

# PROGRAMMING LOGIC FOR BUSINESS

LAURA SARET

FOURTH EDITION

FOURTH EDITION

# Programming Logic for Business

**Laura Saret, Ed.D**  
Oakton Community College



Boston Burr Ridge, IL Dubuque, IA Madison, WI New York San Francisco St. Louis  
Bangkok Bogotá Caracas Kuala Lumpur Lisbon London Madrid Mexico City  
Milan Montreal New Delhi Santiago Seoul Singapore Sydney Taipei Toronto

# McGraw-Hill Higher Education

A Division of The McGraw-Hill Companies

## PROGRAMMING LOGIC FOR BUSINESS

Published by McGraw-Hill/Irwin, an imprint of The McGraw-Hill Companies, Inc. 1221 Avenue of the Americas, New York, NY, 10020. Copyright © 2001, 1992, 1987, 1984, by the McGraw-Hill Companies, Inc. All rights reserved. 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 consent of The McGraw-Hill Companies, Inc., including, but not limited to, in any network or other electronic storage or transmission, or broadcast for distance learning. Some ancillaries, including electronic and print components, may not be available to customers outside the United States.

This book is printed on acid-free paper.

2 3 4 5 6 7 8 9 0 QPD/QPD 0 9 8 7 6 5 4 3

ISBN 0-07-366096-5

Vice president/Editor in chief: *Robert Zwettler*  
Publisher: *David Brake*  
Executive Editor: *Linda Schreiber*  
Senior sponsoring editor: *Jodi Mcpherson*  
Developmental editor: *Melissa Forte (supplements)/Craig S. Leonard*  
Marketing manager: *Nicole Young*  
Project manager: *Christina Thornton-Villagomez*  
Production supervisor: *Debra Sylvester*  
Coordinator freelance design: *Mary Christianson*  
Media technology producer: *David Barrick*  
Cover designer: *Anthony O'Neill*  
Cover and opener image: © *PhotoDisc*  
Compositor: *Shepherd Incorporated*  
Typeface: *10/12 Palatino*  
Printer: *Quebecor Printing Book Group/Dubuque*

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

Saret, Laura.  
Programming with business logic/Laura Saret.—4th ed.  
p. cm.  
Rev. ed. of: Programming logic for business. 3rd ed. c1992.  
Includes index.  
ISBN 0-07-366096-5 (alk. paper)  
1. Business—Data processing. 2. Computer programming. 3. Structured programming.  
I. Saret, Laura. Programming logic for business. II. Title.

HF5548.2 .S238 2001  
650'.0285'51—dc21

00-056611

# **Programming Logic for Business**

## **Dedication—**

This book is dedicated to my family.

To my husband, Larry, who has been my best friend for more than 27 years.

To my children, Marla and Jeff, who have grown and changed as rapidly as the world of programming has changed. When I typed the first edition on an electric typewriter, they were just babies, and object-oriented programming didn't even exist. Now, they are young adults, we all use computers for our writing, and objects are everywhere. One thing hasn't changed—they still have an endless supply of hugs, and they know when I need them most!

To my parents, Harvey and Marilyn Kaplan, who helped make me what I am today.

To my sister, Sheri, who gives so much to everyone and takes so little in return.

To my "little" brothers, Steve and Michael, who are still trying to get me into trouble!

# Preface

Most books on computer programming languages do not teach students how to program. Instead, these books concentrate on the rules of a particular programming language, with a minimal amount of text devoted to problem solving. But learning a programming language without first learning programming logic is like learning to spell and use punctuation in English without first learning how to express your thoughts so that you are clearly understood. This book concentrates on problem solving or programming logic, with little emphasis on specific programming languages. Hierarchy charts, pseudocode, flowcharts, and data dictionaries are developed throughout the text as tools for problem solving.

*Programming Logic for Business*, 4th edition, focuses on programming logic for business applications. The book was initially developed from lecture notes and handouts that I used to teach a course on programming logic and techniques at Oakton Community College. Since the first edition was published, the world of computer programming has changed considerably. These changes were reflected in the second, third, and now, the fourth edition. Each edition of the book has been improved as a result of feedback from my colleagues across the country and the excellent suggestions of my own students. This edition is no exception.

