

الفصل الثانى : الوراثة من الصفحة الأم و أدوات الملاحة.
(Master Page and Navigation tools) .

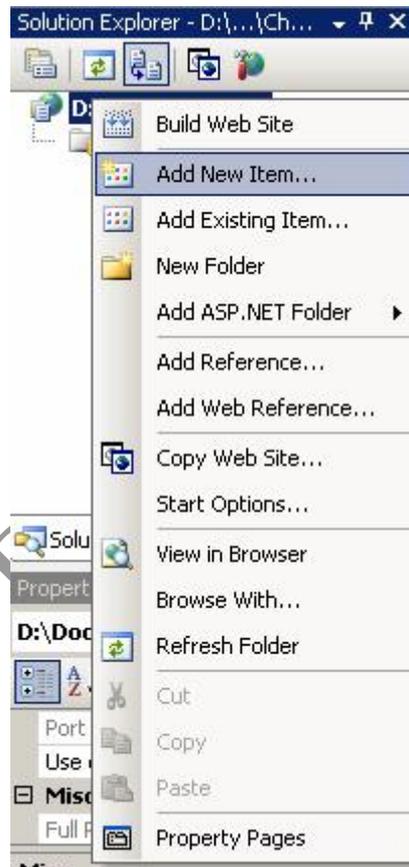
DRAFT BY Ahmed Essam

تعتبر من إحدى المميزات الجديدة و التي سوف تحدث فارقا كبيرا من حيث توفير الوقت و الجهد المبذول في ضبط و إعداد تصميم الصفحات ، ولأن معظم المطورين - على ما أظن - يعانون من التعامل مع التصميم (User interface Design) ؛ حيث كان - هذا الموضوع - يسبب الكثير من الأزعاج للمطورين لإستهلاكه الكثير من الوقت ظهرت هذه الميزة الجديدة - "توريث واجه المستخدم" التي يوفرها (Visual Studio.net 2005) - بحل لهذه المشكلة ، فهي من أهم المميزات التي تساعد و توفر الكثير من الوقت و الجهد .

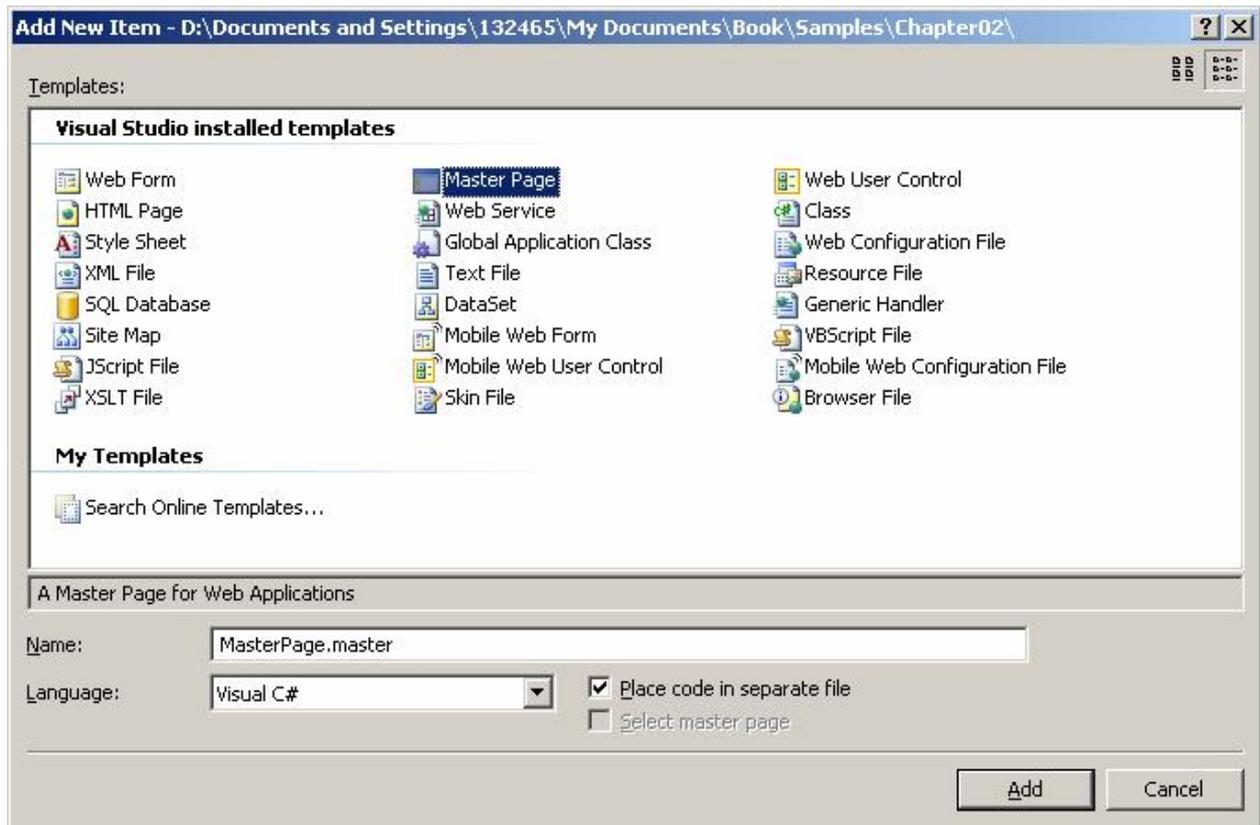
أولا : (Master Page)

الآن سوف إيضاح مثال بسيط للتعامل مع الـ (Master Page) أو ما يسمى بتوريث واجه التعامل . وسيتم تناول مميزات هذه الخاصية وسنبدأ بالتالي :-

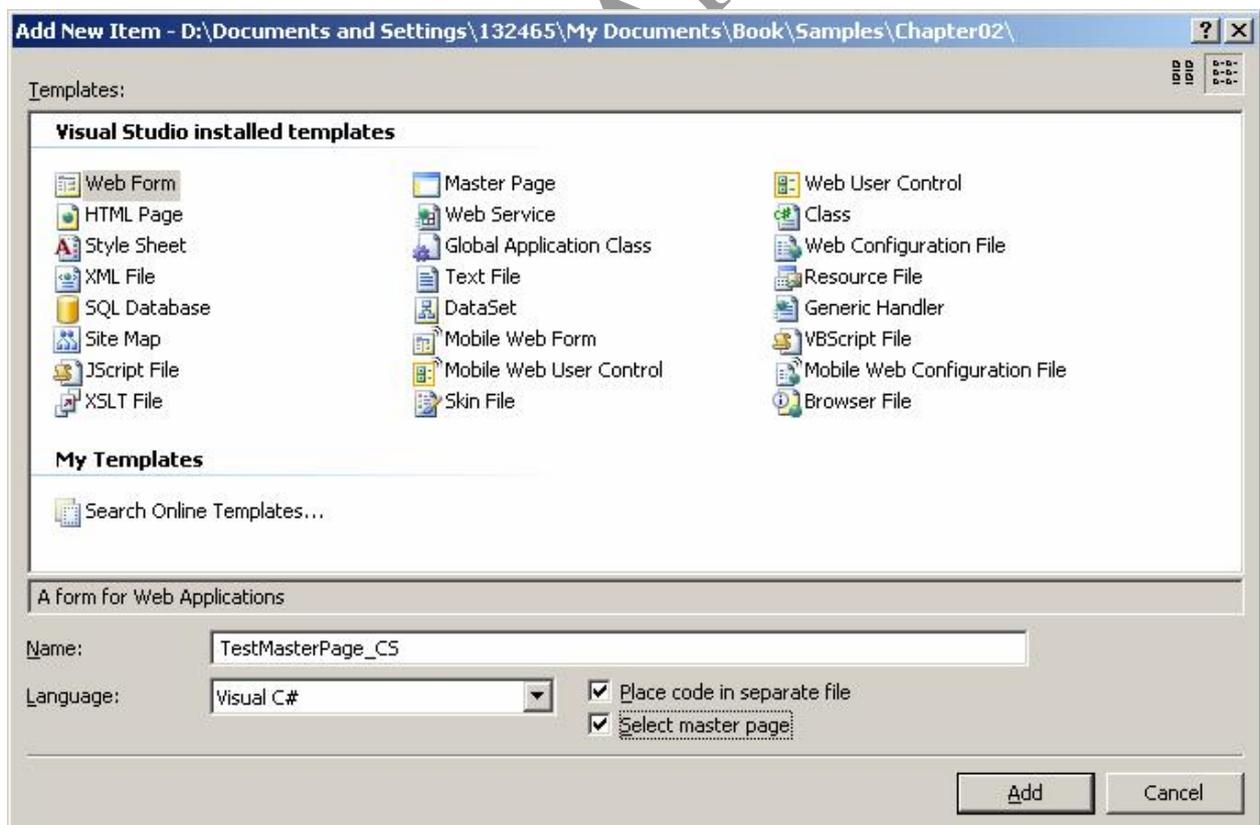
١- مثال صغير لكيفية استخدام الـ (MasterPage) و كيف يمكن تفعيل هذه الميزة الجديدة .



