid of failing because they know that failure is nothing more than a part of learning.

If you ever dreamed about writing your own programs, rest assured that you can. Programming can be lots of fun, but it can also be frustrating, annoying, and time-consuming. That's why Wiley publishes this particular book — to help you discover how to program a computer with minimum inconvenience and maximum enjoyment.

Whether you want to pick up computer programming for fun, to start a new career, or to help make your current job easier, consider this book your personal guide through the sometimes scary — and initially intimidating — world of computer programming.

Although this book won't turn you into a programming wizard overnight, it can teach you enough about programming to help you understand how programming works, what the strengths and weaknesses of different programming languages are, and how you can get started writing programs all by yourself.

Who Should Buy This Book

You should buy this book if you want to learn how computer programming works without getting bogged down in the technical details of a particular programming language. When you understand how computer programming works, you'll better understand how to use a specific programming language with cryptic names like C++ or Java. But you should buy this book if you especially want to know any of the following:

- How computer programs work
- The common parts of every computer programming language
- How to write programs for multiple platforms such as Windows, Mac OS X, or Linux
- Whether to write your next computer program by using Visual Basic, C++, Perl, SmallTalk, C#, or some other programming language

Like any skill, you can learn programming only by practicing it. To help you get hands-on experience, the CD enclosed with this book includes trial versions of four language compilers so you can practice writing programs on any computer that runs Windows, Mac OS X, or Linux.

The three main languages you learn in this book are BASIC, C++, and a scripting language called Revolution.

BASIC is specially designed to introduce beginners to programming, so you can practice writing BASIC programs in two programming languages: Liberty BASIC and REALbasic.

Liberty BASIC represents the BASIC language in its purest and simplest form so you can understand the concepts of programming without getting lost in the technical details. When you understand how BASIC works, you can study REALbasic to see a version of the BASIC language that includes advanced programming features similar to more powerful languages such as C++.

This book also provides examples in C++, which is the most popular programming language in use today. If you want to write programs professionally, you must at least become familiar with the way C++ works.

You also learn a nontraditional programming language called Revolution, which uses English-like sentences to control your computer. Scripting languages like Revolution are designed to be easy to write and understand. They also provide commands capable of solving complicated problems more easily than traditional programming languages like BASIC or C++.

In addition, the Revolution programming language is based on AppleScript, which is a programming language used to automate a Mac OS X computer, so after you're familiar with Revolution, you also know most of the AppleScript programming language.

By learning BASIC, C++, and Revolution, you're exposed to three different programming languages, styles, and approaches to solving problems so you can better understand the advantages and limitations of any programming language. Then you can choose the best programming language for your needs.

How This Book Is Organized

To help you find what you need quickly, this book consists of five parts, and each part covers a certain topic about computer programming. Whenever you need help, just flip through the book, find the part that covers the topic you're looking for, and then keep the book at your side as you get back to work.

Part I: Programming a Computer

If computer programming seems a mysterious arcane science, relax. This part of the book demystifies all the common myths about computer programming, shows you exactly how computer programs work, and explains why programming isn't as difficult as many people think.

This part also shows you how programming has evolved, why so many different programming languages exist, and how programming follows easy-to-remember principles so you can start programming your own computer right away.

Part II: The Building Blocks of Programming

Although literally thousands of different programming languages are available for you to learn, every programming language tends to work in similar ways. So in this part of the book, you learn the basic building blocks of writing and creating a program regardless of the particular programming language you use.

To help you understand the building blocks of programming, each chapter provides plenty of examples in different programming languages so you can see how they accomplish the same task. You can also try out the examples on your own computer.

Part III: Advanced Programming Topics

After you master the basics of writing a program, you need to worry about making your program work efficiently, eliminating problems, and designing a user interface so other people will know how to use it. In this part of the book, you learn how programmers fine-tune their software (and what the consequences might be if they don't).

Part IV: Internet Programming

The Internet is fast becoming an integral part of the computer world, so this part of the book introduces you to the basics of various Internet languages, including HTML (which designs the appearance of Web pages), JavaScript, and Java.

In this part, you also see how other people create cool Web pages that look good and can display forms and respond to users. You can use this information to create Web sites that interact with users.

Part V: The Part of Tens

To help gently guide you toward writing your own programs for money, this part of the book provides information about programming jobs you might want to pursue and how to find more tools and source code to help you learn more about programming all by yourself.

