

CONTENTS INCLUDE:

- Development Process
- Lifecycle
- Faces-config.xml
- The JSF Expression Language
- JSF Core Tags
- JSF HTML Tags and more...

JavaServer Faces

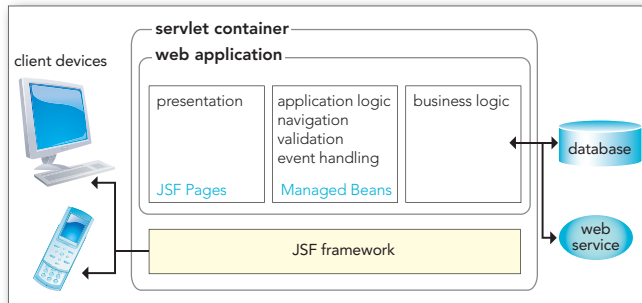
By Cay S. Horstmann

ABOUT THIS REFCARD

JavaServer Faces (JSF) is the “official” component-based view technology in the Java EE web tier. JSF includes a set of predefined UI components, an event-driven programming model, and the ability to add third-party components. JSF is designed to be extensible, easy to use, and toolable. This refcard describes the JSF development process, standard JSF tags, the JSF expression language, and the `faces-config.xml` configuration file.

DEVELOPMENT PROCESS

A developer specifies JSF components in JSF pages, combining JSF component tags with HTML and CSS for styling. Components are linked with *managed beans*—Java classes that contain presentation logic and connect to business logic and persistence backends. Entries in `faces-config.xml` contain navigation rules and instructions for loading managed beans.



A JSF page has the following structure:

JSP Style	Proper XML
<pre><html> <%@ taglib uri="http:// java.sun.com/jsf/ core" prefix="f" %> <%@ taglib uri="http:// java.sun.com/jsf/ html" prefix="h" %> <f:view> <head> <title>...</ title> </head> <body> <h:form> ... </h:form> </body> </f:view> </html></pre>	<pre><?xml version="1.0" encoding="UTF-8"?> <jsp:root xmlns:jsp="http://java.sun.com/ JSP/Page" xmlns:f="http://java.sun.com/jsf/core" xmlns:h="http://java.sun.com/jsf/html" version="2.0"> <jsp:directive.page contentType="text/ html;charset=UTF-8"/> <jsp:output omit-xml-declaration="no" doctype-root-element="html" doctype-public="-//W3C//DTD XHTML 1.0 Transitional//EN" doctype-system="http://www.w3.org/TR/ xhtml1/ DTD/xhtml1-transitional.dtd"/> <f:view> <html xmlns="http://www.w3.org/1999/ xhtml"> <head> <title>...</title> </head> <body> <h:form>...</h:form> </body> </html> </f:view> </jsp:root></pre>

These common tasks give you a crash course into using JSF.

Text field

page.jspx
`<h:inputText value="#{bean1.luckyNumber}"/>`

faces-config.xml

```
<managed-bean>
<managed-bean-name>bean1</managed-bean-name>
<managed-bean-class>com.corejsf.SampleBean</
managed-bean-class>
<managed-bean-scope>session</managed-bean-scope>
</managed-bean>
```

com/corejsf/SampleBean.java

```
public class SampleBean {
public int getLuckyNumber() { ... }
public void setLuckyNumber(int value) { ... }
...
}
```

Button

page.jspx
`<h:commandButton value="press me" action="#{bean1.
login}"/>`

faces-config.xml

```
<navigation-rule>
<navigation-case>
<from-outcome>success</from-outcome>
<to-view-id>success.jspx</to-view-id>
</navigation-case>
<navigation-case>
<from-outcome>error</from-outcome>
<to-view-id>error.jspx</to-view-id>
<redirect/>
</navigation-case>
</navigation-rule>
```

ICEfaces
Ajax for Java EE

SCALAB OPEN SOURCE AJAX PUSH
N SOURCE AJAX PUSH NO JAVASC
IX PUSH NO JAVASCRIPT INTEROPE
CRIPT INTEROPERABILITY SCALABI
RABILITY SCALABILITY SECURITY OPEN S
ALABILITY SECURITY OPEN SOURCE AJA

The Leading Ajax Solution
for Java Developers

www.icefaces.org

Button, continued

com/corejsf/SampleBean.java

```
public class SampleBean {
    public String login() { if (...) return
        "success"; else return "error"; }
    ...
}
```

