



الموسوعة التعليمية الكبرى ASP.Net الدرس الاول

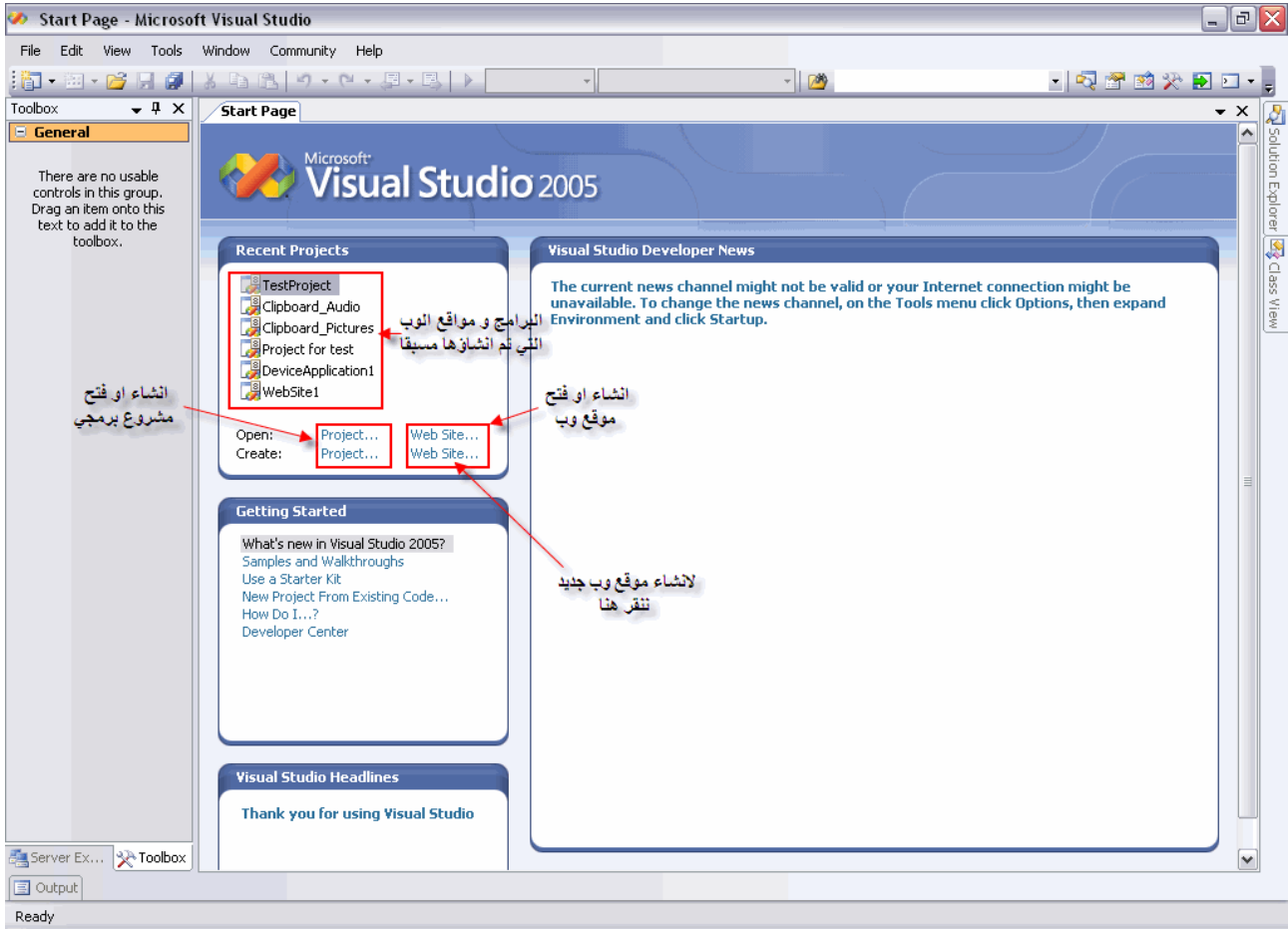
اخوتي واخواتي الافاضل ..
اقدم لكم سلسلة الدروس التعليمية الخاصة بتعليم برمجة وتصميم مواقع الوب باستخدام ASP.NET مع VB.net ...
بعد التطور الذي شهده العالم التقني في مجال تطور الوب و المعلومات التي يقدمها والتي لايمكن حصرها في مجال معين دون اخر ، وانما مواقع الوب باتت تتنوع وتقدم مختلف العلوم و المعارف ..
وقد كانت مواقع الوب سابقا لاتتعدى لغة HTML .. لكن تطور مواقع الوب وظهور متطلبات اخرى غير مجرد عرض المعلومات ، فقد كانت البذرة الاساسية لظهور تقنيات اخرى تساعد مصمم مواقع الوب على انتاج مواقع وب ديناميكية ذات قوة و متانه ..
وربما اخر ما وصل اليه التطور من قوة و سهولة وشيوعا في انتاج مواقع الوب الديناميكية هو (PHP & ASP) ، لكن شركة مايكروسوفت انتجت انتاجها القوي و الذي تجاوزت فيه جميع منتجاتها السابقة وهذا المنتج هو ASP.NET .. واما الان وحتى كتابة هذا الدرس فقد صدرت تقنية الاياكس الجديدة و التي زادت الامكانيات في مجال تصميم و برمجة مواقع الوب ..
سنعمد من خلال هذه السلسلة التعليمية على شرح كيفية برمجة وتصميم مواقع الوب باستخدام ASP.NET مع لغة البرمجة VB.NET ...
وتجدر الاشارة الى انني ساقوم باستخدام الاصدار Visual Studio 2005 الاصدار الاحترافي .. خلال شرح الدروس التعليمية .
وساحاول ان اتوخي قدر الامكان السهولة في شرح الدروس ..
كما انه ستكون فواصل زمنية ثابتة بين درس و اخر من اجل ان يتسنى للطلاب تطبيق الدرس ، بل الزيادة عليه وتجربة امكانيات اخرى اكثر من الامكانيات المشروحة في الدرس نفسه ..

اخيرا
ارجو ان يستفاد الجميع من هذه السلسلة التعليمية وينتفعوا بها ..
وبامكانكم مراسلتي على عنوان البريد الالكتروني الموضح ادناه من اجل الاستفسارات و الاسئلة ، راجيا ان تكون الاسئلة او الاستفسارات هي ضمن حدود الدرس وضمن مادته العلمية ..

دعائي للجميع بالتوفيق

المهندس
عدي ياسين
بكالوريوس هندسة حاسوب
engauday@hotmail.com
السبت - 2007 / 7 / 21

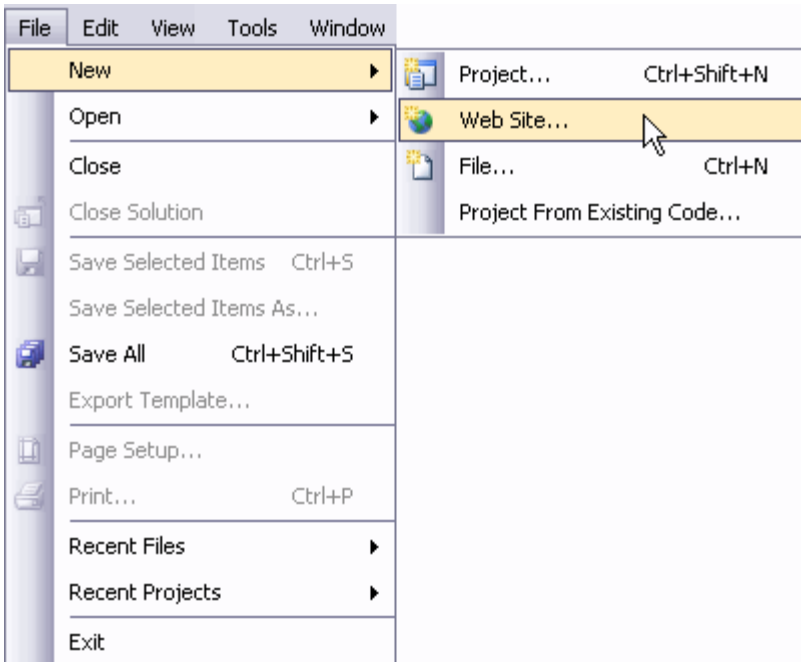
بداية الدرس ...
نقوم بتشغيل بيئة فيجوال ستوديو دوت نت لنظهر لنا النافذة الرئيسية :



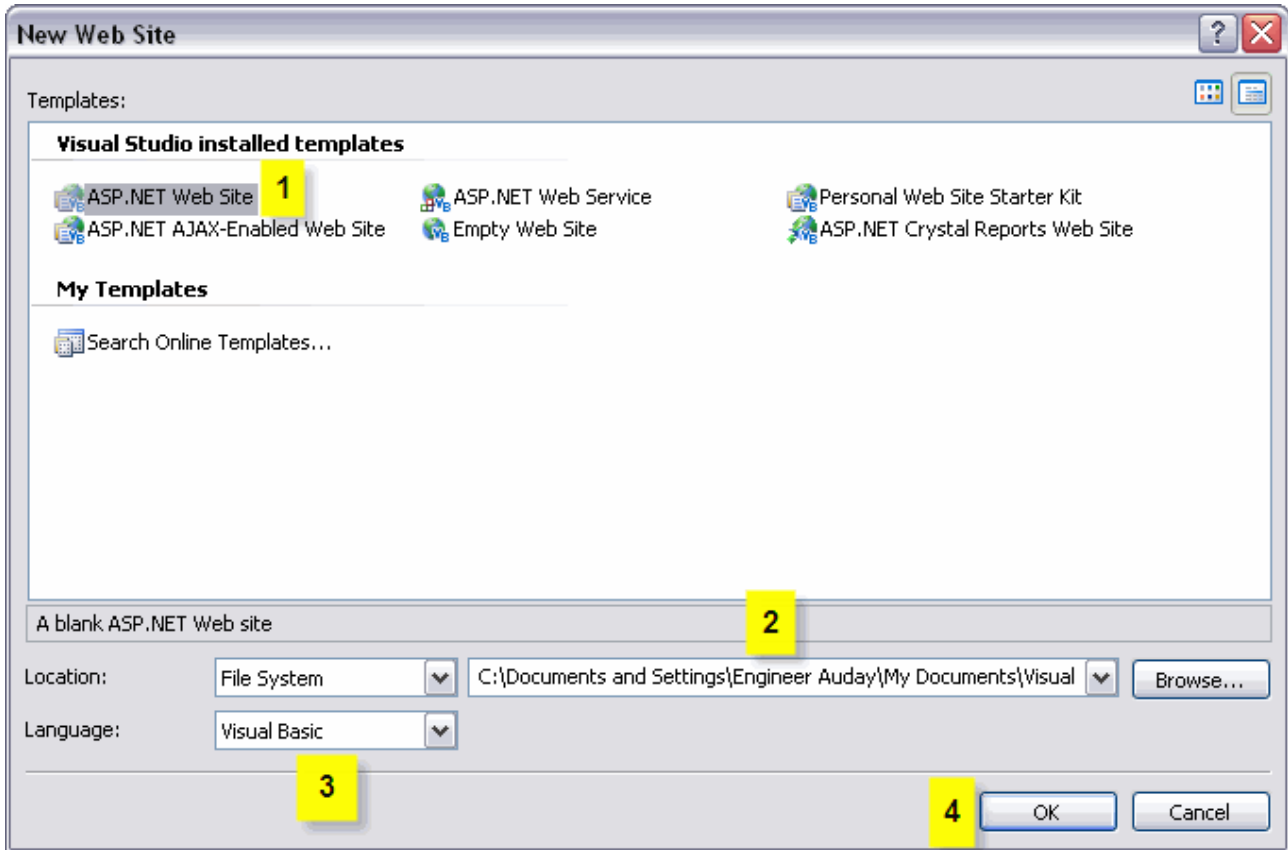
ولاجل ان نقوم بانشاء موقع وب جديد ، ننقر على الكتابة Web Sit... المقابلة لـ Create: او يمكننا استخدام طريقة اخرى لانشاء موقع وب جديد من خلال ..

File → New → Web Sit...

