Dotnet
Full Stack with GenAI Integration

Introduction of Dot Net Framework

- What is Dot Net?
- Why Dot Net?
- Advantages of Dot Net
- Component of Dot Net
- What is Framework?
- Versions of Framework
- CLR architecture
- CTS architecture
- CLS architecture
- DLR architecture
- What is JIT?
- Types of JIT?
- Pre-JIT COMPILER
- Econo-JIT COMPILER
- Normal-JIT COMPILER

Getting Started with Visual Studio

- Understanding Project and Solution Explorer
- Creating a Project and Solution
- Understanding Solution Explorer
- Comments in C#
- Debug Project
- Understanding Execution Process

Assemblies in .Net

- What is Assembly?
- Types of Assembly?
- How to make Private Assembly
- How to make Shared Assembly
- What is DLL HALL Problem?

- What is SN Key?
- What is GAC?
- How to Deploy Assembly in Cache
- What is MSIL?
- What is DLL & EXE?
- How make DLL& EXE.
- Use of Ilasm.exe (IL Assembler) & Use of Ildasm.exe (IL Disassembler).

Object Oriented Programming System (OOPS)

- What is OOPS?
- What is Class & Object?
- Difference b/w Class, struct, interface and enum
- Types of Variable
- Static Variable
- Const Variable
- Read Only Variable
- Instance & Static Members
- Inheritance & Types of Inheritance
- What is Interface?
- Encapsulation
- Abstraction
- Polymorphism
- Types of Polymorphism
- Static Polymorphism
- Dynamic Polymorphism
- Concept of Method Overloading
- Concept of Method Overriding
- Concept of Early Binding
- Concept of Late Binding
- Concept of Up Casting
- Operator Overloading?
- What is Constructor?
- Types of Constructor
- Default Constructor
- Parameterized Constructor
- Copy Constructor
- Private Constructor
- Static Constructor

- Concept of This Keyword in Constructor
- Concept of Base Keyword in Constructor
- Concept of Base Keyword in Constructor
- Abstract Class & Abstract Method
- Static Class & Static Method
- Sealed Class & Sealed Method
- Partial Class & Partial Method
- Different types of Parameter
- Input Type
- Output Type
- Ref Type
- Params Type
- Dynamic Type

C# LANGUAGE FUNDAMENTALS

Introductions of C#

- What is C#?
- Namespaces in C#?
- Create namespace in C#
- Data types in C#
- Value type
- Reference type
- Null able type
- Heap
- Stack
- Looping in C#
- For,While,Dowhile
- Foreach
- Conditional statement
- Switch case
- If, if-else, Nested if
- GoTo
- Break
- Continue
- Convertion Types
- Implicit {Boxing}
- Explicit {UnBoxing}

Array

- What is Array?
- Types of Array
- Simple Array
- Multi Array
- Jagged Array

File Handling

- Introductions of File Handing
- Working with Files and Directories
- Working with Directories
- Working with Files Using the File and FileInfo Classes
- Working with Directories Using the Directories and Directories Info Classes

Exception Handling

- Introductions of Exception Handling
- Role of Exception Handling
- What is Error, bug and exceptions?
- Try, catch & Finally block
- System Level & Application Level Exception
- Handel Multiple Exception
- Throw Exception

Software Engineering Models and Concept (Project Based)

- What is Software Development Life Cycle?
- Phases of a SDLC
- Project initiation and planning
- Feasibility study
- System design
- Coding
- Testing
- Implementation
- Maintenance

Collections

- Introductions of Collections
- Types of collections
- Generic
- List
- Dictionary
- Stack

- Queue
- Non Generic
- ArrayList
- HashTable
- Stack
- Queue

Delegate

- Introductions of Delegate
- Types of delegate
- SingleCast
- MultiCast
- Generic delegate

Events

- Introductions of Events
- Understanding Events and Event Handler
- Types of Event
- System Define Event
- User Define Event
- Event Life Cycle
- Binding Events
- Working with Event Handler

