

- Online Training
  - C# .NET Online Training
  - Program
  - ASP.NET Core Training
  - Microservices Online Training using .NET Core
  - Microsoft Azure Training

#### Introduction & Environment Setup

- How Computer Works
- Introduction to Programming Languages
- How Computer Programs Works
- Different Types of Applications
- Programming Methodologies
- Algorithm, Pseudocode, Programs, and Flowcharts
- Introduction to .NET Framework
- .NET Framework Architecture and Components
- Introduction to C# Programming Language
- How to Download and Install Visual Studio on Windows
- Creating First Console Application using Visual Studio
- .NET Developer Roadmap for 2025
- Coding Standard Best Practices

#### C# .NET Basics

- Basic Structure of C# Program
- Methods and Properties of Console Class in C#
- Data Types in C#
  - Literals in C#
  - Type Casting in C#
  - Variables in C#
  - Operators in C#
- Control Flow Statements in C#
  - If-Else Statements in C#
  - Switch Statements in C#
  - Loops in C#
    - While Loop in C#
    - Do While Loop in C#
    - For Loop in C#
    - Break Statement in C#
    - Continue Statement in C#
    - Goto Statement in C#
  - Functions in C#
  - User-Defined Functions in C#
  - Call By Value and Call By Reference in C#
  - Recursion in C#
  - User Input and Output in C#
  - Command Line Arguments in C#
  - String in C#
  - Static Keyword in C#
  - Static vs Non-Static Members in C#
  - Const and Read-Only in C#
  - Properties in C#
  - Why We Should Override ToString Method in C#
  - Override Equals Method in C#
  - Difference Between Convert.ToString and ToString Method in C#
  - Stack and Heap Memory in .NET
  - Boxing and Unboxing in C#

#### OOps in C#

- Object Oriented Programming (OOPS) in C#
- Class and Objects in C#
- Constructors in C#
  - Types of Constructors in C#
  - Why We Need Constructors in C#
- Static vs Non-Static Constructors in C#
- Private Constructors in C#
- Destructors in C#
- Garbage Collection in .NET Framework
- Differences Between Finalize and Dispose in C#
- Access Specifiers in C#
- Encapsulation in C#
- Abstraction in C#
- Inheritance in C#
  - Types of Inheritance in C#
  - How to use Inheritance in Application Development
  - IsA and HasA Relationship in C#
  - Generalization and Specialization in C#
- Abstract Class and Abstract Methods in C#
  - Abstract Class and Abstract Methods Interview Questions in C#
  - How to Use Abstract Classes and Methods in C#
- Interface in C#
  - Interface Interview Questions and Answers in C#
  - Interface Realtime Examples in C#
  - Multiple Inheritance in C#
  - Multiple Inheritance Realtime Example in C#
  - Polymorphism in C#
  - Method Overloading in C#
  - Operator Overloading in C#
  - Method Overriding in C#
  - Method Hiding in C#
  - Partial Class and Partial Methods in C#
  - Sealed Class and Sealed Methods in C#
  - Extension Methods in C#
  - Static Classes in C#
  - Variable Reference and Instance of a Class in C#

#### OOPS Real-Time Examples

- Real-time Examples of Encapsulation Principle in C#
- Real-time Examples of Inheritance Principle in C#
- Real-time Examples of Polymorphism Principle in C#
- Real-time Examples of Interface in C#
- Real-time Examples of Abstract Class in C#

#### Exception Handling

- Exception Handling in C#
  - Multiple Catch Blocks in C#
  - Finally Block in C#
  - How to Create Custom Exceptions in C#
  - Inner Exception in C#
  - Exception Handling Abuse in C#

#### Events, Delegates and Lambda Expression in C#

- Course Structure of Events, Delegates and Lambda Expression
- Roles of Events, Delegates and Event Handler in C#
- Delegates in C#
- Multicast Delegates in C#
- Delegates Real-Time Example in C#
- Generic Delegates in C#
- Anonymous Method in C#
- Lambda Expressions in C#
- Events in C# with Examples

