1

Introducing the Programming C# Certification

WHAT YOU WILL LEARN IN THIS CHAPTER

- ➤ Getting certified
- ➤ Understanding Microsoft certifications
- ➤ Understanding Microsoft certification exams
- ➤ Studying for the exam

This chapter is an introduction to Microsoft certifications in general and the Programming C# certification more specifically. In this chapter, you learn about the world of Microsoft certifications and why you might consider taking this exam and getting certified. The chapter also presents information on how exam questions are considered and written for Microsoft exams and describes how you can use this book to study for Exam 70-483.

A complete list of the topic areas that are covered on Exam 70-483 is also included to help you understand what to expect for objectives of each exam as you work toward your MCSD certification using the C# programming language.

Due to the nature of the content of Chapter 1, there are no code downloads for this chapter.

GETTING CERTIFIED

Certifications have been around for many years. Hardware manufacturers certify components, car dealers provide certified used cars, developers certify software to run on specific operating systems—and that's just to name a few.

What this basically means is that the term *certification* can have many different definitions depending on the context in which it is used. In this book, certification refers to Microsoft certifications. (Specifics about Microsoft certification are presented in the section "What Is MS Certification?" later in this chapter.) Like many other large companies in the IT industry, Microsoft has established and maintains a certification program to show developers' aptitude in designing and developing programs using Microsoft's products.

Although each organization, including Microsoft, has its own certification program, benefits, and requirements, there are still a lot of similarities among certifications. For example, most, if not all organizations, deliver their certification exams through an exam delivery partner (EDP). This has typically been through EDPs such as Prometric, Pearson Vue, and Certiport.

Each program requires that the candidate register in the program and take specific exams and meet certain requirements before the participant is awarded a certification. Some programs require one exam for a certification credential, and others require multiple exams. For example, Cisco, Microsoft, Adobe, Novell, and Oracle all have certifications you can earn by taking one knowledge-based exam. Some certifications are more difficult than others.

The more complex, multi-exam scenarios are found in the high-end certifications such as the Microsoft Certified Master (MCM), which is changing to Microsoft Certified Solutions Master (MCSM), or Cisco Certified Internetworking Expert (CCIE) certifications. These certifications and exams require much more than just a knowledge-based exam. These certifications require a candidate to complete lab-based portions, meaning that the participant performs actual tasks in either a real or emulated environment.

For the Cisco exam, you visit the testing center and configure the necessary network switches, routers, and firewalls according to a specification. The exam team then introduces bugs, or essentially breaks your configuration, and you have to troubleshoot the issues and fix it.

The MCM program has different requirements depending on the certification you are seeking. For most of the MCM certifications, you attend classroom training, take knowledge-based exams, and take a final lab-based exam over a 21-day period—that is, 21 days straight with no breaks in between. Some of the MCM programs permit the candidate to take training at different institutions and then take the requisite knowledge-based exams with a final performance-based lab-style exam at the end, hosted by Prometric.

Obviously, the more stringent the requirements for a certification, the more credibility the certification holds in the industry. It also means a higher cost, but with that rigorous certification in hand, you can also demand—and usually get—more money for consulting fees or a higher salary. But that is getting into the next section: why you should get certified.

Why Get Certified?

Obviously, if you purchased this book, you have already decided to get a certification, or at least take a certification exam. Of course you may also be just borrowing the book because you are curious about what might be involved in getting certified. Either way, this section describes some of the reasons why you might consider getting certified.

Having spent a lot of time pursuing certifications in the past, plus working as a Certification Product Planner, the reasons I have come across are varied and many. For the most part, to the reasons are summarized and rationalized for why certifications are good and why you may want to pursue them.

In the IT industry, especially in the realm of the developer world, most of the programmers who have been in the industry for some time came through academia and hold university degrees, typically in computer science. When you think about it, a Bachelor or Master's degree is a certification from a certain perspective. The degree shows the world that the person whose name is indicated on the degree certificate has met the requirements as set forth by a board of some sort, usually the university faculty and a governing body.

Not every programmer, database developer, database administrator, or other IT professional, however, has attended a four-year degree program at a university. Many have instead taken classes at two-year certificate programs. Whatever the institution or schooling background, upon successfully completing the program, students acquire a diploma, certificate, degree, or other named piece of documentation that indicates they have achieved some specific level of knowledge.