How to Use This Book

This book is meant to show you the basics of computer programming without bogging you down with the technical details of any particular programming language. Typed code often looks like chicken scratches or the random characters that a monkey might type if left alone with a keyboard. So you can use this book as a tutorial (to show you how programming works) and as a reference (to help refresh your memory for understanding different programming techniques).

Ideally, you want to use this book along with your computer. Read some of the book and then try what you just read on your computer so that you can see with your own eyes how programming works.

Foolish assumptions

To get the most out of this book, you need access to a computer (because trying to understand computer programming without a computer is like trying to learn to drive without a car). To take full advantage of this book, you need a computer running Microsoft Windows, Mac OS X, or Linux.

Icons used in this book

Icons highlight useful tips, important information to remember, or technical explanations that can amuse you for a moment before you forget all about them. Keep an eye open for the following icons throughout the book:

TIP

This icon highlights useful information that can save you time (as long as you remember it, of course).

REMEMBER

This icon reminds you to do something or emphasizes an important point that you don't want to forget.

WARNING!

Watch out! This icon tells you how to avoid potential headaches and trouble.

ON THE CD

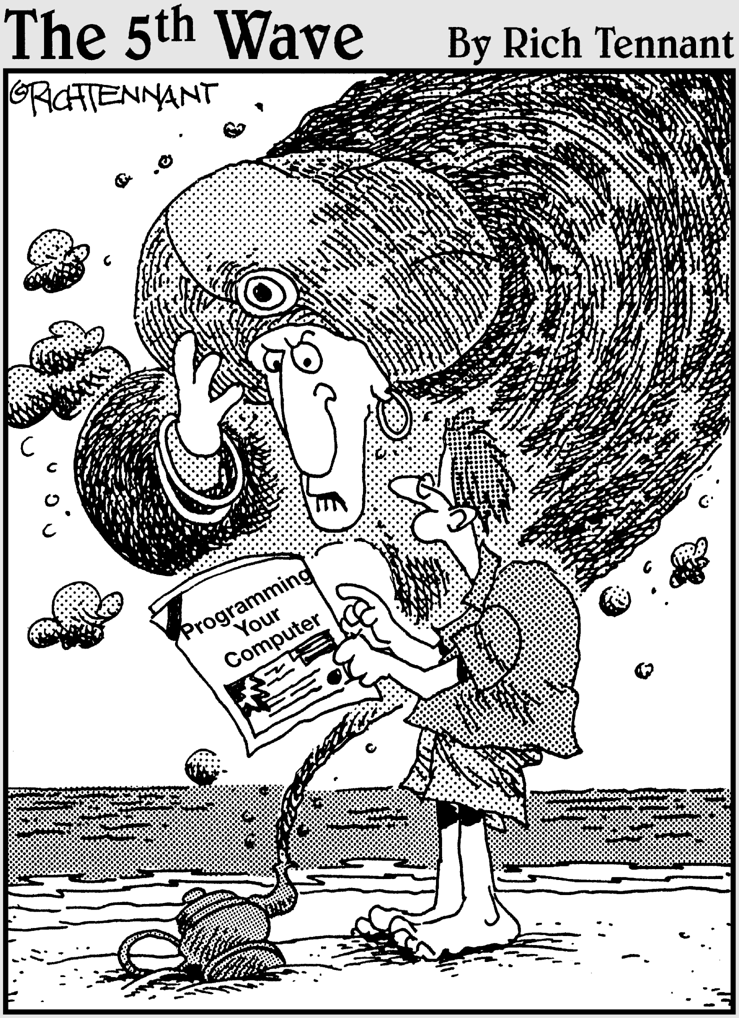
This icon identifies the name of a particular file on the CD that contains a sample program printed in the book. By loading the sample program off the CD, you don't have to type the program yourself.

TECHNICAL STUFF

This icon highlights information that's nice to know but which you can safely ignore if you choose. (If you want to become a real programmer, however, you need to cram your brain with as much technical information as possible so that you can fit in with the rest of the programmers in the world.)

Part I

Programming a Computer



“Can’t I just give you riches or something?

In this part . . .

Figuring out how to program a computer may seem intimidating, so this part of the book gently guides you through the wonderful world of computer programming. First, you see exactly what programs do and how professionals write programs.

Next, you discover why so many different programming languages exist and why some are more popular than others. You get to know the different tools that programmers use to create, edit, and distribute a program from start to finish.

Finally, this part shows you what to consider if you decide to write a program. You see the pros and cons of using different programming languages. You also find out how people can write programs even if they possess very little programming experience.

