

FULL STACK (MERN STACK) with AI

(DURATION: 1 year Part Time/ 6 Months Full Time)

MODULE 1 : DESIGNING FRONT END OF WEBSITE USING HTML, CSS & TAILWIND

➤ **HTML & HTML (5.0)**

- HTML Tags, Attributes, Headings, List, Image
- Hyperlink, Frames, Form, Table
- HTML5 Semantic Attributes, HTML5 SVG element
- Audio Tag, Video tag
- New Form Elements, Canvas
- Drag and Drop API, Geolocation, HTML5 Compatibility

➤ **Cascading Style Sheet (CSS 3 & 5)**

- Backgrounds, Box Model, Dimension
- Pseudo-class, Pseudo-element
- Navigation Bar Image Gallery, Image Opacity
- Gradients, Shadows, 2D & 3D Transforms
- Progress Bars, Multistep Form
- Animations, Image Slideshow, Image Hover Overlay
- Animation Button, Vertical tabs
- Overlay effect, Rounded Corners, Fixed Menu Bars
- Slide Bar on Scroll, Cards, Tabs etc.

PROJECT

Tailwind CSS

Use of AI in Web Designing (ChatGPT etc)

- Getting Started with Tailwind CSS, Setting up Tailwind CSS
- Core Concepts Customization Base Styles Layout Flexbox & Grid Spacing
- Utility Classes, Exploring various utility classes for styling elements
- Responsive design with utility classes
- Styling Components, Customizing the default theme
- Designing responsive layouts, Integration with Frameworks (React)

MODULE 2 : PROGRAMMING CONCEPTS & SCRIPTING

➤ **Programming Through 'C' language**

- Data Types, Variable, Constants, Keywords.
- Types of Errors, **C Arithmetic**, C Programming Structure
- Condition Checking (if-else), Loops (While, For, Do-while)
- Switch Case, Continue & Break statements
- Functions, Pointers , Recursion
- Array (1-D, 2-D), String, Structures File Handling

EXAM

➤ **Object Oriented Programming through C++**

- Object and Classes, Structure Vs Classes
- Passing Object to function
- Data Encapsulation/Data Hiding, Access Specifiers
- Inheritance (Multi-Level, Hybrid, Multiple Inheritance)
- Friend Class, Friend Functions
- Polymorphism (Virtual Functions)

EXAM

➤ **JavaScript**

- About Javascript and version, Variable in Javascript, Javascript operator
- Condition Check (If-else statement), Switch statement
- Loop (While Loop, Do while Loop, For Loop), Nesting of Loops
- Function, Array, Strings, Error Handling, Event Handling, Javascript Objects
- DOM Manipulation, ES6+ Features, Asynchronous JavaScript
- JavaScript in the Browser (Cookies, Getlocation API)

PROJECT

➤ **jQuery**

- Anatomy of a jQuery Script, Selecting: The Core of jQuery
- Images, Slideshows, Menus , Panels and Panes
- Decorating: CSS with jQuery
- Enhancing: Adding Effects with jQuery
- Tabs, Forms, Animating , Resizing, Scrolling
- Lists & Tables

➤ **Bootstrap (Designing a **Mobile Responsive** WebSite)**

- Structure of Bootstrap Application Responsive Web Design(RWD)
- Using page headers and panels, Adding Media Objects, Thumbnails
- Grid, Forms, Plugins, Navigation Components, Bootstrap CSS, etc

PROJECT

MODULE 3 : REACT JS

➤ **Getting Started with React.js:**

- What is React.js and its benefits, Setting up the development environment
- Creating a basic React application, Understanding React components

➤ **JSX (JavaScript XML):**

- Introduction to JSX and its role in React, Writing JSX code to create UI elements
- JSX expressions and using JavaScript within JSX

➤ **Components and Props:**

- Creating functional components, Understanding class components
- Passing data to components using props, State and lifecycle methods

➤ **Handling Events:**

- Adding event handlers to components
- Understanding synthetic events, Commonly used events in React.js

➤ **Lists and Keys:**

- Rendering lists of data with React
- Using keys to optimize rendering performance

➤ **Forms and Controlled Components:**

- Creating forms with React,
- Managing form data using controlled components
- Styling in React.js:

➤ **Styling components using inline styles**

- Utilizing CSS classes and modules

➤ **React Router (for web applications):**

- Implementing client-side routing in React.js applications

➤ **State Management (introductory level):**

- Basic introduction to state management libraries like Redux or Context API

➤ **Introduction to Hooks:**

- Understanding React Hooks and their benefits
- Working with useState and useEffect

