Navodaya Computer Saksharta Mission®

A National Literacy Programme of Information Technology & Skill Development

Digital India
Power To Empower

Member of
Quality Council of India
Computer Society of India

AN ISO 9001 : 2008 CERTIFIED ORGANIZATION

An Autonomous Institution Registered Under Planning Commission-Trust & Societies Act. NCT New Delhi
RJ 2013/00056856, Section 60 (B) 4 VOL 2901/1878 Act,1882 & ROS/North/091/2010 Act,1860
Ministry of HRD (Department of Higher Education) Courses Registered Under CR Act

Government of India

Appreciatated by
President | Prime Minister | Vice-President
FMO | MHA | MHRD | MCIT | MSME | MSJE | MCA | MMA | CM | Governer

CC-C#

Certificate Course in C

COURSE CODE : CC-038

COURSE CONTENT & SYLLABUS

H.O. : 3-KHA-4, Sector 3, Vigyan Nagar, Kota-324005, (Rajasthan) India
Tel.: 0744-2412009 Fax: 0744-2411150 Mob. +91 94629 67201
visit us : www.navodayaindia.in | www.ncsm.in | E-mail : contact@ncsm.in, info@ncsm.in

CC-C#

Certificate Course in C #

COURSE CONTENTS

COURSE CODE – CC-038 **ELIGIBILITY : 12TH CLASS & ABOVE**

DURATION : 3 MONTH

- MS.NET FRAMEWORK INTRODUCTION
- DEVELOPING CONSOLE APPLICATION
- LANGUAGE BASICS
- INTRODUCTION TO OBJECT ORIENTED FEATURES
- PROGRAMMING OBJECT ORIENTED – ENCAPSULATION
- INHERITANCE
- INTERFACE & POLYMORPHISM
- EXCEPTION HANDLING
- WORKING WITH COLLECTIONS AND GENERICS
- OPERATOR OVERLOADING, PARTIAL CLASS, ATTRIBUTES, REFLECTION, CONFIGURATION
- WORKING WITH COMPONENTS / ASSEMBLIES
- DATA STREAMS AND FILES
- WIN FORMS
- DATA ACCESS USING ADO.NET
- DATA ACCESS USING ADO.NET – DATASET
- WINDOWS SERVICES
- XML
- MULTITHREADING
- DEBUGGING AND TRACING
- DELEGATES & EVENTS

CC-C#

Certificate Course in C

COURSE SYLLABUS

MS.NET FRAMEWORK INTRODUCTION

A. The .NET Framework - an Overview
B. Framework Components
C. Framework Versions
D. Types of Applications which can be developed using MS.NET
E. MS.NET Base Class Library
F. MS.NET Namespaces
G. MSIL / Metadata and PE files.
H. The Common Language Runtime (CLR)
I. Managed Code
J. MS.NET Memory Management / Garbage Collection
K. Common Type System (CTS)
L. Common Language Specification (CLS)
M. Types of JIT Compilers
N. Security Manager

DEVELOPING CONSOLE APPLICATION

A. Introduction to Project and Solution in Studio
B. Entry point method - Main
C. Compiling and Building Projects
D. Using Command Line Arguments
E. Importance of Exit code of an application
F. Different valid forms of Main
G. Compiling a C# program using commandline utility CSC.EXE

LANGUAGE BASICS

A. Why Datatypes
B. Global, Stack and Heap Memory
C. Common Type System
D. Reference Type and Value Type
E. Datatypes & Variables Declaration

F. Implicit and Explicit Casting
G. Checked and Unchecked Blocks – Overflow Checks
H. Casting between other datatypes
I. Boxing and Unboxing
J. Enum and Constant
K. Operators
L. Control Statements
M. Working with Arrays
N. Working with Methods
O. Pass by value and by reference and out parameters
P. Misc...

INTRODUCTION TO OBJECT ORIENTED FEATURES

A. What is an Object?
B. What is not an Object?
C. What is state of an Object?
D. What is the Lifecycle of an Object?
E. How will you distinguish two Objects?
F. What is the relationship between Class and Object?
G. Define Application using Objects?
H. Principles of Object Orientation
I. Encapsulation
J. Inheritance
K. Polymorphism
L. Encapsulation is binding of State and Behavior together
M. Inheritance is based on "is a" relationship
N. Understanding Polymorphism with Examples

PROGRAMMING OBJECT ORIENTED – ENCAPSULATION

A. Write a class and add field members to the class
B. Create an Object of the Class and understand the difference between object and reference
C. Access the members of the object
D. Copy the reference in another reference variable
E. Abandoning the object
F. Working with Methods
G. Working with Properties
H. Constructor & Destructor
I. Working with "static" Members

