

Technical Training Solutions



OOP Design Patterns & Best Practices for .NET Development

Technical
Course
Outline

(4 Day Course)

Course Summary

When building modern and often complex software applications, software developers will work with code that will affect an application's performance, its stability/robustness, its complexity, its resource usage and among other things, the application's ability to scale well. This course covers many of the well-known and widely accepted Gang of Four (GoF) software design patterns in use today, how they are implemented within the .NET Framework and how .NET developers can utilize them in their own libraries. Additionally, the course dives into many of the concepts and techniques used in general programming as well as specific to .NET development that are considered best-practices to help build a better quality of code that is readable, performs well, is scalable and easier to support.

Depending on audience, code examples and labs will be taught in VB .NET or C#.

Intended Audience:

While this course does cover some generalized concepts and techniques, these concepts and techniques will be applied to .NET application development. As such, the course is designed for current .NET application developers who have existing knowledge and experience in VB .NET or C#.

Prerequisites:

To ensure the best learning experience for all participants, the following pre-requisites *must be met* in order to participate in this course:

- At least 6 months prior experience with VB .NET or C#.

Course Contents:

Object-Oriented Principles as Design Principle Best Practices

- Abstraction, Encapsulation, Inheritance, Polymorphism
- Cohesion & Coupling
 - Loose vs. Tight Coupling
 - Composition vs. Inheritance
 - Interface-Based Programming

Design Patterns by Category

- Creational
 - Singleton
 - Factory Method
 - Abstract Factory
- Structural
 - Adapter
 - Decorator
 - Facade
 - Proxy
- Behavioral
 - Iterator
 - Observer
 - Strategy
 - Template Method

Microsoft Patterns & Practices (overview)

- .NET Framework Guidelines and Best Practices
 - Type & Member Design Guidelines

Software Guidelines (overview)

- Understanding Guideline Sentiments

Best Practice Opportunities

- Standard Eyebrow Raisers
 - Duplicated Code & Lengthy Methods
 - Coupling & Cohesion
 - XML Strings as Input
 - Out of the Ordinary Syntax
 - Looping
 - Accessing Collection Items
- .NET-Specific
 - Casting vs. Converting vs. Parsing
 - Debug vs. Release Mode
 - XML Code Comments
 - VB .NET and "Option Strict"
 - C# and Optional Curly Braces
 - C# switch Fall-Through and break
 - Generics & Boxing
 - Exceptions & Exception Handling
 - Refactoring
 - StringBuilders
 - Overriding ToString()
 - Garbage Collection/Memory Management
 - ADO .NET Specific
 - ASP .NET Specific