كما موضح في الصورة التالية :

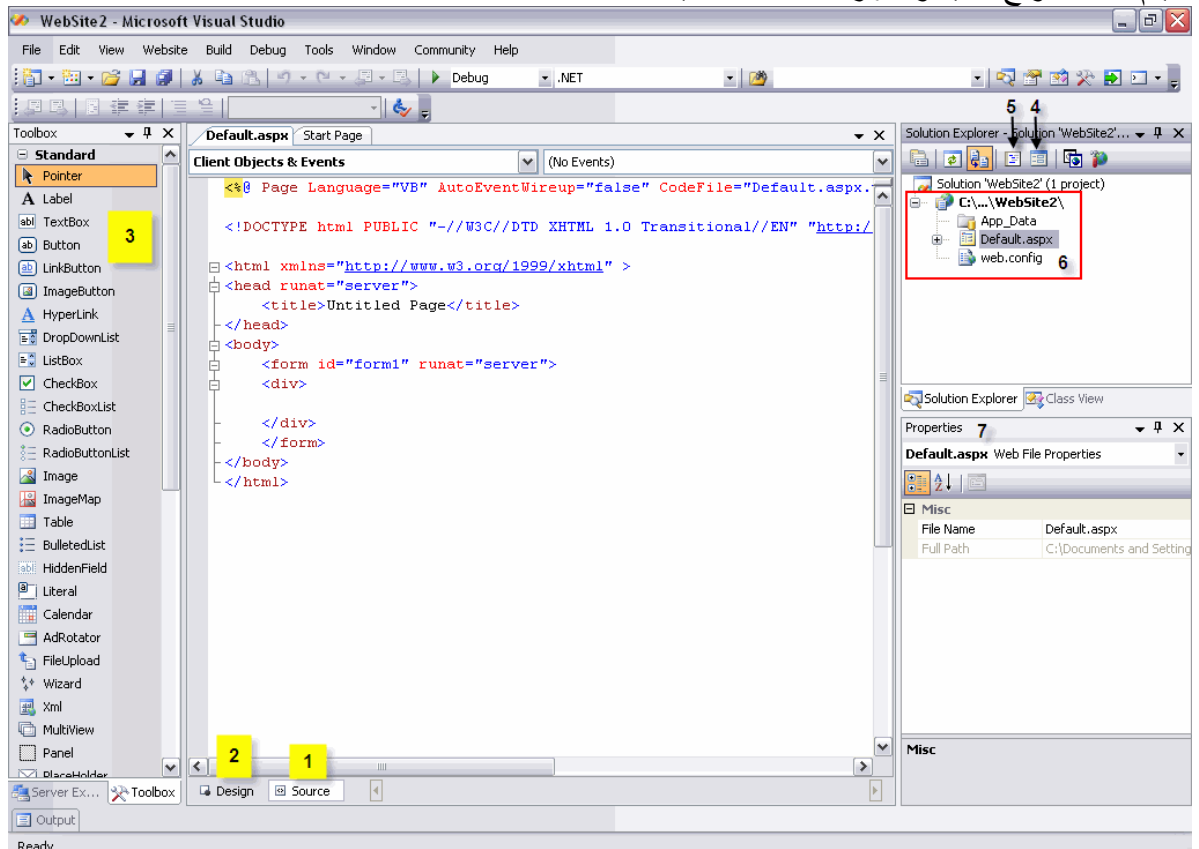


ستظهر لنا النافذة التالية :



ننتبع الارقام الموضحة في الصورة اعلاه ، حيث نختار موقع ASP.NET من الرقم 1 ، ونحدد مكان خزن الموقع من الرقم 2 ، ونحدد لغة البرمجة المستخدمة من 3، واخيرا ننقر على OK من 4 ..

سيتم انشاء موقع جديد وتظهر لنا النافذة التالية ..

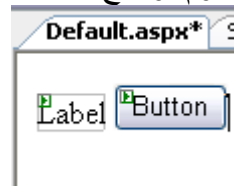


- حيث تشير الارقام الى :
- الرقم 1 : نافذة الكود .
 - الرقم 2 : نافذة التصميم .
 - الرقم 3 : صندوق الادوات .
 - الرقم 4 : نافذة التصميم .
 - الرقم 5 : نافذة الكود .
 - الرقم 6 : موقع الوب وملفاته .
 - الرقم 7 : نافذة الخصائص ..

ملاحظة: نلاحظ من خلال شرح الارقام الموجودة في الصورة اعلاه انه توجد اكثر من نافذة تصميم واكثر من نافذة كود .. في الحقيقة ان نافذة التصميم هي نافذة واحدة كما سنتعلم خلال هذا الدرس و الدروس القادمة ، اما نافذة الكود فهي نافذتان .. حيث بإمكاننا ان نعزل كود لغة VB.net في صفحة خاصة به نصل اليها من خلال نقر الزر المكتوب فوقه الرقم 5 ، اما النافذة التي نصل اليها من خلال النقر على 1 فهي نافذة الكود المصدري .

الان نذهب الى نافذة التصميم من خلال النقر على الكتابة الموجودة اسفل الرقم 2 او النقر على الزر الموجود اسفل الرقم 4 .

ستظهر لنا نافذة التصميم حيث يمكننا وضع الادوات والكتابات وغيرها من عناصر تصميم الصفحة .. نقوم بوضع نسخة من الاداة Label و الاداة Button على الصفحة بالشكل التالي :



يتم وضع الادوات على نافذة العمل من خلال سحب الادوات من صندوق الادوات وافلاتها على نافذة العمل او التصميم ..

نلاحظ ان الاداتين كلاهما على سطر واحد .. نضع المؤشر عند نهاية الاداة Label وننقر مفتاح Enter من لوحة المفاتيح ليصبح الزر Button اسفل عنصر التسمية Label ..



حيث اننا كلما نقرنا مفتاح Enter سينزل الزر بمقدار سطر واحد ..

ملاحظة: سنتعلم في الدروس القادمة كيفية استخدام الجداول من اجل ترتيب مواقع الادوات ضمن الصفحة .

كبداية .. ننقر على source الموجودة اسفل الصفحة (التي كان يظهر عليها الرقم 1) ..



سنفتح لنا نافذة الكود المصدري ..

ومن ضمن الكود المكتوب نلاحظ الكود التالي :

```
<asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
```

```
<br />
```

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

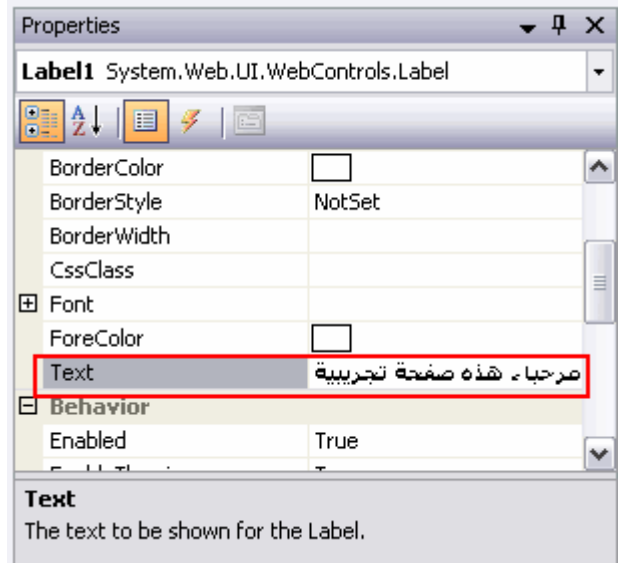
حيث ان السطر الاول من الكود يوضح طريقة اضافة الاداة Label الى النافذة ، اما الكود الاخير فهو يضيف الزر Button الى الصفحة ..

اما السطر الموجود في المنتصف (
) فهو من اجل اضافة سطر بين الاداتين ..

نلاحظ من خلال كود اضافة الاداة عنصر التسمية Label ، انه حدد لنا الاسم البرمجي ID و حدد لنا النص المكتوب داخل الاداة Text .. اما runat فهو يوضح ان هذه الاداة هي اداة خادم وب (server) ... وهذا هو الفرق بين ادوات ASP.net و ادوات HTML ..

ستتوضح الصورة اكثر عندما نتعلم استخدام ادوات HTML ..

الان لنكتب الجملة (Hello ,this is my test) بدل العبارة Label في الخاصية Text ..
<asp:Label ID="Label1" runat="server" Text="Hello ,this is my test"></asp:Label>
وننقر على Design الموجودة بالقرب من Source من اجل الانتقال من نافذة كود المصدر الى نافذة التصميم ..
نلاحظ ان الجملة (Hello ,this is my test) قد كتبت في عنصر التسمية ..
كذلك يمكننا التحكم بصورة اكثر بساطة في خصائص الادوات من خلال نافذة الادوات Properties ..
نتأكد من اننا حدد اداة عنصر التسمية Label ثم نغير قيمة الخاصية Text من نافذة الخصائص ونكتب فيها ..
مرحبا ، هذه صفحة تجريبية



نلاحظ ان الجملة كتبت في عنصر التسمية ..
كذلك يمكننا ان نفعل الشيء نفسه من اجل اداة الزر Button من خلال تحديدها وكتابة النص الذي نريده ان يظهر عليها من خلال اسناده الى الخاصية Text ..

تجربة الصفحة وتشغيل خادم الوب :

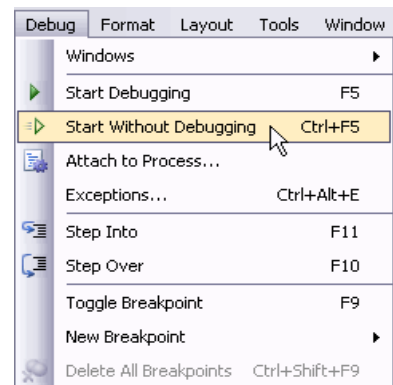
ان الصفحات التي نقوم بانشاؤها باستخدام ASP.Net تحتاج الى خادم وب من اجل تشغيلها بصورة صحيحة .
في الاصدارات السابقة كنا نقوم بعمل مجلد وهمي ليتم تشغيله من قبل IIS .. ولكن الان قامت شركة مايكروسوفت بتطوير خادم وب يعمل تلقائيا عند تجربة الصفحة .. وبذلك لانحتاج لعمل اي اعدادات مسبقة لاختبار الصفحات التي نقوم بانشاؤها ..

نلاحظ انه بإمكاننا تشغيل واختبار الصفحة من خلال الذهاب الى قائمة Debug ومن ثم اختيار الامر Start Without Debugging .

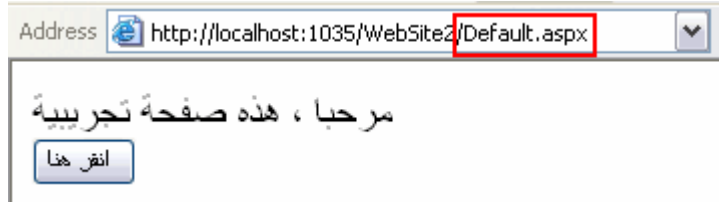
عندها ستقوم بيئة التشغيل بتشغيل خادم الوب الافتراضي ومن ثم فتح مستعرض الانترنت و عرض الصفحة .

سنلاحظ بجانب ساعة الحاسوب ايقونه مصغرة لخادم الوب

م 09:23 خادم الوب



ونلاحظ ان الصفحات التي يتم انشاؤها باستخدام ASP.Net تأخذ اللاحقة .aspx وهنا الصفحة هي Default.aspx



مرحبا ، هذه صفحة تجريبية

انقر هنا

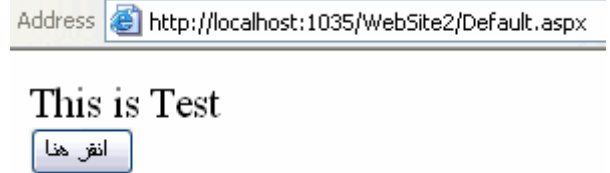
نشاهد عبارة الترحيب التي كتبناها في عنصر التسمية Label و النص على الزر Button .. الان حاول النقر على الزر ... تلاحظ انه لن يحدث شيء ، السبب في ذلك اننا لم نقم بكتابة اي كود ليتم تنفيذه عند النقر على الزر .. نغلق متصفح الويب .. و نرجع الى بيئة VS.net ..

