

CLONESKILLS, INC.

Prepare To Lead.
Lead With Skills.

SUN J2EE Web Component Developer (SCWCD) Certification Preparation Training



Prepare to lead.
Lead with skills.

CLONESKILLS

The Global Training Company for Cultivating IT Leadership

CLONESKILLS, Inc.

5348 Vegas Drive, Suite 733

Las Vegas, Nevada, 89108

Main : 1.800.836.5696

Email: info@cloneskills.com

W
E

B
U
I
L
D

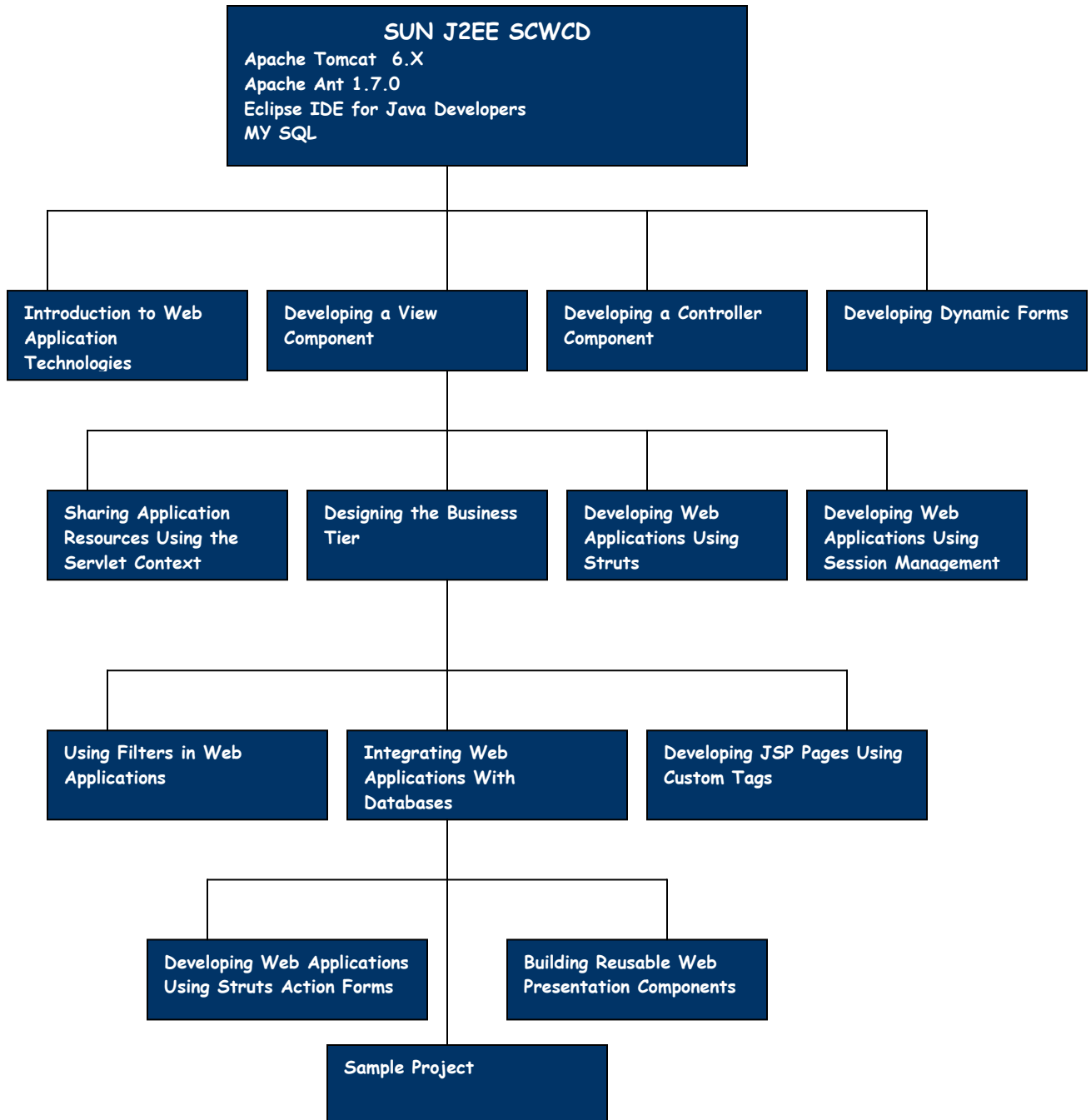
C
A
R
E
E
R

1

We differentiate our training from others by delivering Role-Based J2EE instruction rather than J2EE Product focused Training.

