uCertify

Course Outline

C# 10.0 All-in-one

03 Jul 2024

1. Course Objective

2. Exercises, Quizzes, Flashcards & Glossary

   Number of Questions

3. Expert Instructor-Led Training

4. ADA Compliant & JAWS Compatible Platform

5. State of the Art Educator Tools

6. Award Winning Learning Platform (LMS)

7. Chapter & Lessons

   Syllabus

   Chapter 1: Introduction

   Chapter 2: Creating Your First C# Console Application

   Chapter 3: Living with Variability — Declaring Value-Type Variables

   Chapter 4: Pulling Strings

   Chapter 5: Smooth Operators

   Chapter 6: Getting into the Program Flow

   Chapter 7: Lining Up Your Ducks with Collections

   Chapter 8: Stepping through Collections

   Chapter 9: Buying Generic

   Chapter 10: Some Exceptional Exceptions

   Chapter 11: Creating Lists of Items with Enumerations

   Chapter 12: Showing Some Class

   Chapter 13: We Have Our Methods

   Chapter 14: Let Me Say This about this

   Chapter 15: Holding a Class Responsible

   Chapter 16: Inheritance: Is That All I Get?

   Chapter 17: Poly-what-ism?

   Chapter 18: Interfacing with the Interface

   Chapter 19: Delegating Those Important Events

1. 📖 Course Objective

The C# 10.0 All-in-one Course is designed to take you from a beginner to an advanced C# developer. This course covers everything you need to know about C# programming, starting from the basics to more advanced topics. You'll learn the core principles of the C# language, explore Object-Oriented Programming (OOP), discover advanced language features, and see how to apply these skills in web development using ASP.NET Core.

2. ⏱ Quiz

Quizzes test your knowledge on the topics of the exam when you go through the course material. There is no limit to the number of times you can attempt it.

300
QUIZ

3. ⚡ flashcards

Flashcards are effective memory-aiding tools that help you learn complex topics easily. The flashcard will help you in memorizing definitions, terminologies, key concepts, and more. There is no limit to the number of times learners can attempt these. Flashcards help master the key concepts.

180
FLASHCARDS

4. 📖 Glossary of terms

uCertify provides detailed explanations of concepts relevant to the course through Glossary. It contains a list of frequently used terminologies along with its detailed explanation. Glossary defines the key terms.

180

GLOSSARY OF TERMS
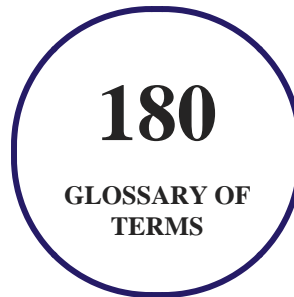
5. 👤 Expert Instructor-Led Training

uCertify uses the content from the finest publishers and only the IT industry's finest instructors. They have a minimum of 15 years real-world experience and are subject matter experts in their fields. Unlike a live class, you can study at your own pace. This creates a personal learning experience and gives you all the benefit of hands-on training with the flexibility of doing it around your schedule 24/7.

6. ⊛ ADA Compliant & JAWS Compatible Platform

uCertify course and labs are ADA (Americans with Disability Act) compliant. It is now more accessible to students with features such as:

- Change the font, size, and color of the content of the course
- Text-to-speech, reads the text into spoken words
- Interactive videos, how-tos videos come with transcripts and voice-over
- Interactive transcripts, each word is clickable. Students can clip a specific part of the video by clicking on a word or a portion of the text.

JAWS (Job Access with Speech) is a computer screen reader program for Microsoft Windows that reads the screen either with a text-to-speech output or by a Refreshable Braille display. Student can easily navigate uCertify course using JAWS shortcut keys.

7. State of the Art Educator Tools

uCertify knows the importance of instructors and provide tools to help them do their job effectively. Instructors are able to clone and customize course. Do ability grouping. Create sections. Design grade scale and grade formula. Create and schedule assessments. Educators can also move a student from self-paced to mentor-guided to instructor-led mode in three clicks.