Radio buttons Cheese Pickle Mustard Lettuce

page.jspx

```
<h:selectOneRadio value="#{form.condiment}>
    <f:selectItems value="#{form.condimentItems}"/>
</h:selectOneRadio>
```

com/corejsf/SampleBean.java

```
public class SampleBean {
    private Map<String, Object> condimentItem = null;
    public Map<String, Object> getCondimentItems() {
        if (condimentItem == null) {
            condimentItem = new LinkedHashMap<String,
                Object>();
            condimentItem.put("Cheese", 1); // label, value
            condimentItem.put("Pickle", 2);
            ...
        }
        return condimentItem;
    }
    public int getCondiment() { ... }
    public void setCondiment(int value) { ... }
    ...
}
```

Validation and Conversion

page.jspx

```
<h:inputText value="#{payment.amount}"
    required="true">
    <f:validateDoubleRange maximum="1000"/>
</h:inputText>
<h:outputText value="#{payment.amount}">
    <f:convertNumber type="currency"/>
    <!-- number displayed with currency symbol and
        group separator: $1,000.00 -->
</h:outputText>
```

Error Messages

Amount Amount: 'too much' is not a number.
Example: 99

page.jspx

```
<h:outputText value="Amount"/>
<h:inputText id="amount" label="Amount"
    value="#{payment.amount}"/>
    <!-- label is used in message text -->
<h:message for="amount"/>
```

Resources and Styles

page.jspx

```
<head>
    <link href="styles.css" rel="stylesheet"
        type="text/css"/>
    ...
</head>
<body>
```

```
...
<h:outputText value="#{msgs.goodbye}!"
    styleClass="goodbye">
    ...
</body>
```

faces-config.xml

```
<application>
<resource-bundle>
    <base-name>com.corejsf.messages</base-name>
    <var>msgs</var>
</resource-bundle>
</application>
```

com/corejsf/messages.properties

goodbye=Goodbye

com/corejsf/messages_de.properties

goodbye=Auf Wiedersehen

styles.css

```
.goodbye {
    font-style: italic;
    font-size: 1.5em;
    color: #eee;
}
```

Table with links

Name	
Washington, George	Delete
Jefferson, Thomas	Delete
Lincoln, Abraham	Delete
Roosevelt, Theodore	Delete

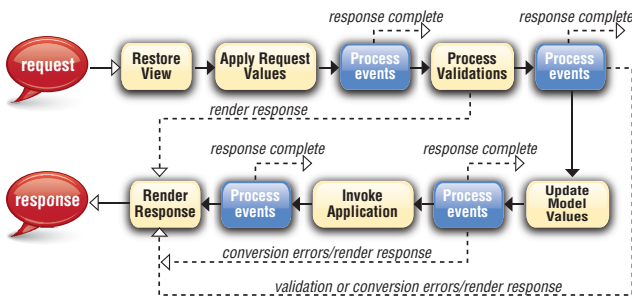
page.jspx

```
<h:dataTable value="#{bean1.entries}" var="row"
    styleClass="table" rowClasses="even,odd">
    <h:column>
        <f:facet name="header">
            <h:outputText value="Name"/>
        </f:facet>
        <h:outputText value="#{row.name}"/>
    </h:column>
    <h:column>
        <h:commandLink value="Delete" action="#{bean1.
            deleteAction}" immediate="true">
            <f:setPropertyActionListener target="#{bean1.
                idToDelete}"
                value="#{row.id}"/>
        </h:commandLink>
    </h:column>
</h:dataTable>
```

com/corejsf/SampleBean.java

```
public class SampleBean {
    private int idToDelete;
    public void setIdToDelete(int value) {idToDelete
        = value; }
    public String deleteAction() {
        // delete the entry whose id is idToDelete
        return null;
    }
    public List<Entry> getEntries() {...}
    ...
}
```

LIFECYCLE



faces-config.xml

The faces-config.xml file contains a sequence of the following entries.

- managed-bean
 1. description, display-name, icon (optional)
 2. managed-bean-name
 3. managed-bean-class
 4. managed-bean-scope
 5. managed-property (0 or more, other optional choices are map-entries and list-entries which are not shown here)
 1. description, display-name, icon (optional)
 2. property-name
 3. value (other choices are null-value, map-entries, list-entries which are not shown here)
- navigation-rule
 1. description, display-name, icon (optional)
 2. from-view-id (optional, can use wildcards)
 3. navigation-case (1 or more)
 - from-action (optional, not common)
 - from-outcome
 - to-view-id
- application
 - resource-bundle
 1. base-name
 2. var
 - action-listener, default-render-kit-id, resource-bundle, view-handler, state-manager, el-resolver, property-resolver, variable-resolver, application-extension (details not shown)
- converter
 - converter-id
 - converter-class
 - Optional initialization (not shown here)
- validator
 - validator-id
 - validator-class
 - Optional initialization (not shown here)
- lifecycle
 - phase-listener
- component, factory, referenced-bean, render-kit, faces-config-extension (details not shown)