Among the changes in this edition are:

- Each chapter now links end-of-chapter exercises and activities to particular sections of each chapter. Students are strongly encouraged to do the exercises as they are reading and studying each section rather than waiting until the end of the chapter. These exercises will help students monitor their own learning and review important terminology and concepts.
- Although the text is still language independent, greater emphasis has been placed on visual, event-driven, and object-oriented concepts. These explanations will help students as they work with visual, event-driven, and object-oriented programming languages.
- The layout of the text is improved. Pseudocode and flowchart solutions are shown on the same page. This improved organization will allow students to use either pseudocode or flowcharts without constant page “flipping.”
- Although the content has not been reduced, this edition has 9, instead of 11 chapters. Topics have been reorganized to assist students who are learning programming logic in the same semester or term as they are learning a programming language.



- Chapter 1 still helps students understand basic terminology and the programming process. In addition, it explains the various programming languages and how procedure-oriented, visual, object-oriented, and Internet languages differ.
- Chapter 2 introduces program flowcharts as a tool for developing logic. In addition, greater emphasis is placed on arithmetic and relational operations to better prepare students to write these types of instructions.
- Even though the programming world is primarily object-oriented, methods and procedures are still developed using structured techniques. Students must be well grounded in these techniques, one of the focuses of Chapter 3. Chapter 3 discusses the transition from modular to structured to object-oriented programming. Students learn about objects and classes. In addition, pseudocode, data dictionaries, and hierarchy charts are used. Top-test, bottom-test, and counter-controlled loops are covered in depth. Local and global variables are also explained in this chapter.
- Chapter 4 now concentrates on report output and input validation. Both paper and screen reports are covered.
- The emphasis in Chapter 5 is on decision logic using the logical operators AND, OR, NOT, and exclusive OR. Nonreport output is also covered in this chapter.
- Chapter 6 focuses on arrays (tables). Previous editions used the term table to refer to arrays. Most programming languages use the term array, so it was decided to use that term in this text. The chapter covers array initialization and search, compile-time and execution-time arrays, positionally-organized and argument-organized arrays, as well as a discussion on how various programming languages differ when working with arrays.
- Chapter 7, which covers sorting, is optional. I have found that working with sort algorithms helps students better understand arrays. The interchange, bubble, and Shell sort algorithms are covered. In addition, indexing is explained.
- Chapter 8 covers control breaks and totals with single and multiple levels.
- Chapter 9 covers sequential and nonsequential file and database processing.

The approach of this book is to teach via examples. All the rules for programming can be listed, but unless students can see how to apply the rules, they won't learn. More than 40 business examples are used in this book to illustrate various topics. Unlike other logic books, most examples are completely solved, so students can see the entire solution process.

The book is organized by topic with sufficient material for a one-semester college course in programming logic. If students have had a strong introductory course, they can skip Chapter 1 or use it for review. If this course is offered concurrently with an introductory course, then the background and vocabulary in Chapter 1 is critical. Chapters 2 through 6 cover basics and should be covered in order. Although important to a student's understanding of programming logic, Chapters 7, 8, and 9 are optional. Chapter 7 requires prior coverage of Chapter 6. Chapters 8 and 9 can be covered any time after Chapter 5.

In addition to a one-semester college course in computer programming, the book can be used:

- As a supplement to a programming language course. Many programming students and instructors have indicated that the examples in this book relate well to the problems assigned in both beginning and advanced programming classes.
- In a high-school course in the fundamentals of computer programming.
- In an industry course to train computer programmers.
- In short, introductory computer and computer-programming courses for business and management personnel.
- By programmers who want to improve their programming skills using object-oriented and structured techniques.
- By people who are interested in writing their own computer programs.

The instructor's manual to accompany the text includes chapter outlines and teaching hints for each chapter as well as the solutions to all exercises.

This book assumes no previous knowledge of information processing. It is expected that the reader can use simple arithmetic and algebraic tools, but no previous training in advanced mathematics is necessary.

A great deal of thought, discussion with students and colleagues, and some aggravation have been invested in this book. Students have always been the best barometer of my teaching methods, and their questions and comments have proved invaluable. As a result, this edition is more polished and has better explanations and organization than the previous three editions resulting in increased understanding for the reader.



