

Crafting Code

Duration: 2 days



www.codurance.com
hello@codurance.com

About

- Hands-on course to help developers write well crafted code
 - Develop techniques that you can apply immediately to real-world projects
 - Well crafted code is
 - Clean
 - Testable
 - Maintainable
 - Expresses the business

Why attend?

- Learn **the Software Craftsmanship** attitude to development
- Write **clean code** that is easy to understand and maintain
- Become proficient in **TDD**
- Use TDD to **focus your code** on business requirements
- Understand **design principles** that lead to clean code
- Learn how to **avoid over engineering**
- Gain confidence in **working with legacy code**

Content

- **TDD lifecycle** and Outside-in TDD
- Writing unit tests that **express intent**, not implementation
- Using unit tests as a tool to drive **good design**
- **Expressive code**
- **Testing and refactoring Legacy Code**
- Introduction to SOLID Principles and Domain-Driven Design

Audience

- Software developers that:
 - are familiar with at least one Object-Oriented language
 - are able to understand Java or C#
 - can write and execute unit tests using a framework (such as JUnit, NUnit, etc.)
 - have a basic understanding of mock objects

Prerequisites

- Developers must:
 - bring their own laptops
 - have a development environment consisting of:
 - their favorite Object-Oriented language
 - a unit test framework
 - a mocking library
 - be able to create projects, build source code and run test cases in their development environment