By the time that you finish this part of the book, you'll have a better idea of how to write a program, what steps to follow, and how to convert your idea for a program into an actual working product that you can sell or give away for others to use. Who knows? With a little bit of imagination and a lot of persistence, you may create the next program that makes so much money that you can start your own software company and make a million bucks.

Chapter 1

Learning Computer Programming for the First Time

In This Chapter

- Learning computer programming
- Understanding how a computer program works
- Knowing how to program a computer

Despite what you may have heard, programming a computer isn't difficult. Computer programming is a skill that anyone can pick up, given enough practice, patience, and caffeinated beverages.

Although computers may seem like tremendously complex electronic beasts, relax. Few people know how internal-combustion engines work, yet people can still figure out how to drive a car. Similarly, anyone can pick up programming skills without worrying (too much) about the specific details that make a computer work.

Why Learn Computer Programming?

The first question that you (or your friends, co-workers, and relatives) may ask is, "Why bother learning to program a computer?" The answer depends on your ultimate goals, but the following list offers some common answers to consider:

- **For fun:** People learn skiing, dancing, gardening, scuba diving, and painting because they enjoy the experience. They may never become professionals or experts in their chosen hobbies, but they enjoy fiddling around nevertheless. Similarly, programming a computer can prove fun because you might design a simple program that displays your boss's ugly face on the computer. More complex programs may make you a million dollars so that you never again need to work for a boss with an ugly face. Figure 1-1 shows a program known as *Comic Life,* which can turn any photograph into a comic book. This program was written in a programming language called Objective-C on a Macintosh.

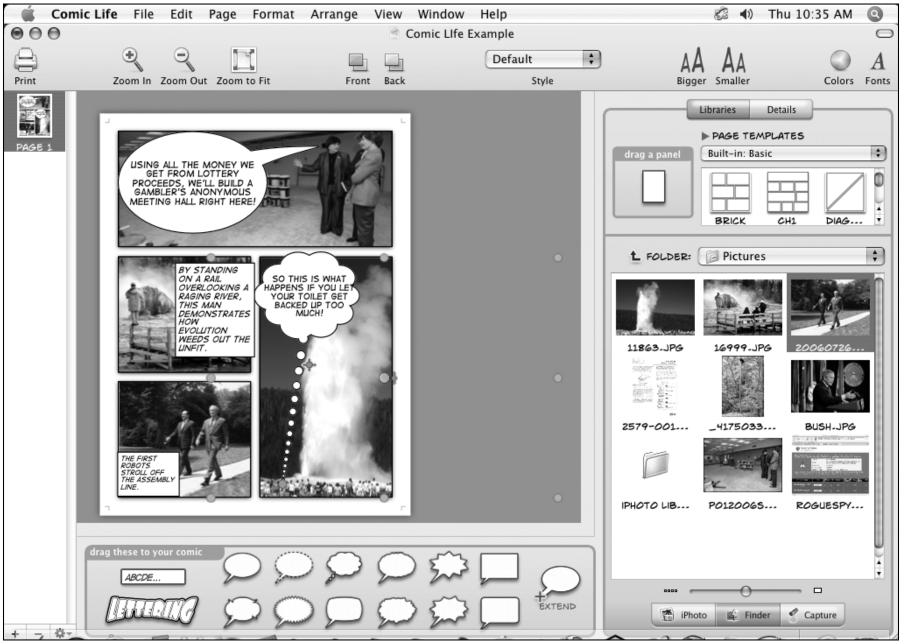


Figure 1-1: Comic Life is a program that can turn any picture into a comic book cartoon.

- **To fill a need:** Many people learn programming with no intention of becoming full-time, professional programmers. They just want a program that solves a particular problem, but they can't find a program that does it, so they write the program themselves. A man once needed a program to help him file his taxes, for example, but he couldn't find one, so he taught himself programming and wound up creating TurboTax, one of the most popular tax-preparation programs in the country. Similarly, a freelance writer got frustrated with trying to write a novel with an ordinary word processor, so he created a word processor, specially designed to organize a story, called Z-Write, as shown in Figure 1-2. Z-Write was written by using a program called REALbasic. Whatever your interests, you can write a program to solve a specific problem that others may find useful as well.
- **For a new or second career:** With computers taking over the world, you're never unemployed for long if you know how to program a computer. Companies are always looking to create new programs, but you also find a growing market for programmers who can maintain and modify the millions of existing programs that do everything from storing hotel reservations to transferring bank deposits electronically. If you know how to program a computer, you're in a much better position to