الشكل ١ [نقوم أولا بإضافة عنصر جديد للمشروع]

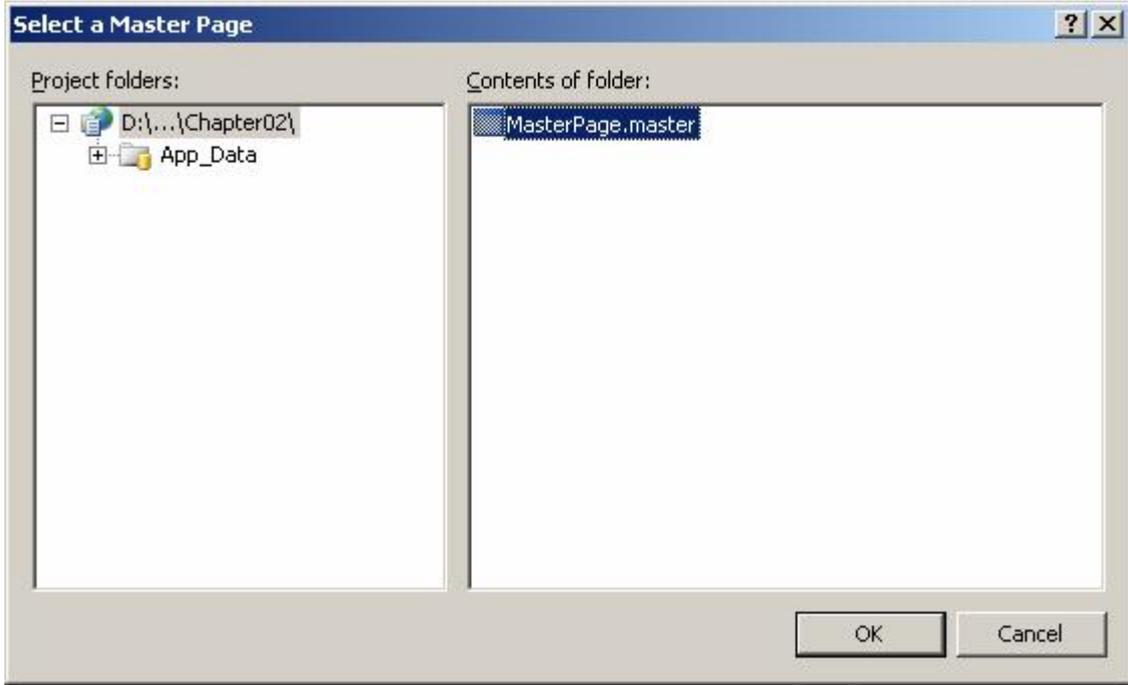


الشكل ٢ [تختار (Master Page) من قائمة القوالب (Templates)]

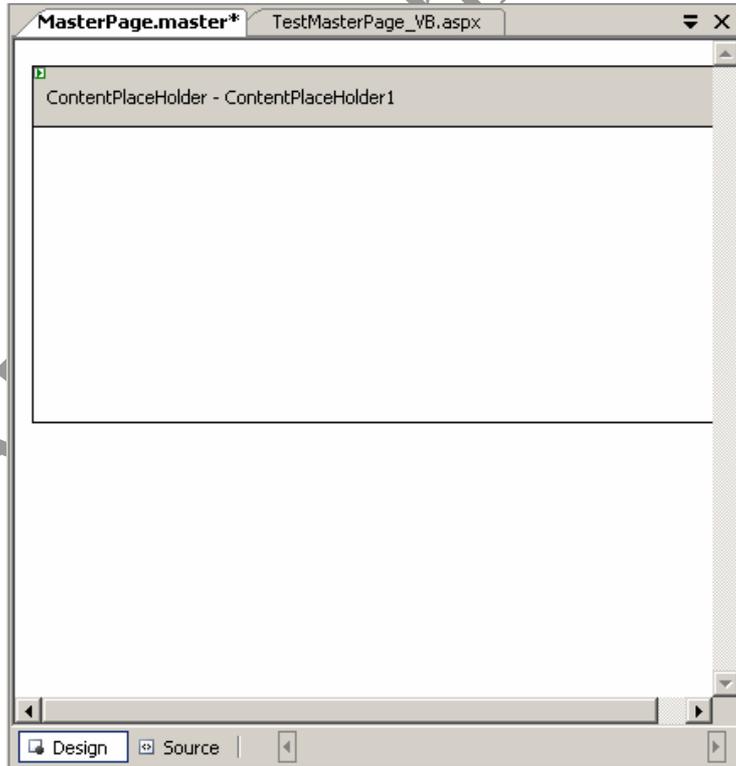


الشكل ٣ [ثم نقوم بعد ذلك بإضافة عنصر جديد هذه المرة سوف نختار (Web Form) من قائمة القوالب و سوف نقوم بتفعيل الخيار

(Select Master Page) بعد ذلك نضغط على الزر (Add)]



الشكل ٤ : [سوف تظهر هذه النافذة و التي سوف تختار منها الصفحة الأم التي نريد ورائة الشكل منها، أختار الصفحة الأم التي تريدها ثم أضغط [(OK)

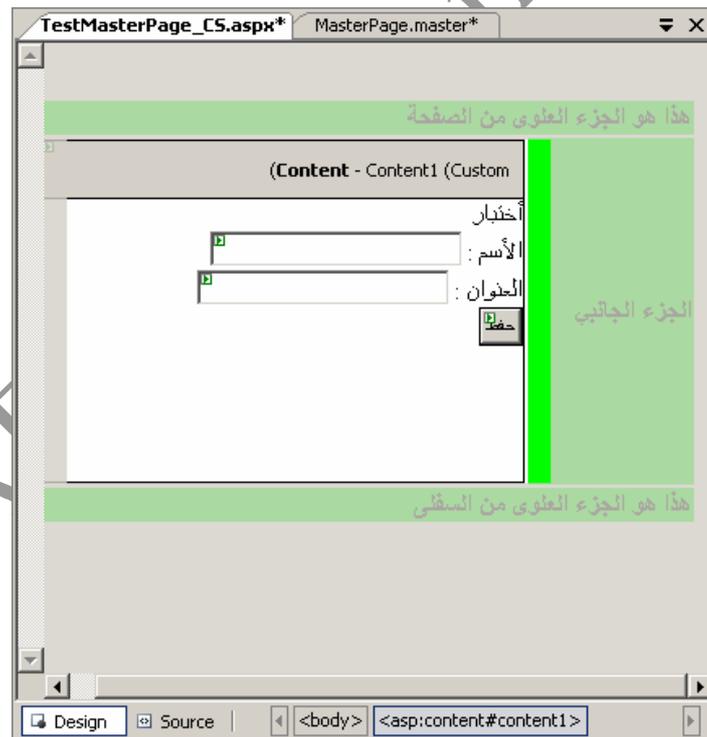


الشكل 5 [هذا هو شكل الـ (Master Page) في حالة التصميم (Design mode) و سوف نلاحظ وجود ما يسمى بالـ (ContentPlaceHolder) و هذا العنصر مهم جدا حيث أنه سوف يحتوى على أى صفحة ترث من الصفحة الأم ، من هنا يتضح لنا كيفية عمل الـ (MasterPage) و هي أن الصفحة الأم صفحة عادية معرفة لدى المترجم بشكل خاص بها (ContentPlaceHolder) و الذى يعتبر همزة الوصل بين الـ (Master Page) و الصفحة التي ترث منها