web.xml

```
<?xml version="1.0"?>
<web-app xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
  http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
  version="2.5">
  <servlet>
    <servlet-name>Faces Servlet</servlet-name>
    <servlet-class>javax.faces.webapp.FacesServlet</
    servlet-class>
    <load-on-startup>1</load-on-startup>
  </servlet>

  <!-- Add this element to change the extension for
  JSF page files (Default: .jsp) -->
  <context-param>
    <param-name>javax.faces.DEFAULT_SUFFIX</param-
    name>
    <param-value>.jspx</param-value>
  </context-param>

  <servlet-mapping>
    <servlet-name>Faces Servlet</servlet-name>
    <url-pattern>*.faces</url-pattern>
    <!-- Another popular URL pattern is /faces/* -->
  </servlet-mapping>

  <welcome-file-list>
    <welcome-file>index.faces</welcome-file>
    <!-- Create a blank index.faces (in addition to
    index.jsp) -->
    <welcome-file>index.html</welcome-file>
    <!-- Use <meta http-equiv="Refresh" content="
    "0; URL=index.faces"/> -->
  </welcome-file-list>
</web-app>
```

THE JSF EXPRESSION LANGUAGE (EL)

An EL expression is a sequence of literal strings and expressions of the form `base[expr1][expr2]...` As in JavaScript, you can write `base.identifier` instead of `base['identifier']` or `base["identifier"]`. The base is one of the names in the table below or a name defined in faces-config.xml.

header	A Map of HTTP header parameters, containing only the first value for each name.
headerValues	A Map of HTTP header parameters, yielding a String[] array of all values for a given name.
param	A Map of HTTP request parameters, containing only the first value for each name.
paramValues	A Map of HTTP request parameters, yielding a String[] array of all values for a given name.
cookie	A Map of the cookie names and values of the current request.
initParam	A Map of the initialization parameters of this web application.
requestScope	A Map of all request scope attributes.
sessionScope	A Map of all session scope attributes.
applicationScope	A Map of all application scope attributes.
facesContext	The FacesContext instance of this request.
view	The UIViewRoot instance of this request.

The JSF Expression Language (EL), continued

There are two expression types:

- Value expression: a reference to a bean property or an entry in a map, list, or array. Examples:
 - `userBean.name` calls `getName` or `setName` on the `userBean` object
 - `pizza.choices[var]` calls `pizza.getChoices().get(var)` or `pizza.getChoices().put(var, ...)`
- Method expression: a reference to a method and the object on which it is to be invoked. Example:
 - `userBean.login` calls the `login` method on the `userBean` object when it is invoked.

In JSF, EL expressions are enclosed in `#{...}` to indicate *deferred evaluation*. The expression is stored as a string and evaluated when needed. In contrast, JSP uses immediate evaluation, indicated by `${...}` delimiters.

