

Online Training

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

Introduction & Environment Setup

- How to Complain Computer Works
- Introduction to Programming Languages
- How to 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#
- Checked and Unchecked Key words 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# Application
- 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 Hiding in C#
- Partial Class and Partial Methods in C#
- Sealed Classes and Sealed Methods in C#
- Extension Methods in C#
- Static Class 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#
- Dictionary Comparison in C#
- Collection Class in C#
- Conversion Between Array List and Dictionary in C#
- List vs Dictionary in C#
- Generic Stack Collection Class in C#
- Concurrent Collection in C#
- Concurrent Dictionary Collection in C#
- Concurrent Queue Collection Class in C#
- Concurrent Stack Collection Class in C#
- Concurrent Bag Collection Class in C#
- BlockingCollection 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

- Multi-threading 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#

Dictionary Comparison in C#

- Collection Class in C#

Conversion Between Array List and Dictionary in C#

- List vs Dictionary in C#

Generic Stack Collection Class in C#

- Concurrent Collection in C#

Concurrent Dictionary Collection in C#

- Concurrent Queue Collection Class in C#

Concurrent Stack Collection Class in C#

- Concurrent Bag Collection Class in C#

BlockingCollection in C#

- How to Cancel a Task in C# using CancellationToken

- How to Create Synchronous Method to Task in C#

Retry Pattern in C#

- How to Attached Child Tasks to a Parent Task in C#

ValueTask in C#

- How to Cancel a Non-Cancellable Task in C#

Asynchronous Streams in C#

- Asynchronous Stream in C#

Asynchronous Disposible in C#

- IDisposable in C#

Fixed and Dynamic Values in Destination or their in C#

- AutoMapper Ignore Method in C#

Managed and Unmanaged Code in C#

- Interlocked Loops in C#

Parallel ForEach in C#

- Parallel Foreach Loop in C#

Parallel Invoke in C#

- Maximum Degree of Parallelism in C#

How to Cancel a Parallel Operations in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

- Parallel Invoke in C#

Performance Tuning in C#

