

# Internationalization in Web Module JSP Applications



## Applies to:

SAP Portal 7.0+. For more information, visit the [Web Dynpro Java homepage](#).

## Summary

The article describes the concept and procedure of developing a Multilanguage capable JSP application

**Author:** Nilesh Rane

**Company:** Larsen & Toubro Infotech Limited

**Created on:** 15 March 2010

## Author Bio



Nilesh Rane is a SAP NetWeaver Consultant in Larsen & Toubro Infotech Limited. He has three years of SAP Portal experience. He has worked extensively in WebDynpro - Java, EPortal, J2EE, SOAP Web Services, Adobe flex & Java Dictionary

## Table of Contents

Concepts .....	3
Internationalization - Purpose .....	3
Internationalization in JSP .....	3
Translation Elements .....	3
Properties files .....	3
ResourceBundles.....	4
Developing a Multi-language Enabled Web Module Project .....	4
Summary.....	17
Related Content.....	18
Disclaimer and Liability Notice.....	19

## Concepts

### Internationalization - Purpose

The goal of Internationalization is to provide the technical foundation to enable programs to support multiple scripts and languages without redesign or modification. Once a user has selected a log-in language, all programs must transparently alter their run-time behavior to meet the expectations of the user. Internationalization furthers generic programming, because hard-encoded search strings, error messages, etc. may work in one language, but not in another.

- The purpose of internationalization is to adapt existing applications so that they can be used in several languages and countries (or regions) without the need for changes to the source code.
- Using the internationalization tools, you can store these text strings separately from the application source code, in a format that can be processed in the standard translation system.
- There should be no Hard Coding of text in the code
- Translations should be maintained for the desired languages<sup>1</sup>.

### Internationalization in JSP

To enable the JSP application to be displayed in the user log on language we need to apply the concept of internationalization.

- **Without Internationalization:**

JSP application is created in logon language. When the application is launched, it gets displayed in the same language.

For instance, a JSP is created in English. When user login language is English, user will see the entire application in English. If the user login language is German, even then user will see the entire application in English.

- **With Internationalization:**

To let the user see the application based on his/her logon language, language translation should be maintained for UI elements.

For instance, the JSP is created in English. Also, the translations are maintained for GERMAN language.

When user login language is English, user will see the entire application in English.

If the user login language is German, user will see the entire application in German.

If the user login language is Chinese, user will see the entire application in English, since no translations are maintained for it and in such case the application gets displayed in Default language.

### Translation Elements

The translations will be maintained for following objects.

1. UI elements like Labels, Text Fields, Titles, Table Column headers.

### Properties files

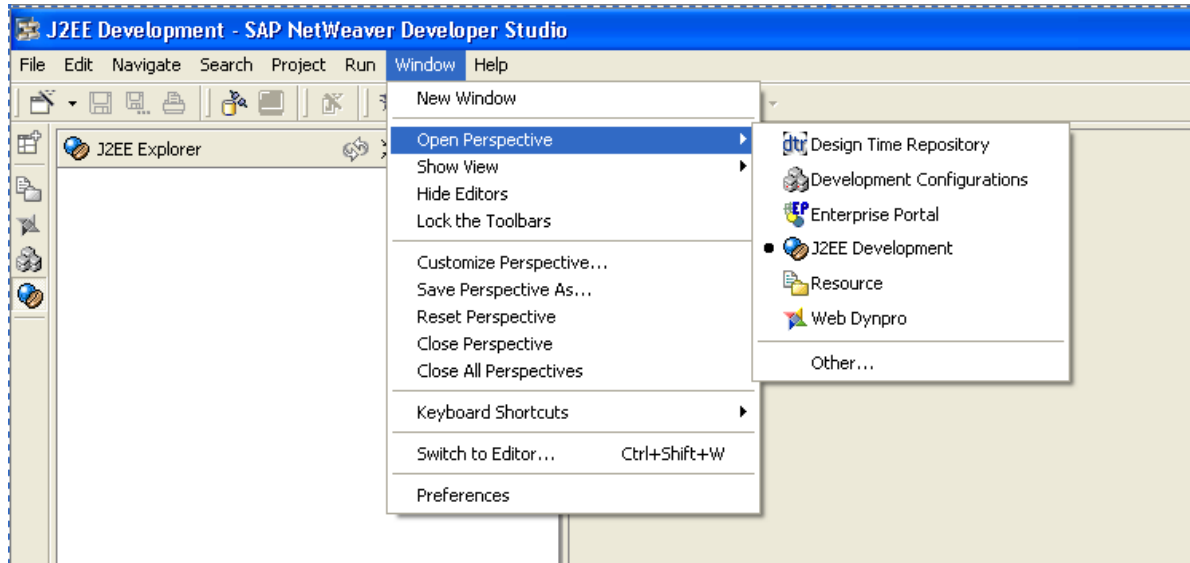
- ◆ JSP applications can be made language independent using properties file
- ◆ It is a central storage location for texts. The main task of the properties files is to support the translation of texts

## ResourceBundles

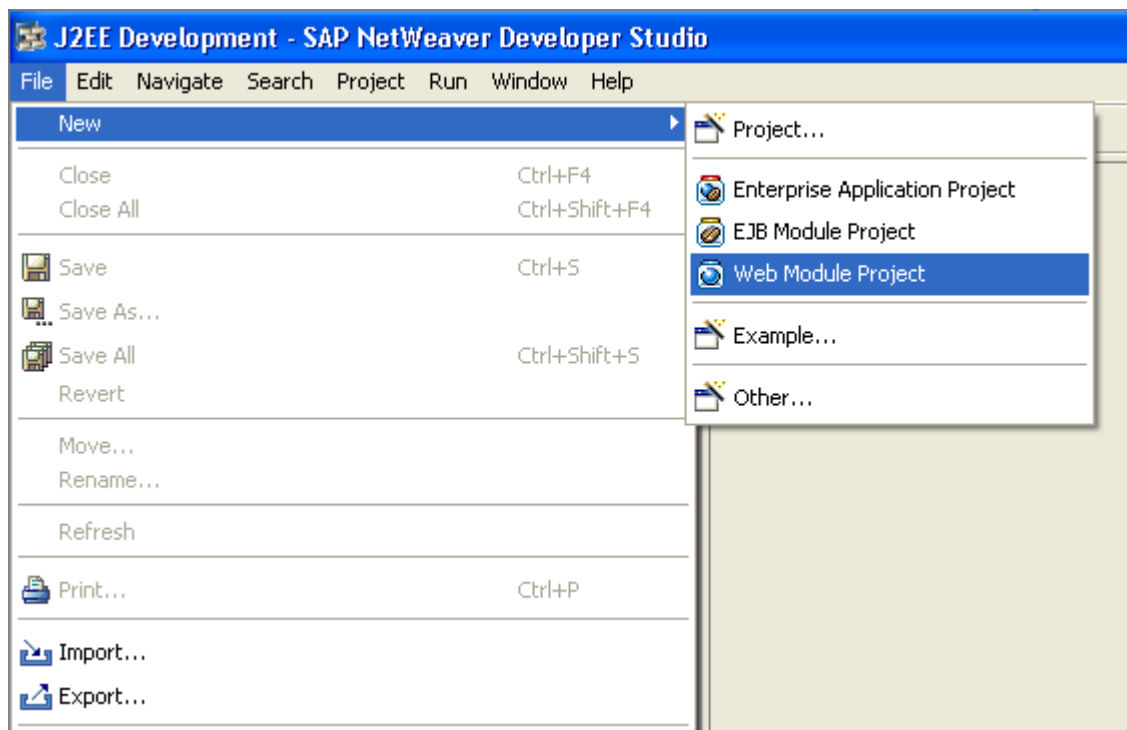
- ◆ The most obvious type of localizable object is a text string, but resource bundles can store any Java object. Whatever you choose to store, you'll put it into some form of `java.util.ResourceBundle`<sup>2</sup>

