Globalcode®
The Developers Company

Scaling developer's knowledge

Especializada em treinamentos Java e J2EE

- **Mais de 850 alunos treinados**
- **Mais de 4.000 em palestras e mini-cursos**
- **Instrutores certificados**
- **Em Florianópolis, na V.Office – f. (48) 224-8580**

www.voffice.com.br **0800-704-5282** www.globalcode.com.br

Globalcode®
The Developers Company

Open-source Education

Dicas e revisões técnicas para certificação SCWCD SCE 310-80

Iniciativa Globalcode

Globalcode®
The Developers Company

Palestrante

Ricardo Jun

ricardo@globalcode.com.br

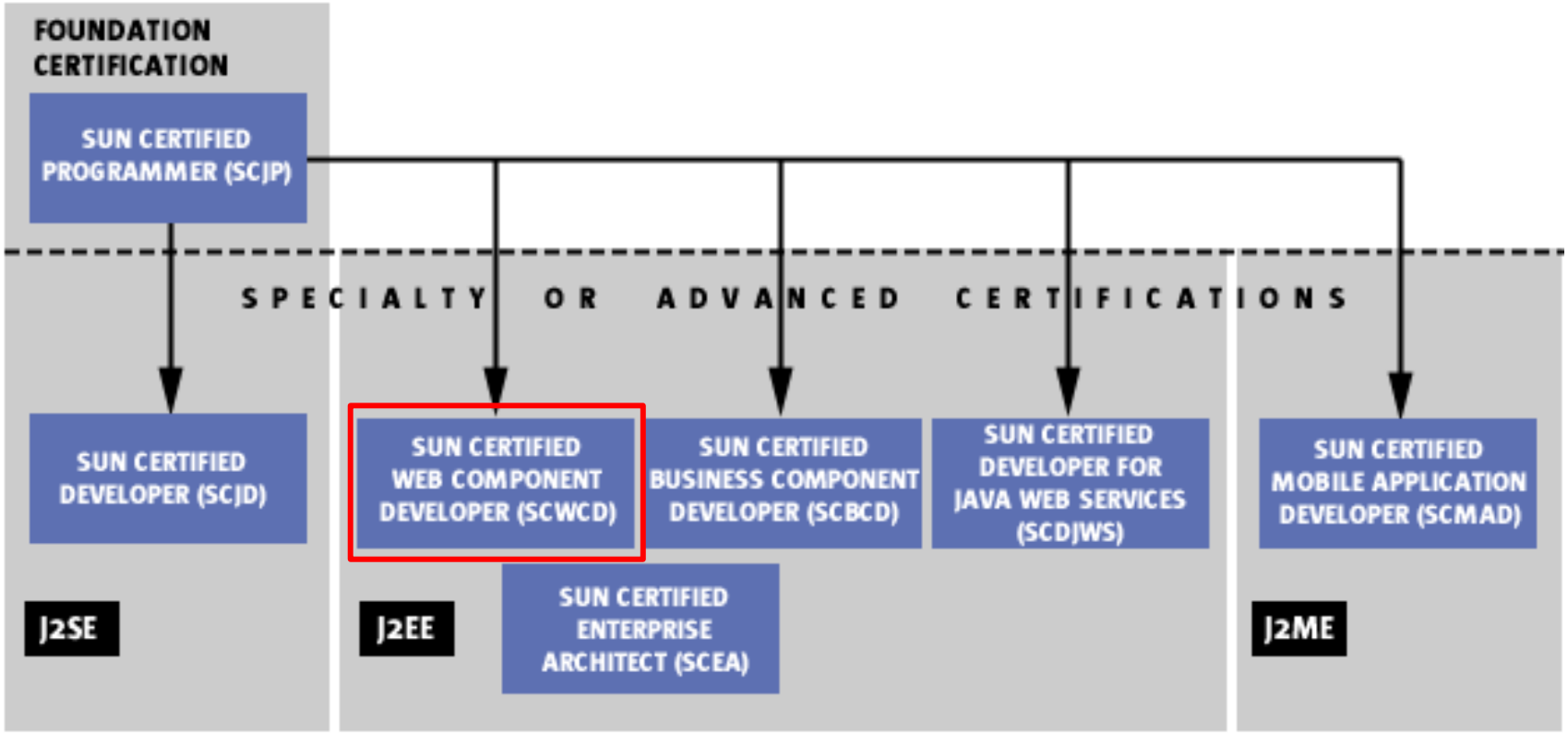
Agenda

1. Certificações oficiais
2. Processo de certificação SWCD
3. Objetivos do exame
4. Revisão técnica e pegadinhas
5. Simulado
6. Correção do simulado

Certificações oficiais

- Representam uma “habilitação” internacional e oficial;
- Formação acadêmica + experiência + certificação = muitos empregos;
- Licitações frequentemente exigem um determinado número de pessoas certificadas;
- Provas são feitas em centros Prometric;
- A certificações agora são vitalícias

Certificações oficiais



Agenda

1. Certificações oficiais
2. Processo de certificação SCWCD
3. Objetivos do exame
4. Revisão técnica e pegadinhas
5. Simulado
6. Correção do simulado
7. Proposta de plano de estudos

Globalcode
The Developers Company

Comprando o voucher

1) Ligar na Sun e solicitar o voucher (0800 55 7863)

3) Você receberá um boleto via e-mail em até 3 dias úteis. Atualmente o voucher custa R$ 330,00.

5) Você terá até 5 dias úteis para realizar o pagamento do boleto.

