



Using Properties Files (Resource Bundles) in JSF

Originals of Slides and Source Code for Examples:

<http://www.coreservlets.com/JSF-Tutorial/>

This somewhat old tutorial covers JSF 1, and is left online for those maintaining existing projects. All **new** projects should use JSF 2, which is both simpler and more powerful. See <http://www.coreservlets.com/JSF-Tutorial/jsf2/>.

Customized Java EE Training: <http://courses.coreservlets.com/>

Java, JSF 2, PrimeFaces, Servlets, JSP, Ajax, jQuery, Spring, Hibernate, RESTful Web Services, Hadoop, Android. Developed and taught by well-known author and developer. At public venues or onsite at *your* location.



For live training on JSF 1 or 2, please see 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.

- Courses developed and taught by Marty Hall
 - JSF 2, PrimeFaces, servlets/JSP, Ajax, jQuery, Android development, Java 6 or 7 programming, custom mix of topics
 - Ajax courses can concentrate on 1 library (jQuery, Prototype/Scriptaculous, Ext-JS, Dojo, etc.) or survey several
 - Courses developed and taught by coreservlets.com experts (edited by Marty)
 - Spring, Hibernate/JPA, EJB3, GWT, Hadoop, SOAP-based and RESTful Web Services
- Contact hall@coreservlets.com for details**

Agenda

- Loading properties files
- Simple messages
- Parameterized messages
- Internationalized messages

5

© 2012 Marty Hall



Simple Messages

Customized Java EE Training: <http://courses.coreservlets.com/>

Java, JSF 2, PrimeFaces, Servlets, JSP, Ajax, jQuery, Spring, Hibernate, RESTful Web Services, Hadoop, Android.
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

6

Displaying Fixed Strings

1. Create a .properties file

- Contains simple keyName=value pairs
- Must be deployed to WEB-INF/classes
 - In Eclipse, this means you put it in "src" folder

2. Load file with f:loadBundle

- basename gives base file name
- var gives scoped variable (Map) that will hold results
 - Relative to WEB-INF/classes, .properties assumed
 - E.g., for WEB-INF/classes/messages.properties
<f:loadBundle basename="messages" var="msgs"/>
 - E.g., for WEB-INF/classes/package1/test.properties
<f:loadBundle basename="package1.test" var="msgs"/>

3. Output messages using normal EL

- `#{msgs.keyName}`

7

WEB-INF/classes/ messages1.properties

```
title=Registration
text=Please enter your first name, last name,
    and email address.  (all on one line)
firstNamePrompt=Enter first name
lastNamePrompt=Enter last name
emailAddressPrompt=Enter email address
buttonLabel=Register Me
```

8

signup1.jsp (.faces)

```
<%@ taglib uri="http://java.sun.com/jsf/core" prefix="f" %>
<%@ taglib uri="http://java.sun.com/jsf/html" prefix="h" %>
<f:view>
  <f:loadBundle basename="messages1" var="msgs"/>
  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
  <HTML>
  <HEAD><TITLE>
    <h:outputText value="#{msgs.title}"/>
  </TITLE>
  <LINK REL="STYLESHEET"
        HREF="./css/styles.css"
        TYPE="text/css">
  </HEAD>
  <BODY>
  <CENTER>
  <TABLE BORDER=5>
    <TR><TH CLASS="TITLE">
      <h:outputText value="#{msgs.title}"/></TH></TR>
  </TABLE>
  <BR>
  <h:outputText value="#{msgs.text}"/>
  <P>
```

9

signup1.jsp (.faces), Cont.

```
<h:form>
  <h:outputText value="#{msgs.firstNamePrompt}"/>:
  <h:inputText value="#{person.firstName}"/>
  <BR>
  <h:outputText value="#{msgs.lastNamePrompt}"/>:
  <h:inputText value="#{person.lastName}"/>
  <BR>
  <h:outputText value="#{msgs.emailAddressPrompt}"/>:
  <h:inputText value="#{person.emailAddress}"/>
  <BR>
  <h:commandButton
    value="#{msgs.buttonLabel}"
    action="#{person.doRegistration}"/>
</h:form>
</CENTER></BODY></HTML>
</f:view>
```

10

signup1.faces: Result

Registration - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://localhost/jsf-messages/signup1.faces

Registration

Please enter your first name, last name, and email address.