○ ملاحظة : طريقة عمل الـ (Master Page) مشابهة لحد كبير بخاصية الـ (Master Slide) الموجودة فى الـ (Power Point)



الشكل ٦ [قم برسم جدول و حافظ على وجود العنصر (ContentPlaceholder)]

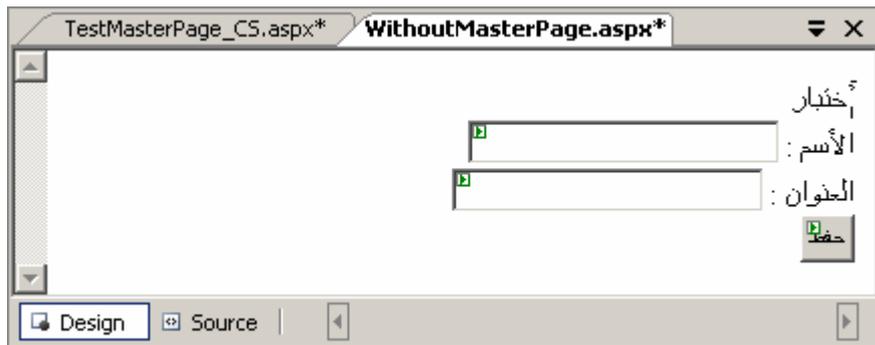


الشكل 7 [قم بفتح الصفحة وسوف يظهر لك الشكل كما هو موضح ، كل ما عليك الآن هو وضع الأدوات و سوف تلاحظ أن الأجزاء الخاصة

بالـ (MasterPage) قد تحول لونها إلى درجة داكنة ؛ مما يعني أنه لا يمكن تعديل هذا الجزء من هذه الصفحة]

و لكن ماذا لو أردنا أن نرث شكل صفحة بدون استخدام معالج الـ (MasterPage) !!!

لاحظ المثال الآتي :-



الشكل ٨ [هذه الصفحة لا ترث من أى صفحة ، قم بفتح الـ (Source) الخاص بها]

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

<!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" dir="rtl">
<head runat="server">
<title>Untitled Page</title>
</head>
<body>
<form id="form1" runat="server">
<div>
أختبار <br />
الأسم :
<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox><br />
العنوان :
<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox><br />
<asp:Button ID="Button1" runat="server" Text="حفظ" /></div>
</form>
</body>
</html>
```

قم بمسح الكود المظلل كما هو موضح فى المقطع السابق

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="WithoutMasterPage.aspx.cs" Inherits="WithoutMasterPage"
MasterPageFile="~/MasterPage.master" %>
<asp:Content ContentPlaceHolderID=ContentPlaceHolder1 ID=ContentHolder
runat=server><div>
أختبار <br />
```

: الأسم

```
<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox><br />
```

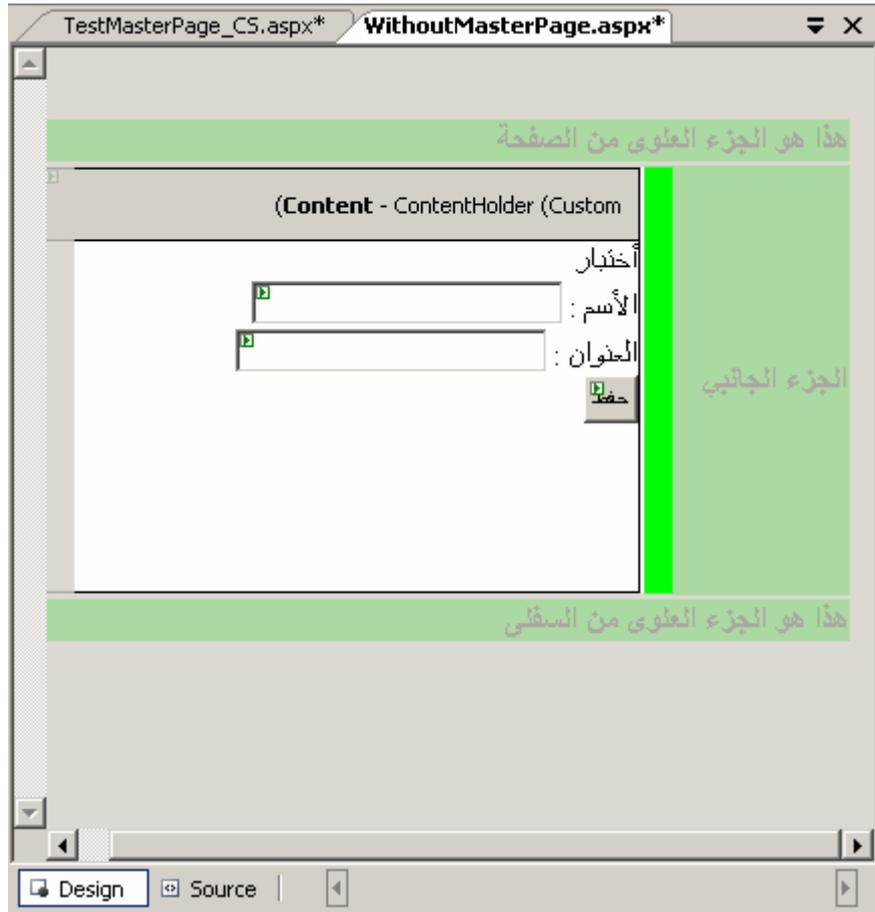
: العنوان

```
<asp:TextBox ID="TextBox2" runat="server"></asp:TextBox><br />
```

```
<asp:Button ID="Button1" runat="server" Text="حفظ" /></div>
```

```
</asp:Content>
```

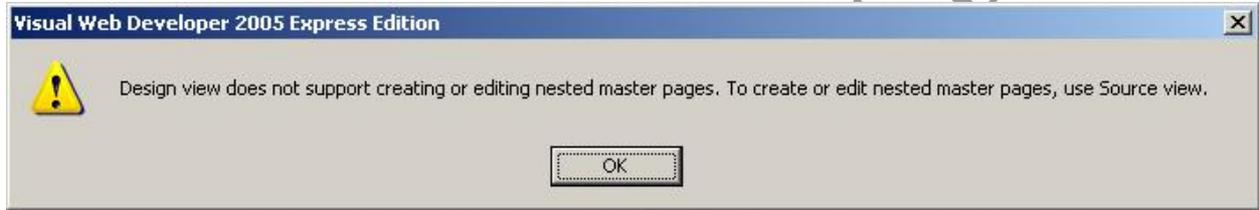
قم بإضافة الكود المظلل و لاحظ الفرق في الشكل التالي :-



الشكل ٩ [هذا هو شكل الصفحة بعد تعديل كود الـ (HTML) ، الآن أصبحت هذه الصفحة ترث من (MasterPage) بدون استخدام المعالج]

٢- إمكانية وراثته شكل من صفحة أم إلى صفحة أخرى

وهي ميزة أخرى مميزه جدا في الـ (MasterPage) ، ويقصد بها (Master Page inherits Master Page) وهذا مفيد في بعض الأحيان ، و لكن إستخدام هذه الخاصية يفقد الـ (VS.net) إمكانية التعامل مع الصفحات التي ترث من الصفحة الأم الوارثة من صفحة أخرى ؛ و سوف تظهر لك رسالة تحذيرية ، كما هو موضح بالشكل [10] .



الشكل 10

بالإضافة الى أنه عند محاولتك لوراثة صفحة أم إلى صفحة أخرى لن تتمكن من استخدام المعالج الخاص بالـ (MasterPage) و سوف تضطر إلى تعديل الكود يدوياً كالتالي :-

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

```
<asp:Content ContentPlaceHolderID=ContentPlaceHolder1 ID=test runat=server >
<div>
<asp:contentplaceholder id="ContentPlaceHolder1" runat="server">
</asp:contentplaceholder>
</div>
</asp:Content>
```

تم الإيضاح سابقاً إلى كيفية عمل وراثته للصفحات يدوياً بدون استخدام المعالج .

