

الفصل الرابع

سلسلة

ASP.NET

خطوة بخطوة حتى الاحتراف

C# & VB

اعداد المهندس

محمد عمر الحاج خلف

الماسٲربيج

في هذا الفصل

- ✓ إنشاء صفحات الماسٲربيج
- ✓ التعديل على محتوى الصفحات الخاضعة لصفحة ماسٲربيج
- ✓ اختيار صفحة الماسٲربيج أثناء التنفيذ

استريبيج بمثابة قالب تخضع له صفحات معينة نحددها خلال التصميم (أو التنفيذ) عند زيارتنا لموقع انترنت ما نجد أن هنالك عناصر عديدة يتم عرضها في معظم Logo فهل من المعقول أن إضافة هذه العناصر المشتركة يتم على كل صفحة على حدى ؟ لا ، ببساطة شديدة يتم تطبيق ذلك التصميم على صفحة الماستريبيج ومن ثم ربط صفحات الموقع بهذه الماستريبيج لتخضع لذلك التصميم ، وبالتالي تصبح صيانة الموقع والتعديل على مظهره مهمة أسهل بكثير حيث إن التعديل على أي عنصر في الماستريبيج سيقابله تعديل تلقائي في جميع صفحات الموقع المرتبطة بهذه الماستريبيج .

سنتعلم في هذا الفصل كيفية التعامل مع صفحة الماستريبيج وكيفية ربط الصفحات بها ، ومن ثم كيفية العادية تأخذ تنسيقات معدلة على الرغم من ارتباطها بصفحة ماستريبيج. وفي النهاية نتعلم كيفية إضافة أكثر من صفحة ماستريبيج وتمكين الزوار من اختيار التصميم الذي يرغبون عرض الموقع به .

إنشاء صفحة الماستريبيج

صفحات الماستريبيج لها اللاحقة Master. ويتم التعامل معها كصفحات الموقع الاعتيادية ذات .aspx ويتم إضافتها من قوائم الفيچوال سنديو :

File ->> New File ->> Master Page ->> Add

حيث سيتم إضافة جديدة للموقع تحمل الاسم MasterPage.master :

```
ASP.net

<%@ Master Language="C#" AutoEventWireup="true" CodeFile="MasterPage.master.cs"
Inherits="MasterPage" %>

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

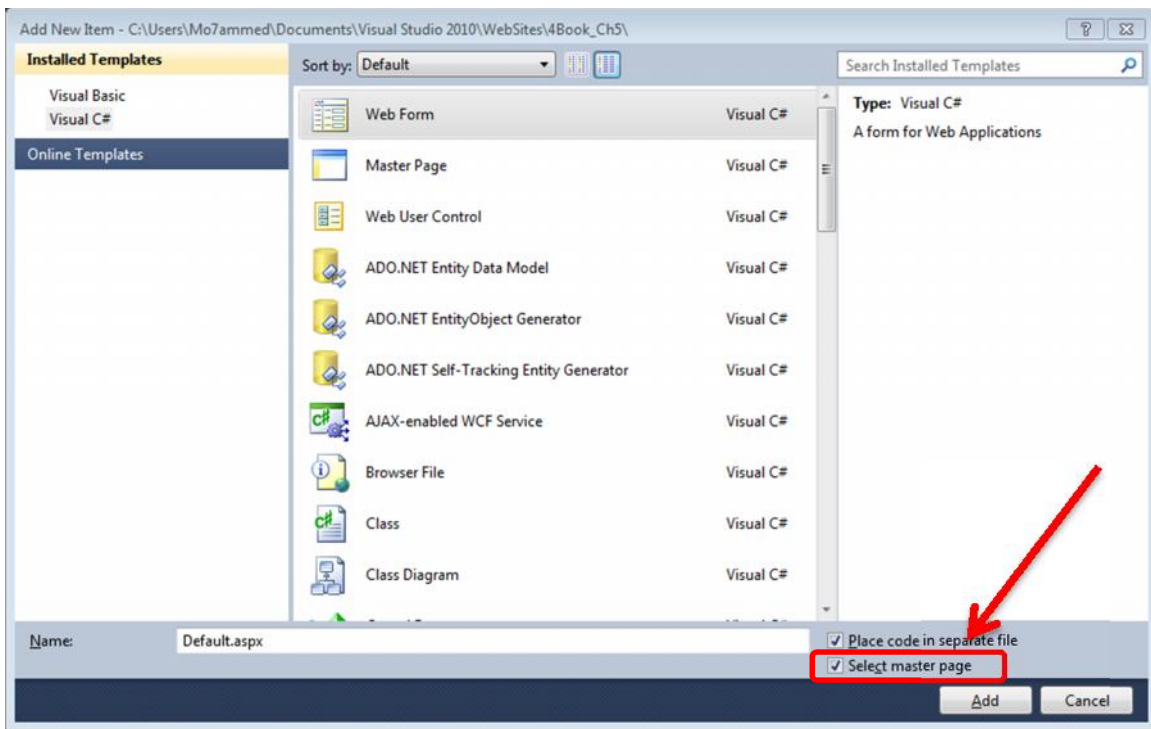
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
  <asp:ContentPlaceHolder id="head" runat="server">
  </asp:ContentPlaceHolder>
</head>
<body>
  <form id="form1" runat="server">
  <div>
    <asp:ContentPlaceHolder id="ContentPlaceHolder1" runat="server">

    </asp:ContentPlaceHolder>
  </div>
</form>
```

```
</body>
</html>
```

