



Jakarta Struts: Using Properties Files (Resource Bundles) Struts 1.2 Version

Core Servlets & JSP book: www.coreservlets.com
More Servlets & JSP book: www.moreservlets.com
Servlet/JSP/Struts/JSF Training: courses.coreservlets.com

Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press



For live Struts training, please see
JSP/servlet/Struts/JSF training courses at
<http://courses.coreservlets.com/>.



Taught by the author of *Core Servlets and JSP*, *More Servlets and JSP*, and this tutorial. Available at public venues, or customized versions can be held on-site at your organization.

Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press

Agenda

- **Three new ideas**
 - Automatically created bean representing request data
 - Using bean:write to output bean properties
 - Using bean:message to output constant strings
- **Defining form beans**
- **Declaring form beans**
- **Outputting properties of form beans**
 - bean:write
 - JSP 2.0 expression language
- **Defining and outputting regular beans**
- **Using properties files**
 - To reuse fixed strings
 - To support internationalization

5

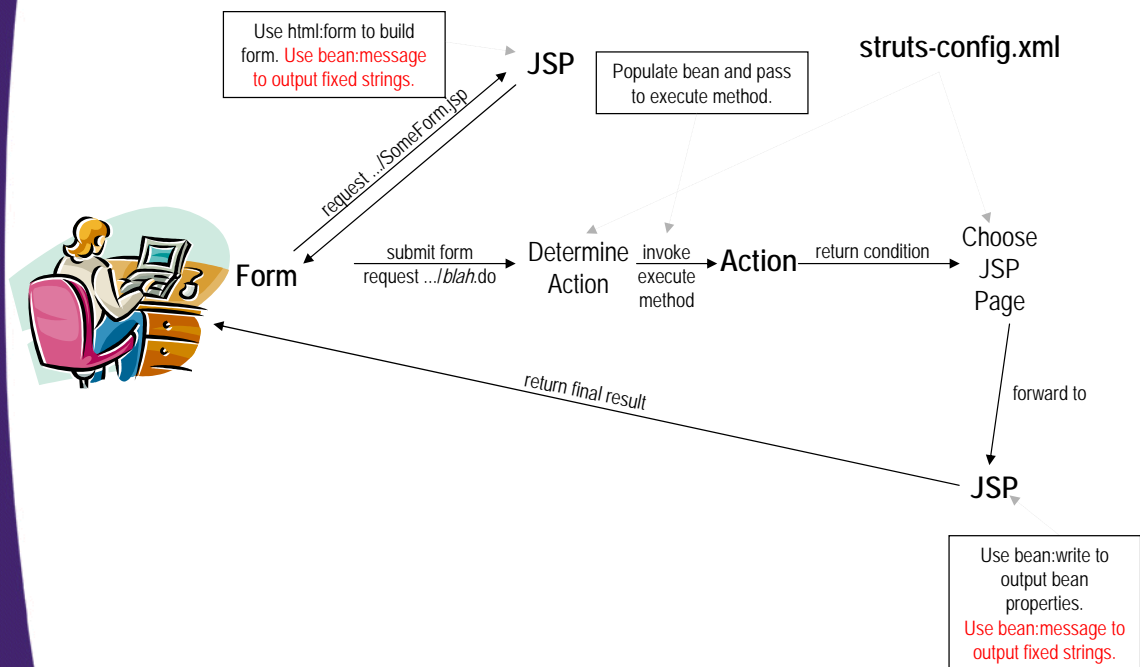
Apache Struts: Messages and Properties Files

www.coreservlets.com



Using Properties Files (Resource Bundles)

Struts Flow of Control



7

Apache Struts: Messages and Properties Files

www.coreservlets.com

New Steps

- 1. Create a properties file in WEB-INF/classes**
 - E.g., WEB-INF/classes/MessageResources.properties
- 2. Define strings in the properties file**
 - some.key1=first message
 - some.key2=second message
 - some.key3=some parameterized message: {0}
- 3. Load the properties file in struts-config.xml**
 - `<message-resources parameter="MessageResources"/>`
- 4. Output the messages in JSP pages**
 - Load the tag library
 - `<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>`
 - Output the messages using `bean:message`
 - First message is `<bean:message key="some.key1"/>`
 - Second: `<bean:message key="some.key2"/>`
 - Third: `<bean:message key="some.key3" arg0="replacement"/>`

8

Apache Struts: Messages and Properties Files

www.coreservlets.com

Advantages of Properties Files

- **Centralized updates**
 - If a message is used in several places, it can be updated with a single change.
 - This is consistent with the Struts philosophy of making as many changes as possible in config files, not in Java or JSP code.
- **I18N**
 - If you use messages pervasively in your JSP pages, you can internationalize your application by having multiple properties files corresponding to the locale, as with standard I18N in Java
 - MessageResources.properties
 - MessageResources_jp.properties
 - MessageResources_es.properties
 - MessageResources_es_mx.properties

9

Apache Struts: Messages and Properties Files

www.coreservlets.com

Example 1: Simple Messages

- **URL**
 - `http://hostname/struts-messages/actions/register.do`
- **Action Class**
 - RegistrationAction
 - Simply returns "success" in all situations
- **Results page**
 - Single page
 - `/WEB-INF/results/confirm-registration.jsp`

10

Apache Struts: Messages and Properties Files

www.coreservlets.com

Step 1 (Modify struts-config.xml)

- **Map incoming .do address to Action classes**
 - As before, we use the action element (to designate that RegistrationAction should handle requests for /actions/register.do).
- **Map return conditions to JSP pages**
 - As before, we use the forward element
- **Declare any form beans that are being used.**
 - As before, we use a form-bean entry with name and type attributes
- **Declare a properties file**
 - The message-resources element is used to refer to a properties file:
`<message-resources parameter="MessageResources" null="false"/>`
 - The parameter attribute refers to the location of the properties file
 - Relative to WEB-INF/classes and with the .properties file extension implied.
 - E.g., "MessageResources" refers to WEB-INF/classes/MessageResources.properties, and "foo.bar.baz" refers to WEB-INF/classes/foo/bar/baz.properties.
 - The null attribute determines whether missing messages should be flagged. If the value is true, references to nonexistent messages result in null. If the value is false, references to nonexistent messages result in warning messages like ???keyName???

Step 1 (Modify struts-config.xml) -- Final Code

```
...
<struts-config>
  <form-beans>
    <form-bean name="registrationBean"
              type="coreservlets.RegistrationFormBean"/>
  </form-beans>
  <action-mappings>
    <action path="/actions/register"
           type="coreservlets.RegistrationAction"
           name="registrationBean"
           scope="request">
      <forward name="success"
              path="/WEB-INF/results/confirm-registration.jsp"/>
    </action>
  </action-mappings>
  <message-resources parameter="MessageResources"
                    null="false"/>
</struts-config>
```

Step 1 (Properties File)

- **WEB-INF/classes/MessageResources.properties**

```
form.title=Registration
form.firstName=First name
form.lastName=Last name
form.emailAddress=Email address
form.buttonLabel=Register Me
form.successString=Success
```

Step 2 (Define a Form Bean)

```
package coreservlets;
import org.apache.struts.action.*;

public class RegistrationFormBean extends ActionForm {
    private String firstName, lastName, emailAddress;

    public String getFirstName() {
        return(firstName);
    }

    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }

    ...
}
```

Step 3 (Create Results Beans)

- Omitted in this simple example

Step 4 (Define an Action Class to Handle Requests)

```
package coreservlets;

import javax.servlet.http.*;
import org.apache.struts.action.*;

public class RegistrationAction extends Action {
    public ActionForward execute(ActionMapping mapping,
                                ActionForm form,
                                HttpServletRequest request,
                                HttpServletResponse response)
        throws Exception {
        return(mapping.findForward("success"));
    }
}
```

Step 5 (Create Form That Invokes *blah.do*)

- Instead of directly listing headings, prompts, and textual messages, they are taken from properties file
 - That way, if the messages change (or if you have multiple versions in different languages), the messages can be updated without modifying the actual JSP pages.
 - Also, some of the prompts are used in more than one page, so extracting the prompts from the properties file limits changes to one location, even though the prompts are used in multiple locations.

