

FULL Stack

(HTML, CSS, Bootstrap, JavaScript, JQuery, AngularJS2, NodeJS, ExpressJS, MongoDB, ReactJS)

Index

HTML:-

- ❖ HTML Introduction and Its Use in Websites
- ❖ Introduction of HTML
- ❖ Differences between HTML4 & HTML5
- ❖ Elements / Tags
- ❖ Attributes and with Practical Examples
- ❖ Forms
- ❖ Tables
- ❖ Audio and Video
- ❖ HTML5 New Elements

CSS:-

- ❖ CSS Introduction and Its Use in Websites
- ❖ CSS Implementations in HTML
- ❖ Animation Properties
- ❖ Background Properties
- ❖ Text Properties
- ❖ Layout Properties
- ❖ Selectors
- ❖ Elements
- ❖ Attributes
- ❖ Psuedo Classes

Bootstrap:-

- ❖ Bootstrap Basic
- ❖ Fixed Layouts
- ❖ Responsive Layouts
- ❖ Tables
- ❖ Images
- ❖ Buttons
- ❖ Forms
- ❖ Glyphicons
- ❖ Navigations
- ❖ Breadcrumbs

FULL Stack

- ❖ Dropdowns
- ❖ Paginations
- ❖ Helper Classes
- ❖ Modal Dialog Boxes
- ❖ Carousels

Javascript:-

- ❖ Javascript Basics and Its Functionalists
- ❖ Advanced Concepts of Javascript
- ❖ Operators
- ❖ Decision Making Statements
- ❖ Looping Statements
- ❖ Functions
- ❖ Events
- ❖ Cookies
- ❖ Dialog Boxes
- ❖ Objects
- ❖ Built-In Objects
- ❖ Error Handling
- ❖ Validations

jQuery Course:-

- ❖ jQuery Basics
- ❖ jQuery BuiltIn Functions
- ❖ jQuery HTML / CSS
- ❖ jQuery Form
- ❖ jQuery Events
- ❖ jQuery UI
- ❖ jQuery Interaction
- ❖ jQuery Widgets
- ❖ jQuery Effects

AngularJS2:-

- ❖ Introduction
- ❖ Bootstrapping Process
- ❖ Typescript Introduction
- ❖ Dev. An App with Data Binding & Directives
- ❖ Forms Inputs & Events
- ❖ Apply Styles to Templates, Components
- ❖ Dependency Injection
- ❖ Implementing Routes
- ❖ Implementation of HTTP & Observables

FULL Stack

NodeJS:-

- ❖ Introduction
- ❖ Environment Setup
- ❖ First Application
- ❖ REPL Terminal
- ❖ Packager Manager (NPM)
- ❖ Callbacks Concept
- ❖ Event Loop
- ❖ Event Emitter
- ❖ Buffers
- ❖ Streams
- ❖ File System
- ❖ Global Objects
- ❖ Utility Modules
- ❖ Web Modules
- ❖ Express Framework
- ❖ RESTFUL API
- ❖ Scaling Application
- ❖ Packaging

ExpressJS :-

- ❖ ExpressJS Basics
- ❖ Advanced Concepts of ExpressJS
- ❖ GET and POST
- ❖ Routing
- ❖ URL Building
- ❖ Middleware
- ❖ Templating
- ❖ Forms
- ❖ Database
- ❖ RESTFUL API's
- ❖ Scaffolding

MongoDB:-

- ❖ MongoDB Basics
- ❖ Environment Setup
- ❖ Database
- ❖ Collection
- ❖ CRUD - Documents
- ❖ Methods

FULL Stack

ReactJS:-

- ❖ **Introduction, Features, Advantages**
- ❖ **Setup and Environment**
- ❖ **Javascript eXtension JSX**
- ❖ **Components**
- ❖ **States**
- ❖ **Props**
- ❖ **Forms Handling**
- ❖ **Events Handling**
- ❖ **Router**
- ❖ **Flux**
- ❖ **Redux**

FULL Stack

HTML Course :-

❖ HTML Introduction and Its Use in Websites

❖ HTML4 & HTML5 :-

- Introduction of HTML
- Differences between HTML4 & HTML5
- Elements / Tags and Its Attributes and with Practical Examples :
 - HTML
 - Body
 - Head
 - Paragraph
 - Headings
 - Meta
 - Title
 - Anchor
 - Image
 - Formatting
 - Lists
 - Tables
 - Forms
 - HTML Entities
 - Audio and Video
 - Style
 - Div
 - Header
 - Footer
 - Nav
 - Section
 - Aside
 - Article
 - Script
 - **Attributes and Its Practical Examples :**
 - Title
 - Style
 - Width
 - Height
 - Src
 - Alt
 - Content
 - Align and more attributes with elements.