# To the Student

You will find this book useful whether you are a programming student, an information technology professional wishing to update your skills by learning object-oriented and structured techniques, a business professional who uses computer services, or an individual merely curious about how to tell the computer to do what you want it to.

The intent of this book is to teach you how to program logically without focusing on a particular programming language. You will find this book useful in *any* programming language class you take because coding a program merely involves translating programming logic into a language that the computer can understand.

Learning the rules of a specific programming language is not hard. Learning how to program—that is, developing the logic necessary to solve a problem—is challenging for most people. Even if you have used computers for many years, this may be your first experience in learning to “think like a computer.” You must unlearn the complicated thought processes you have become used to in dealing with everyday life and learn to tell the computer, in simplified terms, exactly what to do—step by step.

This book assumes no previous knowledge of computers or programming. You should be able to use simple arithmetic tools but no training in advanced mathematics is required. If you have had an introductory course or previous experience with computers, many of the concepts in the beginning of the book will be a review for you. Regardless of your experience level, however, you will need to devote considerable time practicing and doing the exercises at the end of each chapter.

I hope you enjoy learning how to program. Whether you end up as a computer programmer; an information technology professional; a manager; or a user of the Internet, games, word processors, spreadsheets, databases, and other prewritten programs, you will appreciate the time, effort, and creativity involved in generating the logic of a computer program.

# Information Technology at McGraw-Hill/Irwin

At McGraw-Hill Higher Education, we publish instructional materials targeted at the higher-education market. In an effort to expand the tools of higher learning, we publish texts, lab manuals, study guides, testing materials, software, and multimedia products.

At McGraw-Hill/Irwin (a division of McGraw-Hill Higher Education), we realize that technology has created and will continue to create new mediums for professors and students to use in managing resources and communicating information to one another. We strive to provide the most flexible and complete teaching and learning tools available, as well as offer solutions to the changing world of teaching and learning.



McGraw-Hill/Irwin is dedicated to providing the tools for today's instructors and students to successfully navigate the world of Information Technology.

## **InformationTechnology**

- **Seminar series**—McGraw-Hill/Irwin's Technology Connection seminar series offered across the country every year demonstrates the latest technology products and encourages collaboration among teaching professionals.
- **McGraw-Hill/Osborne**—This division of The McGraw-Hill Companies is known for its best-selling Internet titles, *Harley Hahn's Internet & Web Yellow Pages* and the *Internet Complete Reference*. For more information, visit Osborne at [www.osborne.com](http://www.osborne.com).
- **Digital solutions**—McGraw-Hill/Irwin is committed to publishing digital solutions. Taking your course online doesn't have to be a solitary venture, nor does it have to be a difficult one. We offer several solutions that will allow you to enjoy all the benefits of having your course material online. For more information, visit [www.mhhe.com/solutions/index.mhtml](http://www.mhhe.com/solutions/index.mhtml).
- **Packaging options**—For more information about our discount options, contact your McGraw-Hill/Irwin Sales representative at 1-800-338-3987 or visit our website at [www.mhhe.com/it](http://www.mhhe.com/it).

# Resources for Instructors

We understand that, in today's teaching environment, offering a textbook alone is not sufficient to meet the needs of the many instructors who use our books. To teach effectively, instructors must have a full complement of supplemental resources to assist them in every facet of teaching, from preparing for class to conducting a lecture to assessing students' comprehension. *Programming Logic for Business* offers a complete, fully integrated supplements package and website, as described below.

## Instructor's Manual

Written by the author of the text, the Instructor's Manual includes chapter outlines and teaching tips for each chapter, as well as solutions to all end-of-chapter exercises. The manual also contains a helpful introduction that explains the features, benefits, and suggested use of the Instructor's Manual.

## Digital Solutions to Help You Manage Your Course

- **PageOut.** PageOut is our Course Website Development Center that offers a syllabus page, URL, McGraw-Hill Online Learning Center content, online exercises and quizzes, gradebook, discussion board, and an area for student Web pages.