## Developing a Multi-language Enabled Web Module Project

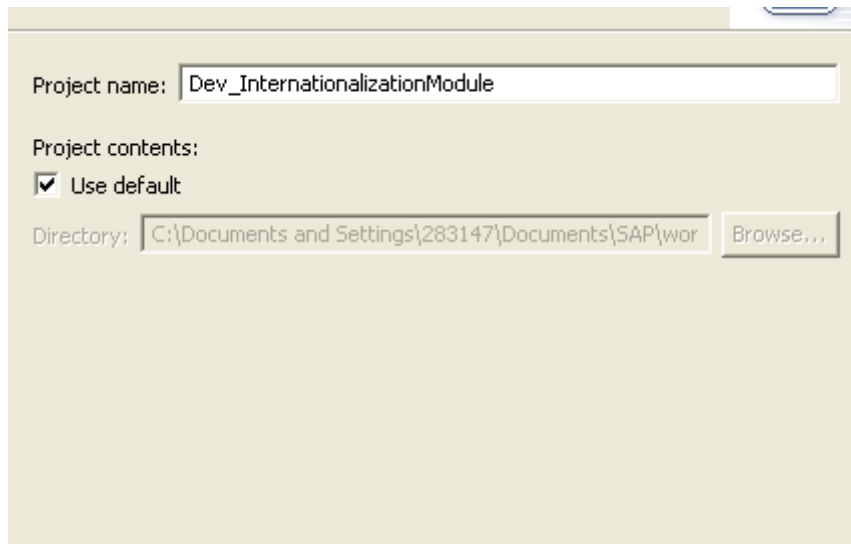
- ◆ Open NWDS in J2EE Development Perspective



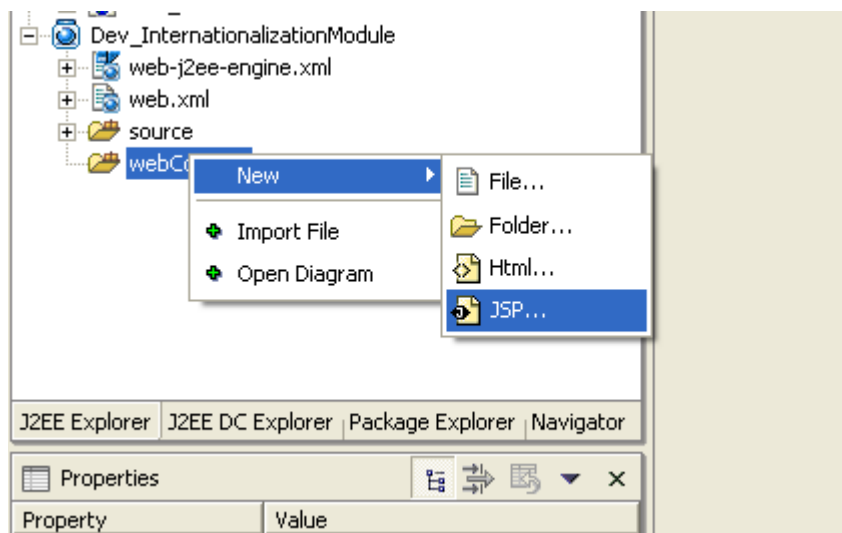
Select File -> New -> Web Module Project



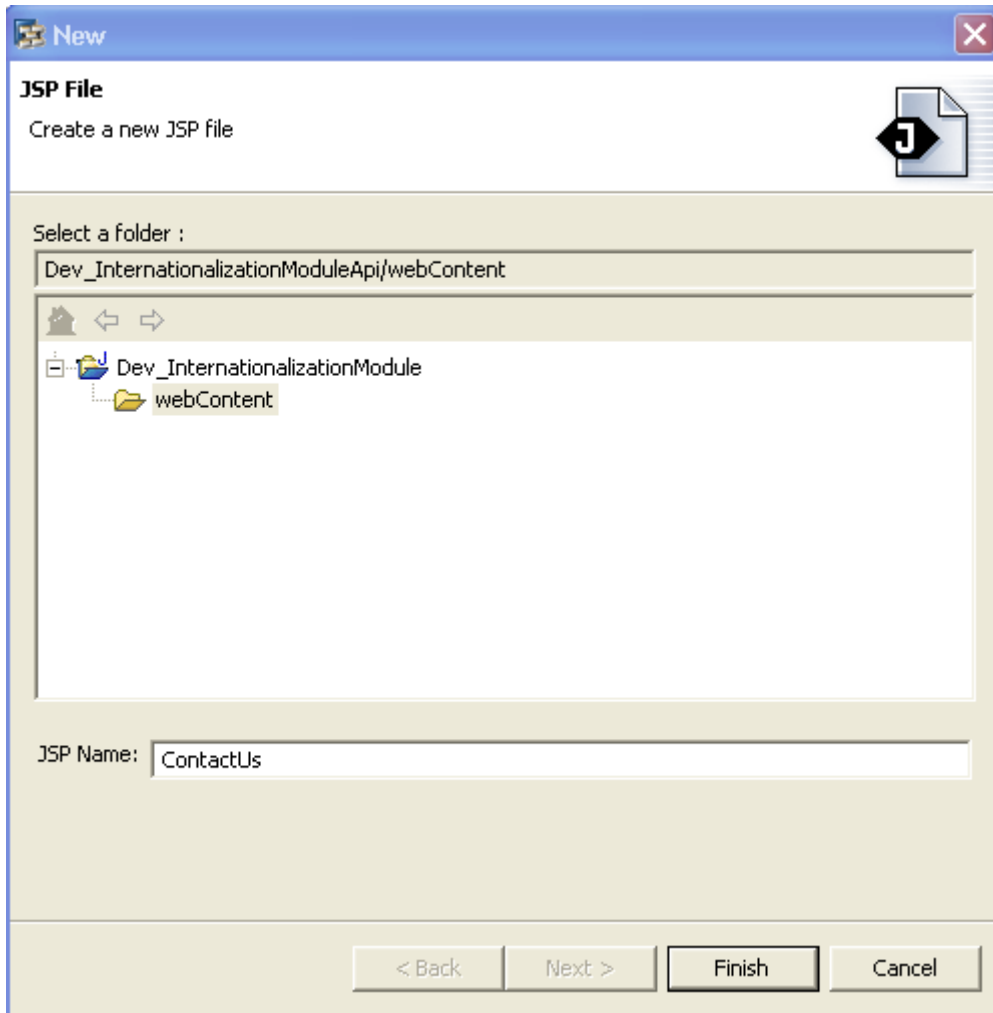
- ◆ Give any meaningful name and click on finish.