One of the problems that graduates face after completing these programs is that the knowledge they gained during the course of their schooling is actually outdated to a certain extent. You might think computer science concepts don't change at their core, and to a great extent, you are correct. But what does change are the technology and tools IT professionals use every day in their pursuit of the computer science career upon which they have embarked. A good example of this is how the Internet and the World Wide Web have changed your concept of what an application is. Just in the short lifespan of the web, you have seen the technology change from static pages with hyperlinks to pages supporting Cascading Style Sheets, JavaScript, ActiveX controls, server-side programming, state-management, and so on. None of these technologies or concepts were taught just a short time ago, yet they are relevant and important today.

Employers looking to hire programmers for developing websites that contain these technologies require some way to identify who has those skills. The Bachelor of Science document certainly doesn't indicate this. Actually, there isn't any way for an employer to know what courses a holder of a degree has actually taken unless the student provides a transcript. There isn't any way to show an employer what knowledge and skills were gained after attending a university either outside of a resume or perhaps a portfolio.

Industry certifications are a way to address some of these issues. When properly implemented, secured, and executed on, industry certifications are an effective way to show existing and potential employers some important information. Certifications can provide the person who holds the credential, the following benefits:

➤   Validation of knowledge

➤   Validation of skills

➤   A way to show continuing education

➤   A means to prove a commitment to maintaining skills

Whatever your reasons for pursuing a certification, you must understand the value of the certification you intend to acquire and perhaps even the process by which the certifications are developed

and maintained. The next few sections present an insight into Microsoft certifications, including what certifications Microsoft makes available, how they fit into the overall certification portfolio for Microsoft, and an insider's view of how the certifications are planned, created, and delivered.

> **EXAM PIRACY AND BRAIN DUMPS**
>
> Many opponents to certifications cite reasons such as exam piracy and brain dumps for their opposition. Some indicate that employers don't value certifications, and therefore they aren't worth the money you would spend on them. Others, such as programmers, tell you that they don't work in a world of multiple choices, so a multiple-choice test isn't representative of their skills and knowledge.
>
> Microsoft is not the only company affected by exam piracy and brain dumps. Any certification program is subject to these same issues. Although there are certain methods in existence to help deal with these problems, they can never be totally eradicated.
>
> Microsoft is actively taking steps to counter some of these issues by performing exam analysis, in the form of statistical forensics, to help identify cheaters and exam centers that are at the heart of the problem. Over the past few years, it has been successful in shutting down testing centers that participate in exam piracy and websites that contribute to brain-dump activities. Because of the way the Internet works, however, it is impossible to completely stop all the brain-dump sites.
>
> One of the best ways to help combat these issues is through the use of education and cooperation with certified professionals. Any time someone asks you for a brain dump or a way to cheat on a Microsoft exam, ensure that you explain the benefit of achieving the certification honestly and report any cheating activity to Microsoft. You can help drive the acceptance of your certifications and help to improve the reputation of these exams by helping to reduce exam piracy and cheating.

What Is MS Certification?

Microsoft certifications have evolved over the years. The exams have changed in their content, and the process for creating the exams has changed somewhat as well. Like most certification programs, the changes are based on customer feedback, changes in the industry, and standards board certifications.

Microsoft certifications are most commonly known as *MCP certifications*. MCP stands for Microsoft Certified Professional. To understand the Microsoft certification landscape, look at some terminology, what certifications are available from Microsoft, and how to obtain them.

Throughout this chapter, the terms *certification* and *credential* are used interchangeably. A certification is defined as a "title" that candidates can use after they complete the requirements set forth for that certification. Credential is another word for a certification. Again, it is a title that candidates can use after completing the requirements for that credential.

An example of a certification would be Microsoft Certified Solutions Associate (MCSA), Microsoft Certified Solutions Developer (MCSD), Microsoft Certified Solutions Expert (MCSE), or Microsoft Certified Solutions Master (MCSM). The following sections describe each of these designations.

Certification Changes

The MCSD and older Microsoft Certified Systems Engineer (MCSE) certifications served the industry well for many years. Like all programs, changes and improvements were a necessity. Some of these changes were brought about by the need to streamline requirements and simplify the program, whereas other driving reasons were employers and hiring managers.