INHERITANCE

A. Inheritance and "is a" relationship

B. Protected Keyword

C. Constructor in Inheritance

D. Type Casting of Reference Types

E. Static and Dynamic Binding and Virtual Methods

F. Abstract Class

G. Object as Parent of all classes

Interface & Polymorphism

A. What is Interface

B. Syntax for Implementation of Interface

C. Explicit Implementation of Interface members

D. Types of Inheritance

Exception Handling

A. What is Exception

B. Rules for Handling Exception

C. Exception classes and its important properties

D. Understanding & using try, catch keywords

E. Throwing exceptions

F. Importance of finally block

G. "using" Statement

H. Writing Custom Exception Classes

Working with Collections and Generics

A. Importance of List and ID ictionary

B. Using Array List and Hash table

C. Understanding IE numerable and IE numerator

D. Sorting Items in the collection using I Comparable

E. Type safety issue with Array List and Hash table classes

F. Writing custom generic classes

G. Working with Generic Collection Classes

Operator Overloading, Partial Class, Attributes, Reflection, Configuration

A. Operator Overloading

B. Partial Classes

C. Importance of Attributes

D. Reflection

E. Configuration Files

Working with Components / Assemblies

A. What is a DLL and how is it different from EXE

B. Types of DLL
C. About Assemblies
D. How to build a Class Library?
E. How to use a Class Library in another Application?
F. What is Namespace?
G. Internal Access Specifier
H. Private Assemblies
I. Shared Assemblies

DATA STREAMS AND FILES

A. Introduction to streams
B. Understanding the difference between Text Stream and Binary Stream
C. To understand importance of Encoding in Text Streams
D. System.IO and Base classes of Stream
E. Standard / Console I/O Streams
F. Sample Programs
   a. Program to Read from Console
   b. Program to Write to Console
   c. Reading and Writing to File
   d. Reading and Writing to Text File
   e. Reading and Writing to Binary File
E. Working with File System of HDD
a. File & File Info
b. Directory & Directory Info
G. Serialization & Deserialization

WINFORMS

A. Introduction
B. Controls
C. Menus and Context Menus
D. Menu Strip, Toolbar Strip
E. Graphics and GDI
F. SDI and MDI Applications
G. Dialog box (Modal and Modeless)
H. Form Inheritance
I. Developing Custom, Composite and Extended Controls
J. Other Misc topics
K. Working with Resource Files
L. Working with Settings

DATA ACCESS USING ADO.NET

A. Session 1:
   1. Introduction to SQL
   2. Evolution of ADO.NET after Native Drivers, ODBC Drivers, DAO/RDO and ADO
   3. About Managed Providers
   4. Important Objects in Managed Provider

B. Session 2:
   1. Creating Database using VS.NET
   2. Establishing Connection with Database
   3. Connection String Formats

C. Session 3:
   1. Executing simple Insert, Update and Delete Statements
   2. Executing Select Statement and using Sql Data Reader
   3. Multiple Active Result Sets(MARS)

D. Session 4:
   1. Prepared Statements
   2. Stored Procedures

E. Session 5:
   1. Managing Transactions
   2. Asynchronous Execution of Sql Statements
   3. To write provider independent code

DATA ACCESS USING ADO.NET – DATASET

A. What is Dataset?

B. Advantages of Data Set

C. Data Set Object Model

D. Programming Data Set - Walkthrough
   1. Fetching data using Fill methos of Data Adapter and filling data into Dataset to create a Data Table
   2. Showing Data Table in Data Grid View Fetching data from Data Row
   3. To get a Data Row from a collection for rows using Select method of Data Table and specifing the condition
   4. Updating data using Update method of Data Adapter
   5. Using Sql command builder to automatically generate Commands of Data Adapter
   6. Adding / Editing / Deleting rows in the Data Table Programma ti cally
   7. Handling Data Adapter events
   8. Handling Concurrency while updating data to the database
   9. Working with Data View

E. Purpose of Creating Foreign Key Contrainst and adding the same to Data Table
F. Working with Data Relation
G. Creating Data Set / Data Table Dynamically (without Data Adapter)
H. Set Primary Key for a Data Table Programmatically
I. Working with Typed Data Set

WINDOWS SERVICES

A. Purpose and Advantage
B. Developing and Deploying
C. Debugging Widows Service
D. Sending Custom events

XML

A. Introduction
B. XML-DOM
C. XML Document, XML Element, XML Attribute
D. Interoperating with Data Set
E. XML Data Document
F. XML Text Reader / XML Text Writer
G. X Path

MULTITHREADING

A. Introduction
B. Application Domains
C. Creating and Managing Threads
D. Threads Priority
E. Thread States
F. Thread Synchronization & Inter-thread Communication
G. Using Monitor

DEBUGGING AND TRACING

A. Debug and Release Compilation
B. Debug and Trace Objects
C. Tracing Switches & Listeners
D. Debugging JavaScript from VS.NET
E. Debugging SQL-Server Stored Procedures

DELEGATES & EVENTS

A. Introduction
B. Delegate Declaration
C. Sample Application

D. CHAT APPLICATION USING DELEGATES

E. UNDERSTANDING += AND -= OPERATOR (EVENTS)

F. CHAT APPLICATION USING DELEGATES AND EVENTS

G. GENERAL SYNTAX FOR DELEGATES AND EVENTS

H. ANONYMOUS METHODS

