Full Stack Web Development using Java

Module 1 – Course Overview

- Overview of Web And web application
- Overview of Mobile and mobile applications
- Web VS Mobile
- Web Designer vs HTML Developer

Module 2 - Core Concepts Of HTML

Elements and Attributes

- Tags
- Introduction
- Types
- Single Tags / Empty Tags / Void Tag / Self Enclosing Tag
- Paired Tags
- Semantic Tags
- Non-Semantic Tags
- Elements
- Introduction
- Types
- Inline elements
- Block Level
- Inline-Block
- Attributes
- Introduction
- Types
- Predefined Attributes
- Global Attributes
- Style
- Id
- class
- Element Specific Attributes
- src
- alt
- type
- Custom / User Defined Attributes

Text tags

- Headings
- h1 to h6
- Paragraph
- Lists
- Ordered List
- Unordered List
- Definition List
- Tables
- article
- aside
- Image Tag
- Figure tag

Links

- Anchor
- Internal linking (Within Project)
- External Linking (from our project to external pages)
- (#) Hash-Based navigation (Within page)
- Text tags
- Graphic Tags

Multi Media Tags

- Audio
- Video

Layout Tags

- Div
- Section
- Header
- Main
- Nav
- Footer

Form tags and form attributes

- Form
- Form attributes
- method
- action
- enc-type

- autocomplete

Form Elements & form element attributes

- Types
- Input types
- text
- password
- file
- hidden
- radio
- checkbox
- submit (Action Items)
- reset (Action Items)
- Dropdown Box
- select
- option
- Attributes of form elements

Module 3 : CSS Introduction

- What is CSS
- Why Use CSS?
- How is CSS Used?
- Creating and Linking a CSS StyleSheet
- How To Test A Stylesheet

Selectors

- What is a selector?
- The Universal/Global Selector
- The Tag/Element Selector
- The ID Selector
- The Class Selector
- The Attribute Selector
- The Relationship Selector
- Psuedo Selector
- Group Selector
- Specificity & When To Use Selectors
- What are Properties?
- The CSS General Rule

Coloring & Formatting

- Types of Colors

- Coloring Text
- Background colors
- Images/URLs in CSS Other background Properties
- Opacity / Transparency
- Gradients

Fonts & text manipulation

- Introduction to types of units
- Text Manipulation
- Font Size, Bolding & Style
- Font Families
- Including external fonts W/Google Fonts
- Using External Fonts

Layout

- Introduction
- The Box Model
- Padding
- Border
- Margin
- Changing Content Size
- CSS Borders
- Margin & Padding
- Float & Display Types

Flexbox

- Introduction
- Flexbox Components
- Creating a Flex Container
- Flex Direction & Wrap
- Content Alignment
- Flex Item Order
- Shrink, Grow & Basis
- The 'Flex' Property
- Item Alignment

Grid

- Grid vs Flexbox
- Introduction
- Differences
- Creating a Grid

- Template Columns & Rows
- Justify & Align Grid
- Row & Column Gaps
- Column & Row Lines
- Grid Area

Animation & Transitions

- The Transitions Property
- Transform property
- Creating Animations w/ Key frames
- Adding an Animation
- Animation Properties

Media Queries

- Media Queries in css
- Introductions
- Properties
-

Module 5: Javascript

What is Javascript? Use of javascript?

