



**CTREC
HILTON**
IT ACADEMY

BUILDING TRAINING SOLUTIONS
FOR THE IT WORLD

Introduction to Programming Microsoft .NET Applications with Microsoft Visual Studio 2005 and 2008

Days:	5
Format:	Instructor-Led
Class Code	4994
Certification Exams:	None
Certification Track:	None

Recommended Course Sequence

Knowledge of prerequisites
noted below.

*Course content is subject to change
without notice.*

Course Description:

This five-day instructor-led course enables introductory-level developers who are not familiar with the Microsoft .NET Framework to gain familiarity with the Visual Studio 2005 or 2008 development environment. Students will also learn basic skills using either Microsoft Visual Basic or Microsoft Visual C# as a programming language.

Target Student:

The target audience for this course includes both novice programmers who have a minimum of three months' programming experience and intermediate-level programmers who are otherwise new to .NET Framework development, and want to learn how to use Visual Basic or C#.

Prerequisites:

Before attending this course, students must have:

- Exposure to developing applications in either a graphical or a non-graphical environment.
- Ability to understand and apply the basics of structured programming, including concepts such as flow control, variables, parameters, and function calls.

Delivery Method: Instructor led, group-paced, classroom-delivery learning model with structured hands-on activities.

At Course Completion:

Upon successful completion of this course, students will be able to:








- Describe the key features of the .NET Framework
- Create a simple Windows Forms application.
- Explain programming fundamentals.
- Create and use data types and variables.
- Control program execution by using conditional statements and loops.
- Explain the fundamentals of object-oriented programming.
- Create simple object-oriented applications.
- Develop the user interface in a windows application.
- Validate user input on a Windows form.
- Implement debugging and exception handling in a .NET application.
- Access data in a .NET application.
- Create simple Web applications and XML Web services.
- Explain the key features of the .NET Framework version 3.0 technologies.
- Test and deploy Microsoft .NET Framework applications.

Introduction to Programming Microsoft .NET Applications with Microsoft Visual Studio 2005 and 2008

Course Outline







Module 1: Getting Started

This module introduces the .NET Framework and the software development life cycle. It also describes the key features of Visual Studio 2005 and 2008.

Lessons	
	Introduction to Microsoft .NET and the .NET Framework.
	Introduction to the Software Development Life Cycle.
	Exploring Visual Studio 2005 and 2008.
	Converting Disks.
	Creating Volumes.
	Creating Fault-Tolerant Volumes.
Lab	
	Working in the Development Environment.







Module 2: Creating a Simple Windows Forms Application

This module explains how to create a Windows Forms application, how to add controls to a form, and how to compile and run the application.

Lessons	
	Creating a Windows Forms Project.
	Adding Controls to a Windows Forms Project.
	Compiling and Running a Windows Forms Project.
Lab	
	Creating a Windows Forms Application.
	Adding Controls to the Main Form.
	Compiling and Testing the Application.

Module 3: Programming Fundamentals

This module explains important programming concepts and terminology. It also covers the main elements of a program and explains how to create and work with items such as functions, properties, and methods. Finally, this module provides guidelines on areas such as naming conventions and code documentation.

Lessons	
	Understanding Programming Concepts.
	Defining Program Structure and Flow.
	Styling and Writing Code.
Lab	
	Displaying the Current Date on a Form.
	Adding a New Form to the Application.
	Adding Controls to the New Form.

Introduction to Programming Microsoft .NET Applications with Microsoft Visual Studio 2005 and 2008

Module 4: Data Types and Variables

This module introduces data types, variables, and constants and explains how to use them. It also explains how to use collections and data type conversion.

Lessons	
<input type="checkbox"/>	Introduction to Data Types.
<input type="checkbox"/>	Defining and Using Variables.
<input type="checkbox"/>	Defining and Using Collections.
<input type="checkbox"/>	Converting Data Types.
Lab	
<input type="checkbox"/>	Implementing Variables and Constants.
<input type="checkbox"/>	Implementing Arrays and Enumerations.

Module 5: Controlling Program Execution

This module describes how to control program execution by writing expressions, conditional statements, and iteration statements.

Lessons	
<input type="checkbox"/>	Writing Expressions.
<input type="checkbox"/>	Creating Conditional Statements.
<input type="checkbox"/>	Creating Iteration Statements.
Lab	
<input type="checkbox"/>	Checking User Input.
<input type="checkbox"/>	Enabling and Disabling Controls.

Module 6: Fundamentals of Object-Oriented Programming

This module introduces students to the concepts of object-oriented programming, defines important terminology, and shows the syntax for defining classes and creating instances.

