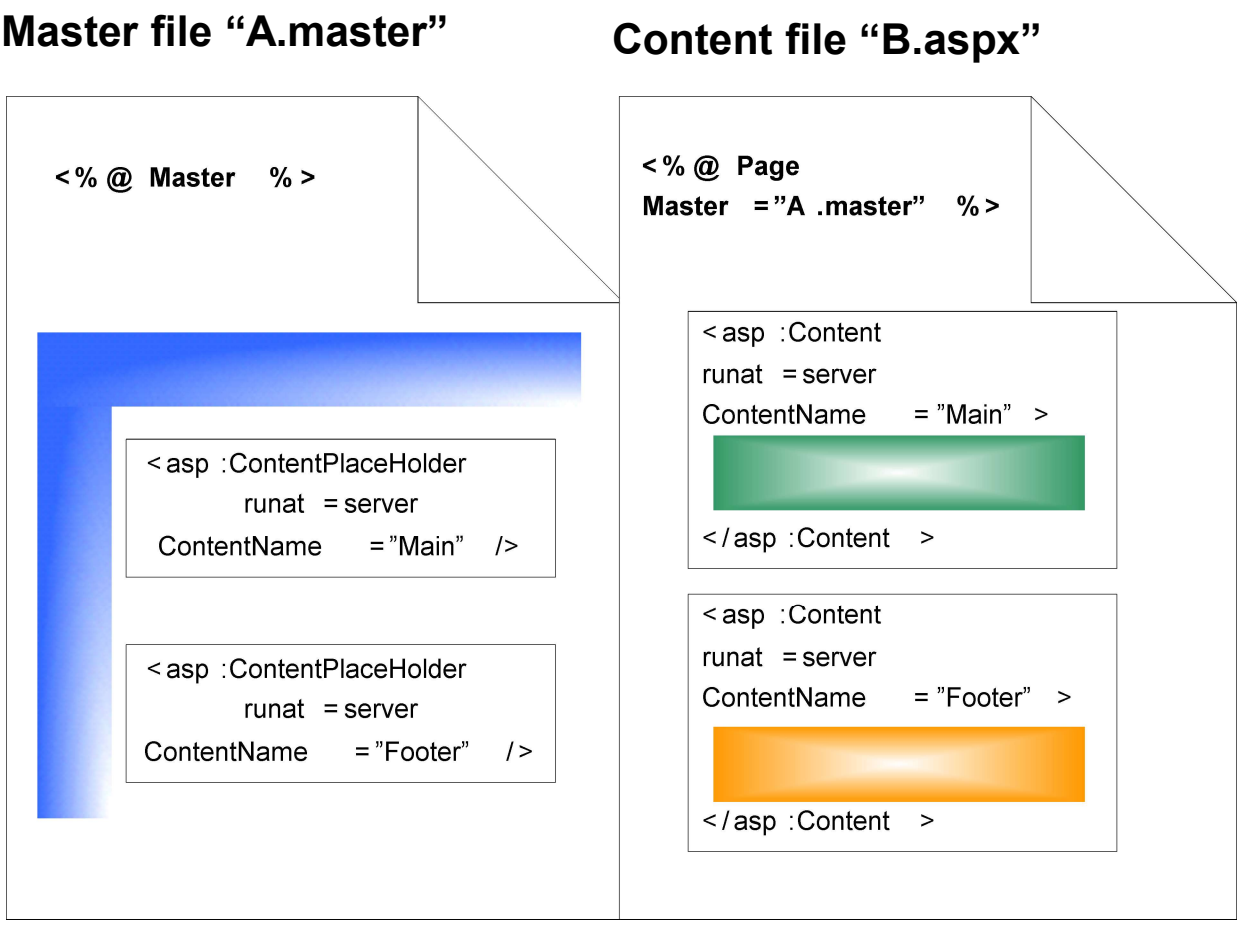
