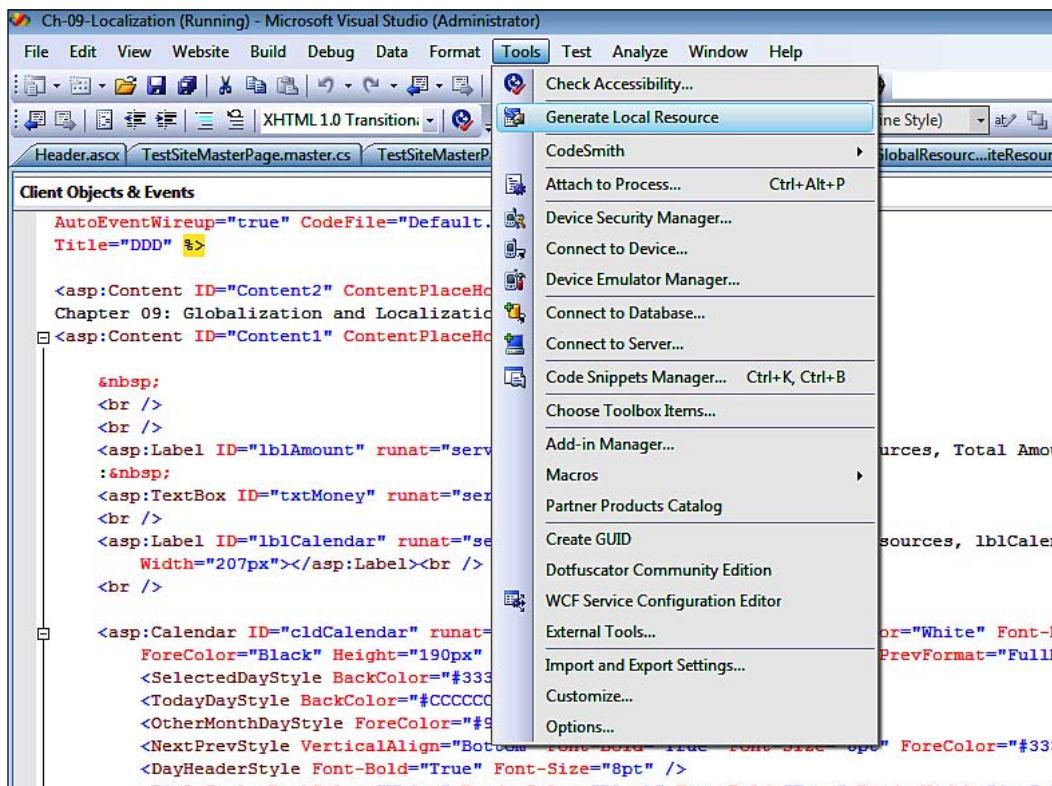


## Creating Local Resources

Creating local resources is easy because we can use VS to generate local .resx files for our pages automatically. Here are the steps:

- 1 Select the `Default.aspx` file in the Solution Explorer.
- 2 Go to the **Tools** option in VS and select the **Generate Local Resource** option (this option will be visible only when you have selected an ASPX file in **Design Mode** or **Source Mode**; it will not be available when you are on a code-behind file). Here is how it should look:



- 3 VS will automatically generate a .resx file under the `App_LocalResources` folder.
- 4 Edit that file and add values using VS.
- 5 Add more files in that folder for additional cultures as required.



If you want only certain pages to show the localized strings, you can restrict the localization behavior throughout the application by putting resource files in the `App_LocalDirectory` folder. This will make the localization page-specific, and not application-wide.

The naming of the `.resx` files should follow a certain pattern as shown here (assuming that you want to localize a page named `MyPage.aspx`):

- `MyPage.aspx.resx`: This is the default resource file for `MyPage.aspx`
- `MyPage.aspx.fr-FR.resx`: This will be used when the culture changes to French, though only `MyPage.aspx` in the application would be localized

## Choosing between Global and Local Resources

Global resource files should be used when we have content that is common across multiple pages in our website. For example, text such as the company name and address can be repeated on many pages, so these items should be put in global resources files. But if we put all of our content only in global resource files, then we will be overloading `App_GlobalResources` and it can grow in size, making it difficult to manage and update the `.resx` file. For large applications having a lot of content in a single resource file, there can be a performance impact.

On the other hand, if we put all of the data only in the local resource files, then there would be a resource file for every page, which would create a lot of maintenance issues. Also, resources inside local resource files cannot be shared with other pages or controls in our website, which means that we would have to duplicate common content in multiple local resource files. So we need to strike a balance by putting only page-specific content in the local resource files and putting generic content in the global resource files.

## Satellite Assemblies

All of the resource files are compiled into assemblies at runtime. These assemblies are known as "satellite assemblies", and have strongly-typed wrappers for the XML resource files. So we don't need to worry about creating resource assemblies in ASP.NET. These assemblies are placed in separate folders (according to the name of the locale) under the `/bin` folder, after you have published your website: