**Globalization and localization**

**Introduction:**

Globalization and localization is used for reusability of software for different culture and locale. For example we have developed one application in English language, after some time customer need same application in German language. Instead of wasting time in building application in German language, we can use Globalization and Localization.

**Globalization:**

Globalization is designing and development of application in such way that it can be easily adapted to multiple culture and locale. Globalization will not translate any application instead it will allow to develop application which can be translated.

With globalization, the application supports number and date format which are depending on the culture. To globalize an application we can use classes from namespace System.Globalization.

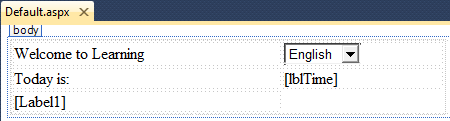
**Localization:**

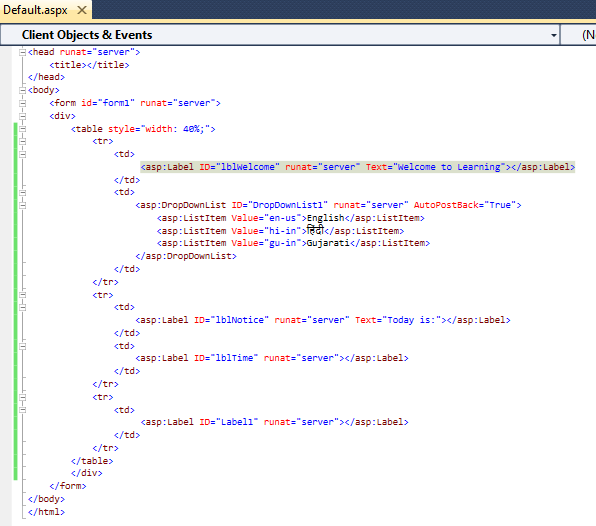
Localization is about translating or customizing application for specific culture so it become available for local clients. For localization, we can use resources from namespace System.Resources.

**Example:**

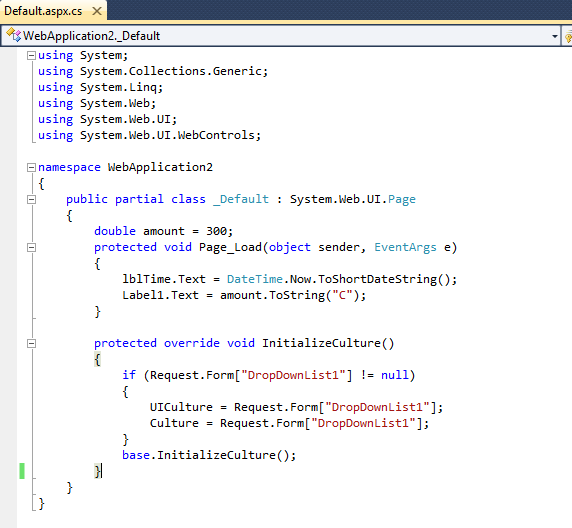
Let us take simple example to understand globalization and localization.

1. On the visual studio menu bar, choose **File->new->project**.
2. **Other Project Types**-> **Visual Studio Solutions**. Name the solution **"GlobalizationSolution"**
3. In **Solution Explorer**, right-click on the “**GlobalizationSolution”-> Add** -> **New Project**->**ASP.NET Web Application->** name the application **“WebApplicationGloLoc”**





Create simple design using HTML table and put every text in label. In globalization, we create user Interface where static content is changed at runtime base on culture for that we put all static text into label which can be easily change at runtime according to culture. Code behind for date and fee are as below.



There are two more things which need to be globalized, first current date/time second amount.



We can use ToString() to create a different representation of the number depending on the locale. For the any number structure, ToString() can be overloaded as below:

Public string ToString(string);

* Here string specifies format for representation. For numeric formatting strings are predefined,

where, **C** specifies the currency notation**,**

**D** creates decimal output,

**E** creates scientific output,

**F** creates fixed-point output**,**

**G** creates general output,

**N** creates number output and

* In order to make date/time globalize we can use public instance methods

ToLongDateString()

ToLongTimeString()

ToShortDateString()

ToShortTimeString()

Which create string representation using the current culture.

After globalizing date and currency, if we run our project currency will be in dollar format. This change occurs because of culture set to your browser. Now if we will change culture of browser then output will be different.

**Google Chrome**

**To Change culture:->Google Chrome ->Settings ->Scroll To bottom->Advanced->Language->Add Language->French(France)->More Actions(three dots)->Check Display Google Chrome in this language->Relaunch.**

After relaunch refresh .aspx page and currency will be in euro format. So we have globalized currency and Datetime but our label are still in English language. Now we will convert UI in required language which is known as localization. For localization we will use one of the feature provided by Microsoft which is Resource file.

**How to use resource file?**

**Design->Tools->Generate local resource**

Once you generate local resource two changes will occur.

1. In your aspx page meta:resourcekey property will be generated for every server control with some values
2. In solution explorer new App\_LocalResources folder will be added.

In this folder we have one file GloLoc(your application name).aspx.resx which will have mapping of meta:resourcekey with it’s value.