<http://www.CloneSkills.com> , JothiP@CloneSkills.com , (916)-296-0228
THE Global Training Company for Cultivating IT Leadership

Training Roadmap



We differentiate our training from others by delivering Role-Based J2EE instruction rather than J2EE Product focused Training.

SUN J2EE Web Component Developer (SCWCD) Certification Preparation Training

SUN J2EE SCWCD Certification Preparation Training Content and Syllabus		
Session #	Training Syllabus	Training Content
1	Introduction to Web Application Technologies	<ul style="list-style-type: none"> ▪ Describe web applications ▪ Describe Java Platform, Enterprise Edition 5 (Java EE 5) ▪ Describe Java servlet technology ▪ Describe JavaServer Pages technology ▪ Define three-tier architecture ▪ Define Model-View-Controller (MVC) architecture
QUIZ # 1		
2	Developing a View Component	<ul style="list-style-type: none"> ▪ Design a view component ▪ Describe the Hypertext Transfer Protocol ▪ Describe the web container behavior ▪ Develop a simple HTTP servlet ▪ Configure and deploy a servlet
QUIZ # 2		
3	Developing a Controller Component	<ul style="list-style-type: none"> ▪ Design a controller component ▪ Create an HTML form ▪ Describe how HTML form data is sent in an HTTP request ▪ Develop a controller servlet ▪ Dispatch from a controller servlet to a view servlet
QUIZ # 3		
4	Developing Dynamic Forms	<ul style="list-style-type: none"> ▪ Describe the servlet life cycle ▪ Customize a servlet with initialization parameters ▪ Explain error reporting within the web form ▪ Repopulating the web form
QUIZ # 4		
5	Sharing Application Resources Using the Servlet Context	<ul style="list-style-type: none"> ▪ Describe the purpose and features of the servlet context ▪ Develop a servlet context listener to initialize a shared application resource
QUIZ # 5		
6	Designing the Business Tier	<ul style="list-style-type: none"> ▪ Describe the Analysis model ▪ Design entity components ▪ Design service components
QUIZ # 6		

We differentiate our training from others by delivering Role-Based J2EE instruction rather than J2EE Product focused Training.

SUN J2EE Web Component Developer (SCWCD) Certification Preparation Training

7	Developing Web Applications Using Struts	<ul style="list-style-type: none"> ▪ Design a web application using the Struts MVC framework ▪ Develop a Struts action class ▪ Configure the Struts action mappings
QUIZ # 7		
8	Developing Web Applications Using Session Management	<ul style="list-style-type: none"> ▪ Design a web application that uses session management ▪ Develop servlets using session management ▪ Describe the cookies implementation of session management ▪ Describe the URL-rewriting implementation of session management
QUIZ # 8		
9	Using Filters in Web Applications	<ul style="list-style-type: none"> ▪ Describe the web container request cycle ▪ Describe the Filter API ▪ Develop a filter class ▪ Configure a filter in the web.xml file
QUIZ # 9		
10	Integrating Web Applications With Databases	<ul style="list-style-type: none"> ▪ Web container request cycle ▪ Describe the Filter API ▪ Develop a filter class
QUIZ # 10		
11	Developing JSP Pages	<ul style="list-style-type: none"> ▪ Describe the Java EE job roles involved in web application development ▪ Design a web application using custom tags ▪ Use JSTL tags in a JSP page
QUIZ # 11		
12	Developing JSP Pages Using Custom Tags	<ul style="list-style-type: none"> ▪ Describe the components in a Struts application ▪ Develop an ActionForm class ▪ Develop a JSP page for a View form ▪ Configure the View forms
QUIZ # 12		
13	Developing Web Applications Using Struts Action Forms	<ul style="list-style-type: none"> ▪ Describe the components in a Struts application ▪ Develop an ActionForm class ▪ Develop a JSP page for a View form ▪ Configure the View forms
QUIZ # 13		
14	Building Reusable Web Presentation Components	<ul style="list-style-type: none"> ▪ Describe how to build web page layouts from reusable presentation components ▪ Include JSP segments ▪ Develop layouts using the Struts Tiles framework
QUIZ # 14		
15	Case Study and Sample Project	Applicable ONLY for Qualified Participants

We differentiate our training from others by delivering Role-Based J2EE instruction rather than J2EE Product focused Training.

APPENDIX - A

Sun Certified Web Component Developer (SCWCD) Overview:

- This certification is for developers specializing in the application of JavaServer Pages and servlet technologies used to present Web services and dynamic Web content.
- To achieve this certification, candidates must successfully complete one exam. Prior to attempting certification, candidates must be certified as a Sun Certified Programmer (SCJP), any edition.

