Java Full Stack Developer – Syllabus

Java Full Stack – course will advance your career as a **Full Stack Developer.** You'll learn **top skills demanded** in the industry, including **Angular**, **React JS**, **Spring**, **Spring Boot**, and **JSPs**, as well as **MVC**, **Micro Services, Database** to build highly scalable **web application**.

Sl. no	Topic	Hrs
	Technologies / Skills Covered	
	Core & Advanced Java + J2EE + Spring + Web Technologies (Html, Css, Bootstrap) + Frontend Technologies (Angular & ReactJs) + Database (Oracle SQL)	
1	**Agile, Scrum, Git, Github and AWS (Introduction to Cloud Computing) (Topics wise + Assignments Discussions)**	
	Understanding the Principles of Agile & Scrum, An Overview of Git, Gihub and Git Rebase, and AWS	
2	**Core and Advanced Java (Topics wise + Assignments Discussions)**	
	Strong Core Java Experience with OOP Data Structures	
3	**J2EE (Topics wise + Assignments Discussions)**	
	Core Java Programming, J2EE, JSF, JDK, UML, Algorithm and Web Services	
4	**Spring, Spring Boot (Topics wise + Assignments Discussions)**	

5	**Web Technologies: Html, Css & Bootstrap (Topics wise + Assignments Discussions)**	
	Strong knowledge in Creating Web Pages with Responsive using Html5, Css3 and Bootstrap5	
6	**Javascript, Ecmascript & Typescript (Topics wise + Assignments Discussions)**	
	Strong Knowledge in Writing the Client-Side Programming, and Additionally adding the Pre – Requisite Typescript of Angular	
7	**Angular (Topics wise + Assignments Discussions)**	
	Well Developed Single Page Applications can be builtup using Angular2v, 4v, 8v, 12v	
8	**ReactJs (Topics wise + Assignments Discussions)**	
	Well Developed Single Page Application can be builtup using ReactJs16v	
9	**Oracle SQL (Topics wise + Assignments Discussions)**	
	Strong knowledge in Oracle SQL Database	
10	**DevOps**	
11	**Project Development & Deployment**	
1	**Git, Github and AWS**	
	Git Introduction	

	Github Introduction	
	Git Commands	
	Introduction to Cloud Computing	
	Identity Access Management	
	Glacier Storage	
	Compute	
	Route S3	
	Databases	
	VPC (Virtual Private Cloud)	
	Security Options	
	Application Services	
2	**Core and Advanced Java**	
	Introduction to Java	
	What is Java, Java Platforms, History of Java, Advantages, Used in, etc.,?	
	Java vs C, Similarities between C and Java, Differences between C and Java	
	JDK, JRE, PATH and CLASSPATH	
	Java Basics	
	Compiling, Execution of a Java Program	
	Data Types	
	Int, float, char, double, Boolean, short, long, byte	
	Statements	

	Decision Making Statements, Loop Statements	
	Strings and Arrays	
	Object Oriented Programming	
	Classes and Objects	
	Flow of Control – Branching and Iteration	
	Encapsulation	
	Inheritance	
	Constructors	
	Different Types of Constructors	
	No Argument Constructors	
	Parameterized Constructors	
	Constructor Overloading	
	Explicit vs Implicit default Constructor	
	Wrapping Classes Accessibility Modes	
	Access Modifiers: Default, Public, Private and Protected	
	Method Overloading	
	Method Overriding	
	Abstract Classes and Methods	
	Polymorphism	
	Static & Dynamic Polymorphism	
	Static Binding & Dynamic Binding	
	Interfaces	
	Interface vs Abstract Classes	
	Role of Interfaces in Real Time Projects	
	Declarative Multiple Inheritance using Interfaces	