FULL Stack

CSS Course :-

❖ CSS Introduction and Its Use in Websites

❖ CSS Implementations in HTML :

- Inline
- Embedded
- External

❖ List of Properties and Its Practical Examples :

- Animation
- Background
- Border
- Colour
- Dimensions
- List
- Multi Column Layout
- Padding
- Margin
- Table
- Text
- Transform
- Visual Formatting
- Selectors
- Elements
- Attributes
- Psuedo Classes

FULL Stack

Bootstrap Course :-

❖ Bootstrap Basic :-

- Introduction, Advantages, Uses
- Get Started, Environment Setup
- Fixed Layout
- Fluid Layout
- Responsive Layout
- Typography
- Tables
- Lists
- Buttons
- Images
- Icons (glyphicons)

❖ Bootstrap Advanced :-

- Navs and Navbar
- Breadcrumbs
- Pagination
- Progress Bars
- Helper Classes
- Modals
- Dropdowns
- Tabs
- Tooltips
- Popovers
- Alerts
- Stateful Buttons
- Accordion Carousel
- Typeahead
- Scrollspy
- Affix

FULL Stack

Javascript Course :-

❖ Javascript Basics and Its Functionalists :-

- Introduction,
- Advantages, Limitations
- Syntax, Enabling & Disabling
- Placement of Javascript, Embedded, External
- Variables Creation

❖ Advanced Concepts of Javascript:

- **Operators :**
 - ❖ Arithmetic
 - ❖ Assignment
 - ❖ Comparison
 - ❖ Conditional / Ternary
 - ❖ Logical
- **Decision Making Statements :**
 - ❖ If
 - ❖ If...Else
 - ❖ Nested If...Else
 - ❖ Switch Case
- **Looping Statements :**
 - ❖ While
 - ❖ Do...While
 - ❖ For
 - ❖ For..In
 - Loop Controls : Break & Continue
- **Functions :**
 - ❖ Call & Define
 - ❖ Parameters
 - ❖ Return Statement
 - ❖ Constructor
 - ❖ Anonymous and Literal
 - ❖ Callbacks
 - ❖ Closures
- **Events :**
 - ❖ OnBlur
 - ❖ OnChange
 - ❖ OnClick
 - ❖ OnFocus
 - ❖ OnLoad
 - ❖ OnSubmit
 - ❖ OnKeyUp, OnKeyDown
 - ❖ OnMouseOut, OnMouseOver
 - ❖ etc.,
- **Cookies :**

FULL Stack

- ❖ Storing
- ❖ Retrieving
- ❖ Deleting
- ❖ Expiring
- Page Redirecting
- **Dialog Boxes** : Alert, Confirm & Prompt
- Void Function
- **Objects** :
- ❖ Object Creation
- ❖ Object Constructor
- ❖ UserDefined Function Object Creation
- ❖ Properties
- ❖ Methods Creation
- **Built-In Objects** :
- ❖ Number
- ❖ Array
- ❖ Date
- ❖ String
- ❖ Math
- ❖ Number
- ❖ RegExp
- **Error Handling** :
- ❖ Try
- ❖ Catch
- ❖ Throw
- ❖ Finally Blocks
- **Validations** :
- ❖ Number
- ❖ Text
- ❖ Range
- ❖ Email
- Empty etc.,.

