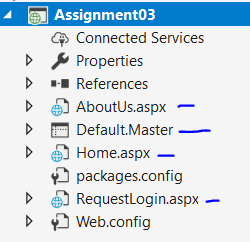
Micah Hoffmann

Assignment 03

7/13/2019

This walkthrough illustrates how to create a master page and several content pages. Master pages allow you to create a page layout — a template page — and then create separate pages containing content that is merged with the master page at run time.

Why would we want to use the master/template page methodology? If you build a static HTML/CSS website or even an ASP.NET web application without master/template pages, each page exists independently and serves HTML independent. This creates redundancy and could present errors if you don’t have equal HTML where you want the webpage to be equal, example; the navigation bar.



The project structure is as follows:

* .Master file
* n number of .aspx template pages

**Master page**

The master page can be thought of as a container. It can hold standard HTML code that is served at all times, but it can also hold template pages that can be swapped based on the request.

<asp:ContentPlaceHolder Is the code that tell the Master page to pick a corresponding .aspx file to serve. It looks at the ID=”Name” which corresponds to the <asp:Content code block in the template file.

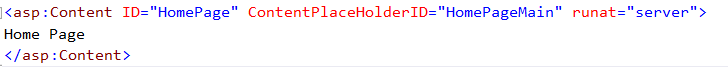


**Template page**

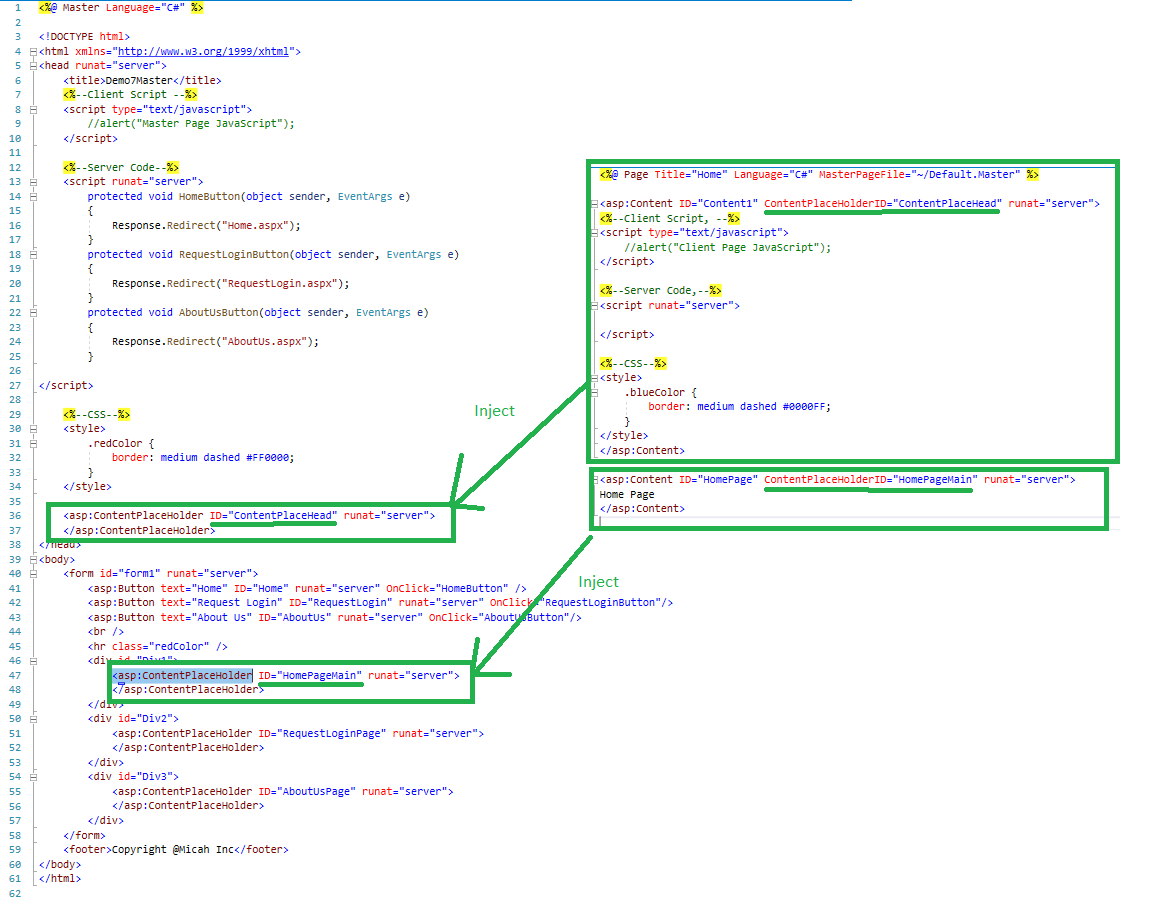
There are two keys points to the template pages.



Which references the MasterPageFile, so they can work together.



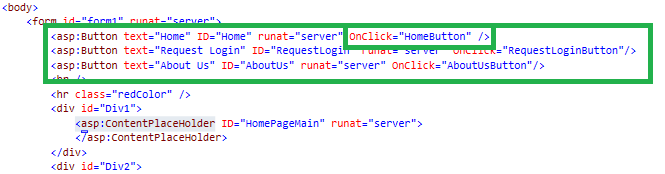
The Content tag has a corresponding ContentPlaceHolderId, which matches in the Masterfile. The Content tag injects all of the code in-between the opening and closing, into the corresponding Master file.



**Shared Navigation**

One of key uses for this master/template methodology, is the ability for the website to have shared features that are controlled in one location. What is one thing that is typically shared across the entire website? The navigation.

So to add a shared navigation, we simply add the code to the Master file, which will then be equally served by each of the .aspx pages. Notice it is not within the <asp:ContentPlaceHolder> code, because we want it to exist at the Master page level.



To get the buttons to function we add some ASP.Net server code. The OnClick=”Name” in the html, must match the method in order to work. Each of the methods use a Response.Redirect, to trigger which .aspx page the website will redirect to.