Step 5 (Create Form That Invokes *blah.do*)

```
<%@ taglib ... prefix="html" %>
<%@ taglib ... prefix="bean" %>
<!DOCTYPE ...>
<HTML>
<HEAD><TITLE><bean:message key="form.title"/></TITLE></HEAD>
<BODY BGCOLOR="#FDF5E6">
<CENTER>
<H1><bean:message key="form.title"/></H1>
<CENTER>
<html:form action="/actions/register">
  <bean:message key="form.firstName"/>:
  <html:text property="firstName"/><BR>
  <bean:message key="form.lastName"/>:
  <html:text property="lastName"/><BR>
  <bean:message key="form.emailAddress"/>:
  <html:text property="emailAddress"/><BR>
  <html:submit>
    <bean:message key="form.buttonLabel"/>
  </html:submit>
</html:form>
</CENTER>
</BODY></HTML>
```

Step 6 (Display Results in JSP)

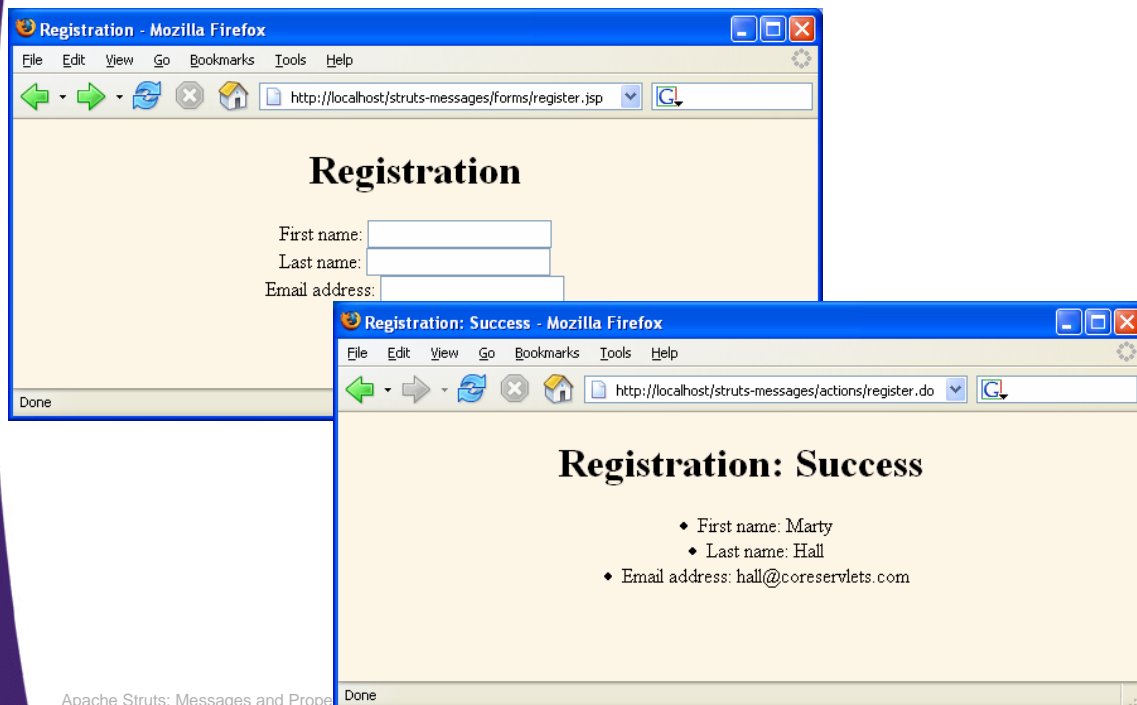
```
<%@ taglib ... prefix="bean" %>
<!DOCTYPE ...>
<HTML>
<HEAD><TITLE>
<bean:message key="form.title"/>:
<bean:message key="form.successString"/>
</TITLE></HEAD>
<BODY BGCOLOR="#FDF5E6"><CENTER>
<H1>
<bean:message key="form.title"/>:
<bean:message key="form.successString"/>
</H1>
<UL>
  <LI><bean:message key="form.firstName"/>:
    <bean:write name="registrationBean"
      property="firstName"/>
  <LI><bean:message key="form.lastName"/>:
    <bean:write name="registrationBean"
      property="lastName"/>
  <LI><bean:message key="form.emailAddress"/>:
    <bean:write name="registrationBean"
      property="emailAddress"/>
</UL>
</CENTER></BODY></HTML>
```

19

Apache Struts: Messages and Properties Files

www.coreservlets.com

Example 1: Results



20

Apache Struts: Messages and Properties Files

Done



Internationalization (I18N)