FULL Stack

jQuery Course :-

❖ jQuery Basics :-

- What is jQuery
- History
- Advantages
- Examples
- Syntax
- Selectors

❖ jQuery Effects :-

- Show()
- Hide()
- Toggle()
- FadeIn()
- FadeOut()
- FadeToggle()
- FadeTo()
- SlideDown()
- SlideUp()
- SlideToggle()
- Animate()
- Delay()

❖ jQuery HTML / CSS :-

- Html()
- Text()
- Val()
- Css()
- Before()
- Prepend()
- After()
- InsertAfter()
- Append()
- AppendTo()
- Clone()
- Remove()
- Empty()
- Detach()
- ScrollTop()
- Attr()
- Prop()
- Offset()
- Position()
- AddClass()
- HasClass()
- ToggleClass()

FULL Stack

- Width()
- InnerWidth()
- InnerHeight()
- OuterWidth()
- OuterHeight()
- Wrap()
- WrapInner()
- WrapAll()
- UnWrap()

❖ jQuery Form :-

- Serialize()
- SerializeArray()

❖ jQuery Events :-

- Click()
- Bind()
- Blur()
- Focus()
- Select()
- Change()
- Submit()
- KeyDown()
- KeyPress()
- KeyUp()
- MouseEnter()
- MouseLeave()
- Hover()
- MouseOver()
- MouseOut()
- Load()
- UnLoad()
- Delegate()

❖ jQuery UI :-

- Introduction
- Features

❖ jQuery Interaction :-

- UI Draggable
- UI Droppable
- UI Resize
- UI Selectable
- UI Sortable

❖ jQuery Widgets :-

- UI Accordion
- UI Autocomplete
- UI Button
- UI Datepicker

FULL Stack

- UI Dialog
- UI Menu
- UI ProgressBar
- UI SelectMenu
- UI Slider
- UI Spinner
- UI Tabs
- UI Tooltip

❖ **jQuery Effects :-**

- UI Hide
- UI Show
- UI Toggle
- UI Add Class
- UI Remove Class
- UI Switch Class
- UI Animation
- UI Erasing
- UI Effect

FULL Stack

AngularJS2 Course :-

❖ Course Objectives :-

- Implement loops and conditions using structural directives
- Implement one way / two way data binding for data interpolation
- Implement event and error handling
- Implement dependency injection and reusability
- Create routing structure for components for navigation
- Implement unit testing using Jasmine framework
- Develop structurally well-formed web applications using components

❖ Introduction :-

- Introduction of AngularJS, Advantages
- Setting up Angular Development Environment
- Understanding Angular Framework
- Angular Architecture
- Angular Modularity
- **Component Structure :-**
- Decorators
- Selectors
- Template
- Style
- Component Class

❖ Bootstrapping Process:-

- Understanding Angular Bootstrap Process
- Understanding System Configuration
- Understanding Implementation of ngModule
- Building First Application

❖ Typescript Introduction :-

- Introduction of Typescript
- Configuring Typescript Transpilar
- DataTypes
- Variables
- Let
- Const
- Working with Class
- Working with Methods

❖ Dev. An App with Data Binding & Directives :-

- Components
- Attribute Directives
- **Structural Directives :-**
- *ngIf
- *ngFor
- *ngSwitch

- More..
- Implement Data Transformation using Pipes

❖ **Forms Inputs & Events :-**

- Implement Data Binding
- Implement One-Way & Two-Way Data Binding
- Implement Event Handling
- Implement Error Handling

❖ **Apply Styles to Templates, Components :-**

- Create External Style Sheets
- Implement Styles with in Component
- Applying Styles to Templates

❖ **Dependency Injection :-**

- Understanding Importance of Refactoring
- Creating Re-usable Components
- Implementing Dependency Injection to Component
- Implementing Promise Pattern

❖ **Implementing Routes :-**

- Implementing Route Path
- Associate Components to Path
- Route with Search Value

❖ **Implementation of HTTP & Observables :-**

- HTTP Client Overview
- Implement RxJS for HTTP Request & Response
- Fetch Data http.get
- Post Data http.post
- Processing JSON Data
- Error Handling
- Understanding Promises vs Observables

❖ **Sample Applications :-**

- Routing
- CRUD Operations
- Services and Dependency Injection
- Http and Observables
- Forms Handling
- Events Handling

FULL Stack

NodeJS Course :-

❖ Introduction :-

- Audience
- Pre-requisites
- About Node
- Execute Node
- Features
- Who use Node
- Concepts
- Where to use
- Where not to use

❖ Environment Setup :-

- Text Editor
- Nodejs Run Time
- Download NodeJs
- Installation
- Executing

❖ First Application :-

- Creating a NodeJs Application
- Make a request to NodeJs Server

❖ REPL Terminal :-

- What is REPL?
- Starting REPL
- REPL Commands
- Stopping REPL

❖ Packager Manager (NPM) :-

- Installing Modules using NPM
- Global vs Local Installation
- Using packages.json
- Attributes of packages.json
- Uninstalling Module
- Updating Module
- Searching Module
- Create a Module