Threading

- WIntroductions of Threading
- Threading Namespace
- Understanding Thread LifeCycle
- Default Threading
- Multi Threading
- Priority in threading Member Info

Introductions of Properties

- set
- get

The SQL Server Management

- What is SQL?
- What is Database?
- Create Statement
- Create database
- Create table

- Use Identity
- Select Statement
- Insert statement
- Update statement
- Delete statement
- Drop statement
- Truncate statement
- Difference between delete, drop & truncate
- Alter Query
- Clauses
- Group By
- Order By
- Basic Queries
- Stored Procedure
- Use Input and Output Type parameter in Stored Procedure
- Function
- Types of Functions
- Trigger
- Cursor
- Union & Union All
- Transaction
- Joins
- Indexes
- Views
- Constraints

System Requirement Database Structure (Project Based)

- Database Table Structure
- Database Design
- DDL
- DML
- Stored Procedure
- Database Security

ADO .NET

- What is ADO.NET?
- Difference between ADO & ADO.NET?
- Architecture of ADO.NET
- Disconnected mode Architecture

- Using SqlDataAdapter Class
- Using DataSet Class
- SqlCommandBuilder Class
- Connected Mode Architecture
- Using SqlConnection Class
- Using SqlCommand Class
- Using SqlDataReader Class
- Understanding ExecuteNonQuery method
- Understanding ExecuteScalar method
- Uunderstanding ExecuteReader method
- Working with DataTable class
- Creating PrimaryKey Column on VirtualTable
- Using Constraints Class
- ForeignKeyConstraints Class
- UniqueConstraints Class
- Store table in datatable
- Executing Stored Procedure
- Executing Function

ASP.NET Framework

- Asp.Net Introduction
- Evolution of Web Development
- ASP .NET different versions
- Understanding protocol and domain
- Server-side programming
- Client-side programming
- HTML and HTML Forms
- Creating a simple html page
- Developing ASP.NET Applications
- Creating Websites
- Creating a New Web Application
- Page Life Cycle
- Understanding Events of Page Life Cycle
- Websites and Web Projects
- Adding Web Forms
- Migrating a Website from a Previous Version
- Difference b/w inline and code behind concept

- Asp.Net special directory
- What is Application Directories
- Different types of Application Directories
- Understanding application label and page label variable
- Need of Global class
- What is Configuration file?
- Need of Configuration file
- Difference b/t app.config & web.con file
- Nested Configuration
- State Management
- Need of state management technique
- Types of state management
- Understanding Server side and Client side state management
- ViewState
- HiddenField
- Cookies
- Cache
- QueryString
- Session
- Session Tracking
- Session Configuration
- Session mode
- Cookieless
- Session Timeout
- Session Mode
- Application
- Transferring Information between Pages
- Transferring Information between Cross Pages
- Web Controls in ASP.Net
- Types of web controls
- Server Controls
- Standard Controls
- Data Controls
- Validation Controls
- Navigation Controls
- Login Controls
- Ajax Extensions

- Ajax ToolKit
- Html Controls
- User Controls
- Data Controls
- DataList
- DetailsView
- FormsView
- GridView
- ListView
- Repeater
- DataGrid
- EntityDataSource
- Object DataSource
- LinqDataSource
- XmlDataSource
- Working with Templates of Data Controls
- Working with events of Data Controls
- CSS
- Understanding CSS
- Types of CSS
- Inline CSS
- Internal CSS
- External CSS
- Selectors in CSS
- Themes & Skins
- Working with Image gallery
- Working with ImageSlide
- Master Page
- Nested Master Page
- Page Tracing
- Mail & Messaging
- Caching Techniques
- Understanding Caching
- Types of Caching
- Output Caching
- Data Caching
- Web Security