	String Handling	
	String, String Buffer and String Builder	
	Exception Handling	
	Usage of Exception Handling	
	Types of Errors	
	Try, Catch, Throw and Finally	
	User Defined Exceptions	
	IO Streams	
	Byte Oriented Streams	
	Character Oriented Streams	
	Java.io File Class	
	Object Serialization	
	Multithreading	
	Object, Attributes (Properties)	
	Operations (Methods)	
	Reference vs Object	
	Real-Time Object-Oriented Examples	
	Inter Object Communication	
	Static Variables, Static Methods, Static Blocks and Static Classes	
	Method Overloading	
	Inheritance	
	Multi-tasking and Multithreading	
	Life Cycle of a thread	
	Java.lang.Thread vs java.lang.Runnable	

	Multithreading Java Applications	
	Synchronization	
	Dead lock	
	Inter-thread Communication	
	Inner Classes	
	Nested Classes	
	Java.lang.Object Class	
	Garbage Collection	
	Java.util.Properties	
	Java.util.StrongTokenizer	
	Dealing with Date and Time	
	Collection Framework	
	List, Set, Map Based and Set Based Collections Interfaces and Classes	
	Cursors in Collections	
	Generics in Collections	
	Collection vs Collections	
	Java.util.Comparator vs java.util.Comparable	
	Annotations	
	Java 8 Features	
	Java Application Development	
	Using Eclipse IDE	
	Packages	
	JDBC	
	The Java Streams Model	

	Working with Files	
	Advanced Stream Techniques	
	Java Serialization	
3	**J2EE**	
	HTTP 1.1 Protocol, Request Response Life Cycle	
	What is J2EE Application, it's architecture and What is an Application Server?	
	Introduction to Servlets, their specifications and diverse types of Server-Side Frameworks	
	JDBC and connecting the Java Application with Database	
	ORM Frameworks available and basics of Hibernate	
4	**Spring, Spring Boot**	
	Spring Introduction	
	Spring Core Modules / Spring Features: IOC, AOP, Container	
	Introduction to IOC	
	Spring Architecture	
	Dependency Injection	
	Types of Dependency Injection	
	Setter Injection	
	Constructor Injection	
	Aware Injection	

	Method Injection	
	Lookup Method Injection	
	Introduction to Design Patterns	
	Factory Design Pattern	
	Strategy Design Pattern	
	Spring Core Modules with Annotations	
	Explicit Wiring	
	Auto Wiring	
	Types of Configuration	
	XML Based Configuration	
	Java Based Configuration	
	Annotation Based Configuration	
	Model Layer	
	Spring DAO	
	Template CallBack Design Pattern	
	Spring JDBC	
	Spring with Hibernate	
	Profiling and Environments	
	Service Layer	
	Programmatic Transactions	
	Declarative Transactions [Annotations]	

	Controller Layer	
	Spring MVC Architecture	
	Handler Mapping	
	View Resolver	
	Front Controller	
	Spring MVC Controller	
	Road Map	
	Annotation Based Controller	
	Validations	
	Multiple Config Files	
	Http Endpoints	
	Spring Boot	
	Starters	
	Auto Configuration	
	Embedded Servers	
	Actuators	
	Aspect Oriented Programming	
	Introduction	
	Advantages of OOP	
	AOP Terminology	
	Declarative Based AOP	

	Annotation Based AOP	
5	**Html, Css and Bootstrap**	
	Html	
	Introduction to Html	
	Html Elements, Attributes	
	Page Creation with Formatting Elements	
	Creating Links and Navigating, Redirecting Pages	
	Displaying the Images, Adjustment of Images along with Content	
	Organizing data with Tables	
	Displaying data in Listings	
	Creating Forms and Layouts with Fields and Input	
	Implementing the Semantic Elements (latest Html elements)	
	Working with Audio and Video (Media Objects)	
	Css	
	Introduction to Css	
	Implementing the Three Different ways of Styles: Inline, Embedded and External to Html Content	
	Applying the Web Colors to Html Content in Three Different ways: Direct Name, Numerical, and HexaDecimal	
	Applying different properties like: padding, margin, float, font, border, background, etc.,	