الان سنكتب كود مبسط يقوم بعرض عبارة معينة على عنصر التسمية عند نقر الزر .. ابسط طريقة لكتابة حدث النقر للزر هو بالنقر على الزر نقرأ مزدوجا .. سنفتح لنا نافذة الكود ، لكنها هذه المرة تختلف عن نافذة كود المصدر ، انها واجهة مخصصة لكتابة الكود بلغة فيجوال بيسك دوت نت .. نكتب الكود المطلوب لتبدو الكود الكلي كما يلي :

```
Partial Class _Default
    Inherits System.Web.UI.Page

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Label1.Text = "This is a Test"
    End Sub
End Class
```

الكود الذي كتبناه هو : Label1.Text = "This is a Test" وذلك عندما ينقر المستخدم على الزر فاننا نسد الجملة (This is a Test) الى الخاصية التابعة لعنصر التسمية Label . نقوم الان بتجربة الكود كما تعلمنا سابقا او من خلال نقر مفاتيحي Ctrl + F5 من لوحة المفاتيح .. عند النقر على الزر فان الصفحة تستجيب لنا بعرض العبارة الجديدة ..



This is Test

انقر هنا

الى هنا ناتي الى نهاية درسا ، في الدروس القادمة سنتعلم كيفية تغيير الخصائص الاخرى للادوات وكيفية استخدام ادوات اخرى ..

ارجو ان تكونوا قد استفدتم من الدرس ..

دعائي للجميع بالتوفيق



الموسوعة التعليمية الكبرى ASP.Net الدرس الثاني

الاخوة و الاخوات الافاضل ..

مرحبا بكم في الدرس الثاني من دروس تعليم برمجة وتصميم مواقع الويب باستخدام ASP.NET & VB.NET .. تعلمنا في الدرس السابق كيفية تشغيل بيئة VS.NET وكيف نقوم بإنشاء موقع وب جديد وكذلك وضع الادوات على النموذج وكتابة كود بسيط جدا من اجل اظهار جملة معينة على عنصر التسمية .. سنتعلم في هذا الدرس كيفية التحكم بخصائص الادوات وكيفية استخدام صندوق النص TextBox ... نشغل بيئة Visual Studio.NET ونقوم بفتح موقع الويب الذي انشأناه في الدرس السابق .. نقوم بتحديد عنصر التسمية Label1 من خلال نقره بزر الفأرة اليسرى نقرة واحدة فقط .. ومن ثم نتجه الى خصائص هذه الاداة في نافذة الخصائص ..

اول خاصية هي ID ، وهي التي تحدد الاسم البرمجي للاداة والذي سيتم استخدامه من اجل الاشارة الى الاداة عند كتابة الكود، كما لاحظنا في الدرس السابق عندما كتبنا : `Label1.Text = "Hello, This is a test"` حيث اننا اشرنا الى اداة عنصر التسمية من خلال اسمها البرمجي المسند الى الخاصية ID وهو Label1 . واما من اجل الوصول الى خصائص اداة معينة فيمكننا ذلك من خلال كتابة اسم الاداة ومن ثم (.) ومن ثم الخاصية المطلوبة . ونلاحظ اننا عندما نقوم بكتابة (.) بعد اسم الاداة ، فان بيئة التطوير تعرض لنا قائمة بخصائص و طرق هذه الاداة ويمكننا ان نختار ما نشاء منها ، وهذا لاجعلنا بحاجة الى حفظ جميع اسماء الخصائص .. ملاحظة: ان حفظ اسماء الخصائص سيتم من خلال كثرة الممارسة وكتابة الكود ، لذلك فكل ما عليك الان هو فهم امكانيات كل خاصية .

في البداية سنتحدث عن الخصائص الاربعة الموضحة في الصورة ادناه :

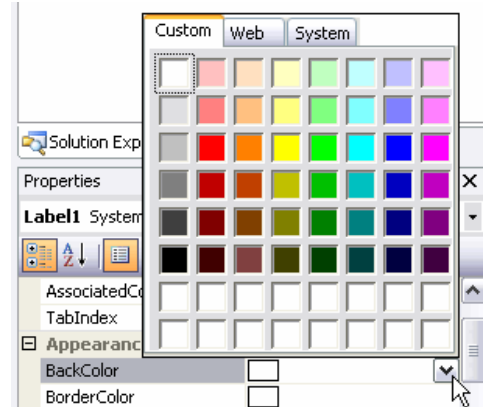
Appearance	
BackColor	<input type="text"/>
BorderColor	<input type="text"/>
BorderStyle	NotSet
BorderWidth	0px

لون خلفية الاداة .
لون الاطار المحيط بالاداة .
نمط الاطار المحيط بالاداة .
عرض الاطار المحيط بالاداة .

نقوم بالنقر على المربع الابيض المقابل للخاصية BackColor .. نلاحظ انه سيظهر لنا على الجانب مربع داخله سهم صغير كما موضح ادناه في الصورة :

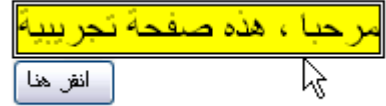


عند النقر حيث يوجد المؤشر في الصورة اعلاه سنتظهر لنا قائمة بالالوان المتاحة والتي يمكننا اسنادها الى هذه الخاصية ..

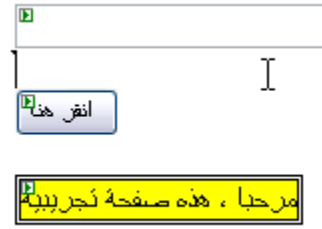


نختار اي لون نريده ان يكون لون الخلفية لاداة عنصر التسمية من خلال النقر عليه .

ونفس الشيء يمكننا عمله مع الخاصية Border Color من اجل تحديد لون الاطار المحيط بالاداة .. لكننا نلاحظ اننا عندما نقوم بتحديد لون معين للاطار فلن نشاهد اي اطار يحيط بالاداة .. كذلك لو اخترنا احدى الاختيارات التي توفرها لنا بيئة التطوير عند الخاصية BorderStyle من اجل تحديد نوع الاطار المحيط بالاداة فلن نشاهد اي اطار يحيط بالاداة .. ان السبب في ذلك ان عرض الاداة المحيط بالاداة هو صفر .. اي لا يوجد اي اطار فلذلك لاعمى من تحديد اي من الخصائص السابقة الذكر والتي تختص بالاطار لانه لا يوجد اطار اصلا .
في الحقل المقابل للخاصية BorderWidth نقوم بادخال قيمة رقمية ولتكن مثلا 3 ..
نلاحظ انه ظهر لنا اطار يحيط بالاداة ونلاحظ تأثير الخصائص التي حددناها مسبقا والخاصة بالاطار ..



الان نقوم باضافة الاداة TextBox ونرتب مواقع الادوات من خلال سحبها وافلاتها على الصفحة لتبدو كما يلي:

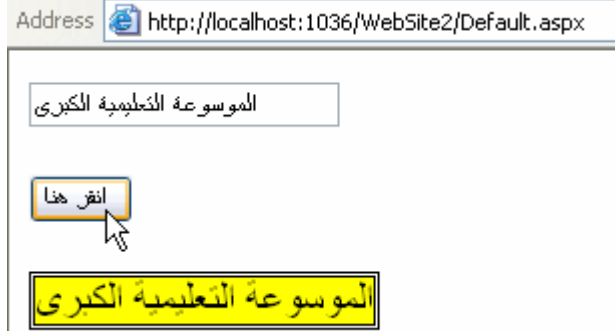


حيث يكون صندوق النص في الاعلى و من اسفله الزر Button ومن ثم عنصر التسمية .
الخطوة التالية هي اننا سنمكن المستخدم من كتابة نص في صندوق النص وعند النقر على الزر سيظهر النص على عنصر التسمية

.. وبسورة ايسر ، عندما ينقر المستخدم على الزر فان قيمة الخاصية Text التابعة لعنصر التسمية ستكون مساوية لقيمة الخاصية Text التابعة لصندوق النص ..
تنقر نقرا مزدوجا على الزر ونكتب الكود التالي :

```
Label1.Text = TextBox1.Text
```

الان نقوم بتجربة الصفحة من خلال النقر على مفتاحي Ctrl+F5 من لوحة المفاتيح ..
نقوم بادخال نص معين و ثم نقر على الزر ، نلاحظ ان النص في عنصر التسمية .



والان لو نقرنا على الصة بالزر الايمن للفارة .. واخترنا من القائمة المنبثقة الامر View Source ..
نلاحظ ان الادوات تم تصييرها على شكل ادوات HTML ..
كذلك نشاهد كيف تم اسناد الخصائص لكل اداة من الادوات ..




```

<input name="TextBox1" type="text" value="الموسوعة التعليمية الكبرى" id="TextBox1" />&nbsp;
<br />
<br />
<input type="submit" name="Button1" value=" انقر هنا " id="Button1" />
<br />
<br />
<span id="Label1" style="display:inline-block;background-color:Yellow;border-
color:Black;border-width:3px;border-style:Double;">الموسوعة التعليمية الكبرى</span></div>

```

الآن نعدل الكود من أجل أن نتحكم ببعض خصائص صندوق النص بصورة برمجية ..
سيصبح الكود النهائي لحدث النقر على الزر كما يلي :

```

Protected Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Button1.Click
Label1.Text = TextBox1.Text
With TextBox1
.BorderStyle = BorderStyle.Dotted
.BorderWidth = 3
End With
End Sub

```

نقوم الآن بتشغيل الصفحة من خلال Ctrl+F5 وننقر على الزر بعد أن نكتب داخل صندوق النص فتبدو النتيجة كما يلي ..



الموسوعة التعليمية الكبرى

من خلال الكود السابق ، تحكمتنا بنمط الاطار المحيط بصندوق النص وكذلك عرض الاطار ..
نلاحظ من خلال الكود السابق اننا اسندنا القيمة 3 لخاصية عرض الاطار ، كذلك يمكننا ان نستخدم اسلوب اخر من أجل تحديد عرض الاطار .. نستبدل الكود السابق الخاص بعرض الاطار بالكود التالي ..

```

.BorderWidth = Unit.Pixel(3)

```

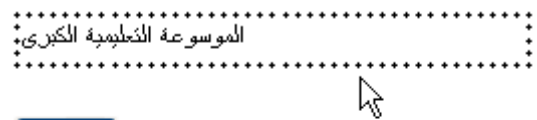
عند تجربة الكود من خلال عرض الصفحة في المتصفح ونقر الزر ، نلاحظ اننا نحصل على النتيجة نفسها .
الآن نظيف سطري الكود التاليين الى الكود السابق ..

```

.Width = 250
.Height = 20

```

وهما يحددان ارتفاع وعرض صندوق النص .. وقد اسندنا لهما قيم رقمية صحيحة ..
عند تجربة الصفحة نلاحظ النتيجة بعد النقر على الزر ..



انقر هنا

الموسوعة التعليمية الكبرى

كذلك يمكننا ان نحدد العرض او الارتفاع بنفس الاسلوب الذي استخدمناه مع عرض الاطار وكما يلي :

```

.Width = Unit.Pixel(250)

```

او يمكننا ان نحدد العرض و الارتفاع على شكل نسبة مئوية .. كما يلي :

```

.Width = Unit.Percentage(30)

```

اما الآن نقوم باضافة اربعة ازرار الى الصفحة لتبدو كما يلي بعد الانتهاء من تصميمها :

حيث سنقوم من خلال الازرار الموضحة بتفعيل صندوق النص او ايقاف تفعيله ، وكذلك اخفاء او اظهار صندوق النص .

انقر هنا

مرحبا ، هذه صفحة تحريبيه

اظهار صندوق النص

اخفاء صندوق النص

عدم تفعيل صندوق النص

تفعيل صندوق النص

نبدأ الان بكتابة الكود الخاص بالازرار الاربعة ..

أولا : الزر اخفاء صندوق النص :

```
Protected Sub Button3_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Button3.Click
    TextBox1.Visible = False
End Sub
```

ثانيا : الزر اظهار صندوق النص :

```
Protected Sub Button2_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button2.Click
    TextBox1.Visible = True
End Sub
```

ثالثا : الزر تفعيل صندوق النص :

```
Protected Sub Button5_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Button5.Click
    TextBox1.Enabled = True
End Sub
```

رابعا : الزر عدم تفعيل صندوق النص :

```
Protected Sub Button4_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Button4.Click
    TextBox1.Enabled = False
End Sub
```

الان نجرب الصفحة .. وننقر على الازرار للتأكد من عملها بنجاح .. حيث سيتم اخفاء او اظهار صندوق النص تبعا للزر الذي تم نقره ، وكذلك عملية تمكين او تفعيل صندوق النص او عدمها ..