7) Você receberá o voucher pelo correio em até 15 dias úteis.

9) O voucher tem duração de 1 ano, no entanto, ele chega a Sun do Brasil com 12 meses de validade, mas até que ele chegue as nossas mãos geralmente restam entre 10 e 11 meses até que ele expire.

Agendando a prova

1) Entre no site da prometric (www.prometric.com)

3) Escolha a opção Locate a Test Center

5) Informe o orgão certificador (Sun Microsystems) e o país.

7) Clique no link Locate a Test Center

9) Selecione o Client Sun Microsystems e o programa Sun Microsystems (310,311)

11) Selecione o exame: CX - 310-080 Sun Certified Web Component Developer for the Java 2 Platform, Enterprise Edition 1.3

13) Escolha o centro mais próximo de você e agende sua prova pelo telefone ou no próprio site da prometric.

A prova

Pré-requisitos : Sun Certified Programmer for the Java 2 Platform (qualquer versão)

Tipo de prova: Questões de multipla escolha, questões com respostas curtas e “drag and drop”

Número de questões: 59

Pontuação necessária: 61% (36 questões)

Duração da prova: 90 min

Portanto se você não sabe e não quer estudar inglês...

... já pode começar a estudar chinês, japonês ou koreano!

A prova está disponível nos seguintes idiomas:
Inglês, Chinês, Japonês ou Koreano

A prova

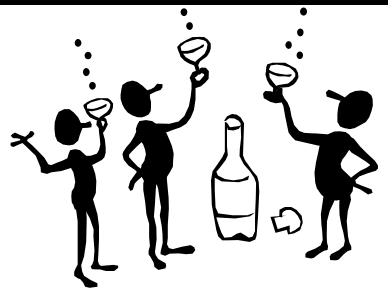
Auto Avaliação

Antes de começar a fazer a prova você terá que responder algumas perguntas fazendo uma auto-avaliação do seu conhecimento, mas o tempo de resposta deste questionário não será subtraido do tempo que você tem para fazer a prova.

O resultado sai na hora

Ao terminar a prova você pode verificar sua pontuação no sistema, e receberá um documento com seu indice de acerto e a sua pontuação por tópico.

A comemoração...



Pronto, basta atualizar seu currículo e enviar um e-mail para a Globalcode dizendo que você é o mais novo profissional certificado do mercado!

Depois de 30 dias de comemoração você receberá um kit contendo:

- Sua “carteirinha” de Sun Certified Web Component Developer the Java 2 Platform Enterprise Edition 1.3
- O certificado
- A documentação para utilização do logo que poderá ser colocado em seu cartão de visitas ou outros documentos.

Você já pode começar a se preparar para a próxima certificação!

Agenda

1. Certificações oficiais
2. Processo de certificação SCWCD
3. Objetivos do exame
4. Revisão técnica e pegadinhas
5. Simulado
6. Correção do simulado

Objetivos do exame

1. O modelo da tecnologia Servlet
3. Estrutura de deployment de aplicações web
5. O modelo Web Container
7. Servlets que gerenciam exceções
9. Servlets com gerenciamento de sessão
11. Desenvolvimento de aplicações web seguras
13. Desenvolvimento de Servlets Thread-safe
15. O modelo da tecnologia JavaServer Pages
17. Desenvolvimento de componentes web reutilizáveis
19. JSPs que utilizam componentes JavaBeans
21. JSPs que utilizam Custom Tags
23. Desenvolvimento de Custom Tag Library

Agenda

1. Certificações oficiais
2. Processo de certificação SCWCD
3. Objetivos do exame
4. Revisão técnica e pegadinhas
5. Simulado
6. Correção do simulado
7. Proposta de plano de estudos

Objetivos do exame

1. O modelo da tecnologia Servlet

- Conhecer os métodos chamados em um HttpServlet em resposta aos métodos HTTP GET, POST e PUT e HEAD

- Identificar a interface e o método que deve ser utilizado para:
  - Recebimento de parâmetros de formulários
  - Leitura de parâmetros de inicialização de Servlets
  - Utilização do HTTP header
  - Recebimento de um stream de texto ou binário da response
  - Redirecionamento de uma request para outra URL

Objetivos do exame

1. O modelo da tecnologia Servlet

- Utilização de atributos nos escopos request, session ou context
- Conhecer os métodos init, service e destroy, assim como o seu propósito e quando são invocados
- Utilização do RequestDispatcher para fazer um include ou forward

