



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:

```
<asp:Label id=lblWelcome meta:resourcekey="lblWelcome"
           Text="Welcome" runat="server">
</asp:Label>
```

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:

```
<asp:Label id=lblWelcome Text="<%$Resources:TestSiteResources,
                               Welcome %>"
          runat="server"></asp:Label>
```

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.

