

.NET FULL STACK



25 YEARS OF
ACADEMIC EXCELLENCE

Visual C#

Introduction to .Net

- .Net Framework
- Features of .Net
- Common Type System
- Common Language Specification
- Common Language Runtime
- MSIL
- Base Class Library
- Assemblies
- Garbage Collection
- Stack and Heap
- Architecture of GC
- Application Domain

.Net Framework 3.0 and 3.5 features

- WPF - Windows Presentation Foundation
- WCF - Windows Communication Foundation
- WWF - Windows Workflow Foundation
- Silver light
- Multi-Targeting
- Multi Framework Support
- LINQ

.Net Framework 4.0 features

- Cloud
- ASP.NET MVC web application
- Sharepoint Project
- Silverlight project

.Net Framework 4.5 features

- Metro UI application Development
- Gaming applications
- HTML5 and CSS 3 integration
- Semantic code analysis

Visual C# Language

- Introduction to VC#
- Features of VC#
- Data Types
- Value Types and Reference Types
- Type Conversion
- Boxing and UnBoxing
- Basic Programming constructs
- Statements & Expressions
- Methods, Arrays
- Structures and Enumerations

Object Oriented Programming with VC#

- Classes and Objects
- Interface
- Data Encapsulation, Data Abstraction
- Fields and Properties
- Access Modifiers
- Abstract Classes and Sealed Classes
- Constructor and Destructor
- Static and Instance Members
- Inheritance
- Method overloading and overriding
- Operator Overloading
- Delegate – Unicast and Multicast
- Event Handling
- Collection, Dictionaries, String, String builder
- Indexers
- Attributes, Namespaces, Generics
- Anonymous methods, Iterators
- Partial Types
- Nullable Types

Error Handling

- Structured Error Handling
- Debugging the application
- Debug and Trace classes

Assemblies

- Introduction to Assemblies
- Disadvantages of COM
- Creating Private and Shared Assemblies
- Strong name, GAC, COM Interoperability
- Satellite Assembly

.NET
FULL STACK

◆ SQL Server

Concepts

- Database concepts
- RDBMS Concepts
- Data Models
- Constraints
- Normalization- 1NF,2NF,3NF,BCNF etc
- ER Diagrams

Sql Server Concepts

- Introduction to Sql Server 2008 R2
- Sql Server 2008 Features
- Server types
- Editions
- Components
- Management Studio
- T-SQL Features
- New Operators
- DML Statements

SQL

- DDL
- Data Types
- System Defined data types
- User defined Data
- Create/modify/delete tables
- Create/modify/delete tables
- Operators
- DML
- DCL
- Using Select Statement

Maintaining Database Integrity

- Constraints
- Rules
- Entity Integrity
- Domain Integrity
- Referential Integrity

Sub Queries

- Built-in function
- Nested sub queries
- Correlated sub queries
- Derived tables
- Recursive queries
- Common table expression

Joins

- Cross join, inner join, self join, outer join

Indexes

- Types of Indexes
- Creating indexes
- Modifying indexes
- Dropping indexes

Views

- About View
- Creating Views
- Altering Views
- Materialized views
- Distributed view

Stored Procedures

- Advantages
- System stored procedures
- Extended stored procedures
- Parameter handling

UI Technologies

- HTML and HTML 5
- CSS
- Javascript
- JQuery
- Bootstrap
- DOM

.NET
FULL STACK

ASP.NET

Introduction to Web Application

- About Web Server
- IIS Web Server 10.0
- Web Support Languages
- Web Browsers
- About Http, TCP/IP, LAN, WAN
- Scripting and Its Importance
- Client-side and Server side Scripting
- HTML
- Java Script
- CSS
- JQuery

Active Server Pages.Net

- Moving from ASP to ASP.NET
- ASP.NET Namespaces
- ASP.NET – New Scenario in Development
- Process with IDE-VS.NET IDE
- Web Pages to Web Forms
- HTML Controls
- ASP.NET Controls