ونلاحظ في الكود السابق وجود شيئين مختلفين عن أكواد الصفحات الاعتيادية ، حيث تظهر الترويسة <@ Master > ، والأمر الثاني هو ظهور أدواتي تحكم من النوع ContentPlaceHolder ، حيث إن هذه الأداة تسمح للصفحات التي تخضع للماستريج أن تحتوي أدوات خاصة بها .

لتوضيح النقاط السابقة قم بإضافة صفحة جديدة Default.aspx واربطها مع صفحة الماستريج :



قم بتفعيل الخيار Select master page كما هو واضح في الـ Add ثم قم بتحديد صفحة الماستريج . OK

سيتم إضافة الصفحة Default.aspx :

```
ASP.net
<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage.master"
AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
</asp:Content>
```

```
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
</asp:Content>
```

لاحظ وجود الخاصية MasterPageFile حيث تحدد صفحة الماستريبيج التي ستخضع لها هذه تكون قيمتها مسار نسبي يدل على صفحة الماستريبيج المطلوبة .

الآن لنعود لصفحة الماستريبيج ونجري عليها بعض التعديلات ، قم بإضافة جدول ذات ثلاث أسطر ContentPlaceHolder

نوان للموقع في الخلية الأولى منه لتصبح صفحة الماستريبيج بالشكل :



Default.aspx وتنفيذ الموقع (الضغط على F5) ستجد أن هذه الصفحة قد خضعت لتنسيق صفحة الماستريبيج وعرضت محتوياتها . قم بإيقاف التنفيذ وشاهد تصميم الصفحة Default.aspx حيث سيظهر كما الشكل التالي :



لاحظ كيفية عرض كامل التصميم باللون الرمادي وهذا كناية على أنك لن تستطيع التعديل عليه من هذه الصفحة بل من صفحة الماستريبيج ، والقسم الوحيد الذي تستطيع تعديله هو المساحة ContentPlaceHolder ، حيث إن جميع التعديلات ضمن هذا القسم ستكون خاصة فقط بهذه الصفحة ولن تتأثر باقي الصفحات بها .

للتوضيح أكثر قم بإضافة صفحة أخرى Default2.aspx واجعلها تخضع للماستريبيج السابقة كما

Default2.aspx ContentPlaceHolder

العبارة " الصفحة الثانية " ، وبشكل مشابه في القسم ContentPlaceHolder

Default.aspx " . قم بتنفيذ الموقع وانتقل بين الصفحتين ولاحظ

هما سيعرض تنسيقات الماستريبيج ولكن كل منهما سيعرض العبارة الخاصة به سواءاً كانت

"الصفحة الأولى" أو "الصفحة الثانية" وذلك دون ماستريبيج بهذه المحتويات .

استخدام الصور والروابط التشعبية في صفحات الماستريج

يجب الحذر عند إضافة الصور والروابط التشعبية إلى صفحة الماستريج ، حيث من الممكن تفسير المسارات والعناوين بأشكال مختلفة ، وذلك بالاعتماد على ما هو مستخدم ، أدوات ASP.NET . HTML

Absolute address : يتم فيه ذكر المسار المطلوب بالكامل

Relative address : يتم فيه حساب المسد

وبالتالي لا يتم كتابة المسار

ASP.NET فإن المسارات النسبية المرتبطة بها تفسر انطلاقاً من مسار صفحة

الماستريج ، فعلى فرض أنك أضفت الأداة Image (ASP.NET) الماستريج كالتالي :

