

Instructor's Summary for *Murach's C# (7th Edition)*

The instructor's materials for *Murach's C# (7th Edition)* will help any college instructor or corporate trainer run an effective course based on the book. This summary introduces you to these materials and helps you get started using them.

At the least, we recommend that you read the topics under *What's included in the instructor's materials* because they not only describe the instructor resources but also our underlying instructional philosophy. Then, the section entitled *How to get started* guides you in getting the materials on your system and gives you charts that summarize the components at a glance.

But first, some thoughts about the modular structure of this book that you should be aware of. This structure is important because it gives you instructional options that you don't have with other books.

About the modular structure of the book	2
Section 1: An introduction to Visual Studio	2
Section 2: The C# language essentials.....	2
Section 3: Object-oriented programming	2
Section 4: Basic skills for working with data	2
Section 5: Database programming.....	3
What's included in the student download	3
Book applications	3
Exercise starts.....	3
Exercise solutions.....	3
What's included in the instructor's materials	4
Book applications, exercises, and solutions.....	4
Objectives	4
Test banks.....	5
Extra exercises and solutions.....	5
Projects and project solutions	6
PowerPoint slides	6
How to get started	7
How to use the zip file.....	7
The student directories and files that get installed.....	8
The instructor directories and files that get installed	8
Any comments?.....	9



Mike Murach & Associates, Inc.

1-800-221-5528 • (559) 440-9071 • Fax: (559) 440-0963

murachbooks@murach.com • www.MurachForInstructors.com

Copyright © 2021 Mike Murach & Assoc. Inc. All rights reserved.

About the modular structure of the book

The content in this book is divided into 5 sections, and after your students complete the first 12 chapters, you can continue with any of the other sections. In other words, the rest of section 3 as well as sections 4 and 5 are written as independent modules that require only the first 12 chapters as a prerequisite. That's what we mean by *modularity*; it lets you choose the subjects that you want to teach, as well as the sequence in which you teach them.

Beyond that, you have some options as to which chapters you assign from each section of the book. Those options are described in the topics that follow. Once you understand your options, you can select the chapters and teaching sequence that's best for your course.

Section 1: An introduction to Visual Studio

The three chapters of section 1 teach your students how to use the Visual Studio IDE to develop Windows Forms applications. In particular, these chapters focus on the skills for designing forms and entering code. These are essential skills your students need to get the most out of Visual Studio.

Section 2: The C# language essentials

The eight chapters of section 2 present a professional subset of the C# language that includes all the skills for developing substantial applications. That includes working with numbers, strings, and dates; coding control structures, methods, and event handlers; and working with arrays and collections. It also includes the best techniques for handling exceptions, validating data, and debugging applications. Although the chapters in this section are best taught in sequence, you can assign the chapter on debugging any time you feel that your students can benefit from these skills.

Section 3: Object-oriented programming

The five chapters of section 3 present a professional set of skills for creating and using classes. Chapter 12 presents the basic skills every student should know. Because many of the other chapters present advanced skills that your students will rarely need, you may want to be selective about which of these chapters you assign. The modularity of these chapters makes that easy to do.

Section 4: Basic skills for working with data

The two chapters of section 4 present some basic skills for developing applications that work with data. That includes working with text and binary files as well as using LINQ to query any data structure. You can decide which of these chapters you want to include in your course, and you can assign them in the sequence that works best for you. The exception is that you must teach the chapter on LINQ before you teach the chapter on the Entity Framework that's in section 5, since you use an implementation of LINQ called LINQ to Entities when you work with the Entity Framework.

Section 5: Database programming

The four chapters of section 5 present the essential skills for developing applications that work with databases. This includes using Entity Framework (EF) Core to map the tables of a database to the objects of an application, using ADO.NET to write custom code that works with databases, and using a DataGridView object to work with the data from a database table. These chapters can be assigned any time after completing chapters 1 through 12 plus chapter 18 (on LINQ), but they should be assigned in sequence.

What's included in the student download

To help your students get the most from our book, our retail website (www.murach.com) lets them download (1) the applications presented in the book, (2) the starting points for the exercises presented at the end of each chapter, and (3) the solutions to the exercises. Appendix A in the book shows them how to download and set up these files on their own systems.

Book applications

All of the applications presented in this book are stored in a directory named *Book Applications* when they're downloaded. Once your students have set up the book applications on their own systems, they can run them to see how they work. They can review all of the code in any application when the book doesn't present it all. And they can copy and paste code from the book applications into their own applications.

Exercise starts

The exercises at the end of each chapter in the book are designed to help your students apply what they've just learned. To give your students a maximum of practice in a minimum of time, many of the exercises start from partial applications. These starting points are provided in the download for this book and are stored in a directory named *Exercise Starts*.