Objetivos do exame

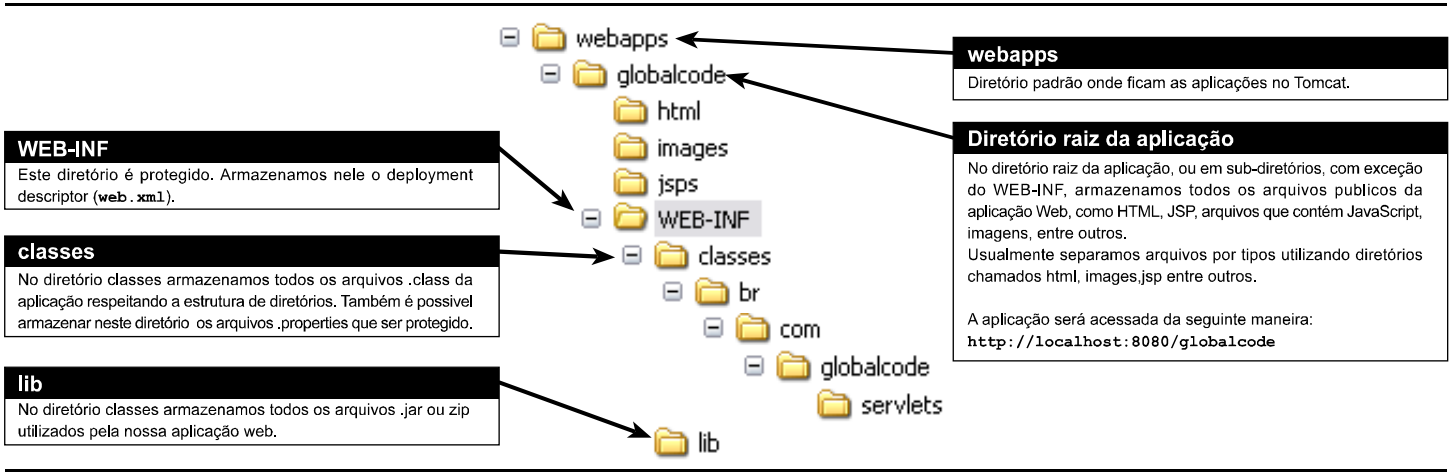
2. Estrutura de deployment de aplicações web

- Conhecer a estrutura de uma aplicação web
- Arquivo WAR;
- Nome do deployment descriptor;
- Conhecer o nome e a utilização dos seguintes elementos do deployment descriptor:
  - Servlet instance
  - Servlet name
  - Servlet class
  - Parâmetros de inicialização
  - URL dos servlets mapeados

Objetivos do exame

2. Estrutura de deployment de aplicações web

- Conhecer a estrutura de uma aplicação web



2. Estrutura de deployment de aplicações web

- Arquivo WAR (Web ARchive)

  Um arquivo WAR agrupa, com ou sem compactação, todos os arquivos de uma Web Application.

  Os servidores de aplicação tem a habilidade de extrair todos os arquivos em um diretório com o mesmo nome do arquivo WAR sem a extensão war.

  Um arquivo WAR pode ser criado com o utilitário jar, ou com qualquer utilitário capaz de criar arquivos zip.

Objetivos do exame

Exemplo de questão

Which of the following statements are TRUE? (Choose Two)

A) Files in the WEB-INF directory can be served dircetly to the client.
B) The servlet class files can reside as a sub-directory of WEB-INF named classes.
C) Files in the WEB-INF directory can not be served directly to the client.
D) The web.xml deployment descriptor can reside outside the WEB-INF directory as long as it has been properly mapped.

Objetivos do exame

Exemplo de questão

Which of the following statements are TRUE? (Choose Two)

A) Files in the WEB-INF directory can be served dircetly to the client.
B) The servlet class files can reside as a sub-directory of WEB-INF named classes.
C) Files in the WEB-INF directory can not be served directly to the client.
D) The web.xml deployment descriptor can reside outside the WEB-INF directory as long as it has been properly mapped.

3. O modelo do Web Container

•Identificar classes,interfaces e elementos do deployment descriptor para:

- Utilizar parâmetros de inicialização do Servlet context
- Eventos da aplicação e listeners
- Configuração da Aplicação Web
- Aplicações Web em ambiente distribuídos
- Session attribute listeners

Objetivos do exame

4. Servlets que gerenciam exceções

- Identificar código correto para manuseio de exceções de negógio para retornar erros HTTP utilizando os métodos **sendError** e **setStatus**

- Dado um conjunto de exceções :
  - Conhecer a configuração que pode ser feita no deployment descriptor para que o container trate as exceptions
  - Utilizar o RequestDispatcher para encaminhar a request para uma página de erro

- Conhecer os métodos utilizados para:
  - Imprimir uma mensagem no log da aplicação
  - Imprimir uma mensagem e uma exceção no log da aplicação

Objetivos do exame

4. Servlets que gerenciam exceções

- Conhecer a configuração que pode ser feita no deployment descriptor para que o container trate as exception

Quando o container recebe uma requisição e faz uma chamada a um dos métodos abaixo, ele poderá receber uma exception do tipo IOException ou ServletException, ou qualquer classe filha.

```
public void doGet(HttpServletRequest req,HttpServletResponse
res) throws IOException, ServletException
```