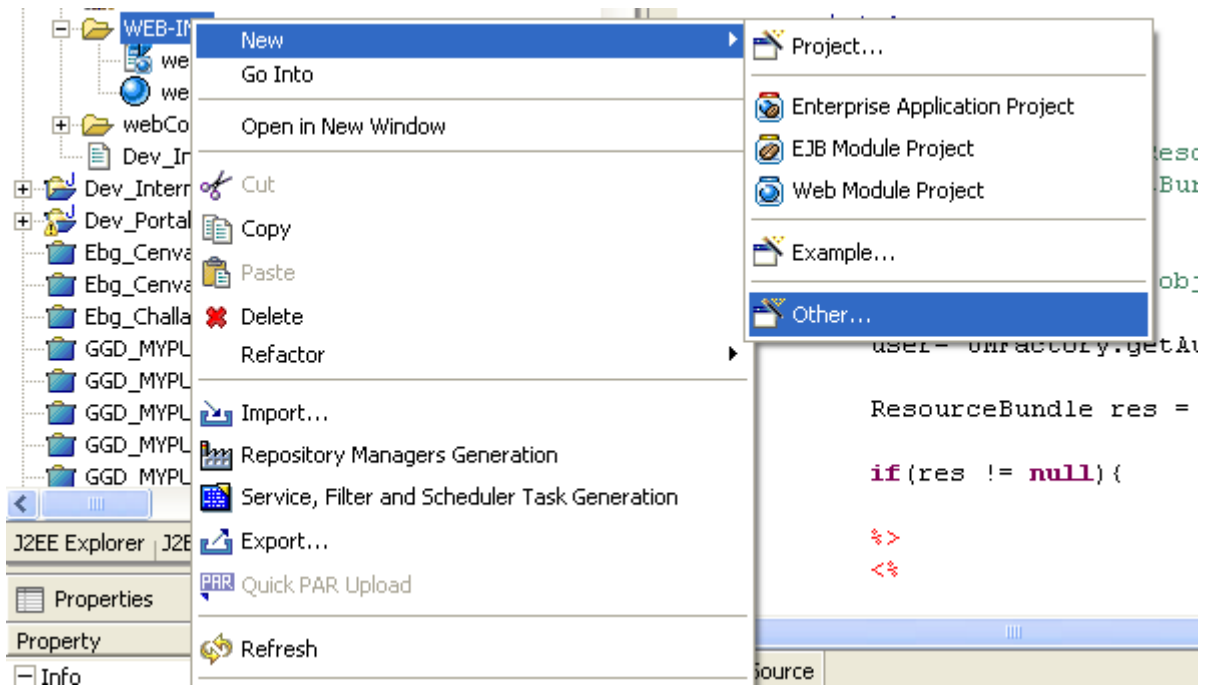
- ◆ Right click on the webContent folder and Select New JSP.



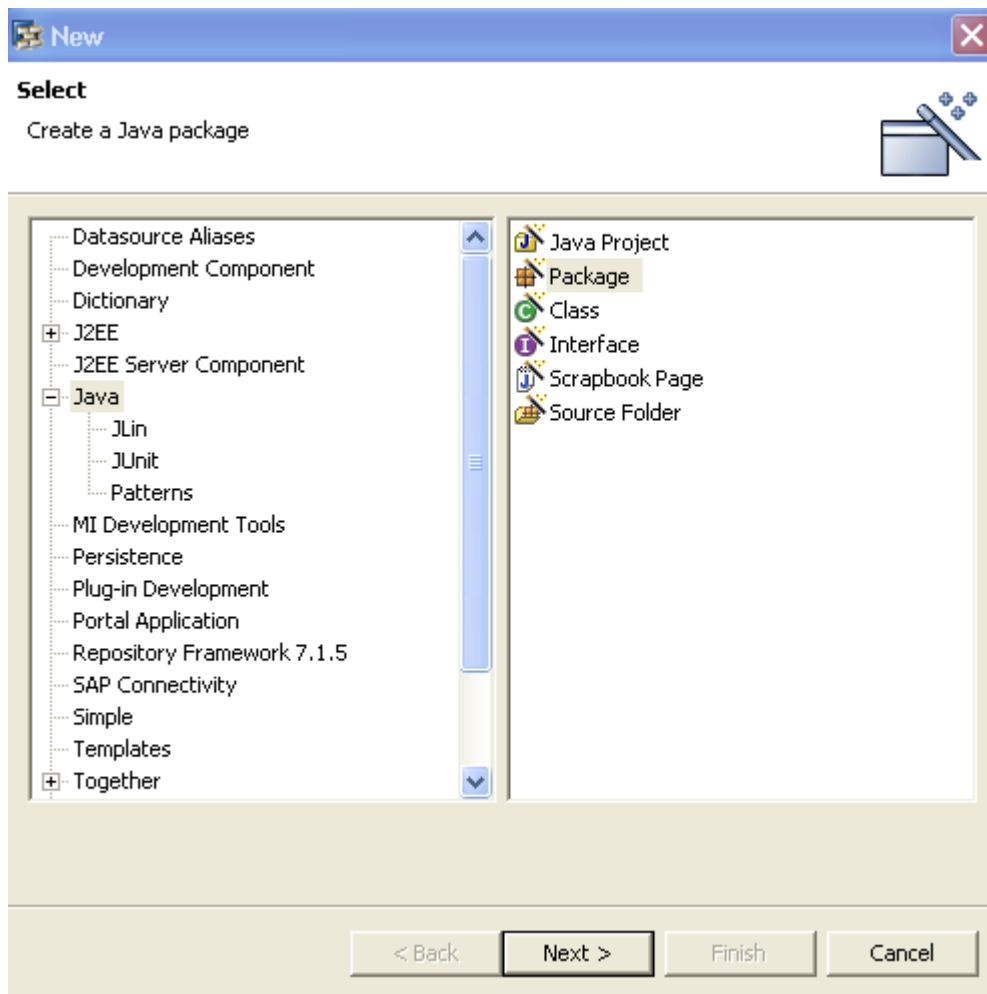
- ◆ Enter name of the JSP file and click on Finish.



- ◆ Open the project from the PackageExplorer and right click on WEB-INF folder and Select New -> Other tab