Exercise solutions

To help students get over any learning obstacles when they're working on their own, the download also provides the solutions to the exercises in a top-level directory named *Exercise Solutions*. That way, students can check the solutions to see how something is done whenever they're wasting time on what is likely to be a trivial coding mistake. We think that providing the solutions is the right approach didactically because it helps students learn faster and better.

We realize, however, that this makes it hard for you to use the book exercises to test or grade your students. That's why the instructor's materials include a second set of exercises as well as a set of projects that can be used for testing. For these extra exercises and projects, the solutions are available only to instructors.

What's included in the instructor's materials

The instructor's materials for this book are designed to save you time in preparing and running an effective course based on the text so that your students gain the C# skills they'll need on the job. So besides the materials in the student download, we provide instructional objectives, test banks, PowerPoint slides, a second set of chapter exercises, and projects that your students must develop on their own from scratch.

Book applications, exercises, and solutions

These are the same materials that your students can download from our retail website. We've included them in the instructor's materials so you can demonstrate and review the book applications and exercise solutions in class, without having to download them separately.

To get the exercise solutions to run correctly on your system, we've included a .bat file that copies the Exercise Starts directory to another file path, where the solutions look for the files. There are instructions for this on page 7, under *How to use the zip file* (students have similar setup instructions in Appendix A of the book).

Objectives

We believe that instructional objectives should be the start of any educational methodology, so we provide a set of objectives for each chapter in the book. We prepared these objectives based on the principles presented by Robert F. Mager in his classic book, *Preparing Instructional Objectives*. As a result, our objectives describe the skills that your students or should have when they complete a chapter, and you should be able to test whether they can apply those skills.

Beyond that, we've tried to make sure that each objective describes a skill that a professional programmer should have. This gives our objectives a real-world context that you usually won't find in the objectives for other books. So, if your students can do what the objectives state when the course is over, you can be sure that they have learned the skills that they will actually need on the job.

If you review the objectives for one of the chapters, you'll see that the first objectives for each chapter are what we refer to as *applied objectives*. These ask the students to apply what they've learned as they develop C# applications. These of course are the critical objectives of a programming course, and they are best tested by having the students do exercises and projects like the ones that we provide.

After the applied objectives for each chapter, you'll find what we refer to as *knowledge objectives*. These objectives define skills like identifying, describing, and explaining the required concepts, terms, and procedures. These objectives determine whether your students are able to talk intelligently about the topics that are presented. And these objectives can be tested by the questions in our test banks.

To help you get the most from the instructional objectives, we have included them at the start of the PowerPoint slides for each chapter. As we see it, if you can convince your students that they only need to be able to do the skills that are described by the objectives, their study becomes more focused and efficient.

Test banks

To test comprehension, the instructor's materials include one test bank for each chapter in the book. We developed these test banks in ExamView, and we provide them in multiple formats, including those that can be used in various LMSs (like Blackboard, D2L Brightspace, and Canvas), as well as Rich Text Format (for Microsoft Word).

Each test bank provides multiple-choice questions that are designed to test the skills described by the objectives for that chapter, and each test question is designed to test the skill described by one objective. This keeps the promise to the students that they will only be expected to do the skills that are described by the objectives.

Besides matching our questions to the objectives, we use this guideline to check the validity of each question: *A professional C# developer should be able to get the right answer.* This guideline eliminates questions that test the knowledge of trivial details that no one should be expected to remember. This guideline also forces us to focus on questions that test the concepts and skills that are required on the job.

Extra exercises and solutions

Because we provide the solutions for the book exercises in the student download for this book, the instructor's materials include a second set of chapter exercises.

We've provided these exercises in a Word document so you can modify them if you want to, as well as a PDF document that you can distribute to your students if you don't want to make any changes. The starting directories and database files are also included so you can distribute them to your students. And the directories for the solutions are provided so you can demonstrate and review them in class.

Both the book exercises and the extra exercises force the students to use all of the critical skills for developing C# applications. So you can assign any of them to your students. The only significant differences are these:

- For the book exercises: You don't need to distribute any files when assigning an exercise, but your students will have the solution to the exercise.
- For the extra exercises: You need to distribute the exercise specification and its starting point when assigning an exercise, but your students won't have the solution to the exercise.

If you assign the extra exercises, keep in mind that some of the starts for these exercises are the solutions to previous exercises. For example, the start for extra exercise 6-2 is our solution to extra exercise 5-3. Because of that, you probably won't want to give your students all the extra exercise starts at once. Instead, you can give them the starts as you assign the exercises.

If you assign any of the extra exercises for chapters 20, 21, or 22, you should know that they use the same MMABooks database that's used by the book applications and the exercises in the book. Because of that, your students should already have the files for this database.

Projects and project solutions

To test how well your students can develop C# applications from scratch, the instructor's materials include a set of projects that you can assign as your students progress through the book. We've provided the projects in both Word and PDF documents, so you can modify them or use them as is, whichever you prefer.