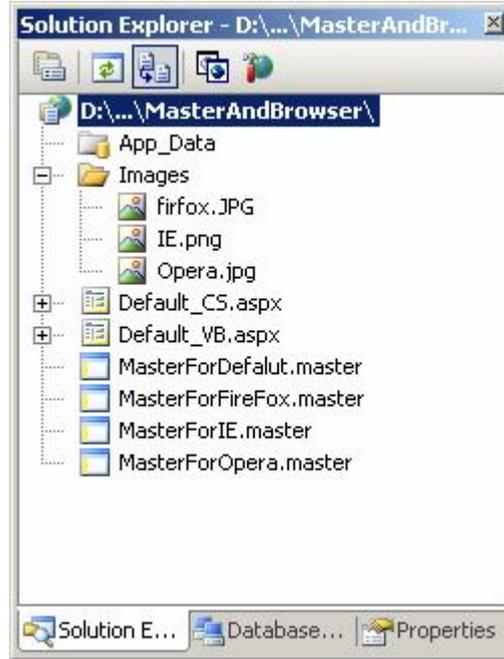
أيضا يمكن عمل (Master Page) افتراضية ، و التي يتم إختيارها في حالة عدم إختيارك لـ (Master Page) ويتم عملها عن طريق إضافة السطر التالي تحت (system.web) في ملف الـ (Web.Config) .

```
<pages masterPageFile="~/WholeSite.master"/>
```

هذا السطر سوف يوفر عليك عناء إختيار الصفحة و تحديدها و ما الى ذلك ؛ حيث أن هذا الخيار سوف يتيح لك ترك (MasterPageFile) فارغة بدون قيمة في موجة الصفحة (Page Directive) كما هو في السطر التالي

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

و الآن لنعرض كيفية تغيير الصفحة الأم في لحظة عرض الصفحة الأساسية.
§ قم ببدأ مشروع جديد و أضف إليه الملفات كما هو موضح في الشكل



الشكل ١١ [في الملفات (Default) قم باختيار اللغة حسب ما تريد VB أو C#]

لا بد و أن تكون الصفحات (Default) ترث الشكل من (MasterPage) و هي موجودة بأسم (MasterForDefault)

§ و الان نقوم بعمل ضبط للصفحة الأم كما سبق بطريقة بسيطة جدا ، و نضيف الكود التالي ليقوم بتغيير الصفحة الأم حسب نوع المستعرض .

```
protected void Page_PreInit(object sender, EventArgs e)  
{  
    switch(Request.Browser.Browser.ToLower())  
    {  
        case "ie":  
            MasterPageFile = "MasterForIE.master";  
            break;  
        case "opera":  
            MasterPageFile = "MasterForOpera.master";  
            break;  
        case "firefox":  
            MasterPageFile = "MasterForFireFox.master";  
            break;  
    }
```

```
default :  
    MasterPageFile = "MasterForDefault.master";  
    break;  
}  
}
```

الكود بصيغة الـC#

```
Protected Sub Page_PreInit(ByVal sender As Object, ByVal e As  
System.EventArgs) Handles Me.PreInit  
    Select Case Request.Browser.Browser.ToLower()  
        Case "ie"  
            MasterPageFile = "MasterForIE.master"  
        Case "opera"  
            MasterPageFile = "MasterForOpera.master"  
        Case "firefox"  
            MasterPageFile = "MasterForFireFox.master"  
        Case Else  
            MasterPageFile = "MasterForDefault.master"  
    End Select  
End Sub
```

الكود بصيغة الـVisual Basic

هذا الكود يقوم بتغيير الصفحة الأم قبل تحديد أي صفحة أو سيتم إختيارها ، هذا طبعا بعد معرفة نوع المستعرض .



الشكل 12 [في حالة المستعرض (Opera)]



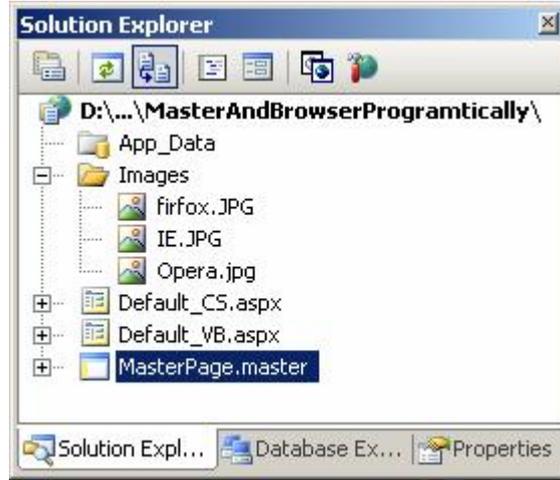
الشكل 13] في حالة المستعرض (FireFox)



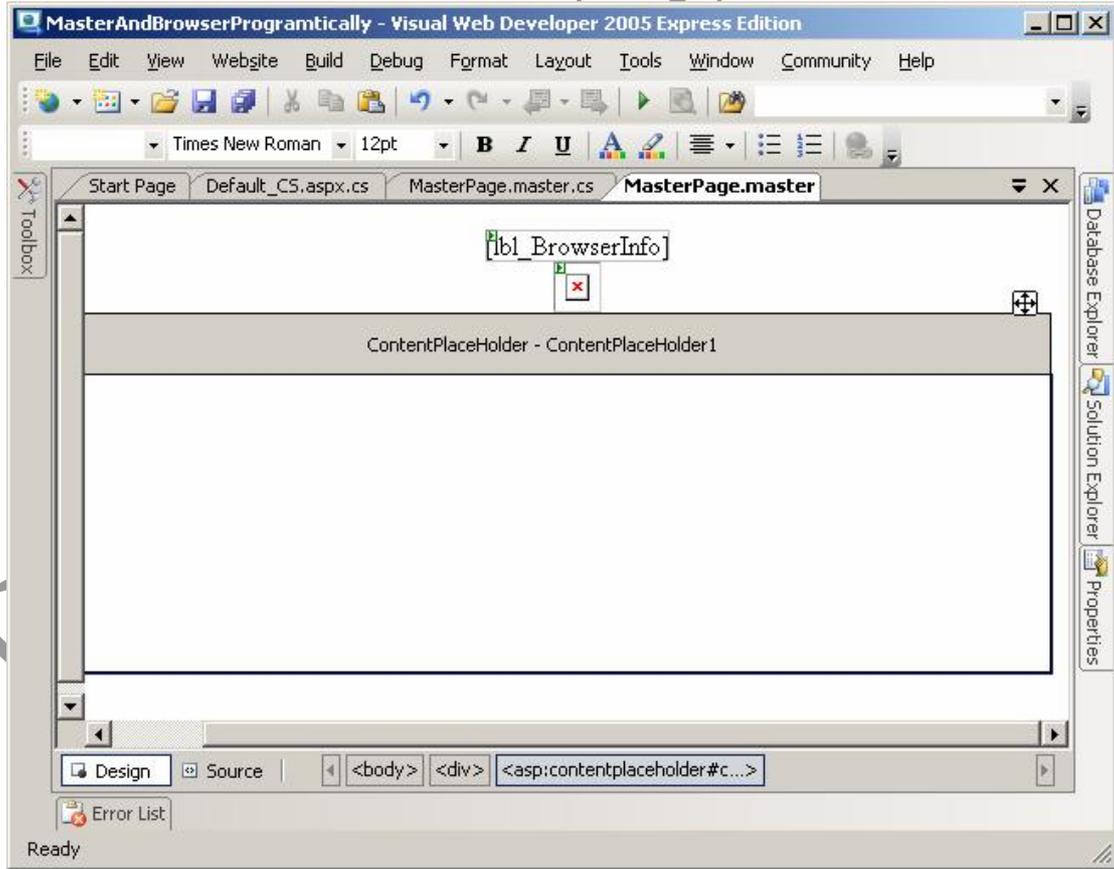
الشكل 14] في حالة استخدام (Internet Explorer)

أرى أننا في المثال السابق صعبنا الأمور على أنفسنا بعض الشيء فما رأيكم أن نقوم بتعديل الـ (MasterPage) برمجياً عوضاً عن التعامل بهذا الشكل ، وهو أن نقوم بعمل صفحة أم لكل مستعرض - الآن نعرض المثال .