Enter first name:

Enter last name:

Enter email address:

Register Me

Done

11

© 2012 Marty Hall



Parameterized Messages

Customized Java EE Training: <http://courses.coreservlets.com/>

Java, JSF 2, PrimeFaces, Servlets, JSP, Ajax, jQuery, Spring, Hibernate, RESTful Web Services, Hadoop, Android.
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

12

Parameterizing Strings

1. Create a .properties file in/under WEB-INF/classes

- Values contain {0}, {1}, {2}, etc.
- E.g., someName=blah {0} blah {1}
- Warning: MyFaces bug prevents single quotes in values

2. Load file with f:loadBundle as before

- basename gives base file name
- var gives scoped variable (Map) that will hold results

3. Output messages using h:outputFormat

- value gives base message
- nested f:param gives substitution values

E.g.:

```
<h:outputFormat value="#{msgs.someName}">
  <f:param value="value for 0th entry"/>
  <f:param value="value for 1st entry"/>
</h:outputFormat>
```

13

messages2.properties

```
title=Registration
firstName=first name
lastName=last name
emailAddress=email address
text=Please enter your {0}, {1}, and {2}.
prompt=Enter {0}
buttonLabel=Register Me
```

14

signup2.jsp (.faces)

```
<%@ taglib uri="http://java.sun.com/jsf/core" prefix="f" %>
<%@ taglib uri="http://java.sun.com/jsf/html" prefix="h" %>
<f:view locale="#{facesContext.externalContext.requestLocale}">
<f:loadBundle basename="messages2" var="msgs"/>
...
<TABLE BORDER=5>
  <TR><TH CLASS="TITLE">
    <h:outputText value="#{msgs.title}"/></TH></TR>
</TABLE>
<BR>
<h:outputFormat value="#{msgs.text}">
  <f:param value="#{msgs.firstName}"/>
  <f:param value="#{msgs.lastName}"/>
  <f:param value="#{msgs.emailAddress}"/>
</h:outputFormat>
<P>
<h:form>
<h:outputFormat value="#{msgs.prompt}">
  <f:param value="#{msgs.firstName}"/>
</h:outputFormat>:
<h:inputText value="#{person.firstName}"/>
```

15

signup2.jsp (.faces), Cont.

```
<BR>
<h:outputFormat value="#{msgs.prompt}">
  <f:param value="#{msgs.lastName}"/>
</h:outputFormat>:
<h:inputText value="#{person.lastName}"/>
<BR>
<h:outputFormat value="#{msgs.prompt}">
  <f:param value="#{msgs.emailAddress}"/>
</h:outputFormat>:
<h:inputText value="#{person.emailAddress}"/>
<BR>
<h:commandButton
  value="#{msgs.buttonLabel}"
  action="#{person.doRegistration}"/>
</h:form>
</CENTER></BODY></HTML>
</f:view>
```

16

signup2.faces: Result



17

© 2012 Marty Hall



Internationalized Messages

Customized Java EE Training: <http://courses.coreservlets.com/>

Java, JSF 2, PrimeFaces, Servlets, JSP, Ajax, jQuery, Spring, Hibernate, RESTful Web Services, Hadoop, Android.
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

18

Localizing Strings

1. Create multiple similarly named .properties files

- blah.properties, blah_es.properties, blah_es_mx.properties

2. Supply locale argument to f:view

```
<f:view locale="#{facesContext.externalContext.requestLocale}">
```

- Determines locale from browser language settings
- Can also set the Locale based on user input
 - locale="#{settings.selectedLocale}"
 - See event-handler section for best approach

3. Load file with f:loadBundle as before

- basename gives base file name
 - Version matching Locale will be used automatically!
- var gives scoped variable (Map) that will hold results

4. Output messages using h:outputFormat or h:outputText

- Same as before

19

messages2.properties

```
title=Registration
firstName=first name
lastName=last name
emailAddress=email address
text=Please enter your {0}, {1}, and {2}.
prompt=Enter {0}
buttonLabel=Register Me
```

20

messages2_es.properties

```
title=Registro  
firstName=primer nombre  
lastName=apellido  
emailAddress=dirección de email  
text=Incorpore por favor su {0}, {1}, y {2}.  
prompt=Incorpore {0}  
buttonLabel=Coloquéme
```

messages2_fr.properties

```
title=Enregistrement  
firstName=prénom  
lastName=nom  
emailAddress=adresse électronique  
text=Merci de entrer votre {0}, {1}, et {2}.  
prompt=Entrez votre {0}  
buttonLabel=Enregistrez moi
```