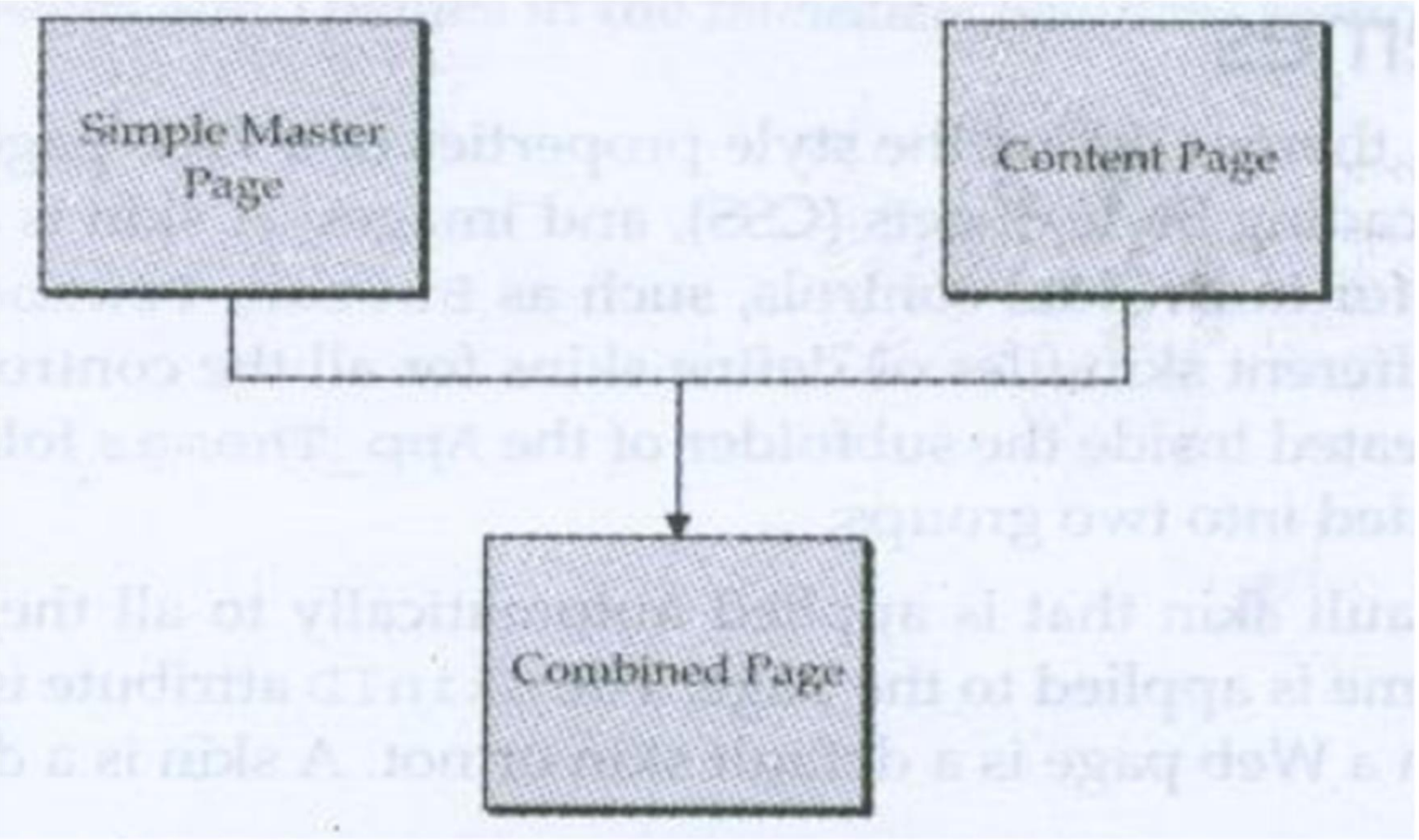
Introduction to ASP.NET Master Page

