C# 2012, Part 3 of 4: Asynchronous Programming

page 1

Meet the expert: Joe Mayo is an author, independent consultant, and instructor specializing in Microsoft .NET and Windows 8 technology. He has several years of software development experience and has worked with .NET since July 2000. Joe has written books and contributes to magazines such as CODE Magazine. He has been an active contributor to the .NET community for years, operating the C# Station Web site, authoring the LINQ to Twitter open source project, and speaking regularly at user groups and code camps. For his community contributions, Microsoft has honored Joe with several Most Valuable Professional (MVP) Awards through the years.

Prerequisites: This course assumes that students understand the C# 3.0 syntax and have a basic understanding of the .NET Framework. No specific experience with Visual Studio 2012 is required.

Runtime: 53:47

Course description: In this course you'll learn about what async is and how it's different from traditional synchronous programming. You'll also learn the benefits of async. Before going into the new async language features, you'll see how the previous async models worked and have a better understanding for what the problems were. This will form a basis of appreciation for how easy the new C# async model is and what the benefits are for you. Next, you'll see the fundamental mechanics of how an async method is defined. You'll learn how to use the new async and await contextual keywords to perform async operations. You'll learn about different types of return values and when you should use each. The course will also cover common techniques of managing async tasks either sequentially or in parallel. You'll learn how to catch exceptions from async code and what type of code to write to ensure that you can catch exceptions. You'll learn how to cancel one or more async operations in predictable ways. Further, you'll see how progress reporting occurs and the ease in which you can integrate it with async methods.

Course outline:

Async Programming
- Introduction
- Where Async Fits In
- Demo: Synchronous Problems
- Previous Async Technologies
- Demo: APM Async Model
- Demo: Event-Based Async Model
- Summary

Writing Async Code
- Introduction
- Anatomy of an Async Method
- Demo: Async Method
- Async Return Values
- Demo: Async Return Values
- About Async Threads
- Awaiting in Sequence
- Demo: Awaiting in Sequence
- Awaiting in Parallel
- Demo: Awaiting in Parallel
- Summary

Except, Cancel, and Progress
- Introduction
- Handling Exceptions
- Demo: Handling Exceptions
- Cancellation
- Demo: Cancellation
- Progress Reporting
- Demo: Progress Reporting
- Summary