As technology changes, IT professionals either keep their skills up to date or they do not. This can create a bit of an issue for hiring managers trying to discern qualifications from resumes submitted. For example, programmers could indicate that they hold an MCSD certification, but don't tell the hiring manager what programming language was used to achieve the credential. If the hiring manager is looking for a developer who could program using C++, the certification didn't actually tell them that. Also, what elective exam did candidates use? How much web experience did they have as opposed to Windows development experience?

Both candidates and hiring managers provided feedback to Microsoft, telling them that it was not easy to determine just what the certification name meant or what requirements were needed for a certification. The elective system made it difficult to determine qualifications. Other feedback indicated that one certification didn't necessarily map to the way the industry thought about job roles and skills qualifications.

As a result of this feedback and industry research, Microsoft made changes to the program and created new certifications and new exams to help address these issues and needs. It termed this new program the *New Generation of Certifications* and labeled the old system as *Legacy Certifications*. Then, just a few short years following that change, Microsoft introduced the current version of certifications, which is the third iteration of the certification, or cert, program. These changes are not designed to confuse you, but instead are intended to help ensure that your credentials have validity and meaning in the workforce.

The Initial Certifications (Version One)

The first iteration of the Microsoft certification program created a base credential known as MCP. This was the starting point for any of the higher certifications that consisted of:

➤ MCSE

➤ MCSD

➤ MCDBA (Microsoft Database Administrator)

These were the mainstream certifications that existed in version one of the Microsoft certification programs. They served Microsoft well for a number of years, and these three credentials became well known in the industry.

A New Generation of Certifications (Version 2)

As with any program, there is the potential not to meet the needs of every concerned party. Feedback to, and research by, Microsoft Learning resulted in some changes to the certification program. The changes were designed to address a couple of key areas: job roles and knowledge validation.

Both of these areas have some commonality. Hiring managers needed a way to identify which specific technologies a potential job candidate might have, and they needed a way to map the certifications to job roles. Only developer certifications are covered here.

Microsoft Certified Technology Specialist (MCTS)

For validating knowledge on a technical subject, Microsoft created the Microsoft Certified Technology Specialist (MCTS) credential. The MCTS certification is not considered to be an entry-level certification but is aimed at candidates who want to prove their knowledge and skills on a specific Microsoft technology.

To achieve an MCTS certification, the candidate would have to pass one or more exams. The MCTS certifications and exams also allowed Microsoft to provide a more valid way of testing candidates' knowledge and skills on a technology by permitting them to include more complete coverage of that technology. To explain this a little better, consider how the older MCSD certification focused on either web or Windows development, which is not bad, except that these two platforms encompass a lot of different programming skill sets.

By using the MCTS exam focus and the different technologies that Microsoft was releasing, adequate coverage of each different technology could now be included in a separate exam. This allowed Microsoft to provide sufficient coverage of a technology on an exam and also clearly state what that technology is. Candidates passing that exam would have proven their knowledge and skills on that technology. Hiring managers now had a way to determine what the job candidate was certified on.

Microsoft Certified Professional Developer (MCPD)

Microsoft also created another layer of certification known as the professional level. This credential is titled Microsoft Certified Professional Developer (MCPD). To achieve an MCPD credential, the candidate must pass any prerequisite MCTS certifications along with the MCPD exam.

The MCPD exams are designed to test a candidate's ability to work as a team lead or development lead and make decisions around application designs. There are also not as many MCPD credentials as there are MCTS credentials, and this is to support the fact that they are intended to focus on a job role as opposed to a breadth of technologies.

The Current Microsoft Certifications (Version 3)

The current version of Microsoft certifications changes the focus a bit more by looking at credentials such as Specialist and Solutions Experts or Solutions Developers. The changes made for the current set of credentials were designed to do the following:

➤ Reduce the number of entry points

➤ Reduce the number of certifications

➤ Clarify certification paths

➤ Enable single base certification to lead to multiple advanced certifications

➤ Streamline the program into a seamless process

Again, the reasons for changing the certification program were in response to industry feedback and research. As noted in the preceding list, the previous generations of certifications were still confusing due to the number of entry points into the program, with multiple paths and many different certifications and exams. The current program focuses on three levels of certification.