ASP.NET Web Forms

- Introduction to Web Forms, Worker process
- Applying styles to controls, themes, skins
- Server control Form
- Page Directive
- Separating code and Design
- New code behind techniques
- Implementing asp.net server controls
- Validation controls

Cookies

- Defining and using collection of cookies
- Limitations of cookies

ADO.NET

- Introduction to ADO.NET
- ADO.NET Architecture
- ADO.NET managed providers
- Connection and Command Objects
- DataReader
- DataAdapter and DataSet
- DataRelations and DataSet
- Connected and Disconnected Environment
- Connection Pooling

- ADO.NET Exceptions
- using Stored Procedures
- N-tier Application
- ADO.NET and XML
- LINQ
- Linq to object
- Linq to sql
- Linq to XML
- ADO.NET Entity Framework
- Object Relational Mapping

ASP.NET Error Handling & Debug

- Page level error handling
- Application level error handling
- Debug using CLR debugger
- User Controls
- Master Pages

Configuring ASP.NET application

- Web Application Configuration
- Web Application Machine configuration
- Global Assembly Cache
- Working with AssemblyInfo
- Web .config

ASP.NET web applications

- About Session and Application object
- Using the Global.asax file
- Managing Application State
- Http Handlers
- Application and session variable
- Application and session events
- Cache services
- Caching overview
- Advantages of Caching
- Page output caching
- Page data caching
- Fragment Caching

.NET
FULL STACK

Security

- Authentication and Authorization
- Windows, Forms, Password authentication
- Authoring users and Roles
- User Account Impersonation

ASP.NET XML Web Services

- Introducing XML web Services
- Difference between DCOM/Remoting/Web services
- ASP.NET Web Services
- Xml web services stack, SOAP, WSDL, UDDI, Sync and Async Calls
- Using Data in Web services
- Using Objects and Intrinsic

◆ ASP.NET MVC

- Design Patterns
- ASP.NET MVC 5 Introduction
- ASP.Net MVC basic architecture
- Difference between ASP.Net and ASP.NET MVC
- ASP.NET MVC Basics
 - o Controller
 - o Action Methods
 - o View
 - o Model
 - o MVC Architecture
 - o View Engine
 - o Razor View Engine & Syntaxes
 - o Execution Flow of MVC
- HTML Helpers
- Templates
- Return Types of Action Methods
- URL Routing
- Advance Routing using attributes in ASP.Net MVC 5
- Wrapper Classes
- Working with Forms in MVC
- Scaffold Templates
- CRUD (Create, Retrieve, Update, Delete Operations on a db table)
- Data Annotations
- Action Filters
- ASP.NET MVC Execution flow or advanced architecture
- Entity Framework
- ADO.Net Entity Data Model
- AutoMapper
- jQuery-AJAX in MVC

- ASP.Net MVC controls
- Telerik controls
- Layout Views
- Partial Views
- Bundling and Minification
- jQuery Mobile Integration

Web API2

- Angular JS
- Dependency Injection
- CustomErrors
- Calling WCF Service in ASP.NET MVC
- Deploying MVC App in IIS
- NuGet Package Manager
- ASP.Net MVC 6 features
- Project Implementation

◆ ASP.NET Core MVC

Introduction

- History of ASP.NET
- Difference between .Net framework and .Net Core.
- What is ASP.NET Core?
- Why ASP.NET Core?
- Difference between .Net core and \ ASP.NET Core.
- ASP.NET Core Features
- Advantages of ASP.NET Core
- MVC Pattern
- Understanding ASP.NET Core MVC
- ASP.NET Core vs. ASP.NET MVC vs. ASP.NET Web Forms
- What ASP.NET Core doesn't have?