Available for free with any McGraw-Hill/Irwin product, PageOut requires no prior knowledge of HTML, no long hours of coding, and a way for course coordinators and professors to provide a full-course website. PageOut offers a series of templates—simply fill them with your course information and click on one of 16 designs. The process takes under an hour and leaves you with a **professionally designed website**. We'll even get you started with sample websites, or enter your syllabus for you! PageOut is so straightforward and intuitive, it's little wonder why over 12,000 college professors are using it. For more information, visit the PageOut website at [www.pageout.net](http://www.pageout.net).

- **Websites.** The Website that accompanies *Programming Logic for Business* is accessible through our Information Technology Supersite at [www.mhhe.com/it](http://www.mhhe.com/it) or at the book site [www.mhhe.com/it/saret](http://www.mhhe.com/it/saret). This site provides additional learning and instructional tools to accompany the text, and supplements for professors to lecture and teach from.

## Technology Connection Seminar Series

McGraw-Hill/Irwin's Technology Connection seminar series offered across the country every year demonstrates the latest technology products and en-

courages collaboration among teaching professionals. Contact your McGraw-Hill/Irwin Sales representative for more details.

## **In Addition, McGraw-Hill/Irwin Offers the Following Products to Help You Create Your Total Solution:**

### **MS Office 2000 Applications Texts and CDs**

Available separately, or packaged with *Programming Logic for Business*, McGraw-Hill offers three applications series: The O'Leary Series, The Advantage Series, or The Interactive Computing Series. Each series features its own unique approach to teaching MS Office in order to meet the needs of a variety of students and course goals.

- *The O'Leary Series* features a project-based, step-by-step walkthrough of applications.
- *The Advantage Series* features a case-based, what, why, and how approach to learning applications to enhance critical thinking skills.
- *The Interactive Computing Series* features a visual, two-page spread to provide a more skills-based approach to learning applications.

Each series offers Microsoft Office User Specialist (MOUS) approved courseware to signify that it has been independently reviewed and approved in complying with the standards of content coverage related to the Microsoft Exams and Certification Program. For more information on Microsoft's MOUS certification program, please visit Microsoft's website at [www.microsoft.com/office/traincert/](http://www.microsoft.com/office/traincert/).

Also available for applications are the *Interactive Computing Series* Computer-Based Training CD-ROM tutorials. These CD-ROMs offer a visual, interactive way to develop and apply software skills. The CD-ROM features a unique "skills-concepts-steps" approach, and includes interactive exercises and performance-based assessment. These CD-ROMs are simulated, so there is no need for the actual software package on the computer.

### **Skills Assessment**

McGraw-Hill/Irwin offers two innovative systems to meet your skills assessment needs. These two products are available for use with any of our applications manual series.

- **ATLAS** (Active Technology Learning Assessment System) is one option to consider for an application skills assessment tool from McGraw-Hill. ATLAS allows students to perform tasks while working live within the Microsoft applications environment. ATLAS provides flexibility for you in your course by offering:
  - Pretesting options
  - Posttesting options
  - Course placement testing
  - Diagnostic capabilities to reinforce skills
  - Proficiency testing to measure skills
  - ATLAS is Web-enabled, customizable, and is available for Microsoft Office 2000

▪ **SimNet** (Simulated Network Assessment Product) is another option for a skills assessment tool that permits you to test students' software skills in a simulated environment. SimNet is available for Microsoft Office 97 (deliverable via a network) and Microsoft Office 2000 (deliverable via a network and the Web). SimNet provides flexibility for you in your course by offering:

- Pretesting options
- Posttesting options
- Course placement testing
- Diagnostic capabilities to reinforce skills
- Proficiency testing to measure skills

For more information on either skills assessment software, please contact your local sales representative, or visit us at [www.mhhe.com/it](http://www.mhhe.com/it).

## Online Courses Available

Online Learning Centers (OLCs) are your perfect solutions for Internet-based content. Simply put, these Centers are "digital cartridges" that contain a book's pedagogy and supplements. As students read the book, they can go online and take self-grading quizzes or work through interactive exercises. These also provide students appropriate access to lecture materials and other key supplements.

Online Learning Centers can be delivered through any of these platforms:

- McGraw-Hill Learning Architecture (TopClass)
- Blackboard.com
- ecollege.com (formally Real Education)
- WebCT (a product of Universal Learning Technology)

McGraw-Hill has partnerships with **WebCT** and **Blackboard** to make it even easier to take your course online. Now you can have McGraw-Hill content delivered through the leading Internet-based learning tool for higher education.

