1. **Overview of the Microsoft .NET Framework**

* Introduction to the .NET Platform and .NET Framework
* Framework Class Library, ADO.NET Library and ASP.NET Library
* Understanding the Common Type System (CTS)
* Introduction to the Need for the Common Language Runtime (CLR)
* Components of CLR and Roles
  + JIT Compiler
  + Type Checker
  + Exception Manager
  + Security Checker
  + Com Marshaler
  + Thread Support
  + Garbage Collector
  + Code manager
  + Class Loader
* Managed Code Vs. Unmanaged Code
* Understanding the Just-In-Time (JIT) Compilation Process
  + MSIL Code
  + Metadata- The Self Explanatory Files
  + Extracting IL Code and Viewing Metadata using ILDASM Tool
  + .NET Assemblies Explanation
  + Difference between .NET Exe File and Other Exe, DLL Description
* Overview of the .NET-Based Languages
* Comparison of the .NET-Based Languages
* The .NET Framework – Versions

1. **Using Microsoft Visual Studio .NET 2019**

* Overview of Visual Studio .NET IDE Features
* Properties Window
* Tool Box
* Solution Explorer
* Server Explorer
* Object Browser
* Editor Browser
* Creating a Console Application Project
* Creating Windows Application Project
* Compiling Running and Debugging Application
* Folder Structure and File Types Created by Application

**THE MODULES – Day 2**

1. **C# Language Fundamentals**

* Structure of a C# Program
* Basic Input/Output Operations
* Commenting a Program
* Recommended Practices

1. **Using Value-Type Variables in C#**

* Naming Variables
* Best Practices for Naming Conventions
* Using Built-In Data Types
* Creating User-Defined Data Types
* Converting Data Types
* Typecasting
* Boxing and Un-boxing data types

1. **C# Statements**

* Introduction to Statements
* Using Selection Statements
* Using Iteration Statements
* Using Jump Statements
* Using Conditional Statements
* Applications Based on All Statements

**THE MODULES – Day 03**

1. **Essentials of Object-Oriented Programming in C#**

* Understanding Namespaces
* Understanding Scope Resolution
* Defining Classes
* Instantiating and Working with Objects
* Difference between Abstraction and Encapsulation
* Understanding and Implementing Encapsulation
* Defining Object-Oriented Systems

1. **String and Arrays in C#**

* String Handling
  + The String and String Builder Class
  + Different Methods and Properties of String and String Builder Class
* Arrays
  + Overview of Arrays
  + Creating and using Single Dimension and Multi Dimension Arrays
  + Jagged Arrays
  + Using foreach Loop
  + Param Keyword

**THE MODULES – Day 04**

1. **Methods and Parameters using C#**

* Using Methods
* Using Parameters
  + Passing value type parameters
  + Passing Reference types (string, Array, object) as parameters
* Passing Parameters using Ref and Out keyword
* Static and Instance Members
* Explaining Constant and Read-only

1. **Creating Objects in C#**

* Using Constructors
* Using Initializer Lists
* Initializing Data

**THE MODULES – Day 5**

1. **Properties and Indexers in C#**

* Data Fields
* Properties
* Using Indexers
* Compare and Contrast between Properties and Indexers

**THE MODULES – Day 06**

1. **Inheritance in C#**

* Deriving classes
* Understanding Type Hierarchy
* Hiding Base Class Member in Derived Class
* Implementing Multi Level Inheritance
* Using base keyword
* Using Static Classes

1. **Access Modifiers and Constructor**

* Access Modifiers in C#
* Default Accessibility Level for Class, Methods and Structures
* Constructor Execution Sequence in Inheritance Scenario
* Default Constructor and Parameterized Constructor
* Constructor in Structure in C#
* Calling Base Class Constructor in Derived Class Constructor
* Using this keyword to Call One Constructor by another Constructor
* Discussing Public, Private and Protected Constructor
* Understanding Static Constructor
* Differentiating Static Constructor and Instance Constructor by call Mechanism
* Implementing Singleton Design Pattern and Understanding Static Class

**THE MODULES – Day 7**

1. **Polymorphism in C#**

* Polymorphism Using Methods
  + Overloading a Method
  + Overriding Virtual Method
* Abstract Class and Abstract Method
* Interface Implementation
* Interface Inheritance and Implementation
* Using Sealed Class and Sealed Method
* Discussed Inheritance and Interface Implementation in Structure
* Discussion to Differentiate Virtual Method, Abstract Method and Interface

**THE MODULES – Day 8**

1. **Operators and Equality Comparision**

* Introduction to operators
* Operator overloading
* Equality Comparison Operators and Methods
  + Comparing Value Equality
  + Comparing Reference Equality
  + Using ==, Equals, ReferenceEquals, CompareTo
  + Comparison by GetHashCode Method
  + Overriding Methods and Operators for Equality Comparison
* Overriding ToString Method

