



# **FULL STACK COURSE CONTENT**

## **Semester 1**

### **Module 1 – HTML**

#### **Introduction To Web And Basic HTML Tags**

- Pieces that make the web work
- Introduction to front-end
- Basic structure of an HTML page
- Attributes, elements and relationships
- Comments
- Basic tags – Paragraph, headings, hr, br
- Basic attributes – ID and class
- Anchor tags
- Including CSS
- Including JS

#### **Structuring Content Using HTML Tags**

- Classification of elements – Block and inline
- Container tags – Div and span, and when to use them
- Semantic tags
- Including media – Images, audio and video
- Responsive media
- Lists – Ordered and unordered list
- Unstyled lists and list-based menus
- Presenting tabular data
- Styling tables

#### **Working with Forms**

- Introduction to forms
- What happens when a form is submitted
- Pieces of a simple form – Form, text input with name attribute, submit button
- Submitting data to another page (action attribute)
- Responsive media
- Changing the HTTP method used for submission (method attribute)
- Adding a label
- Default value for inputs
- Types of buttons
- Various input types – Password, textarea, checkbox, radio buttons, select dropdown, file input
- HTML5 input types – Email, URL, number, date
- HTML5 input and form attributes – Required, placeholder, novalidate

#### **Deeper Understanding of Client-Server Communication**



- Symbols in HTML
- HTTP Basics
- Introduction to chrome dev tools – The network tab
- Understanding parts of a URL
- Further exploration

## **Module 2 – CSS**

### **Introduction to CSS, CSS Inheritance and Various Selectors**

- Including CSS
- Choosing the way to include CSS
- User-agent styles
- Parts of a CSS rule
- Selectors – Universal, ID, class, type, attribute
- Pseudo classes – Link-related classes
- Structural Pseudo classes – :first-letter, :first-line, :first-child, :last-child, :nth-child
- Combinators – No spaces, child selector, descendant selector, combining selectors using comma
- Inheritance – Initial and inherit values

### **The Cascade, Specificity and Style Resolution**

- Cascade – User-agent vs author styles
- Reset stylesheet
- Normalize Stylesheet
- Cascade of embedded styles, external stylesheets, inline styles
- Introduction to specificity – how selectors affects styles applied
- Specificity – ID vs class vs type selector
- Specificity calculation and style resolution in case of complex selectors
- Use of !important
- Units for CSS properties – Unitless and unitless, absolute vs relative
- Default values for properties (initial values)
- Computed and resolved styles

### **Important CSS Properties**

- Box model in depth
- Various ways of specifying colors
- Background properties
- Using background properties with image sprites
- Typography related properties
- Handling overflow
- Hiding elements
- Floats

### **Deeper Understanding of Client-Server Communication**

- Introduction to responsive web design (RWD)
- Fluid layouts
- Fluid images
- Media queries



- Further exploration

## **Module 3 – JavaScript**

### **Introduction, Variables, Scopes & Data Types**

- History of JavaScript
- Setting up the environment
- Running JavaScript in the browser and Node
- Comments
- Variables and primitive data types
- Falsy and truthy values
- Variable scopes, scope chain
- Using arrays
- The type of operator

### **Operators, Control Flow & Functions**

- Variable hoisting
- Operators and expressions
- Control flow – Branching and looping
- Function declaration and usage
- Anonymous function and function expression
- IIFE
- Function hoisting
- Functions call context (this keyword)
- Handling variable number of arguments
- Callbacks – passing functions as arguments

### **Functions (Continued) & Objects**

- Returning functions
- Closures
- Object declaration using literal syntax
- Accessing properties and methods
- Adding and deleting properties
- Listing object properties – for..in loop and object.keys()
- Constructor function and the new keyword
- Function – functions as objects
- call(), apply() and bind() as methods of functions

### **Built-in Objects and Functions**

- Array methods
- Date methods
- JSON
- Number methods
- String methods

### **Exception Handling & Browser Objects**



- Strict mode execution
- Error objects
- Exception handling
- Window
- Navigator
- Location
- History

## **DOM and Event Handling**

- The document object
- Nodes and the DOM tree
- Node relationships and DOM tree traversal
- Methods for DOM manipulation
- Various browser events
- Different ways to handle events
- Event object properties and methods

## **Introduction to ES6**

- Introduction, setting up, running ES6 code, JS transpilers
- Scope, let & const
- Template literals, default parameters
- Destructuring arrays & objects
- Rest & spread

## **More ES6**

- Arrow functions
- Classes & inheritance
- Modules
- Promises

## **Module 4 – TypeScript**

### **Basics of TypeScript**

- Introduction to TypeScript
- Why TypeScript?
- Setting up TypeScript
- JavaScript vs TypeScript
- Type annotations, variable declarations, basic datatypes, type inference
- Advanced types, type erasure and error behavior
- Classes, constructors & methods, inheritance & polymorphism, access modifiers

### **Deeper Dive into TypeScript**

- Interfaces, properties & methods, interfaces & classes, extending interfaces
- Namespaces, namespaces using multiple files
- Modules, import, export, namespaces vs modules
- Generics, generic functions, generic classes



## **Module 5 – Angular**

### **Introduction to Angular**

- What is Angular?
- Why Angular?
- Angular versions
- Where does Angular fit?
- Multi page application (MPA)
- Single page application (SPA)
- Setting up Angular
- Create your first Angular app
- Serve your Angular app
- Edit your first Angular component

