Programming in C#

Description

This training course teaches developers the programming skills that are required for developers to create Windows applications using the Visual C# language. During their five days in the classroom students review the basics of Visual C# program structure, language syntax, and implementation details, and then consolidate their knowledge throughout the week as they build an application that incorporates several features of the .NET Core 6.0.

- If you want a course .NET Framework 4.7, please contact us directly

Classroom Registration Price (CHF)
3800
Virtual Classroom Registration Price (CHF)
3550
Course Content
Module 1: Review of Visual C# Syntax

- Lesson 1: Overview of Writing Application by Using Visual C#
- Lesson 2: Data Types, Operators, and Expressions
- Lesson 3: Visual C# Programming Language Constructs

Module 2: Creating Methods, Handling Exceptions, and Monitoring Applications

- Lesson 1: Creating and Invoking Methods
- Lesson 2: Creating Overloaded Methods and Using Optional and Output Parameters
- Lesson 3: Handling Exceptions
- Lesson 4: Monitoring Applications

Module 3: Basic types and constructs of Visual C#

- Lesson 1: Implementing Structs and Enums
- Lesson 2: Organizing Data into Collections
- Lesson 3: Handling Events

Module 4: Creating Classes and Implementing Type-Safe Collections

- Lesson 1: Creating Classes
- Lesson 2: Defining and Implementing Interfaces
- Lesson 3: Implementing Type-Safe Collections

Module 5: Creating a Class Hierarchy by Using Inheritance

- Lesson 1: Creating Class Hierarchies
- Lesson 2: Extending .NET Framework Classes

Module 6: Reading and Writing Local Data

- Lesson 1: Reading and Writing Files

- Lesson 2: Serializing and Deserializing Data
- Lesson 3: Performing I/O by Using Streams

Module 7: Accessing a Database

- Lesson 1: Creating and Using Entity Data Models
- Lesson 2: Querying Data by Using LINQ

Module 8: Accessing Remote Data

- Lesson 1: Accessing Data Across the Web
- Lesson 2: Accessing Data by Using OData Connected Services

Module 9: Designing the User Interface for a Graphical Application

- Lesson 1: Using XAML to Design a User Interface
- Lesson 2: Binding Controls to Data

Module 10: Improving Application Performance and Responsiveness

- Lesson 1: Implementing Multitasking
- Lesson 2: Performing Operations Asynchronously
- Lesson 3: Synchronizing Concurrent Access to Data

Module 11: Integrating with Unmanaged Code

- Lesson 1: Creating and Using Dynamic Objects
- Lesson 2: Managing the Lifetime of Objects and Controlling Unmanaged Resources

Module 12: Creating Reusable Types and Assemblies

- Lesson 1: Examining Object Metadata
- Lesson 2: Creating and Using Custom Attributes
- Lesson 3: Generating Managed Code
- Lesson 4: Versioning, Signing, and Deploying Assemblies

Module 13: Encrypting and Decrypting Data

- Lesson 1: Implementing Symmetric Encryption
- Lesson 2: Implementing Asymmetric Encryption

Lab / Exercises
Module 1: Review of Visual C# Syntax Module 2: Creating Methods, Handling Exceptions, and Monitoring Applications Module 3: Basic types and constructs of Visual C# Module 4: Creating Classes and Implementing Type-Safe Collections Module 5: Creating a Class Hierarchy by Using Inheritance Module 6: Reading and Writing Local Data Module 7: Accessing a Database Module 8: Accessing Remote Data Module 9: Designing the User Interface for a Graphical Application Module 10: Improving Application Performance and Responsiveness Module 11: Integrating with Unmanaged Code Module 12: Creating Reusable Types and Assemblies Module 13: Encrypting and Decrypting Data
Documentation

- Digital courseware included

Participant profiles

- Experienced developers

Prerequisites

- Know how to name, declare, initialize and assign values ??to variables
- Master arithmetic, relational and logical operators
- Master the keywords of the C # language
- Master the if and for statement and also the Visual Studio IDE
- Know how to connect and sort SQL Server data in a loop

Objectives

- Describe the core syntax and features of Visual C#
- Create methods, handle exceptions, and describe the monitoring requirements of large-scale applications
- Implement the basic structure and essential elements of a typical desktop application

Niveau
Intermédiaire
Duration (in Days)
5
Reference
CSHARP