#### Multi-Threading

- Multithreading in C#
  - Thread class in C#
  - How to Pass Data to Thread Function in Type Safe Manner in C#
  - How to Retrieve Data from a Thread Function in C#
  - Join Method and IsAlive Property of Thread Class in C#
  - Thread Synchronization in C#
  - Lock in C#
  - Monitor Class in C#
  - Mutex Class in C#
  - Semaphore Class in C#
  - SemaphoreSlim Class in C#
  - Deadlock in C#
  - Performance Testing of a Multithreaded Application
  - Thread Pool in C#
  - Foreground and Background Threads in C#
  - AutoResetEvent and ManualResetEvent in C#
  - Thread Life Cycle in C#
  - Threads Priorities in C#
  - How to Terminate a Thread in C#
  - Inter Thread Communication in C#
  - How to Debug a Multi-threaded Application in C#

#### Collections in C#

- Arrays in C#
- 2D Arrays in C#
- Advantages and Disadvantages of Arrays in C#
- Collections in C#
  - ArrayList in C#
  - Hashtable in C#
  - Non-Generic Stack in C#
  - Non-Generic Queue in C#
  - Non-Generic SortedList in C#
  - Advantages and Disadvantages of Non-Generic Collection in C#
  - Generic Collections in C#
  - Generics in C#
  - Generic Constraints in C#
  - Generic List Collection in C#
  - How to Sort a List of Complex Type in C#
  - Comparison Delegate in C#
  - Dictionary Collection Class in C#
  - Conversion Between Array List and Dictionary in C#
  - List vs Dictionary in C#
  - Generic Stack Collection Class in C#
  - Generic SortedList Collection Class in C#
  - Generic SortedSet Collection Class in C#
  - Generic SortedDictionary Collection Class in C#
  - Generic LinkedList Collection Class in C#
  - Concurrent Collection in C#
  - ConcurrentDictionary Collection Class in C#
  - ConcurrentQueue Collection Class in C#
  - ConcurrentStack Collection Class in C#
  - ConcurrentBag Collection Class in C#
  - BlockingCollection in C#

#### File Handling

- File Handling in C#
  - FileStream Class in C#
  - StreamReader and StreamWriter in C#
  - File Class in C#
  - TextWriter and TextReader in C#
  - BinaryWriter and BinaryReader in C#
  - StringWriter and StringReader in C#
  - FileInfo Class in C#
  - DirectoryInfo Class in C#
  - Export and Import Excel Data in C#

#### Asynchronous Programming

- Introduction to Concurrency
- Async and Await in C#
- Task in C#
- How to Return a Value from Task in C#
- How to Execute Multiple Tasks in C#
- How to Limit Number of Concurrent Tasks in C#
- How to Cancel a Task in C# using Cancellation Token
- How to Create Synchronous Method using Task in C#
- Retry Pattern in C#
- Only One Pattern in C#
- How to Control the Result of a Task in C#
- Task-Based Asynchronous Programming in C#
- Chaining Tasks by Using Continuation Tasks
- How to Attach Child Tasks to a Parent Task in C#
- ValueTask in C#
- How to Cancel a Non-Cancellable Task in C#
- Asynchronous Streams in C#
- How to Cancel Asynchronous Stream in C#

#### Parallel Programming

- Task Parallel Library in C#
- Parallel For in C#
- Parallel Foreach Loop in C#
- Parallel Invoke in C#
- Maximum Degree of Parallelism in C#
- How to Cancel Parallel Operations in C#
- Atomic Methods Thread Safety and Race Conditions in C#
- Interlocked vs Lock in C#
- Parallel LINQ in C#
- Parallel LINQ in C#
- Parallel.Invoke in C#
- How to Control the Result of a Task in C#
- Task-Based Asynchronous Programming in C#
- Chaining Tasks by Using Continuation Tasks
- How to Attach Child Tasks to a Parent Task in C#
- ValueTask in C#
- How to Cancel a Non-Cancellable Task in C#
- Asynchronous Streams in C#
- How to Cancel Asynchronous Stream in C#

#### AutoMapper

- AutoMapper in C#
- AutoMapper Complex Mapping in C#
- AutoMapper Conditional Mapping in C#
- AutoMapper Ignore Method in C#
- Fixed and Dynamic Values in Destination Property in AutoMapper
- Reverse Mapping in C#

#### Optional Parameter, Indexers and Enums

- How to make Optional Parameters in C#
- Indexers in C#
- Indexers Real-Time Example in C#
- Enums in C#