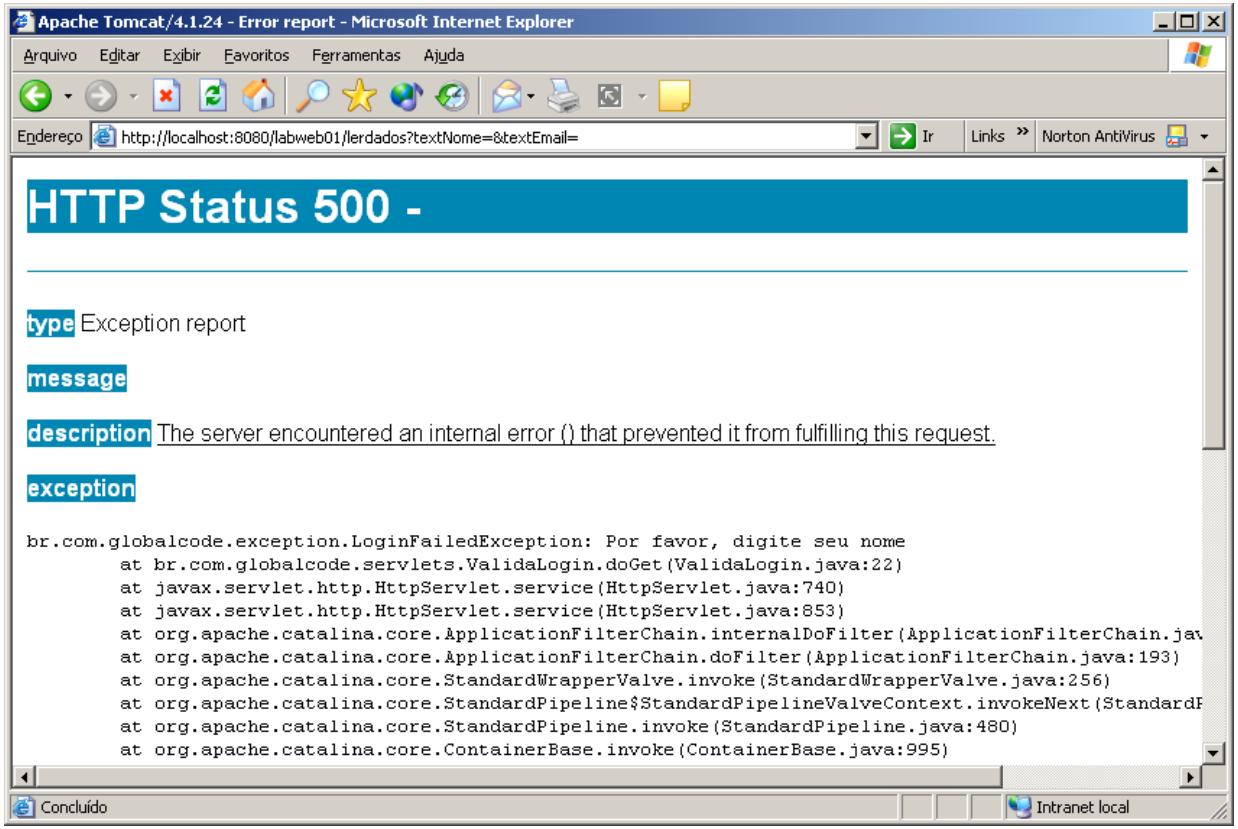
```
public void doPost(HttpServletRequest
req,HttpServletResponse res) throws IOException,
ServletException
```

Ao receber uma exception o container apresenta a mensagem de erro em uma página padrão.

Objetivos do exame

4. Servlets que gerenciam exceções

No Tomcat a página é:



Objetivos do exame

4. Servlets que gerenciam exceções

Para apresentar uma **página de erro personalizada** de acordo com a exceção encontrada podemos configurar no deployment descriptor qual página queremos que seja exibida quando determinada exception for recebida pelo container, ou seja, o container irá **automáticamente redirecionar para a página configurada** quando receber a exception especificada.

```
<web-app>
<servlet>... </servlet>
<servlet-mapping> ...</servlet-mapping>
<error-page>
  <exception-type>LoginFailedException</exception-type>
  <location>/LoginInvalido.html</location>
</error-page>
</web-app>
```

Objetivos do exame

4. Servlets que gerenciam exceções

- Imprimir uma mensagem de erro no log da aplicação
- Imprimir uma mensagem e uma exceção no log da aplicação

Estes métodos são encontrados em ServletContext e GenericServlet:

- void log(String msg)
- void log(String msg, Throwable t)

Os métodos para log da classe GenericServlet **concatenam no inicio da mensagem de erro o nome do Servlet** e os métodos da classe ServletContext não.

Objetivos do exame

Exemplo de questão

An exception caught in your servlet can be logged how?

A) log(message, throwable)
B) log(message, throwable.getMessage())
C) logger(message, throwable.getMessage())
D) logger(message, throwable)

Objetivos do exame

Exemplo de questão

An exception caught in your servlet can be logged how?

A) log(message, throwable)
B) log(message, throwable.getMessage())
C) logger(message, throwable.getMessage())
D) logger(message, throwable)

Objetivos do exame

5. Servlets que utilizam sessões

- Identificar os métodos e interfaces utilizadas para:
  - Obter a sessão
  - Ler e armazenar objetos na sessão
  - Responder a um evento quando um objeto for adicionado na sessão
  - Responder a um evento quando uma sessão for criada ou destruída
  - Invalidar uma sessão
- Identificar quando uma sessão será invalidada

Objetivos do exame

5. Servlets que utilizam sessões

- Identificar quando uma sessão será invalidada

Configuração no deployment descriptor

```
<web-app>
...
<servlet-mapping> ...</servlet-mapping>
<session-config>
  <session-timeout>30</session-timeout>
</session-config >
...
</web-app>
```

- 0 ou menos significa que a session nunca expira
- Tempo em minutos

Objetivos do exame

5. Servlets que utilizam sessões

- Identificando quando uma sessão será utilizada

Utilizando métodos da classe HttpSession

`void setMaxInactiveInterval(int seconds)`

- O método setMaxInactiveInterval(int seconds) afeta somente a sessão em que ele foi chamado
- Um valor negativo significa que a session não deve expirar

`int getMaxInactiveInterval()`

- Retorna o tempo máximo que uma session pode ser mantida sem nenhuma interação do usuário

Objetivos do exame

Exemplo de questão

Select the correct statements about the following web.xml:

```
<web-app>
   ...
    <session-config>
      <session-timeout>15</session-timeout>
    </session-config>
....
```

This entry in the Deployment Descriptor causes what?

