Full Stack Web Development Course

260 Hours

Introduction:

This comprehensive course offers a deep dive into the world of full-stack web development, covering essential technologies and frameworks to equip participants with the skills needed to build dynamic web applications.

From frontend design using HTML, CSS, and JavaScript to backend development with databases and server-side scripting, this course provides a holistic understanding of modern web development practices.

Module 1: Intro and basic concepts

- Introduction to Computers: Understanding the fundamental components and functionalities of a computer system.
- Binary: Learning the binary number system, its significance in computing, and how to perform basic binary arithmetic.
- Intro to Algorithms: Gaining a foundational understanding of algorithms, their importance in problem-solving, and basic algorithmic concepts.
- UML: Introduction to Unified Modeling Language (UML) for visualizing and documenting software systems.
- Pseudo-code: Learning to write pseudo-code to represent algorithms in a simplified, human-readable format.
- Simple Logic Questions: Practicing basic logic questions to develop problem-solving and critical thinking skills.

Module 2: Basic programming using JavaScript

- Installations: Setting up the development environment, including installing necessary software and tools for JavaScript programming.
- The IDE: Introduction to Integrated Development Environments (IDEs), focusing on popular choices for JavaScript development and their features.

- Types: Understanding JavaScript data types, including numbers, strings, booleans, objects, and undefined/null.
- Strings: Learning how to manipulate and utilize strings in JavaScript, including common methods and operations.
- Statements: Familiarizing with basic JavaScript statements, including variable declarations, assignments, and expressions.
- Conditions: Exploring conditional statements (if, else if, else, switch) to control the flow of the program based on different conditions.
- Loops: Understanding and implementing loops (for, while, do-while) to execute repetitive tasks efficiently.
- Arrays: Learning about arrays, their properties, and methods to store and manipulate collections of data.
- Functions: Introduction to functions, understanding their importance, how to define and invoke them, and the concept of scope and closures.

Module 3: HTML/CSS

- Introduction: Overview of HTML and CSS roles in web development.
- Basic Syntax: Structure of HTML documents and CSS styles.
- Internal/External CSS: Linking and applying CSS styles.
- HTML Tags Selectors: Styling HTML tags with CSS selectors.
- ID & Class Selectors: Targeting elements using ID and class selectors.
- Child/Descendant Selectors: Styling nested elements.
- Attribute Selectors: Using attributes to style elements.
- Pseudo Classes: Styling elements based on state (e.g., :hover).
- Colors, Fonts, Backgrounds: Applying colors, fonts, and background styles.
- Positioning: Techniques for element layout (static, relative, absolute, fixed, sticky).
- Box Model: Understanding margins, borders, padding, and content.
- Floats: Using float property for layout.
- Flexbox: Flexible and responsive layout model.
- Grid: Creating complex, two-dimensional layouts.
- Transformations & Animations: Applying transformations and creating animations.
- Responsive Layout: Principles and techniques for responsive design.
- CSS Frameworks: Introduction to frameworks like Bootstrap and Foundation.

Module 4 Source control (Git)

- Basic Concepts: Understanding version control and the role of Git and GitHub.
- Installations: Setting up Git on your local machine.

- Repositories: Creating and managing Git repositories.
- Main Branch: Understanding the main branch and its significance.
- Committing Changes: Saving changes to the repository with commit messages.
- Push: Uploading local changes to a remote repository.
- Clone: Copying a remote repository to your local machine.
- Branching: Creating and managing branches for parallel development.
- Merging: Combining changes from different branches.
- Pull Request: Submitting changes for review and integration.
- Issues: Tracking and managing project tasks and bugs using GitHub Issues.

Module 5: JavaScript

- DOM: Manipulating the Document Object Model to dynamically interact with web pages.
- Closures: Understanding closures and their use in JavaScript.
- IIFE: Immediately Invoked Function Expressions for encapsulation.
- Function Constructor: Creating objects using function constructors.
- Module Pattern: Structuring code using the module pattern for better organization.
- Prototype: Leveraging prototypes for inheritance and object properties.
- this: Understanding the context of this in different scenarios.
- ECMAScript 6+: Exploring new features introduced in ES6 and beyond.
- Class: Using classes for object-oriented programming.
- Lambda Expression: Writing concise functions with arrow syntax.
- New Object Methods: Utilizing new methods for objects introduced in ES6+.
- Collections: Working with Map, Set, and other collection types.
- Promise: Managing asynchronous operations with Promises.
- Async/Await: Simplifying asynchronous code with async/await syntax.
- Iterator: Implementing and using iterators for custom iteration behavior.
- Generator: Creating generators for controlled iteration.
- Reflection: Using reflection for dynamic code analysis and manipulation.

Module 6: SQL/No-SQL

- Environment Setup: Setting up development environments for SQL and No-SQL.
- MySQL Server: Installing and configuring MySQL.
- Northwind DB: Using the Northwind sample database.
- Tables: Creating and managing tables.
- Relations: Defining table relationships.
- Data Types: Understanding SQL data types.

- ERD: Creating Entity-Relationship Diagrams.
- Retrieving data and filtering with SELECT and WHERE
- Scalar Functions: Using scalar functions.
- Joins: Combining tables with JOINs.
- Grouping: Aggregating data with GROUP BY.
- Sub Queries: Writing subqueries.
- DML-CRUD: Performing Create, Read, Update, Delete operations.
- Stored Procedures: Using stored procedures.
- MongoDB: Introduction to MongoDB.
- Installation: Setting up MongoDB.
- Collections: Managing collections.
- Documents: Working with documents.
- CRUD Operations: Create, Read, Update, Delete in MongoDB.
- Indexing: Optimizing with indexing.

Module 7: React

- Introduction & Installation: Overview and setting up React.
- Components: Creating functional and class components with JSX.
- Props & State: Managing data and state within components.
- Events & Lifecycle: Handling events and using lifecycle methods.
- Fetching Data: Retrieving data from APIs.
- Rendering: Conditional rendering and working with component children.
- Refs: Accessing DOM elements.
- Styling: Advanced styling options.
- Hooks: Using hooks like useEffect and useContext.
- TypeScript: Integrating TypeScript with React.
- Routing: Implementing navigation with React Router.
- Forms: Handling form inputs and submissions.
- Context API: Managing global state.

Module 8: NodeJS

- V8 & Modules: Introduction to the V8 engine and using modules with require.
- npm: Managing packages with npm.
- File System: Working with the fs module.
- Streams & Buffers: Understanding streams, buffers, and pipes.
- Blocking vs Asynchronous: Differences between blocking and asynchronous operations.
- Events & Event Loop: Handling events and understanding the event loop.
- Timers & EventEmitter: Using timers and the EventEmitter class.

- HTTP: Creating a simple server and understanding routing and HTTP methods.
- JSON: Working with JSON data.
- Express Server: Setting up an Express server and building a REST API.
- Express Generator: Using Express Generator for scaffolding.
- Connecting to DB: Integrating databases with Node.js.
- Authentication: Implementing authentication mechanisms.

Module 9: Cloud

- Deployment Basics: Introduction to deploying applications to the cloud.
- Heroku: Deploying and managing applications on Heroku.
- Netlify: Hosting static websites and serverless functions with Netlify.
- AWS Fundamentals: Overview of Amazon Web Services (AWS) core services.
- S3: Using Amazon S3 for storage solutions.
- Lambdas: Creating and deploying serverless functions with AWS Lambda.

Module 10: Full Stack Project

- Culminating project integrating frontend, backend, and database technologies learned throughout the course.