ASP.net

```
<asp:Image ID="Image1" runat="server" ImageUrl="~/Images/Picture1.gif" />
```

حيث تم إعطاء الخاصية ImageUrl القيمة ~/Images/Picture1.gif بالتالي يتم تفسير ذلك العنوان على أنه يوجد ضمن نفس المجلد الذي يحتوي صفحة الماستريج (الماستريج والمجلد Images) لموقع التي تخضع لصفحة الماستريج هذه تتوضع في مجلد آخر فلا بأس حيث سيتم تفسير العنوان بيج وليس انطلاقاً من .aspx التي تخضع لها ، وهذا فقط في حال ASP.NET وليس HTML .

HTML فإن الأمر سيختلف تماما ، حيث سيتم تفسير العناوين النسبية

.aspx وليس صفحة الماستريج التي تخضع لها .وبالتالي فإن استخدام

العناوين النسبية في صفحات الماستريج مع أدوات HTML أمر معقد ويتطلب دقة كبيرة ، وبالتالي نريد استخدام العناوين النسبية في صفحات الماستريج ينصح بأحد الأساليب الثلاثة التالية :

أولاً : استبدال جميع الأدوات من النمط HTML ASP.NET حيث يتم تفسير

العناوين بالنسبة لصفحة الماستريج وليس صفحات .aspx وهذا الأمر أسهل بكثير .

ثانياً : في حال الإصرار على استخدام أدوات HTML فينصح بشدة كتابة العناوين الكاملة وليس عناوين نسبية ، فعلى فرض أن الموقع يوجد ضمن مجلد يدعى MyWebSite داخله بمجلد يدعى Images فعندئذٍ نكتب عنوان الصورة المرادة بشكل مشابه للعنوان للتالي :

HTML

```

```

ثالثاً : استخدام الطريقة Page.ResolveUrl() والتي تقوم بتحويل العناوين النسبية إلى عناوين مطلقة (كاملة) حيث يتم استدعاء هذه الطريقة ضمن جسم الوسم :

ASP.net

```
<%@ Master Language="C#" AutoEventWireup="true" CodeFile="MasterPage2.master.cs" Inherits="MasterPage2" %>

<!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></title>
  <asp:ContentPlaceHolder id="head" runat="server">
  </asp:ContentPlaceHolder>
</head>
<body>
  <form id="form1" runat="server">
  <div>
    <img alt="Koala Image" src='<%=Page.ResolveUrl("~/Images/Koala.jpg") %>' />

    <asp:ContentPlaceHolder id="ContentPlaceHolder1" runat="server">
    </asp:ContentPlaceHolder>

  </div>
  </form>
</body>
</html>
```

حيث تم كتابة عنوان نسبي بالنسبة لصفحة الماستريج ، أثناء التنفيذ يتم تحويل ذلك العنوان إلى عنوان مطلق (كامل) وطلب الصورة على أساسه .

الماستريج والملف web.config

بإمكاننا تسجيل صفحة الماستريج ضمن ملف الإعدادات web.config وبالتالي سيتم تطبيق هذه بيج على جميع صفحات الموقع .

XML

```
<configuration>
  <system.web>
    <pages masterPageFile="~/MasterPage.master" />
    <compilation debug="true" strict="false" explicit="true"
      targetFramework="4.0"/>
  </system.web>
</configuration>
```

ملاحظات حول الطريقة السابقة

أولاً : لن يتم تطبيق الماستريج المحددة في الملف web.config Content ، أي يجب يكون لصفحة.aspx شكل شبيه بالتالي :

ASP.net

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default4.aspx.cs"
  Inherits="Default4" %>

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

ثانياً : في حال كانت الصفحة مرتبطة مع صفحة ماستريج عبر الخاصية MasterPageFile هذه الماستريج هي التي سيتم تطبيقها على تلك الصفحة وليس الماستريج المحددة ضمن الملف web.config

التعديل على محتوى الصفحات الخاضعة لصفحة ماستريج

تقتضي فكرة الماستريج أساسا على أن يتم تطبيق تصميم معين على صفحات عديدة ، ولكن هناك بعض الأمور غير فرض علينا عند تطبيق صفحات الماستريج ، على سبيل (Title) ، حيث ستقوم جميع الصفحات التي تخضع للماستريج بعرض

بيج ، في الفقرات التالية سنتعلم كيفية تعديل بعض الخصائص للصفحات الخاضعة للماستريج .