A) Only allows users to be connected for 15 minutes total.
B) Only allows users to be connected for 15 days total.
C) Sets the session timeout interval to 15 days.
D) Sets the session timeout interval to 15 minutes.

Objetivos do exame

Exemplo de questão

Select the correct statements about the following web.xml:

```
<web-app>
   ...
    <session-config>
      <session-timeout>15</session-timeout>
    </session-config>
....
```

This entry in the Deployment Descriptor causes what?

A) Only allows users to be connected for 15 minutes total.
B) Only allows users to be connected for 15 days total.
C) Sets the session timeout interval to 15 days.
D) Sets the session timeout interval to 15 minutes.

Objetivos do exame

6. Desenvolvimento de aplicações Web seguras

•Identificar afirmações corretas sobre:
- Autenticação e autorização
- Integridade de dados
- Auditoria
- Código malicioso

•Conhecer os elementos de deployment descriptor para declarar:
- Security constraint
- Web resource
- Configuração de login
- Roles

•Identificar afirmações corretas sobre os seguintes mecanismos de autenticação: BASIC, DIGEST, FORM e CLIENT-CERT

Objetivos do exame

6. Desenvolvimento de aplicações Web seguras

- Autenticação

  - Validar se o usuário realmente é quem ele diz que é...
  - Normalmente a autenticação é feita através de usuário e senha

- Autorização

  - Verificar se um usuário, que já foi autenticado, tem permissão para acessar determinado recurso

Objetivos do exame

6. Desenvolvimento de aplicações Web seguras

- Identificar afirmações corretas sobre os seguintes mecanismos de autenticação: BASIC, DIGEST, FORM e CLIENT-CERT

BASIC: a forma mais fácil de utilizar segurança declarativa, suportada por todos os bronwsers, sem criptorgrafia, aparência não é customizável;

FORM: Muito similar ao BASIC, ou seja, sem criptografia, com aparência customizável pois você é responsável por desenvolver o formulário HTML

Objetivos do exame

6. Desenvolvimento de aplicações Web seguras

- Identificar afirmações corretas sobre os seguintes mecanismos de autenticação: BASIC, DIGEST, FORM e CLIENT-CERT

DIGEST: Não é suportado por todos os browsers, nem por todos os containers pois sua implementação não é obrigatória

CLIENT-CERT: Com criptografia, portanto mais seguro que BASIC, mas requer o uso de certificado digital

7. Desenvolvimento de Servlets Thread-safe

- Identificar quais escopos de atributos são thread-safe:

  - Variáveis locais
  - Atributos de instância (Instance variables)
  - Atributos estáticos (Class variables)
  - Atributos de request
  - Atributos de sessão
  - Atributos de contexto

- Diferenciar multi-threaded e single-threaded Servlets

- Conhecer a interface utilizada para fazer com que um Servlet seja thread-safe

Objetivos do exame

8. O modelo da tecnologia Java Server Pages (JSP)

- Utilizar diretivas, declarações, scriptlet e expressões

- Conhecer os equivalentes em XML das tags JSP

- Conhecer a diretiva page para:
  - Importar uma ou mais classes
  - Indicar a utilização de sessão
  - Definir a página de erro
  - Definir que o JSP é uma página de erro

Objetivos do exame

9. Desenvolvimento de componentes reutilizáveis

- Conhecer os objetos implicitos de um JSP

- Identificar e ordenar as etapas do ciclo de vida do JSP:
  - Tradução (Page translation)
  - Compilação
  - Carregamento de classe
  - Criação da instância
  - Chamda ao método jspInit
  - Chamada ao método _jspService
  - Chamada ao método jspDestroy

Objetivos do exame

9. Desenvolvimento de componentes reutilizáveis

- Conhecer os objetos implicitos de um JSP

Os objetos implicitos de um JSP podem ser utilizados sem ser declarados em um JSP, pois são implicitamente declarados quando o JSP é transformado em um Servlet. Os objetos implicitos são:

- **request**: HttpServletRequest
- **response**: HttpServletResponse
- **session**: HttpSession
- **application**: ServletContext
- **config**: ServletConfig
- **out**: JspWriter
- **page**: Object
- **exception**: Throwable
- **pageContext**: PageContext

9. Desenvolvimento de componentes reutilizáveis

- Conhecer os objetos implicitos de um JSP

Os objetos implicitos de um JSP que não foi declarado como error page são definidos como parâmetros ou variáveis dentro do método _jspService no Servlet gerado a partir do JSP. Vejamos um exemplo:

```
public void _jspService(HttpServletRequest request,
HttpServletResponse response) throws
IOException,ServletException {
    javax.servlet.jsp.PageContext pageContext = null;
    HttpSession session = null;
    ServletContext application = null;
    ServletConfig config = null;
    JspWriter out = null;
    Object page = this;
    ...
```

9. Desenvolvimento de componentes reutilizáveis

- Conhecer os objetos implicitos de um JSP

Quando um JSP é declarado como error page, além dos objetos implicitos vistos anteriormente também fica disponível a Exception que gerou o erro fazendo o JSP ser executado.

```
public void _jspService(HttpServletRequest request,
HttpServletResponse response)throws IOException,
ServletException {

    // outros objetos implicitos

    Throwable exception = (Throwable)
    request.getAttribute("javax.servlet.jsp.jspException");
    ...
```

9. Desenvolvimento de componentes reutilizáveis

- Conhecer os objetos implicitos de um JSP

No Tomcat 4.x você encotra os arquivos resultantes da transformação de um JSP em Servlet no diretório:

TOMCAT_HOME/work/Standalone/localhost/**NomeWebAPP**/

Onde **NomeWebAPP** é o nome da aplicação web onde se encontra o JSP/Serlvet que você deseja visualizar.

