Course: .NET Design Patterns

Module 1: Introduction to OO Design Thinking

- Design Patterns makes use of OOPs concepts - Composition, Aggregation, Inheritance, Encapsulation
- Interface-vs-Implementation
- Dependency Inversion
- Patterns Classification – Creational, Structural, Behavioural

Module 2: Introduction of Creational Pattern

Implementing the list of Creational Patterns:

- Understanding & Implementing the Factory Design Pattern
- Advantages of using Abstract Factory Design Pattern
- Overview of Builder Design Pattern & how it is different from Abstract factory Design Pattern
- Cloning of an object using Prototype Design Pattern
- Understanding & Implementing the Singleton

Demo: Implementing the Factory, Abstract Factory, Builder, Prototype & Singleton Design Pattern

Module 3: Introduction of Structural Pattern

Implementing the list of Structural Patterns:

- Overview of Adapter Design Pattern & Understand when to use it
- Introduction of Bridge Design Pattern
- Compose objects using Composite Design Pattern
- Add additional functionalities using Decorator Design Pattern
- Create object without constructor call using Façade Design Pattern
- Implementing Flyweight Design Pattern
- Understanding Proxy Design Pattern

Demo: Implementing the Adapter, Composite, Decorator, Faced, Fly weight & Proxy Pattern

Module 4: Introduction of Behavioural Pattern

Implementing the list of Behavioural Pattern:

- Overview of Chain of Responsibility Design Pattern
- Understanding the Command Design Pattern
- Usage of Interpreter Design Pattern
- Introduction of Iterator Design Pattern
- Allow objects to communicate using Mediator Design Pattern
- Overview of Memento Design Pattern
- Understanding the Observer Design Pattern
- Alter the behaviour of an object using State Design Pattern
- Overview of Strategy Design Pattern
- Overview of Visitor Design Pattern
- Implementing the Template Method Design Pattern

Demo: Chain of Responsibility, Mediator, Observer, Template Method