

---

## SE030CS – Advanced C#

Duration: 4 Days; Instructor-led

### WHAT YOU WILL LEARN

C# is relatively newer language that inherits many beautiful features of various famous languages. These including language features that cut across some 3GL to 5GL. Therefore, to understand and leverage on all these useful and interesting features will take extra efforts by C# developers.

This course intended to help new C# developers to understand more difficult and advanced features of C# language, so that they can understand and acquired the skill in using these features in their development.

### AUDIENCE

This course is for C# programmers who want to learn and apply more advanced features of C#.

### PREREQUISITES

- The participant should familiar with the basic language constructs in C# and know about object-oriented concepts like polymorphism, inheritance, abstraction, encapsulation, and most importantly, know how to compile or run a C# application in Visual Studio

### METHODOLOGY

This course will be conducted with interactive lectures, PowerPoint presentations, discussion and practical exercises.

### COURSE OUTLINE

#### Module 1 – Attributes

- Overview of Attributes
- Defining Custom Attributes
- Retrieving Attribute Values

#### Module 2 – Reflection, Metadata, And Emitting Objects

- Reflecting on Objects
- Adding Assembly Metadata
- Emitting Objects by Using Builder Classes

#### Module 3 – Delegates

- Comparing a Static Method with an Instance Method
- Single Cast Delegates
- Multicast Delegates
- Variance in Delegates

#### Module 4 – Events

- Creating Custom Events
- Passing Data to an Event Argument
- Using Event Accessors
- Handling Interface Events
- Handling Explicit Interface Events

#### Module 5 – Lambda Expressions

- The Usefulness of Lambda Expressions
- Lambda Expression with (and Without) Parameters
- Types of Lambda Expressions
- Expression-Bodied Members
- Local Variables in a Lambda Expression
- Using Tuples in a Lambda Expression
- Event Subscription with Lambda Expressions

#### Module 6 – Generic Programming

- The Motivation Behind Generics
- A Quick Look into the List Class
- Generic Delegates
- Predicate Delegate
- The Default Keyword in Generics
- Implementing Generic Interface
- Generic Constraints
- Using Covariance and Contravariance
- Covariance with Generic Delegate
- Covariance with Generic Interfaces
- Contravariance with Generic Delegates
- Contravariance with Generic Interface
- Self-Referencing Generic Types

#### Module 7 – Thread Programming

- Foundations in Thread Programming
- Coding Multithreaded Programs in C#
- Using the ThreadStart Delegate
- Using the ParameterizedThreadStart Delegate
- Foreground Thread vs. Background Thread
- Thread Safety
- An Alternative Approach Using the Monitor Class
- Deadlock

#### Module 8 – Asynchronous Programming

- Using a Synchronous Approach
- Using Thread Class
- Using the ThreadPool Class
- Using Lambda Expressions with ThreadPool
- Using the IAsyncResult Pattern
- Using an Event-based Asynchronous Pattern (EAP)
- Understanding Tasks
- Using a Task-based Asynchronous Pattern (TAP)
- Asynchronous Programming with Async / Await

#### Module 9 – LINQ

- C# Features That Support LINQ
- Query Syntax and Method Syntax in LINQ
- Basic LINQ Query Operations
- Data Transformations with LINQ