قم ببدأ مشروع جديد و أضف اليه الملفات كما هو موضح في الشكل



الشكل ١٥ [أبدأ مشروع جديد و ضع به الملفات الموجودة في الشكل]



الشكل 16 [في الصفحة الأم قم بإضافة (Label) و (Image) لمنطقة العرض و بعد ذلك قم بتغيير أسم الـ (Label) إلى (lbl_BrowserInfo) و قم بتغيير أسم الـ (Image) إلى (mg_browserImage)]



الشكل ١٧ [قم بالدخول على الصفحة (Default_xx) - xx تعنى حسب اللغة التى تستخدمه - قم بأضافة زر و غير الـ (Text) الخاص به كما هو موضح]

```
protected void Button1_Click(object sender, EventArgs e)
{
    Image imgHolder = (Image)Master.FindControl("img_BrowserImage");
    هنا نقم بالحصول على العنصر Image فى الصفحة الأم و وضعة فى متغير من نفس النوع حتى
    يتسنى لنا التعامل معه مثل باقى العناصر.
    Label lblHolder = (Label)Master.FindControl("lbl_BrowserInfo");
    هنا نقم بالحصول على العنصر Label فى الصفحة الأم و وضعة فى متغير من نفس النوع حتى
    يتسنى لنا التعامل معه مثل باقى العناصر.
    imgHolder.ImageUrl = "~/\\Images\\" + Request.Browser.Browser + ".jpg";
    نقوم بتغيير مسار الصورة الى الصورة حسب نوع المستعرض و ذلك بدمج مسار مجلد الصور مع
    نوع المستعرض بالأضافة الى أمتداد الصورة
    lblHolder.Text = string.Format("<pre>{0}</pre>", Request.Browser.Browser);
    عرض أسم المستعرض
}

```

الكود بصيغة C#

```
Protected Sub Button1_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button1.Click

```

```
Dim imgHolder As Image = CType(Master.FindControl("img_BrowserImage"), Image)
```

هنا نقوم بالحصول على العنصر Image في الصفحة الأم و وضعة في متغير من نفس النوع حتى يتسنى لنا التعامل معه مثل باقى العناصر.

```
Dim lblHolder As Label = CType(Master.FindControl("lbl_BrowserInfo"), Label)
```

هنا نقوم بالحصول على العنصر Label في الصفحة الأم و وضعة في متغير من نفس النوع حتى يتسنى لنا التعامل معه مثل باقى العناصر.

```
imgHolder.ImageUrl = "~\Images\" + Request.Browser.Browser + ".jpg"
```

نقوم بتغيير مسار الصورة الى الصورة حسب نوع المستعرض و ذلك بدمج مسار مجلد الصور مع نوع المستعرض بالأضافة الى امتداد الصورة

```
lblHolder.Text = String.Format("<pre>{0}</pre>", Request.Browser.Browser)
```

عرض أسم المستعرض

End Sub

الكود بصيغة VB.net



الشكل 18 شكل ناتج المثال على مستعرض أنترنت أكسلورار و ذلك قبل الضغط على "عرض بيانات المستعرض"



الشكل 19 شكل ناتج المثال على مستعرض أوبرا و ذلك قبل الضغط على "عرض بيانات المستعرض"



الشكل 20]شكل ناتج المثال على مستعرض فايرفوكس و ذلك قبل الضغط على "عرض بيانات المستعرض" [



الشكل 21]شكل ناتج المثال على مستعرض إنترنت إكسبلورار و ذلك بعد الضغط على "عرض بيانات المستعرض" [



الشكل 22 شكل ناتج المثال على مستعرض أوبرا و ذلك بعد الضغط على "عرض بيانات المستعرض"

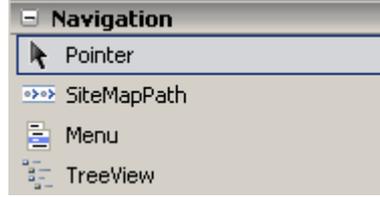


الشكل 23 شكل ناتج المثال على مستعرض فايرفوكس و ذلك بعد الضغط على "عرض بيانات المستعرض"

و الآن و بعد الانتهاء من سرد كيفية التعامل مع الـ (Master Page) ننقل الى كيفية الربط و التعامل مع الـ (Navigation tools)

ثانيا : أدوات الملاحة (Navigation Tools)

و هي من أحد المميزات التي توفر الكثير من الوقت و الجهد ، و الآن لنعرض أدوات الملاحة الجديدة في (VS.net 2005) و هي موضحة في الشكل [24]

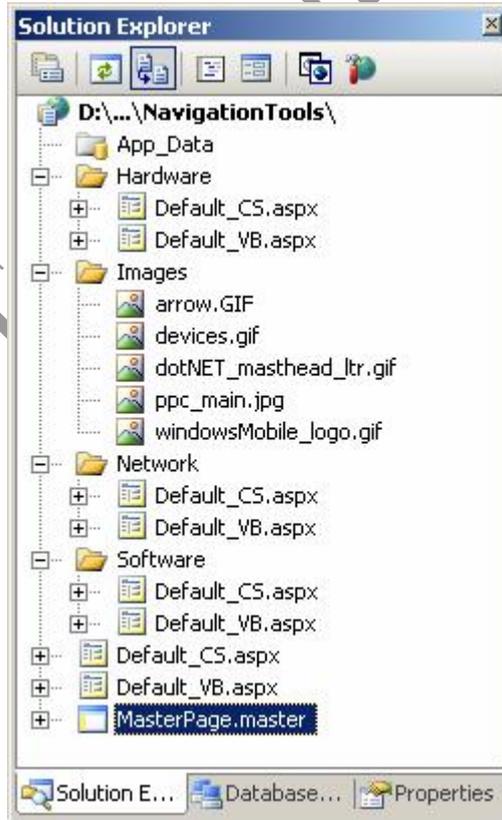


الشكل 24

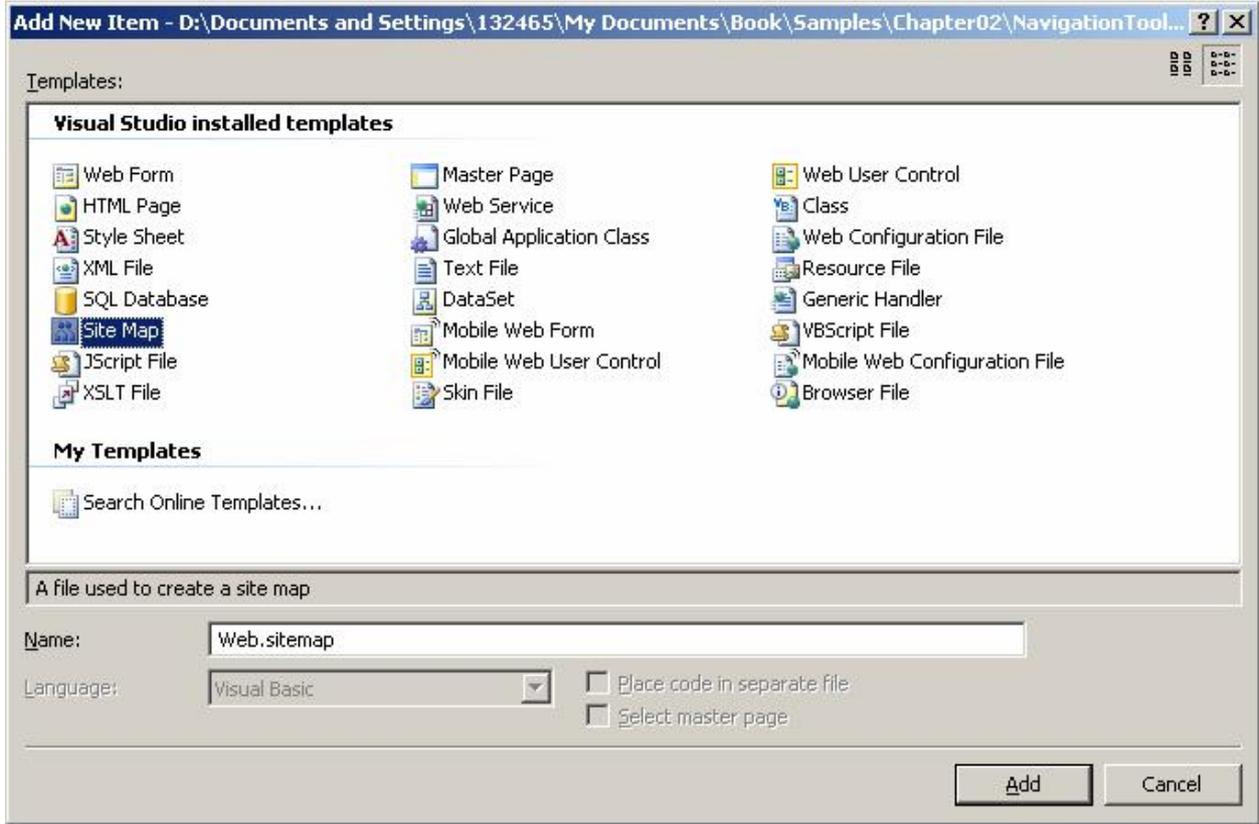
والآن سوف نشرح هذه الأدوات واحدة تلو الأخرى - فلا تقلق- وسوف يرفق كالعادة أمثلة توضح كيفية التعامل مع هذه الأدوات

