

Full Stack DOT Net Professionals

Duration: 150 Hrs.

C#.Net

DOT.NET Framework Introduction

- The .NET Framework - an Overview
- Framework Components
- Framework Versions
- Types of Applications which can be developed using DOT.NET
- DOT.NET Base Class Library
- DOT.NET Namespaces
- MSIL, Metadata and PE files
- The Common Language Runtime (CLR)
- Managed Code
- Memory Management and Garbage Collection
- Common Type System (CTS)
- Common Language Specification (CLS)
- Types of JIT Compilers
- Security Manager

DOT.NET and Entry Point Method –Main

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

C # Language Syntax

- Basics of Data type
- Global, Stack and Heap Memory
- Common Type System
- Reference Type and Value Type
- Data types & Variables Declaration
- Implicit and Explicit Casting
- Checked and Unchecked Blocks
- Casting between other data types
- Boxing and Unboxing
- Enum and Constant
- Operators
- Control Statements
- Working with Arrays
- Working with Methods
- Pass by value and by reference and out parameters
- this and base keyword
- Indexer
- Properties-Read only & Write Only Properties
- Static keyword, static constructor

- Nullable Type
- Object Initializer
- C# Structure
- Access Specifiers
- String Handling

OOPs-Concepts

Encapsulation

- Understanding Encapsulation Concept through an example

Inheritance

- Introduction to Inheritance
- Constructor & Inheritance
- Type Casting of Reference Types
- Static and Dynamic Binding
- Abstract Class

OOPs-Interface & Polymorphism

- Introduction to Polymorphism
- Overview of Interface
- Interface with examples
- Types of Inheritance Collections and Generics
- Introducing Collections
- Benefits of Collection Classes
- Understanding and using commonly used

Collections & Generics

- Advantages of Generics
- How Generics Work at Runtime
- Constraint on Type Parameters
- Generic Methods
- Generic Collections
- Selecting a Collection Class

Assemblies and GAC

- What is a DLL and how is it different from EXE
- Types of DLL
- What is an Assembly Assemblies
- How to build a Class Library?
- How to use a Class Library in another Application?
- What is Namespace?
- Internal Access Specifier
- Types of Assemblies
- Global Assembly Cache

Exception Handling

- Defining Exception
- Understandings try and catch keywords
- Using “finally” block
- “using” statement
- Throwing exceptions
- Creating User defined
- Custom Exception class

IO Streams

- Definition of stream
- Types of Stream
- Standard I/O Streams – Console
- Handling text in files
- Dealing with Binary files
- Serialization & Deserialization

Unsafe Code Reflection and Attributes

- Definition of Reflection
- Using Reflection to read type information
- Attributes
- Pre-defined Attributes
- Custom Attributes
- Using Reflection to read custom attributes

More on Classes

- Operator Overloading
- Partial Classes and Methods
- Anonymous Types
- Extension Methods
- Caller Method Information
- Configuration File

Developing GUI Application Using WINFORMS

- Basic Controls
- Panel & Layouts
- Drawing and GDI Devices
- MenuStrip, ToolStrip and ContextMenuStrip
- Model and Modeless Dialog boxes
- Mutiple Document Interface(MDI)
- Form Inheritance
- Building Login Form
- Working with Resource Files and Setting
- Notify Icon Controls
- Using Components like Timer, FileSystemWatcher, Process, BackgroundWorker
- Drag and Drop
- Working with Advanced Controls like TreeView and ListView

Delegates & Events

- Introduction to Delegates
- Events Declaration, Raising and Handling
- Anonymous Methods

Multithreading

- Threading Overview
- Scheduling
- Thread States
- Programming Threads
- Methods of Thread Class
- Thread Pool
- Thread Synchronization
- Monitor
- Mutex
- Semaphore
- Events
- Asynchronous Programming using async and wait keywords

Database Programming Using ADO.NET

- Introduction to ADO.NET
- Understanding the Role of Managed Provider and ADO.NET Objects
- Connecting to Database and Connection Pooling
- Performing Insert, Update and Delete Operations
- Fetching Data from database - Executing Select Statements
- How to implement Login facility with database
- Use of Multiple Active Result Sets
- Parameterized Prepared Statements
- Inserting Image into Database table
- Executing Stored Procedure
- Writing Common Code for Execution of Stored Procedures

Managing Data using DataSet

- Introduction DataSet and its Object Model
- Filling DataSet using DataAdapter
- Binding DataSet to DataGridView
- Updating changes to database using DataAdapter
- Managing DataTable Programmatically
- DataAdapter events
- Working with DataViews
- Constraints in DataTable
- Using DataRelations object
- Creating DataSet and DataTable dynamically
- Working with Typed DataSet

Debugging and Diagnostics

- Definition of Debugging
- List of Debugging Windows
- Break Point Hit Count and Condition

- Debugging Exception

Windows Form

- Creating Word Processor
- Creating Namespace (Class Library)
- Windows Form Control:
 - DateTimePicker
 - Progressbar
 - Masked TextBox
 - ImageList
 - Data Gridview
 - ListBox
 - Error Provider
 - StatusStrip
 - Tab Control
 - Web Browser
 - PrintForm

ASP.NET

Web Programming Introduction

- Understanding role of Web Server and Web Browser.
- Brief about HTTP Protocol.
- HTTP Request structure.
- Form Tag and comparison between Get and Post methods.
- HTTP Response Structure

ASP.NET Introduction & Controls

- ASP.NET Introduction
- First ASP.NET Application.
- Auto Postback Property.
- Event Handler Parameters.
- Dynamically initializing Controls
- IsPostBack property of Page class
- ListControls
- Comparison between HtmlControls and WebControls
- Control Properties and Methods
- FileUpload Control

ASP.NET Architecture

- What is AppDomain
- Life Cycle of ASP.NET Page
- How Control Manages its State
- What is EnableViewState
- How Control raises events
- Event handling in Web Forms
- Writing Custom Classes in WebApplication

CSS & Themes

- Work with CSS
- Use Themes to Customize a Site
- Name Skins within a Theme
- Do server-side Styles using Themes
- Add contents of a Theme and Skin
- Apply themes and Profiles.