Master Page Architecture



Simple Master Page



ASP.NET Master Page

- A master page is an ASP.NET file with the extension .master (for example, MySite.master) with a predefined layout that can include static text, HTML elements, and server controls. The master page is identified by a special @ Master directive that replaces the @ Page directive that is used for ordinary .aspx pages. The directive looks like the following.
- Ex. <%@ Master Language="VB" %>
- The @ Master directive can contain most of the same directives that a @ Control directive can contain. For example, the following master-page directive includes the name of a code-behind file, and assigns a class name to the master page.
- Ex. <%@ Master Language="VB" CodeFile="MasterPage.master.vb" Inherits="MasterPage" %>

ASP.NET Master Page

- In addition to the @ Master directive, the master page also contains all of the top-level HTML elements for a page, such as html, head, and form. For example, on a master page you might use an HTML table for the layout, an img element for your company logo, static text for the copyright notice, and server controls to create standard navigation for your site. You can use any HTML and any ASP.NET elements as part of your master page.

Replaceable Content Placeholders

- In addition to static text and controls that will appear on all pages, the master page also includes one or more ContentPlaceHolder controls.
- These placeholder controls define regions where replaceable content will appear.
- In turn, the replaceable content is defined in content pages. After you have defined the ContentPlaceHolder controls, a master page might look like the following.

Master Page

```
<% @ Master Language="VB" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
      "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server" >
      <title>Master page title</title>
</head>
<body>
<form id="form1" runat="server">
<table><tr> <td>
          <asp:contentplaceholder id="Main" runat="server" /></td>
          <td>
          <asp:contentplaceholder id="Footer" runat="server" /></td>
          </tr>
</table>
 </form>
</body>  </html>
```

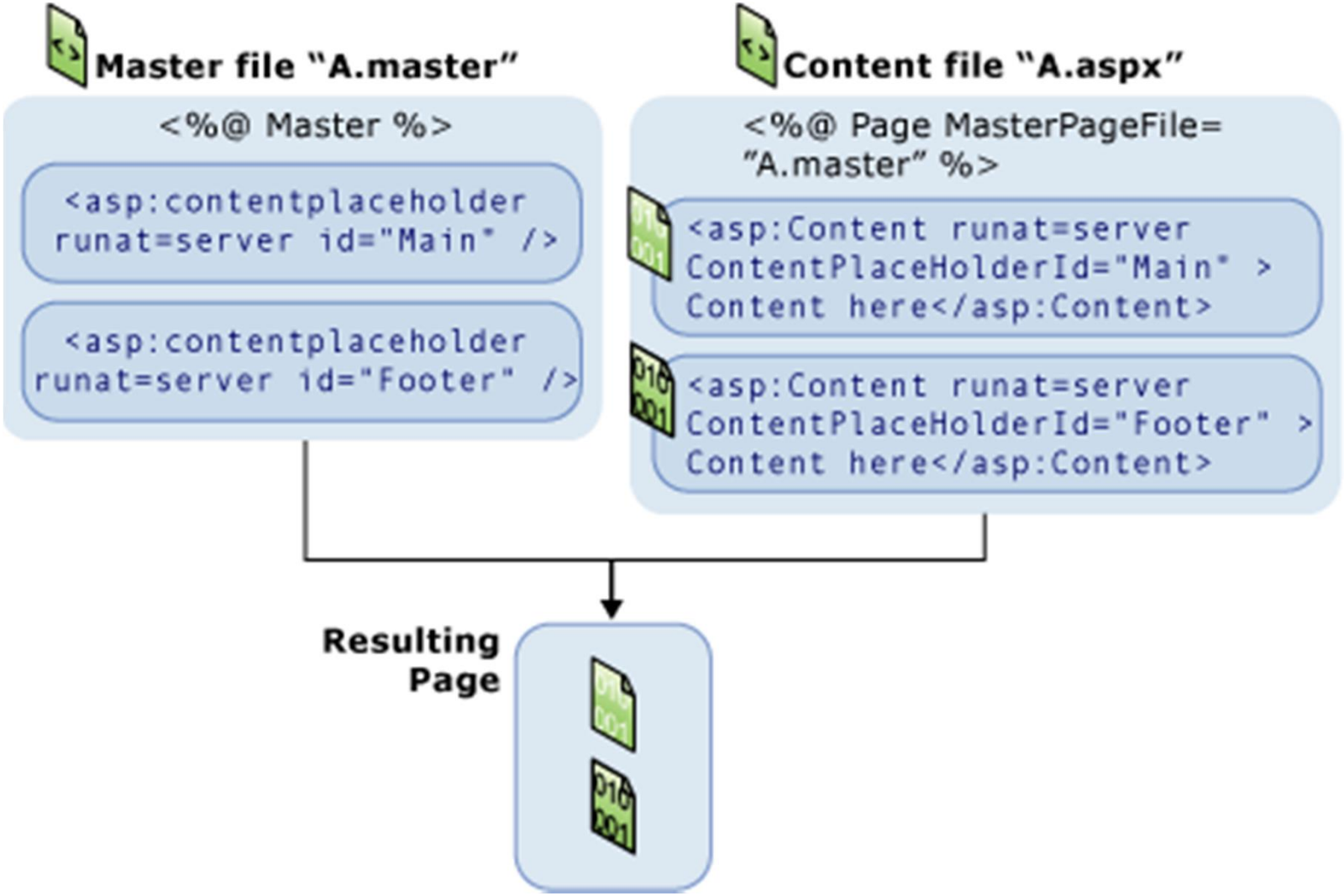
Content Pages

- You define the content for the master page's placeholder controls by creating individual content pages, which are ASP.NET pages (.aspx files and, optionally, code-behind files) that are bound to a specific master page.
- The binding is established in the content page's @ Page directive by including a MasterPageFile attribute that points to the master page to be used. For example, a content page might have the following @ Page directive, which binds it to the Master1.master page.