- Need of security
- Types of security
- Forms Authentication
- Windows Authentication
- Confidentiality with SSL
- Password Format
- Types of password format
- Md5
- SHA1
- Clear
- Performance tuning
- Ajax
- Understanding Gmail,Facebook and Twitter API's
- Deploying ASP.NET Applications
- ASP.NET Applications and the Web Server
- How Web Servers Work
- Understanding Virtual Directory
- Web Application URLs
- Web Farms
- Internet Information Services (IIS)
- Installing IIS
- Registering the ASP.NET File Mappings
- Verifying That ASP.NET Is Correctly Installed
- Managing Websites with IIS Manager
- Creating a Virtual Directory
- Configuring a Virtual Directory
- Deploying a Simple Site
- Web Applications and Components
- Other Configuration Steps
- Code Compilation
- Verifying That ASP.NET Is Correctly Installed
- Deploying a Visual Studio IDE
- Creating a Virtual Directory for a New Project
- Copying a Website
- Publishing a Website
- Verifying That ASP.NET Is Correctly Installed

Technical Design & Development (Project Based)

- Working with Project
- Programming Language: C# (Asp.Net)
- Designing Tools
- HTML
- Using CSS
- Using Ajax
- Using JavaScript

ADVANCE .NET MVC

- Introduction to ASP.NET MVC
- Brief history of ASP.NET MVC
- Understanding ASP.NET MVC Design Pattern
- Setup & Installation process
- Compare ASP.Net Web Forms with ASP.NET MVC
- Life cycle of ASP.NET MVC
- Key Benefits of ASP.NET MVC
- Understanding ASP.NET MVC project templates

View Engine

- What is View Engine
- Role of View Engine
- Types of View Engine
- Compare ASP View & Razor View Engine
- Understanding Razor syntax
- Accessing model data in Razor views
- Working with Razor Layouts
- Understanding Razor Expressions

Models

- What is MVC
- Understanding scaffolding template
- Scaffolding template with Entity Framework
- Defining a data model
- Understanding Model Binding
- Working with explicit model binding
- Responding to the edit post request

Views

- Purpose of Views
- Understanding View Conventions

- Types of View
- Understanding ViewBag, TempData,ViewData and ViewDataDictionary
- Creating strongly-typed views
- Adding a View
- Using Razor syntax in View
- Code Block in View
- Working with Partial View
- Introduction of Layout/Master page
- Working with Layout

Controllers and Actions

- What is the Controller
- Use of Controller
- What is Action and Action Results
- Action Filters
- Authorize Filters
- Result Filters
- Built In Filters
- Processing Input and Output Action

Membership, Authorization And Security

- Session and application state
- Using the Authorize Attribute to Login
- Securing Controller Action
- Authorize Attribute with Forms Authentication
- Windows Authentication
- Using Role wise Login
- Managing users and roles
- Configuring OAuth And OpenID Providers
- Login via OAuth and OpenID
- Security Implications of External Login

URL Routing

- Introducing URL Routing
- Routing Approaches
- Defining Attribute and Traditional Routes
- Understanding Route Parameters
- A Details look at URL Generation
- Creating and Registering a Simple Route
- Ambient Route Values

- Using Static URL Segments
- Constraining Routes
- Bypassing the Routing System

Forms And Helper Methods

- Using Form Helper Methods
- The Action and the Methods
- Using GET or POST Method
- Using Template Helper Methods
- Using Model Metadata
- Making Helpers Do your Bidding Inside HTML Helpers
- Setting up the Edit Forms
- Using Strongly Typed Helpers
- Helpers and Model Metadata
- Helpers and Model State
- Html.Hidden
- Html.Password
- Html. RadioButton
- Html.CheckBox
- Html.DropDownList
- Html.Partial and Html.RenderPartial
- Html.Action and Html.RenderAction

Data Annotations And Validation

- Using Validation Annotations
- Custom Error Messages and Localization
- Using Range validator Attribute
- Regular Expression validator Attribute
- Required validator Attribute
- String Length validator Attribute
- Custom Validation and Annotation
- Using display format