الى هنا انتهى درسنا ..

ارجو ان تكونوا قد استفدتم من الدرس ..

دعاني للجميع بالتوفيق

المهندس
عدي ياسين
بكالوريوس هندسة حاسوب
engauday@hotmail.com
الاحد : 2007 / 7 / 22



الموسوعة التعليمية الكبرى ASP.Net الدرس الثالث

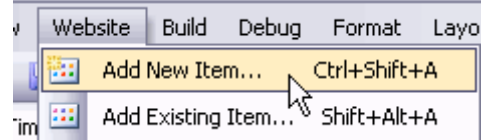
مرحبا بالاخوة و الاخوات في درس جديد من دروس تصميم وبرمجة المواقع باستخدام ASP.NET & VB.NET .
في هذا الدرس سنتعلم :

** استخدام الاداة RadioButtonList

** استخدام الاداة CheckBoxList

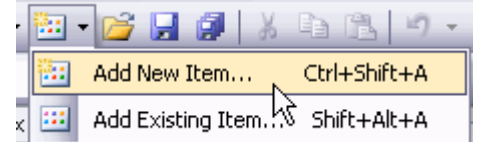
نبدأ درسنا باضافة صفحة جديدة الى موقعنا ..

من القائمة Website نختار الامر Add New Item... كما موضح في الصورة ادناه :

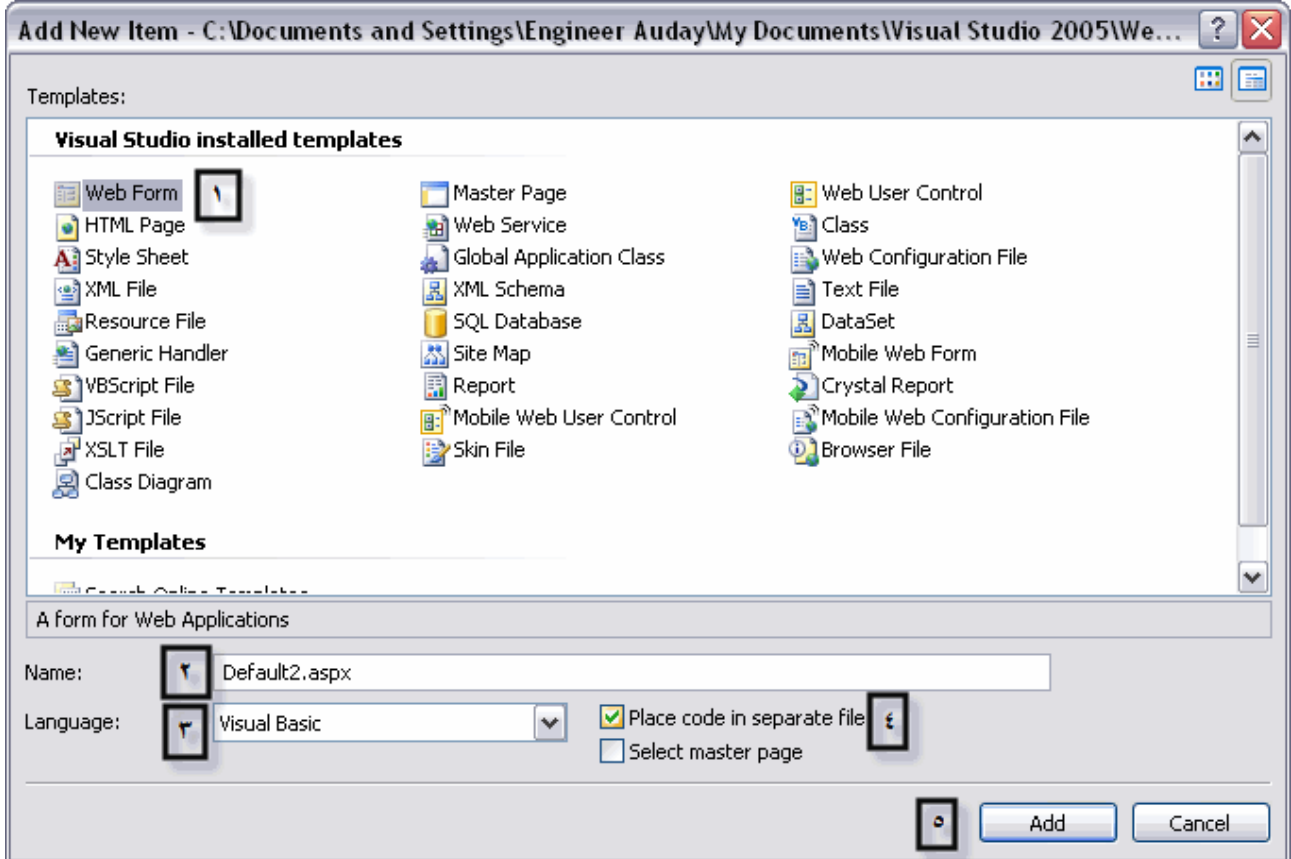


او من خلال مفاتيح الاختصار Ctrl + Shift + A ..

او باستخدام الطريقة الثالثة كما موضح في الصورة ادناه :



عندها ستظهر لنا النافذة التالية :

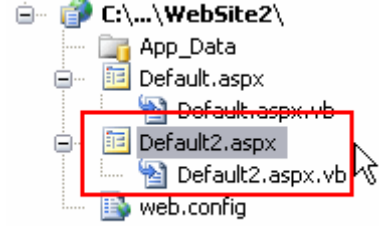


ننتبع الارقام الموضحة في الصورة اعلاه حيث ان :

الرقم 1 : يشير الى اننا سنقوم بانشاء نموذج وب ..

- الرقم 2 : نحدد اسم الصفحة الجديدة .
الرقم 3 : نوع اللغة المستخدمة .
الرقم 4 : نبين ان الكود ستوضع في ملف منعزل عن الصفحة .
الرقم 5 : ننقر على Add من اجل اتماما اضافة الصفحة الجديدة .

.. نلاحظ ظهور الصفحة في مستعرض الحلول Solution Explorer ..



حيث ان الكود سيتم خزنه داخل Default2.aspx.vb .
الان نقوم باضافة نسخة من الاداة TextBox ومن اسفلها نضيف نسخة من الاداة RadioButtonList وثم نضيف نسخة من الاداة Button .. كما يلي ..

الموسوعة التعليمية الكبرى

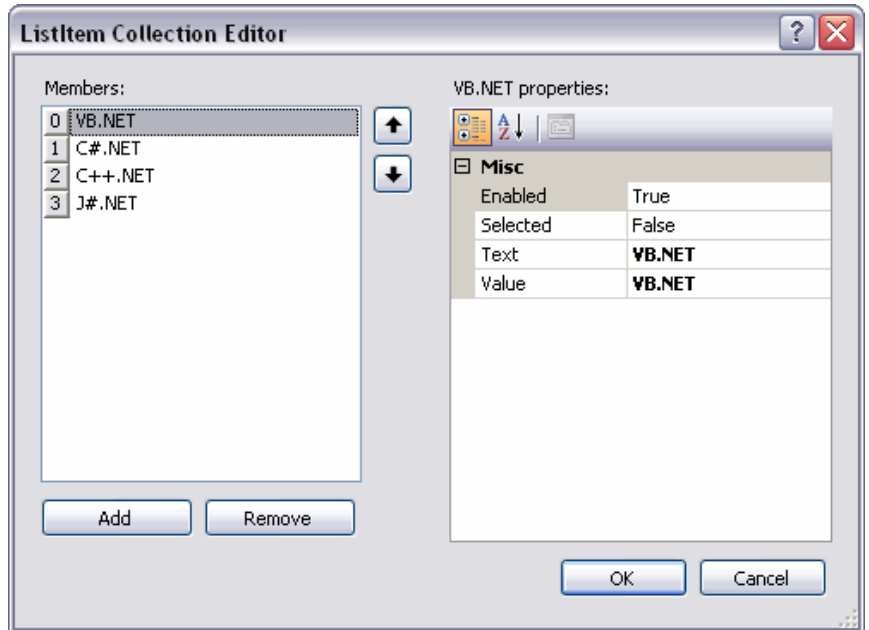
Unbound

Button

الان نقوم باضافة عناصر الى الاداة RadioButtonList من خلال النقر على الحقل المقابل للخاصية Item حيث يظهر لنا مربع صغير ننقر عليه لتظهر لنا النافذة الخاصة باضافة عناصر ..



من خلال النافذة المخصصة لاضافة عناصر جديدة نضيف مجموعة من العناصر ونعدل خاصية ال-Text لها لتبدو كما يلي :



اخيرا ننقر على زر OK حيث ستكون النتيجة كما يلي :

الموسوعة التعليمية الكبرى

- VB.NET
- C#.NET
- C++.NET
- J#.NET

Button

الان مهمتنا كتابة كود يقوم بعرض الخيار الذي يختاره المستخدم من لغات البرمجة داخل صندوق النص .. بمعنى ان نقر المستخدم على VB.NET ثم نقر الزر سيتم عرض VB.NET في صندوق النص .
ننقر نقرًا مزدوجًا على الزر Button من اجل اضافة الكود المطلوب ..

```
Protected Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles  
Button1.Click  
    Dim myList As ListItem  
    For Each myList In RadioButtonList1.Items  
        If myList.Selected = True Then  
            TextBox1.Text = myList.Text  
        End If  
    Next  
End Sub
```

قبل تجربة الصفحة ، لنتحدث قليلا عن الكود ..

في البداية قمنا بتعريف المتغير myList على انه عنصر قائمة .. ثم استخدمنا For .. Each من اجل التنقل بين جميع عناصر القائمة الموجودة في الاداة RadioButtonList .
بعد ذلك قمنا بتفحص العنصر الذي تم تحديده ... وعندما نجده نجعل النص في صندوق النص مساويا للنص في العنصر المحدد ..

اما الان نقوم بتجربة الصفحة من خلال النقر على Ctrl+F5 ..
نلاحظ اننا لايمكننا ان نختار اكثر من عنصر وهذه احدى ميزات الاداة RadioButtonList .. نقوم باختيار عنصر معين ثم ننقر على الزر نلاحظ ظهور نص الزر داخل صندوق النص ..

VB.NET

- VB.NET
- C#.NET
- C++.NET
- J#.NET

Button

ربما نود في بعض الحالات ان نمكن المستخدم من اختيار اكثر من عنصر في قائمة عناصر معينة ..

سنجرب ذلك من خلال استخدام الاداة CheckBoxList ..

نقوم الان باضافة نسخة من الاداة CheckBoxList و عنصر تسمية ونضيف عناصر لتبدو النتيجة كما يلي :

- VB.NET
 C#.NET
 C++.NET
 J#.NET

- Microsoft
 IBM
 Borland
 GenX

Button

الشركات المفضلة لديك هي

الآن سنعدل الكود الخاص بالزر .. سيتم عرض الشركات التي يتم اختيارها داخل عنصر التسمية ..

```
Protected Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Button1.Click
    Dim myList As ListItem
    For Each myList In RadioButtonList1.Items
        If myList.Selected = True Then
            TextBox1.Text = myList.Text
        End If
    Next
    For Each myList In CheckBoxList1.Items
        If myList.Selected = True Then
            Label1.Text += "</br>" & myList.Text
        End If
    Next
End Sub
```

الكود لايفرق عن الكود الخاص بالاداة RadioButtonList ..

نحرب الآن صفحة ، نقوم باختيار شركة واحدة او اكثر وننقر على الزر .. نلاحظ ظهور العناصر المحدد في عنصر التسمية ..
نختار شركات اخرى وننقر الزر مرة اخرى .. نلاحظ ظهور ??? خطأ .
ان الشركات السابقة تم اعادة كتابة اسمائها داخل عنصر التسمية ..