➤ **Deploying React.js Applications:**

- Preparing the app for deployment
- Hosting options for React.js applications

PROJECT

- What is MongoDB
- MongoDB vs SQL Databases
- Installing MongoDB (Configuring and starting MongoDB)
- Shell vs Drivers
- CRUD Operations (Create, Read, Update, Delete)
 - Creating Databases & Collections, Comparing JSON & BSON
 - Finding, Inserting, Deleting & Updating Elements
 - FindingMany, InsertingMany, DeletingMany, UpdatingMany
 - Arrays, Accessing Structured Data
 - **Advanced Create Operations**
 - Working with Ordered Inserts
 - Importing Data
 - **Advanced Read Operations**
 - Methods, Filters & Operators
 - Query Selectors & Projection Operators
 - Comparison Operators
 - Embedded Fields & Arrays
 - Element Operators
 - Working with “\$type” • “\$regex” and “\$expr”
 - Array Query Selectors • Cursors and Projection
 - **Advanced Update Operations**
 - CRUD Operation on Element Fields, CRUD Operation on Arrays
 - **Advanced Delete Operations**
 - “deleteOne()” & “deleteMany()”
- **Working With Indexes**
 - Single Field Index & Compound Indexes Indexes For Sorting
 - Partial Filters & Partial Index, Time-To-Live (TTL) Index
 - Query Diagnosis, Query Planning, Multi-Key Indexes, Text Indexes & Sorting

MODULE 5: NODE JS

➤ Introduction to Node.js

- Node.js is and its architecture.
- Differences between Node.js and traditional server-side technologies.
- Installing Node.js and npm (Node Package Manager).

➤ Node.js Module System

- Intro Node.js Module System, Importing Node.js Core Modules
- Importing npm modules and your own files
- Global npm modules and nodemon,

➤ Basic JavaScript and Asynchronous Programming:

- Reviewing fundamental JavaScript concepts.
- Understanding asynchronous programming and the event loop in Node.js.
- Working with callbacks, Promises, and Async/Await.

➤ Building a Basic Server:

- Creating a simple HTTP server using Node.js core modules.
- Handling HTTP requests and responses.
- Routing and handling different HTTP methods (GET, POST, etc.).

➤ Working with Databases (MongoDB)

- Connecting to databases like MongoDB, MySQL, or PostgreSQL.
- Performing CRUD operations (Create, Read, Update, Delete) on the database.
- Using an Object-Relational Mapping (ORM) library.

➤ Working with Express.js:

- Introduction to Express.js, a popular Node.js web framework.
- Setting up routes, middleware, and handling templates.
- Handling forms, file uploads, and cookies.

➤ Authentication and Security:

- Implementing user authentication and authorization.
- Handling passwords securely using hashing and salting.
- Preventing common security vulnerabilities like SQL injection and Cross-Site Scripting (XSS).

➤ Testing and Debugging:

- Writing unit tests and integration tests for Node.js applications.
- Using testing frameworks like Jest or Mocha.
- Debugging Node.js applications using debugging tools.

PROJECT

MODULE 6 Express.js

Introduction to Express.js

- Understanding Node.js and its role in web development
- Explaining what Express.js is and its advantages
- **Setting Up the Development Environment**
- **Routing and Middleware**
 - Defining routes and handling HTTP requests (GET, POST, PUT, DELETE)
 - Working with route parameters and query strings
 - Implementing middleware for request processing
- **Templating Engines**
 - Integrating and using templating engines (EJS, Pug, Handlebars)
 - Rendering dynamic content on web pages
- **Working with Forms and Data**
 - Handling form submissions
 - Validating user input
 - Connecting to databases (e.g., MongoDB, MySQL) using middleware like Mongoose
- **Authentication and Security**
 - Implementing user authentication and sessions
 - Securing routes and data
 - Managing passwords and user credentials
- **Error Handling**
 - Understanding error handling in Express.js
 - Custom error handling and error middleware
- **RESTful API Development**
 - Building RESTful APIs with Express.js
 - Handling CRUD operations (Create, Read, Update, Delete)

PROJECT

MODULE 7: Rest API

- Client–Server concept, Introduction to HTTP, Request & Response cycle
- HTTP Methods (GET, POST, PUT, DELETE) Introduction to JSON
- Creating GET API, POST API, Reading request data, Sending JSON response
- CRUD Operations (POST, GET, PUT, DELETE), Create a REST API

MODULE 8: GIT & GITHUB

- Git Basics & Git Configuration, Git Workflow
- Branching and Merging, Git Logs and History
- Stores and shares your Git projects online

ADD: **PLOT No. 8, VAISHALI ENCLAVE, METRO ROAD, PITAMPURA, DELHI-110034**
(NEAR GULAB SWEETS, OPP: METRO PILLAR NO. 350)

Website: www.niceitservices.com Ph: 9873459848, 7838152268