There are three main tiers in the latest certification program:

➤ **Solutions Associate level:** Designed to be the foundation for certifications in Microsoft proving technical skills.

➤ **Solutions Expert level:** Expands on the knowledge of those at the Associate level and requires more rigor in the exams and knowledge tested. Candidates at this level should be able to build solutions using multiple technologies including cloud computing.

➤ **Solutions Master level:** The top of the Microsoft certification program. Consists of certifications that require knowledge-based exams along with performance-based testing. Those who hold a Masters certification demand higher salaries.

Other Microsoft Certifications: The MTA

All the preceding certifications can be thought of as the technical certifications. Think of technical certifications as a set of exams and credentials intended to validate skills. There is another set of credentials that fall under the acronym MTA (Microsoft Technology Associate).

Actually, Microsoft is careful not to actually refer to the MTA as a certification. It is more of a certificate. The MTA is aimed at high school students and post-secondary institutions that offer two-year certificate programs; although, four-year universities can certainly deliver them as well.

Although the exams are technical in nature, they are designed to be entry level, and 80 percent of the content is intended to be knowledge level as opposed to implementation-specific. What that means is the questions are designed to test candidates on their understanding of the concepts, such as the following:

➤ What is a class in object-oriented programming(OOP)?

➤ What is a tuple in a database?

The MTA exams are used by some schools to augment their existing tests, and sometimes to replace them, for determining a student's knowledge of a subject area. These exams are also designed to serve a few more purposes, the most pertinent being that they provide students with a sense of achievement, helping them to realize their progression in their learning. Plus, they provide a means to introduce students to the world of certifications by exposing them to a Microsoft exam environment. If they pass the exam, they get access to the Microsoft Certified Professional community, where they can start preparing for the more technical certifications with the help of the MCP community and resources available there. The MTA has been well received by the academic community.

THINGS TO KNOW ABOUT THE TEST

For most developers and IT professionals outside of Microsoft, or even Microsoft Learning, the exam development process is a black box. For a developer, it's analogous to a Windows Communication Foundation (WCF) service. You know how to call it and get a result back, but you have no real insight into the algorithms that make it work. You can guess at it, but you're never quite certain.

It's always an eye-opening experience when someone steps into the process for the first time and sees what it takes to create these exams. The next section describes how the exam questions are written, but first it can help you to understand how an exam is created.

How the Test Is Created

At one time, Microsoft certification exams focused on product features. After all, it was the features of the product that developers were using to create their applications, and it was the features of the product that customers asked for, so it stood to reason that the features of the product were the important aspects to be testing on. Or does that logic make sense?

The history of these exams has shown that this methodology doesn't quite present a good testing experience, nor does it provide any validation that a candidate can actually use the features. It merely shows that developers can memorize what a feature is or does. The current process has been put in place to overcome these issues and to address some others as well. Psychometrics has been added to the certification exams. *Psychometrics* is a field of study dealing with the theories and techniques used to validate knowledge and skills through a measurement process. In this case, this measurement is a test. Before getting into how psychometrics is involved, first look at how the exam envisioning and design has changed to better address industry needs.

Microsoft releases new versions of software, on average, about every two years. A Product Manager and a Product Planner in Microsoft Learning work together to evaluate the changes in the next version of the product and how it will impact the industry. For example, a careful evaluation was made of all the technologies that make up Microsoft's .NET Framework to determine how the new features will be applied by developers in creating Windows or web-based applications. How have the data access mechanisms changed? What is new in WCF services?

After this information is evaluated, the Product Manager and Product Planner start to seek out developers in the industry who use these new technologies in their organization. As you can imagine, these developers will typically be early adopters who partner with Microsoft to gain access to early builds of the software. They also consist of Microsoft Most Valuable Professionals (MVPs) and Microsoft Certified Trainers (MCT). The criteria are clear. These developers must use the new software in real-world scenarios, and can describe how the new features are used and will be used by the industry.

Microsoft then hosts focus group sessions, typically in Redmond, Washington, with these industry experts to determine how the technologies are used. These sessions do not focus on features only. The sessions are designed to extract product-usage scenarios from these experts on how they use the technology, regardless of feature sets, in the real world. Obviously, there must be a focus on the new aspects of the software, or the exam becomes a rehash of the previous version.