- <%@ Page Language="VB" MasterPageFile="~/MasterPages/Master1.master" Title="Content Page" %>

Content Pages

- In the content page, you create the content by adding Content controls and mapping them to ContentPlaceHolder controls on the master page.
- For example, the master page might have content placeholders called Main and Footer.
- In the content page, you can create two Content controls, one that is mapped to the ContentPlaceHolder control Main and the other mapped to the ContentPlaceHolder control Footer, as shown in the following figure.


Master file "A.master"
<%@ Master %>
<asp:contentplaceholder runat=server id="Main" />
<asp:contentplaceholder runat=server id="Footer" />
Content file "A.aspx"
<%@ Page MasterPageFile= "A.master" %>
<asp:Content runat=server ContentPlaceHolderId="Main" > Content here</asp:Content>
<asp:Content runat=server ContentPlaceHolderId="Footer" > Content here</asp:Content>
Resulting Page

Content Page

- A content page might look like the following.

```
<% @ Page Language="VB"MasterPageFile="~/Master.master"
        Title="Content Page 1" %>

    <asp:Content ID="Content1" ContentPlaceHolderID="Main"
        Runat="Server"> Main content. </asp:Content>

    <asp:Content ID="Content2" ContentPlaceHolderID="Footer"
        Runat="Server" > Footer content. </asp:content>
```

Content Page

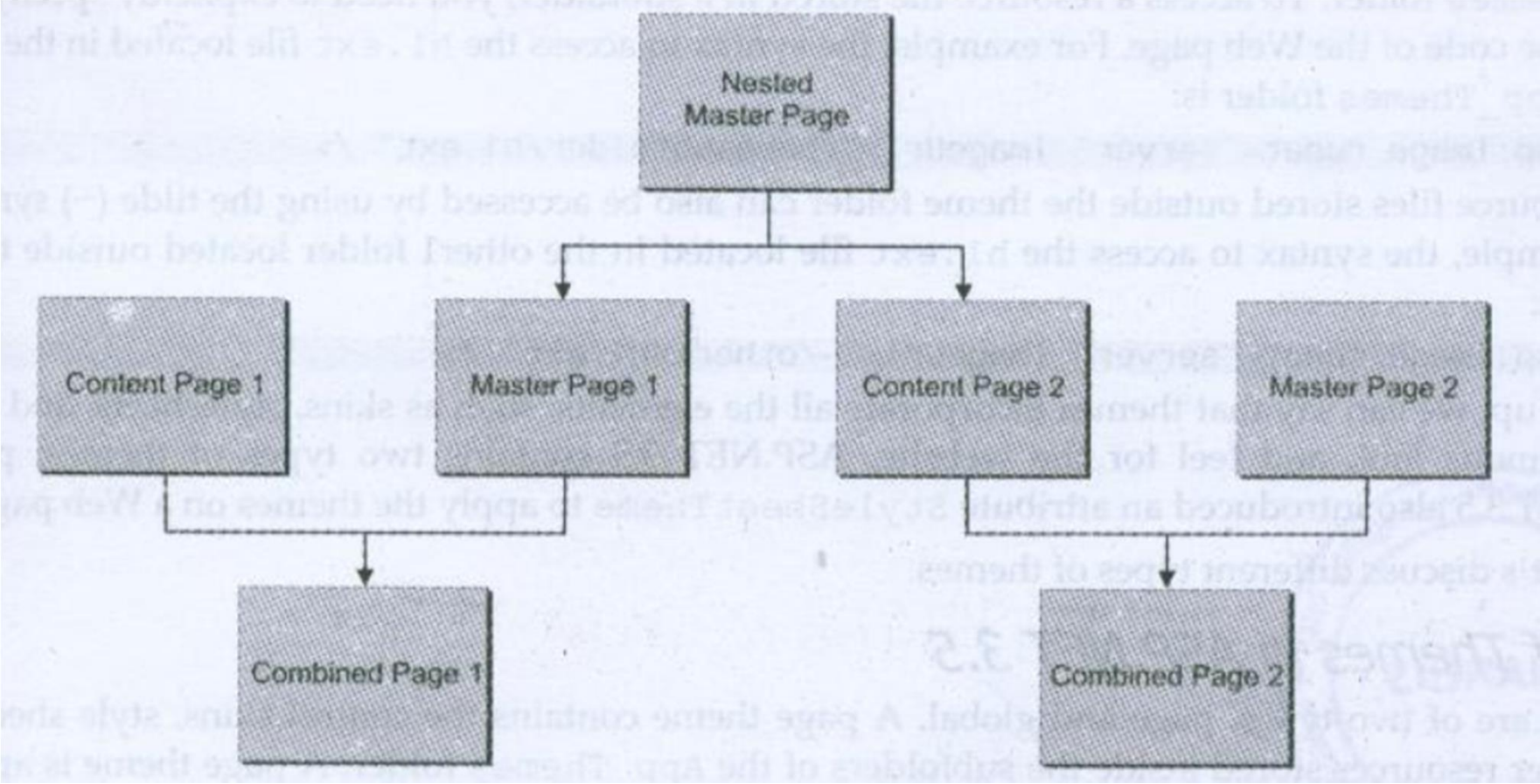
- The @ Page directive binds the content page to a specific master page, and it defines a title for the page that will be merged into the master page.
- Note that the content page contains no other markup outside of the Content controls. (The master page must contain a head element with the attribute runat="server" so that the title setting can be merged at run time.)

