

# Import a Custom Master Page and Site Page with an Image in Visual Studio 2010

(Microsoft Corporation)

This walkthrough demonstrates how to import a SharePoint custom master page and a site page that has an image into a Visual Studio SharePoint project. This walkthrough shows how to accomplish the following tasks:

- Create a custom master page and a site page by using an image in SharePoint Designer.
- Export a custom master page, image, and site page to a SharePoint solution (.wsp) file.
- Import and deploy the .wsp file into a Visual Studio SharePoint project by using the Import SharePoint Solution Package project.

**Note:** Your computer might show different names or locations for some of the Visual Studio user interface elements in the following instructions. The Visual Studio edition that you have and the settings that you use determine these elements.

## Prerequisites

You must have the following components to complete this walkthrough:

- Supported editions of Microsoft Windows and SharePoint.
- Visual Studio 2010.
- SharePoint Designer 2010.

## Create Items in SharePoint Designer

This example shows how to create three items in SharePoint Designer for export: a custom master page, a site page that references the custom master page, and an image file to appear on the site page. The image is added to the /images/ folder in SharePoint.

### To create a custom master page in SharePoint Designer

1. In SharePoint Designer, in the Navigation pane, click **Master Pages**.
2. On the ribbon, click **Blank Master Page**.
3. At the bottom of SharePoint Designer, click the **Code** tab.
4. Replace the existing markup with the following markup.

```
<%@ Master Language="C#" %>
<%@ Register tagprefix="SharePoint" namespace="Microsoft.SharePoint.WebControls" a
ssembly="Microsoft.SharePoint, Version=14.0.0.0, Culture=neutral, PublicKeyToken=7
1e9bce111e9429c" %>
<html dir="ltr">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<SharePoint:RobotsMetaTag runat="server" __designer:Preview="" __designer:Values="
&lt;P N='InDesign' T='False' /&gt;&lt;P N='ID' T='ctl100' /&gt;&lt;P N='Page' ID='1
' /&gt;&lt;P N='TemplateControl' ID='2' /&gt;&lt;P N='AppRelativeTemplateSourceDir
ectory' R='-1' /&gt;"></SharePoint:RobotsMetaTag>
<title>Web Page</title>
</head>
<body>
<form id="form1" runat="server">
<asp:ContentPlaceHolder id="ContentPlaceHolderMain"
```

# Import a Custom Master Page and Site Page with an Image in Visual Studio 2010

(Microsoft Corporation)

```
runat="server">
  </asp:ContentPlaceHolder>
</form>
</body>
</html>
```

5. Save the page as mybasic1.master.

## Add an Image to the Content Database in SharePoint Designer

Now you can add an image to display on the site page. The image is deployed to the SharePoint content database.

### To add an image to the content database in SharePoint Designer

1. On the ribbon, click **All Files**, and then, in the tree view, select **images**.
2. On the ribbon, click **Import Files**, select a file of your choice, and then click **OK**. In this example, the file is named myimg1.png. Optionally, you can create a subfolder to help organize the images.
3. Close the **Import** dialog box.

## Create a Site page

This basic site page uses the custom master page and displays the image that you added in the previous step.

### To create a site page

1. In the Navigation pane, click **Site Pages**.
2. On the ribbon, click the **Page** button and then click **ASPX**. Name the new file mycontentpage1.aspx. Optionally, you can create a subfolder to help organize the site pages.
3. In the list, click **MyContentPage1.aspx** to open its properties page and then, at the bottom of the page, click the **Edit** link.
4. At the bottom of the page, click the **Code** button.
5. Replace the existing markup with the following markup.

```
<%@ Import Namespace="Microsoft.SharePoint.ApplicationPages" %>
<%@ Register Tagprefix="SharePoint" Namespace="Microsoft.SharePoint.WebControls" Assembly="Microsoft.SharePoint, Version=14.0.0.0, Culture=neutral, PublicKeyToken=71e9bce11e9429c" %>
<%@ Register Tagprefix="Utilities" Namespace="Microsoft.SharePoint.Utilities" Assembly="Microsoft.SharePoint, Version=14.0.0.0, Culture=neutral, PublicKeyToken=71e9bce11e9429c" %>
<%@ Register Tagprefix="asp" Namespace="System.Web.UI" Assembly="System.Web.Extensions, Version=3.5.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" %>
<%@ Import Namespace="Microsoft.SharePoint" %>
<%@ Assembly Name="Microsoft.Web.CommandUI, Version=14.0.0.0, Culture=neutral, PublicKeyToken=71e9bce11e9429c" %>
<%@ Page Language="C#" Inherits="Microsoft.SharePoint.WebControls.LayoutsPageBase" MasterPageFile="../../catalogs/masterpage/mybasic1.master" meta:progid="SharePoint.WebPartPage.Document" %>
```

# Import a Custom Master Page and Site Page with an Image in Visual Studio 2010

(Microsoft Corporation)

```
<asp:Content ID="Main" ContentPlaceHolderID="ContentPlaceHolderMain" runat="server"
">

</asp:Content>
```

6. Save the updated site page.

## Export the Items from SharePoint

Export the items from SharePoint to a SharePoint solution (.wsp) file.

### To export items from SharePoint Designer

1. In SharePoint Designer, on the Navigation bar, click **Team Site**, and then, on the ribbon, click **Save As Template**.
2. In the **Save as Template** dialog box, type a file name and template name, select the **Include Content** check box, and then click **Finish**. This saves the contents of the site in the .wsp file.
3. After the solution exports, click the **Solution Gallery** link to display the list of available solution files.
4. Click the new .wsp file and save it to the system.

## Import the Items into Visual Studio

Import the .wsp file into Visual Studio. After the content is imported, you can customize it, add more items, and then deploy it.

### To import items from the .wsp file into Visual Studio

1. In Visual Studio, create an **Import SharePoint Solution** project.
2. On the **Select items to import** page, under **Module** in the **Type** column, select only the files in the following table for import.

File Name	Description
_catalogsmasterpage_	The custom master page.
images_	The image file in the SharePoint file system.
SitePages_	The site page.

3. Click **Finish** to import the selected items.
4. In **Solution Explorer**, click the custom master page and set its **Deployment Conflict Resolution** property to **Automatic**. This helps ensure that that any deployment conflicts are resolved automatically.
5. If your new master page has the same name as an existing page, make sure that the existing page is not marked as either a Default Master Page or a Custom Master Page in SharePoint Designer. If an

# Import a Custom Master Page and Site Page with an Image in Visual Studio 2010

(Microsoft Corporation)

existing master page is marked as either Default Master Page or Custom Master Page, you will get a deployment error that states that the master page cannot be deleted. To avoid this problem, do this:

- If the existing master page is set as Default Master Page, temporarily set another master page as Default Master Page. After you deploy the files to SharePoint, set your new master page as Default Master Page.
- If the existing master page is set as Custom Master Page, temporarily set another master page as Custom Master Page. After you deploy the files to SharePoint, set your new master page as Custom Master Page.

6. On the **Build** menu, click **Deploy Solution**.

7. Open the SharePoint site to view the deployed items.

An alternative way to import files into Visual Studio and deploy them to SharePoint is to add the files into modules in Visual Studio. For more information, see [How to: Import a Master Page or Theme and Using Modules to Include Files in the Solution](#).