The exam prep guide is the output of this focus group—well, sort of. The prep guide structure is explained a bit later in the section that details the objectives that this exam will test on, but for now, just know that the prep guide is the result of the focus group. The information taken from the focus group is formulated into the exam design document that gets a further validation pass by more industry experts. This validation step is known as a *blueprinting process*, where other industry experts who have never seen the list before and who did not participate in the focus group can look at each outlined objective and rate it based on relevancy, importance, and frequency. These values are fed into a spreadsheet that executes some magical psychometric formulas that spit out the number of required exam questions for each objective to appropriately measure the candidate's knowledge on the test. After the blueprinting is complete and the data is assembled, the exam question writing can begin. Note that as of the writing of this book, the exam questions are still in multiple choice or true/false format. Some newer items are being tested that consist of drag-and-drop or choosing code segments, but the bulk of the questions are multiple choice.

> **NOTE** *Microsoft is committed to moving to a performance-based testing environment for all its certifications at some point. There are many hurdles to overcome before it is a reality, but that will change the face of Microsoft certifications considerably.*

How Questions Are Written

Just how do the questions get written? Microsoft Learning works with various partners to create the content for the exams in a clear process that is guided and overseen by the Product Planners, the Content Development Managers, and the Project Managers at Microsoft.

Taking the exam design document and the blueprint values, a team of item writers is assembled to begin the process. These item writers must be industry experts as well, who work with the technology on a daily basis. They receive training on effective exam question writing.

This might sound a little strange at first. You may be saying to yourself, "Why would you need to have training on how to write a test question? If you know the technology, you can write a test question on it."

Although there is some truth to that thought, writing an effective exam question is not always an easy task. Here are the reasons why:

➤ The question must test the objective it maps to.

➤ The question must be worded in a technically accurate and correct form.

➤ Slang or nicknames cannot be used. (An example of this is in the IT world where in North America the acronym DMZ has been used to represent the perimeter network for security purposes. In certain other countries, DMZ has negative connotations.)

➤ Wording and terms must take into account translation into other languages.

➤ Each question must be legally defensible. That is to say, if the question is asking for one answer, there can be only one correct answer among the available answer choices. All other answers *must* be 100 percent incorrect.

➤ The writer cannot make up technologies or answers that do not exist in the product just to provide a wrong answer.

➤ Questions cannot be tricky with subtle wording that hides clues.

➤ Questions cannot be simply recall questions where a candidate would normally look up the answer in MSDN or use IntelliSense. An example would be writing a question that tests the order of parameters for a method call for a class in the base class library.

➤ One question on the exam cannot give away the answer to another question on the exam.

➤ Questions must be written to the correct cognitive level.

As you can see, there are quite a few rules involved in the acceptance criteria for the questions. Most writers think they will turn out their questions with minimal trouble because they know the technology so well, but they soon find out that good exam questions take hard work and careful thought.

After the questions have been written, the next formal part of the process is to hold an Alpha session. The Alpha session typically involves the lead item writer plus six to eight more subject matter experts who go over each written question. The original item writers are not involved in this process so that nobody's feelings get hurt when the questions are critically reviewed. It also helps the subject matter experts in the room to focus on being honest about the question's merits.

Any problematic questions either get fixed or completely rewritten during this five-day session. The output of this session is the set of questions included in the beta version of the exam. The beta version is where as many as 500 sets of eyes have a chance to evaluate the questions. Each beta candidate has the opportunity to provide comments and feedback on the items at the end of the exam. The feedback and comments are reviewed after the beta has completed, and a post-beta session is held where even more subject matter experts are involved. The task this time is to validate the comments and feedback, and then to set the passing score for the exam. At this stage, any questions that did not perform well on the exam or have technical issues are deleted from the final pool of exam questions.

This entire process can take anywhere from six to nine months from the design phase to release of an exam. The exam's planning process starts much sooner than that, of course, but the actual exam design, development, testing, and release portion can take this long.