Lessons	
<input type="checkbox"/>	Introduction to Object-Oriented Programming.
<input type="checkbox"/>	Defining a Class.
<input type="checkbox"/>	Creating a Class Instance.
Lab	
<input type="checkbox"/>	Creating a SalesPerson Class.
<input type="checkbox"/>	Creating and Using a SalesPerson Object.

Introduction to Programming Microsoft .NET Applications with Microsoft Visual Studio 2005 and 2008

Module 7: Creating Object-Oriented Applications

This module describes how to design classes by using the Class Designer tool in Visual Studio, and also describes how to use inheritance and interfaces.

Lessons	
<input type="checkbox"/>	Designing Classes with the Class Designer Tool.
<input type="checkbox"/>	Implementing Inheritance.
<input type="checkbox"/>	Defining and Implementing Interfaces.
Lab	
<input type="checkbox"/>	Creating a Base Class.
<input type="checkbox"/>	Creating Derived Classes.

Module 8: Building a User Interface

This module explains how to develop an application by using features such as modal and modeless forms, menus, toolbars, status bars, tool tips, and the HelpProvider control Instances.

Lessons	
<input type="checkbox"/>	Managing Forms and Dialog Boxes.
<input type="checkbox"/>	Creating Menus and Toolbars.
<input type="checkbox"/>	Providing User Assistance.
Lab	
<input type="checkbox"/>	Adding a Menu and a Toolbar to an Application.
<input type="checkbox"/>	Adding a Status Bar and Tooltips to an Application.

Module 9: Validating User Input

This module explains how to restrict user input on a form, and how to use field-level and form-level validation.

Lessons	
<input type="checkbox"/>	Restricting User Input.
<input type="checkbox"/>	Implementing Field-Level Validation.
<input type="checkbox"/>	Implementing Form-Level Validation.
Lab	
<input type="checkbox"/>	Adding an ErrorProvider Component to a Form.
<input type="checkbox"/>	Providing Visual Cues to the User by Enabling an OK Button.

Introduction to Programming Microsoft .NET Applications with Microsoft Visual Studio 2005 and 2008

Module 10: Debugging and Exception Handling

This module introduces students to the types of errors that can occur in an application, and describes how to use a combination of debugging and exception handling to detect and diagnose these errors.

Lessons
<ul style="list-style-type: none">■ Types of Errors.■ Debugging Applications.■ Handling Exceptions in Applications.
Lab
<ul style="list-style-type: none">■ Detecting Logic Errors.■ Handling Run-Time Errors.

Module 11: Accessing Data

This module introduces students to data access in .NET Framework applications, and shows how to access data both by using the Visual Studio integrated development environment (IDE) and by writing code.

Lessons
<ul style="list-style-type: none">■ Overview of Data Access.■ Accessing Data by Using the Visual Studio Integrated Development Environment.■ Programmatic Access to Data.
Lab
<ul style="list-style-type: none">■ Displaying Data by Using a DataGridView Control.■ Access Data Programmatically by Using ADO.NET.

Module 12: Creating Web Applications and XML Web Services







This module introduces students to ASP.NET, and describes how to create simple Web applications and XML Web services.

Lessons
<ul style="list-style-type: none">■ Creating Web Applications.■ Creating and Using XML Web Services.
Lab
<ul style="list-style-type: none">■ Creating a Web Application.■ Creating and Using an XML Web Service.

Introduction to Programming Microsoft .NET Applications with Microsoft Visual Studio 2005 and 2008







Module 13: Exploring .NET Framework 3.0 Technologies

This module introduces the new .NET Framework 3.0 technologies and explains how to create a Windows Presentation Foundation application and a Windows Communication Foundation service.

Lessons	
	Introduction to the .NET Framework 3.0 Technologies
	Introduction to Windows Presentation Foundation.
	Introduction to Windows Communication Foundation.
Lab	
	Building a Windows Presentation Foundation Application.
	Building a Windows Communication Foundation Service.
	Accessing a Windows Communication Foundation Service from a Windows Presentation Foundation Client

Module 14: Testing and Deploying Microsoft .NET Framework Applications

This module provides an overview of software testing and explains how to use the Object Test Bench (OTB). It also explains how to deploy Microsoft .NET Framework applications by using both Windows Installer and ClickOnce.

Lessons	
	Overview of Testing.
	Creating Object Test Bench Objects.
	Deploying Microsoft .NET Framework Applications.
Lab	
	Testing an Application.
	Deploying an Application by Using ClickOnce.
	Deploying an Application by Using Windows Installer