Title

لإظهار عنوان مختلف (صفحة.aspx) عن عنوان الماستريج يكفي ببساطة أن نضيف الخاصية Title للترويصة <%@ Page %> وإعطائها القيمة التي نرغب بظهورها كعنوان لهذه أي تصبح الترويصة شبيهة بالتالي :

ASP.net

```
<%@ Page Title="Page1" Language="C#" MasterPageFile="~/MasterPage.master" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
```

استخدام الخاصية Page Header

C# يمكن كحل بديل للمسألة السابقة ، استخدام الخاصية Page.Header.Title
VB ليتم إسناد العنوان أثناء تحميل الصفحة ، إذاً نضيف لكود تحميد :

C#

```
protected void Page_Load(object sender, EventArgs e)  
{  
    Page.Header.Title = "Page2";  
}
```

VB

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)  
    Handles Me.Load  
  
    Page.Header.Title = "Page2"  
  
End Sub
```

كما أنه بالإمكان التعديل على تنسيقات الصفحة الخاضعة لصفحة ماستريج عبر هذه الخاصية أيضاً. مثلاً لتغيير لون الخلفية لصفحة خاضعة لصفحة ماستريج نضيف الكود التالي في حدث تحميل الصفحة (سنجعل لون الخلفية أحمر) :

C#

```
protected void Page_Load(object sender, EventArgs e)
{
    Style myStyle = new Style();
    myStyle.BackColor=System.Drawing.Color.Red;
    Page.Header.StyleSheet.CreateStyleRule(myStyle,null,"html");
}
```

VB

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    Handles Me.Load

    Dim myStyle As New Style()
    myStyle.BackColor = System.Drawing.Color.Red
    Page.Header.StyleSheet.CreateStyleRule(myStyle, Nothing, "html")

End Sub
```

مات المفتاحية Keywords

Meta و الكامات المفتاحية keywords

لتبحث عن صفحة ما على الانترنت ، ويتم إضافة هذه البيانات ضمن الوسم <Meat> نصرح عنه ضمن الوسم <head> HTML العادية . ولكن كيف نضيف كلمات مفتاحية لصفحة .aspx . خضع لماستريج وهي أساساً لاتحتوي على وسوم <head> ؟ يكمن الحل بإضافتها بشكل برمجي باستخدام C# VB :

قم بالبداية باستدعاء فضاء الأسماء

```
System.Web.UI.HtmlControls;
```

C#

```
protected void Page_Load(object sender, EventArgs e)
{
    // Create Meta Description
    HtmlMeta metaDesc = new HtmlMeta();
    metaDesc.Name = "DESCRIPTION";
    metaDesc.Content = "A sample of using HtmlMeta controls";
}
```

```

// Create Meta Keywords
HtmlMeta metaKeywords = new HtmlMeta();
metaKeywords.Name = "KEYWORDS";
metaKeywords.Content = "HtmlMeta,Page.Header,ASP.NET";

// Add Meta controls to HtmlHead
HtmlHead head = (HtmlHead)Page.Header;
head.Controls.Add(metaDesc);
head.Controls.Add(metaKeywords);
}

```

VB

```

Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
    Handles Me.Load

    ' Create Meta Description
    Dim metaDesc As New HtmlMeta()
    metaDesc.Name = "DESCRIPTION"
    metaDesc.Content = "A sample of using HtmlMeta controls"

    ' Create Meta Keywords
    Dim metaKeywords As New HtmlMeta()
    metaKeywords.Name = "KEYWORDS"
    metaKeywords.Content = "HtmlMeta,Page.Header,ASP.NET"

    ' Add Meta controls to HtmlHead
    Dim head As HtmlHead = CType(Page.Header, HtmlHead)
    head.Controls.Add(metaDesc)
    head.Controls.Add(metaKeywords)
End Sub

```

HTML الناتج عن تنفيذ الصفحة السابقة سنجده يحتوي على الوسم <meta> <head> وفيه الكلمات المفتاحية التي تم تحديدها في الكود السابق :

HTML

```

<meta name="DESCRIPTION" content="A sample of using HtmlMeta controls" />
<meta name="KEYWORDS" content="HtmlMeta,Page.Header,ASP.NET" />

```

اختيار صفحة الماستريج أثناء التنفيذ

من المفيد في حالات عديدة وجود أكثر من ماستريج واحدة ، فمن الممكن مثلا إنشاء أكثر من تصميم وجعل زائر الموقع يختار التصميم الذي يرغب به ليتم عرض صفحات الموقع به . من الممكن أيضاً عرض تصميم ما قبل تسجيل الدخول للموقع ومن ثم عرض تصميم آخر بعد تسجيل