Redirecting User to Another Page

- Redirecting Options
- Response.Redirect
- Server.Transfer
- Cross Page Postback
- Determine how the pages was invoked
- Passing values between pages

Server Control

- Label, Literal, TextBox, RadioButton, RadioButtonList, ChckeBox, CheckBoxList, DropDownList, Button, LinkButton, ImageButton, Image, Hyperlink, Calendar, ImageMap, MultiView, View, ListBox, Wizard, Panel, Menu.
- Adrotator Control.
- File Download.
- SiteMap, SitemapPath Control, TreeView Control.

Master Pages

- Introduction to MasterPage
- ContentPlaceHolder and Content tags
- Accessing controls of MasterPage in ContentPage
- URL's in MasterPages
- UniqueID and ClientID

Working with User Control

- Overview of User Controls
- Creating a User Control
- Adding Properties to User Control
- Adding Events to User Control
- Using User Control in Web Form
- Rendering Clients Scripts Using Page. ClientScript methods

Validation Controls

- Base Validator
- Required Field Validator
- Compare Validator
- Range Validator
- Regular Expression Validator
- Custom Validator
- Causes Validation Property
- Grouping - Validation Group Property
- Page.Validators and Page.IsValid

ASP.NET State Management

- Static Members
- View State
- Hidden Field in Form
- Query String
- HttpContext
- Cookies-HttpCookie
- Sessions-HttpSessionState
- Application-HttpApplicationState
- Application State variables.
- Summary of All Features

WebConfiguration File and Global.asax

- Introduction to Configuration files
- Page setting in web.config
- Custom Errors
- URL Re-Writing
- Tracing
- Using ConfigSource Attribute
- Using Location Section
- HttpApplication class-Global.asax

Authentication & Authorization

- What is Authentication and Authorization
- Types of Authentication
- Forms Authentication
- Role based Authentication
- Windows and Basic Authentication
- What is ASP.NET Impersonation
- Using location section in web.config

Data Bound Controls

- Databinding traditional way
- SqlDataSource
- GridView
- DetailsView
- FormView
- DataList
- Repeater
- ListView
- DataPager
- ADO.NET

AJAX.NET

- What is AJAX
- What is AJAX.NET
- Script Manager
- Update Panel
- Update Progress
- Timer
- AJAX Control Toolkit

- Web Services.
- XML Web Server Control.
- Writing and Reading XML.
- XSLT.
- Page Level & Application Level Error Handling.
- LINQ.

Understanding & Publishing Web Application

- Introduction to ASP.NET Web Application
- Advantages of IIS Applications
- Creating web application in IIS
- Converting File System application to IIS Application
- Using Virtual Directory
- Publishing ASP.NET Website
- Culture specific formatting
- WCF Introduction.
- WPF Introduction

ASP.NET MVC

- Introduction of Web Technology
- What is ASP.NET MVC
- Role of Model, View, and Controller
- How ASP.NET MVC Works
- Benefits of using ASP.NET MVC

Getting Started with MVC

- ASP.NET MVC project templates
- Understanding the structure of an ASP.NET MVC project
- Naming conventions
- Creating views
- Defining controllers
- Defining a data model
- Overview of coding standards follows during programming

Creating an application in MVC

- Creating strongly-typed views
- Understanding URLs and action methods
- Using HTML helpers
- Handling form post-backs
- Data validation

Razor View Engine

- Razor Basics
- Razor design goals
- Implementation of Razor view
- Razor syntax
- Accessing Model Data in Razor views

Strength ASP.NET MVC Applications

- ASP.NET application architecture best practices
- Implementing a Repository and Entity Framework Data Model

Designing and Implementing Databases with SqlServer

- Introduction to ADO.NET
- Creating Tables and Relationships
- SQL Fundamentals
- Stored Procedures
- Insert, Update, Delete, Select commands

View Techniques

- Defining and using custom HTML Helpers
- Defining a layout / MVC Master Page
- Using Styles
- Defining and using partial views
- Razor Helper Method syntax

Implementing Navigation in MVC web apps

- Defining view-model classes
- Implementing Data Filtering in a Controller
- Understanding the Routing mechanism
- Adding custom entries to a route table
- Defining defaults, parameters, and validation
- Generating URLs and Hyperlinks
- Custom Route constraints

MVC State Management

- Using hidden fields
- Session and Application State

Using AJAX and jQuery with ASP.NET MVC

- Overview of AJAX and ASP.NET MVC
- Unobtrusive AJAX
- Using AJAX Action Links

ASP.NET Web API with MVC

- Overview of the ASP.NET Web API
- Building servers and clients

FTP Management

- Uploading and downloading FTP contents

Sending Emails

- Designing email panel
- How to send an email to various users
- Sending auto emails

Deployment

- Deploying application on Web Server

Project

- Getting customer's requirements
- Preparing database and business logics
- Developing application
- Testing and implementing the project
- Troubleshooting the project application after implementation