نكتب سطر الكود التالي قبل جميع اسطر الكود داخل حدث النقر التابع للزر Button1 :

Label1.Text = "الشركات المفضلة لديك هي "

الآن نحرب الصفحة ونلاحظ عدم ظهور هذه المشكلة من جديد والسبب اننا كل مرة يتم نقر الزر فيها نقوم بافراغ النص الموجود في عنصر التسمية Label1 ونسند اليه النص (الشركات المفضلة لديك هي) ..

الآن سنقوم بخطوة اخيرة وهي اننا سنقوم بحذف العناصر الموجودة داخل الاداة CheckBoxList .. ونقوم باضافة هذه العناصر من خلال الكود ..

من اجل حذف العناصر ، ننقر على الاداة لتحديدها ، ثم ننقل على المربع الصغير الموجود في الحقل المقابل لـ Items ..
ومن النافذة نقوم باختيار عنصر وننقر الزر Remove حتى يتم حذف جميع العناصر .
الآن ومن اجل اضافة العناصر من خلال الكود .. ننقر على الصفحة نقرا مزدوجا . ونكتب الكود التالي داخل حدث تحميل الصفحة.

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
With CheckBoxList1.Items
    .Add("Microsoft")
    .Add("IBM")
    .Add("Borland")
    .Add("GenX")
End With
End Sub
```

الآن نقوم بتجربة الصفحة .. ونختار شركات معينه ومن ثم ننقر على الزر ..
نلاحظ ظهور خطأ وهو ان اسماء الشركات تكررت داخل CheckBoxList ..
وكلما نقرونا الزر فان اسماء الشركات تتكرر ..

من اجل معالجة هذا الخلل نعدل الكود ليصبح كما يلي :

```
If Me.IsPostBack = False Then
    With CheckBoxList1.Items
        .Add("Microsoft")
        .Add("IBM")
        .Add("Borland")
        .Add("GenX")
    End With
End If
```

الان نجرب الصفحة .. ونقوم باختيار اسماء الشركات وننقر على الزر فنلاحظ عدم حدوث اي خلل وتبقى اسماء الشركات نفسها .. نجرب اختيار اسماء شركات جديدة وكذلك نحدد لغة برمجة معينة وننقر الزر ، نلاحظ ان الكود يعمل بنجاح ودون اي خلل ...

نغلق المتصفح ونرجع الى بيئة التطوير ..

عند تعديل الكود ، فاننا اضفنا Me.IsPostBack حيث ان هذا الكود يتأكد ان كانت الصفحة قد تم تحميلها اكثر من مرة او انها المرة الاولى .. فان كانت النتيجة هي True .. فهذا معناه ان الصفحة تم ارسالها الى السيرفر والا فهذا معناه اننا نشغل الصفحة لأول مرة ..

بذلك فاننا نقوم باضافة العناصر الى الاداة CheckBoxList فقط عند تحميل الصفحة وعندما تكون غير مرسله الى السيرفر ..

تبقى لدينا امر مهم وهو اضافة نص للزر ، فليس من المناسب ان يكون النص المعروض فيه هو Button ، نقوم بتحديد الزر ونكتب له النص (انقر هنا) عند الخاصية Text التابعة للزر ..

كذلك ننقر على الصفحة وندخل لها النص (الموسوعة التعليمية الكبرى) من اجل ان يكون عنوان للصفحة (في الحقل المقابل للخاصية Title) ، فليس من المعقول ان نقوم بتصميم موقع تعرض لنا احدى صفحاته العنوان Untitled ..

الى هنا انتي درسنا

ارجو ان تكونوا قد استفدتم من المعلومات الموجودة في الدرس ..

ارجو من الجميع تطبيق الدرس وتجنب عملية نسخ الكود ، حيث ان كتابة الكود باليد ترسخ المعلومات في الذاكرة اكثر من نسخه ولصقه ..

كذلك ارجو ان يحاول الجميع تطوير الصفحة وتجربة الخصائص الاخرى التي لم يتم شرحها في الدروس .

دعائي للجميع بالتوفيق

المهندس

عدي ياسين

بكالوريوس هندسة حاسوب

engauday@hotmail.com

الاثنين : 2007 / 7 / 23



الموسوعة التعليمية الكبرى ASP.Net الدرس الرابع

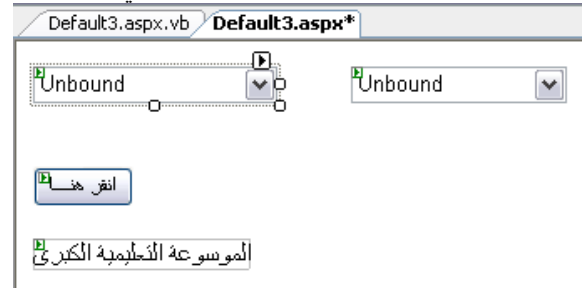
مرحبا بالاخوة و الاخوات في درس جديد من دروس تصميم وبرمجة المواقع باستخدام ASP.NET & VB.NET .
نتعلم في هذا الدرس :

** استخدام الاداة DroupDownList .

** التعامل مع الالوان .

** التعامل مع الخط .

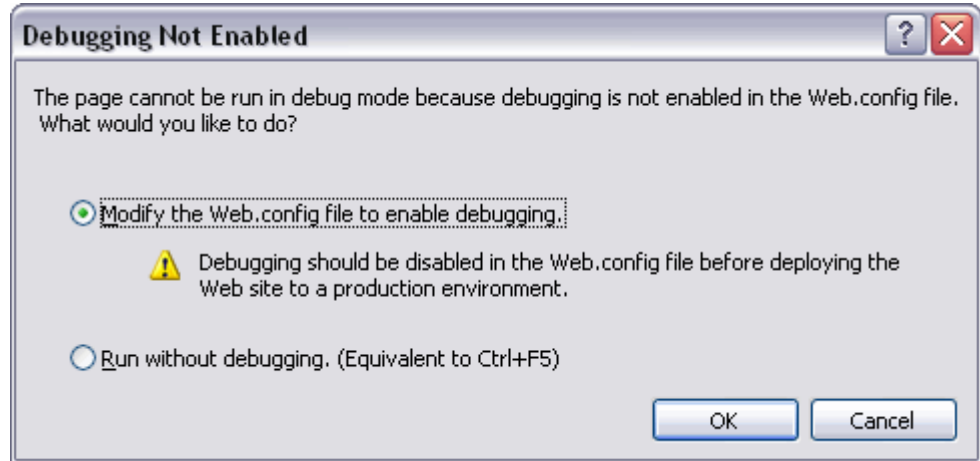
نقوم بانشاء صفحة جديدة .. ونضع عليها نسختان من الاداة DroupDownList و نسخة من الاداة Button و Label ..
تبدو الصفحة بعد الانتهاء من تصميمها كما يلي :



مهمتنا في هذا الدرس هي ان نقوم بادخال اسماء الوان في القائمة المنسدلة الاولى و انواع مختلفة من انماط الاطارات ، وعندما يحدد المستخدم الخيارات التي يريدھا .. يتم تطبيقها على عنصر التسمية .. سنقوم باضافة العناصر للقوائم المنسدلة من خلال الكود ..
ننقر على الصفحة نقرا مزدوجا ونكتب الكود التالي لحدث التحميل للصفحة :

```
Protected Sub Page_Load(ByVal sender As Object, ByVal  
If Me.IsPostBack = False Then  
With DropDownList1.Items  
.Add("--Please Select Item--")  
.Add("Red")  
.Add("Yellow")  
.Add("Blue")  
.Add("Green")  
End With  
With DropDownList2.Items  
.Add("--Please Select Item--")  
.Add("Dotted")  
.Add("Dashed")  
.Add("Double")  
.Add("Groove")  
.Add("Ridge")  
.Add("Inset")  
.Add("Outset")  
End With  
End If  
End Sub
```


الآن يمكننا تجربة الصفحة ولكن هذه المرة ننقر على الزر F5 من لوحة المفاتيح لتشغيل الـ Debug ..
ستظهر لنا النافذة التالية :



نحدد الخيار الأول و ننقر الزر OK ..
ستظهر لنا الصفحة ، ننقر على احدى القوائم المنسدلة ونرى العناصر التي فيها ..



الموسوعة التعليمية الكبرى

الآن نقوم بكتابة الكود اللازم لاسناد الخصائص المختارة الى عنصر التسمية ..
ننقر على الزر نقرا مضاعفا .. ونكتب الكود التالي :

```
Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs)
    Select Case DropDownList1.SelectedItem.Text
        Case "Red"
            Label1.BackColor = Drawing.Color.Red
        Case "Yellow"
            Label1.BackColor = Drawing.Color.Yellow
        Case "Blue"
            Label1.BackColor = Drawing.Color.Blue
        Case "Green"
            Label1.BackColor = Drawing.Color.Green
    End Select
    Select Case DropDownList2.SelectedItem.Text
        Case "Dotted"
            Label1.BorderStyle = BorderStyle.Dotted
        Case "Dashed"
            Label1.BorderStyle = BorderStyle.Dashed
        Case "Double"
            Label1.BorderStyle = BorderStyle.Double
        Case "Groove"
            Label1.BorderStyle = BorderStyle.Groove
        Case "Ridge"
            Label1.BorderStyle = BorderStyle.Ridge
        Case "Inset"
            Label1.BorderStyle = BorderStyle.Inset
        Case "Outset"
            Label1.BorderStyle = BorderStyle.Outset
    End Select
End Sub
```

نلاحظ اننا استخدمنا العبارة Select Case من اجل تحديد الخيار المطلوب .. وتبعاً للخيار فاننا نحدد قيمة الخاصية التابعة لعنصر التسمية ..
لو قمنا بتجربة الصفحة بنقر F5 ونختار خيارات محددة من القوائم .. نلاحظ تطبيق الخيارات على عنصر التسمية ..

Yellow Double

انقر هنا

الموسوعة التعليمية الكبرى

نضيف الان ثلاث نسخ من الاداة CheckBox .. كما يلي :

Unbound Unbound
Bold Italic Underline
انقر هنا

الموسوعة التعليمية الكبرى

الان نقوم بكتابة الكود اللازم لتطبيق خصائص صناديق الاختيار .. ننقر على الزر نقر مزدوجاً ونكتب الكود التالي في النهاية ..

```
If CheckBox1.Checked = True Then  
    Label1.Font.Bold = True  
Else  
    Label1.Font.Bold = False  
End If  
If CheckBox2.Checked = True Then  
    Label1.Font.Italic = True  
Else  
    Label1.Font.Italic = False  
End If  
If CheckBox3.Checked = True Then  
    Label1.Font.Underline = True  
Else  
    Label1.Font.Underline = False  
End If
```

في الكود اعلاه نتأكد ان كانت الاداة تم اختيارها او لا وتبعاً لذلك نحدد او لانحدد خاصية الخط لعنصر التسمية .
الان نجرب الصفحة ونلاحظ ان جميع الخيارات التي نحدد بها خصائص عنصر التسمية سيتم تطبيقها ..

Yellow Ridge
Bold Italic Underline
انقر هنا

الموسوعة التعليمية الكبرى

الى هنا وصلنا الى نهاية الدرس ..
ارجو ان تكونوا قد استفدتم من معلومات الدرس .. نلتقي في الدرس القادم ومعلومات جديدة .
دعائي للجميع بالتوفيق

المهندس
عدي ياسين
بكالوريوس هندسة حاسوب
engauday@hotmail.com
الثلاثاء : 2007 / 7 / 24



الموسوعة التعليمية الكبرى ASP.Net الدرس الرابع

مرحبا بالاخوة و الاخوات في درس جديد من دروس تصميم وبرمجة المواقع باستخدام ASP.NET & VB.NET
الالوان :

تعلمنا في الدرس السابق لمحة سريعة حول كيفية التعامل مع الالوان ..

ان الخاصية Color تشير الى الكائنة Color من فضاء الاسماء System.Drawing .
يمكننا التعامل مع الالوان بعدة طرق ..

** باستخدام القيم اللونية ARGB(alpha,red,green,blue) . نقوم بتحديد كل قيمة برقم صحيح Integer يتراوح بين 0 و 255 ، ان الخاصية alpha تمثل الشفافية للالوان ، و عادة ما نحدد لها القيمة 255 لتكون غير شفافية (ظاهرة) .

** من خلال اسماء الالوان في (.NET) المعرفة مسبقا ..

** باستخدام اسم لون HTML ..

من اجل التعامل مع الالوان يحتاج الى استيراد فضاء الاسماء System.Drawing .

نقوم باضافة صفحة جديدة ونعطيها الاسم Color ..

نصمم الصفحة لتبدو كما في الشكل ادناه باضافة نسخة من الاداة Label و ثلاث نسخ من الاداة Button ..

الموسوعة التعليمية الكبرى

ARGB

.NET

HTML Name

سنقوم باضافة الكود اللازم لاستخدام كل طريقة من الطرق السابقة الذكر في التعامل مع الالوان ..

كود الزر ARGB :