ASP.NET Core

- ASP.NET Core Environment Setup
- ASP.NET Core Web App (Model-View-Controller) Template
- Connected Services

.NET
FULL STACK

- Dependencies
- Properties
- launchSettings.json
- wwwroot
- css
- js
- lib
- Controllers
- Models
- Views
- appsettings.json
- Program.cs
- Startup.cs
- Middleware Services. Types of Middleware Services.
- Configuring and setting up the Web Server by CreateDefaultBuilder()
- In Process Hosting
- Out of Process Hosting
- What is IIS Express, IIS Web Server and Kestrel Web Server.
- Understanding Life Cycle of ASP.Net Core Request

Controllers & Action Methods

- Controllers Overview
- Action Methods and IActionResult object
- Passing data from Controller to View
- Action Filters
- Building Custom Action Filters
- Asynchronous Action Methods

Routing

- Url Routing Overview
- Convention based Routes
- Attribute Routing
- Routing Constraints
- Custom Routes

Views

- Introducing Razor View
- Advantages of Razor View
- Razor Syntax
- Strongly typed Views
- Layout Page
- Partial Page

Helpers

- Html Helpers
- Built-In Html Helpers
- URL helpers

- Tag Helpers
- Custom Tag Helpers

Model Binding

- Html Form behavior
- Model Binder Overview
- DefaultModelBinder
- Binding to Complex Classes
- IFormCollection Model Binding
- IFormFile Model Binder
- Bind Attribute
- TryUpdateModelAsync

Data Annotations

- Data Annotations Overview
- Validations with Data Annotations
- Server Side and Client Side Validation
- Remote Validation
- State management Techniques
- Cookies
- Sessions

Security

- Authentication and Authorization
- Implementing Security using ASP.NET Core Identity

MVC and Entity Framework Core

- Basic CRUD Operations using Entity Framework
- Separation of work using BO Classes
- Writing Generic Class / Repository
- Caching in Repository

EF Core Development Approaches

- Code-First
- Database-First

.NET
FULL STACK

ASP.NET Core Dependency Injection Handling Errors in ASP.NET Core MVC Applications

ASP.NET Core - Web Caching

- Cache Tag Helpers
- Memory Caching Introduction
- In-Memory Caching
- Response Cache
- Distributed Cache

Module Development

- Understanding Areas
- Adding Areas
- Defining Area Routes
- Linking between Areas

Bundling & Minification

- What is Bundling and Minification in ASP.net Core?
- Bundler and Minifier Extension
- How to Bundle your files
- How to minify your Bundles
- Convert to Gulp

Web API and JQuery Ajax

- Introduction to Web API
- AJAX implementation using JQuery
- Calling the Web API with JQuery Ajax
- Creating a Web API that Supports
CRUD Operations using EF

Integrating Angular in ASP.NET Core Web application.

Hosting MVC Core Application on Microsoft Azure

◆ UI TECHNOLOGIES

HTML 4 HTML BASICS

- WHAT IS HTML
- Features of HTML
- Tags: paired tags (vs) Unpaired tags
- List of tags
- Installing Editor and Browse

HTML Structure

- Syntax of html
- Head (vs) Body
- First example in HTML

HTML Document Basic

- Heading
- Texting formatting tags
- Preformatted text
- Lines

Abbreviations, Images and Hyperlinks

- Meta tag
- List Tags
- Ordered List
- Unordered List
- Tables
- Forms
- What is Form
- Importance of Form in Realtime
- Input tag , types and attributes
- Dropdownlist and Option Group
- Fieldset and Label
- Textarea ,Label
- Submitting the form to server
- DIV and Span

CSS 2

- CSS Basics
- What is CSS
- Need of CSS in real web sites
- Syntax of CSS
- What is Selector
- What is property
- ID Selector
- Class Selector

.NET
FULL STACK

Color properties

- Color
- Background Color
- Types of Color: Named colors, Rgb colors, Hexadecimal colors
- Front Properties
- Text Properties
- Background Properties
- Bullets and Numbering
- DIV and its css properties
- Importances of Box Model
- Padding properties
- Border Properties
- Margin properties
- Advanced
- Opacity
- Display
- Visibility
- Overflow

