

الفصل الخامس

سلسلة

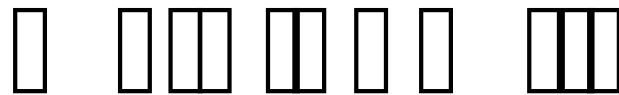
ASP.NET

خطوة بخطوة حتى الاحتراف

C# & VB

اعداد المهندس

محمد عمر الحاج خلف



الـتـيـمـات Themes

في هذا الفصل

- ✓ إنشاء الثيمات
- ✓ Skin ١١١١١١١١
- ✓ CSS ١١١١١١١١
- ✓ تطبيق الثيمات بشكل ديناميكي

يمكن للثيم الواحد أن يحتوي على العديد من ملفات الـ Skin يمكننا من تعديل الخصائص التي تتحكم بمظهر أدوات ASP.NET ، على سبيل المثال لنفترض أنك تريد تعديل مظهر جميع أدوات TextBox (التي كانت مضافة أم لم يتم إضافتها بعد لا فرق) حيث سنجعلها تظهر بخلفية شفافة .

App_Themes ولنسمه Simple ثم نضيف داخل هذا المجلد ملف Skin وليكن اسمه TextBox.skin حيث يمكن إضافته من القائمة File > New File > Skin File ونسميه TextBox.skin . نكتب داخله الكود التالي :

XML

```
<asp:TextBox BackColor="Yellow" BorderStyle="Dotted" Runat="server" />
```

الآن نضيف TextBox إلى صفحة .aspx . سمها ماشئت وأضف عليها أداة TextBox . ليصبح كود الصفحة كاملاً :

ASP.net

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

<!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>
</head>
<body>
<form id="form1" runat="server">
<div>
<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
</div>
</form>
</body>
</html>
```

نفذ الصفحة السابقة ولاحظ كيف ستظهر الأداة TextBox . Skin

٥٥٥٥٥٥

المجلدات الفرعية ضمن المجلد App_Theme ستترجم أثناء التنفيذ إلى صفوف Classes . يجب أعطائها أسماء مغايرة لأسماء الصفوف البرمجية .

(Named Skin) Skin ٥ ٥ ٥ ٥

٥ ٥ ٥ ٥ له في الفقرة السابقة يدعي بـ Skin ٥ ٥ ٥ ٥ (Default Skin) ، وهذا يعني بأن جميع أدوات الـ TextBox التي سنضيفها إلى تلك الصفحة سيتم تطبيق الـ Skin الافتراضي عليها ، على العموم بإمكاننا تسمية الـ Skin أي إعطاؤه معرف فريد وبالتالي نستطيع تحديد الأدوات التي ستطبق هذا الـ Skin بدلاً من تطبيقه على الجميع .

٥ ٥ ٥ ٥ ٥ ٥ ٥ ٥ App_Themes داخل مجلد الثيمات Simple2 ٥ ٥ ٥ ٥ ٥ ٥ Simple2 TextBox.skin ٥ ٥ واكتب فيه الكود التالي :

XML ٥ ٥

```
<asp:TextBox SkinID="DashedTextBox" BorderStyle="Dashed"
    BorderWidth="5px" Runat="Server" />

