TEKNOLOGISK INSTITUT

Curriculum for 89076: ”Gang of Four Design Patterns in C#”

Contents

Design Patterns are well-established and reusable solutions to frequently recurring problems within object-oriented programming. In the legendary book [1] published in 1994, the authors (affectionately dubbed “Gang of Four”) describe and categorize 23 such patterns. These design patterns are still relevant even today and many more patterns have emerged in the community since then.

As these patterns were originally formulated in a programming language-independent manner – more than two decades before the present, modern version of C# – these design patterns can be presented much more current, relevant, modern, and elegant formulations in C# 7.x. To emphasize but a few examples of that , the Iterator and Observer design patterns are now directly embedded in the C# language. Similarly, these days the Builder pattern is most often used for creating Fluent APIs in .NET.

This course consists of both module presentations and practical exercises.

Prerequisites

This course is aimed at C# developers with an interest in modern approaches to solving concrete programming problems. Prior knowledge of C# and Visual Studio is required – in particular “classic” object-oriented programming and interfaces. The course is at the Intermediate level and is ideal for developers who have some experience in C#, but lack the experience of structuring intermediate- or large-scale .NET projects.

Agenda

The course has a duration of 4 course days following the agenda outlined below.

Day 1 (Creational Patterns)	Introduction Module 01: “What are Design Patterns?” Module 02: “Abstract Factory” Module 03: “Builder” Module 04: “Factory Method” Module 05: “Prototype” Module 06: “Singleton”
Day 2 (Structural Patterns)	Module 07: “Adapter” Module 08: “Bridge” Module 09: “Composite” Module 10: “Decorator” Module 11: “Façade” Module 12: “Flyweight”
Day 3 (Behavioral Patterns)	Module 13: “Proxy” Module 14: “Iterator” Module 15: “Chains of Responsibility” Module 16: “Template Method” Module 17: “Strategy”

	Module 18: “Memento”
Day 4 (Behavioral Patterns)	Module 19: “Command” Module 20: “State” Module 21: “Interpreter” Module 22: “Visitor” Module 23: “Observer” Module 24: “Mediator” Conclusion

Referencer

[1] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides (a.k.a. “Gang of Four ☺)
Design Patterns: Elements of Reusable Object-Oriented Software
ISBN-13: 978-0-201-63361-0
Addison-Wesley (1994)

Module 02: Abstract Factory

- [Gamma et al., Chapter 3, pp. 87 – 95]

Module 03: Builder

- [Gamma et al., Chapter 3, pp. 97 – 106]

Module 04: Factory Method

- [Gamma et al., Chapter 3, pp. 107 – 116]

Module 05: Prototype

- [Gamma et al., Chapter 3, pp. 117 – 126]

Module 06: Singleton

- [Gamma et al., Chapter 3, pp. 127 – 134]
- Jon Skeet: http://csharpindepth.com/Articles/General/Singleton.aspx
- Lazy<T> class: https://msdn.microsoft.com/en-us/library/dd642331.aspx

Module 07: Adapter

- [Gamma et al., Chapter 4, pp. 139 – 150]

Module 08: Bridge

- [Gamma et al., Chapter 4, pp. 151 – 162]

Module 09: Composite

- [Gamma et al., Chapter 4, pp. 163 – 174]

Module 10: Decorator

- [Gamma et al., Chapter 4, pp. 175 – 184]

Module 11: Façade

- [Gamma et al., Chapter 4, pp. 185 – 194]

Module 12: Flyweight

- [Gamma et al., Chapter 4, pp. 195 – 206]

Module 13: Proxy

- [Gamma et al., Chapter 4, pp. 207 – 218]

Module 14: Iterator

- [Gamma et al., Chapter 5, pp. 257 – 272]

Module 15: Chain of Responsibility

- [Gamma et al., Chapter 5, pp. 223 – 232]

Module 16: Template Method

- [Gamma et al., Chapter 5, pp. 325 – 330]

Module 17: Strategy

- [Gamma et al., Chapter 5, pp. 315 – 324]

Module 18: Memento

- [Gamma et al., Chapter 5, pp. 283 – 292]

Module 19: Command

- [Gamma et al., Chapter 5, pp. 233 – 242]

Module 20: State

- [Gamma et al., Chapter 5, pp. 305 – 314]

Module 21: Interpreter

- [Gamma et al., Chapter 5, pp. 243 – 256]

Module 22: Visitor

- [Gamma et al., Chapter 5, pp. 331 – 344]

Module 23: Observer

- [Gamma et al., Chapter 5, pp. 293 – 304]

Module 24: Mediator

- [Gamma et al., Chapter 5, pp. 273 – 282]