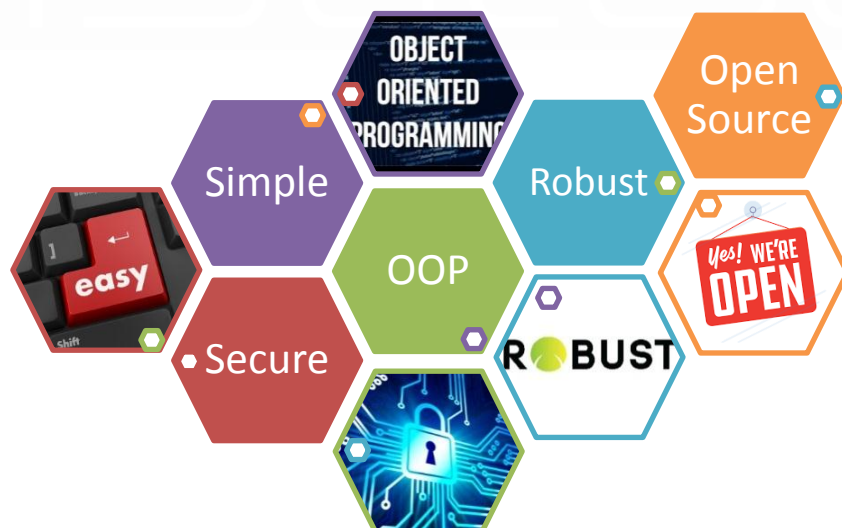


# sterling institute

The finest institute of programming & technology

## INTERNSHIP CONTENT FOR JAVA FULL STACK

Duration: 4 Months (6 Hours / Day)



“WE BELIEVE IN PROVIDING QUALITY NOT ONLY CERTIFICATION”

**STERLING INSTITUTE**

G-5, 150 New Azad Nagar Behind Akashwani, Daly College Road Indore – 452001

Contact: +91 9109913455

[www.sterlinginstitute.in](http://www.sterlinginstitute.in)

## Training Content

Core Java	Advance Java	Angular JS
<ol style="list-style-type: none"> <li>1. Basics of Programming Languages</li> <li>2. Genesis of Java</li> <li>3. Java Basics</li> <li>4. Environment Configuration</li> <li>5. Variables &amp; Operators</li> <li>6. OOP</li> <li>7. JVM Architecture</li> <li>8. Exception Handling</li> <li>9. Reflection API</li> <li>10. Collection Framework</li> <li>11. I/O</li> <li>12. Multithreading</li> </ol>	<ol style="list-style-type: none"> <li>1. Client Server Architecture</li> <li>2. HttpServlet</li> <li>3. Apache Tomcat</li> <li>4. Servlet Config-Context</li> <li>5. JSP</li> <li>6. JSP Translation</li> <li>7. Scripting Elements</li> <li>8. Directive Elements</li> <li>9. Action Elements</li> <li>10. Cookies</li> <li>11. Session Tracking Mechanism</li> <li>12. Implicit Objects</li> <li>13. Filters</li> </ol>	<ol style="list-style-type: none"> <li>1. Introduction to JavaScript</li> <li>2. Single Page Application</li> <li>3. Create an Application using Angular CLI</li> <li>4. Workflow</li> <li>5. Modular Concept</li> <li>6. Application Structure</li> <li>7. Application Module</li> <li>8. Component</li> <li>9. Directives</li> <li>10. EventListeners</li> <li>11. Data Binding</li> <li>12. String Interpolation</li> <li>13. Binding</li> <li>14. life cycle hook</li> <li>15. View Child and children</li> <li>16. Routing</li> <li>17. Pipes</li> <li>18. Built-in Pipes</li> <li>19. Custom PipesAngular – Modules</li> <li>20. Lazy Loading</li> <li>21. Services</li> <li>22. Gaurds</li> <li>23. HTTP Service</li> <li>24. Forms</li> <li>25. Reactive Forms</li> <li>26. Template-driven Forms</li> <li>27. Angular Material</li> <li>28. Angular Testing</li> </ol>
Design Patterns	Mailing	
<ol style="list-style-type: none"> <li>1. Bridge</li> <li>2. Factory</li> <li>3. Adapter</li> <li>4. Singleton</li> <li>5. Decorator</li> </ol>	<ol style="list-style-type: none"> <li>1. E-mail API</li> <li>2. (javax.mail)</li> <li>3. Gmail SMTP Server</li> <li>4. Gmail POP Server</li> <li>5. Gmail MIME</li> <li>6. Gmail IMAP</li> <li>7. Sending Email</li> <li>8. Receiving Email</li> <li>9. Sending Attachment</li> </ol>	
Tools	Servers	
<ol style="list-style-type: none"> <li>1. Log4J</li> <li>2. JUnit</li> <li>3. Maven</li> <li>4. GitHub</li> </ol>	<ol style="list-style-type: none"> <li>1. JBoss</li> <li>2. Apache Tomcat</li> </ol>	
Hibernate	Spring Core	Spring MVC
<ol style="list-style-type: none"> <li>1. Introduction to ORM Tools</li> <li>2. Hibernate Architecture</li> <li>3. Environment</li> <li>4. Configuration</li> <li>5. Sessions</li> <li>6. Persistent Class</li> <li>7. Mapping Files</li> <li>8. Mapping Types</li> <li>9. Hibernate with Log4j</li> </ol>	<ol style="list-style-type: none"> <li>1. Spring-Architecture</li> <li>2. Environment Setup</li> <li>3. First Spring Application</li> <li>4. IoC Container</li> <li>5. Bean Definition</li> <li>6. Bean Scopes</li> <li>7. Bean Life Cycle</li> <li>8. Bean Post Processors</li> <li>9. Bean Definition Inheritance</li> </ol>	<ol style="list-style-type: none"> <li>1. Introduction to Spring MVC</li> <li>2. Form Handling               <ol style="list-style-type: none"> <li>a. Page Redirection</li> <li>b. Static Pages</li> <li>c. Form Tag Library</li> <li>d. TextBox</li> <li>e. Password</li> <li>f. Textarea</li> <li>g. Checkboxes</li> </ol> </li> </ol>

