

Agenda

1. *Definitions*
2. *Creating Resource files for different cultures*
3. *Designing the WebForm & linking controls to keys in Resource files*
4. *Alternatives to initialize culture settings of the page.*

Deccansoft

Globalization & Localization

Globalization is the process of designing and developing applications that function for multiple cultures.

Localization is the process of customizing your application for a given culture and locale.

Concerns

- While dealing with multilingual application if some dynamic (language independent) text has to be stored in the database table the column datatypes to be taken must be either nChar, nVarchar, nText.
- In Web.Config: `<globalization requestEncoding="Unicode" responseEncoding="Unicode"/>`

Three steps for building a Multi-Lingual Website or WebApplication

- Creation of Resource files
- Designing the WebForm so that the controls in it are linked to **Keys** of resource files
- Set the culture settings with thread executing the request from the client

Creation of Resource Files:

Local Resource File:

These files must be present in the folder **App_LocalResources** and the filename is based on the name of WebForm with which the resource file is associated with.

Steps:

- Create a new Web Application with Default.aspx
- To the root folder → RightClick → Add ASP.NET Folder → **App_LocalResources**
- To App_LocalResources → RightClick → Add a new Item → Resource File (provide name and Value pairs).

\AppName\Default.aspx.

\AppName\App_LocalResources\Default.aspx.resx – Culture Neutral Resource File.

\AppName\App_LocalResources\Default.aspx.fr.resx – French Neutral Resource File.

\AppName\App_LocalResources\Default.aspx.fr-BE.resx – French Belgium Resource File.

\AppName\App_LocalResources\Default.aspx.de.resx – German Neutral Resource File

KEYS	Default.aspx.resx	Default.aspx.fr.resx	Default.aspx.fr-BE.resx	Default.aspx.De.resx
IblAddressKey.ForeColor	Black	Red	Pink	Blue
IblAddressKey.Text	Address1 in Default	Address1 in French	Address1 in French - Belgium	Address1 in German
IblRealNameForeColorKey	Black	Red	Pink	Blue
IblRealNameTextKey	Microsoft (from Default)	Microsoft (in French)	Microsoft (in French Belgium)	Microsoft (in German)

Global Resource File in the folder App_GlobalResources:

This is Global to all the WebForms in a given web application. It can have any name and must be present in the Root directory of the web application

On Application → RightClick → Add ASP.NET Folder → App_GlobalResources	\AppName\App_GlobalResources\Settings.resx
On App_GlobalResources → RightClick → Add a new Item → Resource File (provide name and Value pairs)	\AppName\App_GlobalResources\Settings.fr.resx
	\AppName\App_GlobalResources\Settings.fr-BE.resx
	\AppName\App_GlobalResources\Settings.de.resx

KEYS	Settings.aspx.resx	Settings.aspx.fr.resx	Settings.aspx.Fr- BE.resx	Settings.aspx.De.resx
lblCompanyName	Microsoft (in Default)	Microsoft (in French)	Microsoft (in French Belgium)	Microsoft (in German)

Design the WebForm so that the controls are linked to Keys in Resource Files:**Steps for setting the Explicit Expression:**

- Select the control → RightClick → Properties → Goto Expressions
- On the lefthand side select the property to be initialized, set ExpressionType to Resources
- If the key has to be fetched from LocalResources File directly set ResourceKey. Otherwise, to fetch from GlobalResourcesFile first set ClassKey with the value of GlobalResource Filename and then set ResourceKey

Explicit Expression From Local Resource File

```
<asp:Label Text="<%$Resources:lblRealNameTextKey %>"
ForeColor="<%$Resources:lblRealNameForeColorKey %>" ID="lblRealName" runat="server"/>
```

Explicit Expression From Global Resource File

```
<asp:Label Text="<%$ Resources:Settings, lblCompanyName %>" ID="lblCompanyName"
runat="server"></asp:Label>
```

Implicit Expression

```
<asp:Label meta:ResourceKey="lblAddressKey" ID="lblAddress" runat="server"/>
```

Based on the key "lblAddressKey" the ForeColor and Text properties are automatically fetched from the resource file

Note: Implicit expression cannot fetch keys from Global Resource File

Fetching Values Programatically

In ASPX Page

```
<asp:Label ID="lblRealName1" runat="server" Font-Bold="False"></asp:Label>
```