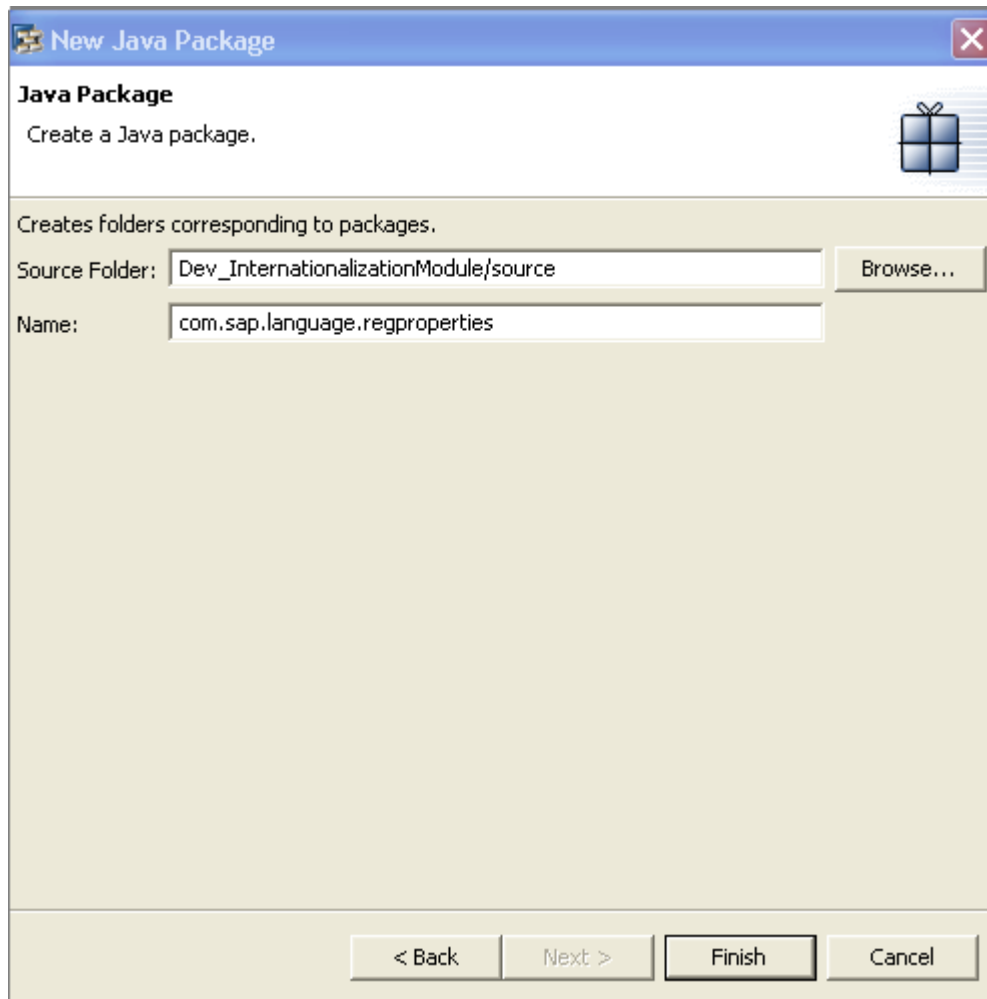


- ◆ Open the project from the Package Explorer and right click on WEB-INF folder and Select New -> Other tab -> Java -> Package and click on Next button.

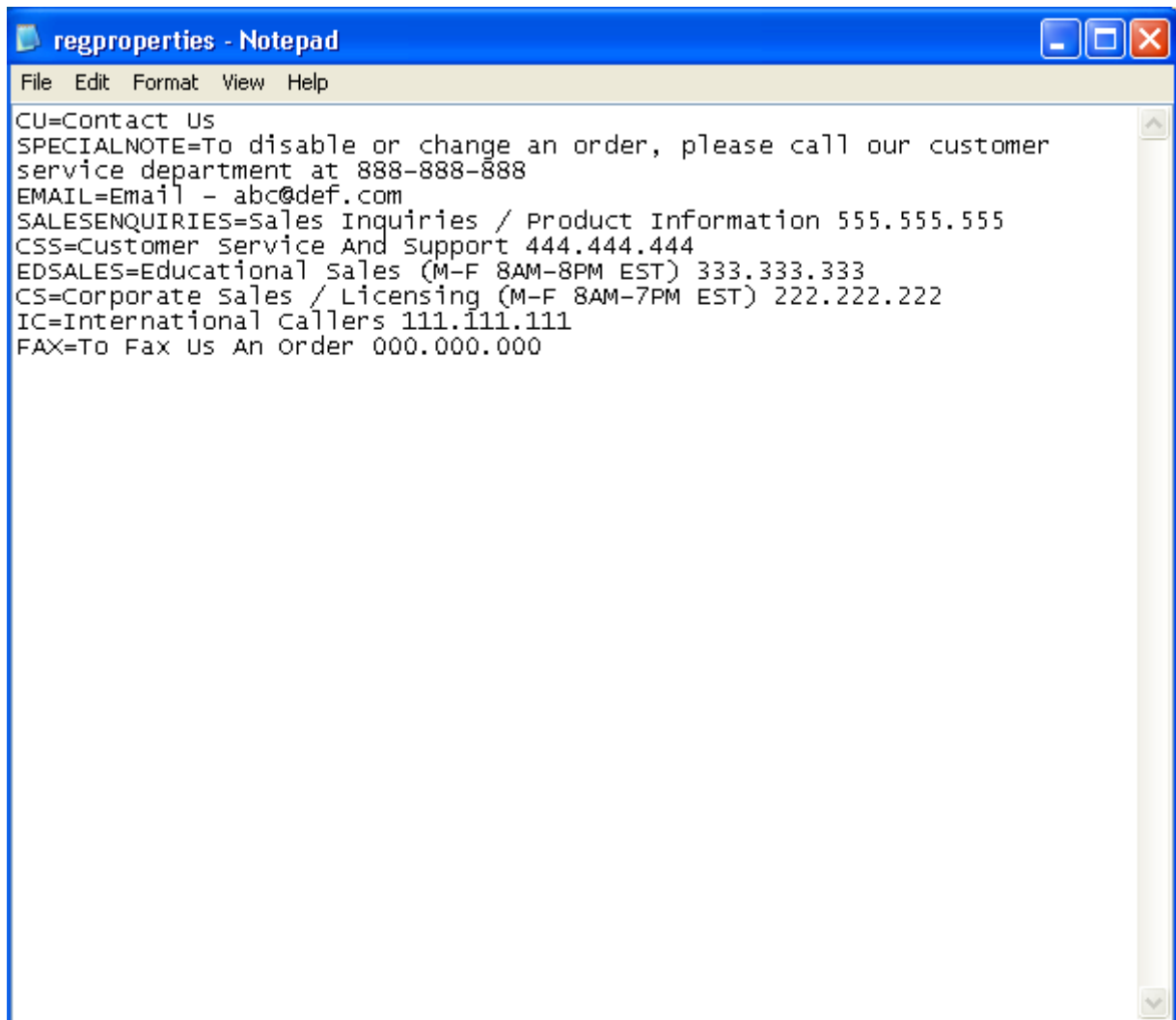




- ◆ Enter Package name and click on Finish.

