

## **Advance Java (JDBC, Swing, Servlets, JSP, JSF and Networking).**

### **JDBC 4.0**

Introduction to JDBC. JDBC architecture. java.sql Package. Connection, Statement, ResultSet. Prepared Statement. Callable Statement. Scrollable and Updatable ResultSet. Batch Updates. ResultSetMetaData. Simple Transaction Management. Four Levels of JDBC drivers, their pros & cons. Features of JDBC 3.0

### **SWING**

Basic swing programs. Component communication & understanding events. Controls and Layouts. Forms and Lists. Option Controls & Making Choices. Menus. Using Common Dialogs

### **SERVLETS**

Need of server side programming. Introduction to Servlets. Servlet Life Cycle. javax.servlet package. ServletConfig, ServletContext, ServletResponse. Supplying initialization parameters to Servlets. Performing database operations in Servlets. Include and forward mechanisms. Applying filters to Servlets. javax.servlet.http Package. HttpServlet Life Cycle. Http request methods GET vs POST. HttpServletRequest, HttpServletResponse. Dealing with Http headers & error codes. Session Tracking, purpose .Hidden form fields, Cookies .Http Session, URL rewriting. Event listeners. Web application security.

### **JSP**

Disadvantages of Servlets. Introduction to JSP. JSP Life Cycle. Creating dynamic Web content with JSP. Scripting elements .Scriptlet. Declaration. Expression. XML syntax for JSP elements .JSP directives page, include and taglib . JSP implicit objects. JSP scopes. Include and forward mechanism. Using a Java bean in a jsp. JSP Model 1 architecture. JSP Model 2 (MVC) architecture. Custom Tag Development. Classic Tags, Simple Tags. Error Handling in a jsp. JSTL. Expression Language .Processing XML in a JSP.

## JSF

Introducing JavaServer Faces (JSF) Technology. List common requirements for web applications. Describe the architecture of JSF web applications. Describe the development view of a JSF application. Walk through a simple JSF web application. Review the life cycle of a JSF application. Create, deploy, and run a simple JSF application. Creating JSF Pages Using Facelets. Describe the hierarchy of UI components. Evaluate the structure of JSF pages. List the tag libraries supported in Facelets . Describe the HTML render kit tag library. List common attributes of the HTML tags. Describe the JSF core tag library. Use common UI components to design Facelets pages. Developing CDI Named Beans. Define a managed bean. Use the JSR-299: Context and Dependency Injection (CDI) annotations. Bind UI components with CDI beans. Use the unified Expression Language (EL). Use the faces-config.xml configuration file. Use CDI bean scopes. Working with Navigation. Use static and dynamic navigation. Define implicit navigation in JSF pages. Configure navigation rules and cases. Describe the navigation evaluation process. Create a bookmarkable view. Creating and Adding Message Bundles. Create a message bundle for multiple languages. Use a message bundle to simplify localization. Localize an application. Using JSF Templates. Create a template and apply it to multiple pages. Describe how to use a decorator. Use debugging to identify issues. Converting and Validating Data. Describe the data conversion and validation process. Use standard data converters and validators. Configure default validators. Develop and use custom converters and validators. Work with data conversion and validation error messages. Use Bean Validation (JSR-303).Working with Data Tables. Use a Data Table component. Use column headers, footers and captions. Apply styles to Data Table elements. Customize a data table with a scroll bar. Enhance a data table with a sort table class. Add a pager component to the table. Handling Events. Describe the JSF Event Model. Use action and value Change Events. Register event listeners. Capture and respond to lifecycle events.Using AJAX and Composite Components with JSF.