8. Award Winning Learning Platform (LMS)

uCertify has developed an award winning, highly interactive yet simple to use platform. The SIIA CODiE Awards is the only peer-reviewed program to showcase business and education technology's finest products and services. Since 1986, thousands of products, services and solutions have been recognized for achieving excellence. uCertify has won CODiE awards consecutively for last 7 years:

- **2014**
  1. Best Postsecondary Learning Solution

- **2015**
  1. Best Education Solution
  2. Best Virtual Learning Solution
  3. Best Student Assessment Solution
  4. Best Postsecondary Learning Solution
  5. Best Career and Workforce Readiness Solution
  6. Best Instructional Solution in Other Curriculum Areas
  7. Best Corporate Learning/Workforce Development Solution

- **2016**
  1. Best Virtual Learning Solution
  2. Best Education Cloud-based Solution
  3. Best College and Career Readiness Solution
  4. Best Corporate / Workforce Learning Solution
  5. Best Postsecondary Learning Content Solution

6. Best Postsecondary LMS or Learning Platform
7. Best Learning Relationship Management Solution

- **2017**
    1. Best Overall Education Solution
    2. Best Student Assessment Solution
    3. Best Corporate/Workforce Learning Solution
    4. Best Higher Education LMS or Learning Platform

- **2018**
    1. Best Higher Education LMS or Learning Platform
    2. Best Instructional Solution in Other Curriculum Areas
    3. Best Learning Relationship Management Solution

- **2019**
    1. Best Virtual Learning Solution
    2. Best Content Authoring Development or Curation Solution
    3. Best Higher Education Learning Management Solution (LMS)

- **2020**
    1. Best College and Career Readiness Solution
    2. Best Cross-Curricular Solution
    3. Best Virtual Learning Solution

9. ⚙ Chapter & Lessons

uCertify brings these textbooks to life. It is full of interactive activities that keeps the learner engaged. uCertify brings all available learning resources for a topic in one place so that the learner can efficiently learn without going to multiple places. Challenge questions are also embedded in the chapters so learners can attempt those while they are learning about that particular topic. This helps them grasp the concepts better because they can go over it again right away which improves learning.

Learners can do Flashcards, Exercises, Quizzes and Labs related to each chapter. At the end of every lesson, uCertify courses guide the learners on the path they should follow.

Syllabus

- What If I Want to Switch Case?

- Looping through a String

- Searching Strings

- Getting Input from Users in Console Applications

- Controlling Output Manually

- Formatting Your Strings Precisely

- StringBuilder: Manipulating Strings More Efficiently

Chapter 5: Smooth Operators

- Performing Arithmetic

- Performing Logical Comparisons — Is That Logical?

- Matching Expression Types at TrackDownAMate.com

Chapter 6: Getting into the Program Flow

- Branching Out with if and switch

- Here We Go Loop-the-Loop

- Looping a Specified Number of Times with for

Chapter 7: Lining Up Your Ducks with Collections

- Writing a New Prescription: Generics

- Classy Generics: Writing Your Own

- Understanding Variance in Generics

Chapter 10: Some Exceptional Exceptions

- Using an Exceptional Error-Reporting Mechanism

- Can I Get an Exceptional Example?

- Working with Custom Exceptions

- Planning Your Exception-Handling Strategy

- Grabbing Your Last Chance to Catch an Exception

- Throwing Expressions

Chapter 11: Creating Lists of Items with Enumerations

- Seeing Enumerations in the Real World

- Working with Enumerations

- Creating Enumerated Flags

- Defining Enumerated Switches

- Working with Enumeration Methods

Chapter 12: Showing Some Class

- A Quick Overview of Object-Oriented Programming

- Defining a Class and an Object

- Accessing the Members of an Object

- Working with Object-Based Code

- Discriminating between Objects

- Can You Give Me References?

