**Document Generated: 11/07/2026**

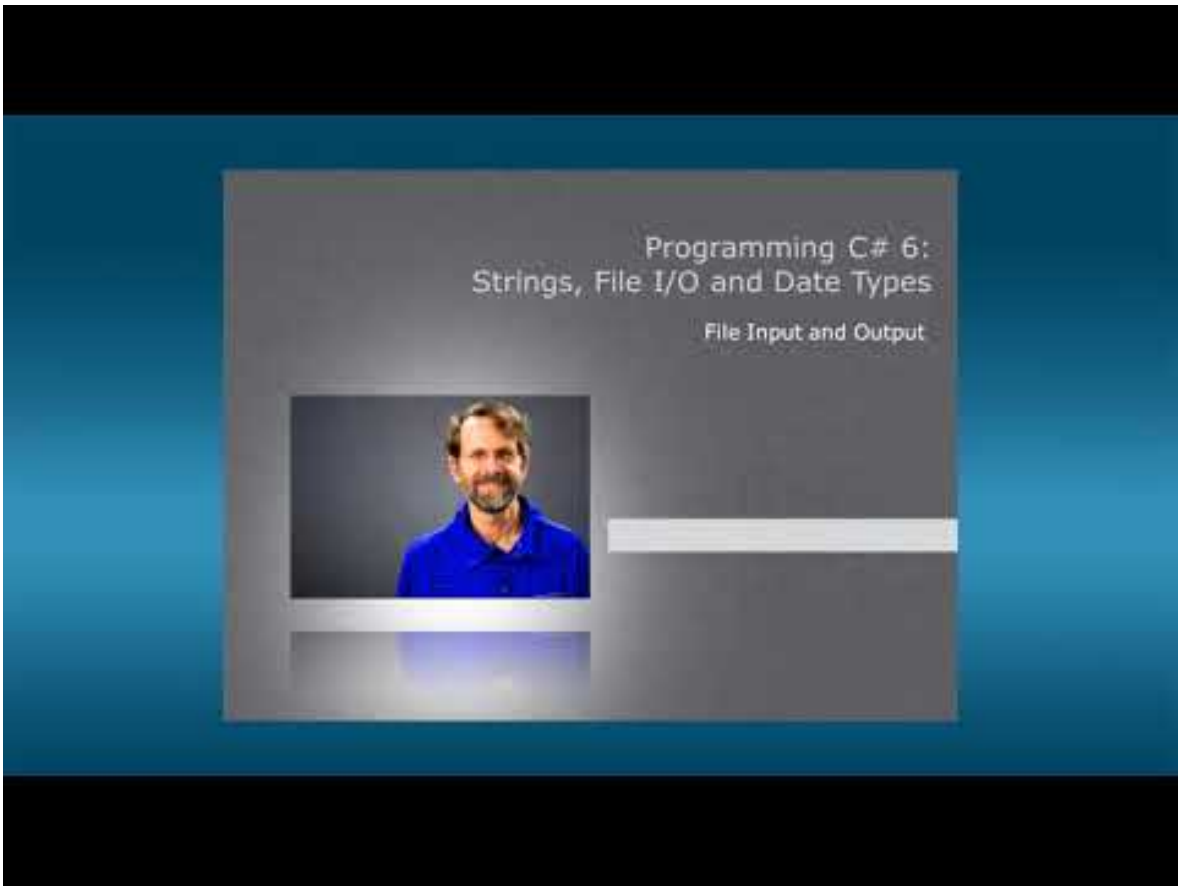
**Learning Style: On Demand**

**Technology:**

**Difficulty: Intermediate**

**Course Duration: 8 Hours**

# Programming C#: Intermediate




## About this course:

The course of Programming C# 6: Intermediate develops on the taught of concepts in the course of C# Fundamentals and covers the more development techniques, instruments, procedures, and innovations utilized by present-day applications of business. The topics of this course incorporate structure new handling events, data types, default and null values, setting precedence, implementing controlled looping, working with XML and file I/O, and creating classes and objects.

This course shows the understudies to build up further developed abilities of programming that are required for designers to make enterprise applications of Windows utilizing the language of C#. In this course, the understudies gain proficiency with the core ideas of the program structure of C#, implementation details, language syntax, and afterward unite their insight as they develop a true application of graphics.

## Course Objective:

- Explain and comprehend the core C# syntax.
- Learn and execute operators and types.
- Comprehend precedence and operators
- Work with dates and strings.
- Utilize unconditional and conditional branching
- Use and Implement classes and objects in code.

## Audience:

- Experienced developers of the software hoping to improve their insight into MS Visual Studio, C#, and .NET system.
- Candidates who have an understanding of C++, Java, Objective-C, and MS Visual Basic.