### **Project Structure, Modules, Bootstrapping**

- Understand project structure
- Modules
- Decorators
- Bootstrapping
- Add Bootstrap to our app
- Use Bootstrap in our app

### **Data Binding and Component Interaction**

- Data binding
- Component interaction

### **Directives and Pipes**

- Directives
- Pipes

### **Services and Dependency Injection**

- Services
- Dependency injection

### **Forms and Validation**

- Angular forms: Template driven forms & reactive forms

### **Routing & Navigation, Introduction to RxJS Observables**

- Building SPAs using routing
- Understanding observables

### **Communicating with Server, Handling Errors, Implementing Authentication in Angular Applications**

- Server communication using Http Client



- Error handling
- Implementing authentication in Angular

## **Semester 2**

### **Module 1 – Java**

#### **Coding Fundamental and Basics of Programming**

- Basics of programming
- Programming environment
- Fundamental keywords
- Basic operators
- Decision making operators
- Loop statements
- Functions in the program
- File I/O

#### **Introduction of Java with Commands**

- History of Java
- Operating environment
- Eclipse configuration
- Identifiers
- Classes and objects
- Declarations
- Java operators

#### **Concept of OOPs**

- OOPS
- Abstraction
- Encapsulation
- Inheritance
- Polymorphism

#### **Classes, Interfaces, Enum etc.**

- Classes
- Interfaces
- Enum

#### **Control Statements**

- If statement
- While statement
- For
- Do-while
- Continue
- Break

#### **I/O and Arrays**



- Sending input to program
- Displaying output
- Displaying output as error
- Arrays
- One dimensional arrays
- Two dimensional arrays

## **String, Math, Formatters etc.**

- Strings
- Identifying length of string
- String comparing
- Searching within string
- Replacing string
- Math functions
- Formatting strings

## **Packages**

- Naming conventions in packages
- Creating package
- Importing all classes in package

## **Exception Handling**

- Why exception handling
- Keywords in exception handling
- Identifiers
- Try-catch
- Throw
- Finally

## **Wrapper Classes**

- Wrappers
- Creating wrapper objects
- Builtin wrapper conversion utilities

## **Generic**

- Why generics
- Type safety
- Create generic methods
- Generics constructors
- Generics interfaces
- Generics restrictions

## **Collections**

- Goals of collection
- Collection interfaces
- List



- Iterator
- Map

## **Files & Threads**

- Reading a file
- Writing a file
- Appending to existing file
- Check file exists
- Delete file, asynchronous process
- What is thread
- Run a method parallelly

## **Anonymous Inner Class**

- Inner classes
- Nested top level class
- Member class
- Local class
- Anonymous classes

## **JAVA 8 – Lambda Expression and Default Methods**

- Lambda expressions
- Convert interfaces to Lambda expressions
- Convert anonymous classes to Lambda expressions
- Default methods

## **Predefined Functional Interface**

- Identify functional interface
- Create functional interface
- Categories of functional interface
- Predicates
- Consumer
- Functions
- Supplier

## **Streams**

- Streams
- forEach
- IntStream
- Map
- Filter
- Limit
- Skip
- Collect

## **Date & Time API**

- Date & time objects





- Manipulate date operations
- Date formatters

## **Module 2**

### **Maven**

- Introduction
- Installation & setup
- Creating maven project
- Build lifecycle
- Dependency management
- Excluding maven dependencies

### **Tomcat**

- Architecture
- Installing Tomcat
- Running Tomcat
- Shutdown Tomcat
- Running two servers

### **Spring Boot Basics CLI**

- How to create a spring boot application in IDE Spring Tool Suite
- How to create a spring boot application from Spring Initializr

### **Spring Boot Restful Services**

- Restful webservices
- PostMan
- REST operations
- Create
- Read
- Update
- Delete

### **Spring Boot JPA**

- Persistence
- ORM
- H2
- CRUD operations using JPA

### **Actuator**

- Production ready features
- Integrate Spring Boot Actuator
- Enable all metrics in Actuator
- Health Endpoint
- Customize health Endpoint
- Env Endpoint



- Info Endpoint
- Metrics Endpoint

## **Logging, Profiles & Security**

- Logging configuration
- White label error page
- Spring profiles & configuration
- Spring security using JWT

## **Spring Boot Restful Services**

- Aspect oriented programming
- Caching

## **Module 3**

### **JPA**

- Spring data overview
- Repository overview
- Spring data JPA
- Crud Repository
- Paging And Sorting Repository
- Jpa Repository
- Query methods
- Named & native queries
- Async queries

### **MongoDB**

- NoSQL
- Key value pair
- Column family
- Graph
- Document based
- MongoDB vs RDBMS
- Install MongoDB & Compass
- Databases
- Collections
- CRUD document
- Projection
- Sort, skip & limit
- Indexing & aggregation
- Backup & restore

## **Module 4**

### **DevOps & Cloud deployment**

- Jenkins



- Git
- Packaging & distributing
- Deployment on AWS

## **Module 5**

### **Placement Services**

#### **Career Services**

- Aptitude
- Interview prep sessions
- Mock interviews
- Resume & LinkedIn profile building
- Github portfolio