```
Protected Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Button1.Click
    Dim alpha As Integer = 255, red As Integer = 0
    Dim green As Integer = 255, blue As Integer = 0
    Label1.ForeColor = System.Drawing.Color.FromArgb(alpha, red, green, blue)
End Sub
```

هنا قمنا بتعريف اربعة متغيرات من نوع صحيح واسندنا لها قيم رقمية .. ومن ملاحظة القيم الرقمية نلاحظ ان النتيجة ستكون هي اللون الاخضر لان باقي القيم اسندنا لها القيمة صفر ، ما عدا قيمة الشفافية التي اسندنا لها القيمة 255 ليكون اللون ظاهر تماما .
بعد ذلك استخدمنا FromArgb(alpha, red, green, blue) من اجل جعل لون الخط اخضر .

كود الزر .NET :

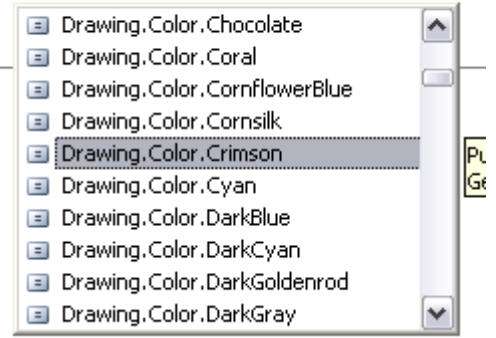
```
Protected Sub Button2_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Button2.Click
    Label1.ForeColor = Drawing.Color.Crimson
End Sub
```

ونلاحظ ان البيئة توفر لنا مجموعة كبيرة الالوان التي يمكننا استخدامها ..

```

Protected Sub Button2_Click(ByVal sender As Object
    Label1.ForeColor = cri
End Sub
d Class

```



كود الزر HTML Name :

```

Protected Sub Button3_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Button3.Click
    Label1.ForeColor = System.Drawing.ColorTranslator.FromHtml("Blue")
End Sub

```

او بإمكاننا كتابة الاسم مباشرة كما يلي ..

```
<asp:TextBox ForeColor ="Red" Text ="Test" ID ="txt" runat ="server" />
```

او من خلال ادخال قيمة بالنظام السداسي عشر لتحديد اللون المطلوب ..

```
<asp:TextBox ForeColor ="#ff50ff" Text ="Test" ID ="txt" runat ="server" />
```

الان نقوم بتجربة الصفحة .. وننقر على الازرار الثلاثة ، نلاحظ تغير لون الخط لعنصر التسمية ..

الموسوعة التعليمية الكبرى

Test

ARGB

.NET

HTML Name

الخطوط Fonts :

الخاصية Font تشير الى الكائن FontInfo المنحدر من فضاء الاسماء : System.Web.UI.WebControls .
والذي يمتلك عدة خصائص تحدد الاسم و الحجم و النمط .
نضيف مجموعة اخرى من الادوات الى الصفحة لتبدو كما يلي :

الموسوعة التعليمية الكبرى

Test

ARGB

.NET

HTML Name

Font-Name

Font-Bold

Font-Size-1

Font-Size-2

الان سنتعلم كيفية التحكم باسم اللون او نمطه او حجمه ..

كود الزر Font-Name :

```
Protected Sub Button4_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles  
Button4.Click  
    Label1.Font.Name = "Tahoma"  
End Sub
```

حيث هنا اسدنا اسم الخط Tahoma لخاصية اسم الخط التابعة لعنصر التسمية Label1 .

كود الزر Font-Bold :

```
Protected Sub Button5_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles  
Button5.Click  
    Label1.Font.Bold = True  
End Sub
```

حيث تحتاج الخصائص (Bold,Italic, Strikeout, Underline) الى احدى القيمتين (True , False) من اجل تفعيل او ايقاف تفعيل الخاصية ..
هنا عند النقر على الزر ، فان الخط يصبح مثخن ..

كود الزر Font-Size-1 :

```
Protected Sub Button6_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles  
Button6.Click  
    Label1.Font.Size = FontUnit.Larger  
End Sub
```

هنا اسدنا القيمة Larger لحجم الخط ..
يمكننا استخدام مجموعة من القيم لهذا النمط من التعامل مع حجم الخط (Smaller , Small ,Medium ,Large ,Larger) .

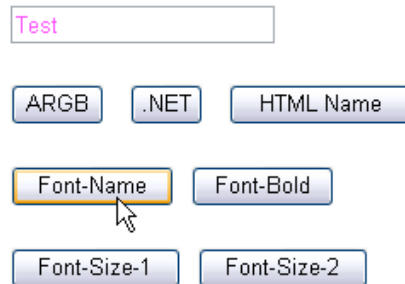
كود الزر Font-Size-2 :

```
Protected Sub Button7_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles  
Button7.Click  
    Label1.Font.Size = FontUnit.Point(14)  
End Sub
```

هنا نسند قيمة رقمية من اجل تحديد حجم الخط ..

الان نقوم بتشغيل الصفحة وتجربة الازرار وملاحظة تاثر خط عنصر التسمية من خلال النقر على الازرار ..

الموسوعة التعليمية الكبرى



التركيز Focus :

يمكننا تحديد خاصية التركيز لعنصر معين من خلال الخاصية Default Focus ...

```
<form id="form1" defaultfocus ="txt" runat="server">
```

حيث يمثل txt صندوق نص قمنا بانشاءه مسبقا ...

نلاحظ عند اضافة الخاصية defaultfocus ="txt" عند تشغيل الصفحة فان التركيز على صندوق النص .

الموسوعة التعليمية الكبرى



كذلك يمكننا التحكم بالتركيز من خلال مفتاح الوصول Access Key ، فمثلا لو اسندنا لمفتاح الوصول الخاص بصندوق النص txt القيمة A ، عندها عند النقر على Alt+A ينتقل التركيز الى txt ..



الآن نقوم بتشغيل الصفحة .. وننقر على مفتاحي Alt+A .. نلاحظ ان التركيز ينتقل الى صندوق النص .

الموسوعة التعليمية الكبرى

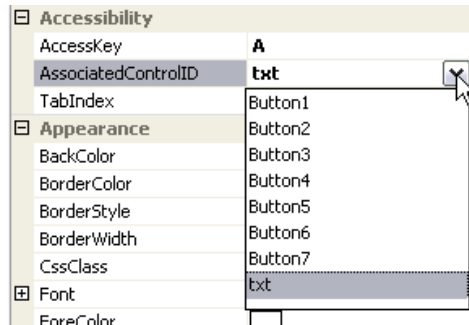
Test

ARGB .NET HTML Name

Font-Name Font-Bold

Font-Size-1 Font-Size-2

كذلك فان عنصر التسمية مشمول بعملية التركيز Focus بالرغم من انه لايقبل التركيز ، حيث نحدد يمكننا من خلال الخاصية AssociatedControlID .. وكذلك نحدد الاداة التي سيرتبط بها عنصر التسمية عند استخدام التركيز .. نقوم الان بحذف مفتاح الوصول من صندوق النص txt وندخل القيمة A لمفتاح الوصول لعنصر التسمية ونحدد txt عند العنصر الذي سينتقل اليه التركيز ..



الآن ننقر مفتاح F5 وننقر على الزرين Alt+A من لوحة المفاتيح فنلاحظ انتقال التركيز الى صندوق النص ..

الزر الافتراضي (Default Button) :

ان ASP.NET يوفر لنا خاصية الزر الافتراضي في صفحات الوب . ان الزر الافتراضي هو الزر الذي (ينقر) عندما ينقر المستخدم على مفتاح Enter من لوحة المفاتيح .
كمثال ، ان كانت صفحة الوب تحتوي على نموذج ، ونحن نريد ان نجعل الزر (ارسال) هو الزر الافتراضي .. فعندما ينقر المستخدم على المفتاح Enter من لوحة المفاتيح فان الصفحة سيتم ارسالها و الحدث Button.Click سيحدث لهذا الزر .
من اجل ان نقوم بعمل زر افتراضي .. فيجب اسناد خاصية الزر الافتراضي لصفحة الوب مع اسم الزر المراد جعله هو الزر الافتراضي :

```
<form id="form1" defaultbutton ="Button1" runat="server">
```

عندها سيكون الزر Button1 هو الزر الافتراضي و الذي يحدث حدث النقر له عندما ينقر المستخدم على مفتاح Enter .

الى هنا وصلنا الى نهاية درسنا ..

ارجو ان تكونوا قد استفدتم من المعلومات المقدمة في هذا الدرس ..

ملتقانا في دروس قادمة ومعلومات جديدة ..

دعائي للجميع بالتوفيق

المهندس

عدي ياسين

بكالوريوس هندسة حاسوب

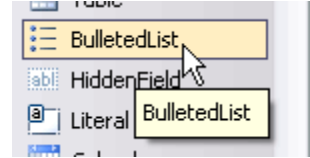
engauday@hotmail.com

الجمعة : 2007 / 7 / 27

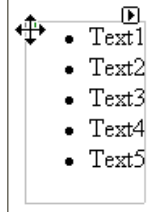


الموسوعة التعليمية الكبرى ASP.Net الدرس السادس

مرحبا بالاخوة و الاخوات في درس جديد من دروس تصميم وبرمجة المواقع باستخدام ASP.NET & VB.NET .
نقوم بانشاء صفحة جديدة ..
نضيف نسخة من الاداة BulletedList ..



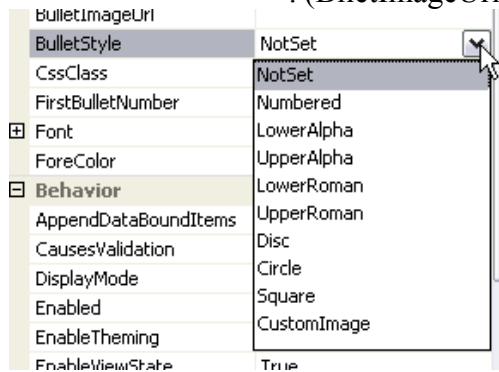
نقوم باضافة عناصر للاداة من خلال النقر على المربع المصغر الذي يظهر عند النقر على الحقل المقابل للخاصية Items .
نضيف العناصر لتبدو الاداة كما يلي :



ان هذه الاداة هي اداة تكافئ القائمة الغير مرتبة unorderedlist او القائمة المرتبة orderedlist ..

الخاصية BulletList :

وهي تحدد نوع القائمة .. نختار ام ارقام (1,2,3,..) Numbered او حروف ابجدية صغيرة (a,b,c,..) LowerAlpha او حروف ابجدية كبيرة (A,B,C,..) UpperAlpha او غيرها ، كذلك نحدد الرمز الممكن عرضه ، سواء دائري او مربع او صورة مخصصة (في حالة استخدامنا الخاصية BulletImageUrl) .



الخاصية BulletImageUrl :

وهي تحدد الصورة التي سيتم عرضها الى اليسار من كل عنصر .

الخاصية FirstBulletNumber :

في النم المرتبة التي تستخدم (Numbered , LowerAlpha , UpperAlpha , LowerRoman , UpperRoman) .. يمكننا تحديد القيمة الاولى للقائمة ..

مثلا ان اعطينا القيمة 3 لهذه الخاصية سيكون الترقيم 3,4,5 بالنسبة الى Numbered او C,D,E بالنسبة الى UpperAlpha .

BulletStyle	LowerAlpha
CssClass	
FirstBulletNumber	3

تكون نتيجة الاختيار السابق هي :

- c. Text1
- d. Text2
- e. Text3
- f. Text4
- g. Text5

في بعض الحالات ، نريد ان نجعل كل عنصر يمثل ارتباط تشعبي الى صفحة اخرى او موقع اخر ..
 يمكننا عمل ذلك من خلال الخاصية DisplayMode ..
 الخاصية DisplayMode تحدد ان كان النص في كل عنصر سيتم عرضه على شكل نص عادي او على شكل نص يحتوي على ارتباط Hyperlink ..
 نضيف نسخة من الاداة Button و Label لتبدو الصفحة كما يلي :

- c. [Text1](#)
- d. [Text2](#)
- e. [Text3](#)
- f. [Text4](#)
- g. [Text5](#)

[Choose Item]

الان نضيف الكود في حدث النقر للاداة من خلال النقر المزدوج عليها ..