- و الآن لنبدأ بالمثل الأول مع الأداة (SiteMapPath) و (Menu) حيث كلا منهم يكمل الآخر، ليس بالضرورة أن تستعمل الأداة في نفس الوقت ، ولكن كشكل أو عرض يفضل وجودهما معا - هذا من وجه نظري الشخصية- و الآن لنعرض المثل .

أولا نقوم بإنشاء مشروع جديد و إضافة ملفات له كما هو موضح في الشكل التالي [25]



الشكل ٢٥



الشكل ٢٦]تقوم بإضافة عنصر جديد و هو (Site Map) من القوالب الموجودة (Templates)

فائدة ملف الـ (Site Map) :
هو الدليل الذي يكون به جميع وصلات المشروع ، أو كما يسمى خارطة الموقع بعد إختيارك للملف (Site Map) سوف تحتاج إلى إدخال بيانات كل وصلة في الموقع الخاص بك و هي كالآتي :-
- (URL) و فيها عنوان الصفحة
- (Title) العنوان الذي سوف يظهر
- (Description) وصف مبسط عن الوصلة و ما تحتوية و الوصف يظهر في شكل (Tip Roles) و فيها يحدد من له الحق في أن يرى هذه الوصلة (سيتم شرح هذه النقطة في الفصل الخاص بالأمن).
بعد أن قمت بإضافة الملف قم بإضافة الوصلات الخاصة بموقعك كما هو الحال في الشكل التالي .

```

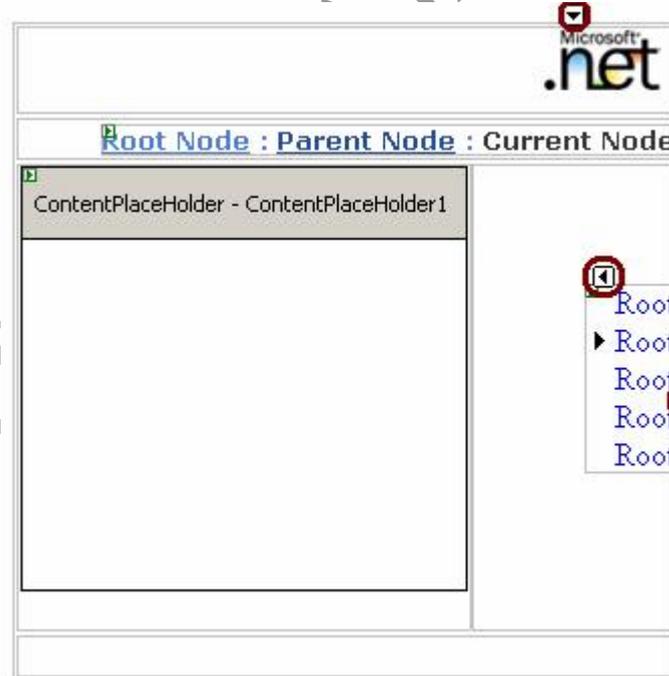
Web.sitemap*
<?xml version="1.0" encoding="utf-8" ?>
<siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0" >
  <siteMapNode url="Default_cs.aspx" title="الرئيسية" description="صفحة الموقع الرئيسية">
    <siteMapNode url="~/Hardware/Default_CS.aspx"
      title="العتاد الصلب"
      description="لمتخصصي العتاد الصلب و الأدوات الخاص بالحاسب">
      <siteMapNode url="~/Hardware/Default_CS.aspx?Category=MB"
        title="كل ما يتعلق بالوحة الأم" description="كل ما يتعلق بالوحة الأم" />
      <siteMapNode url="~/Hardware/Default_CS.aspx?Category=PR"
        title="كل ما يتعلق بالمعالج" description="كل ما يتعلق بالمعالج" />
    </siteMapNode>

    <siteMapNode url="~/Software/Default_CS.aspx"
      title="البرمجيات"
      description="للمبرمجين و المطورين">
      <siteMapNode url="~/Software/Default_CS.aspx?Category=Op" title="نظم التشغيل" />
      <siteMapNode url="~/Software/Default_CS.aspx?Category=Bu" title="برامج أعمال" />
      <siteMapNode url="~/Software/Default_CS.aspx?Category=Ga" title="ألعاب" />
      <siteMapNode url="~/Software/Default_CS.aspx?Category=Ut" title="برامج خدامية" />
    </siteMapNode>

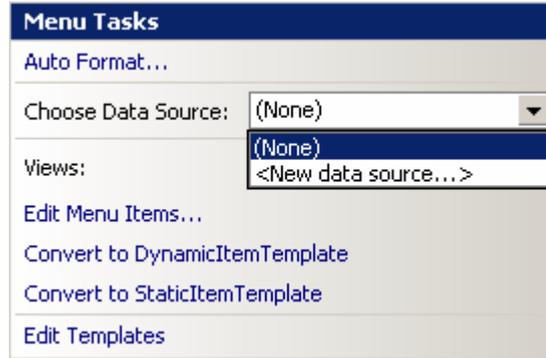
    <siteMapNode url="~/Network/Default_CS.aspx"
      title="شيكات"
      description="للمبرمجين و المطورين">
      <siteMapNode url="~/Network/Default_CS.aspx?Category=ViOp" title="شيكات الألياف الضوئية" />
      <siteMapNode url="~/Network/Default_CS.aspx?Category=VOIP" title="الهاتف خلال الأترنت" />
      <siteMapNode url="~/Network/Default_CS.aspx?Category=WiLe" title="شيكات لا سلكية" />
    </siteMapNode>
  </siteMapNode>
</siteMap>