<asp:TextBox BorderStyle="Double" BorderWidth="5px" Runat="Server" />
```

٥ ٥ ٥ SkinID الخاصية وذلك بإعطاء الخاصية Skin ٥ ٥ ٥ ٥ ٥ ٥ ٥ ٥ DashdTextBox (وهو اسم اختياري) ، وهذا يعني أن أدوات الـ TextBox والتي يتم إسناد نفس الاسم السابق لخاصيتها SkinId فإنها ستطبق هذا الـ Skin ٥ ٥ ٥ ٥ ٥ ٥ TextBox ٥ ٥ ٥ ٥ ٥ ٥ والتي لا تحتوي على أي قيمة للخاصية SkinId فإنها ستخضع للـ Skin الافتراضي (وهو السطر الأخير من الكود السابق لأنه لا يملك اسم) . أضف صفحة جديدة واجعل قيمة الخاصية Theme تساوي القيمة Simple2 ، ثم أضف أداتين من النوع TextBox وأعط الأداة الأولى فقط القيمة DashdTextBox للخاصية SkinId واترك الثانية بلا تعديلات ثم نفذ لترى أن الأداة الأولى Skin ٥ ٥ ٥ ٥ المسمى ، في حين أن الثانية ستطبق الـ Skin ٥ ٥ ٥ ٥ . ٥ ٥ ٥ ٥ ٥ ٥ :

ASP.net ٥ ٥

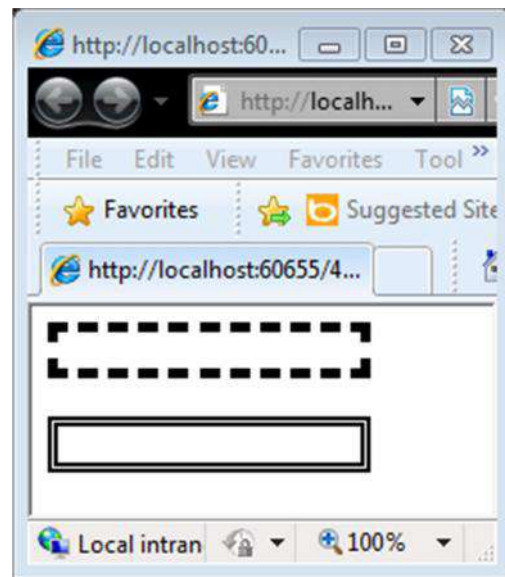
```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default2.aspx.cs"
    Inherits="Default2" Theme="Simple2" %>

<!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>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:TextBox ID="TextBox1" runat="server" SkinID="DashedTextBox" />
```

للتوضيح :

الصفحة الناتجة عن تنفيذ الفقرة السابقة



لنفترض أنك قمت بتغيير خصائص أداة ما ، ثم قمت بتطبيق ثيم على نفس هذه الأداة ، فأي الخصائص سيتم تطبيقها وقت العرض ؟

لتوضيح هذه الفكرة أنشئ صفحة جديدة وأضف إليها أداة Label وحيدة وغير لون الخلفية لهذه الصفحة باستخدام خاصية `BackColor="Blue"` (`BackColor="Blue"`) كما تعلمنا في الفقرات السابقة وليكن اسمه `show.skin` مع الثيمات المنشأة مسبقاً ، واكتب فيه الكود التالي :

```
<asp:Label BackColor="Red" Runat="Server" />
```

□ □ □ □ الصفحة فيصبح بالشكل التالي بعد تطبيق الثيم السابق عليها :

ASP.net □ □ □

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default3.aspx.cs"
Inherits="Default3" Theme="Simple3" %>

<!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>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:Label ID="Label1" runat="server" Text="Label" BackColor="Blue" />
    </div>
  </form>
</body>
</html>
```

□ □ □ □ □ □ □ □ □ □ إلى الآن هو إعطاء خاصية لون الخلفية للأداة Label اللون الأزرق ، ثم تطبيق ثيم يعطي نفس الخاصية السابقة اللون الأحمر ، فأيهما سيتم تطبيقه ؟ نفذ الصفحة السابقة لترى أنه سيتم تطبيق الخصائص المحددة بالثيم ، وبالتالي ظهور الأداة Label بخلفية حمراء .

