

C# Full Stack Development with React JS and ASP.NET

Section 1: HTML 5

- Introduction Web
- > What are the different Types of Web Apps overview?
- Introduction to HTML
- Define HTML Structure
- Difference between Tag vs Element
- > Difference between Semantic vs Non-semantic elements
- > Difference between Block level elements vs Inline elements
- HTML Elements
- HTML Forms & its Attributes
- HTML Input Elements
- HTML Global Attributes
- HTML Element Specific Attributes

Section 2: CSS

- > What is CSS?
- Understanding the CSS Syntax.
- > CSS Selectors.
- How To Add CSS in HTML.
- > Cascading Order.
- CSS Colors.
- ➤ CSS Backgrounds.
- CSS Borders
- CSS Margins
- CSS Padding
- CSS Height and Width



- CSS Text, Fonts, Links, Tables
- CSS Float and Clear
- > The display: inline-block Value

Section 3: BOOTSTRAP

- > What is Bootstrap?
- Get Start with Bootstrap
- Containers
- Grid System
- Structure of a Bootstrap Grid
- Bootstrap Colors
- Bootstrap Tables
- Bootstrap Jumbotron
- Bootstrap Alerts
- Bootstrap Buttons
- Bootstrap Navs
- Define Navigation Bars
- Bootstrap Forms
- Bootstrap Modal

Section 4: INTRODUCTION TO JAVASCRIPT

- Introduction
- > What is JavaScript?
- JavaScript Features
- JavaScript Libraries
- Understanding HTML, CSS, JavaScript
- JavaScript Version
- > ES5
- ➤ ES6/ES2015



- ► ES7/ES2016
- ≻ ES8/ES2017
- > Data Types
- > Number
- > String
- >> Boolean
- Undefined
- ≻ Null
- Variable Mutation
- Basic Operator
- Operator Precedence
- Shorthand operator
- Problem statement
- Decision making statement if else
- ➤ if else example
- Comparing the problem statement solution
- Switch statement example
- ≻ Boolean logic
- Boolean logic example
- Ternary operator
- Ternary operator example
- > Tips
- Functions
- Function statement
- Function Statement Example
- > What is Array
- Real world Problem Ecommerce
- Real world problem transport
- > Space and the Complexity
- Built in function Complexity
- Examples of Array



- > Objects, When to use it
- Objects and Methods
- Loops and Iteration
- > For Loop
- Continue and Break Statement

Section 5: ES6 JAVASCRIPT

- History of JavaScript
- Features
- let & const and its example
- Arrow Functions
- > Alternative
- Tips and Arrow Functions
- > Exports and Imports
- > Tips for exports and imports
- Classes
- Classes example
- > Inheritance
- Spread and rest Parameter
- Destructuring

React JS Course Content

Section 1: Getting Started with React

- What is ReactJS
- Installation or Setup
- Create React App
- Advantages of ReactJS



- Workflow of ReactJS
- Node Setup
- How to use NPM and its purpose
- ES6 Introduction
- > JS vs JSX vs TS vs TSX

Section 2: Components

- Creating Components
- Basic Component
- Nesting Components
- Higher Order Components

Section 3: Overview OF JSX

- Introduction of Virtual DOM
- Difference between JS and JSX
- > Why use JSX
- JSX Attributes
- JSX Styling

Section 4: Props in React

- Introduction
- Default props
- PropTypes
- Passing down props using spread operator
- Props.children and component composition
- > Detecting the type of Children components
- Props Drilling



Section 5: State in React

- Basic State
- Common Antipattern
- > setState()
- State, Events and Managed Controls

Section 6: React Component Lifecycle

- > Component willMount
- Component UnMount
- Component Update
- Lifecycle method call in different states
- React Component Container

Section 7: Handling Events

- > Event handling in React
- Binding event handlers
- Arrow functions vs. regular functions
- Primary Events
- Sharing the events between components

Section 8: Communicate Between Components

- Child to Parent Components
- Parent to Child Components
- Not-related Components



Section 9: Conditional Rendering

- > Conditional rendering with if statements
- Conditional Rendering using logical && in JSX
- Conditional Rendering using ternary operator
- Preventing Component from Rendering

Section 10: Lists and Keys

- Steps to Create and Traverse React JS Lists
- Rendering lists inside Components
- ➤ Key in React List

Section 11: ReactJS Keys

- What is a key in React?
- Assigning keys to the list
- Difference between keys and props in React

