

SUMMARY

We teach high-quality object-oriented design using design patterns in our course, *Agile Software Development with Design Patterns*. We also teach good design as part of our course, *Sustainable Test-Driven Development*. People might consider these two approaches to be in opposition to each other having come from different eras in the history of our industry. We disagree. We believe there is tremendous synergy between them and in fact find great value in teaching them together through our course, *Emergent Design*.

DESCRIPTION

Net Objectives has been teaching courses on design patterns, agile software development, and sustainable Test-Driven Development for many years. We believe in the value they bring to the practice of software development. Emergent Design shows how to incorporate these complementary approaches powerfully for more effective design and development.

DESIGN PATTERNS

The traditional view of the design patterns movement is that they represent reusable design elements; in reality, they deliver value at a much *deeper* level. There is a reason that certain particular patterns in design have been singled out, named, and focused upon. When you study the patterns, you are in fact studying the DNA of good design: how to create flexibility, extensibility, and maintainability without creating excessive complexity and over-design. Not every good design is a pattern but every design should reflect the level of quality of a pattern. Patterns provide a deep and rich language that improves the fidelity of communication and collaboration across the team. Also, patterns are not limited to design; there are patterns in every part of development including analysis, implementation, and testing.

TEST-DRIVEN DEVELOPMENT

Test-Driven Development (TDD) is a very powerful tool for developers to create high-quality software solutions. By using tests as both analysis and design tools, we can increase the velocity of work and improve the confidence of the team, allowing for a more aggressive pursuit of maximum business value. However, it is easy to create sustainability problems as the suite of tests becomes large.

If the team is not properly trained then the test themselves can become a maintenance burden especially in an Agile environment where new requirements are constantly flowing into the work. This is such a common problem that it has led many to conclude that TDD is not inherently sustainable. We disagree. We believe sustainability comes from what you think TDD is, how you do it, and how you integrate it with the development process overall.

EMERGENT DESIGN

We offer high-value, stand-alone courses in both design patterns and test-driven development. But the greatest value comes by combining them into a single course offering because we can show the tremendous synergy between the two skills. Bad designs are hard to test. By moving tests to the fore, we can avoid design mistakes that might otherwise be subtle and easy-to-miss. Patterns give us wisdom about all aspects of our work including how to test various design elements that might otherwise seem challenging. Emergent, or “just-in-time,” design helps us to reduce waste and risk in our process.

COURSE OBJECTIVES

Emergent Design is conducted over five days. In the first three days, we focus on the qualities, principles, and practices of good object-oriented analysis and design using the patterns as guidance. Exercises are done at tables and whiteboards. In the final two days, we teach the sustainable process of TDD through lecture and hands-on coding exercises. Through this process the students learn how TDD leads to patterns, how patterns lead to testability, and how they, together, empower the team to create high-quality software in an efficient and sustainable manner.

CONTACT US

info@netobjectives.com
1.888.LEAN-244 (1.888.532.6244)

LEARN MORE

www.NetObjectives.com
portal.NetObjectives.com

LEARNING OBJECTIVES

In this course, you will understand:

- What patterns are
- How patterns lead to testability
- How TDD leads to patterns
- How patterns and TDD work together to empower teams
- How patterns and TDD combine to realize high-quality software efficiently and sustainably

COURSE OUTLINE

Days 1-3: Design patterns

- Qualities, principles, and practices of good object-oriented analysis and design using design patterns
- Exercises in design

Days 4-5: Test-Driven Development

- Sustainable approach to TDD
- How TDD leads to patterns
- Coding exercises

LEVEL

Foundational

TARGET AUDIENCE

Experienced developers, testers, designers, architects, technical leads, and those who manage development teams. They should have basic object-oriented experience.

Lesser experienced developers will find value and should be included if part of an experienced team. We can provide pre-course material to bring them up to speed.

ATTENDEE MATERIALS

Course materials will be provided at the start of the class.

ROOM SETUP AND EQUIPMENT

- One computer per two students.
- Students usually sit at tables, four students per table.
- Flip chart and whiteboard for the instructor.
- A projector with screen.

PREREQUISITES

Knowledge of an object-oriented language (Java, C#, C++, etc.) is important in order to understand the various implementation options we investigate. Net Objectives has online assessments for students who are uncertain about their knowledge level, and online training tools that can help prepare you to get the most out of the course. In particular, the participant must be familiar with inheritance, polymorphism, encapsulation, abstract classes and composition.

For the final two days, students should have computers with a working IDE installed on them.

COURSE LENGTH

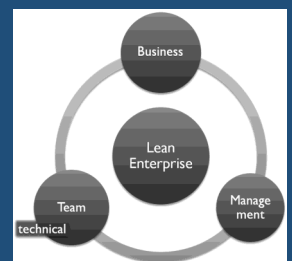
Five days

MAXIMUM NUMBER IN CLASS

24

NET OBJECTIVES

We are committed to delivering the principles, practices, and perspectives that businesses must know in order to maximize their return on their technology solution and software development efforts. We combine our experience and a time proven approach based on lean thinking to continuously extend the capability of what is possible in creating effective technology delivery organizations (IT or product). We provide these learned methods to our clients to assist them in achieving their goals and in assisting them in making their organizations more successful.



Full course descriptions may be found at
www.NetObjectives.com/training

Lean • Agile • Kanban Patterns •
TDD • ATDD • Assessments •
Consulting Training • Coaching