C# .NET COURSE CONTENT

Our Strength
Duration: 10 Hours   **Shiva Sir**
Prerequisite: C++   **0731-4069788**

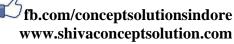| C# Syllabus	❖ **Introduction to .NET 5.0** Introduction of Language, Introduction of Web Application, Platform Dependence , Independence & Portability, Introduction of .NET Framework, Introduction of Compiler & MSIL, Introduction of CLR, Introduction of CLS, CTS, CLI ❖ **OOPS Implementation** Encapsulation, Abstraction, Class & object , Relation b/w objects and reference variables, Method Overloading & Types of Methods, Description of Memory blocks: Stacks, Heap & Class Area, Constructors and Destructor & Garbage Collection,Using Scope &, Dispose Method, Usage of this keyword , Static Data members, Constructors & methods, Relation & Implementing Has-A relation, Implementing Is-A relation using Inheritance, Method Overriding & Hiding, Usage of Base keyword, Static & Dynamic binding, Runtime polymorphism and generalization, Abstract classes & methods, Interfaces and Role based Inheritance, Usage of sealed, partial, out, ref, pram keyword, Namespace, Assembly & GAC, Property & Indexer, Windows Application, Delegates & Event, Event Handling Windows App. Component., Object class and its role ❖ **String Handling** String class & concept of Immutability, Pool of String Constants & its use, Common String operations, Use of String Builder. ❖ **Exception Handling** Difference b/w exception and error, Exception Handling & Robustness, Common, Exceptions and Errors, Try and catch block, Exception handlers, Throw keyword and its usage, Role of finally, Exception Class Methods & properties, Creating User defined Exceptions ❖ **Multithreading in .NET** Multitasking and Multithreading, Process & thread, States of ThreadThread based execution model of .NET Applications, Implementing User threads, Thread synchronization, Inter Thread communication, Thread Interrupting, Daemon Threads and their use ❖ **Input Output Streams** Streams & their advantages over conventional input output, Reading & writing data byte by byte, line by line, and in one go. Redirecting Standard input & output streams, Serialization & Deserialization, Serialization Formatter, Xml Formatter Binary Formatter, Soap Formatter, Json Formatter, File class and File System Interaction, Zip & Unzip Files ❖ **Reflection** Understanding object representation of classes, Loading classes dynamically, Instantiating classes indirectly, Obtaining information of classes dynamically Invoking methods indirectly, Referencing data members indirectly, Overriding access policy and referencing private members ❖ **Collection Framework** Role and Importance of Collection Framework, Index Based Collection Array List, Stack, Queue, Use Define Collection, Map Based Collection Use of Key & Value pair (Entry), Hash Table, Sorted List, Searching elements in List, Hash and Tree based collections, Role of equals and hashCode() methods, Role of Comparable interfaces, Type safety and Generics, Generic Collection, List ,Stack ,Queue , Dictionary,Linked List ,Sorted List.

.NET MVC CORE COURSE CONTENT

Our Strength

Duration: 30 Hours　　　　　　　　　　　　　　　　　　　**Shiva Sir**

Prerequisite: C#　　　　　　　　　　　　　　　　　　　　**0731-4069788**

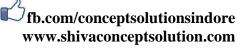|	
CORE.NET Syllabus	❖ **Introduction to .ASP.NET Core** What is .NET Core, Difference between NET Core and NET CORE, Core Features, Difference between ASP.NET Web Forms, CORE and Core ❖ **Getting Started with ASP.NET CORE** Environment Setup, Create First Empty application, Directory Structure, ASP.NET CORE project templates, Understanding the structure of an ASP.NET CORE project, Naming conventions, Creating views, Defining controllers, Defining a data model Pass data from View to Controller, Get data from Controller to view, Route table, Redirection, type of Action result, Http Verb ❖ **Creating a Complete ASP.NET CORE Application** Creating, strongly-typed views, Understanding URLs and action methods, Using HTML helpers, Handling form post-backs, Data validation. ❖ **Using the Razor View Engine** Getting started with Razor, Razor design goals, Implementing a Razor view Razor syntax Accessing Model Data in Razor views ❖ **Industrial-Strength ASP.NET CORE Application** ASP.NET application architecture best practices, Implementing a Repository and Entity Framework Data Model, Using Dependency Injection, Implementing a custom controller factory ❖ **Test Driven Development (TDD) with ASP.NET CORE 5.0** Recap of TDD, Unit Testing Frameworks TDD Hand-on with ASP.NET CORE ❖ **View Techniques** Defining and using custom HTML Helpers defining a layout/CORE Master Page Using Styles, Defining and using partial views, Razor Helper Method syntax ❖ **Implementing Navigation in CORE 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

.NET MVC CORE COURSE CONTENT

- ❖ **CORE State Management**
  Using hidden fields, Session and Application State, Custom model bindings

- ❖ **Additional Techniques**
  View Scaffold Templates, Controller Scaffold Templates, Global Filters, Model binding, Asynchronous Controllers
- ❖ **Security**
  Authentication & Authorization
  Implementing Security using ASP.NET Core Identity

- ❖ **Web Caching**
  Cache Tag Helpers, Memory Caching Introduction, In-memory caching, Response Cache

- ❖ **Web API & AJAX**
  Introduction to API, AJAX implementation using JQUERY, CRUD Operation using API, JSON Serialialization in API, Overview of AJAX and ASP.NET CORE, Unobtrusive AJAX, Using AJAX Action Links Overview of jQuery, jQuery Techniques, Using jQuery UI

   ASP.NET CORE 5.0 & LINQ – working with Data
  Language features used in LINQ, Creating simple LINQ queries, Using LINQ queries in a Web application, Defining a data repository, Using LINQ to define a data access component

   ASP.NET CORE 5.0 Techniques & Best
   View scaffold templates, Controller scaffold templates, Dependency injection

.NET MVC CORE COURSE CONTENT

Our Strength

Duration: 30 Hours

Shiva Sir

Prerequisite: C#

0731-4069788