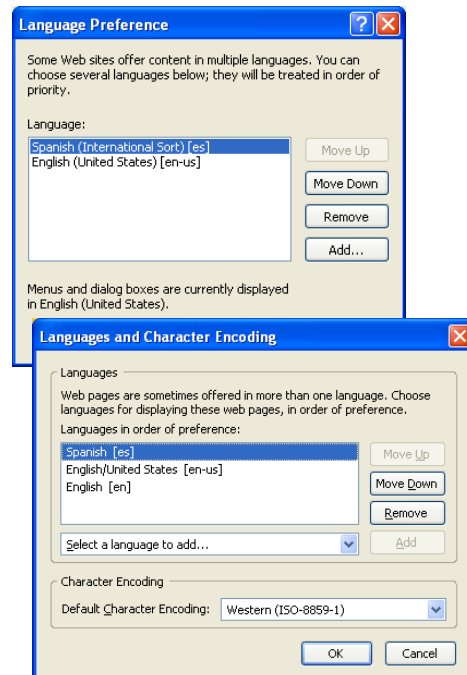
signup2.jsp (.faces)

```
<%@ taglib uri="http://java.sun.com/jsf/core" prefix="f" %>
<%@ taglib uri="http://java.sun.com/jsf/html" prefix="h" %>
<f:view locale="#{facesContext.externalContext.requestLocale}">
<f:loadBundle basename="messages2" var="msgs"/>
...
<TABLE BORDER=5>
  <TR><TH CLASS="TITLE">
    <h:outputText value="#{msgs.title}"/></TH></TR>
</TABLE>
<BR>
<h:outputFormat value="#{msgs.text}">
  <f:param value="#{msgs.firstName}"/>
  <f:param value="#{msgs.lastName}"/>
  <f:param value="#{msgs.emailAddress}"/>
</h:outputFormat>
<P>
<h:form>
<h:outputFormat value="#{msgs.prompt}">
  <f:param value="#{msgs.firstName}"/>
</h:outputFormat>:
<h:inputText value="#{person.firstName}"/>
...
```

23

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, Advanced, General, Languages
 - Click Add, select language, Add
 - Move to top of list using “Move Up”



24

signup2.faces: Result (Browser Language English)



The screenshot shows a Mozilla Firefox browser window titled "Registration - Mozilla Firefox". The address bar contains the URL "http://localhost/jsf-messages/signup2.faces". The page content includes a heading "Registration" in a stylized box, followed by the instruction "Please enter your first name, last name, and email address." Below this are three input fields: "Enter first name:", "Enter last name:", and "Enter email address:". A "Register Me" button is positioned below the email field. The status bar at the bottom of the browser window displays "Done".

25

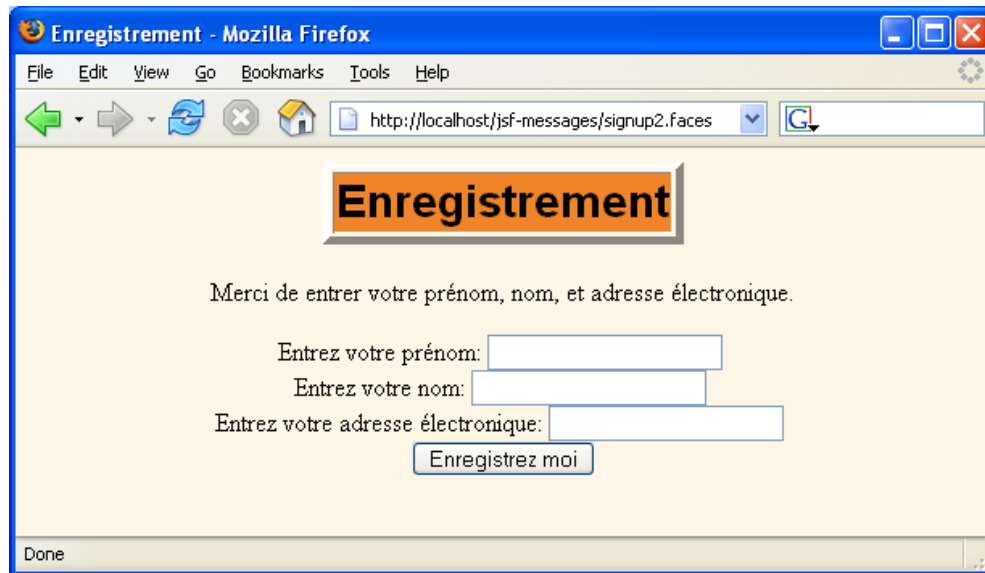
signup2.faces: Result (Browser Language Spanish)



The screenshot shows a Mozilla Firefox browser window titled "Registro - Mozilla Firefox". The address bar contains the URL "http://localhost/jsf-messages/signup2.faces". The page content includes a heading "Registro" in a stylized box, followed by the instruction "Incorpore por favor su primer nombre, apellido, y dirección de email." Below this are three input fields: "Incorpore primer nombre:", "Incorpore apellido:", and "Incorpore dirección de email:". A "Coloquéme" button is positioned below the email field. The status bar at the bottom of the browser window displays "Done".

26

signup2.faces: Result (Browser Language French)



27

Summary

- **Deploy one or more .properties file to WEB-INF/classes**
 - In Eclipse, you put .properties file in “src” folder, and it gets deployed to WEB-INF/classes automatically
- **Load the file with f:loadBundle**
- **Output values using normal EL**
 - h:outputText (and a few others) for simple values
 - h:outputFormat for parameterized values
- **Set view’s locale if I18N needed**
 - Extract it from browser setting or user setting
 - Automatically loads locale-specific resource bundle

28



Questions?

Customized Java EE Training: <http://courses.coreservlets.com/>

Java, JSF 2, PrimeFaces, Servlets, JSP, Ajax, jQuery, Spring, Hibernate, RESTful Web Services, Hadoop, Android.
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.