```

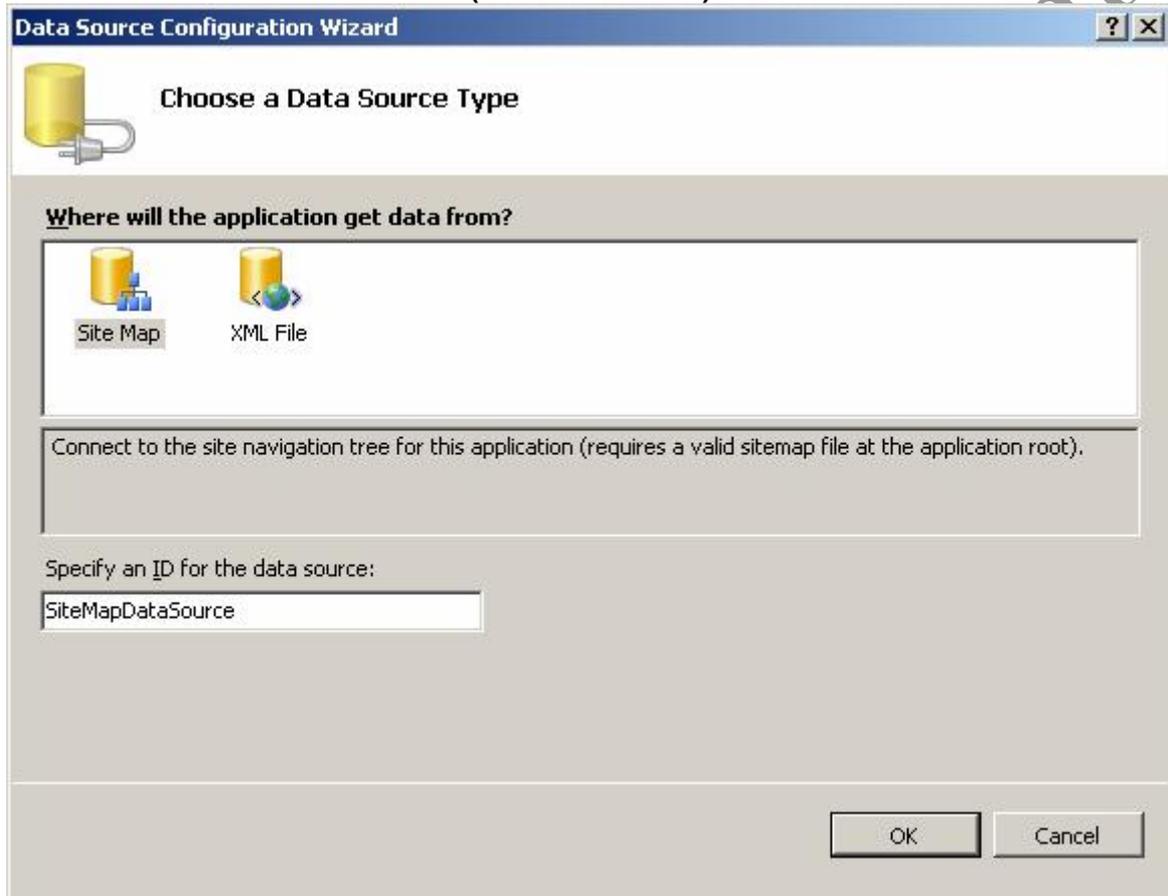
الشكل ٢٧



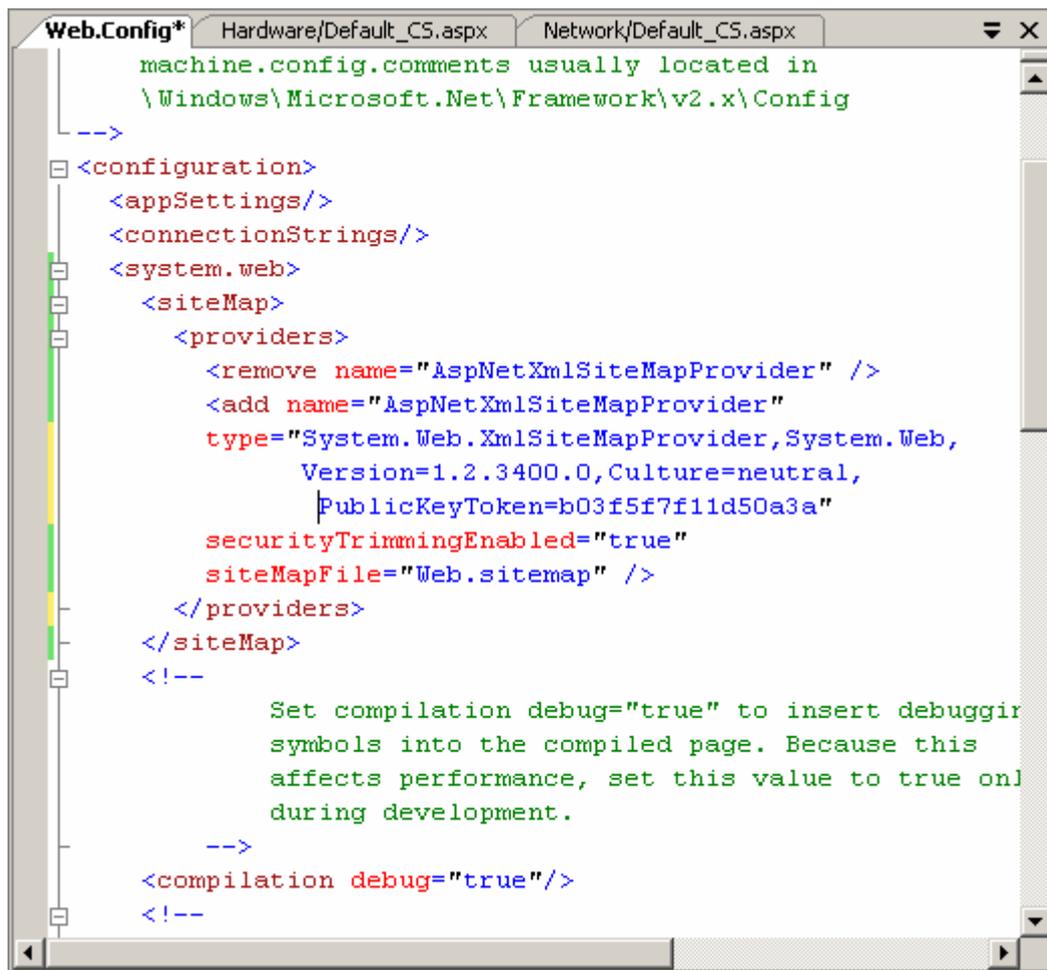
الشكل ٢٨] يعد ذلك نقوم بإضافة بعض العناصر للصفحة الأم (Master Page) و هي عبارة عن جدول وبه (Menu و Site Map Path) (Content Place Holder) ، الدوائر الحمراء تشير الى ما يسمى بالمساعد الذكي (Smart Tag) حيث إن الضغط عليها تظهر لك قائمة بأهم الخواص للعنصر التابع له]



الشكل ٢٩] عند الضغط على الـ (Smart Tag) الخاص بالـ (Menu) كما في الشكل السابق نقوم الآن بتحديد مصدر البيانات للقائمة ونختار (New Data source)

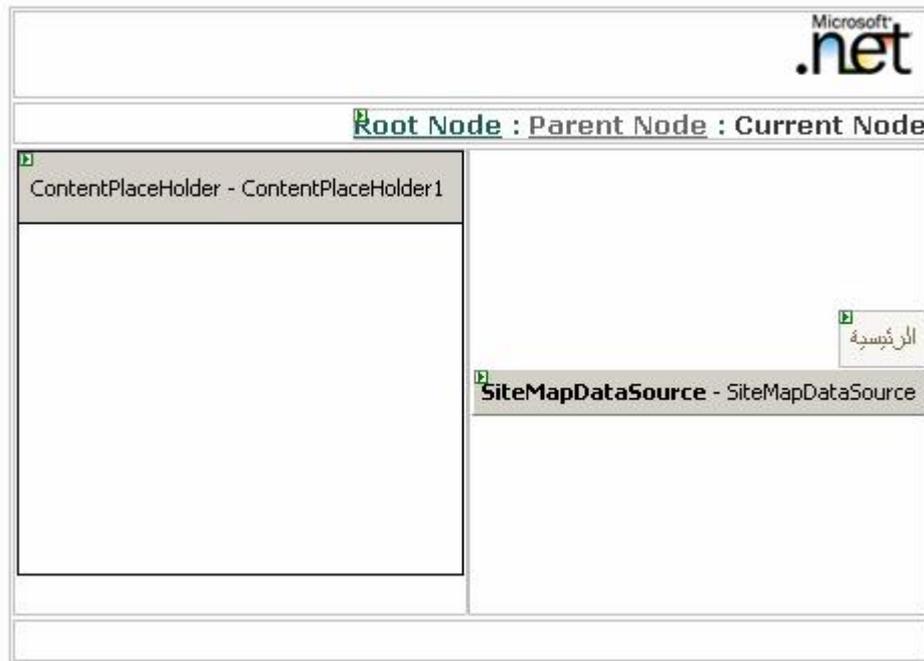


الشكل ٣٠] عند الاختيار سوف تظهر هذه النافذة و التي تختار منها نوع مصدر البيانات التي سوف تستخدمها القائمة و سوف نختار الـ (Site Map)