Trend of Javascript

Types of Implementing Javascript

- Inline
- Internal
- External

Basics of Javascript

- What are the Datatypes? Example for each one of them
- Variables & Constants

Operators

- What is Operators?
- Types of Operators

Conditions / Decision making

- What is decision-making
- Types of Conditions
- Examples for each one of them

Loops / Iterations

- What are loops? Usage?
- Types of loops
- Entry controlled vs exit controlled loops

DataStorage Techniques & DOM Manipulation

- What are the types of data storage?
- Strings & Methods
- Arrays & Methods
- Objects & Methods
- Array of Objects

DOM & Dom Manipulations

- What is DOM & Dom Manipulation Methods?
- Types of DOM Manipulation Methods
- Dom Selectors
- getelementById()
- getElementByTagName()
- getElementByClassName()
- querySelector()
- querySelectorAll()
- Attaching Event
- addEventListener()
- Detaching Event
- removeEventListener()
- Manipulating Elements
- createElement()
- appendChild()
- removeChild()
- replaceChild()
- Manipulating Attributes
- setAttribute()
- removeAttribute()
- Adding Css using Javascript
- document.getElementById('idvalue').style.property =
- 'value'
- Classlist
- addClass
- removeClass
- InnerHTML

- Document.write() vs innerHTML

Forms & Events

- What is forms and validations?
- Events and Event handling
- Types of form validations
- field validations
- form validation
- Regular expressions
- form validation using regex
- form validation without regex

Error Handling

- What are the types of errors in JS?
- syntax errors and type errors
- assignment errors & reference errors
- How to handle the errors?
- What is exception handling?

OOJS (Object oriented Principles)

- Data abtraction
- Data Encapsulation
- ES5
- Object Literal
- Object Constructor
- Es6+
- Classical
- Polymorphism
- Operators Overloading
- Function Overloading
- Inheritance
- Types of inheritance
- Js Supported Types

Module 6: React js

Introduction to React.js:

- What is React.js?
- Why use React.js?
- React.js vs Other JavaScript Frameworks/Libraries.

Setting Up React:

- Installing Node.js and npm.
- Create React App.
- Understanding the project structure.

React Components:

- Functional Components.
- Class Components.
- State and Props.
- Component Lifecycle Methods.

JSX (JavaScript XML):

- Syntax and usage.
- Embedding expressions in JSX.
- JSX vs HTML.

Handling Events in React:

- Event handling in React.
- Binding event handlers.
- Event delegation.

React State and Props:

- State and its importance.
- Updating state.
- Props and PropTypes.

Conditional Rendering:

- Using if statements.
- Ternary expressions.
- Logical && operator.

Lists and Keys:

- Rendering lists.
- Keys and their significance.
- Dynamic lists.

Forms in React:

- Controlled components.
- Handling form submissions.
- Form validation.

React Router:

- Navigation in a React application.
- Route parameters.
- Nested routes.

Managing State in React:

- Local state vs global state.
- Context API.
- State management libraries (e.g., Redux).

Hooks in React:

- Introduction to hooks.
- useState, useEffect, useContext, etc.
- Custom hooks.

Higher-Order Components (HOCs):

- What are HOCs?
- Creating and using HOCs.

Error Handling in React:

- Error boundaries.
- componentDidCatch method.

Testing React Applications:

- Unit testing with Jest and Enzyme.
- React Testing Library.

Optimizing Performance:

- Memoization.
- PureComponent and React.memo.
- Virtual DOM and its role.

Server-Side Rendering (SSR) and Next.js:

- Benefits of SSR.
- Introduction to Next.js.

React and REST APIs:

- Fetching data from APIs.
- Axios and other HTTP libraries.

Authentication and Authorization:

- User authentication.
- Authorization with React.

Deployment of React Applications:

- Building and deploying React apps.
- Platforms like Netlify, Vercel, etc.

Integration with Backend Technologies:

- Connecting React with various backend technologies (Node.js, Django, Flask, etc.).

Module 7: Bootstrap

Introduction to Bootstrap:

- What is Bootstrap?
- History and background.
- Advantages of using Bootstrap.

Getting Started with Bootstrap:

- Downloading and installing Bootstrap.
- CDN integration.

Bootstrap Grid System:

- Understanding the grid system.
- Grid classes for different screen sizes.
- Nesting columns.

Bootstrap Typography:

- Headings and paragraphs.
- Text alignment and styling.
- Responsive text.