Additional Techniques

- View scaffold templates
- Model Validation
- Global filters
- Model binding
- Bundling and Display Modes

Using Ajax, jQuery with ASP.NET MVC

- Overview of Jquery
- Unobtrusive Ajax and the 'this' context
- Using Ajax action links
- Using Ajax Helpers
- Autocomplete with jQuery UI
- JSON & Client-Side Template
- Bootstrap Plug-in

Language Integrated Query (LINQ)

Introduction to LINQ

- What is LINQ
- Need of LINQ
- How LINQ Works
- Using Linq via Method based and Query based synta

Standard Linq Operators

- Sorting Data
- Filtering Data
- Joining Data
- Partitioning Data

Linq Provider

- LINQ to Object
- LINQ to XML
- LINQ to SQL

Entity Framework

- Overview of the Entity Framework
- Difference b/w Linq to Sql and Entity Framework
- Using Schema First Approach
- Using Model First Aproach
- Using Code First Aproach
- Using Stored Procedure in Code First Approach
- Using Sql Profiler with Stored Procedure
- Creating/Managing one to many and many to one relations
- Using Table, Column and Key attributes
- Associations & Foreign Keys

WEBAPI

ASP.Net Core WEBAPI

- What is WebAPI?
- Why select Web API?

- Advantage of WebAPI
- Understanding Http Request Verbs & Constraints
- Http Response Http Status Code
- Creating a Web API Project
- Implementing GET, PUT, POST & DELETE method
- ASP NET Web API and SQL Server

Create New ASP.NET Core Web API & Domain Models

- Introduction
- Create New ASPNET Core Web AP
- Understand ASP.NET Core Web API and File Structure
- Understand REST and HTTP Verbs
- Routing In ASP.NET Core Web API
- Running and Calling ASP.NET Core Web API
- Understanding Our Domain (Domain Driven Development)
- Create Domain Models
- Adding Entity Framework Core Packages
- Creating the DBContext Class
- Adding ConnectionString To The Database In Appsettings.Json
- Understanding Dependency Injection & Injecting DbContext Into Our Application
- Run EF Core Migrations

Create New Controller - Regions Controller and CRUD Operations / Action Methods

- Introduction
- Create New Controller (Regions Controller)
- Get All Regions Action Method
- Get Region By Id Action Method
- DTOs and Domain ModelsI
- Change Methods To Use DTOs
- Create Region Action Method
- Update Region Action Method
- Delete Region Action Method

Asynchronous Programming, Repository Pattern and Automapper

- Introduction
- Asynchronous Programming and Implement Async Await In our Methods
- Repository Pattern and Benefits
- Implement Repository Pattern In ASP.NET Core Web APIs
- Add Other Methods To Repository

- Automapper Introduction
- Implement Automapper

Functionality For Walks - Create, Read, Update, Delete For Walks

- Introduction
- Seeding Data Using Entity Framework Core
- Creating Walks Controller and Create New Walk Action Method
- Get All Walks Action Method
- Navigation Properties In Entity Framework Core
- Get Walk By Id Action Method
- Update Walk Action Method
- Delete Walk Action Method

Validations in ASP.NET Core Web API

- Introduction To Model Validations
- Adding Model Validations To Endpoints
- Custom Validate Model Attribute

Filtering, Sorting, Pagination in ASP.NET Core Web API

- Introduction
- Seeding Data in SQL Server
- Filtering
- Sorting
- Pagination

Securing our ASP.NET Core API - Authentication and Authorization - JWT Tokens

- Introduction
- Authentication Flow
- Setting Up Authentication - Install Nuget Packages
- Testing Authentication Without JWT Token
- Setting Up Identity DbContext
- Seeding Roles
- Run EF Core Migrations To Create Identity Database
- Setting Up Identity
- Create Auth Controller and Register Action Method
- Create Login Action Method
- Create JWT Token in ASP.NET Core Web API
- Inject TokenRepository and Create Token
- Role Based Authorization
- Add Authorization To Swagger