JSF CORE TAGS

Tag	Description/ Attributes												
f:view	Creates the top-level view <table border="1"> <tr> <td>locale</td> <td>The locale for this view.</td> </tr> <tr> <td>renderKitId (JSF 1.2)</td> <td>The render kit ID for this view</td> </tr> <tr> <td>beforePhase, afterPhase</td> <td>Phase listeners that are called in every phase except "restore view"</td> </tr> </table>	locale	The locale for this view.	renderKitId (JSF 1.2)	The render kit ID for this view	beforePhase, afterPhase	Phase listeners that are called in every phase except "restore view"						
locale	The locale for this view.												
renderKitId (JSF 1.2)	The render kit ID for this view												
beforePhase, afterPhase	Phase listeners that are called in every phase except "restore view"												
f:subview	Creates a subview of a view <table border="1"> <tr> <td>binding, id, rendered</td> <td>Basic attributes</td> </tr> </table>	binding, id, rendered	Basic attributes										
binding, id, rendered	Basic attributes												
f:facet	Adds a facet to a component <table border="1"> <tr> <td>name</td> <td>the name of this facet</td> </tr> </table>	name	the name of this facet										
name	the name of this facet												
f:attribute	Adds an attribute to a component <table border="1"> <tr> <td>name, value</td> <td>the name and value of the attribute to set</td> </tr> </table>	name, value	the name and value of the attribute to set										
name, value	the name and value of the attribute to set												
f:param	Constructs a parameter child component <table border="1"> <tr> <td>name</td> <td>An optional name for this parameter component.</td> </tr> <tr> <td>value</td> <td>The value stored in this component.</td> </tr> <tr> <td>binding, id</td> <td>Basic attributes</td> </tr> </table>	name	An optional name for this parameter component.	value	The value stored in this component.	binding, id	Basic attributes						
name	An optional name for this parameter component.												
value	The value stored in this component.												
binding, id	Basic attributes												
f:actionListener f:valueChangeListener	Adds an action listener or value change listener to a component <table border="1"> <tr> <td>type</td> <td>The name of the listener class</td> </tr> </table>	type	The name of the listener class										
type	The name of the listener class												
f:setPropertyChange Listener (JSF 1.2)	Adds an action listener to a component that sets a bean property to a given value. <table border="1"> <tr> <td>value</td> <td>The bean property to set when the action event occurs</td> </tr> <tr> <td>binding, id</td> <td>The value to set it to</td> </tr> </table>	value	The bean property to set when the action event occurs	binding, id	The value to set it to								
value	The bean property to set when the action event occurs												
binding, id	The value to set it to												
f:converter	Adds an arbitrary converter to a component <table border="1"> <tr> <td>converterId</td> <td>The ID of the converter</td> </tr> </table>	converterId	The ID of the converter										
converterId	The ID of the converter												
f:convertDateTime	Adds a datetime converter to a component <table border="1"> <tr> <td>type</td> <td>date (default), time, or both</td> </tr> <tr> <td>dateStyle</td> <td>default, short, medium, long, or full</td> </tr> <tr> <td>timeStyle</td> <td>default, short, medium, long, or full</td> </tr> <tr> <td>pattern</td> <td>Formatting pattern, as defined in <code>java.text.SimpleDateFormat</code></td> </tr> <tr> <td>locale</td> <td>Locale whose preferences are to be used for parsing and formatting</td> </tr> <tr> <td>timezone</td> <td>Time zone to use for parsing and formatting</td> </tr> </table>	type	date (default), time, or both	dateStyle	default, short, medium, long, or full	timeStyle	default, short, medium, long, or full	pattern	Formatting pattern, as defined in <code>java.text.SimpleDateFormat</code>	locale	Locale whose preferences are to be used for parsing and formatting	timezone	Time zone to use for parsing and formatting
type	date (default), time, or both												
dateStyle	default, short, medium, long, or full												
timeStyle	default, short, medium, long, or full												
pattern	Formatting pattern, as defined in <code>java.text.SimpleDateFormat</code>												
locale	Locale whose preferences are to be used for parsing and formatting												
timezone	Time zone to use for parsing and formatting												