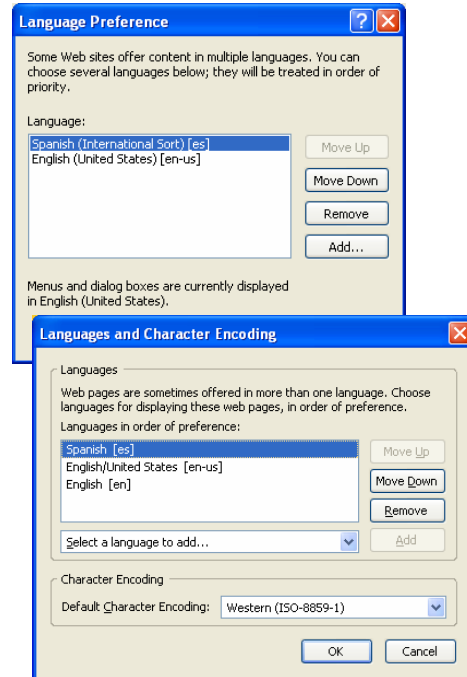
Slides © Marty Hall, <http://www.coreservlets.com>, books © Sun Microsystems Press

Loading Locale-Specific Properties Files

- **Default properties file**
 - When you say
`<message-resources parameter="someName" ...>`
WEB-INF/classes/someName.properties is loaded and treated as the default file
- **More specific properties file**
 - The system automatically looks for additional, specialized files corresponding to your Locale
 - someName_es.properties, someName_es_mx.properties, etc.
 - Entries from more specific file override entries from default file
 - Locale is automatically determined by browser language settings
 - Locale can also be set explicitly (e.g., based on incoming checkbox value) in an Action with setLocale

Setting Language Preferences in Browsers

- **Internet Explorer**
 - Tools, Internet Options, Languages
 - Click Add, select language, OK
 - Move to top of list using "Move Up"
- **Firefox**
 - Tools, Options, Languages
 - Click Add, select language, Add
 - Move to top of list using "Move Up"



23

Apache Struts: Messages and Properties Files

www.coreservlets.com

Internationalizing the Registration Code for English, Spanish, French

- **Need MessageResources_es.properties and MessageResources_fr.properties**
 - This is *all* that is needed.
 - No changes to any config file or code!
- **MessageResources_es.properties**

```
form.title=Registro
form.firstName=Primer nombre
form.lastName=Apellido
form.emailAddress=Dirección de email
form.buttonLabel=Coloquéme
form.successString=Éxito
```