Objetivos do exame

Exemplo de questão

Given the following page directive, which two implicit objects are not available to the jsp page? (Choose two)

```
<%@page isErrorPage='false' session='false'
isThreadSafe='false' %>
```

A) session
B) page
C) exception
D) pageContext
E) request

Objetivos do exame

Exemplo de questão

Given the following page directive, which two implicit objects are not available to the jsp page? (Choose two)

```
<%@page isErrorPage='false' session='false'
isThreadSafe='false' %>
```

A) session
B) page
C) exception
D) pageContext
E) request

Objetivos do exame

10. JSPs que utilizam JavaBeans

•Conhecer as tags e seus atributos para:

- Utilizar a tag jsp:useBean
- Utilizar a tag jsp:getProperty
- Utilizar a tag jsp:setProperty

•Conhecer o código equivalente a utilização das tags acima nos escopos:

- request
- session
- application

Objetivos do exame

11. JSPs que utilizam Custom Tags

- Identificar declarações corretas de tags no deployment descriptor
- Identificar diretivas de utilização de tags em um JSP
- Identificar a utilização correta de tags em um JSP, incluindo:
  - Tags vazias
  - Tags com atributos
  - Tags com JSP no corpo
  - Tags aninhadas

11. JSPs que utilizam Custom Tags

- Identificar declarações de tags no deployment descriptor

Para evitar que o todos os JSPs que utilizam uma determinada tag tenham que conhecer o endereço físico do Tag Library Descriptor (TLD) que especifica a tag é comum vincularmos o endereço físico do TLD a uma URI. Isto é feito no deployment descriptor da aplicação.

Objetivos do exame

11. JSPs que utilizam Custom Tags

- Identificar declarações de tags no deployment descriptor

Trecho do web.xml:

```
<web-app>
...
<taglib>
  <taglib-uri>http://www.globalcode.com.br/tld</taglib-uri>
  <taglib-location>/tags/gc.tld</taglib-location>
</error-page>
...
</web-app>
```

Objetivos do exame

11. JSPs que utilizam Custom Tags

- Identificar diretivas de utilização de tags em um JSP

Sempre que formos utilizar uma tag em um JSP precisamos adicionar a diretiva taglib, informando onde está o TLD (físca ou logicamente) e o prefixo que iremos utilizar para a tag.

Exemplo:

```
<%@taglib uri="/WEB-INF/tlds/apostilaWeb.tld" prefix="gc" %>
<%@taglib uri="http://www.globalcode.com.br/tld"
prefix="gc"%>
```

Utilização do endereço físico do arquivo TLD

Utilização do endereço lógico do arquivo TLD

uri : informa a localização física ou lógica do descritor da biblioteca (arquivo .tld)
prefix : informa o prefixo a ser utilizado como nome da biblioteca

Objetivos do exame

12. Desenvolvimento de Custom Tag Libraries

•Conhecer os elementos do descritor da tag (TLD) para:
- Nome da tag
- Declarar a Tag Handler Class
- Declarar o tipo de conteúdo aceito pela tag
- Declarar atributos da tag

•Conhecer os valores que devem ser atribuídos ao elemento bodycontent para definir:
- Uma empty tag
- Tags que contém JSP
- Tags que contém conteúdo que será utilizado pela tag

12. Desenvolvimento de Custom Tag Libraries

- Conhecer os elementos do descritor da tag (TLD) para: nome da tag, tag handler, tipo de conteúdo da tag e atributos

Definir o nome da tag:

```
<name>cliente</name>
```

Este nome será utilizado nas páginas JSP

Definir o tag handler:

```
<tag-class>br.com.globalcode.taglib.ImprimeCliente</tag-class>
```

Classe que implementa a interface javax.servlet.jsp.tagext.Tag

12. Desenvolvimento de Custom Tag Libraries

- Conhecer os elementos do descritor da tag (TLD) para: nome da tag, tag handler, tipo de conteúdo da tag e atributos

Tipo de conteúdo da tag:

```
<body-content>JSP</body-content>
```

Esta tag pode receber um dos seguinte valores:
- **empty**: a tag deve estar obrigatóriamente vazia
- **JSP**: o conteúdo da pode ser qualquer código JSP válido
- **tagdependent**: o conteúdo da tag não deve ser processado pelo interpretador JSP, pois será utilizado pela tag

12. Desenvolvimento de Custom Tag Libraries

- Conhecer os elementos do descritor da tag (TLD) para: nome da tag, tag handler, tipo de conteúdo da tag e atributos

Definindo atributos da tag:

```
<attribute>
  <name>nome</name>
  <required>true</required>
  <rtexprvalue>true</rtexprvalue>
</attribute>
```

Onde o elemento <**name**> define o nome do atributo, o elemento <**required**> define se ao chamar a tag é obrigatória a passagem do atributo e o elemento <**rtexprvalue**> define se podemos passar uma expression para ser avaliada na chamada da tag

Objetivos do exame

Exemplo de questão

The following is an entry in Jakarta's utility tag-lib descriptor.
Select the true statements about this tag.

```
  <tag>
    <name>MacroCopy</name>
    <tag-class>org.apache.taglibs.utility.basic.MacroCopy</tag-
class>
    <description>The Description</description>
    <attribute>
     <name>name</name>
     <required>true</required>
     <rtexprvalue>true</rtexprvalue>
    </attribute>
  </tag>
```

A) The tag must have the attribute "name" supplied. -
B) This is an example of an empty tag.
C) The attribute "name" can be a JSP runtime expression. -
D) This is an example of a nested tag.

