

C# Programming Crash Course

Modality: On Demand

Duration: 11 Hours

About this Course:

What is C# ?

C# is an object-oriented programming language from Microsoft's C family of languages. It is a hybrid of C and C++ and was designed to improve productivity in web app development. It is used to build a variety of secure client, client-server and database applications, XML web services, distributed components and much more. It runs on the .NET framework and another very popular language was inspired by it; Java.

Knowing the suite of C languages developed by Microsoft is a guaranteed career booster. They're powerful, in-demand, and can create almost any kind of application or software under the sun. However, they're not exactly the most beginner friendly languages in the world. While they all have the same roots, they have distinct differences too. Thankfully C# is generally considered to be the least confusing of the family, and a level up from Java as far as skill is concerned. This crash course will teach you everything you need to know and make you a C# aficionado in a matter of hours.

Course Objectives:

- Understand C# fundamentals
- Create games with C# and Unity
- Develop mobile apps with C# and Xamarin
- Build software with C#, Mono and Windows Forms
- Make websites with C# and ASP.NET
- Create console applications
- Everything You Need to Get up to Speed

C# is a great language, but it's not perfect. With that in mind, this online course will start off by outlining what exactly C# is good at, and where its weaknesses lie. Following that we'll look at the various platforms that utilize C#, which you'll be using during the course to complete a wide variety of projects.

After that there's nothing left to do but dive straight in. Step by step, section by section, you'll build your knowledge of C# with practical projects along the way to put your developing skills into practice. A section each is dedicated to setting up your environment; data types and their uses; variables, expressions and operators; statements and flow control; organization; classes and structs; inheritance; interfaces, and so much more.

All you need to get started is a basic understanding of programming and a familiarity with the concept of object-oriented programming (don't worry if you don't have any actual experience). Suitable for beginners and intermediates alike, by the end of this course you'll have a solid understanding of C#,

its strengths and weaknesses, what projects you can use it for, and you'll be more than ready to start using it in the real world.

Course Outline:

Introduction to the Course

- Course Introduction
- Final Application Demonstration

Overview Of C#

- Section Introduction
- C# Strengths & Weaknesses
- Popular Uses Of C#
- Demystifying The .Net Framework
- Section Summary
- Section 2 Quiz

Setting Up Environment

- Section Introduction
- Downloading And Installing Visual Studio
- Creating A New Console Project
- Environment Break Down
- Preventing The Console From Auto Exiting
- Section Summary
- Section 3 Quiz

Data Types & Their Uses

- Section Introduction
- Categories Of Data Types
- Predefined Data Types
- Arrays Part 1 - Single Dimensional And Jagged
- Arrays Part 2 - Multidimensional And Arrays As Objects
- Enums
- Section Summary
- Section 4 Quiz

Variables, Expressions, & Operators

- Section Introduction
- Variables - Implicit And Explicit Creation And Conversion
- Definite Assignment
- Expressions And Operators
- Section Summary
- Section 5 Quiz

Statements & Flow Control

- Section Introduction
- Breaking Down Statements
- Selection Statements
- Looping Statements
- Jumping Statements
- Section Summary
- Section 6 Quiz

Organization

- Section Introduction
- Files
- Namespaces
- Section Summary
- Section 7 Quiz
- Classes And Structs
- Section Introduction
- Classes And Inheritance
- Classes Vs Structs
- Fields And Constants
- Methods
- Method Modifiers
- Properties And Accessors
- Static Members
- Constructors
- Indexers
- Overloading Operators
- Section Summary
- Section 8 Quiz

Inheritance

- Section Introduction
- A Brief Overview Of Inheritance
- Access Modifiers
- Class Conversion

- Polymorphism
- Abstract Classes And Members
- Sealed Methods And Classes
- Hiding Inherited Members
- Section Summary
- Section 9 Quiz

Interfaces

- Section Introduction
- Breaking Down Interfaces
- Defining And Using Interfaces With Polymorphism
- Extending Interfaces
- Avoiding Common Interface Problems
- Section Summary
- Section 10 Quiz

Delegates & Events

- Section Introduction
- Defining, Creating, And Using Delegates
- Multicast Delegates
- Defining, Creating, And Handling Events
- Section Summary
- Section 11 Quiz

Exception Handling

- Section Introduction
- A Broad Overview Of Exceptions
- Try, Catch, And Finally
- Section Summary
- Section 12 Quiz

Attributes, Preprocessor Directives, & Documentation

- Section Introduction
- Defining And Utilizing Attributes
- Defining And Utilizing Preprocessor Directives
- Comments
- Section Summary
- Section 13 Quiz

Setting Up Notebook Application Components

- Section Introduction
- Creating Notebook App And Implementing IPageable Interface
- Creating A Structure For Holding Page Data
- Creating Textual Message Class And Implementing IPageable Interface
- Creating Child MessageList Class And Implementing Enums
- Creating The Image Class
- Section Summary
- Section 14 Quiz

Tying Components Together In The New Notebook Class

- Section Introduction
- Creating A List Of Pages In The New Notebook Class
- Utilizing Delegates, Dictionaries, And Indexers To Interact With Notebook
- Using Constructors And The Params Modifier To Initialize Notebook Dictionary
- Instantiating Notebook And Using Do While Loop to Accept Continuous Input
- Fixing Key Not Found Exception With Try And Catch
- Using Switch Statement To Extract Secondary Commands Part 1
- Using Switch Statement To Extract Secondary Commands Part 2
- Section Summary
- Section 15 Quiz

Defining Input, Output, Delete & Show Functions

- Section Introduction (0:55)
- Defining Input And Output Methods For TextualMessage Class
- Adding, Showing, And Removing List Pages
- Overriding Input And Utilizing enums In MessageList Class
- Defining Input And Output Functions For Image Class
- Section Summary
- Section 16 Quiz

Improving Modularity Using Events And NotebookLogger Class

- Section Introduction
- Creating The New NotebookLogger Class
- Defining And Calling Events In the Notebook Class Part 1
- Defining And Calling Events In the Notebook Class Part 2
- Defining And Attaching - Detaching Event Handlers In NotebookLogger
- Toggling Event Handlers In The NotebookLogger Class
- Section Summary
- Section 17 Quiz

Conclusion

- Additional Resources
- Closing Message