

C# PROGRAMMING AND .NET

Subject Code: 10IS761
Hours/Week : 04
Total Hours : 52

I.A. Marks : 25
Exam Hours: 03
Exam Marks: 100

PART – A

UNIT – 1

6 Hours

The Philosophy of .NET: Understanding the Previous State of Affairs, The .NET Solution, The Building Block of the .NET Platform (CLR,CTS, and CLS), The Role of the .NET Base Class Libraries, What C# Brings to the Table, An Overview of .NET Binaries (aka Assemblies), the Role of the Common Intermediate Language, The Role of .NET Type Metadata, The Role of the assembly Manifest, Compiling CIL to Platform – Specific Instructions, Understanding the Common Type System, Intrinsic CTS Data Types, Understanding the Common Language Specification, Understanding the Common Language Runtime A tour of the .NET Namespaces, Increasing Your Namespace Nomenclature, Deploying the .NET Runtime.

UNIT – 2

6 Hours

Building C# Applications: The Role of the Command Line Compiler (csc.exe), Building C# Application using csc.exe Working with csc.exe Response Files, Generating Bug Reports, Remaining g 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, an Interesting Aside: The System. Environment Class.

UNIT – 3

8 Hours

C# Language Fundamentals: The Anatomy of Basic C# Class, Creating objects: Constructor Basics, The Composition of a C# Application, Default assignment and Variable Scope, The C# Member Initialisation Syntax, Basic Input and Output with the Console Class, Understanding Value Types and Reference Types, The Master Node: System, Object, The System Data Types (and C# Aliases), Converting Between Value Types and Reference Types: Boxing and Unboxing, Defining Program Constants, C# Iteration Constructs, C# Controls Flow Constructs, The Complete Set of C# Operators, Defining Custom Class Methods, Understating Static Methods, Methods Parameter Modifies, Array Manipulation in C#, String Manipulation in C#, C# Enumerations, Defining Structures in C#, Defining Custom Namespaces.

UNIT – 4

6 Hours

Object- Oriented Programming with C#: Forms Defining of the C# Class, Definition the “Default Public Interface” of a Type, Recapping the Pillars of

OOP, The First Pillars: C#'s Encapsulation Services, Pseudo- Encapsulation: Creating Read-Only Fields, The Second Pillar: C#'s Inheritance Supports, keeping Family Secrets: The “ Protected” Keyword, Nested Type Definitions, The Third Pillar: C #’s Polymorphic Support, Casting Between .

PART – B

UNIT – 5

6 Hours

Exceptions and Object Lifetime: Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception Handling, the System. Exception Base Class, Throwing a Generic Exception, Catching Exception, CLR System – Level Exception(System. System Exception), Custom Application-Level Exception(System. System Exception), Handling Multiple Exception, The Family Block, the Last Chance Exception Dynamically Identifying Application – and System Level Exception Debugging System Exception Using VS. NET, Understanding Object Lifetime, the CIT of “new”, The Basics of Garbage Collection,, Finalization a Type, The Finalization Process, Building an Ad Hoc Destruction Method, Garbage Collection Optimizations, The System. GC Type.

UNIT – 6

6 Hours

Interfaces and Collections: Defining Interfaces Using C# Invoking Interface Members at the object Level, Exercising the Shapes Hierarchy, Understanding Explicit Interface Implementation, Interfaces As Polymorphic Agents, Building Interface Hierarchies, Implementing, Implementation, Interfaces Using VS .NET, understanding the IConvertible Interface, Building a Custom Enumerator (IEnumerable and Enumerator), Building Cloneable objects (ICloneable), Building Comparable Objects (I Comparable), Exploring the system. Collections Namespace, Building a Custom Container (Retrofitting the Cars Type).

UNIT – 7

8 Hours

Callback Interfaces, Delegates, and Events, Advanced Techniques: Understanding Callback Interfaces, Understanding the .NET Delegate Type, Members of System. Multicast Delegate, The Simplest Possible Delegate Example, , Building More a Elaborate Delegate Example, Understanding Asynchronous Delegates, Understanding (and Using)Events.
The Advances Keywords of C#, A Catalog of C# Keywords Building a Custom Indexer, A Variation of the Cars Indexer Internal Representation of Type Indexer . Using C# Indexer from VB .NET. Overloading operators, The Internal Representation of Overloading Operators, interacting with Overload Operator from Overloaded- Operator- Challenged Languages, Creating Custom Conversion Routines, Defining Implicit Conversion Routines, The Internal Representations of Customs Conversion Routines

UNIT – 8**6 Hours**

Understanding .NET Assemblies: Problems with Classic COM Binaries, 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 CarLibrary's, Manifest, Exploring the CarLibrary's Types, Building the Multifile 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, Using a Shared Assembly

Text Books:

1. Andrew Troelsen: Pro C# with .NET 3.0, 4th Edition, Wiley India, 2009.
Chapters: 1 to 11 (up to pp. 369)
2. E. Balagurusamy: Programming in C#, 2nd Edition, Tata McGraw Hill, 2004.
(Programming Examples 3.7, 3.10, 5.5, 6.1, 7.2, 7.4, 7.5, 7.6, 8.1, 8.2, 8.3, 8.5, 8.7, 8.8, 9.1, 9.2, 9.3, 9.4, 10.2, 10.4, 11.2, 11.4, 12.1, 12.4, 12.5, 12.6, 13.1, 13.2, 13.3, 13.6, 14.1, 14.2, 14.4, 15.2, 15.3, 16.1, 16.2, 16.3, 18.3, 18.5, 18.6)

Reference Books:

1. Tom Archer: Inside C#, WP Publishers, 2001.
2. Herbert Schildt: C# The Complete Reference, Tata McGraw Hill, 2004.