JSF Core Tags, continued

Tag	Description/ Attributes																						
f:convertNumber	Adds a number converter to a component <table border="1"> <tr> <td>type</td> <td>number (default), currency, or percent</td> </tr> <tr> <td>pattern</td> <td>Formatting pattern, as defined in <code>java.text.DecimalFormat</code></td> </tr> <tr> <td>maxFractionDigits</td> <td>Maximum number of digits in the fractional part</td> </tr> <tr> <td>minFractionDigits</td> <td>Minimum number of digits in the fractional part</td> </tr> <tr> <td>maxIntegerDigits</td> <td>Maximum number of digits in the integer part</td> </tr> <tr> <td>minIntegerDigits</td> <td>Minimum number of digits in the integer part</td> </tr> <tr> <td>integerOnly</td> <td>True if only the integer part is parsed (default: false)</td> </tr> <tr> <td>groupingUsed</td> <td>True if grouping separators are used (default: true)</td> </tr> <tr> <td>locale</td> <td>Locale whose preferences are to be used for parsing and formatting</td> </tr> <tr> <td>currencyCode</td> <td>ISO 4217 currency code to use when converting currency values</td> </tr> <tr> <td>currencySymbol</td> <td>Currency symbol to use when converting currency values</td> </tr> </table>	type	number (default), currency, or percent	pattern	Formatting pattern, as defined in <code>java.text.DecimalFormat</code>	maxFractionDigits	Maximum number of digits in the fractional part	minFractionDigits	Minimum number of digits in the fractional part	maxIntegerDigits	Maximum number of digits in the integer part	minIntegerDigits	Minimum number of digits in the integer part	integerOnly	True if only the integer part is parsed (default: false)	groupingUsed	True if grouping separators are used (default: true)	locale	Locale whose preferences are to be used for parsing and formatting	currencyCode	ISO 4217 currency code to use when converting currency values	currencySymbol	Currency symbol to use when converting currency values
type	number (default), currency, or percent																						
pattern	Formatting pattern, as defined in <code>java.text.DecimalFormat</code>																						
maxFractionDigits	Maximum number of digits in the fractional part																						
minFractionDigits	Minimum number of digits in the fractional part																						
maxIntegerDigits	Maximum number of digits in the integer part																						
minIntegerDigits	Minimum number of digits in the integer part																						
integerOnly	True if only the integer part is parsed (default: false)																						
groupingUsed	True if grouping separators are used (default: true)																						
locale	Locale whose preferences are to be used for parsing and formatting																						
currencyCode	ISO 4217 currency code to use when converting currency values																						
currencySymbol	Currency symbol to use when converting currency values																						
f:validator	Adds a validator to a component <table border="1"> <tr> <td>validatorId</td> <td>The ID of the validator</td> </tr> </table>	validatorId	The ID of the validator																				
validatorId	The ID of the validator																						
f:validateDoubleRange f:validateLongRange f:validateLength	Validates a double or long value, or the length of a string <table border="1"> <tr> <td>minimum, maximum</td> <td>the minimum and maximum of the valid rang</td> </tr> </table>	minimum, maximum	the minimum and maximum of the valid rang																				
minimum, maximum	the minimum and maximum of the valid rang																						
f:loadBundle	Loads a resource bundle, stores properties as a Map <table border="1"> <tr> <td>basename</td> <td>The resource bundle name</td> </tr> <tr> <td>value</td> <td>The name of the variable that is bound to the bundle map</td> </tr> </table>	basename	The resource bundle name	value	The name of the variable that is bound to the bundle map																		
basename	The resource bundle name																						
value	The name of the variable that is bound to the bundle map																						
f:selectItems	Specifies items for a select one or select many component <table border="1"> <tr> <td>binding, id</td> <td>Basic attributes</td> </tr> <tr> <td>value</td> <td>Value expression that points to a <code>SelectItem</code>, an array or <code>Collection</code> of <code>SelectItem</code> objects, or a <code>Map</code> mapping labels to values.</td> </tr> </table>	binding, id	Basic attributes	value	Value expression that points to a <code>SelectItem</code> , an array or <code>Collection</code> of <code>SelectItem</code> objects, or a <code>Map</code> mapping labels to values.																		
binding, id	Basic attributes																						
value	Value expression that points to a <code>SelectItem</code> , an array or <code>Collection</code> of <code>SelectItem</code> objects, or a <code>Map</code> mapping labels to values.																						
f:selectItem	Specifies an item for a select one or select many component <table border="1"> <tr> <td>binding, id</td> <td>Basic attributes</td> </tr> <tr> <td>itemDescription</td> <td>Description used by tools only</td> </tr> <tr> <td>itemDisabled</td> <td>Boolean value that sets the item's disabled property</td> </tr> <tr> <td>itemLabel</td> <td>Text shown by the item</td> </tr> <tr> <td>itemValue</td> <td>Item's value, which is passed to the server as a request parameter</td> </tr> <tr> <td>value</td> <td>Value expression that points to a <code>SelectItem</code> instance</td> </tr> </table>	binding, id	Basic attributes	itemDescription	Description used by tools only	itemDisabled	Boolean value that sets the item's disabled property	itemLabel	Text shown by the item	itemValue	Item's value, which is passed to the server as a request parameter	value	Value expression that points to a <code>SelectItem</code> instance										
binding, id	Basic attributes																						
itemDescription	Description used by tools only																						
itemDisabled	Boolean value that sets the item's disabled property																						
itemLabel	Text shown by the item																						
itemValue	Item's value, which is passed to the server as a request parameter																						
value	Value expression that points to a <code>SelectItem</code> instance																						
f:verbatim	Adds markup to a JSF page <table border="1"> <tr> <td>escape</td> <td>If set to true, escapes <, >, and & characters. Default value is false.</td> </tr> <tr> <td>rendered (JSF 1.2)</td> <td>Basic attributes</td> </tr> </table>	escape	If set to true, escapes <, >, and & characters. Default value is false.	rendered (JSF 1.2)	Basic attributes																		
escape	If set to true, escapes <, >, and & characters. Default value is false.																						
rendered (JSF 1.2)	Basic attributes																						

JSF HTML TAGS

Tag	Description
h:form	HTML form
h:inputText	Single-line text input control <input type="text" value="12345678901234567890"/>
h:inputTextarea	Multiline text input control <input type="text" value="line one line two line three"/>
h:inputSecret	Password input control <input type="password" value="*****"/>
h:inputHidden	Hidden field
h:outputLabel	Label for another component for accessibility