	Applying the Display and Position properties, etc.,	
	Creating the Web Page with Content	
	Creating the Grid Layout with Sample Content with Responsive	
	Working with Media Queries	
	Bootstrap	
	Introduction to Bootstrap, Usage, Advantages	
	Discussing about different types of Resolutions	
	Implementing the Bootstrap Required Links into Web Pages	
	Create the Fixed Width, Fluid Width Web Pages	
	Implementing the Components of Bootstrap:	
	Tables, Lists, Forms, Images,	
	Modals, Tabs, Cards, Navigations	
6	**Javascript, Ecmascript and Typescript**	
	Javascript	
	Introduction about Javascript, DOM etc?	
	Writing the Programs and Displaying the Data	
	Understanding about Keywords, Identifiers	
	Declarations: var, let, const	
	Implementing Statements: Decision Making Statements and Loop Statements	
	Working with Arrays and Implementing with Examples	

		Understanding the Concept of Functions: Call and Define, Functions with Parameters, Return Statement, Arrow Functions, Call Back Functions	
		Implementing the Object Oriented Programming in Javascript: What is Object, Class, Constructor, Method, Property etc., and its terminology	
		Creating the concept with Class and Object includes the Methods, Properties, and Accessing the Data	
		Proptotype, Closures concepts implementation	
		Promise, and Async and Await concepts implementation	
		Ecmascript	
		Ecmascript Introduction	
		Declarations of Identifiers with let & const	
		Working with Loops	
		Functions, Passing Parameters, Return Statement, Arrow Functions, Call Back Functions	
		Spread Operators	
		Map Object with Set and Get	
		Promise, Async and Await with Promise	
		Typescript	
		Typescript Introduction, Advantages of Typescript	
		Declaring Variables or Identifiers using let & const and accessing them	

	Working with Decision Making Statements	
	Working with Loop Statements	
	Working with Functions	
	Understanding and Implementing Tuple Operators	
	Creating Interfaces, Accessing with Objects	
	Creating Class's, Interfaces, and Declaring Objects	
	Implementing Inheritance	
	Creating Custom Modules	
7	**Angular**	
	Introduction to Angular, Features, Advantages, Single Page Application	
	Setting up the Env., and Creating the Angular Project	
	Understanding the Angular Project Flow, Precising the Folder Structure	
	Creating the Components through Command or Manual and Connecting to Project	
	Adding additional Dependencies through NPM, Installing or Updating or Deleting the Packages or Dependencies according to Project Requirements	
	Creating the Routing Concept with Group of Components	
	Working with Property Binding, Event Binding	
	Property Binding with One Way & Two Way	
	Implementing the Two Way Property Binding with Forms	
	Creating the Forms with Template Driven and	

	Implementing Form Validations with Regular Expressions using Pattern	
	Creating the Forms with Reactive Forms and Implementing Form Validation with Regular Expressions using Pattern	
	Creating the Form with File Upload with Validation	
	Creating the Form with Adding Dynamic Elements on Fly Based on the User Actions	
	Understanding the concept of Services, Creating the Services, and accessing the Services into Components	
	Implementing the concept of Observables	
	Implementing the concept of Dependency Injection	
	Understanding the concepts of @Input() and @Output() Decorators and Implementation	
	Understanding and Implementing the concept of Directives	
	Working with Built-in Directives, Creating the Custom Directives	
	Understanding and Implementing the concept of Pipes	
	Working with Built-in Pipes and Creating the Custom Pipes	
	Implementing the CRUD Operations	
	Implementing the concept of Lazy Loading	
	Implementing the concept of Life Cycle Hooks	
	Creating the Custom Modules and Implementing into the Components	

	Creating the Single Page Application	
8	**ReactJS**	
	ReactJS Introduction	
	Scope of ReactJS	
	What is ReactJS?	
	Understanding the Terminology of ReactJS	
	What is Single Page Application?	
	Understanding the ReactJS Architecture	
	Setting up the ReactJS Environment	
	Creating the First Project in ReactJS	
	Understanding the Project Folder & Files Structure	
	Working with assets - Html, Css, Images, Javascripts in ReactJS	
	Understanding the Term: Component.	
	What is Component?	
	Creating the new Component(s)	
	Accessing the Created Component(s)	
	Working with Re-Usable Component(s)	
	Understanding the Terms: 'props' and 'state'	
	Working with 'props'	
	Implementing the 'props' into Component(s)	
	Creating the 'state' object, setting the 'properties' into it	
	Accessing the 'properties' from 'state' object into Components	

