

Test-Driven Development Using Visual Studio

Course Summary

Description

An emerging software engineering paradigm is test-driven development, where tests are written early and testing is performed continuously during the development process. Problems are discovered early and corrected when they are found. This one-day course for developers explains the methodology of test-driven development and the use of the unit testing framework that comes with Visual Studio 2008 Team System.

The course is practical, with many example programs and tests written in C#, including a cumulative case study. The goal is to quickly bring you up to speed in doing unit testing in your .NET development projects. The student will receive a comprehensive set of materials, including course notes and all the programming examples.

Objectives

At the end of this course, students will be able to:

- Understand the principles of test-driven development
- Acquire fluency in developing tests using the Visual Studio unit testing framework
- Efficiently exercise test suites using Visual Studio and MSTest

Topics

- Test-Driven Development
- Visual Studio Unit Testing Fundamentals
- More about Unit Testing Framework

Prerequisites

The student should have a basic knowledge of the .NET Framework and experience programming in C# with Visual Studio.

Duration

One day

Test-Driven Development Using Visual Studio

Course Outline

I. Test-Driven Development

- D. What Is Test-Driven Development (TDD)?
- E. Functional Tests / Customer Tests
- F. Unit Tests / Programmer Tests
- G. Test Automation
- H. Simple Design
- I. Refactoring
- J. A Visual Studio Test Drive
- K. TDD with Legacy Code

II. Visual Studio Unit Testing Fundamentals

- D. Structure of Unit Tests
- E. Unit Testing Framework
- F. Assertions
- G. Test Cases
- H. Test Classes
- I. Test Runners
- J. Ignoring Tests
- K. Initialization and Cleanup

III. More about Unit Testing Framework

- D. Expected Exceptions
- E. Custom Asserts
- F. Test Lists
- G. Debugging Unit Tests
- H. MSTest
- I. Refactoring