The projects are numbered by section. In addition, the description for each project indicates what chapters your students must complete before they can do that project. For example, the description for project 2-1 (the first project for section 2) indicates that your students will have the skills they need to complete that project after they've read chapters 1 through 9.

Keep in mind, however, that the sections don't have to be read in sequence. Because of that, the projects for sections 4 require only the skills presented in chapters 1 through 12, and the projects for section 5 require only the skills presented in chapters 1 through 12 plus 18 (the LINQ chapter).

The projects for section 5 require a TechSupport database, and we provide the files for it. You can distribute the files to your students to install the database on their own systems. Or, you can run the SQL script we've provided to create the database on a server at your school or company or on the computers in a lab.

Because most of the projects take only an hour or two to develop, you can use selected projects as tests that are done in computer lab. That of course is the only sure way to see whether your students can do the applied objectives for this book. And the major goal of our book is to help your students meet those objectives at a professional level.

All of the solutions to the projects are also included in the instructor's materials. That way, you can present them in class or compare them with your students' solutions.

PowerPoint slides

The PowerPoint slides present all of the critical information from the figures in the book. That includes all of the screen shots, diagrams, tables, and code that you may want to review in class. As a result, these slides make it easy for you to review any of the skills that your students have difficulty with.

Beyond the book information, the slides for each chapter start with the instructional objectives so you can review them in class. And they end with the screen shots for the book exercises, the extra exercises, and the projects. That makes it easy for you to answer questions and provide additional information about these exercises.

How to get started

You can request the instructor's materials for our book from our instructor website (www.murachforinstructors.com) and download them from your account page there. The download is available as a zip file. Then, you can install the materials on your computer as detailed below.

Once the installation is done, you can do a thorough review of all of the materials. In particular, you'll want to run some of the book applications and solutions to the various sets of exercises to see the level of competence that our book develops. You'll also want to click through some of the PowerPoint slides to see how they can help you review and reinforce the information that's presented in the book.

To help you find what you're looking for, the entire file structure for the instructor's materials is shown on the next page.

How to use the zip file

1. Download the zip file of instructor's materials from your Murach account page.
2. Create a directory named *Murach* directly on your hard drive.
3. Unzip the zip file into the *Murach* directory to create a file structure that starts with
`C:\Murach\C#`
4. Use the File Explorer to find the file named *copy_exercises.bat* in the `C:\Murach\C#\Student Download\` directory.
5. Double-click the *copy_exercises.bat* file. This will copy the Exercise Starts directory to a `C:\C#` directory. Then, the file paths will match up with the exercise instructions in the book, and the exercise solutions will work properly when you run them.

The student directories and files that get installed

C:\Murach\C#\Student Download\...	Contents
Book Applications	The applications presented in this book.
Exercise Starts	The starting points for the exercises that are at the end of each chapter in the book.
Exercise Solutions	The model solutions to the book exercises.
Database	The files for the MMABooks database and the SQL script used to create this database.
copy_exercises.bat	A routine that copies the Exercise Starts directory to a path starting with C:\C#, a file structure that matches up with the instructions in the book.

The instructor directories and files that get installed

C:\Murach\C#\Instructors\...	Contents
Objectives.docx Objectives.pdf	A document in both Word and PDF formats that contains the instructional objectives for all chapters.
Extra Exercises\Extra exercises.docx Extra Exercises\Extra exercises.pdf	The specifications for a set of chapter exercises that are similar to the exercises in the book; in Word and PDF formats, so you can change the specs or hand them out as is.
Extra Exercises\Extra Starts	The starting points for the extra exercises.
Extra Exercises\Extra Solutions	The model solutions for the extra exercises.
Projects\Projects.docx Projects\Projects.pdf	The specifications for a set of projects that require your students to create applications from scratch; in Word and PDF formats, so you can change the specs or hand them out as is.
Projects\Project Solutions	The model solutions for the projects.
Projects\Database	The files for the TechSupport database used by the projects for section 5 and the SQL script used to create this database.
Projects\Files	The text and binary data files used by the projects.
Slides	One PowerPoint file for each chapter. Each file starts with the instructional objectives, and ends with screen shots for the various sets of exercises and projects.
Test Banks	One test bank per chapter, organized by format: ExamView, RTF (Word), Blackboard (which can be imported into Canvas and D2L Brightspace), Respondus, and IMS QTI (the standard test bank format).

Any comments?

If you have any comments about our book or its instructional materials, we would be delighted to hear from you. If you discover any errors in our applications or exercise solutions, we would appreciate hearing about them. And if you want to let us know that you're going to adopt our book for your course, that would make our day.

Just email us at the addresses below. But whether or not we hear from you, we want to thank you for your interest in our C# book.

Anne Boehm, Author
anne@murach.com

Judy Taylor, Educational Liaison
judy@murach.com