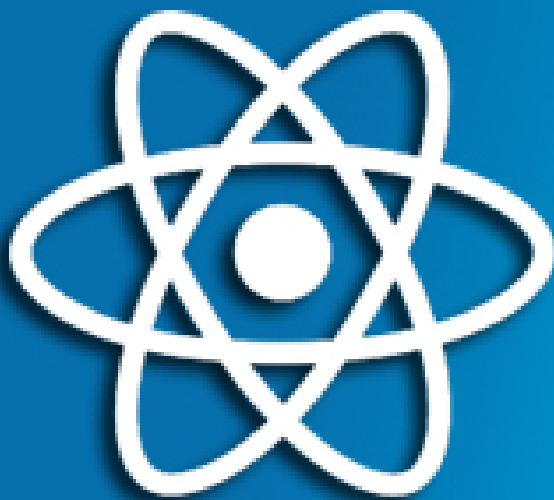




Fusion Software
Institute

WWW.FUSION-INSTITUTE.COM

REACT JS



React JS

TABLE OF CONTENTS

1. INTRODUCTION TO HTML

- *What is HTML (Hypertext Markup Language)?*
- *Setting up a text editor and web browser for development*
- *Creating your first HTML document*
- *Document structure and elements*
- *Text editors and Integrated Development Environments (IDEs)*

2. HTML DOCUMENT STRUCTURE

- *HTML elements, tags, and attributes*
- *Headings, paragraphs, and line breaks*
- *Lists: ordered lists*
- *unordered lists and list items*
- *Hyperlinks: creating links to other web pages (element)*
- *Linking to external resources (images, stylesheets, and scripts)*
- *Semantic HTML*

3. TEXT FORMATTING AND MULTIMEDIA TEXT

FORMATTING:

- *, , <u>, <s>, and <mark>*
- *Creating and formatting tables: <table>, <tr>, <td>, <th>, <caption>*
- *Adding images: element and attributes*
- *Working with multimedia: <audio> and <video> elements*

TABLE OF CONTENTS

- *Embedding content from other websites (iframes)*
- *Accessibility considerations: alt text and ARIA attributes*

4.FORMS AND INPUT ELEMENTS

CREATING FORMS:

- *<form> element and its attributes*
- *Form controls: text input, password input, textarea, checkboxes, radio buttons, and select dropdowns*
- *Form submission and action attributes Labels, fieldsets, and legends for form accessibility*
- *Validating user input with HTML5 attributes (required, pattern, etc.)*
- *Introduction to form handling with JavaScript (event listeners)*

5.LINKS, LISTS, AND NAVIGATION:

- *Creating internal links (anchor tags)*
- *Relative vs. absolute URLs*
- *Lists for navigation menus and site structures*
- *Creating navigation menus using <nav> and *
- *HTML comments: <!-- comment --> Best practices for SEO-friendly HTML*

TABLE OF CONTENTS

6. ADVANCED HTML TOPICS

- *Meta tags and their importance for SEO and social sharing*
- *Charset, viewport, and other meta attributes*
- *HTML entities for special characters*
- *Using <div> and for layout and styling*
- *HTML5 semantic elements for improved structure*
- *Embedding custom fonts and icon libraries*

7. HTML FORMS AND DATA COLLECTION

- *More advanced form elements: file uploads, input types (email, date, etc.)*
- *Form attributes: autocomplete, novalidate, and enctype*
- *Building accessible and user-friendly forms*
- *HTML5 form validation and error messages*

8. HTML AND CSS INTEGRATION:

- *Introduction to CSS (Cascading Style Sheets)*
- *Inline CSS vs. internal CSS vs. external*
- *CSS Linking external stylesheets to HTML documents*
- *Applying CSS styles to HTML elements using selectors*
- *CSS properties: color, font-size, margin, padding, and more*
- *Creating a basic styled webpage*

TABLE OF CONTENTS

CSS

1: INTRODUCTION TO CSS

- *What is CSS and its role in web development*
- *Inline, internal, and external CSS*
- *Basic syntax and CSS rules*
- *Selectors: element selectors, class selectors, and ID selectors*
- *CSS properties and values*

2: STYLING TEXT AND FONTS

- *Font properties: font-family, font-size, font-weight, and font-style*
- *Text properties: color, text-align, text-decoration, and text-transform*
- *Line height and letter spacing*
- *Web-safe fonts and Google Fonts*
- *Creating custom font stacks*

3: COLORS AND BACKGROUNDS

- *CSS color values: hexadecimal, RGB, RGBA, HSL, and named colors*
- *Background properties: background-color, background-image, background-repeat, and background-size*
- *Background positioning and gradients*
- *Applying background effects*

TABLE OF CONTENTS

4: BOX MODEL AND LAYOUT

- *Understanding the CSS box model*
- *Margin, border, padding, and content areas*
- *Box-sizing property Display property and its values (inline, block, inline-block, flex, grid)*
- *Controlling element dimensions (width and height)*