❖ Callbacks Concept :-

- Using Props
- Default Props
- State and Props

❖ Event Loop :-

- Validating Props

❖ Event Emitter :-

- Set State
- Force Update
- Find DOM Node

- ❖ **Buffers :-**
 - Lifecycle Methods
- ❖ **Streams :-**
 - Simple
 - Complex
- ❖ **File System :-**
 - Simple
 - Child
- ❖ **Global Objects :-**
 - What is Refs
 - Using Refs
- ❖ **Utility Modules :-**
 - What is Keys
 - Using Keys
- ❖ **Web Modules :-**
 - Install a React Router
 - Add a Router
 - Create Components
- ❖ **Express Framework :-**
 - What is Flux
 - Flux Elements
 - Flux Props
- ❖ **RESTFUL API :-**
 - Install Redux
 - Create Files and Folders
 - Actions
 - Reducers
 - Store
 - Root Component
 - Other Components
- ❖ **Scaling Application :-**
 - Install React CSS Transitions Group
 - Add a CSS File
 - Appear Animation
 - Enter and Leave Animations
- ❖ **MongoDB :-**
 - Connecting Node and MongoDB
 - Database Creation, Drop
 - Collection Operations
 - Documents Operations
- ❖ **Application :-**
 - Node and MongoDB Application

FULL Stack

ExpressJS Course :-

❖ ExpressJS Basics :

- Introduction, Advantages
- Environment Setup
- Basic Application

❖ Advanced Concepts of ExpressJS :

- Request
- Response
- GET
- POST
- Routing
- HTTP Methods
- URL Building
- Middleware
- Templating
- Static Files
- Form Data
- Database
- Cookies
- Sessions
- File Upload
- Authentication
- REST FUL API's
- Scaffolding
- Debugging

FULL Stack

MongoDB Course :-

❖ MongoDB Basics :

- Introduction, Advantages
- History, Features
- No SQL Databases
- Advantages over RDBMS

❖ Environment Setup :

- Install MongoDB
- MongoDB Shell
- MongoDB Data Model
- MongoDB Datatypes

❖ Database :

- Create Database
- Drop Database

❖ Collection :

- Create Collection
- Drop Collection

❖ CRUD : Documents :

- Insert Documents
- Update Documents
- Delete Documents
- Query Documents

❖ Methods :

- Limit()
- Sort()
- Skip()
- More..

FULL Stack

ReactJS Course :-

❖ Introduction :-

- Audience
- Pre-requisites
- About React
- Features
- Advantages
- Limitations

❖ Environment Setup :-

- Create of Root Folder
- Install Global Packages
- Add Dependencies and Plugins
- Create the Files
- Set Compiler, Server and Loaders
- Index.html
- App.jsx and Main.js
- Running the Server

❖ JSX :-

- What is JSX
- Using JSX
- Nested Elements
- Attributes
- Javascript Expressions
- Styling
- Components
- Naming Convention

❖ Components :-

- Stateless
- Stateful

❖ State :-

- What is State?
- Props

❖ Props Overview :-

- Using Props
- Default Props
- State and Props

❖ Props Validation :-

- Validating Props

❖ Component API :-

- Set State
- Force Update
- Find DOM Node

❖ Component Life Cycle :-

- Lifecycle Methods

FULL Stack

❖ **Forms :-**

- Simple
- Complex

❖ **Events :-**

- Simple
- Child

❖ **Refs :-**

- What is Refs
- Using Refs

❖ **Keys :-**

- What is Keys
- Using Keys

❖ **Router :-**

- Install a React Router
- Add a Router
- Create Components

❖ **Flux Concept :-**

- What is Flux
- Flux Elements
- Flux Props

❖ **Using Flux and Redux :-**

- Install Redux
- Creating Components
- Working with States
- Composing Components
- Actions
- Event Handlers
- Reducers
- Store
- Root Component
- Other Components
- Unit Testing – Tools, React, Redux

❖ **Animations :-**

- Install React CSS Transitions Group
- Add a CSS File
- Appear Animation
- Enter and Leave Animations

❖ **Higher Order Components :-**

- What is Higher Order Components

FULL Stack