JSF HTML tags, continued

Tag	Description	
h:outputLink	HTML anchor	
h:outputFormat	Like <code>outputText</code> , but formats compound messages	
h:outputText	Single-line text output	
h:commandButton	Button: submit, reset, or pushbutton	<input type="button" value="press me"/>
h:commandLink	Link that acts like a pushbutton.	register
h:message	Displays the most recent message for a component	Amount <input type="text" value="too much"/> Amount: 'too much' is not a number. Example: 99
h:messages	Displays all messages	
h:graphicImage	Displays an image	
h:selectOneListbox	Single-select listbox	<input type="list" value="Cheese"/> Cheese Pickles Mustard Lettuce
h:selectOneMenu	Single-select menu	<input type="text" value="Pickles"/> Cheese Pickles Mustard Lettuce Onions
h:selectOneRadio	Set of radio buttons	<input type="radio"/> Cheese <input type="radio"/> Pickles
h:selectBooleanCheckbox	Checkbox	Receive email: <input checked="" type="checkbox"/>
h:selectManyCheckbox	Multiselect listbox	<input checked="" type="checkbox"/> Cheese <input type="checkbox"/> Pickles <input checked="" type="checkbox"/> Mustard
h:selectManyListbox	Multiselect listbox	<input type="list" value="Cheese"/> Cheese Pickles Mustard Lettuce Onions
h:selectManyMenu	Multiselect menu	<input type="text" value="Lettuce"/>
h:panelGrid	HTML table	
h:panelGroup	Two or more components that are laid out as one	
h:dataTable	A feature-rich table component	
h:column	Column in a data table	

Basic Attributes

Attribute	Description
id	Identifier for a component
binding	Reference to the component that can be used in a backing bean
rendered	A boolean; false suppresses rendering
styleClass	Cascading stylesheet (CSS) class name
value	A component's value, typically a value binding
valueChangeListener	A method binding to a method that responds to value changes
converter	Converter class name
validator	Class name of a validator that's created and attached to a component
required	A boolean; if true, requires a value to be entered in the associated field

Attributes for h:form

Attribute	Description
binding, id, rendered, styleClass	Basic attributes
accept, acceptcharset, dir, enctype, lang, style, target, title	HTML 4.0 attributes (acceptcharset corresponds to HTML accept-charset)
onblur, onchange, onclick, ondblclick, onfocus, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onreset, onsubmit	DHTML events

Attributes for h:inputText, h:inputSecret, h:inputTextarea, and h:inputHidden

Attribute	Description
cols	For <code>h:inputTextarea</code> only—number of columns
immediate	Process validation early in the life cycle
redisplay	For <code>h:inputSecret</code> only—when true, the input field's value is redisplayed when the web page is reloaded
required	Require input in the component when the form is submitted
rows	For <code>h:inputTextarea</code> only—number of rows
valueChangeListener	A specified listener that's notified of value changes
binding, converter, id, rendered, required, styleClass, value, validator	Basic attributes
accesskey, alt, dir, disabled, lang, maxlength, readonly, size, style, tabindex, title	HTML 4.0 pass-through attributes— <code>alt</code> , <code>maxlength</code> , and <code>size</code> do not apply to <code>h:inputTextarea</code> . None apply to <code>h:inputHidden</code>
onblur, onchange, onclick, ondblclick, onfocus, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onselect	DHTML events. None apply to <code>h:inputHidden</code>

Attributes for h:outputText and h:outputFormat

Attribute	Description
escape	If set to true, escapes <code><</code> , <code>></code> , and <code>&</code> characters. Default value is true.
binding, converter, id, rendered, styleClass, value	Basic attributes
style, title	HTML 4.0

Attributes for h:outputLabel

Attribute	Description
for	The ID of the component to be labeled.
binding, converter, id, rendered, value	Basic attributes

Attributes for h:graphicImage



Attribute	Description
binding, id, rendered, styleClass, value	Basic attributes
alt, dir, height, ismap, lang, longdesc, style, title, url, usemap, width	HTML 4.0
onblur, onchange, onclick, ondblclick, onfocus, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onmouseup	DHTML events

Attributes for h:commandButton and h:commandLink

Attribute	Description
action	If specified as a string: Directly specifies an outcome used by the navigation handler to determine the JSF page to load next as a result of activating the button or link. If specified as a method binding: The method has this signature: <code>String methodName()</code> ; the string represents the outcome
actionListener	A method binding that refers to a method with this signature: <code>void methodName(ActionEvent)</code>
charset	For <code>h:commandLink</code> only—The character encoding of the linked reference
image	For <code>h:commandButton</code> only—A context-relative path to an image displayed in a button. If you specify this attribute, the HTML input's type will be image.
immediate	A boolean. If false (the default), actions and action listeners are invoked at the end of the request life cycle; if true, actions and action listeners are invoked at the beginning of the life cycle.