#### .NET Framework Architecture

- DOT NET Framework
- Common Language Runtime in .NET Framework
- .NET Program Execution Process
- Intermediate Language (IL/DASM & ILASM) Code in C#
- Common Type System in .NET Framework
- Common Language Specification in .NET Framework
- Managed and Unmanaged Code in .NET Framework
- App Domain in .NET Framework
- Strong and Weak Assemblies in .NET Framework
- How to Install an Assembly into GAC in .NET Framework
- DLL Hell Problem and Solution in .NET Framework

#### Var, Dynamic and Reflection

- Reflection in C#
- Dynamic Type in C#
- Var Keyword in C#
- Dynamic vs Reflection in C#
- Ref Returns End Ref Locals in C#
- Generalized Async Return Types in C#
- Expression Bodied Members in C#
- Thrown Expression in C#
- Async Main in C#
- Named Parameters in C#

#### C# 7 New Features

- C# 7 New Features
- Enhancement in Out Variables in C#
- Pattern Matching in C#
- Digit Separators in C#
- Local Functions in C#
- Ref Returns End Ref Locals in C#
- Generalized Async Return Types in C#
- Expression Bodied Members in C#
- Thrown Expression in C#
- Async Main in C#
- Named Parameters in C#

#### C# 8 New Features

- C# 8 New Features
- Readonly Structs in C#
- Static Local Functions in C#
- Disposable Ref Structs in C#
- Nullable Reference Types in C#
- Pattern Matching in C#
- Using Declarations in C#
- Static Local Functions in C#
- Disposable Ref Structs in C#
- Null-Comparing Assignment Operator in C#
- Unmanaged Constructed Types in C#
- Stackalloc in C#

#### Most Popular C# Books

- Most Recommended C# Books
- Most Recommended Data Structure and Algorithms Books using C#

#### Events, Delegates and Lambda Expression in C#

- Course Structure of Events, Delegates and Lambda Expression
- Roles of Events, Delegates and Event Handler in C#
- Delegates in C#
- Multicast Delegates in C#
- Delegates Real-Time Example in C#
- Generic Delegates in C#
- Anonymous Method in C#
- Lambda Expressions in C#
- Events in C# with Examples

#### Multi-Threading

- Multithreading in C#
  - Thread class in C#
  - How to Pass Data to Thread Function in Type Safe Manner in C#
  - How to Retrieve Data from a Thread Function in C#
  - Join Method and IsAlive Property of Thread Class in C#
  - Thread Synchronization in C#
  - Lock in C#
  - Monitor Class in C#
  - Mutex Class in C#
  - Semaphore Class in C#
  - SemaphoreSlim Class in C#
  - Deadlock in C#
  - Performance Testing of a Multithreaded Application
  - Thread Pool in C#
  - Foreground and Background Threads in C#
  - AutoResetEvent and ManualResetEvent in C#
  - Thread Life Cycle in C#
  - Threads Priorities in C#
  - How to Terminate a Thread in C#
  - Inter Thread Communication in C#
  - How to Debug a Multi-threaded Application in C#

#### Collections in C#

- Arrays in C#
- 2D Arrays in C#
- Advantages and Disadvantages of Arrays in C#
- Collections in C#
  - ArrayList in C#
  - Hashtable in C#
  - Non-Generic Stack in C#
  - Non-Generic Queue in C#
  - Non-Generic SortedList in C#
  - Advantages and Disadvantages of Non-Generic Collection in C#
  - Generic Collections in C#
  - Generics in C#
  - Generic Constraints in C#
  - Generic List Collection in C#
  - How to Sort a List of Complex Type in C#
  - Comparison Delegate in C#
  - Dictionary Collection Class in C#
  - Conversion Between Array List and Dictionary in C#
  - List vs Dictionary in C#
  - Generic Stack Collection Class in C#
  - Generic SortedList Collection Class in C#
  - Generic SortedSet Collection Class in C#
  - Generic SortedDictionary Collection Class in C#
  - Generic LinkedList Collection Class in C#
  - Concurrent Collection in C#
  - ConcurrentDictionary Collection Class in C#
  - ConcurrentQueue Collection Class in C#
  - ConcurrentStack Collection Class in C#
  - ConcurrentBag Collection Class in C#
  - BlockingCollection in C#

#### File Handling

- File Handling in C#
  - FileStream Class in C#
  - StreamReader and StreamWriter in C#
  - File Class in C#
  - TextWriter and TextReader in C#
  - BinaryWriter and BinaryReader in C#
  - StringWriter and StringReader in C#
  - FileInfo Class in C#
  - DirectoryInfo Class in C#
  - Export and Import Excel Data in C#