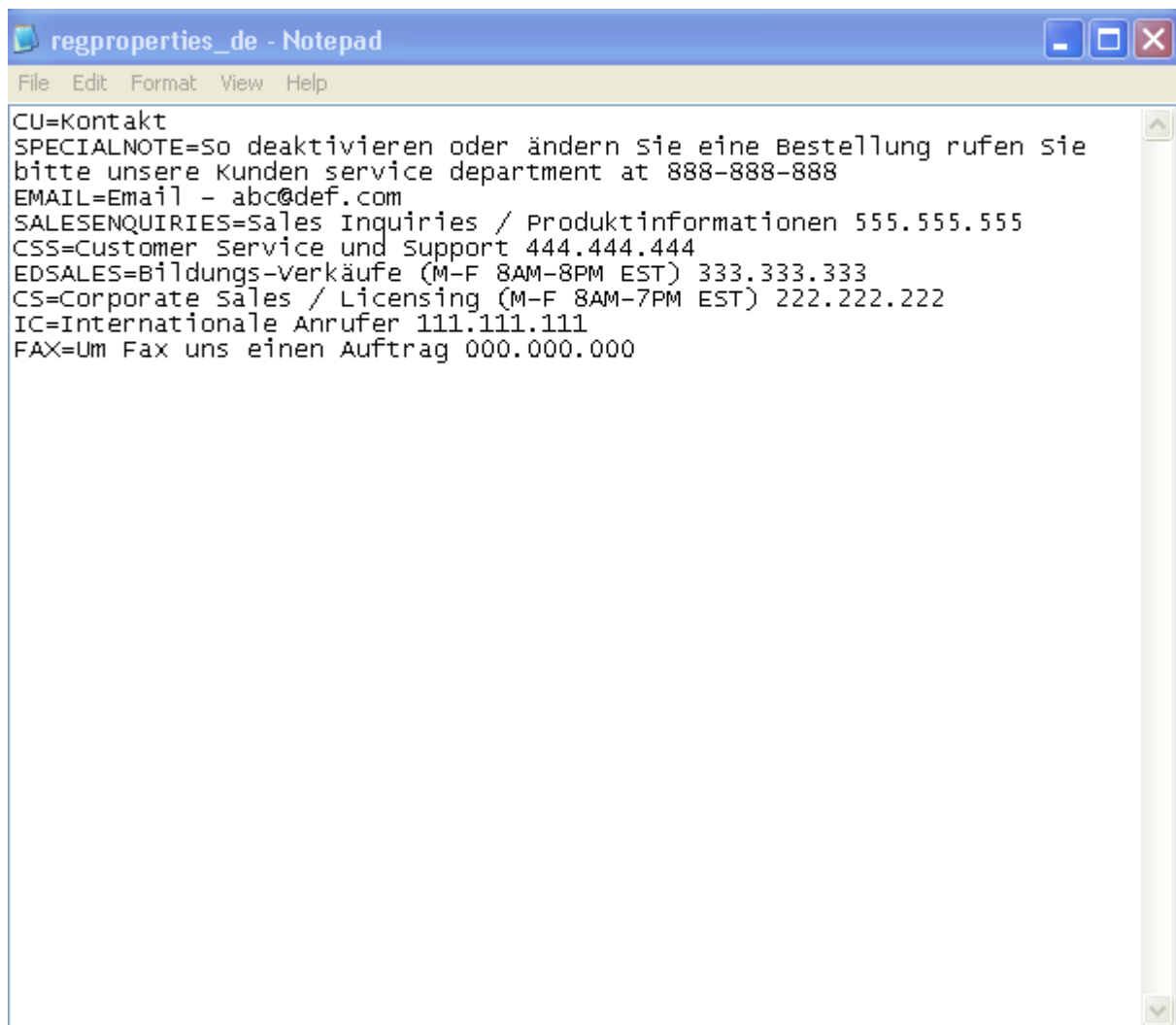


- ◆ Open notepad and create a text file regproperties and save with extension .properties



```
regproperties - Notepad
File Edit Format View Help
CU=Contact Us
SPECIALNOTE=To disable or change an order, please call our customer
service department at 888-888-888
EMAIL=Email - abc@def.com
SALESENQUIRIES=Sales Inquiries / Product Information 555.555.555
CSS=Customer Service And Support 444.444.444
EDSALES=Educational Sales (M-F 8AM-8PM EST) 333.333.333
CS=Corporate Sales / Licensing (M-F 8AM-7PM EST) 222.222.222
IC=International Callers 111.111.111
FAX=To Fax Us An Order 000.000.000
```

- ◆ These properties file can be used for storing strings for Internationalization and localization; these are known as Property Resource Bundles.
- ◆ Each parameter is stored as a pair of strings, one storing the name of the parameter (called the key), and the other storing the value.
- ◆ Each line in a .properties file normally stores a single property. Several formats are possible for each line, including key=value, key = value, key:value, and key value.
- ◆ Now, create another notepad file regproperties\_de.properties with the same keys but with the different values. These values are nothing but the German translation of the values stored in regproperties file.



```
CU=Kontakt
SPECIALNOTE=So deaktivieren oder ändern sie eine Bestellung rufen sie
bitte unsere Kunden service department at 888-888-888
EMAIL=Email - abc@def.com
SALESENQUIRIES=Sales Inquiries / Produktinformationen 555.555.555
CSS=Customer Service und Support 444.444.444
EDSALES=Bildungs-verkäufe (M-F 8AM-8PM EST) 333.333.333
CS=Corporate Sales / Licensing (M-F 8AM-7PM EST) 222.222.222
IC=Internationale Anrufer 111.111.111
FAX=Um Fax uns einen Auftrag 000.000.000
```

- ◆ Open the project in package explorer. Copy and paste these two properties files into the package which you have created (com.sap.language.regproperties).
- ◆ Open contactUs.jsp which you have created and write following code in the <body></body> tag.

```

<%
try {
    /* create an user object from the current user */
    IUser user;
    user= UMFactory.getAuthenticator().getLoggedInUser(request, response);

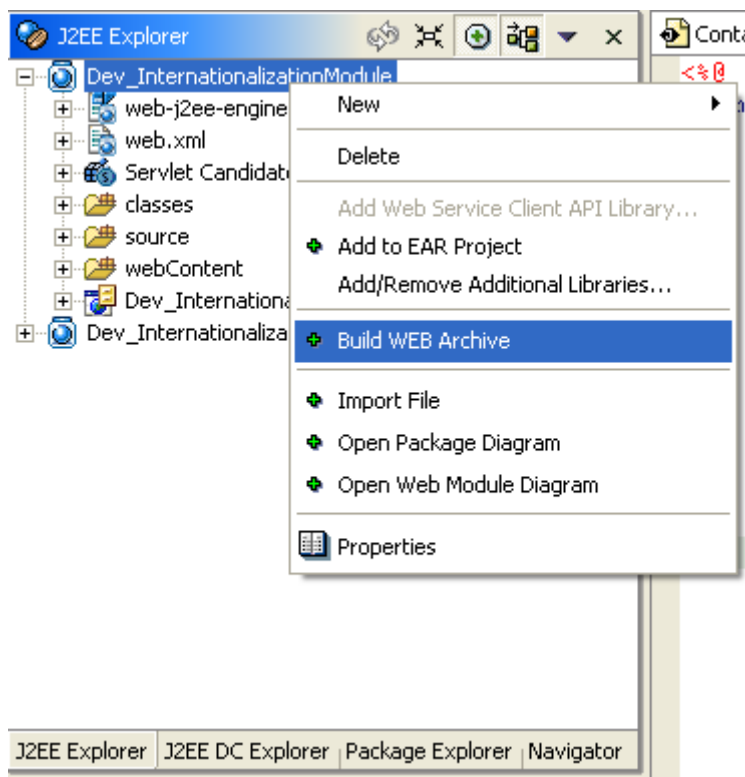
    ResourceBundle res = ResourceBundle.getBundle("com.sap.language.regproperties.regproperties",
                                                user.getLocale());

    if(res != null){

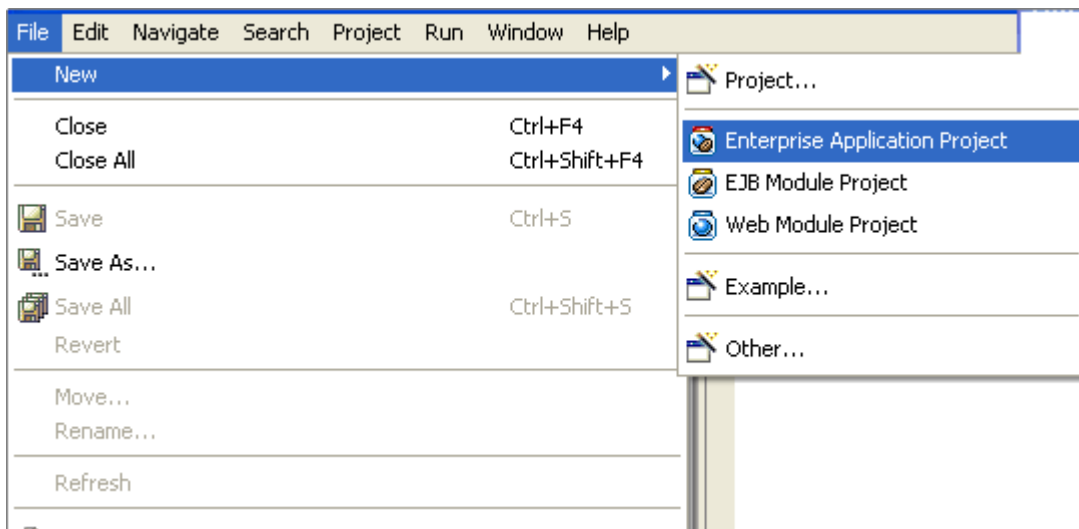
        Enumeration bundleKeys = res.getKeys();
        int m = 0;
        while (bundleKeys.hasMoreElements()) {
            String key = (String)bundleKeys.nextElement();
            String value = res.getString(key);
            if(m==0){
                %> <h4><%out.println(value);%></h4></br><%
            }else{
                %><h5>
                <%out.println(value);%
                m++;%></h5>
            }
        }
    }
} catch (Exception e) {
    out.println(e.getMessage());
}
%>
</h4>