h:commandButton and h:commandLink, continued

Attribute	Description
type	For <code>h:commandButton</code> : The type of the generated input element: button, submit, or reset. The default, unless you specify the image attribute, is submit. For <code>h:commandLink</code> : The content type of the linked resource; for example, text/html, image/gif, or audio/basic
value	The label displayed by the button or link. You can specify a string or a value reference expression.
accesskey, alt, binding, id, lang, rendered, styleClass, value	Basic attributes
coords (h:commandLink only), dir, disabled (h:commandButton only), hreflang (h:commandLink only), lang, readonly, rel (h:commandLink only), rev (h:commandLink only), shape (h:commandLink only), style, tabindex, target (h:commandLink only), title, type	HTML 4.0
onblur, onchange, onclick, ondblclick, onfocus, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onmouseup, onselect	DHTML events



Attributes for h:outputLink

Attribute	Description
accesskey, binding, converter, id, lang, rendered, styleClass, value	Basic attributes
charset, coords, dir, hreflang, lang, rel, rev, shape, style, tabindex, target, title, type	HTML 4.0
onblur, onchange, onclick, ondblclick, onfocus, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onmouseup	DHTML events

Attributes for: h:selectBooleanCheckbox, h:selectManyCheckbox, h:selectOneRadio, h:selectOneListbox, h:selectManyListbox, h:selectOneMenu, h:selectManyMenu

Cheese Pickle Mustard Lettuce Onions

Attribute	Description
disabledClass	CSS class for disabled elements—for <code>h:selectOneRadio</code> and <code>h:selectManyCheckbox</code> only
enabledClass	CSS class for enabled elements—for <code>h:selectOneRadio</code> and <code>h:selectManyCheckbox</code> only
layout	Specification for how elements are laid out: <code>lineDirection</code> (horizontal) or <code>pageDirection</code> (vertical)—for <code>h:selectOneRadio</code> and <code>h:selectManyCheckbox</code> only
binding, converter, id, immediate, styleClass, required, rendered, validator, value, valueChangeListener	Basic attributes
accesskey, border, dir, disabled, lang, readonly, style, size, tabindex, title	HTML 4.0— <code>border</code> is applicable to <code>h:selectOneRadio</code> and <code>h:selectManyCheckbox</code> only. <code>size</code> is applicable to <code>h:selectOneListbox</code> and <code>h:selectManyListbox</code> only
onblur, onchange, onclick, ondblclick, onfocus, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onmouseup, onselect	DHTML events

Attributes for h:message and h:messages

Amount Amount: 'too much' is not a number.
Example: 99

Attribute	Description
for	The ID of the component whose message is displayed—applicable only to <code>h:message</code>
errorClass	CSS class applied to error messages
errorStyle	CSS style applied to error messages
fatalClass	CSS class applied to fatal messages
fatalStyle	CSS style applied to fatal messages
globalOnly	Instruction to display only global messages—applicable only to <code>h:messages</code> . Default: false
infoClass	CSS class applied to information messages
infoStyle	CSS style applied to information messages
layout	Specification for message layout: <code>table</code> or <code>list</code> —applicable only to <code>h:messages</code>
showDetail	A boolean that determines whether message details are shown. Defaults are false for <code>h:messages</code> , true for <code>h:message</code>
showSummary	A boolean that determines whether message summaries are shown. Defaults are true for <code>h:messages</code> , false for <code>h:message</code>
tooltip	A boolean that determines whether message details are rendered in a tooltip; the tooltip is only rendered if <code>showDetail</code> and <code>showSummary</code> are true
warnClass	CSS class for warning messages
warnStyle	CSS style for warning messages
binding, id, rendered, styleClass	Basic attributes
style, title	HTML 4.0

Attributes for h:panelGrid

Attribute	Description
bgcolor	Background color for the table
border	Width of the table's border
cellpadding	Padding around table cells
cellspacing	Spacing between table cells
columnClasses	Comma-separated list of CSS classes for columns
columns	Number of columns in the table
footerClass	CSS class for the table footer
frame	Specification for sides of the frame surrounding the table that are to be drawn; valid values: <code>none</code> , <code>above</code> , <code>below</code> , <code>hsides</code> , <code>vsides</code> , <code>lhs</code> , <code>rhs</code> , <code>box</code> , <code>border</code>
headerClass	CSS class for the table header
rowClasses	Comma-separated list of CSS classes for columns
rules	Specification for lines drawn between cells; valid values: <code>groups</code> , <code>rows</code> , <code>columns</code> , <code>all</code>
summary	Summary of the table's purpose and structure used for non-visual feedback such as speech
binding, id, rendered, styleClass, value	Basic attributes
dir, lang, style, title, width	HTML 4.0
onclick, ondblclick, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onmouseup	DHTML events

