

Course Learning Objectives:

- The concept of .NET framework, building blocks of .NET framework and application development using IDE.
- C# programming language, use of windows forms and GUI based programs.
- OOP concepts, concept of assemblies and string manipulation.
- Designing of web applications and validating forms using validation controls, interacting with database using server side programming.

Course Outcomes: After completing the course, the students will be able to,

- Understand .NET framework, its runtime environment and application development using IDE of Visual Studio 2010 and higher versions.
- Develop well-defined programs using the C# programming language; learn to use Windows forms and to create GUI-based programs.
- Able to apply the principles of object-oriented programming and develop assemblies and deployment in .NET.
- Apply and build web applications and validation form data using validation controls.
- Create dynamic web applications that interact with a database using server-side programming.
- Understand Constructing classes, method sand instantiate objects.
- Understand and implement string manipulation, events and exception handling within .NET application environment.

UNIT-I**12Hrs.**

Introduction: Principles of .NET, Overview of .NET Framework, Review of OOP Concepts – C# language fundamentals – Basic Elements of C# – Program Structure and simple Input and Output Operations – Data types –Value types –Reference types – Identifiers – Variables – Constraints –Literals – Operators and Expressions – Statements – Arrays and Structures. Object Oriented Programming Concepts: Encapsulation – Encapsulation Services – Pseudo- Encapsulation: Creating Read-Only Fields- Inheritance - Namespace – Polymorphism – Interface and Overloading – Multiple Inheritance – Property – Indexes – Delegates and Events – Publish/Subscribe Design Patterns- Operator Overloading– Method Overloading.

UNIT-II**12Hrs.**

C# Concepts for creating Data Structures - File Operation – File Management systems – Stream Oriented Operations- Multitasking – Multithreading – Thread Operation – Synchronization– Exceptions and Object lifetime, Building C# Applications: The Role of the Command Line Compiler – Building C # Applications, Working with csc.exe, Response Files– Generating Bug Reports – Remaining C# Compiler Options – The Command Line

Debugger (cordbg.exe) – Using the Visual Studio .NET IDE – Other Key Aspects of the VS.NET IDE – C# "Preprocessor:" Directives.

UNIT-III

12Hrs.

.NET ASSEMBLERS and Windows Applications: An Overview of .NET Assembly – Building a Simple File Test Assembly– A C# Client Application– A Visual Basic .NET Client Application– Cross Language Inheritance– Exploring the Car Library's– Manifest– Exploring the Car Library's Types– Building the Multi-file Assembly– Using Assembly– Understanding Private Assemblies– Probing for Private Assemblies (The Basics) – Private Assemblies XML Configurations Files– Probing for Private Assemblies (The Details) – Understanding Shared Assembly – Understanding Shared Names– Building a Shared Assembly– Understanding Delay Signing– Installing/Removing Shared Assembly. Building Windows application –Working with C# controls– Event handling – Graphics Device Interface (GDI).

UNIT-IV

12Hrs.

ADO.NET and Database Connectivity: Introduction to ADO.NET– Major Components of ADO.NET– Establishing Database Connections– Connection objects– Command objects– Datasets– Data readers– Querying databases– Data Grid Views– Data Validation.

REFERENCE BOOKS:

1. Stephen C. Perry – "Core C# and .NET", Pearson Education, 2006.
2. S. ThamaraiSelvi and R. Murugesan—"A Textbook on C#" —, Pearson Education, 2003.
3. Andrew Troelsen, Pro C# with .NET 3.0 Special Edition, Dream tech Press, India, 2007.
4. E. Balagurusamy, Programming in C#, 5th Reprint, Tata McGraw Hill, 2004. (ForProgramming Examples)
5. Tom Archer, Inside C# WP Publishers, 2001.
6. Herbert Scheldt, C#: The Complete Reference, Tata McGraw Hill, 2004.
7. Robinson et al, -"Professional C#", Fifth Edition, Wrox Press, 2002.