Image Upload In ASP.NET Core Web API

- Introduction
- Create Domain Model For Image Upload
- DbContext changes and Migrations For Image Upload
- New Controller and Action Method for Image Upload
- Image Upload Repository Implementaion
- Serving Static Files Through ASPNET Core Web API

Advanced Functionality in ASP.NET Core Web APIs

- Introduction To Logging
- Add Logging to Console in ASP.NET Core Web API
- Add Logging to Text File In ASP.NET Core Web APIs
- Global Exception Handling In ASP.NET Core Web APIs

Versioning In ASP.NET Core Web API

- What is Versioning and How To Implement Versioning in ASP.NET Web API
- Implement Versioning in ASP.NET Core Web API By Folder Structure
- Implement Versioning By Nuget Package - Proffered Approach
- Fix Versioning In Swagger

Consuming our Web API

- Consuming REST Web APIs
- Creating ASP.NET MVC Web Application
- GET - Consume Web API
- POST - Consume Web API
- GET single Region - Edit Region Prerequisite
- PUT - Updating Using Web API
- DELETE - Deleting a Resource using Web API

Frontend

HTML

Introduction to HTML

- headings
- paragraph
- formatting tags
- image
- anchor
- lists
- table
- id
- class
- iframe

- div
- span
- form
- audio
- video
- html vs html5

CSS

Introduction to CSS

- ways to use CSS (inline , internal, external)
- css selectors
- css color
- border
- padding
- margin
- height
- width
- lists
- text
- display (none, inline , block , inline- block)
- flex
- grid
- position
- z-index
- media

Java Script

- Js Basics
- keywords
- identifiers
- datatypes
- variables and constants
- tokens
- operators

Decision Making Statements

Conditions Statements Or Selection Statements

- if
- if else
- nested if else

- if else if ladder
- switch case

Iteration Statements Or Loops

- for
- while
- do while
- for in
- for of

Jump Statements

- continue
- break

Functions

- function declarations, definitions and calling
- types of functions
- anynomous functions
- flat arrow functions or arrow functions
- self invoked function (IIFE)
- storage classification or varibale scope (var let const and without keywords)
- recursion

Strings Methods

- array Methods
- Global Execution Context and Function execution context
- scope chain rule and lexical scoping
- closure

Objects

- fliteral and keys
- object functions
- nested objects
- Object Prototypes
- setter and getter methods

Set and Map

WeakMap and WeakSet

Dates and Time

OOPS

- literal and keys
- object functions
- nested objects

- Object Prototypes
- setter and getter methods

DOM Manuplation

- DOM
- BOM
- event handling
- time based events

Advanced Js

How Javascript Works

- hoisting
- strict mode
- closure
- synchronous and asynchronous programming

Event propagation

- bubbling
- capturing

Event loop

function curring

ajax and json

calling api using ajax

callback hell

promise

async await

fetch

modules

version

Typescript

- Why Typescript
- Basic Types
- Class and Interfaces
- Modules

Angular

INTRODUCTION TO ANGULAR

- What is Angular
- What is SPA
- Comparison between different versions
- Advantages and Disadvantages
- Key features

ENVIRONMENTAL SETUP

- Node /NPM
- Angular CLI
- Setup Project
- Directory Structure
- Code Editors
- How Angular Application Boot

MODULES

- What is Module
- Decorator
- NgModule
- Core Module
- Bootstrapping
- Shared Modules

COMPONENTS

- Introduction to Components
- Component Architecture Patterns
- Metadata
- State and Behaviors

TEMPLATES

- Template Expression or Interpolation
- Data Binding (Event ,Property , Two Way)
- *ngIf , *ngFor , *ngSwicth
- Built-in Structural and Attributes Directive