Culture Demos

In ASPX Page

```
<asp:Label ID="lblDate" runat="server"></asp:Label>
<asp:Label ID="lblAmount" runat="server"></asp:Label>
```

Default.aspx.cs

```
protected void Page_Load(object sender, EventArgs e)
{
    lblRealName1.Text = GetLocalResourceObject("lblRealNameTextKey").ToString();
    lblRealName1.ForeColor =
System.Drawing.Color.FromName(GetLocalResourceObject("lblRealNameForeColorKey").ToString());
    decimal amt = 100.02M;
    lblAmount.Text = amt.ToString("C");
    lblDate.Text = DateTime.Now.ToString();
}
```

We can use **GetGlobalResourceObject** for reading the Key – Value pair from Global resource file

Any one of the following alternatives can be used for associating the culture with the current thread / request in execution:

Alternative 1

In Web.Configfile: <globalization culture="de-AT" uiCulture="de-AT">

Note: If configured here the value would remain same for all the webforms of the project and is not based on client's choice of language.

Alternative 2:

In ASPX Page

```
<%@ Page UICulture="auto" Culture="auto" ... %> or
<%@ Page UICulture="de-AT" Culture="de-AT" ... %>
```

- Request.**UserLanguages** is a collection and is based on the value (comma separated) of a request header "Accept-Language"

- If the value of Culture or UICulture is set to "auto" then the value is taken from first value of request header **"User-Languages"**. This request header is initialized in browser by going to Tools → Internet Options → Languages.
- If configured here then it has to be done for every page and the value to be mentioned must be constant or "auto".

Note: If the Culture settings of the thread has to be set programatically, it must done either in InitializeCulture method of the Page class or before it (Application_BeginRequest of Global.asax)

Alternative 3:

In aspx page

```
<asp:DropDownList AutoPostBack="true" ID="ddlCulture" runat="server"
onselectedindexchanged="ddlCulture_SelectedIndexChanged">
<asp:ListItem Value="en-US" Text="English"></asp:ListItem>
<asp:ListItem Value="de-AT" Text="German"></asp:ListItem>
<asp:ListItem Value="fr-Be" Text="French"></asp:ListItem>
<asp:ListItem Value="hi-IN" Text="Hindi"></asp:ListItem>
</asp:DropDownList>
```

Code behind

```
protected void ddlCulture_SelectedIndexChanged(object sender, EventArgs e)
{
    Response.Cookies ["culture"].Value = ddlCulture.SelectedValue;
    Response.Redirect(Request.Path);
}

protected override void InitializeCulture()
{
    if (Request.Cookies["culture"] == null)
        return;
    Page.Culture = Request.Cookies["culture"].Value;
    Page.UICulture = Request.Cookies ["culture"].Value;
}

protected void Page_Load(object sender, EventArgs e)
{
    if (!IsPostBack)
    {
        if (Request.Cookies["culture"] != null)
            ddlCulture.SelectedValue = Request.Cookies["culture"].Value;
    }
}
```

```
}
```

Alternative 4:

Following should be written in Global.asax and this global to all the WebForms of the application

```
Global.asax

protected void Application_BeginRequest(object sender, EventArgs e)
{
    if (Request.Cookies["culture"] == null)
    return;

    System.Globalization.CultureInfo ci = new
System.Globalization.CultureInfo(Request.Cookies["culture"].Value);

    System.Threading.Thread.CurrentThread.CurrentCulture = ci;
    System.Threading.Thread.CurrentThread.CurrentUICulture = ci;
}
```

Notes:

1. Based on the **Culture** value of the CurrentThread the Formatting functions (like Date or Currency) render their output.
2. Based on **UICulture** of the CurrentThread the framework loads the appropriate resource file for all the culture specific lookups.
3. The Culture and UICulture values can be set either in **Web.Config – Globalization Tag** or **Application_BeginRequest** or in **<%Page Directive** or in **InitializeCulture** method of page class. This is also the increasing order of their precedence

For languages like Arabic and Urdu do the following

1. In aspx page:
`<HTML dir="<%=lang %>">`
2. Add the following in page class (before page_load)
`public string lang;`
3. Add the following in Page_Load
`lang = "rtl" //For Setting the Direction of the language.`