Attributes for h:panelGroup

Attribute	Description
binding, id, rendered, styleClass	Basic attributes
style	HTML 4.0

Attributes for h:dataTable

Attribute	Description
bgcolor	Background color for the table
border	Width of the table's border
cellpadding	Padding around table cells
cellspacing	Spacing between table cells
columnClasses	Comma-separated list of CSS classes for columns
first	Index of the first row shown in the table
footerClass	CSS class for the table footer
frame	Frame Specification for sides of the frame surrounding the table that are to be drawn; valid values: none, above, below, hsid, vsides, lhs, rhs, box, border
headerClass	CSS class for the table header
rowClasses	Comma-separated list of CSS classes for rows
rules	Specification for lines drawn between cells; valid values: groups, rows, columns, all
summary	Summary of the table's purpose and structure used for non-visual feedback such as speech

Attributes for h:dataTable, continued

Attribute	Description
var	The name of the variable created by the data table that represents the current item in the value
binding, id, rendered, styleClass, value	Basic attributes
dir, lang, style, title, width	HTML 4.0
onclick, ondblclick, onkeydown, onkeypress, onkeyup, onmousedown, onmousemove, onmouseout, onmouseover, onmouseup	DHTML events

Attributes for h:column

Attribute	Description
headerClass (JSF 1.2)	CSS class for the column's header
footerClass (JSF 1.2)	CSS class for the column's footer
binding, id, rendered	Basic attributes

ABOUT THE AUTHOR



Cay S. Horstmann

Cay S. Horstmann has written many books on C++, Java and object-oriented development, is the series editor for Core Books at Prentice-Hall and a frequent speaker at computer industry conferences. For four years, Cay was VP and CTO of an Internet startup that went from 3 people in a tiny office to a public company. He is now a computer science professor at San Jose State University. He was elected Java Champion in 2005.

List of recent books/publications

- Core Java, with Gary Cornell, Sun Microsystems Press 1996 - 2007 (8 editions)
- Core JavaServer Faces, with David Geary, Sun Microsystems Press, 2004-2006 (2 editions)
- Big Java, John Wiley & Sons 2001 - 2007 (3 editions)

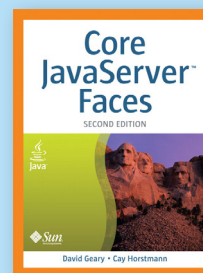
Blog

<http://weblogs.java.net/blog/cayhorstmann>

Web site

<http://horstmann.com>

RECOMMENDED BOOK



Core JavaServer Faces delves into all facets of JSF development, offering systematic best practices for building robust applications and maximizing developer productivity.

BUY NOW

books.dzone.com/books/jsf

Get More FREE Refcardz. Visit refcardz.com now!

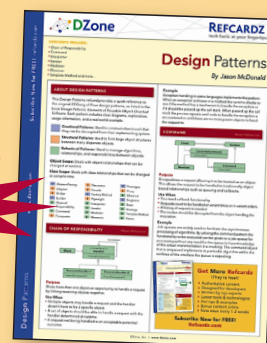
Upcoming Refcardz:

- Core Seam
- Core CSS: Part III
- Hibernate Search
- Equinox
- EMF
- XML
- JSP Expression Language
- ALM Best Practices
- HTML and XHTML

Available:

- Essential Ruby
- Essential MySQL
- JUnit and EasyMock
- Getting Started with MyEclipse
- Spring Annotations
- Core Java
- Core CSS: Part II
- PHP
- Getting Started with JPA
- JavaServer Faces
- Essential CSS: Part I
- Struts2
- Core .NET
- Very First Steps in Flex
- C#
- Groovy
- NetBeans IDE 6.1 Java Editor
- RSS and Atom
- GlassFish Application Server
- Silverlight 2

Visit refcardz.com for a complete listing of available Refcardz.



Design Patterns
Published June 2008



DZone communities deliver over 4 million pages each month to more than 1.7 million software developers, architects and decision makers. DZone offers something for everyone, including news, tutorials, cheatsheets, blogs, feature articles, source code and more. "DZone is a developer's dream," says PC Magazine.

DZone, Inc.
1251 NW Maynard
Cary, NC 27513

888.678.0399
919.678.0300

Refcardz Feedback Welcome
refcardz@dzone.com

Sponsorship Opportunities
sales@dzone.com

ISBN-13: 978-1-934238-19-6
ISBN-10: 1-934238-19-8



9 781934 238196

\$7.95