Section 12: React Hooks

- useState()
- useEffect()
- useContext()
- > useRef()
- useReducer()
- Custom React Hooks
- useDispatch(), useSelector, useMemo, useLayoutEffect()

Section 13: Styling in React

- CSS in React
- Different approaches for styling (CSS, CSS-in-JS, CSS Modules)
- > Inline styles



- Styling Libraries
- Popular CSS frameworks (e.g., Bootstrap, Material-UI)

Section 14: Router

- Introduction to React Router
- Setting up React Router
- Creating routes
- Navigating with React Router
- Using Link and NavLink
- Nested Routes and Dynamic Routing
- Nested routes
- Passing parameters to routes
- Query String, accessing current URL

Section 15: React JS Forms

- React Forms
- > Lists of Form components
- Control Input elements
- Controlled and Uncontrolled form components
- Adding Forms in React
- Handling React Forms
- Submitting React Forms
- Multiple Input Fields
- React JS Form validations
- Custom Validations

Section 16: State Management with Redux

Introduction to Redux



- Understanding the need for state management
- Basic concepts: actions, reducers, store
- Setting Up Redux
- Installing Redux and setting up a store
- Creating actions and reducers
- Connecting React with Redux
- Using connect to connect components to the store
- Dispatching actions

Section 17: Asynchronous Programming and API Integration

- AJAX and Fetch API
- Making HTTP requests in React
- Fetching data from an API
- Async/Await and Promises

Section 18: Handling Errors in React Application

- Error Handling and Debugging
- Debugging React apps
- Performance Optimization
- > Memoization
- React.memo and PureComponent

Section 19: ReactJS Virtual DOM

- > What is DOM?
- Disadvantages of real DOM
- Virtual DOM
- How does virtual DOM actually make things faster?
- How virtual DOM helps React?
- Virtual DOM Key Concepts
- Differences between Virtual DOM and Real DOM



- Section 20: Unit Testing Overview
- Section 21: Lazy Loading
- Section 22: Code Splitting
- Section 23: Server-Side Rendering
- Section 24: Micro FrontEnd Overview
- Section 25: Deploying a React Application

Section 26: C# Fundamentals

- Introduction to .NET in C#
- Variables and Data Types
- Operators and Expressions
- Control Structures (if, switch)
- > Loops (for, while, foreach, do-while)
- Error Handling with Try-Catch
- Methods and Parameters
- Namespaces and Assemblies

Section 27: Object-Oriented Programming (OOP)

- Classes and Objects
- Constructors and Destructors
- Encapsulation, Abstraction
- Inheritance and Polymorphism
- Interfaces vs Abstract Classes
- Access Modifiers (public, private, etc.)
- Static vs Instance Members

Section 28: C# Advanced Topics

Collections (List, Dictionary, Queue, Stack)



- > Generics
- Lambda Expressions
- LINQ (Language Integrated Query)
- Exception Handling (Advanced)
- ➤ File I/O (Reading/Writing files)
- Nullable Types and Null Safety

Section 29: Working with APIs in C#

- Understanding RESTful APIs (GET, POST, PUT, DELETE)
- Using HttpClient for making API calls
- Sending and Receiving JSON
- Deserialization with System.Text.Json or Newtonsoft.Json
- Handling API Authentication (Bearer Tokens, API Keys)
- Calling External APIs and Consuming Responses
- Creating Unit Tests for API Calls

Section 30: Building APIs with ASP. Net Core

- Introduction to ASP. Net Core
- Project Structure (Controllers, Models, Program.cs, etc.)
- Creating Controllers and Endpoints
- Routing and Attributes
- Model Binding and Validation
- Dependency Injection (DI)
- Middleware and Request Pipeline
- Swagger / OpenAPI Integration
- Error Handling and Logging in APIs

Section 31: Working with Databases and SQL



- Introduction to Relational Databases and SQL Basics
- CRUD Operations with SQL (SELECT, INSERT, UPDATE, DELETE)
- Joins and Query Optimization
- Entity Framework Core (EF Core)
- Code-First and Database-First Approaches
- DbContext and DbSet
- LINQ to SQL
- Migrations
- Connecting C# to SQL Server
- Handling Relationships (One-to-Many, Many-to-Many)

Section 32: Advanced

- Asynchronous Programming with async / await
- Working with Azure/AWS for Deployment
- JWT Authentication in APIs
- Role-Based Authorization
- Versioning in APIs