Bootstrap CSS Classes:

- Styling buttons.
- Typography classes.
- Background and text color classes.

Bootstrap Components:

- Navigation bar.
- Forms and input groups.
- Dropdowns and alerts.

Bootstrap Layout Components:

- Containers and container-fluid.
- Jumbotron.
- Cards and card groups.

Responsive Design in Bootstrap:

- Responsive utilities.
- Visibility classes.

Bootstrap Navigation Components:

- Navs and navbars.
- Breadcrumbs.
- Pagination.

Bootstrap Modals:

- Creating and triggering modals.
- Modal options and events.

Bootstrap Carousel:

- Creating image sliders.
- Carousel options and customization.

Bootstrap Forms:

- Basic form structure.
- Form validation.
- Input types and styling.

Bootstrap Icons:

- Using built-in icons.
- Icon customization.

Bootstrap Tables:

- Basic table structure.
- Table styles and classes.
- Responsive tables.

Bootstrap Alerts and Badges:

- Creating alerts.
- Using badges for notifications.

Bootstrap Images:

- Responsive images.
- Image shapes and sizes.

Bootstrap Utilities:

- Spacing and margin classes.
- Text and background utilities.

Customizing Bootstrap:

- Theming and customization.
- Overriding default styles.

Integration with JavaScript:

- Using Bootstrap JavaScript plugins.
- Modal interactions with JavaScript.

Accessibility in Bootstrap:

- Designing accessible websites.
- ARIA roles and attributes.

Using Bootstrap with Other Libraries:

- Integrating Bootstrap with jQuery.
- Bootstrap and Angular, React, or Vue.js.

Bootstrap and Responsive Web Design:

- Creating a responsive layout.
- Media queries in Bootstrap.

Bootstrap and Flexbox:

- Integrating Flexbox for layout.
- Flexbox utility classes.

Bootstrap and Sass:

- Working with Sass in Bootstrap.
- Customizing variables.

Building a Responsive Website with Bootstrap:

- Putting it all together.
- Tips for responsive web design.

Bootstrap Best Practices:

- Optimal use of classes and components.
- Performance considerations.

Bootstrap Extensions and Plugins:

- Third-party plugins.
- Extensions for specific use cases.

Troubleshooting in Bootstrap:

- Common issues and solutions.
- Debugging Bootstrap applications.

Latest Features in Bootstrap:

- Updates and new features.
- Keeping up with Bootstrap releases.

Real-world Projects with Bootstrap:

- Building a portfolio.
- Creating a landing page.

Module 8: Material UI

Introduction to Material-UI:

- What is Material-UI?
- Advantages of using Material-UI.
- Integration with React.

Getting Started with Material-UI:

- Installing Material-UI.
- Setting up a basic Material-UI project.
- Basic usage of Material-UI components.

Material Design Principles:

- Understanding the principles of Material Design.
- Material-UI's adherence to these principles.

Material-UI Components:

- Buttons and icons.
- Typography.
- Cards and Paper components.

Material-UI Layout:

- Grid system.
- Containers and spacing.
- Responsive design with breakpoints.

Material-UI Styling:

- Theming in Material-UI.
- Customizing styles using makeStyles.
- Overriding default styles.

Material-UI AppBar and Navigation:

- Creating an AppBar.
- Navigation with Drawer and Tabs.
- Breadcrumbs.

Material-UI Forms:

- Textfields and forms.
- Selects and Autocomplete.
- Form validation.

Material-UI Data Display Components:

- Tables and DataGrid.
- Lists and Menus.
- Tooltip and Popover.

Material-UI Dialogs and Modals:

- Creating dialogs.
- Modals and popups.
- Snackbar for notifications.

Material-UI Icons:

- Usage of Material Icons.
- Customizing icons.

Material-UI SVG Icons:

- Using SVG icons.

- Customizing SVG icons.

Material-UI Grid System:

- Understanding the Grid component.
- Responsive layout with Grid.