```
Protected Sub BulletedList1_Click(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.BulletedListEventArgs) Handles BulletedList1.Click
    Dim ItemText As String = _
        BulletedList1.Items(e.Index).Text
    Label1.Text = " You Choose Item " & ItemText
End Sub
```

ان الكود السابق يخبر المستخدم اي عناصر القائمة تم نقره .. حيث يمكننا تطوير الكود من اجل التفاعل مع كل عنصر ..
 نقوم بتشغيل الصفحة وننقر على العنصر الاول ونلاحظ عنصر التسمية ..

- c. [Text1](#)
- d. [Text2](#)
- e. [Text3](#)
- f. [Text4](#)
- g. [Text5](#)

You Choose Item
Text1

الى هنا وصلنا الى نهاية درسنا ..
 ارجو ان تكونوا قد استفدتم من المعلومات المقدمة في هذا الدرس ..
 ملتقانا في دروس قادمة ومعلومات جديدة ..
 دعائي للجميع بالتوفيق

المهندس
 عدي ياسين
 بكالوريوس هندسة حاسوب
 engaday@hotmail.com
 الجمعة : 2007 / 7 / 27



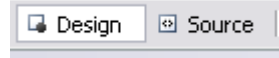
الموسوعة التعليمية الكبرى ASP.Net الدرس السابع

مرحبا بالاخوة و الاخوات في درس جديد من دروس تصميم وبرمجة المواقع باستخدام ASP.NET & VB.NET .
في درسنا هذا سنتعامل مع اداة الجدول ..
كما تعلمنا في كل درس جديد نقوم بانشاء صفحة جديدة ..
بعد ذلك نصمم الصفحة لتبدو كما يلي :

Rows: Cols:

Put Border Around Cells

كذلك يمكننا اتباع اسلوب اخر في تصميم الصفحة .. هذا الاسلوب من خلال كتابة كود في نافذة ال-Source ..



ننقر على ال-Source و نكتب الكود التالي :

```
<form id="form1" runat="server">
  <div>
    Rows:
    <asp:TextBox ID="txtRows" runat="server" /> &nbsp;
    Cols:
    <asp:TextBox ID="txtCols" runat="server" /> <br /> <br />
    <asp:CheckBox ID="chkBorder" runat="server"
      Text = "Put Border Around Cells" />
    <br /> <br />
    <asp:Button ID="cmdCreate" runat="server" Text="Create" />
    <br /> <br />
    <asp:Table ID="tbl" runat="server" />

    &nbsp;<br />
  </div>
</form>
```

ان الكود اعلاه يعطينا ناتج لتصميم الصفحة كما وضعنا مسبقا ..
حيث تتكون الصفحة من نسختين من الاداة TextBox احدهما سنقوم باستخدامه من اجل السماح للمستخدم بادخال عدد الصفوف و
الآخر من اجل ادخال عدد الاعمدة المكونة للجدول ..
اما نسخة اداة الاختبار فهي من اجل وضع حدود للجدول (ان تم اختيارها) ..
بعد ذلك اداة الزر ونسخة من اداة الجدول ..
الفكرة في صفحاتنا هذه هي اننا سنجعل المستخدم يدخل عدد الاعمدة و الصفوف المكونة للجدول ويختار ان كان يريد حدود حول
الخلايا او لا ، ثم عند النقر يتم انشاء الجدول بالاعتماد على الاختيارات التي اختارها مسبقا .

الان لدينا حديثين سنقوم بكتابة الكود الخاص بهما ..
الحدث الاول هو حدث تحميل الصفحة ، والكود المستخدم فيه هو من اجل انشاء حدود للجدول بصورة عامة وتحديد عرض
الحدود (وهو يختلف عن الحدود التي نقوم بانشائها حول الخلايا من خلال اختيار اداة صندوق الاختبار) ..
ويمكننا الاستغناء عن هذا الكود ان كنا لانريد ان نقوم بانشاء حدود للجدول ..

اما الكود لحدث النقر على الزر فهو يقوم بانشاء الجدول بالاعتماد على ادخالات المستخدم .

كود حدث تحميل الصفحة :

```
Protected Sub Page_Load(ByVal sender As Object
    tbl.BorderStyle = BorderStyle.Inset
    tbl.BorderWidth = Unit.Pixel(1)
End Sub
```

نلاحظ اننا استخدمنا الخاة BorderStyle من اجل تحديد نوع الاطار المحيط بالجدول ..
كذلك حددنا عرض حدود الجدول من خلال BorderWidth .

كود حدث النقر للزر :

```
Protected Sub cmdCreate_Click(ByVal sender As Object, ByVal e As EventArgs)
    tbl.Controls.Clear()
    Dim i, j As Integer
    For i = 0 To Val(txtRows.Text - 1)
        Dim rowNew As New TableRow()
        tbl.Controls.Add(rowNew)
        For j = 0 To Val(txtCols.Text - 1)
            Dim cellNew As New TableCell()
            cellNew.Text = "Example Cell(" & i.ToString() & ", " & j.ToString() & ")"
            If chkBorder.Checked = True Then
                cellNew.BorderStyle = BorderStyle.Inset
                cellNew.BorderWidth = Unit.Pixel(1)
            End If
            rowNew.Controls.Add(cellNew)
        Next
    Next
End Sub
```

في بداية الكود نقوم بافراغ الجدول من جميع محتوياته .. (الخلايا) .
بعد ذلك نقوم بتعريف متغيرين i, j .. والذين نستخدمهما من اجل عمل الدورات الخاصة بانشاء الصفوف و الخلايا التي موجودة داخلها ..
الفكرة في الكود تتلخص في عمل دوراه بعدد الصفوف ، ومن ثم عمل دوراه اخرى داخل الدواره الاولى من اجل انشاء الخلايا لكل صف من الصفوف .
ونستخدم Controls.Add من اجل اضافة الصفوف للجدول و الخلايا للصفوف .

الان يمكننا ان نجرب الصفحة من خلال تشغيلها بنقر مفتاح F5 من لوحة المفاتيح .

Rows: Cols:

Put Border Around Cells

ندخل القيم كما هو موضح في الصورة اعلاه .. وننقر على الزر
سيتم انشاء الجدول بعدد الصفوف و الاعمدة التي تم ادخال قيمها في صناديق النص ..
كذلك سيتم انشاء حدود للخلايا ان كان صندوق الاختبار مختارا .

Rows: Cols:

Put Border Around Cells

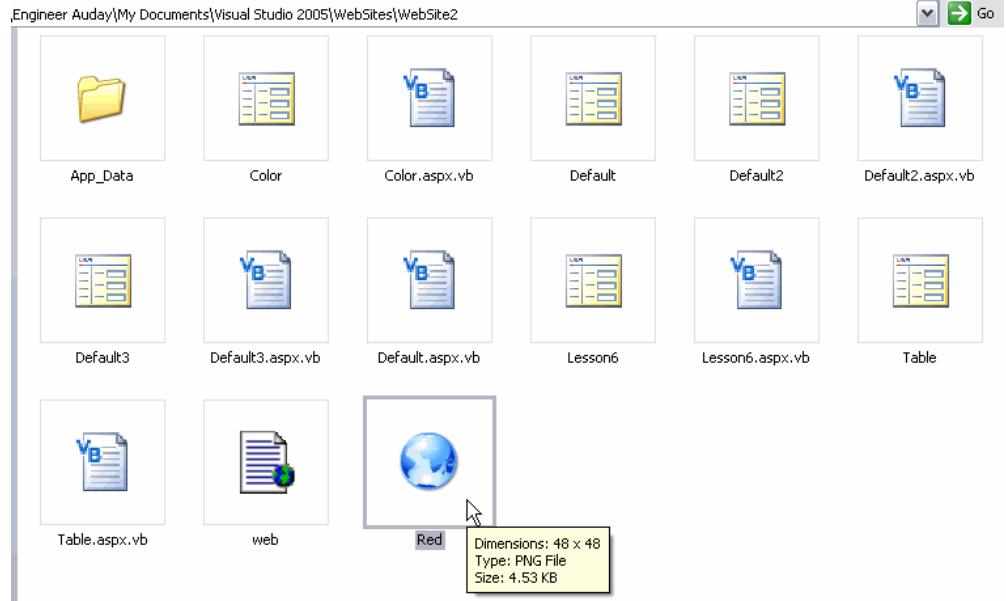
Create

Example Cell(0,0)	Example Cell(0,1)	Example Cell(0,2)
Example Cell(1,0)	Example Cell(1,1)	Example Cell(1,2)

كذلك يمكننا وضع ادوات داخل كل خلية من خلايا الجدول ...
نقوم بتعديل كود حدث النقر للزر ليصبح كما يلي :

```
Protected Sub cmdCreate_Click(ByVal sender As Object, ByVal e As S
tbl.Controls.Clear()
Dim i, j As Integer
For i = 0 To Val(txtRows.Text - 1)
Dim rowNew As New TableRow()
tbl.Controls.Add(rowNew)
For j = 0 To Val(txtCols.Text - 1)
Dim cellNew As New TableCell()
Dim lblNew As New Label()
lblNew.Text = "الموسوعة التعليمية الكبرى"
Dim imgNew As New System.Web.UI.WebControls.Image()
imgNew.ImageUrl = "Red.png"
cellNew.Controls.Add(lblNew)
cellNew.Controls.Add(imgNew)
If chkBorder.Checked = True Then
cellNew.BorderStyle = BorderStyle.Inset
cellNew.BorderWidth = Unit.Pixel(1)
End If
rowNew.Controls.Add(cellNew)
Next
Next
End Sub
```

من خلال الكود السابق ، قمنا باضافة اداة عنصر التسمية و اداة الصورة لكل خلية من الخلايا ..
من اجل نجاح عمل الكود ن يجب وضع صورة بالاسم Red داخل مجلد الموقع ..



نقوم بتجربة الصفحة من خلال نقر مفتاح F5 من لوحة المفاتيح ..
وندخل قيم لعدد الاعمدة و الصفوف ومن ثم ننقر على الزر من اجل انشاء الجدول .

Rows: Cols:

Put Border Around Cells

Create

الموسوعة التعليمية الكبرى	الموسوعة التعليمية الكبرى	الموسوعة التعليمية الكبرى
الموسوعة التعليمية الكبرى	الموسوعة التعليمية الكبرى	الموسوعة التعليمية الكبرى

الى هنا تعلمنا كيفية انشاء جدول بصورة برمجية وكتابة نص داخل كل خلية ..
كذلك وضع الادوات داخل كل خلية (عنصر التسمية ، اداة الصورة) .
يمكننا عمل نفس الامور السابقة من خلال التصميم (كما سنتعلم من خلال الدروس السابقة) .
ان المرونة الحقيقية في الجدول هي ان كل جدول ، صف جدول ، خلية جدول تمثل كائن بمميزات كاملة .. يمكننا اسناد لكل خلية
اطار مختلف ولون اطار مختلف ولون نص من خلال التحكم بخصائصها .

تعلمنا من خلال الدروس السابقة كيفية التعامل مع الالوان و الخطوط .. لكننا سنكمل درسنا هذا من خلال اضافة نسختين من
صندوق القائمة المنسدلة يحتوي الاول على جميع الالوان و الثاني على جميع الخطوط المعدة على الحاسب ..
وعندما يحدد المستخدم عدد الاعمدة و الصفوف ، يحدد معهما لون الخلفية للخلايا و نمط الخط ..
نكمل تصميم الصفحة لتبدو كما يلي .

Rows: Cols:

Put Border Around Cells

Unbound Unbound

Create

###

الان نعدل كود تحميل الصفحة ليبدو كما يلي :

```

Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.Event
    If Me.IsPostBack = False Then
        Dim colorArray As String() = System.Enum.GetNames( _
            GetType(System.Drawing.KnownColor))
        DropDownList1.DataSource = colorArray
        DropDownList1.DataBind()

        Dim fonts As New System.Drawing.Text.InstalledFontCollection()
        For Each family As Drawing.FontFamily In fonts.Families
            DropDownList2.Items.Add(family.Name)
        Next
    End If
    tbl1.BorderStyle = BorderStyle.Inset
    tbl1.BorderWidth = Unit.Pixel(1)
End Sub

```

وكما تعلمنا مسبقا .. فمن اجل اضافة عناصر الى صندوق القائمة المنسدلة فيجب علينا التأكد ان لا يتم اضافة العناصر في كل مرة يتم فيها ارسال الصفحة الى الخادم .. ويتم ذلك من خلال Me.IsPostBack ونلاحظ اننا عرفنا المتغير colorArray على انه مصفوفة من النصوص وبذلك جعلناها كمصدر للبيانات لعنصر القائمة المنسدلة من خلال الخاصية DataSource . الان ننقر على صندوق القائمة المنسدلة الاول ونكتب له الكود التالي ..