ENVIRONMENTAL SETUP

- Node /NPM
- Angular CLI
- Setup Project
- Directory Structure
- Code Editors
- How Angular Application Boot

MODULES

- What is Module
- Decorator
- NgModule
- Core Module
- Bootstrapping
- Shared Modules

COMPONENTS

- Introduction to Components
- Component Architecture Patterns
- Metadata
- State and Behaviors

TEMPLATES

- Template Expression or Interpolation
- Data Binding (Event ,Property , Two Way)
- *ngIf , *ngFor , *ngSwicth
- Built-in Structural and Attributes Directive

PIPES

- Built-in Pipes
- Custom Pipes

Routing

- Routes , RouterModule , routerLink
- Child Routing
- Params
- Guards

CUSTOM DIRECTIVES

- Structured Directives
- Attribute Directives

VALIDATIONS

- Template Driven
- Reactive Form
- Custom Validators

ASYNCHRONOUS OPERATIONS

- Observable vs Promise
- Network Call
- HTTP Request | Response

SERVICES

- Introduction to Services
- Building a Service
- Dependency Injections
- Injectors and Providers

LIFECYCLE HOOKS

- Component LifeCycle
- All LifeCycle Hooks

LOCAL DATA STORAGE

- Local Storage
- Session Storage
- Cookies

COMMUNICATION BETWEEN COMPONENTS

- @Input()
- @Output() with EventEmitter
- @Viewchid()
- Shared Service
- Subject

SERVER-SIDE RENDERING

- Introduction to Server-Side Rendering
- Angular Universal

Angular Testing

- Introduction to Testing
- Unit Testing with Karma and Jasmine

LAZY LOADING

UI COMPONENTS

- Angular Material
- PrimeNG

Third Party Modules

- Social Login
- Pagination
- Search
- Filter
- JWT Token
- File Upload
- Many More

Angular Application Deployment

- Build Application
- Deploy in Heroku

Generative AI Integration

Introduction to AI & Large Language Models

Understanding GPT Models

- What is OpenAI
- How to get API key
- Understanding Library
- How to generate text

- How to perform sentiment analysis
- How to generate image from text
- How to implement chatbot

Partners : ORACLE WORKFORCE DEVELOPMENT PROGRAM | redhat. CERTIFIED TRAINING PARTNER | UBIC Unica Business and Innovation Centre | edexcel | Java

DUCAT
The IT Training School
E-mail: info@ducatindia.com
Visit us: www.ducatindia.com
www.facebook.com/ducateducation

NOIDA
A-43 & A-52, Sector-16,
Noida - 201301, (U.P.) INDIA
70-70-90-50-90
+91 99-9999-3213

2.0 NOIDA SEC-63
H-43 Sector-63
Noida-201301
70-70-90-50-90
+91 7042175774

GHAZIABAD
1, Anand Industrial Estate,
Near ITS College, Mohan Nagar,
Ghaziabad (U.P.)
70-70-90-50-90
+91 9810851363

PITAMPURA (DELHI)
Plot No. 366, 2nd Floor,
Kohat Enclave, Pitampura,
(Near- Kohat Metro Station)
Above Allahabad Bank,
New Delhi- 110034.
70-70-90-50-90

SOUTH EXTENSION (DELHI)
D-27,South Extension-1
New Delhi-110049
70-70-90-50-90
+91 98-1161-2707

VIKASPURI (DELHI)
4, A-Block, Vikaspuri, Delhi,
Nearest Metro Janakpuri West,
Pillar No. 628, New Delhi - 110018
85-95-52-67-42
+91 98-1161-2707

GURGAON
1808/2, 2nd floor old DLF,
Near Honda Showroom,
Sec.-14, Gurgaon (Haryana)
70-70-90-50-90

FARIDABAD
16/5 Mathura Road, Sector 19,
Behind Metro Pillar No 687,
Faridabad
70-70-90-50-90
+91-9355333011