Position

- Static
- Absolute
- Relative
- Fixed
- Z-index
- Hyperlink Styles
- a:link
- a:hover
- a:visited
- a:active

Types of Style Sheets

- Internal Style Sheet/Embedded Style Sheet
- Internal Style Sheet
- External Style Sheet

◆ CSS Selectors

Style Precedences

- Default Style precedence
- Limport
- Menubar in realtime
- Table style in realtime
- Static page template
- Header
- Nav
- Container
- Leftcol
- Pagecontent
- Footer

- Responsive web design
- Need of RWD in realtime
- Extra-small,small,medium,and Large devices
- Media queries in CSS file
- Viewportmeta tag

Java Script 5

- Introduction to Java script
- Programming fundamentals
- Operators
- Control statements
- Functions
- Arrays
- JavascriptOOP(Advanced JavaScript)
- What is Object
- Properties
- Methods
- Object literals
- Constructor functions
- Demo: creating and calling objects in different ways
- Manipulating objects
- Reading object keys
- Object array
- Prototype
- Inheritances

JSON

- What is JSON
- Object literal(vs) JSON
- Importance of JSON in realtime
- JSON.stringify()
- • JSON.parse()

.NET
FULL STACK

Type conversion

- String functions
- Date functions
- Advanced
- NoScript
- Closures
- Hoisting
- **DOM Manipulation**
- What is DOM
- Window object
- Document Object
- Element Object
- Event Handling
- **Validations**
- What is validation
- Need of validation in realtime
- Submit event
- Required field validation
- Using regular expression
- Types of Javascript
- Internal Javascript/Embedded Javascript

◆ Javascript 6&7/ Ecma Script 6&7

- Fundamentals
- What is EcmaScript
- EcmaScript(vs)JavaScript
- Setting-up the Environment

OOP

- Classes
- Constructors
- Parameter Less Constructor
- Properties
- Methods
- Summary

Search suggestions & Calculations

- Details
- List
- Output

Storage

- LocalStorage
- Session storage
- Geo location
- Canvas and SVG
- 7)CSS3
- CSS 3 Basics
- What is CSS 3
- New features of css3
- CSS 3 first example

Box model properties

- Resize
- Word wrap
- Border Radius
- Box shadow
- Text shadow
- Transformations
- Transitions
- Animations by using theKeyframes
- Creating basic page contact
- Tables
- Tables styles
- Tables background colour
- Buttons
- Inheritance
- Method overriding

Other Concepts

- Set and get methods
- DefaultArguments
- Arrow functions
- Let
- Const
- Rest
- Destructuring
- Multiline strings

HTML 5

- Html 5 Basic
- What is html 5
- New features of html 5
- Advantages of html 5
- Syntax of html 5
- Deprecated elements and attributes
- New doctype for html 5
- New input types
- NEW INPUT ATTRIBUTES
- MULTIMEDIA
- Audio Tag
- Video Tag
- Audio formats and video formats

.NET
FULL STACK

SEMANTIC AND STRUCTURAL TAGS

- Article
- Header
- Nav
- Section
- Aside
- Footer

PROGRESS BAR

- Meter
- Progress

IMAGES

- Figure
- Figcaption

Collapsible

- Details
- SuButton colours
- Table sizes
- Glyphicons
- Visibility
- Show
- Hidden

Images

- Rounded
- Circle
- Thumbnail
- Jumbotron
- Paging
- Panels
- Panels heading
- Panel body

Nav Bar

- Navbar
- Navbar inverse
- Navbar fixedtop
- Navbar header
- Navbar nav
- Breadcrumbs
- Dropdown
- Progressbar
- Tabs
- Forms
- Model
- Grid system(responsive web design)
- Extra small
- Small
- Medium
- Large