Material-UI Themes:

- Creating and customizing themes.
- ThemeProvider.

Material-UI State Management:

- Using local state.
- Integration with state management libraries (Redux, MobX).

Material-UI Routing:

- React Router integration.
- Navigation with Link.

Material-UI Form Libraries Integration:

- Integration with Formik.
- Using Yup for form validation.

Material-UI Animation:

- Transition component.
- React Spring for advanced animations.

Material-UI Testing:

- Testing components with Jest.
- React Testing Library.

Material-UI Accessibility:

- Designing accessible interfaces.
- ARIA roles and attributes.

Material-UI and Server-Side Rendering (SSR):

- Implementing SSR with Next.js.
- Server-side rendering considerations.

Material-UI and Mobile Development:

- Responsive design for mobile.
- Touch and gesture support.

Material-UI and Theming:

- Creating dark themes.
- Dynamic theming.

Module 9: Introduction To Java

Introduction To Java

- History of Java
- What is Java , Java Flavors, characteristics
- JVM Architecture
- Bytecode
- Class Loader
- Unicode
- Class path
- Path

Fundamentals of Java Programming

- Object oriented concepts (OOP)
- Keywords, Datatypes, Variables, Operators, Casting
- Selection statement (if, switch)

Control flow , Conditional Statements

- Control statements (while, do while , for)
- Conditional statements (if, else, elseif)
- Static
- Arrays

Object Oriented Programming with Java

- Classes and Objects
- Structure of a class – its internals (Data Members,
- methods)
- Using static
- Constructor
- This keyword
- Modifiers
- Playing with the object (copying, casting)
- Garbage collection
- Abstract class

Inheritance

- Basics (extends keyword)

- Modifiers and their scope
- Deriving a class
- Super, final keyword
- Why java does not support multiple inheritance?

Polymorphism

- Overloading a method
- Overloading a constructor
- Method overriding
- Accessing base class method

Packages and Interfaces

- Basics
- Modifiers and their scope chart
- Setting classpath
- Compiling and accessing a packaged class
- Types of packages
- User-defined package

Exploring java.lang package

- String, StringBuffer, Arrays
- Wrapper classes

Exception Handling

- Basics
- Hierarchy of exceptions
- Handling exception – Try, catch, finally, throw, throws
- User defined exceptions
- Basics, Thread class , Runnable Interface
- Thread model
- Life cycle – start(), run()
- Scheduling
- Deadlocks / concurrency issues
- Synchronization – as a block, as a modifier
- Daemon thread

I/O Streams

- Introduction
- Hierarchy of streams
- IO Stream, Byte Stream, Character Streams
- Buffered Input Stream, BufferedOutputStream

- Reader and Writer class
- Buffered Reader, Print Writer
- Serialization

Collection Framework

- Basics, hierarchy
- Legacy classes – Vector, Queue, Stack, Enumeration,
- Dictionary, Properties
- List, ArrayList, LinkedList
- Set, HashSet, TreeSet,
- Map, HashMap, TreeMap,
- Generics
- Annotations
- Boxing / Unboxing
- Enums

Spring core

- Types of containers
- Xml configuration Core annotations
- Component – scan
- What can be injected into a bean
- Understanding core annotations
- Spring MVC
- Handler Mapping
- Controllers
- View resolvers
- Validators
- Interceptors
- Spring DAO
- Jdbc templates
- Exception hierarchy
- Spring ORM
- Hibernate template
- Integration with hibernate
- Spring AOP
- Point cut
- Advisors
- Types of advice
- Spring – webservices
- Spring security
- Spring JPA

Module 10: Hibernate

Introduction to Hibernate:

- What is Hibernate?
- Object-Relational Mapping (ORM) concept.
- Advantages of using Hibernate.

Setting Up Hibernate:

- Downloading and configuring Hibernate.
- Integrating Hibernate with Java projects.
- Hibernate configuration file (hibernate.cfg.xml).