```

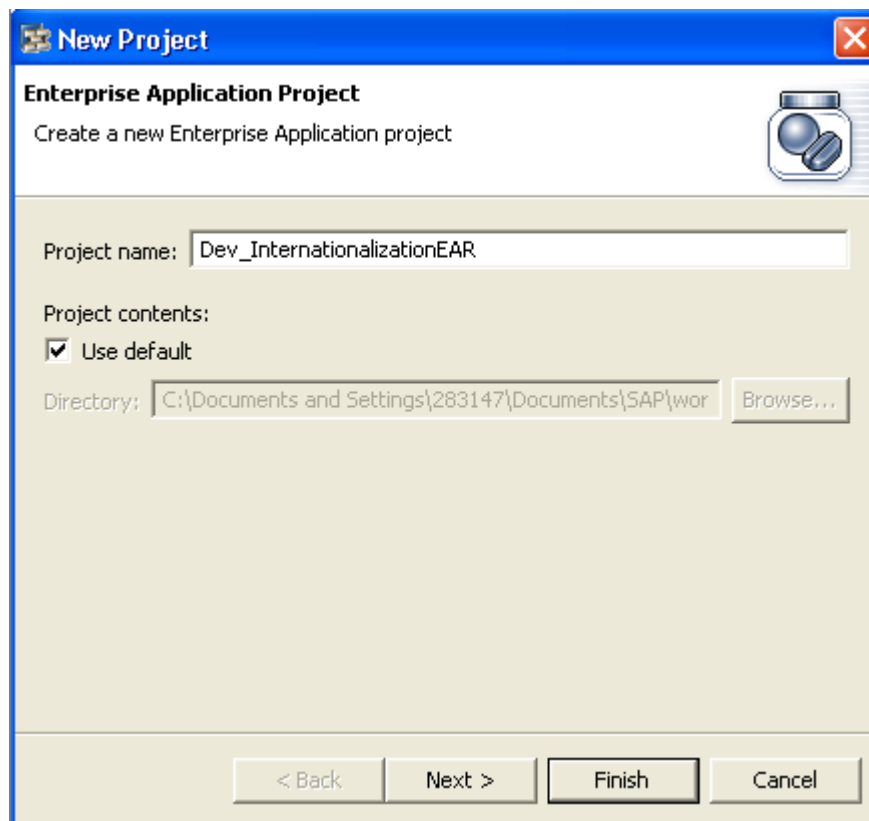
- ◆ Import the necessary packages & save the jsp file. Right click on Web Module Project and click on Build Web Archive to build the project.



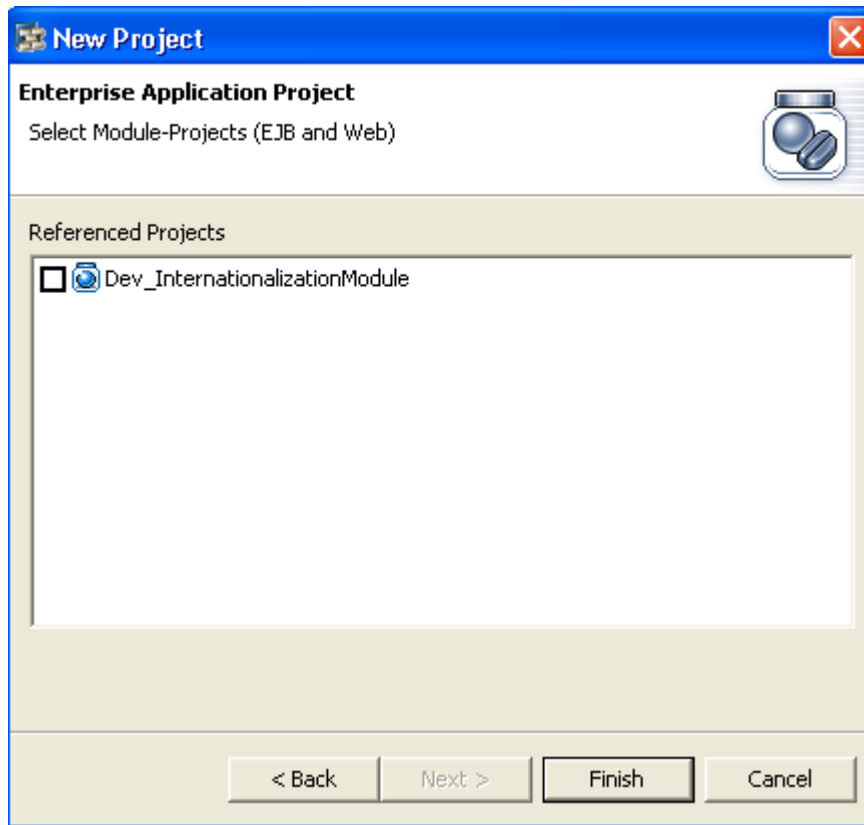
- ◆ To deploy this Web Module Project we need Enterprise Application Project. Select File -> New -> Enterprise Application Project