EXAM TIPS AND TRICKS

Not every process is perfect, and even with this many subject matter experts looking at exam questions, some minor issues can escape notice. When you take the exam, remember there is a comment period at the end where you can submit your feedback on the exam or on individual questions. Don't be afraid to be brutally honest. At the same time, ensure that you provide usable feedback. Responses such as "This question stinks" are not actionable and do not identify issues with the question. The feedback can go a long way to help improve the quality of the exam questions.

HOW TO STUDY FOR THE EXAM USING THIS BOOK

And now you come back to the reason why you bought this book. Your objective is to study for and pass the exam, and you purchased this book to help you do that. Outside of the great information presented in this book, you gain advice on how you can use the book more effectively to help in your exam preparation.

Although there are many ways to start preparing for an exam, only a structured method helps to ensure success. Typically, when looking at exam preparation, a candidate faces a situation similar to an author staring at the first blank page for a book he is writing. Where do you start? Not only that, but you may also be thinking that preparing is going to be hard because you don't know what you don't know. You're not sure you want to study everything because you should already know most of the content that will be covered, but how do you know what to focus on? The following sections can help you determine just that.

Prep Guide

The first thing you should do is to focus on the exam prep guide that lists the objectives for the exam. (The objectives are included at the end of this chapter for your convenience, but you can also find them at `www.microsoft.com/learning/en/us/exam.aspx?ID=70-483`.) These objectives provide you with an idea as to what could be covered on the exam. There is a caveat that comes with the prep guides, however. You may notice the following wording under each objective that states,

"This objective may include but is not limited to…."

This text is an indication that the listed items after this text are the identified areas of coverage from the exam design sessions. The list is typically not complete for various reasons, such as not all topics were thought of during the design or complete coverage may not be possible.

Regardless of the reasons that there may not be complete coverage, the items are just indicators of what you may see a question written on. The other issue is that the exam designers and the item writers are, for the most part, different people. This means that the person writing the exam question was not present during the exam design and therefore was not privy to the conversations around this topic. They also have their own experiences that they bring to the process for what they will draw upon for writing the questions they have been assigned.

This doesn't mean there is a disconnect in the process or it is flawed. It is similar to the exams you took in school. You were expected to understand the subject to the extent that you could answer any question on the subject. You were not given explicit topic coverage on those exams either. These exams are similar in the prep guides in that they offer a little more information as to what may be covered, but as long as you fully understand the subject, you should be able to answer any question related to it.

Functional Groups

So step one is complete; you have reviewed the prep guide and evaluated the objectives, and now you have an idea what the exam questions will test on. Before you spend time studying topics, you should rate what you think your knowledge is for each of these objectives. Don't worry too much about the

bolded items in the objectives listed here (the ones with the percentage ratings in parentheses). Those items are known as *Functional Groups*, and they are a convenient way to group related objectives. The percentage listed gives you an idea of how much of the exam a particular Functional Group will take up. You can use this value to determine where to focus your time studying if you want.

Practice Questions

After you have the objectives rated, turn your attention to practice questions. Practice test questions are a great way to evaluate your knowledge against what you think you know, and against reality. They are also a great way to focus your mind on how the exams are written and what the experience may be like when you take the actual exam. The other advantage you gain from the practice questions is the ability to identify your weak areas, allowing you to focus your study and maximizing your investment in preparation.

Preparation

The preparation part is primarily where this book comes in. The chapters of this book map directly to the exam objectives. This means that the book is focused training for the exam. This doesn't mean that it is a cheat sheet or a brain dump. To gain the most benefit, you need to read and understand the content of the chapters. You also then get to apply that understanding through the code labs in each chapter. These labs are designed to reinforce the theory presented in the chapter. Each chapter will also contain practice questions, Cheat Sheets, and Key Terms to help you focus on the right content.

The prep guide found on the Microsoft Learning Web site will help you to identify the key aspects of the exam itself along with the skills measured, a list of preparation materials such as courses or books, as well as a community section designed to provide resources from your fellow exam candidates and developers. The community can be a great study resource as well.

The authors of the book have done their best to evaluate the exam objectives and to provide you with material designed to help you prepare for the exam. Your study habits and how well you understand the content presented here will be factors in your success. The more experience you gain with the technology and the more you practice the labs in this book, the greater your chances to successfully pass the exam.

THE 70-483 OBJECTIVES

The following section lists the objectives for Exam 70-483, the topic of this book. The objectives are taken directly from the prep guides that you can find online at `www.microsoft.com/learning` under the Certifications tab.