JavaServer Pages (JSP) and servlets are the key web tier technologies defined in the Java 2 Platform, Enterprise Edition (J2EE platform). The Web Component Development With Servlet and JSP Technologies course provides experienced Java developers a way to obtain the knowledge and skills necessary to quickly build web applications from JSP and servlet technologies using the Tomcat web container and the Struts framework. Students are exposed to the current methods for analyzing, designing, developing, and deploying web applications with Java technologies. Real-world lab exercises provide students with experience in constructing and deploying the small- to medium-scale web applications found in intranet and low-volume commercial sites.

This course also is an ideal method for preparing for the revised Sun Certified Web Component Developer (SCWCD) for J2EE 1.4 certification examination. However, an SCWCD candidate should also spend six months building real-world web applications using the JSP and servlet technologies before taking the exam.

Software Components Used in this Training:

- Apache Tomcat 6.X
- Apache Ant 1.7.0
- Eclipse IDE for Java Developers
- MY SQL

Exam Objectives

The Servlet Technology Model

- ✓ For each of the HTTP Methods (such as GET, POST, HEAD, and so on) describe the purpose of the method and the technical characteristics of the HTTP Method protocol, list triggers that might cause a Client (usually a Web browser) to use the method; and identify the HttpServlet method that corresponds to the HTTP Method.
- ✓ Using the HttpServletRequest interface, write code to retrieve HTML form parameters from the request, retrieve HTTP request header information, or retrieve cookies from the request.
- ✓ Using the HttpServletResponse interface, write code to set an HTTP response header, set the content type of the response, acquire a text stream for the response, acquire a binary stream for the response, redirect an HTTP request to another URL, or add cookies to the response.
- ✓ Describe the purpose and event sequence of the servlet life cycle: (1) servlet class loading, (2) servlet instantiation, (3) call the init method, (4) call the service method, and (5) call destroy method.

The Structure and Deployment of Web Applications

- ✓ Construct the file and directory structure of a Web Application that may contain (a) static content, (b) JSP pages, (c) servlet classes, (d) the deployment descriptor, (e) tag libraries, (d) JAR files, and (e) Java class files; and describe how to protect resource files from HTTP access.
- ✓ Describe the purpose and semantics of the deployment descriptor.
- ✓ Construct the correct structure of the deployment descriptor.
- ✓ Explain the purpose of a WAR file and describe the contents of a WAR file, how one may be constructed.

SUN J2EE Web Component Developer (SCWCD) Certification Preparation Training

The Web Container Model

- ✓ For the ServletContext initialization parameters: write servlet code to access initialization parameters; and create the deployment descriptor elements for declaring initialization parameters.
- ✓ For the fundamental servlet attribute scopes (request, session, and context): write servlet code to add, retrieve, and remove attributes; given a usage scenario, identify the proper scope for an attribute; and identify multi-threading issues associated with each scope.
- ✓ Describe the Web container request processing model; write and configure a filter; create a request or response wrapper; and given a design problem, describe how to apply a filter or a wrapper.
- ✓ Describe the Web container life cycle event model for requests, sessions, and web applications; create and configure listener classes for each scope life cycle; create and configure scope attribute listener classes; and given a scenario, identify the proper attribute listener to use.
- ✓ Describe the RequestDispatcher mechanism; write servlet code to create a request dispatcher; write servlet code to forward or include the target resource; and identify and describe the additional request-scoped attributes provided by the container to the target resource.

Session Management

- ✓ Write servlet code to store objects into a session object and retrieve objects from a session object.
- ✓ Given a scenario describe the APIs used to access the session object, explain when the session object was created, and describe the mechanisms used to destroy the session object, and when it was destroyed.
- ✓ Using session listeners, write code to respond to an event when an object is added to a session, and write code to respond to an event when a session object migrates from one VM to another.
- ✓ Given a scenario, describe which session management mechanism the Web container could employ, how cookies might be used to manage sessions, how URL rewriting might be used to manage sessions, and write servlet code to perform URL rewriting.

SUN J2EE Web Component Developer (SCWCD) Certification Preparation Training

Web Application Security

- ✓ Based on the servlet specification, compare and contrast the following security mechanisms: (a) authentication, (b) authorization, (c) data integrity, and (d) confidentiality.
- ✓ In the deployment descriptor, declare a security constraint, a Web resource, the transport guarantee, the login configuration, and a security role.
- ✓ Compare and contrast the authentication types (BASIC, DIGEST, FORM, and CLIENT-CERT); describe how the type works; and given a scenario, select an appropriate type.