24

Apache Struts: Messages and Properties Files

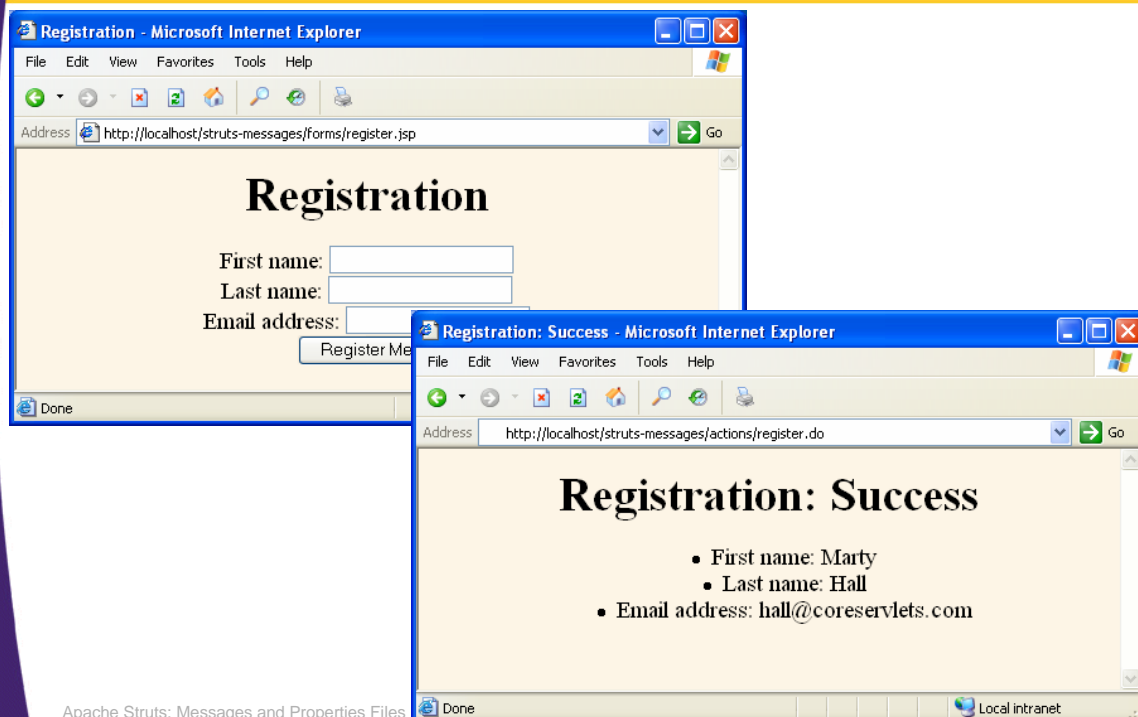
www.coreservlets.com

Internationalizing the Registration Code for English, Spanish, French

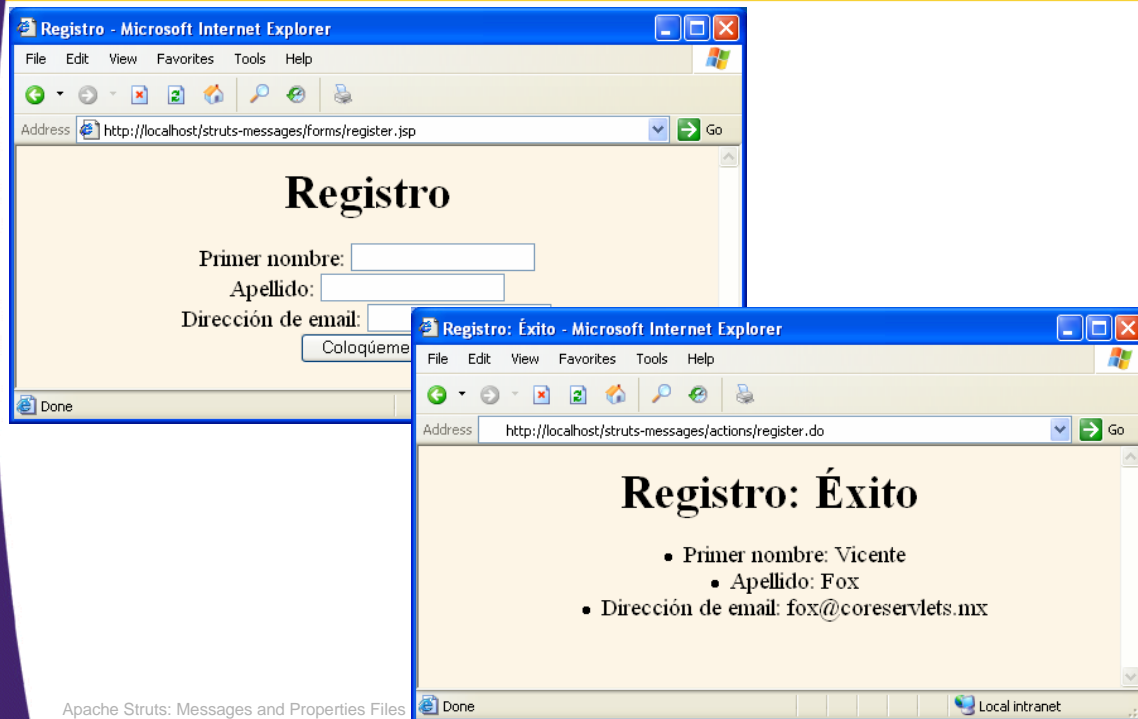
- **MessageResources_fr.properties**

```
form.title=Enregistrement  
form.firstName=Prénom  
form.lastName=Nom  
form.emailAddress=Adresse électronique  
form.buttonLabel=Enregistrez Moi  
form.successString=Succès
```

Example 2: Results (Browser with English as First Language)