Manage Program Flow (25 Percent)

Under this category, you will find topics that deal with threading, program flow, events, callbacks, and exception handling—all are important to managing how your application is executed.

Implement Multithreading and Asynchronous Processing

This objective may include but is not limited to use the Task Parallel library (ParallelFor, Plinq, and Tasks); create continuation tasks; spawn threads by using ThreadPool; unblock the UI; use async and await keywords; and manage data by using concurrent collections.

See Chapter 7, "Multithreading and Asynchronous Processing."

Manage Multithreading

This objective may include but is not limited to synchronize resources; implement locking; cancel a long-running task; and implement thread-safe methods to handle race conditions.

See Chapter 7, "Multithreading and Asynchronous Processing."

Implement Program Flow

This objective may include but is not limited to iterate across collection and array items; program decisions by using switch statements, if/then, and operators; and evaluate expressions.

See Chapter 2, "Basic Program Structure."

Create and Implement Events and Callbacks

This objective may include but is not limited to create event handlers; subscribe to and unsubscribe from events; use built-in delegate types to create events; create delegates; lambda expressions; and anonymous methods.

See Chapter 6, "Working with Delegates, Events, and Exceptions."

Implement Exception Handling

This objective may include but is not limited to handle exception types (SQL exceptions, network exceptions, communication exceptions, and network timeout exceptions); catch typed versus base exceptions; implement `try-catch-finally` blocks; throw exceptions; determine when to rethrow versus throw; and create custom exceptions.

See Chapter 6, "Working with Delegates, Events, and Exceptions."

Create and Use Types (24 Percent)

Creating and using types will take you into the world of C# data. It covers the built-in types that C# provides such as `int` and `string` but also delves into the more complex types such as structs, enums, and classes.

Create Types

This objective may include but is not limited to create value types (structs, enum), reference types, generic types, constructors, static variables, methods, classes, extension methods, optional and named parameters, and indexed properties; and create overloaded and overridden methods.

See Chapter 3, "Working with the Type System."

Consume Types

This objective may include but is not limited to box or unbox to convert between value types; cast types, convert types, and handle dynamic types; and ensure interoperability with unmanaged code, for example, dynamic keyword.

See Chapter 4, "Using Types."

Enforce Encapsulation

This objective may include but is not limited to enforce encapsulation by using properties, by using accessors (public, private, and protected), and by using explicit interface implementation.

See Chapter 3, "Working with the Type System."

Create and Implement a Class Hierarchy

This objective may include but is not limited to design and implement an interface; inherit from a base class; and create and implement classes based on `IComparable`, `IEnumerable`, `IDisposable`, and `IUnknown` interfaces.

See Chapter 5, "Creating and Implementing Class Hierarchies."

Find, Execute, and Create Types at Runtime Using Reflection

This objective may include but is not limited to create and apply attributes; read attributes; generate code at run time by using CodeDom and lambda expressions; and use types from the System. Reflection namespace (Assembly, PropertyInfo, MethodInfo, and Type).

See Chapter 8, "Creating and Using Types with Reflection, Custom Attributes, the CodeDOM, and Lambda Expressions."

Manage the Object Life Cycle

This objective may include but is not limited to manage unmanaged resources; implement IDisposable, including interaction with finalization; manage IDisposable by using the Using statement; and manage finalization and garbage collection.

See Chapter 5, "Creating and Implementing Class Hierarchies."

Manipulate Strings

This objective may include but is not limited to manipulate strings by using the `StringBuilder`, `StringWriter`, and `StringReader` classes; search strings; enumerate string methods; and format strings.

See Chapter 4, "Using Types."

Debug Applications and Implement Security (25 Percent)

This section focuses on aspects for understanding how you work with the tools and features of the .NET Framework to debug your applications and for implementing security in your code for encryption and validation.

Validate Application Input

This objective may include but is not limited to validate JSON data; data collection types; manage data integrity; evaluate a regular expression to validate the input format; use built-in functions to validate data type and content out of scope; and writing regular expressions.

See Chapter 11, "Input Validation, Debugging, and Instrumentation."