	Working with Event Handlers ex., onClick, onHover, onBlur, onFocus, onChange etc.,	
	Creating the methods into Component(s)	
	Writing the definition(s) of the methods	
	Passing 'parameters' into methods	
	Understanding the concept of Property Binding	
	Two types of Property Binding., 1. One Way Property Binding and 2. Two Way Property Binding	
	Asking the User to input the value, accessing those value(s)	
	Creating the group of Component(s) and accessing those through 'state' object	
	Creating the CRUD operations	
	Understanding the concept of Routing	
	Creating the group of Component(s) and aligning them into Routing	
	Fetching the Data from RESTAPI into Component(s) and accessing the Data	
	Understanding and Working with Material UI (ReactJS UI Framework)	
	Creating the Button, Radio, Form, Table etc.,	
	Working with Form, Two – Way Property Binding	
	Creating the Validation with Error Hints,	
	Applying the Regular Expressions to Input Fields	
	Working with the concept of Observables	

	Working with the concept of Services	
	Implementing the concept of Observables & Subscribe with Services	
	Implementing the concept of Observables & Subscribe with Services using RESTAPI	
	Understanding the Terms: babel, webpack	
	Understanding the concept of Redux	
	Understanding the Terms: Component, Action, Reducer, Data Store	
	Understanding and Implementing the concept of Redux	
	Understanding the Term: Sage	
	Implementing the functionality of Redux with Saga	
	Understanding the Terms: Life Cycle Hooks	
	What is the usage of Life Cycle Hooks	
	How many types of Life Cycle Hooks: 1. Class Based Life Cycle Hooks and 2. Functional Based Life Cycle Hooks	
	Understanding the Class Based Life Cycle Hooks and its methods: componentDidMount(), componentDidUpdate(), componentWillUnmount() etc.,	
	Understanding the Functional Based Life Cycly Hooks and its methods: useState(), useEffect(), useContext(), etc.,	
	Understanding the concept of Styled Components	
	Working with ReactJS Developer Tools	
	Implementing the concept of ContextAPI	
	Understand and Implementing the Unit Tests in ReactJS	

	Creating the Single Page Application in ReactJS	
9	**Oracle SQL**	
	Introduction to Oracle Database	
	Retrieving Data using SQL SELECT Command Statement	
	Restrict and Sort Data	
	Usage of Single – Row Functions to Customize Output	
	Invoke Conversation Functions and Conditional Expressions	
	Aggregate Data using Group Functions	
	Display Data from Multiple Tables using Joins	
	Use Sub-Queries to solve Queries	
	The Set Operators	
	Data Manipulation Statements	
	Use of DDL Statements to Create and Manage Tables	
	Schema Objects	
	Management of Schema Objects	
	Date and Time Operations	
	Retrieve Data using Sub – Queries	
	Built In Functions	
10	**DevOps**	
	DevOps Process	
	What is DevOps in Cloud	
	Code Pipeline	

	Code Commit, Code Build	
	Code Deploy	
11	**Project Development**	
	Capstone Project	
	Capstone Project will stitch all the concepts learnt during the program	
	E-Commerce	
	Create a dynamic and responsive Java e-commerce web application using technologies such as Angular/ReactJs, Hibernate, MySQL, and Docker.	
	Food Delivery	
	Build a food-delivery app from scratch using your knowledge of the Spring framework, web services, and MySQL with a strong back end to support operations.	
	Entertainment	
	Create an entertainment application like BookMyShow using your back-end knowledge, API development, and HTML and CSS skills.	
	HealthCare	
	Build a dynamic and effective healthcare app. Create rich UI for effective diagnosis and health advice with Angular, Javascript ES6, and more	