```

Protected Sub DropDownList1_SelectedIndexChanged(ByVal sender As Object
    tbl1.BackColor = System.Drawing.ColorTranslator.FromHtml( _
        DropDownList1.SelectedItem.Text)
End Sub

```

اما صندوق القائمة الثاني فنكتب له الكود التالي :

```

Protected Sub DropDownList2_SelectedIndexChanged(ByVal ser
    tbl1.Font.Name = DropDownList2.SelectedItem.Text
End Sub

```

نقوم الان بتجربة الصفحة ونحدد عدد الصفوف و الاعمدة ، ونحدد اللون ونوع الخط ثم ننقر على الزر ونلاحظ النتيجة ..

Rows: Cols:

Put Border Around Cells

اطوسوعث التعليميث الكبرى	اطوسوعث التعليميث الكبرى
اطوسوعث التعليميث الكبرى	اطوسوعث التعليميث الكبرى

الى هنا وصلنا الى نهاية درسنا .. ارجو ان تكونوا قد استفدتم من المعلومات المقدمة في هذا الدرس .. ملتقانا في دروس قادمة ومعلومات جديدة .. دعائي للجميع بالتوفيق

المهندس
عدي ياسين
بكالوريوس هندسة حاسوب
engauday@hotmail.com
الاحد : 2007 / 8 / 5



الموسوعة التعليمية الكبرى ASP.Net الدرس الثامن

مرحبا بالاخوة و الاخوات في درس جديد من دروس تصميم وبرمجة المواقع باستخدام ASP.NET & VB.NET . سنتحدث في هذا الدرس عن كائنات التحقق من المحتوى Validation Controls .. عندما نصمم موقع ونقوم بوضع نموذج فيه نطلب من خلاله معلومات من المستخدم .. فقد تحدث بعض الاخطاء عندما يقوم المستخدم بمليء الحقول بالمعلومات ، كأن يهمل ادخال معلومات حقل مهم ويتركه فارغا ، او ان يقوم المستخدم بادخال المعلومات بشكل خاطئ مثل ان يدخل صيغة بريد الكتروني خاطئة و التي تسبب لنا مشاكل فيما بعد عند محاولة ارسال رسالة بريد الكتروني للمستخدم . كذلك ممكن ان يدخل المستخدم قيم خارج نطاق قيم محدد ، وغير ذلك الكثير من الاخطاء الاخرى . ان ASP.NET يوفر لنا الحلول المناسبة لتلك المشاكل من خلال تقديمه لنا عناصر التحقق من المحتوى Validation Controls والتي مهمتها التحقق من صحة المحتوى و المعلومات التي ادخلها المستخدم وذلك باختبار الحقول وارجاع تقرير عن الخطا بصورة تلقائية ...

عناصر التحقق من المحتوى The Validation Controls :

ان ASP.NET يوفر لنا خمسة عناصر للتحقق من المحتوى .. اربعة منها مخصصة لاغراض ثابتة .. والخامس يمكننا تخصيصه نحن .

1. Required Field Validator (اداة الحقل مطلوب) :

المحتوى يتحقق في حالة ان الحقل لم يترك فارغا .

2. Range Validator (اداة النطاق) :

المحتوى يتحقق في حالة ان عنصر الادخال يحتوي على قيمة تقع ضمن نطاق محدد (رقمي ، ابجدي ، تاريخ) .

3. Compare Validator (اداة المقارنه) :

المحتوى يتحقق ان كان عنصر الادخال يحوي قيمة تطابق القيمة في آخر .

4. Regular Expression Validator (اداة التعبير النظامي) :

المحتوى يتحقق ان كانت القيمة في عنصر الادخال تطابق تعبير نظامي محدد .

5. Custom Validator (الأداة المخصصة) :

يتحقق المحتوى تبعا للدالة التي عرفها المبرمج .

كل عنصر من عناصر التحقق من صحة المحتوى يمكننا اسناده الى عنصر ادخال واحد فقط .. في حين ان عنصر الادخال الواحد يمكن ان نسند اليه اكثر من عنصر تحقق من صحة المحتوى .

لو استخدمنا Range Validator و Regular Expression Validator و Compare Validator ... فان التحقق من صحة المحتوى تنجح ان كان عنصر الادخال فارغا لانه لا توجد قيمة من اجل التحقق من صحتها . ان لم يكن هذا ما نريده فيجب عندها اضافة الاداة RequiredField Validator لعنصر الادخال .

يوفر لنا ASP.NET الاداة Validation Summary وهي تختلف عن الادوات السابقة كونها لاتساعدنا في التحقق من صحة محتوى عنصر ادخال ، لكنها تفيدنا لتزويد قائمة بكل اخطاء عناصر التحقق من صحة المحتوى ضمن الصفحة .

معالجة المحتوى :

يمكننا استخدام ادوات التحقق من صحة المحتوى من اجل التحقق من صفحة بصورة تلقائية عندما يرسل المستخدم المعلومات او بصورة يدوية باستخدام الكود .
علما ان الطريقة الاولى هي الاكثر شيوعا .
عند استخدام التحقق التلقائي ، فان المستخدم يستقبل صفحة اعتيادية ويبدأ بملى المعلومات في عناصر الادخال وعندما ينتهي يقوم بنقر الزر لارسال الصفحة .

كل زر Button يمتلك خاصية Causes Validation والتي يمكن اسناد القيمة True لها او القيمة False .
ان ما يحدث عندما ينقر المستخدم الزر يعتمد على قيمة الخاصية Causes Validation .
اذا كانت القيمة False مسنده للخاصية Causes Validation فان ASP.NET سوف يهمل جميع عناصر التحقق من صحة المحتوى .. وسيتم ارسال الصفحة ومعالجة الكود بصورة طبيعية .
اذا كانت القيمة True مسنده للخاصية Causes Validation (وهي الحالة الافتراضية) .. فان ASP.NET بصورة تلقائية سيتحقق من صحة محتوى الصفحة عندما ينقر المستخدم الزر .. ويتم ذلك بتحقيق المحتوى لكل عنصر تحقق في الصفحة .
اذا فشل التحقق من المحتوى لاي عنصر ، فان ASP.NET سيرجع الصفحة مع بعض معلومات الاخطاء ، اعتمادا على اعداداتنا .

ملاحظة: أغلب ازرار الربط التي يمكن استخدامها لارسال الصفحة توفر لنا الخاصية Causes Validation ، مثل :
. LinkButton , ImgeButton , Bulleted List

التحقق من المحتوى من جهة العميل :

في اغلب المستعرضات الجديدة (Internet Explorer 5 او الاصدار الاحدث او اي اصدار من المستعرض Firefox) .
ASP.NET يضيف بصورة تلقائية كود JavaScript من جانب العميل ، وفي هذه الحالة عندما ينقر المستخدم زر (له الخاصية Causes Validation تساوي True) .. فان نفس رسائل الخطأ ستظهر دون الحاجة لارسال و رجوع الصفحة من الخادم .
على كل حال .. حتى لو تم التأكد من محتوى الصفحة من جهة العميل .. فان ASP.NET سيتحقق من المحتوى عند وصولها للخادم Server .
قد يقوم المستخدم بحذف كود التحقق من المحتوى من جهة العميل (كود JavaScript) ويستمر بالعمل مع الصفحة .. لذلك فان ASP.NET سيتأكد ان التطبيق يمكن الاعتماد عليه قدر الامكان .

الفئات The Validator Classes :

ان فئات ادوات التحقق يمكن ايجادها في فضاء الاسماء :

System.Web.UI.WebControls

والذي ينحدر من فئة التحقق الاساسية BaseValidator Class ، هذه الفئة تعرف الدوال الاساسية لعناصر التحقق من صحة المحتوى .

الخاصية ControlToValidate :

تحدد العنصر الذي سيتم التحقق من محتواه .. كل عنصر تحقق يمكنه التأكد من محتوى عنصر ادخال واحد .

الخاصية ErrorMessage :

ان لم ينجح التحقق من محتوى ، فان عنصر التحقق يمكنه عرض رسالة نصية (تسند للخاصية ErrorMessage) .

الخاصية Display :

تمكننا ان نحدد ان كانت رسالة الخطأ ستضاف ديناميا عند الحاجة او ان لها مساحة ثابتة محجوزه (Static) .. ان الحالة الثابتة Static مفيدة عندما يكون عنصر التحقق ضمن جدول ، ولاتريد ان يتغير عرض الخلية عند عرض رسالة الخطأ .
ويمكننا اختيار القيمة None لاختفاء رسالة الخطأ .

الخاصية ForeColor :

هذه الخاصية للتحكم بلون الخط .

الخاصية IsValid :

بعد اتمام التحقق من المحتوى ، فان هذه الخاصية تعيد True او False بالاعتماد على ان صحة المحتوى صحيحة او خاطئة .
يمكننا اختبار حالة الصفحة من خلال معرفة قيمة الخاصية IsValid بدل من التحقق من جميع عناصر التحقق من المحتوى .

: Enabled الخاصية

عند اسناد القيمة False لها .. فان التحقق التلقائي سوف لن يتم لهذا العنصر عند ارسال الصفحة .

: EnableClientScript الخاصية

عند اسناد القيمة True لهذه الخاصية فان ASP.NET سيضيف كود JavaScript و DHTML للسماح بالتحقق من المحتوى من جهة العميل في المستعرضات Browsers التي تدعم تلك الخاصية .

عند استخدام عناصر التحقق من صحة المحتوى .. فان الخصائص التي نحتاج لاسناد قيمة لها هي ..

ControlToValidate و Error Message

اضافة لذلك نحتاج لاسناد قيم اخرى لبعض العناصر وكما يلي :

: RequiredField Validator

لانحتاج لاسناد قيم لخصائص اخرى .

: Range Validator

اكبر قيمة Maximum Value ، اصغر قيمة Minimum Value ، النوع Type

: Compare Validator

العنصر الذي سيتم مقارنه محتواه ControlToCompare ، المعامل Operator ، النوع Type ، القيمة التي سيتم المقارنه معها ValueToCompare

: RegularExpression Validator

Validation Expression تعبير التحقق من المحتوى

: Custom Validator

دالة التحقق من المحتوى من جهة العميل ClientValidationFunction ، حدث التحقق للخادم ServerValidate Event

حتى هذه المرحلة تكلمنا عن ادوات التحقق من صحة المحتوى واخذنا فكرة عن اغلب ما يمكن ان نحتاج الى معرفته عن اساسيات هذه الادوات .

في الدرس القادم سنبدأ بالأمثلة التطبيقية واستخدام اغلب المفاهيم التي تعلمناها في هذا الدرس .

الى هنا وصلنا الى نهاية درسنا ..

ارجو ان تكونوا قد استقذتم من المعلومات المقدمة في هذا الدرس ..

ملتقانا في دروس قادمة ومعلومات جديدة ..

دعائي للجميع بالتوفيق

المهندس

عدي ياسين

بكالوريوس هندسة حاسوب

engauday@hotmail.com

الجمعة : 2007 / 10 / 26