5: POSITIONING AND LAYOUT TECHNIQUES

- *Positioning schemes: static, relative, absolute, fixed, and sticky*
- *CSS floats and clearing floats*
- *CSS positioning and stacking contexts*
- *Centering elements horizontally and vertically*
- *CSS Flexbox for responsive layouts*

6: RESPONSIVE WEB DESIGN WITH CSS

- *Introduction to responsive design principles*
- *Media queries and breakpoints*
- *Creating responsive layouts with CSS Grid*
- *Fluid typography and responsive images*
- *Testing and debugging responsive websites*

7: CSS TRANSITIONS AND ANIMATIONS

- *Transition properties: transition-property, transition-duration, and transition-timing-function*

TABLE OF CONTENTS

- *Creating smooth transitions Keyframes and animation properties*
- *CSS animations for interactive elements*
- *Cross-browser compatibility for animations*

8: CSS FLEXBOX AND GRID LAYOUT

- *In-depth exploration of CSS Flexbox for advanced layouts*
- *Advanced CSS Grid layout techniques*
- *Building complex responsive designs with Flexbox and Grid*
- *Overcoming common layout challenges*

JAVASCRIPT

VARIABLES AND DATA TYPES:

- *Variables: Used to store data.*
- *“var” vs “let” vs “const”*
- *Data Types: Include numbers, strings, booleans, arrays, objects, and more.*

OPERATORS

- *Arithmetic: +, -, *, /, %.*
- *Comparison: ==, ===, !=, !==, <, >, <=, >=.*
- *Logical: && (AND), || (OR), ! (NOT).*
- *Ternary Operator*

TABLE OF CONTENTS

CONTROL FLOW:

- *Conditional Statements: if, else if, else.*
- *Switch Statements: Used for multi-case branching.*

FUNCTIONS:

- *Blocks of code that can be called with arguments and can return a value.*
- *Types of functions: Arrow, Constructor, IIFE, anonymous functions.*
- *Function methods: call(),bind() and apply()*
- *What is closure ?*

ARRAYS:

- *Ordered collections of data.*
- *Arrays methods : pop(),push(),shift(),unshift() etc.*
- *Higher Order functions: Map, filter, reduce.*

OBJECTS:

- *Key-value pairs used to represent structured data.*
- *Accessing data : dot notation, bracket notation,*
- *Creating new key-values*

LOOPS:

- *for loop: Repeats code for a specified number of iterations.*
- *while loop: Repeats code while a condition is true.*
- *do...while loop: Similar to while loop, but code is executed at least once.*

TABLE OF CONTENTS

SCOPE AND CLOSURES:

- *Variables have different levels of visibility (scope).*
- *Lexical Scope*
- *Hoisting*
- *Functional scope, block scope & global scope.*
- *Closures allow inner functions to access variables from outer functions.*

DOM MANIPULATION:

- *Interacting with the Document Object Model (DOM) to manipulate web pages.*
- *DOM methods*
- *DOM Events*
- *Event Propagation: bubbling & capturing*

EVENTS:

- *Handling user interactions or actions on a web page*

ASYNCHRONOUS JAVASCRIPT:

- *Handling async javascript*
- *Event loop*
- *Async await.*
- *Try catch*
- *Promises: promises.all(), promises.race(), promises.any()*

TABLE OF CONTENTS

ES6 FEATURES

- *What is ES6*
- *A word on babel*
- *Block scope, let & const*
- *Template literals*
- *Arrow functions*
- *Spread and Rest operators*
- *Destructuring*
- *Classes - Inheritance, Static properties and methods*
- *Inheritance*
- *Static properties and methods*

TABLE OF CONTENTS

REACT

- *What is React ?*
- *What is virtual DOM ?*
- *Development setup Create-react-app Creating your first react app.*
- *What is JSX ?*
- *React children*
- *React Fragments*
- *Conditional Rendering*
- *List & Keys*
- *State vs props*
- *What is Props drilling ?*

REACT HOOKS:

- *useState(),useEffect(),useContext(),useReducer(),useMemo() etc.*
- *Understanding Components based architecture.*
- *Creating Reusable Components*
- *Handling forms in react*
- *Controlled & Uncontrolled Components*
- *Event Handling*

STYLING:

- *Conditional styling*
- *Inline*

TABLE OF CONTENTS

HANDLING API DATA:

- *Fetching data from APIs (e.g., Axios or Fetch)*

STATE MANAGEMENT:

- *useState*
- *Redux (Redux-toolkit)*
- *context*
- *api useReducer*

REACT ROUTERS:

- *Introduction to React Router*
- *Setting up routes and navigation*
- *Nested routes and route parameters*

ADVANCED REACT CONCEPT :

- *Higher Order Functions(HOC)*
- *Code splitting and lazy loading*
- *React Fibre*

REACT ECOSYSTEM:

- *Node package manager(Npm)*
- *axios*
- *React Developer Tools*
- *React Router*
- *Redux*
- *Styled components*

2 LIVE PROJECTS