At McGraw-Hill, we have the following service agreements with **WebCT** and **Blackboard**:

- Instructor Advantage

Instructor Advantage is a special level of service McGraw-Hill offers in conjunction with WebCT designed to help you get up and running with your new course. A **team of specialists** will be immediately available to ensure everything **runs smoothly** through the **life** of your adoption.

- Instructor Advantage Plus

Qualified McGraw-Hill adopters will be eligible for an even higher level of service. A **certified WebCT or Blackboard specialist** will provide a full day of **on-site training** for you and your staff. You will then have **unlimited** e-mail and phone support through the life of your adoption. Please contact your local McGraw-Hill representative for more details.

## Interactive Companion CD-ROM

This free student CD-ROM, designed for use in class, in the lab, or at home by students and professors alike, includes a collection of interactive tutorial labs on some of the most popular and difficult concepts topics to illustrate in information technology. By combining video, interactive exercises, animation, additional content, and actual “lab” tutorials, we expand the reach and scope of the textbook. The lab titles are listed below:

### Available Now:

- Computer Anatomy
- Binary Numbers
- Storage
- Email
- Learning to Program I
- Learning to Program II
- Network Communications
- Intro to Multimedia
- Databases
- Workplace Issues (ergonomics/privacy/security)

### Coming Soon:

- User Interfaces
- Photo Editing
- Word Processing
- Spreadsheets
- Directories, Folders, Files
- Using Files
- CPU Simulator
- Troubleshooting
- Web Pages and HTML
- SQL Queries

## PowerWeb for I.T.

PowerWeb is an exciting new online product. A nominally priced token grants students access through our website to a wealth of resources—all corresponding to computer literacy. Features include an interactive glossary; current events with quizzing, assessment, and measurement options; Web survey; links to related text content; and WWW searching capability via Northern Light, an academic search engine. Visit the PowerWeb site at **[www.dushkin.com/powerweb](http://www.dushkin.com/powerweb)**.

# Contents

---

<b>1</b>	<b>Introduction to Computer Programming</b>	<b>1</b>
1.1	Introduction: The Computer Revolution	2
1.2	The Mystery of Computers	3
1.3	How Does Data Become Information?	3
1.4	Why Do We Organize Data?	4
1.5	How Are Data Organized?	5
1.6	What Is Hardware?	7
1.7	What Is Software?	9
1.8	What Are the Various Programming Languages?	10
1.9	How Do Procedure-Oriented, Visual, Object-Oriented, and Internet Languages Differ?	12
1.10	What Is the Programming Process?	17
	Summary	19
	Key Terms and Concepts	20
	Questions and Exercises	21

---

<b>2</b>	<b>Getting Started with Program Flowcharts, Arithmetic, and Relational Operations</b>	<b>27</b>
2.1	What Is a Program Flowchart, and Why Do Programmers Use Them to Develop Program Logic?	28
2.2	What Are the Flowchart Symbols?	29
2.3	What Are the Rules for Drawing Flowcharts?	32
2.4	Example: Open, Read, Move, Write, Stop, and Close Instructions; Input Areas, Output Areas, and Work Areas	34
2.5	Example: Looping and Checking for End of File	37
2.6	How Are Addresses and Names Used in Primary Storage?	39
2.7	Example: Naming and Defining Areas in Storage	40
2.8	Example: Assignment Instructions, Counters, Work Areas, and Connectors	42
2.9	Example: Counters as Totals	44
2.10	What Are Arithmetic Operations and How Are They Used?	46
2.11	Example: Arithmetic Operations	48
2.12	What Are Relational Operations and How Are They Used?	50
2.13	Example: Relational Operations	51



- 2.14 What Is the Difference Between Branching and Using Subroutines? **53**
- 2.15 Example: Screen Output, Compare Areas, Subroutines, Literals (Constants) versus Field Names (Variables) **54**
- Summary **61**
- Key Terms and Concepts **63**
- Questions and Exercises **64**