Advantages of Master

- PagesMaster pages provide functionality that developers have traditionally created by copying existing code, text, and control elements repeatedly; using framesets; using include files for common elements; using ASP.NET user controls; and so on.

- They allow you to centralize the common functionality of your pages so that you can make updates in just one place.
- They make it easy to create one set of controls and code and apply the results to a set of pages. For example, you can use controls on the master page to create a menu that applies to all pages.
- They give you fine-grained control over the layout of the final page by allowing you to control how the placeholder controls are rendered.
- They provide an object model that allows you to customize the master page from individual content pages.

Nested Master Page



Nested Master Page

- The concept of Nested Master Pages is very much the same as that of Simple Master Page.
- The only difference is that the Nested Master Pages are used in websites that have several hierarchical levels.

Nested Master Page (MasterPage.master)

```
<%@ Master Language="VB" CodeFile="MasterPage.master.vb"
    Inherits="MasterPage" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
    Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

    <html xmlns="http://www.w3.org/1999/xhtml" >
    <head runat="server">
        <title>Untitled Page</title>
    </head>
    <body>
        <form id="form1" runat="server">
            <table cellpadding="0" cellspacing="0“
              style="height:100%;width:100%" border="0"  >
```

Nested Master Page (MasterPage.master)

```
<tr>
    <td valign="top">
        <asp:Image ID="Image0“
          ImageUrl="~/Images/Logo.bmp"
          runat="server" Height="140px" Width="107px"   />
    </td>
    <td valign="top">
         <asp:Image ID="Image1”
          ImageUrl="~/Images/Header.bmp"
          runat="server" Width="837px" Height="140px"   />
    </td>
</tr>
```

Nested Master Page (MasterPage.master)

```
        <tr>
                <td valign="top">
                </td>
                <td valign="top">
                        <asp:contentplaceholder
                                id="ContentPlaceHolder1"  runat="server">
                        </asp:contentplaceholder>
                </td>
        </tr>
        </table>
    </form>
    </body>
</html>
```

Hrd.master

```
<%@ Master Language="VB" MasterPageFile="~/MasterPage.master“
        CodeFile="Hrd.master.vb" Inherits="Hrd" %>
        <asp:Content ID="Content1“
              ContentPlaceHolderID="ContentPlaceHolder1" runat="server">
          <table cellpadding="0" cellspacing="0" border="1">
             <tr>
                <td>
                     <asp:Label ID="Label1" runat="server" Text="Human
                                Resource Department"></asp:Label>
                </td>
             </tr>
             <tr>
                <td>
                   <asp:contentplaceholder id="ContentPlaceHolder1“
                                runat="server">
                   </asp:contentplaceholder>
                </td>
             </tr>
          </table>
      </asp:Content>
```

Default2.aspx

```
<%@ Page Language="VB"
MasterPageFile="~/MasterPage.master"
AutoEventWireup="false" CodeFile="Default2.aspx.vb"
Inherits="Default2" title="Untitled Page" %>


<asp:Content ID="Content1“
ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
</asp:Content>
```