The JavaServer Pages (JSP) Technology Model

- ✓ Identify, describe, or write the JSP code for the following elements: (a) template text, (b) scripting elements (comments, directives, declarations, scriptlets, and expressions), (c) standard and custom actions, and (d) expression language elements.
- ✓ Write JSP code that uses the directives: (a) 'page' (with attributes 'import', 'session', 'contentType', and 'isELIgnored'), (b) 'include', and (c) 'taglib'.
- ✓ Write a JSP Document (XML-based document) that uses the correct syntax.
- ✓ Describe the purpose and event sequence of the JSP page life cycle: (1) JSP page translation, (2) JSP page compilation, (3) load class, (4) create instance, (5) call the jspInit method, (6) call the _jspService method, and (7) call the jspDestroy method.
- ✓ Given a design goal, write JSP code using the appropriate implicit objects: (a) request, (b) response, (c) out, (d) session, (e) config, (f) application, (g) page, (h) pageContext, and (i) exception.
- ✓ Configure the deployment descriptor to declare one or more tag libraries, deactivate the evaluation language, and deactivate the scripting language. 6.7 Given a specific design goal for including a JSP segment in another page, write the JSP code that uses the most appropriate inclusion mechanism (the include directive or the jsp:include standard action).

SUN J2EE Web Component Developer (SCWCD) Certification Preparation Training

Building JSP Pages Using the Expression Language (EL)

- ✓ Given a scenario, write EL code that accesses the following implicit variables including `pageScope`, `requestScope`, `sessionScope`, and `applicationScope`, `param` and `paramValues`, `header` and `headerValues`, `cookie`, `initParam` and `pageContext`.
- ✓ Given a scenario, write EL code that uses the following operators: property access (the `.` operator), collection access (the `[]` operator).
- ✓ Given a scenario, write EL code that uses the following operators: arithmetic operators, relational operators, and logical operators.
- ✓ Given a scenario, write EL code that uses a function; write code for an EL function; and configure the EL function in a tag library descriptor.

Building JSP Pages Using Standard Actions

- ✓ Given a design goal, create a code snippet using the following standard actions: `jsp:useBean` (with attributes: `'id'`, `'scope'`, `'type'`, and `'class'`), `jsp:getProperty`, and `jsp:setProperty` (with all attribute combinations).
- ✓ Given a design goal, create a code snippet using the following standard actions: `jsp:include`, `jsp:forward`, and `jsp:param`.

Building JSP Pages Using Tag Libraries

- ✓ For a custom tag library or a library of Tag Files, create the `'taglib'` directive for a JSP page.
- ✓ Given a design goal, create the custom tag structure in a JSP page to support that goal.
- ✓ Given a design goal, use an appropriate JSP Standard Tag Library (JSTL v1.1) tag from the "core" tag library.

SUN J2EE Web Component Developer (SCWCD) Certification Preparation Training

Building a Custom Tag Library

- ✓ Describe the semantics of the "Classic" custom tag event model when each event method (doStartTag, doAfterBody, and doEndTag) is executed, and explain what the return value for each event method means; and write a tag handler class.
- ✓ Using the PageContext API, write tag handler code to access the JSP implicit variables and access web application attributes.
- ✓ Given a scenario, write tag handler code to access the parent tag and an arbitrary tag ancestor.
- ✓ Describe the semantics of the "Simple" custom tag event model when the event method (doTag) is executed; write a tag handler class; and explain the constraints on the JSP content within the tag.
- ✓ Describe the semantics of the Tag File model; describe the web application structure for tag files; write a tag file; and explain the constraints on the JSP content in the body of the tag.

J2EE Patterns

- ✓ Given a scenario description with a list of issues, select a pattern that would solve the issues. The list of patterns you must know are: Intercepting Filter, Model-View-Controller, Front Controller, Service Locator, Business Delegate, and Transfer Object.
- ✓ Match design patterns with statements describing potential benefits that accrue from the use of the pattern, for any of the following patterns: Intercepting Filter, Model-View-Controller, Front Controller, Service Locator, Business Delegate, and Transfer Object.

Disclaimer:

CloneSkills is an independent training and Solutions Company, we provide J2EE learning solutions, we are not affiliated to Sun Microsystems. At CloneSkills we exclusively promote the J2EE Business Solutions, and do not compete with either Sun Microsystems.

We differentiate our training from others by delivering Role-Based J2EE instruction rather than J2EE Product focused Training.