## **Java Framework with (Hibernate, Spring, Struts and EJB).**

### **HIBERNATE.**

Issues with Persistence layers and Object-Relational Mapping (ORM). Hibernate Overview and Benefits. Hibernate architecture overview. POJO (Plain Old Java Object) Based Mapping. Overview of the Hibernate distribution. Configuring Hibernate. hibernate.cfg.xml file. SessionFactory configuration. Connection properties, Database dialect. Configuration class, Session Interface. "Hello World" Program for Hibernate. Mapping a Class. Persistent Entity Class, Hibernate Mapping .File, Mapping the Entity Class .Primary keys: Id property, Generated Id. Hibernate Type System. Working with sessions and Persistent Objects.Logging - log4j Overview and configuration for Hibernate. Inserting and Updating Entities. HQL - Hibernate Query Language Overview. The Query Interface. Creating and working with queries. Named Queries, Projection Queries, Aggregate Queries. Transaction Overview and Transactions in Hibernate. Hibernate Transaction API (in Managed and Non-managed Environments). The lifecycle of managed objects. Persistent, transient, and detached objects. The Persistence (Session) Context (Lifespan, Relation to Managed Objects, Propagation). Contextual Sessions. Synchronization to the Database. The Session as cache. Detached Objects and Optimistic .Locking. Versioning overview and Using Versioning. Locking Objects. Object Relationship Overview. Mapping Collections of Value Objects. Entity Relationships: 1-N, N-1, N-N, 1-1. Mapping Entity Relationships. Uni and Bi-directional Relationships. The Relationship "inverse". Cascading Over Relationships. Queries Across Relationships (Lazy and Eager). Entity Inheritance with Hibernate. Table-per-class mapping. Table per Subclass mapping. Table per Concrete Class mapping. Projection Queries, Aggregate queries. Bulk updates and deletes. Native SQL Queries. Query Filters. Overview of the Criteria API. Working Querying with the Criteria API. Query by Example

### **SPRING**

Introduction to Spring Framework. Inversion of Control and Dependency Injection. IOC Container. Bean Creation. Construction Injection. Setter Injection. Auto Wiring. Bean Initialization and Destruction. Aware interfaces: BeanNameAware, BeanFactoryAware. Lookup Method Injection. Use of ApplicationContext. ResourceBundleMessageResource. Events and Event Handling. Aspect Oriented Programming (AOP). Introduction to AOP. Implementation of JDK dynamic proxy. ProxyFactoryBean and Advices. Applying Advices: Before, After, Around and AfterThrowing. Using Point Cut and Advisor for fine grain control of advices. Spring Web MVC. MVC Architecture. Front Controller and DispatcherServlet. Spring Controller. Context . Configuration. View Resolution. InternalResourceViewResolver. Stereotype Controllers. Request mapping by . annotations. Path Variables. Handler method parameters. Handler method return types. Command beans and working with forms. Binding results and errors. Other Controller

annotations. Other View resolvers. Exception Handling. JDBCTemplate. Introduction to Template Design Pattern. Applicability of Template in JDBC. JDBCTemplate class. DataSource Configuration. Dao Support. Transactions. Transaction Manager. IDeclarative Transactions. Transaction attributes. Transaction Annotations. Transaction Template

## **STRUTS**

Difference between Struts 1.x VS Struts 2.x. Struts 2 Architecture. Hello World Struts 2 example. brief comparison with other web application framework. About Struts 2 actions. Implementing actions. About Model driven actions. Interceptors in actions. Declaring Interceptors. Building your own interceptors. OGNL and struts 2. Data transfer and type conversion. Built-in type converters. Struts 2 tag API. Using OGNL to set attributes on tags. Data Tags. Control Tags. UI component tags. Results in detail. Life after the action .Commonly used result types .Exploring the Struts validation framework. Understanding Struts 2 internationalization.

## **EJB:**

Developing Applications for the Java EE 6 Platform. Survey of Java EE Technologies. Enterprise Application Architecture. Web Technology Overview. Developing Servlets. Developing With JavaServer Pages Technology . JavaServer Faces. EJB Overview. Implementing EJB 3.0 Session Beans. The Java Persistence API. Implementing a Transaction Policy. Developing Asynchronous Java EE Applications and Messaging. Developing Message-Driven Beans. Web Service Model Implementing Java EE Web Services with JAX-WS and JAX-RS. Implementing a Security Policy