## Prerequisite:

- This course is a mid-profession level course that requires the understudies to have an essential comprehension of the programming language of C#.
- A half-year of programming involvement with an object-oriented condition is important for the possibility to completely get a handle on the substance of this course.

## Course Outline:

### Chapter 01 - Types and Operators

- **Topic A: Converting Data Types - Part 1**
- Converting Data Types - Part 2
- Converting Data Types - Part 3
- **Topic B: Using Convert Or Parse - Part 1**
- Using Convert Or Parse - Part 2
- Using Convert Or Parse - Part 3
- **Topic C: Value and Reference Types - Part 1**
- Value and Reference Types - Part 2
- Value and Reference Types - Part 3
- **Topic D: Operators - Part 1**
- Operators - Part 2
- Operators - Part 3

### Chapter 02 - Precedence and Nulls

- **Topic A: Logical Operators & Precedence - Part 1**
- Logical Operators & Precedence - Part 2
- Logical Operators & Precedence - Part 3
- **Topic B: Nulls and Default Values - Part 1**
- Nulls and Default Values - Part 2
- Nulls and Default Values - Part 3
- **Topic C: Null Handling Operators - Part 1**
- Null Handling Operators - Part 2
- Null Handling Operators - Part 3
- **Topic D: Nullable Data Types - Part 1**
- Nullable Data Types - Part 2
- Nullable Data Types - Part 3

**Chapter 03 - XML and File I/O**

- **Topic A: .NET Framework Classes - Part 1**
- .NET Framework Classes - Part 2
- .NET Framework Classes - Part 3
- **Topic B: Working with XML - Part 1**
- Working with XML - Part 2
- Working with XML - Part 3
- **Topic C: File Input and Output - Part 1**
- File Input and Output - Part 2
- File Input and Output - Part 3

**Chapter 04 - Strings and Dates**

- **Topic A: Working with Strings - Part 1**
- Working with Strings - Part 2
- Working with Strings - Part 3
- **Topic B: Methods of String Class - Part 1**
- Methods of String Class - Part 2
- Methods of String Class - Part 3
- **Topic C: Formatting Strings - Part 1**
- Formatting Strings - Part 2
- Formatting Strings - Part 3
- **Topic D: Using StringBuilder - Part 1**
- Using StringBuilder - Part 2
- Using StringBuilder - Part 3
- **Topic E: Working with Dates and Times - Part 1**
- Working with Dates and Times - Part 2
- Working with Dates and Times - Part 3
- **Topic F: Using TimeSpan Structure - Part 1**
- Using TimeSpan Structure - Part 2
- Using TimeSpan Structure - Part 3

**Chapter 05 - Conditionals and Looping**

- **Topic A: Conditional Branching - Part 1**
- Conditional Branching - Part 2

- Conditional Branching - Part 3
- **Topic B: Switch Statements - Part 1**
- Switch Statements - Part 2
- Switch Statements - Part 3
- **Topic C: Repeating Code Blocks - Part 1**
- Repeating Code Blocks - Part 2
- Repeating Code Blocks - Part 3
- **Topic D: Controlled Looping - Part 1**
- Controlled Looping - Part 2
- Controlled Looping - Part 3
- **Topic E: Unconditional Branching - Part 1**
- Unconditional Branching - Part 2
- Unconditional Branching - Part 3

**Chapter 06 - Objects and Classes**

- **Topic A: Objects and Classes - Part 1**
- Objects and Classes - Part 2
- Objects and Classes - Part 3
- **Topic B: Class Properties and Methods - Part 1**
- Class Properties and Methods - Part 2
- Class Properties and Methods - Part 3
- **Topic C: Creating a Class - Part 1**
- Creating a Class - Part 2
- Creating a Class - Part 3
- **Topic D: Visual Studio for Classes - Part 1**
- Visual Studio for Classes - Part 2
- Visual Studio for Classes - Part 3
- **Topic E: Implement Class Code - Part 1**
- Implement Class Code - Part 2
- Implement Class Code - Part 3
- **Topic F: Add Class Methods - Part 1**
- Add Class Methods - Part 2
- Add Class Methods - Part 3

## Credly Badge:




**Display your Completion Badge And Get The Recognition You Deserve.**

Add a completion and readiness badge to your Linkedin profile, Facebook page, or Twitter account to validate your professional and technical expertise. With badges issued and validated by Credly, you can:

- Let anyone verify your completion and achievement by clicking on the badge

- Display your hard work and validate your expertise
- Display each badge's details about specific skills you developed.

Badges are issued by QuickStart and verified through Credly.

Find Out More or See List Of Badges