React JS

Introduction to React JS

- Importance of React Js in web_Applications
- Features
- History of ReactJS
- Building single-page applications and mobile applications
- React ->a JavaScript library
- Need of virtual DOM and its Mechanism on performance of applications that rely on frequent updates

Benefits of React JS

- Declarative
- Reusable components
- Fast rendering
- JSX
- Server-side rendering
- Strong developer community

Virtual DOM

Cons of React JS

Prerequisites for Learning ReactJS

- Basic JavaScript syntax
- Asynchronous JavaScript
- The DOM
- HTML and CSS
- ES6+

ECMAScript 6

- Arrow functions
- Classes
- Promises
- The spread operator
- Similarities and differences between JavaScript and ReactJs

ReactJS - Environment Setup

ReactJS - JSX(JavaScript XML)

- Importance of jsx files and its working
- use className instead of class in JSX
- Concept of modular programming

.NET
FULL STACK

ReactJS - Components

- Class level components
- Functional Level components
- Efficiency code of using Functional components

ReactJS - State

- Purpose of State
- Mutability of State
- Importance of setState(),useState() hook and render() hook

ReactJS - Props Overview

- Terminology of passing data from a parent component to a child component using props
- Concept of Immutability in props

ReactJS - Props Validation

ReactJS - Component Life Cycle

ReactJS - Events

ReactJS - Refs

ReactJS - Keys

ReactJS - Router

ReactJS - Flux concept

ReactJS - Higher order Components

ReactJS - Best Practices

◆ Angular JS

Introduction

- 1.Introduction
- 2.What is Angular
- 3.Architecture of Angular Apps
- 4.Setting Up the Development environment
- 5.Your First Angular App

Angular fundamentals

- 1.Introduction
- 2.Building Blocks of Angular Apps
- 3.Controllers
- 4.Templates
- 5.Directives
- 6.Services
- 7.Filters
- 8.Dependency Injection

Displaying Data and Handling Event

- 1.Introduction
- 2.Expressions
- 3.Event Binding
- 4.Two-way Binding

Modules

- 1.Creating a module
- 2.Using module in the application
- 3.Creating a module with dependencies
- 4.Creating a contacts for the module

Directives

- 1.What is a directive
- 2.Built In directives
- 3.Creating the custom directive
- 4.Compile , link and transclusion
- 5.Working with isolated scope

Controllers

- 1.What is a directive
- 2.Creating the Controller and using in the application
- 3.Creating a model and function inside a controller
- 4.Configuring simple events with function

Scopes

- 1.What is \$scope
- 2.What is \$rootScope
- 3.\$scope vs \$rootScope
- 4.Change detection and watchers

Filters

- 1.What is filter
- 2.Built in filters usage in the application
- 3.Creating a custom filter

Services

- 1.What is service
- 2.built in services in the application
- 3.creating the custom service
- 4.factory vs provider vs services

Http

- 1.calling the server side apis by using \$http
- 2.getting the data from the server
- 3.ALL CRUD operations by using the \$http
- 4.Working with promises

.NET
FULL STACK

Error Handling

1. Identifying the simple errors and trouble shooting
2. Configuring the special error handling functions
3. Displaying generic error messages

Forms

1. Configuring the Angular Forms with input controls
2. Configuring the validations for the input controls
3. Use of novalidate
4. Configuring project specific error message
5. Routing
 1. Configuring single page application
 2. Configuring single page application in Real time application
 3. Configuring the routes and route parameter
 1. Configuring the navigation using the location service
 2. Enabling the html5 specific routing

Integration

1. Angular JS in the JAVA Applications
2. Angular JS in the .net Applications
3. Angular JS in the NODEJS Applications
4. Angular JS in the Python Applications
5. Angular JS in the PHP Applications

.NET FULL STACK



🌐 sathyatech.com

☎ 91009 20092 - 91009 40094



2nd Floor, Sri Sai Arcade,
Beside Aditya Trade Centre, Ameerpet