<ul style="list-style-type: none"> <li>10. O/R Mappings</li> <li>11. Annotations</li> <li>12. Hibernate Query Language (HQL)</li> <li>13. Criteria Queries</li> <li>14. Native SQL</li> <li>15. Hibernate Caching</li> <li>16. Batch Processing</li> <li>17. Interceptors</li> </ul>	<ul style="list-style-type: none"> <li>10. Dependency Injection</li> <li>11. Constructor Injection</li> <li>12. CI Dependent Object</li> <li>13. CI with Collection</li> <li>14. CI with Map</li> <li>15. Injecting Inner Beans</li> <li>16. Injecting Collection</li> <li>17. Beans Auto-Wiring</li> <li>18. Annotation Based Configuration</li> <li>19. Java Based Configuration</li> <li>20. Event Handling in Spring</li> <li>21. Custom Events in Spring</li> <li>22. AOP with Spring Framework</li> <li>23. JDBC Framework</li> <li>24. Transaction Management</li> <li>25. Web MVC Framework</li> <li>26. Logging with Log4J</li> </ul>	<ul style="list-style-type: none"> <li>h. Radiobutton</li> <li>i. Dropdown</li> <li>j. Listbox</li> <li>k. Hidden</li> <li>l. Errors</li> <li>m. Upload</li> <li>3. Handler Mapping               <ul style="list-style-type: none"> <li>a. Bean Name URL Handler Mapping</li> <li>b. Controller Class Name Handler Mapping</li> <li>c. Simple URL Handler Mapping</li> </ul> </li> <li>4. Controller               <ul style="list-style-type: none"> <li>a. Multi Action Controller</li> <li>b. Properties Method Name Resolver</li> <li>c. Parameter Method Name Resolver</li> <li>d. Parameterized View Controller</li> </ul> </li> <li>5. View Resolver               <ul style="list-style-type: none"> <li>a. Internal Resource View Resolver</li> <li>b. XML View Resolver</li> <li>c. Resource Bundle View Resolver</li> <li>d. Multiple Resolver Mapping</li> </ul> </li> <li>6. Integration               <ul style="list-style-type: none"> <li>a. Hibernate Validator</li> <li>b. Generate RSS Feed</li> <li>c. Generate XML</li> <li>d. Generate JSON</li> <li>e. Generate Excel</li> <li>f. Generate PDF</li> <li>g. Using log4j</li> </ul> </li> </ul>
<b>RESTful Web Services</b>	<b>Client's Project</b>	
<ul style="list-style-type: none"> <li>1. Resources</li> <li>2. Messages</li> <li>3. Addressing</li> <li>4. Methods</li> <li>5. Statelessness</li> <li>6. Caching</li> <li>7. Security</li> <li>8. Java (JAX-RS)</li> </ul>	<ul style="list-style-type: none"> <li>1. Project will be based on client's Requirement</li> </ul>	