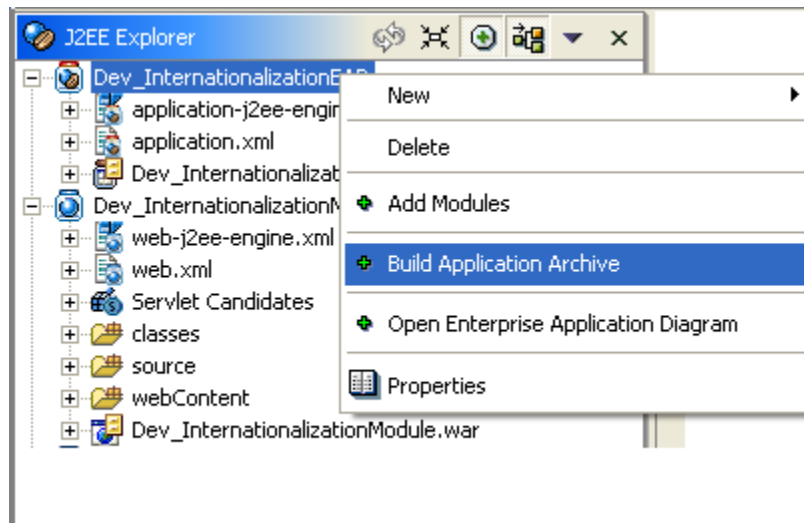
- ◆ Enter Project Name and click on Next button



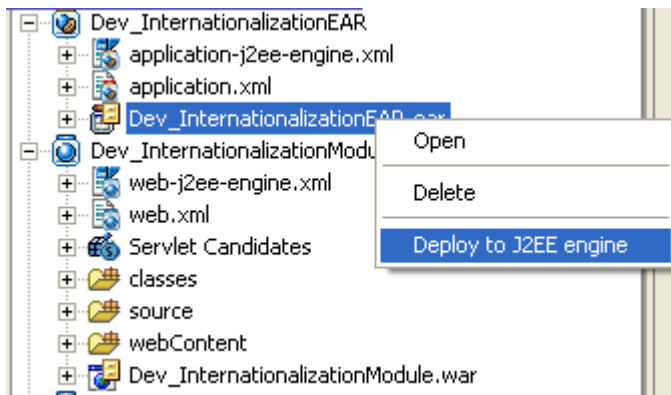
- ◆ Select your Web Module Project From the List of Module Projects & click on Finish button



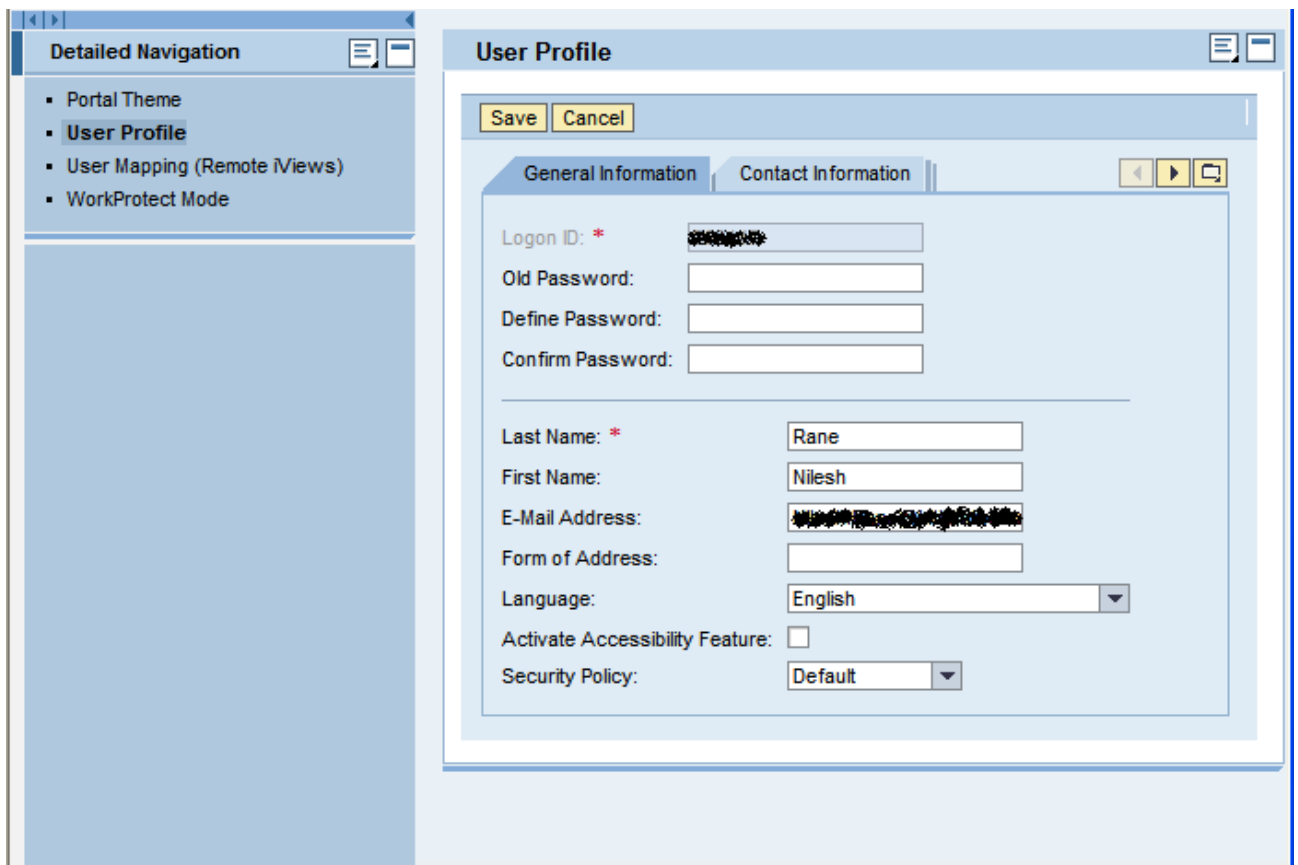
- ◆ Right click on Enterprise application Project and select Build Application Archive