- Classes That Contain Classes Are the Happiest Classes in the World

- Generating Static in Class Members

- Defining const and readonly Data Members

Chapter 13: We Have Our Methods

- Defining and Using a Method

- Method Examples for Your Files

- Having Arguments with Methods

- Using the Call-by-Reference Feature

- Defining a Method with No Return Value

- Returning Multiple Values Using Tuples

Chapter 14: Let Me Say This about this

- Passing an Object to a Method

- Comparing Static and Instance Methods

- Accessing the Current Object

- Using Local Functions

Chapter 15: Holding a Class Responsible

- Restricting Access to Class Members

- Why You Should Worry about Access Control

- Defining Class Properties

- Using Target Typing for Your Convenience

- Dealing with Covariant Return Types

- Getting Your Objects Off to a Good Start — Constructors

- Using Expression-Bodied Members

Chapter 16: Inheritance: Is That All I Get?

- Why You Need Inheritance

- Inheriting from a BankAccount Class (a More Complex Example)

- Using Interfaces to Manage Change in Object-Oriented Programs

Chapter 19: Delegating Those Important Events

- E.T., Phone Home — The Callback Problem

- Defining a Delegate

- Pass Me the Code, Please — Examples

- A More Real-World Example

- Shh! Keep It Quiet — Anonymous Methods

- Stuff Happens — C# Events

Chapter 20: Can I Use Your Namespace in the Library?

- Dividing a Single Program into Multiple Source Files

- Working with Global using Statements

- Dividing a Single Program into Multiple Assemblies

- Putting Your Classes into Class Libraries

- Going Beyond Public and Private: More Access Keywords

- Putting Classes into Namespaces

- Working with Partial Methods

Chapter 21: Improving Productivity with Named and Optional Parameters

- Exploring Optional Parameters

- Looking at Named Parameters

- Using Alternative Methods to Return Values

- Dealing with null Parameters

Chapter 22: Interacting with Structures

- Comparing Structures to Classes

- Creating Structures

- Working with Read-only Structures

- Working with Reference Structures

- Using Structures as Records

- Using the New Record Type

Chapter 23: Writing Secure Code

- Designing Secure Software

- Building Secure Windows Applications

- Using System.Security

Chapter 24: Accessing Data

- Getting to Know System.Data

- How the Data Classes Fit into the Framework

- Getting to Your Data

- Using the System.Data Namespace

Chapter 25: Fishing the File Stream

- Going Where the Fish Are: The File Stream

- StreamWriting for Old Walter

- Pulling Them Out of the Stream: Using StreamReader

- More Readers and Writers

- Exploring More Streams than Lewis and Clark

Chapter 26: Accessing the Internet

- Getting to Know System.Net

- How Net Classes Fit into the Framework

- Using the System.Net Namespace

Chapter 27: Creating Images

- Coding in the Code Editor

- Using the Tools of the Trade

- Using the Debugger as an Aid to Learning

Chapter 31: Customizing Visual Studio

- Setting Options

- Creating Your Own Templates

Chapter 32: Introducing WPF

- Understanding What WPF Can Do

- Introducing XAML

- Diving In! Creating Your First WPF Application

- Whatever XAML Can Do, C# Can Do Better!

Chapter 33: Understanding the Basics of WPF

- Using WPF to Lay Out Your Application

- Arranging Elements with Layout Panels

- Exploring Common XAML Controls

Chapter 34: Data Binding in WPF

- Getting to Know Dependency Properties

- Exploring the Binding Modes

- Investigating the Binding Object

- Editing, Validating, Converting, and Visualizing Your Data

- Finding Out More about WPF Data Binding

Chapter 35: Practical WPF

- Commanding Attention

- Using Built-In Commands

- Using Custom Commands

- Using Routed Commands

Chapter 36: Programming for Windows 10 and Above

- What is the Universal Windows Platform (UWP)?