Perform Symmetric and Asymmetric Encryption

This objective may include but is not limited to choose an appropriate encryption algorithm; manage and create certificates; implement key management; implement the System.Security namespace; hashing data; and encrypt streams.

See Chapter 12, "Using Encryption and Managing Assemblies."

Manage Assemblies

This objective may include but is not limited to version assemblies; sign assemblies using strong names; implement side-by-side hosting; put an assembly in the global assembly cache; and create a WinMD assembly.

See Chapter 12, "Using Encryption and Managing Assemblies."

Debug an Application

This objective may include but is not limited to create and manage compiler directives; choose an appropriate build type; and manage programming database files and symbols.

See Chapter 11, "Input Validation, Debugging, and Instrumentation."

Implement Diagnostics in an Application

This objective may include but is not limited to implement logging and tracing; profiling applications; create and monitor performance counters; and write to the event log.

See Chapter 11, "Input Validation, Debugging, and Instrumentation."

Implement Data Access (26 Percent)

Most applications work with data in some form or another. Data may be stored in database systems, or it may be stored in flat files. Flat files may be text files, comma-separated value (CSV) files, or XML files. Knowing how to access this data for reading and writing is crucial for developers.

Perform I/O Operations

This objective may include but is not limited to read-and-write files and streams; read and write from the network by using classes in the System.Net namespace; and implement asynchronous I/O operations.

See Chapter 9, "Working with Data."

Consume Data

This objective may include but is not limited to retrieve data from a database; update data in a database; consume JSON and XML data; and retrieve data by using web services.

See Chapter 9, "Working with Data."

Query and Manipulate Data and Objects by Using LINQ

This objective may include but is not limited to query data by using operators (projection, join, group, take, skip, and aggregate); create method-based LINQ queries; query data by using query comprehension syntax; select data by using anonymous types; force execution of a query; and read, filter, create, and modify data structures by using LINQ to XML.

See Chapter 10, "Working with Language Integrated Query (LINQ)."

Serialize and Deserialize Data

This objective may include but is not limited to serialize and deserialize data by using binary serialization, custom serialization, XML Serializer, JSON Serializer, and Data Contract Serializer.

See Chapter 9, "Working with Data."

Store Data in and Retrieve Data from Collections

This objective may include but is not limited to store and retrieve data by using dictionaries, arrays, lists, sets, and queues; choose a collection type; initialize a collection; add and remove items from a collection; use typed versus nontyped collections; implement custom collections; and implement collection interfaces.

See Chapter 9, "Working with Data."

SUMMARY

This chapter provided an overview of the Microsoft certification program and what to expect from this book in preparing for the 70-483 Exam, which focuses on Windows Store applications development using C#.

This chapter explained the history of Microsoft certifications, how they have changed over the years, and why those changes were made. This will help you understand how the certification program is positioned in the industry, and what you can expect as a result of achieving a Microsoft certification.

The process of creating certifications and exams is a complex task that involves many participants, lots of research and planning, and an orchestrated set of procedures to create exams that are relevant in the industry and provide a good balance of feature and usage scenario coverage.

The list of objectives for the exam that you will take on your way to the MCSD certification will help you focus on key areas of coverage for your studies.

ADDITIONAL READING AND RESOURCES

Following are some additional useful resources to help you understand the topics presented in this chapter:

Training and certification resources and information
    `http://www.microsoft.com/learning`
Industry trends related to Microsoft developer tools and technologies
    `http://msdn.microsoft.com/en-us/aa497440`

> **NOTE** *As most developers who focus on the Microsoft tools and platforms are aware, the ultimate resource for news and information on developing on the Microsoft platform is MSDN. MSDN documentation can be installed on your local computer when you install Visual Studio. You can also get the latest developer documentation directly on the web at* `http://msdn.microsoft.com/en-us/`. *Microsoft categorizes developer topics into developer centers that focus on Visual Studio, Windows, Windows Phone, Windows Azure, and Office. All are reachable through the MSDN website.*

EXAM TIPS AND TRICKS

The Review of Key Terms and the Cheat Sheet for each chapter can be printed off to help you study. You can find these files in the ZIP file for each chapter at `www.wrox.com/remtitle.cgi?isbn=1118612094` on the Download Code tab. Due to the nature of the content in this chapter, no Cheat Sheet or Review of Key Terms is included.