#### Asynchronous Programming

- Introduction to Concurrency
- Async and Await in C#
- Task in C#
- How to Return a Value from Task in C#
- How to Execute Multiple Tasks in C#
- How to Limit Number of Concurrent Tasks in C#
- How to Cancel a Task in C# using Cancellation Token
- How to Create Synchronous Method using Task in C#
- Retry Pattern in C#
- Only One Pattern in C#
- How to Control the Result of a Task in C#
- Task-Based Asynchronous Programming in C#
- Chaining Tasks by Using Continuation Tasks
- How to Attach Child Tasks to a Parent Task in C#
- ValueTask in C#
- How to Cancel a Non-Cancellable Task in C#
- Asynchronous Streams in C#
- How to Cancel Asynchronous Stream in C#

#### Parallel Programming

- Task Parallel Library in C#
- Parallel For in C#
- Parallel Foreach Loop in C#
- Parallel Invoke in C#
- Maximum Degree of Parallelism in C#
- How to Cancel Parallel Operations in C#
- Atomic Methods Thread Safety and Race Conditions in C#
- Interlocked vs Lock in C#
- Parallel LINQ in C#
- Parallel LINQ in C#
- Parallel.Invoke in C#
- How to Control the Result of a Task in C#
- Task-Based Asynchronous Programming in C#
- Chaining Tasks by Using Continuation Tasks
- How to Attach Child Tasks to a Parent Task in C#
- ValueTask in C#
- How to Cancel a Non-Cancellable Task in C#
- Asynchronous Streams in C#
- How to Cancel Asynchronous Stream in C#

#### AutoMapper

- AutoMapper in C#
- AutoMapper Complex Mapping in C#
- AutoMapper Conditional Mapping in C#
- AutoMapper Ignore Method in C#
- Fixed and Dynamic Values in Destination Property in AutoMapper
- Reverse Mapping in C#

#### Optional Parameter, Indexers and Enums

- How to make Optional Parameters in C#
- Indexers in C#
- Indexers Real-Time Example in C#
- Enums in C#

#### .NET Framework Architecture

- DOT NET Framework
- Common Language Runtime in .NET Framework
- .NET Program Execution Process
- Intermediate Language (IL/DASM & ILASM) Code in C#
- Common Type System in .NET Framework
- Common Language Specification in .NET Framework
- Managed and Unmanaged Code in .NET Framework
- App Domain in .NET Framework
- Strong and Weak Assemblies in .NET Framework
- How to Install an Assembly into GAC in .NET Framework
- DLL Hell Problem and Solution in .NET Framework

#### Var, Dynamic and Reflection

- Reflection in C#
- Dynamic Type in C#
- Var Keyword in C#
- Dynamic vs Reflection in C#
- Ref Returns End Ref Locals in C#
- Generalized Async Return Types in C#
- Expression Bodied Members in C#
- Thrown Expression in C#
- Async Main in C#
- Named Parameters in C#

#### C# 7.X New Features

- C# 7 New Features
- Enhancement in Out Variables in C#
- Pattern Matching in C#
- Digit Separators in C#
- Local Functions in C#
- Ref Returns End Ref Locals in C#
- Generalized Async Return Types in C#
- Expression Bodied Members in C#
- Thrown Expression in C#
- Async Main in C#
- Named Parameters in C#

#### C# 8 New Features

- C# 8 New Features
- Readonly Structs in C#
- Static Local Functions in C#
- Disposable Ref Structs in C#
- Nullable Reference Types in C#
- Pattern Matching in C#
- Using Declarations in C#
- Static Local Functions in C#
- Disposable Ref Structs in C#
- Null-Comparing Assignment Operator in C#
- Unmanaged Constructed Types in C#
- Null-Coalescing Assignment Operator in C#
- Stackalloc in C#

#### Most Popular C# Books

- Most Recommended C# Books
- Most Recommended Data Structure and Algorithms Books using C#

#### Events, Delegates and Lambda Expression in C#

- Course Structure of Events, Delegates and Lambda Expression
- Roles of Events, Delegates and Event Handler in C#
- Delegates in C#
- Multicast Delegates in C#
- Delegates Real-Time Example in C#
- Generic Delegates in C#
- Anonymous Method in C#
- Lambda Expressions in C#
- Events in C# with Examples