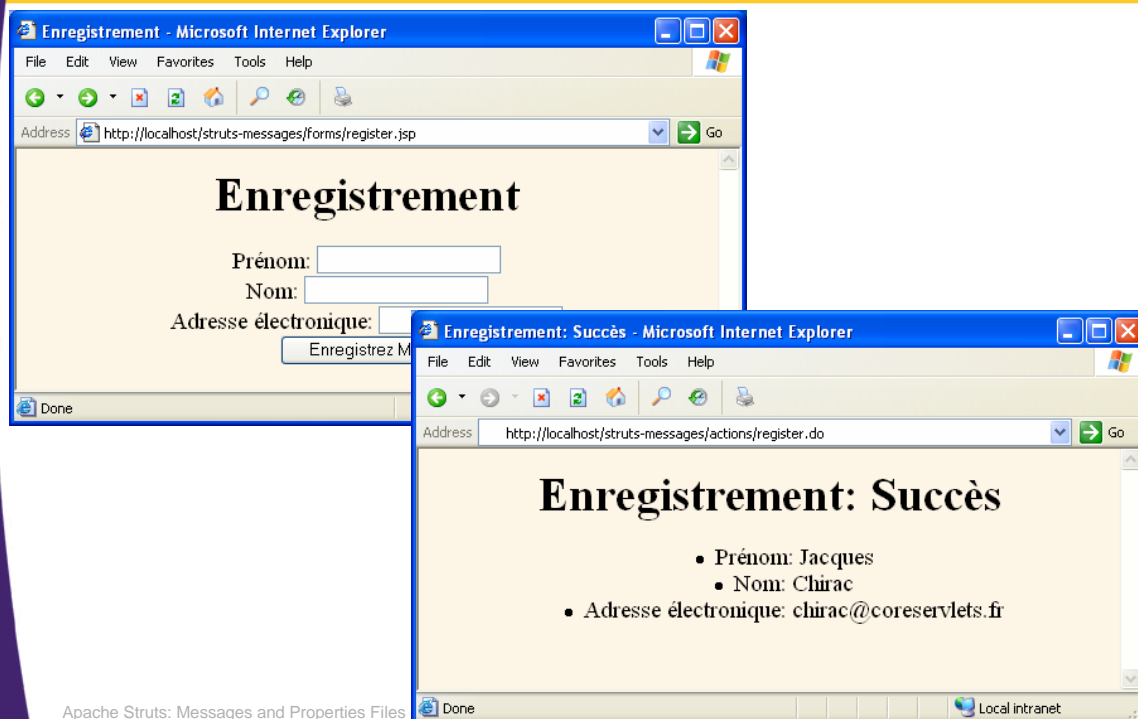
Example 2: Results (Browser with Spanish as First Language)



27

Apache Struts: Messages and Properties Files

Example 2: Results (Browser with French as First Language)



28

Apache Struts: Messages and Properties Files

Parameterized Messages

- **Properties File**

- error.missing=You must enter {0}, you idiot!
- error.number={0} and {1} are not whole numbers!

- **JSP Pages**

- `<bean:message key="error.missing" arg0="your first name"/>`
- `<bean:message key="error.number" arg0="<%= Math.random() %>" arg1="{user.firstName}"/>`

- **Problem**

- Not simple to take the argN values from resource bundle (properties file), so does not work well for I18N

- **Examples and details**

- In section on automatic validation

29

Apache Struts: Messages and Properties Files

www.coreservlets.com

Dynamic Keys

- **The key name can be dynamically computed**

- Convenient for making mappings that can be put into text files and can be easily internationalized

- **Example**

- Properties File
 - jhu=Johns Hopkins University
 - upenn=University of Pennsylvania
 - umbc=University of Maryland Baltimore County
- JSP
 - `<bean:message key="{user.school}"/>`
 - `<bean:message key="<%= school %>"/>`

30

Apache Struts: Messages and Properties Files

www.coreservlets.com

Summary

- **Create a properties file in WEB-INF/classes**
 - E.g., WEB-INF/classes/MessageResources.properties
- **Define strings in the properties file**
 - some.key=some message
- **Load the properties file in struts-config.xml**
 - `<message-resources parameter="MessageResources"/>`
- **Output the messages in JSP pages**
 - `<bean:message key="some.key"/>`
- **Internationalize by creating multiple properties files**
 - MessageResources_xx.properties
 - Use browser settings or setLocale to determine language



Questions?

Core Servlets & JSP book: www.coreservlets.com
More Servlets & JSP book: www.moreservlets.com
Servlet, JSP, Struts, JSF, and Java Training Courses:
courses.coreservlets.com