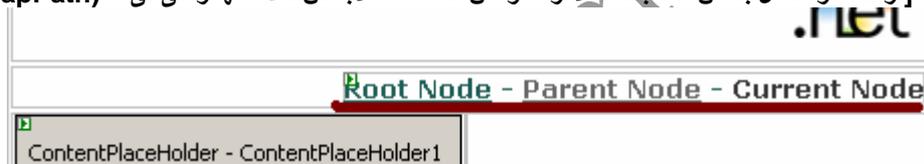


```
Web.Config* Hardware/Default_CS.aspx Network/Default_CS.aspx
machine.config.comments usually located in
\Windows\Microsoft.Net\Framework\v2.x\Config
-->
<configuration>
  <appSettings/>
  <connectionStrings/>
  <system.web>
    <siteMap>
      <providers>
        <remove name="AspNetXmlSiteMapProvider" />
        <add name="AspNetXmlSiteMapProvider"
            type="System.Web.XmlSiteMapProvider, System.Web,
                Version=1.2.3400.0, Culture=neutral,
                PublicKeyToken=b03f5f7f11d50a3a"
            securityTrimmingEnabled="true"
            siteMapFile="Web.sitemap" />
      </providers>
    </siteMap>
  <!--
    Set compilation debug="true" to insert debuggiri
    symbols into the compiled page. Because this
    affects performance, set this value to true onl
    during development.
  -->
  <compilation debug="true"/>
  <!--
```

الشكل ٣١] بعد ذلك نقوم بالدخول على الملف (Web.Config) و سنضيف بعض السطور به، و هي لتعريف ما يسمى بالمخصص (Provider) و هنا يتم تعريف بعض المكتبات بالإضافة إلى معلومات أخرى الشئ المتغير أو الحي فيها هو الاسم (Name) و الملف الخارطة (sitemapfile) كما نرى في الشكل]

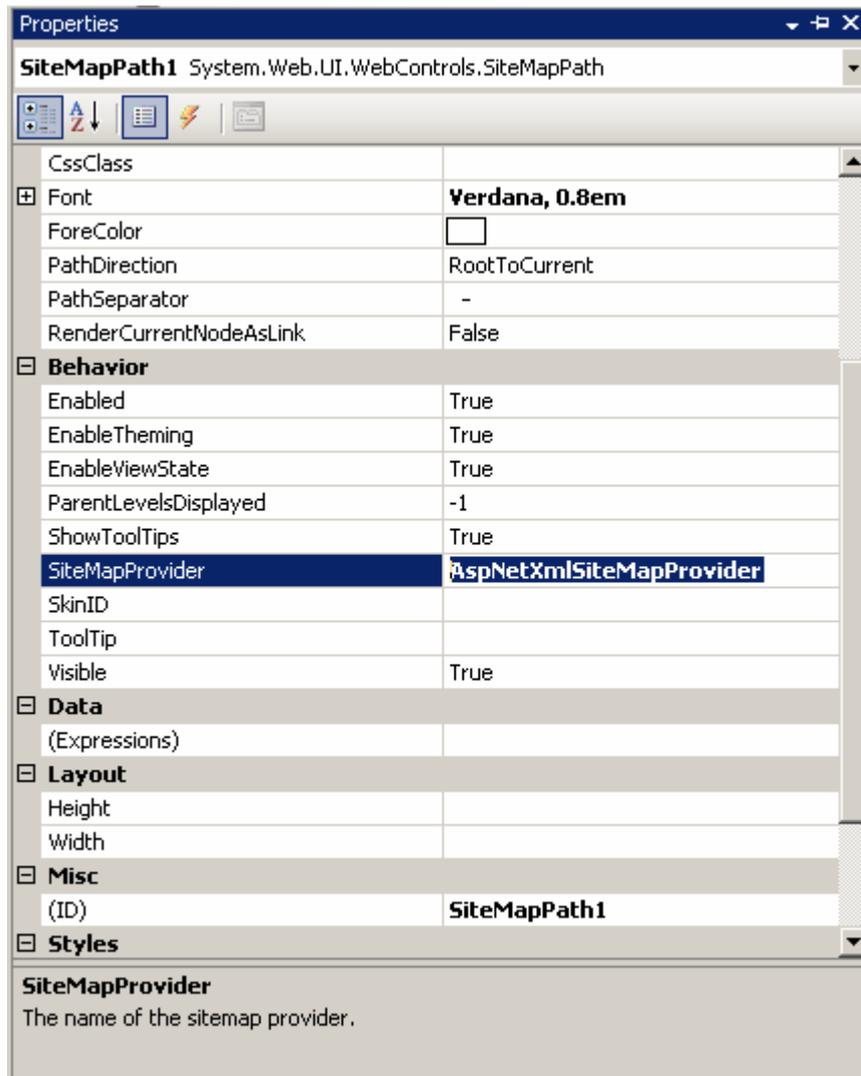


الشكل ٣٢] و هذا هو الشكل بعد أن قمنا بأعداد الأدوات و لكن هناك نقطة لا بد من ملاحظتها و هي في الـ (siteMapPath)



الشكل ٣٣] كما نرى نقوم بتفعيل العنصر (siteMapPath)

DRAFT BY

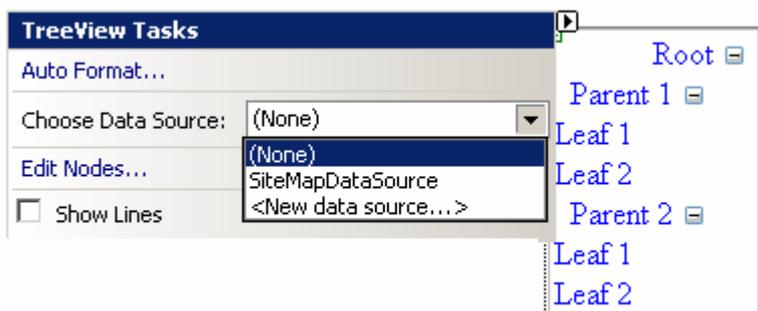


الشكل ٣٤ [تقوم هنا بأدخال أسم المخصص الذي أدخلناه في ملف الـ (Web.config)]

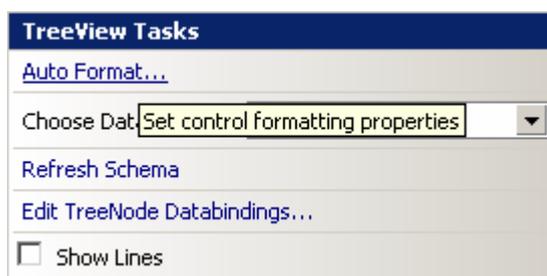


الشكل 35 [هذا هو الشكل يعد أن قمنا بضبط جميع الإعدادات الخاصة بهذه الأدوات]

و الآن لنرى المثال مع الأداة (TreeView)



الشكل ٣٦] نقوم بوضع العنصر (TreeView) و بعد ذلك نقوم بالضغط على الـ (Smart Tag) الخاص بها و نختار (SiteMapDataSource) و نحن الآن نكمل على المثال السابق ؛ حيث أننا سوف نستغل عنصر جلب البيانات المستخدم في العنصر (Menu) [



الشكل ٣٧] لتعديل الشكل، كل ما عليك هو الضغط على (Auto Format) [



الشكل ٣٨] أختار الشكل الذي تريده و من ثم (OK) [



الشكل 39 [الشكل النهائي للمثال]

• وفي النهاية هذا الفصل أود أن نتذكر سوياً مُلخصاً لما تعلمناه في هذا الفصل:

١. التعامل مع الصفحة الأم بالمعالج و يدويا و برمجيا
٢. التعامل مع أدوات الملاحة

- a. (SiteMapPath)
- b. (Menu)
- c. (TreeView)
- d. (SiteMapDataSource)