الدخول . يتم تحديد صفحة الماستريج التي تخضع لها الصفحة المطلوبة وذلك في مرحلة مبكرة من دورة حياة الصفحة ، أي قبل أن يتم إطلاق حدث تحميل الصفحة Page_Load نريد تحديد صفحة الماستريج التي سيتم تطبيقها يجب فعل ذلك في حدث مبكر وليكن Page_PreInit وهو أول حدث يتم إطلاقه في دورة حياة الصفحة .

ابدأ مشروع جديد ، أضف صفحتي ماستريج (MasterPage , MasterPage2) منهما ذات تصميم مختلف عن الأخرى .

أضف صفحة جديدة ولتكن Default.aspx واجعلها تخضع للماستريج الأولى MasterPage DropDownList1 وعدل خصائصها كما في الكود :

```

ASP.net

<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage.master"
AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">

<asp:DropDownList ID="DropDownList1" runat="server" AutoPostBack="True"
onselectedindexchanged="DropDownList1_SelectedIndexChanged">
<asp:ListItem Value="MasterPage.master">التصميم </asp:ListItem>
<asp:ListItem Value="MasterPage2.master">التصميم </asp:ListItem>
</asp:DropDownList>

</asp:Content>
    
```

وفي حدث تغيير القيمة المختارة للأداة DropDownList1 نضيف الكود التالي :

```

C#

protected void DropDownList1_SelectedIndexChanged(object sender, EventArgs e)
{
    Session["MasterSelected"] = DropDownList1.SelectedValue;
    Server.Transfer(Request.FilePath);
}
    
```

```

VB

Protected Sub DropDownList1_SelectedIndexChanged(ByVal sender As Object,
ByVal e As System.EventArgs) Handles DropDownList1.SelectedIndexChanged
    
```

```
Session("MasterSelected") = DropDownList1.SelectedValue
Server.Transfer(Request.FilePath)
```

End Sub

تقتضي الفكرة بتخزين اسم صفحة الماستريج التي اختارها المستخدم في ملف جلسة Session تقوم كل صفحة يتم طلبها بقراءة القيمة المخزنة لمعرفة الماستريج الذي سيتم تطبيقه عليها، حيث نقوم بقراءة هذه القيمة في بداية دورة حياة الصفحة انطلاق حدث تحميل الصفحة لذلك نضيف الكود التالي :

C#

```
protected void Page_PreInit(object sender, EventArgs e)
{
    string masterPage;
    masterPage = (string)Session["MasterSelected"];
    if (masterPage != null)
    {
        Page.MasterPageFile = masterPage;
    }
    else
    {
        Page.MasterPageFile = "MasterPage.master";
    }
}
```

VB

```
Protected Sub Page_PreInit(ByVal sender As Object, ByVal e As EventArgs) Handles
    Me.PreInit

    Dim masterPage As String
    masterPage = DirectCast(Session("MasterSelected"), String)
    If masterPage IsNot Nothing Then
        Page.MasterPageFile = masterPage
    Else
        Page.MasterPageFile = "MasterPage.master"
    End If
End Sub
```

قم بتنفيذ المشروع واختر التصميم الذي ترغب به .

إلى هنا نكون قد عرضنا لمحة سريعة عن الاستخدامات الشائعة لصفحات الماستريج ، والتي تفيد وبشكل كبير في تشارك الأدوات عبر صفحات عديدة وهذا ماسنراه بشكل عملي أكثر في فصول قادمة إن شاء الله وعلى وجه الخصوص عند العمل مع أدوات الأجاكس AJAX . وتفيد أيضا صفحات الماستريج في أمور تنسيق صفحات الموقع كما رأينا .

قد تجد بساطة شديدة في الأمثلة المعروضة وهذا أمر متعمد وذلك للتركيز على الفكرة التي يتم طرحها ولك بعدئذ تطوير الأمثلة وإغنائها بإفكارك البرمجية . في الفصل القادم سنتطرق للناحية الجمالية لصفحات الموقع حيث سنتعلم كيفية استخدام الـ ثيمات وملفات التنسيق CSS وغيرها من المواضيع فإلى اللقاء حتى ذلك الحين .

m-hajjhalaf@hotmail.com

<http://www.facebook.com/mohammed.hajkhalaf>

– سوريا