1. **Exception Handling**

* Checked and Unchecked Statements
* Try, Catch and Finally
* Creating Custom Exception
* Exception Handling Best Practices
* Do’s and Don’ts of Exception Handling

**THE MODULES – Day 09**

1. **Collection Classes in C#**

* Understanding Collection
* Using Different Collections viz. ArrayList, Stack, Queue, SortedList
* Understanding Different Interface viz. IEnumerable,
* IEnumerator, IComparable, IComparer, IList, IEquatable
* Hashing Mechanism
* Generic Collection Classes
* Performance Improvement using Generic Collection over Non-Generic version

**THE MODULES – Day 10**

1. **Delegates and Events in C#**

* Creating and using Delegates
  + Multicast Delegates
  + Anonymous Method
* When to Use Delegates, Events and Interfaces
* Covariance and Contravariance in Delegates
* Generic Delegates
* Comparing C++ Templates and C# Delegates
* Implementing Polymorphism Using Delegates
* Defining and using Events
* Creating Custom Events and Using it
* Passing Event Arguments

1. **Creating Windows Applications**

* Creating a Windows Form
* Windows Form Controls
* Writing Code for Control Events
* Understanding Delegates and Events Implemented in windows Forms

Writing a Common Method Called for Click Event of Multiple Buttons

**THE MODULES – Day 11**

1. **Destroying Objects and Resource Management in C#**

* Objects and Memory
* Using Destructors
* Destroying Objects
* Programming for the Garbage Collector
* Implementing the IDisposable Interface
* Understanding and Implementing Object Pooling
  + Cloning Objects implementing Shallow and Deep Copies

1. **File Handling**

* FileSystemInfo Base Class, FileInfo Class and their Members
* Streams
* Reader/Writer
* Basic File IO

1. **Serialization**

* Serialization Scenarios
* Serialization Attributes
* Object Graph
* Serialization Process
* Serialization Example
* Deserialization Example

**THE MODULES – Day 12**

1. **Threading**
   * Threading & Synchronization
   * Life cycle of a thread
   * Different Thread Methods and Properties
   * Synchronizing critical data using Synchronization objects
   * Thread Pool
2. **Language Enhancements in C# 2.0**

* Static Classes
* Property Accessors
* External Aliases
* Nullable types
* Iterators
* Partial types
  + Generics

1. **Language Enhancements in C# 3.0**

* Implicitly typed local variables
* Anonymous Types
* Extension Methods
* Object and Collection Initializer
* Lambda Expressions
* Query Expressions
  + Expression Trees

**THE MODULES – Day 13**

1. **Language Enhancement in C# 4.0, 5.0 and .NET 4.5**

* Named and Optional Parameters
* Co and Contra variance
* Dynamic Typing and Late Binding
* Parallelization Overview
* Task Parallel Library
* Threads Vs. Tasks
  + Parallel Extensions in .NET 4.5
  + Async and await in C# 5.0

1. **C# 6.0 Features**

* using Static.
* Auto property initializer.
* Dictionary Initializer.
* nameof Expression.
* New way for Exception filters.
* await in catch and finally block.
* Null – Conditional Operator.
* Expression – Bodied Methods
* Easily format strings – String interpolation

**THE MODULES – Day 14-15**

1. **C# 7.0 Features**

* [Tuples](https://csharp.today/c-7-0-tuples/)
* [Deconstruction](https://csharp.today/c-7-deconstruction/)
* Non-'NULL' able reference type
* Minimizing [Out variables](https://csharp.today/c-7-out-variables/)
* [Patterns](https://csharp.today/c-7-0-patterns/) Matching
* [Readability Improvements](https://csharp.today/c-7-literal-improvements/) with Literals
* [Local functions](https://csharp.today/c-7-local-functions/)
* [Expression-bodied members](https://csharp.today/c-7-expression-bodied-members/)
* [Throw Expressions](https://csharp.today/c-7-throw-expressions/)

1. **.NET Remoting**

* Understanding Application Domain and Remoting Architecture
* Accessing .NET Components Across Application Domain
  + .NET Remoting architecture
  + Creation of Proxy Objects by the CLR
  + Using the Channel Services to Transport Remote
  + Object Across Application Domains
  + Using TCP Channel, Using HTTP Channel
* Formatter for Creating Message and Encoding it
  + Soap Formatter
  + Binary Formatter
* Activation Model
  + Server Activated
    - * Creating a Single Call Object
      * Creating a Singleton Object
  + Client Activated
    - * Managing Lifetime with Lease Manager
* Hosting .NET Remote Component
  + Using Framework Classes
* Using Configuration Files