Name	Date modified	Туре	Size
🔰 en-GB	10/17/2008 9:28 AM	File Folder	
📔 en-US	10/17/2008 9:28 AM	File Folder	
\mu fr-FR	10/17/2008 9:28 AM	File Folder	
🔰 hi-IN	10/17/2008 9:28 AM	File Folder	
App_Code.compiled	10/17/2008 9:28 AM	COMPILED File	1 KB
🚳 App_Code.dll	10/17/2008 9:28 AM	Application Extens	5 KB
App_GlobalResources.compiled	10/17/2008 9:28 AM	COMPILED File	1 KB
App_GlobalResources.dll	10/17/2008 9:28 AM	Application Extens	5 KB
App_Web_cp0Igs.dll	10/17/2008 9:28 AM	Application Extens	4 KB
App_Web_izm_yrbr.dll	10/17/2008 9:28 AM	Application Extens	4 KB
App_Web_q7efoqr5.dll	10/17/2008 9:28 AM	Application Extens	4 KB

Now that we have created the resource files for the different cultures and languages, we need a way to load them at runtime when the user dynamically changes their culture. Fortunately, implementing this in ASP.NET 3.5 is quite easy. See the following code:

String welcome = Resources.TestSiteResources.Welcome;

In this line of code, we are using the Resources namespace, which was created automatically by ASP.NET when it compiled the resource files into satellite assemblies. We use the TestSiteResources class, with the same name as that of the resource file we created. We then access the Welcome property, which will give the actual text from the resource file, based on the current culture.

Implicit Versus Explicit Localization

If we want to localize the text of the Label control lblWelcome, we can do the same thing using these methods, in ASP.NET 3.5:

• **Implicit localization:** Here, we specify the new meta tags in the control definition and let ASP.NET get the value from the resource files based on the resourcekey attribute:

—**[** 221]—

For this to work, we need to have page-specific resource files in the /App_LocalDirectory folder. Implicit localization helps in trimming down the size of the global resource files which further helps in overall resource management. We should use implicit localization when we have mostly page-specific content.

We do not need to do anything manually to set these implicit localization properties. Just open your web page in the Design mode, go to **Tools** | **Generate Local Resources**. This will automatically create a resource file for your page. You only need to set the values (Control.Property) of different fields for each control in the resource file editor in VS.

• **Explicit localization:** This works when we have global resource files. Here, we use expressions to set the values from the resource files, as shown below:

We can set this using the VS IDE. Select the **Label** control, go to the **Properties** window, and select **Expressions | Text**. Then, choose **Resources** from the drop-down list and enter the class name (TestSiteResources, for this example) and the **Resource key** (Banner). This is the recommended way to localize the UI controls on a page.

Oh-09-Localization - Microsoft Visual Studio (Administrator)		
File Edit View Website Build Debug Data Format Tools Test Analyze		
📅 • 🖾 • 📸 🖬 🕼 👗 🐴 🖾 🤊 • 🖓 • 🖓 • 🖏 🕨 Debug 🔹 .NET	• 🛃 session	- 💀 📅 📾 🋠 💌 🗆
i 💭 🖳 🔄 達 標 萱 😫 XHTML 1.0 Transition 🔹 🚱 💂 i Style Application: Manu	ual 🔹 Target Rule: (New Inline Style) 🔹 🖅 🖽 📮 🕅	None) - (Default Font) - (Default - B I U A 2
Default.aspy Header.ascx TestSiteMasterPage.master.cs Header.ascx.cs Default.asp	App_GlobalResourciteResources.rest = ×	Solution Explorer - Solution 'Ch-09-Localization' (1 project)
asp:Panel#Panel1		
Select Language: Select a language 💌		Solution 'Ch-09-Localization' (1 project)
asp:Label#lblWekome	1	B- () E_\TestSite\
Test Website for Localization	2	a App_Code
1	þ	G App_GlobalResources G App_LocalResources
Ib/Welcome Expressions		Default.aspx
polivicionie capitosions		- Default.aspx.cs
	d it by selecting an expression type and setting the propertie	
that expression.		TestSiteMasterPage.master
Bindable properties:	Expression type:	Web.Config
III BorderColor	Resources	
III BorderStyle	Concernence of the second se	
III BorderWidth	Expression properties:	
III CssClass	ClassKey TestSiteResources	20
III Enabled	ResourceKey Banner	2
III EnableTheming		me System.Web.UI.WebControls.Label
III Fradieviewstate		1 III 7 III
III Height		ssions)
III SkinID	ClassKey	IbfWelcome
III TabIndex	For a global resource, specify the class name for the reso	
Test -	(for example the name of a .resx file in the App_GlobalRe	esour atedControlID
	OK Cance	Style NotSet
	100	BorderWidth
		CssClass
	*	(Expressions)
<u> </u>	<u>1</u>	The expressions that are bound to properties of this control.

-[222]-