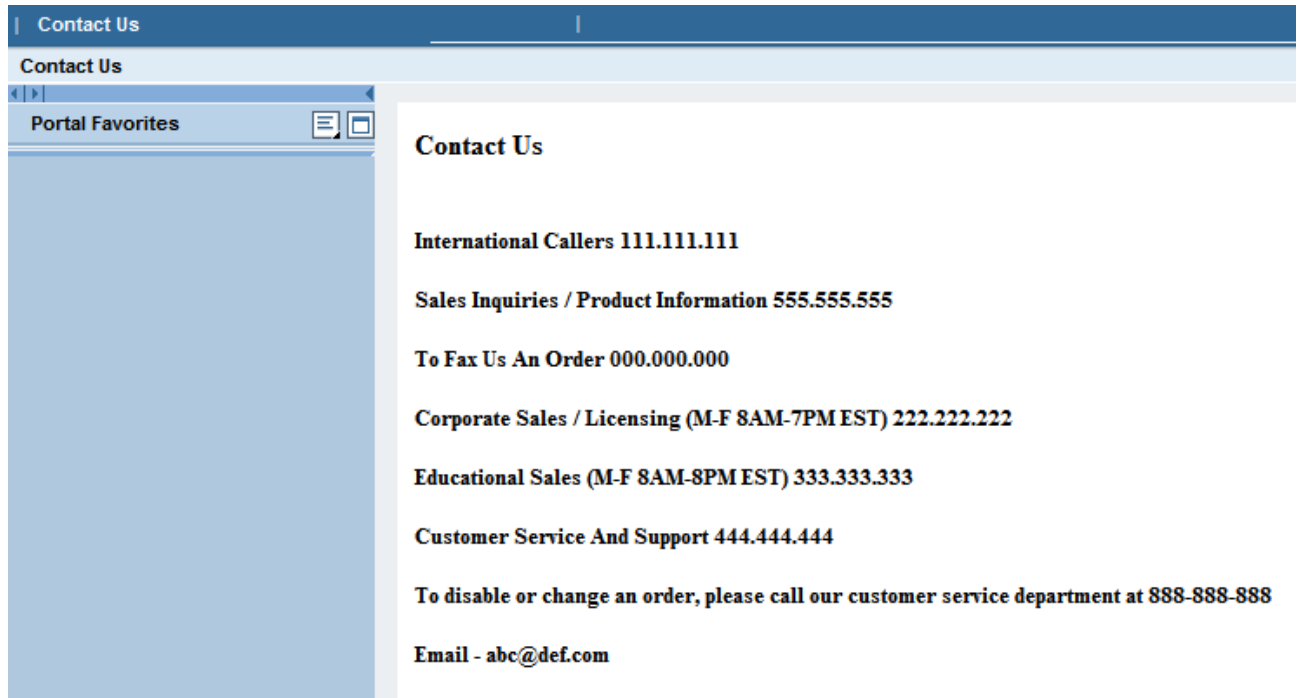
- ◆ Right click on .ear file and select Deploy To J2EE engine



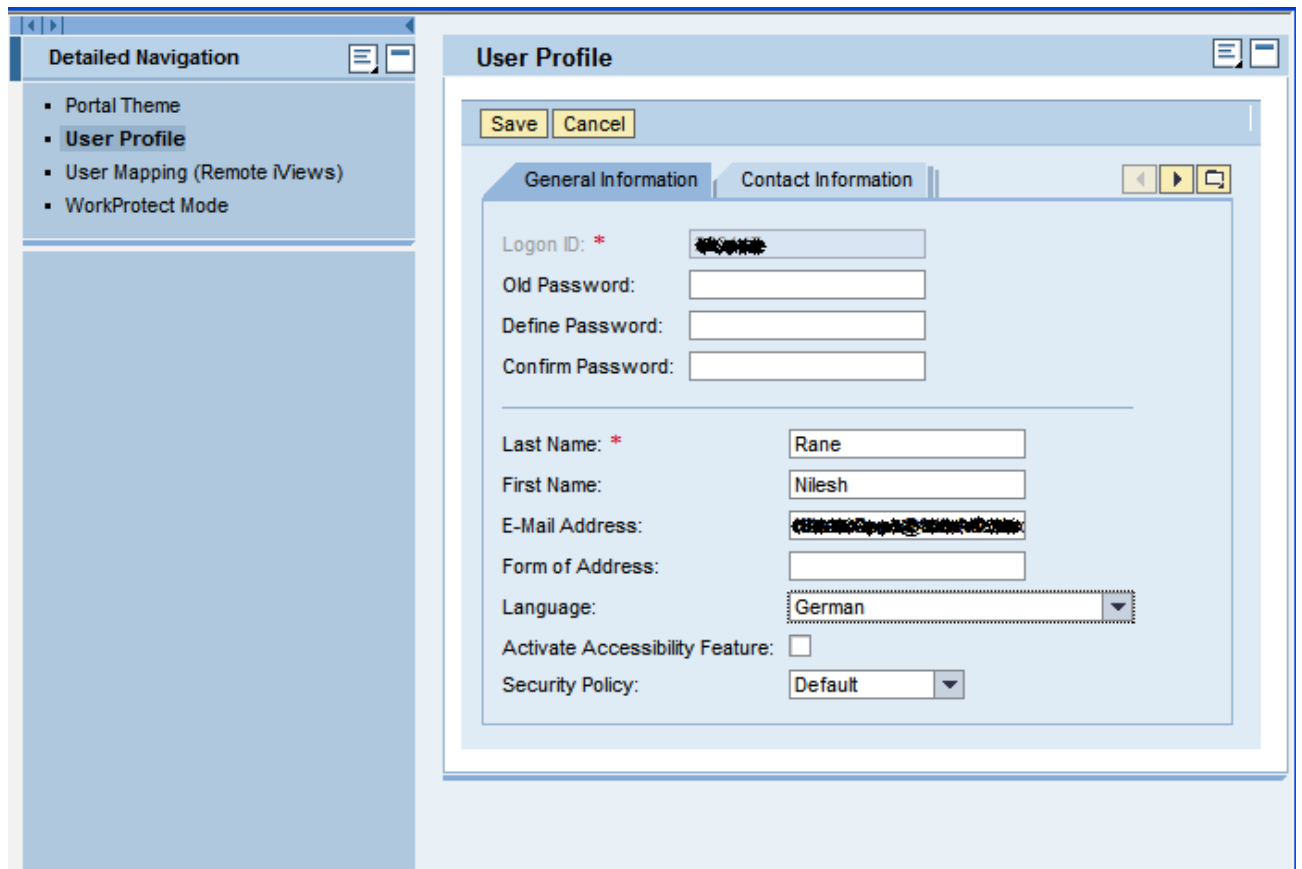
- ◆ Create an iview (select url iview from iview template) of this application on Portal (for this you need Content Administration rights). Assign this iview to a Role on portal.
- ◆ Result : -
- ◆ Logon to Portal and open personalize link. Select English as default language in the User Profile.



- ◆ Click on the iview which you have created.



- ◆ Since the user language is English, the information will be displayed in English
- ◆ Now, again open the User Profile and change user language to German(de)



- ◆ Save the changes and open a new session.
- ◆ Click on the iview which you have created. You will find that the portal iview (jsp page which you have created) is displayed in the German Language.





- ◆ Please note that you have maintained the translation (properties file) only in German language. Now, again open the User Profile and change the language to Chinese (note that you have not maintained the translation for your JSP for the Chinese language).
- ◆ Open a new session of portal and the same application. You'll find the same application is displayed in the English language, but the portal objects (Masthead etc) displayed in the Chinese language.
- ◆ Based on the login language of user, the text is translated.

## Summary

Thus, we have understood the need for Internationalization and the method to implement it for Web Module/JSP pages.

## Related Content

<http://java.sun.com/developer/technicalArticles/Intl/ResourceBundles/>

<https://forums.sdn.sap.com/thread.jspa?threadID=116590>

<http://www.sdn.sap.com/irj/scn/go/portal/prtroot/docs/library/uuid/c069b075-e48f-2c10-66ba-f57442c9076c?QuickLink=index&overridelayout=true>

For more information, visit the [Web Dynpro Java homepage](#).

## Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.

---

<sup>1</sup> Internationalization Purpose, Bijal Parmar

<http://www.sdn.sap.com/irj/scn/go/portal/prtroot/docs/library/uuid/c069b075-e48f-2c10-66ba-f57442c9076c?QuickLink=index&overridelayout=true>

<sup>2</sup> Java Internationalization: Localization with ResourceBundles, Sun Developer Network

<http://java.sun.com/developer/technicalArticles/Intl/ResourceBundles>