- Devices Supported by the UWP

- Creating Your Own UWP App

- Working with .NET Core Applications

Chapter 37: Creating a Basic ASP.NET Core App

- Understanding the ASP.NET Core Templates

- Developing a Basic Web App

Chapter 38: Employing the Razor Markup Language

- Avoiding Nicks from Razor

- Creating Variables

- Keeping Things Logical

- Implementing Loops

Chapter 39: Generating and Consuming Data

- Understanding Why These Projects Are Important

- Serialized Data Isn't for Breakfast

- Developing a Data Generator and API

- Creating a Consumer Website

10. Live Labs

The benefits of live-labs are:

- Exam based practical tasks
- Real equipment, absolutely no simulations
- Access to the latest industry technologies
- Available anytime, anywhere on any device
- Break and Reset functionality
- No hardware costs

Lab Tasks

Creating Your First C# Console Application

- Creating a Source Program
- Executing a Source Program

Pulling Strings

- Executing the ModifyString Example Program
- Building a Sentence
- Using Advanced Pattern Matching
- Parsing Numeric Input
- Handling a Series of Numbers
- Formatting Output Using Trim and Pad Methods

Smooth Operators

- Demonstrating Operator Overloading

Getting into the Program Flow

- Calculating the Interest Using the if Statement
- Calculating the Amount of Principle Through Looping

Lining Up Your Ducks with Collections

- Calculating the Average of Numbers
- Processing Arrays by Using the foreach Loop
- Sorting Arrays of Data

Stepping through Collections

- Iterating Through a Directory of Files

Buying Generic

- Creating a Priority Queue

Some Exceptional Exceptions

- Demonstrating Exception Handling

Creating Lists of Items with Enumerations

- Demonstrating the Working of Enumerations

Showing Some Class

- Demostrating Working with Object-Based Code

We Have Our Methods

- Implementing Default Arguments

Let Me Say This about this

- Passing an Object to a Method
- Employing Static Properties and Methods
- Employing Instance Properties and Methods
- Using Attributes with Local Functions

Holding a Class Responsible

- Solving the Rounding Problem
- Using Target Typing

Inheritance: Is That All I Get?

- Inheriting from a BankAccount Class

Poly-what-ism?

- Demonstrating the Polymorphic Nature of AbstractInheritance

Interfacing with the Interface

- Demonstrating the Implementation of the Incomparable IComparable<T> Interface

Delegating Those Important Events

- Demonstrating the Action, Func, and Predicate Delegate Types
- Using Static Anonymous Methods

Can I Use Your Namespace in the Library?

- Analyzing a Project with a Class Library

Interacting with Structures

- Using Supplemental Struct Elements
- Demonstrating a read-only Structure

Writing Secure Code

- Creating a Windows Application

Accessing Data

- Setting Up a Sample Database Schema

Fishing the File Stream

- Using StreamWriter

Accessing the Internet

- Demonstrating the Use of System.Net Namespace

Creating Images

- Demonstrating the Use of System.Drawing Namespace

Customizing Visual Studio

- Developing a Project Template
- Developing an Item Template

Introducing WPF

- Creating a WPF Application

Understanding the Basics of WPF

- Creating a Canvas
- Creating a Data Entry Form
- Demonstrating Display-Only Controls

Data Binding in WPF

- Binding a Data Object with XAML
- Validating the Data

Practical WPF

- Demonstrating the Use of Routed Commands

Programming for Windows 10 and Above

- Analyzing a UWP Project

Creating a Basic ASP.NET Core App

- Analyzing a Basic Web App

Employing the Razor Markup Language

- Overviewing a Razor Page

Generating and Consuming Data

- Developing a Data Generator and API

Here's what you get

52	51	01:28
LIVE LABS	VIDEO TUTORIALS	HOURS

GET IN TOUCH:

3187 Independence Drive
Livermore, CA 94551,
United States

+1-415-763-6300

support@ucertify.com

www.ucertify.com