Mapping Entities:

- Defining entity classes.
- Mapping entities to database tables.
- Primary keys and generated values.

Hibernate Annotations:

- @Entity, @Table, @Id, and other annotations.
- Configuring relationships with @OneToMany, @ManyToOne, etc.
- @GeneratedValue and @Temporal annotations.

Hibernate XML Mapping:

- Mapping entities using XML.
- Advantages and disadvantages of XML mapping.

Hibernate Session and SessionFactory:

- Opening and closing sessions.
- SessionFactory configuration and creation.

CRUD Operations with Hibernate:

- Saving and updating entities.
- Retrieving entities by primary key.
- Deleting entities.

Hibernate Query Language (HQL):

- Basic HQL syntax.
- Named queries.
- Parameters and pagination.

Criteria API:

- Creating queries with the Criteria API.
- Query by Example (QBE).

Native SQL Queries:

- Executing native SQL queries with Hibernate.
- Result set mapping.

Caching in Hibernate:

- First-level and second-level caching.
- Configuring caching in Hibernate.

Lazy Loading and Eager Loading:

- Fetching strategies.
- Lazy loading and proxy objects.

Hibernate Transactions:

- Managing transactions with Hibernate.
- ACID properties.

Transaction Isolation Levels:

- Understanding different isolation levels.
- Configuring isolation levels in Hibernate.

Batch Processing with Hibernate:

- Executing batch inserts and updates.
- Configuring batch processing.

Listeners and Callbacks:

- Entity listeners.
- Callback methods (@PrePersist, @PostLoad, etc.).

Inheritance Mapping in Hibernate:

- Table per class hierarchy.
- Table per subclass.
- Table per concrete class.

Associations in Hibernate:

- One-to-One, One-to-Many, and Many-to-Many relationships.
- Bidirectional and unidirectional associations.

Fetching Strategies:

- Select, subselect, and join fetching.
- Fetching strategies for associations.

Hibernate Validation:

- Bean Validation (JSR 303/349) in Hibernate.
- Validating entities.

Integration with Spring:

- Integrating Hibernate with the Spring framework.
- Spring Data JPA.

Hibernate and Web Applications:

- Using Hibernate in web applications.
- Session-per-request pattern.

Optimistic and Pessimistic Locking:

- Versioning and optimistic locking.
- Pessimistic locking.

Auditing with Hibernate Envers:

- Enabling auditing in Hibernate.
- Retrieving historical data.

Hibernating Spring Boot Applications:

- Setting up Hibernate in Spring Boot.
- Spring Data JPA with Hibernate.

Multi-Tenancy in Hibernate:

- Configuring multi-tenancy.
- Database and schema-based multi-tenancy.

Performance Tuning in Hibernate:

- Batch processing for performance.
- Indexing and query optimization.

Security Considerations in Hibernate:

- Preventing SQL injection.
- Secure coding practices.

Hibernate and NoSQL Databases:

- Integrating Hibernate with NoSQL databases.
- Considerations for non-relational data stores.

Module 10: MySQL

Introduction:

- What is SQL
- what are Relational Databases?
- RDBMS Benefits and Limiations
- SQL vs NoSQL Databases

Basics Syntax - I:

- SQL Keywords
- Data Types
- Operators

Basics Syntax - II:

- DDL
- DML
- DCL
- TCL

Aggregate Queries:

- SUM
- COUNT
- AVG
- MIN
- MAX
- GROUP BY
- HAVING

Data Constraints:

- Primary Key
- Foreign Key
- Unique
- NOT NULL
- CHECK

Join Queries:

- INNER JOIN
- LEFT JOIN
- RIGHT JOIN
- FULL OUTER JOIN

Advanced SQL - I:

- Sub Queries
- Conditional functios
- Views
- Indexes

Advanced SQL - II:

- Recursive Queries
- Window Functions