Objetivos do exame

Exemplo de questão

The following is an entry in Jakarta's utility tag-lib descriptor.
Select the true statements about this tag.

```
  <tag>
    <name>MacroCopy</name>
    <tag-class>org.apache.taglibs.utility.basic.MacroCopy</tag-
class>
    <description>The Description</description>
    <attribute>
     <name>name</name>
     <required>true</required>
     <rtexprvalue>true</rtexprvalue>
    </attribute>
  </tag>
```

A) The tag must have the attribute "name" supplied.
B) This is an example of an empty tag.
C) The attribute "name" can be a JSP runtime expression.
D) This is an example of a nested tag.

Objetivos do exame

12. Desenvolvimento de Custom Tag Libraries

- Identificar os valores válidos para retorno dos métodos:
  - doStartTag
  - doAfterBody
  - doEndTag
  - PageContext.getOut

- Identificar métodos em custom tags handlers para acessar variáveis implicitas de um JSP ou atributos da página

- Identificar métodos que retornam um outer tag handler dentro de uma inner tag handler

Objetivos do exame

13. Design Patterns

- Os Design Patterns abordados são:
  - Value Objects
  - MVC
  - Data Access Objetct
  - Business Delegate

- Dado um cenário com uma lista de problemas, selecionar o desing patterns que melhor resolvem um determinado problema.

- Relacionar desing patterns aos beneficios que podem ser obtidos com sua utilização

Business Delegate

- **Definição**: em aplicações distribuídas, o acesso remoto / local a EJB's via JNDI Naming Service e tratamento de erros pode se tornar complexo à medida que o projeto cresce.

- **Solução**: criar uma classe intermediária para acessar os EJB's que contempla as regras de nomes de componentes para lookups, propriedades do servidor J2EE, tratamento de exceptions, etc.;

Model-view-controller

- **Definição**: divide o aplicativo em dados, comportamento e apresentação.
- Aplicando MVC podemos reaproveitar o mesmo dado para múltiplas visualizações;
- Podemos reaproveitar o comportamento (eventos) da solução;
- É um “pattern” de arquitetura, criado há muito tempo. Pode ser aplicado em qualquer linguagem, mais facilmente com OOP.

Model-view-controller

- 115.000 resultados na busca sobre framework MVC no google
- Struts, WebWorks, Spring, PicoContainer são exemplos de frameworks J2EE
- Você ainda não fez um framework MVC?
- A modinha dos jovenzitos do Inversion of Control

Data Access Object

- **Definição**: centraliza o serviço de persistência de objetos em um pequeno conjunto de classes, evitando por exemplo que código SQL se espalhe pelo código da solução.

- Mesmo utilizando framework de persistência, utilize Data Access Object

- Exemplo no JAREF

Objetivos do exame

Exemplo de questão

You have data stored in a database and clients that need to access it with
standard browsers or Wireless appliances such as PDA's, which pattern
would best for your application?

A) Value Object
B) Data Access Object
C) Buisness Delegate
D) Model View and Control

Objetivos do exame

Exemplo de questão

You have data stored in a database and clients that need to access it with
standard browsers or Wireless appliances such as PDA's, which pattern
would best for your application?

A) Value Object
B) Data Access Object
C) Buisness Delegate
D) Model View and Control

Agenda

1. Certificações oficiais
2. Processo de certificação SCWCD
3. Objetivos do exame
4. Revisão técnica e pegadinhas
5. Simulado
6. Correção do simulado

Agenda

Tempo para resolução do simulado

Agenda

1. Certificações oficiais
2. Processo de certificação SCJP
3. Objetivos do exame
4. Revisão técnica e pegadinhas
5. Simulado
6. Correção do simulado

Agenda

Correção do simulado

Objetivos do exame

Questão 1

1) You add web components to a J2EE application in a package called WAR (web application archive). A WAR has a specific hierarchial directory structure. Which directory stores the deployment descriptor for a Web application located in the myapp directory?

A) myapp/WEB-INF/public_html
B) myapp/WEB-INF/classes
C) myapp/WEB-INF
D) myapp/WEB-INF/lib

Objetivos do exame

Questão 1

1) You add web components to a J2EE application in a package called WAR (web application archive). A WAR has a specific hierarchial directory structure. Which directory stores the deployment descriptor for a Web application located in the myapp directory?

