C# 8

C# Fundamentals, .NET Fx Intro, Types, Classes, Attributes, Delegates, Generics, Async Streams

C# 8 is the premier development language for the .NET platform. C# was designed from scratch with .NET in mind. Most of the internals of .NET Core 3 and Visual Studio are written in it. It has been selected by the majority of application teams creating commercial .NET Core 3 projects. C# builds on the rich common heritage of languages such as C++ and Java - but avoids their pitfalls and adds certain interesting new concepts, such as LINQ. There are aspects of C# that developers already know, there are some they have experienced similar but slightly different syntax in other languages, and some that are innovative ([v8 new] async streams).

C# can be used to develop stand-alone apps, local and distributed components, web services and mobile code. It produces code that can target desktop PCs, mobile devices, servers and IoT devices. C# 8 can be used for DB [EF Core], UWP, ASP.NET, WebAssembly (Blazor) & security projects. Hence it is an excellent all-round development language for all .NET Core applications. This intensive course aims to take experienced software engineers rapidly through all the major aspects of C# 8 - using plenty of demo source code and hands-on labs to show it in action. This is an ideal first course for those moving to the C# 8 language and .NET Core 3.

Target Audience

Experienced software engineers wishing to rapidly get up to speed with C#.

Prerequisites

Programming experience with an OO language such as C++, TypeScript or Java, along with good exposure to object-oriented design.

No previous experience of C# or .NET is needed.

This course covers C# 8 using Visual Studio 2019.

Contents of One-Day Training Course

C# and the .NET Core

What is the .NET Core?
The Base Class Library
The CLR
How C# is used with .NET
Delivery of C# functionality in assemblies

A C# Project Walk through

Solutions, projects and files
Parts of a C# project
Structure of code
Setting up a solution with a C# app and class library

Base Types

Built-in data types
.NET value types and reference types
How C# and .NET data types compare
Building code in C# that is callable from other languages

Language Fundamentals

Main starting point
Flow control, operators
Variables, methods
Enumerators, bit flags, arrays, indexers
Namespaces

Class Fundamentals

Members, constructors, visibility, ref and out, constant fields, structs
Fields & properties, methods, nested types

Inheritance

Single inheritance only (for classes)
Virtual functions
Override and new keywords
Designing libraries using inheritance

Delegates And Events

Equivalent of function pointers
Defining and exposing delegates
Registering an interest in a delegate
Async info with events
Design pattern for event handling

Interfaces

When to use interfaces
Multiple inheritance & hierarchies
Abstract classes vs. interfaces

Exception Handling

`Try .. catch … finally`
Detecting and responding to exceptions
Strategies for exception handling

Generics & Constraints

Generics (for methods and classes)
Constraints
Partial types
Anonymous methods
Type inferencing

Expression Bodied Members

Succinct member definitions
Methods, constructors, properties, indexers

Specialist Features

Null conditional operator
Auto-property initializer
`nameof`

Calling C Code

Calling out to C code from C#
Passing parameters / accepting return val

C# 8 - What's new

Nullable reference types, async streams, range & indices, mixins, switch expressions