## **3 Structured Techniques and Object-Oriented Programming**

**79**

- 3.1 What Are Straight-Line Flowcharts and What's Wrong with Them? **80**
- 3.2 Why Did We Progress from Modular to Structured Programming? **81**
- 3.3 What Are the Rules for Writing a Structured Program? **82**
- 3.4 Example: Using Structures **87**
- 3.5 If Structured Programming Was So Great, Why Do Programmers Now Use Object-Oriented Programming? **91**
- 3.6 How Does Object-Oriented Programming Look at a Problem? **92**
- 3.7 What Are Objects and Classes? **93**
- 3.8 How Do Objects Interact with Each Other? **96**
- 3.9 Example: Creating a Class **96**
- 3.10 Why Do We Need to Focus Our Energy on Structured Programming Concepts? **97**
- 3.11 Example: Using Process and Error Subroutines **98**
- 3.12 What Is Pseudocode? **98**
- 3.13 What Are Hierarchy (Structure) Charts? **100**
- 3.14 Example: Using Local and Global Variables **101**
- 3.15 What Are Housekeeping and Termination Functions? **105**
- 3.16 Example: Initialization and Termination Functions; Local and Global Variables; Total at End of Report **106**
- 3.17 Example: Counter-Controlled Loops **112**
- Summary **113**
- Key Terms and Concepts **117**
- Questions and Exercises **118**

## **4 Report Output and Validating Input**

**131**

- 4.1 What Types of Reports Can Be Created? **132**
- 4.2 What Types of Lines Can Be Included on a Report? **133**
- 4.3 How Are Lines Spaced on a Page or Screen? **135**
- 4.4 Example: Heading Lines on Printed Reports, No Data Input from Outside the Program, and Using External Subroutines **137**
- 4.5 Example: Screen Output with Heading and Footing Lines **143**
- 4.6 How Do We Include the Date on a Report? **148**
- 4.7 Example: Exception Reports, Using an External Subroutine to Obtain the Current Date, Double Spacing Detail Lines, and Printing the Page Number in a Footing Line **149**
- 4.8 How Does Garbage Input Generate Output Errors? **154**

- 4.9 What Types of Data Validation Can Be Done? **155**
- 4.10 What Are the Differences between Batch and On-Line Data Entry and Validation? **162**
- 4.11 Example: Input Validation for a Batch Program **163**
- 4.12 Example: Using an Interactive Program to Create and Validate an Input File **170**
- 4.13 What Is a Menu-Driven Program? **178**
- 4.14 How Do Event-Driven Programming Languages Enter and Validate Data? **179**
- Summary **181**
- Key Terms and Concepts **183**
- Questions and Exercises **183**

## 5

### More on Decision Logic

200

- 5.1 What Are Control Fields and Codes and How Are They Used? **201**
- 5.2 Example: Control Fields **202**
- 5.3 What Are Decision Tables and How Are They Used? **207**
- 5.4 Example: Multiple Control Codes and Nonreport Output **209**
- 5.5 What Does a Search (Extract) Program Do? **213**
- 5.6 Example: Search (Extract) Program with Selection Based on One Criterion **213**
- 5.7 How Are Selection Criteria Specified by the User? **218**
- 5.8 How Do You Use the Logical Operators AND, OR, NOT, and Exclusive OR (XOR)? **218**
- 5.9 How Can You Improve Program Efficiency When Using Logical Operators? **225**
- Summary **232**
- Key Terms and Concepts **233**
- Questions and Exercises **234**

## 6

### Using Arrays (Tables)

242

- 6.1 What Is an Array (Table)? **243**
- 6.2 Example: Array versus Nonarray Processing (Positionally Organized Arrays) **245**
- 6.3 How Do We Put Values into Arrays? **251**
- 6.4 Example: Arrays in Validation Programs (Compile-Time Arrays) **252**
- 6.5 Example: Arrays in Validation Programs (Execution-Time Arrays) **258**
- 6.6 What Is the Difference Between a Sequential Search and a Binary Search? **261**
- 6.7 Example: Binary Search **262**
- 6.8 How Do We Work with Multidimensional Arrays? **265**
- 6.9 Example: Finding Values in Two-Dimensional, Argument-Organized Arrays **267**
- 6.10 Example: Values as Part of a Range **275**
- 6.11 How Do Various Programming Languages Differ in How They Work with Arrays? **276**
- Summary **278**
- Key Terms and Concepts **279**
- Questions and Exercises **279**