A) myapp/WEB-INF/public_html
B) myapp/WEB-INF/classes
C) myapp/WEB-INF
D) myapp/WEB-INF/lib

Objetivos do exame

Questão 2

2) Which of the following methods may not be defined in a JSP page?
Select one choice

A) void jspInit()
B) void jspDestroy()
C) void _jspService(HttpServletRequest request, HttpServletResponse response)
D) All of the above
E) None of the above

Objetivos do exame

Questão 2

2) Which of the following methods may not be defined in a JSP page?
Select one choice

A) void jspInit()
B) void jspDestroy()
C) void _jspService(HttpServletRequest request, HttpServletResponse response)
D) All of the above
E) None of the above

Objetivos do exame

Questão 3

3) "Acme Inc" decides to create a new web application for offering its services to clients over Internet. The company has a software development team with programmers who are good in UI programming but lacks business component skills. So the company decides to outsource its business services development to an external software house. Which design patterns, when applied in combination, will result in decoupled client software along with an increased network performance? (Choose 3)

A) Buisiness Delegate
B) Front Controller
C) Transfer Object
D) Service Locator
E) Intercepting Filter
F) Model-View-Controller

Objetivos do exame

Questão 3

3) "Acme Inc" decides to create a new web application for offering its services to clients over Internet. The company has a software development team with programmers who are good in UI programming but lacks business component skills. So the company decides to outsource its business services development to an external software house. Which design patterns, when applied in combination, will result in decoupled client software along with an increased network performance? (Choose 3)

A) Buisiness Delegate
B) Front Controller
C) Transfer Object
D) Service Locator
E) Intercepting Filter
F) Model-View-Controller

Objetivos do exame

Questão 4

4) A client types the URI :
http://www.someserver.com/servlet/SomeServlet?name=fred
The value of name can be obtained by: (Choose 2)

A) doGet(HttpServletRequest request, HttpServletResponse response){
String name = response.getParameter("name");
...
B) doGet(HttpServletRequest request, HttpServletResponse response){
String name = request.getParameter("name");
...
C) doGet(HttpServletRequest request, HttpServletResponse response){
String name = request.getInitParameter("name");
...
D) service(HttpServletRequest request, HttpServletResponse response){
String name = request.getParameter("name");
...

Objetivos do exame

Questão 4

4) A client types the URI :
http://www.someserver.com/servlet/SomeServlet?name=fred
The value of name can be obtained by: (Choose 2)

A) doGet(HttpServletRequest request, HttpServletResponse response){
String name = response.getParameter("name");
...

B) doGet(HttpServletRequest request, HttpServletResponse response){
String name = request.getParameter("name");
...

C) doGet(HttpServletRequest request, HttpServletResponse response){
String name = request.getInitParameter("name");
...

D) service(HttpServletRequest request, HttpServletResponse response){
String name = request.getParameter("name");
...

Objetivos do exame

Questão 5

5) A class that will recieve notification if an Attribute is added to or removed
from a session will implement which interface?

A) HttpSessionListener
B) HttpSessionObjectListener
C) HttpSessionAttributeListener
D) HttpSessionBindingListener

Objetivos do exame

Questão 5

5) A class that will recieve notification if an Attribute is added to or removed
from a session will implement which interface?

A) HttpSessionListener
B) HttpSessionObjectListener
C) HttpSessionAttributeListener
D) HttpSessionBindingListener

Objetivos do exame

Questão 6

6) If a servlet implements SingleThreadModel interface which of the following methods
MUST also be implemented? (Choose all that apply)

A) synchronize
B) lock
C) No methods need to be implemented. -
D) release

Objetivos do exame

Questão 6

6) If a servlet implements SingleThreadModel interface which of the following methods
MUST also be implemented? (Choose all that apply)

A) synchronize
B) lock
C) No methods need to be implemented
D) release

Objetivos do exame

Questão 7

7) Select the correct mandatory sub-elements of the <taglib> element in web.xml.(Choose 2)

A) uri
B) id
C) taglib-location
D) taglib-uri
E) location

Objetivos do exame

Questão 7

7) Select the correct mandatory sub-elements of the <taglib> element in web.xml.(Choose 2)

A) uri
B) id
C) taglib-location
D) taglib-uri
E) location

Objetivos do exame

Questão 8

8) You have an Applet that needs to recieve a large block of data from a server, manipulate it on the client and return it to the server, which pattern
is best suited for this situation?

A) Business Delegate
B) Model View and Control
C) Value Objects
D) Data Access Object

Objetivos do exame

Questão 8

8) You have an Applet that needs to recieve a large block of data from a server, manipulate it on the client and return it to the server, which pattern
is best suited for this situation?

A) Business Delegate
B) Model View and Control
C) Value Objects
D) Data Access Object

Objetivos do exame

Questão 9

9) Your JSP Bean "People" has a private member "friend" with proper setters and getters, you would have which line in your jsp page to obtain it's value?

A) <jsp:getProperty name="friend" />
B) <jsp:getProperty name="People" property="friend"/>
C) <jsp:getProperty name="friend" property="People"/>
D) <jsp:getProperty property="People.friend"/>

Objetivos do exame

Questão 9

9) Your JSP Bean "People" has a private member "friend" with proper setters and getters, you would have which line in your jsp page to obtain it's value?

A) <jsp:getProperty name="friend" />
B) <jsp:getProperty name="People" property="friend"/>
C) <jsp:getProperty name="friend" property="People"/>
D) <jsp:getProperty property="People.friend"/>

Objetivos do exame

Questão 10

10) The statement :
The web client obtains the username and password from the user and transmits it to the server using Base64 encoding.

Best describes which Authentication scheme?

A) Http Basic Authentication
B) Form Based Authentication
C) Http Digest Authentication
D) Https Client Authentication

Objetivos do exame

Questão 10

10) The statement :
The web client obtains the username and password from the user and transmits it to the server using Base64 encoding.

Best describes which Authentication scheme?

A) Http Basic Authentication
B) Form Based Authentication
C) Http Digest Authentication
D) Https Client Authentication

Agenda

1. Certificações oficiais
2. Processo de certificação SCJP
3. Objetivos do exame
4. Revisão técnica e pegadinhas
5. Simulado
6. Correção do simulado