لنقم بإجراء تعديل في الصفحة السابقة ، حيث نحذف الخاصية Theme="Simple3" □ □ □ عوضاً عنها StylesheetTheme="Simple3" ، أي يصبح كود الصفحة بعد التعديل بالشكل :

ASP.net □ □ □

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default3.aspx.cs"
Inherits="Default3" StylesheetTheme="Simple3" %>

<!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>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:Label ID="Label1" runat="server" Text="Label" BackColor="Blue" />
    </div>
  </form>
```

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default4.aspx.cs"
Inherits="Default4" Theme="Simple3" %>

<!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>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Label ID="Label1" runat="server" Text="Label" />
            <br />
            <asp:Label ID="Label2" runat="server" Text="Label" EnableTheming="False" />
        </div>
    </form>
</body>
</html>
```

```

    </form>
</body>
</html>

```

□ □ □ □ □ □ السابقة ولاحظ تطبيق الثيم Simple3 □ □ Label1 وعدم تطبيقه على الأداة Label2.

تسجيل الثيمات في الملف web.config

عوضاً عن إضافة الخاصية Theme □ □ StylesheetTheme □ □ □ □ □ □ □ □ □ □ الموقع على حدى ، فإنه من الأسهل تسجيل الثيم ضمن ملف الإعدادات web.config ليتم تطبيقه على كامل صفحات الموقع . ويتم تسجيل الثيم ضمن الملف web.config □ □ □ □ □ □ :

XML □ □ □

```

<configuration>
  <system.web>
    <compilation debug="true" targetFramework="4.0" />
    <pages theme="Simple3" />
  </system.web>
</configuration>

```

كما أنه □ □ □ □ □ □ استبدال الخاصية Theme ضمن الملف السابق بالخاصية StylesheetTheme لتطبيق الثيم المحدد .

بعد تسجيل الثيم ضمن الملف web.config ، فإنه بالإمكان إلغاء الثيم عن صفحة ما وذلك من خلال إعطاء الخاصية EnableTheming التابعة للصفحة المحددة القيمة False □ □ □ □ □ □ □ □ □ □ صفحة جديدة وأضف عليها أداة Lable وقم بتسجيل الثيم Simple3 في ملف الإعدادات كما هو موضح في الكود بالأعلى ثم نفذ الصفحة ولاحظ ظهور الأداة Label بخلفية حمراء وهذا دليل على تطبيق الثيم المحدد ، عد إلى وضع التصميم وعدل الخاصية EnableTheming إلى القيمة False ليصبح كود الصفحة بالشكل التالي :

ASP.net □ □ □

```

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default5.aspx.cs"
Inherits="Default5" EnableTheming="False" %>

<!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>Themes in ASP.NET</title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
    </div>
  </form>
</body>
</html>

```

وبالتالي لن يتم تطبيق الثيم على هذه الصفحة .

إلى CSS

كبدل للـ Skins وهو الأسلوب الأكثر شيوعاً ، حيث نلاحظ أن معظم المطورين والمواقع تعتمد على هذه الملفات . سنقوم ضمن هذه الفقرة بإنشاء ثيم جديد يعتمد على CSS .

١١١١١١١١

من الأفضل أن تقوم بإنشاء موقع جديد لتنفيذ هذه الفقرة ومايلها ، أو إن إردت الاستمرار بالعمل على الموقع القديم ، فقم بإنشاء ملف جديد في web.config إلى وضعه الأصلي دون التعديلات التي قمنا بإضافتها في الفقرة السابقة .

١١١١١١١١

فهي بحد ذاتها تقنية غنية جداً ، ولكن سأكتفي ضمن هذه الفقرات بتوضيح كيفية تطبيقها وربطها مع .aspx

App_Theme (إنشئه إن لم يكن منشأ مسبقاً) ثم StyleTheme CSS ونسمه SimpleStyle.css New File File وبالتالي تصبح لدينا البنية التالية :

App_Themes\StyleTheme\SimpleStyle.css

نفتح الملف السابق ونكتب داخله :

```
html
{
    background-color:gray;
    font:14px Georgia,Serif;
}
label
{
    font-weight:bold;
}
input
{
    background-color:Yellow;
    border:double 3px orange;
}
.button
{
    background-color:#eeeeee;
}
.content
{
    margin:auto;
    width:600px;
    border:solid 1px black;
    background-color:White;
    padding:10px;
}
```

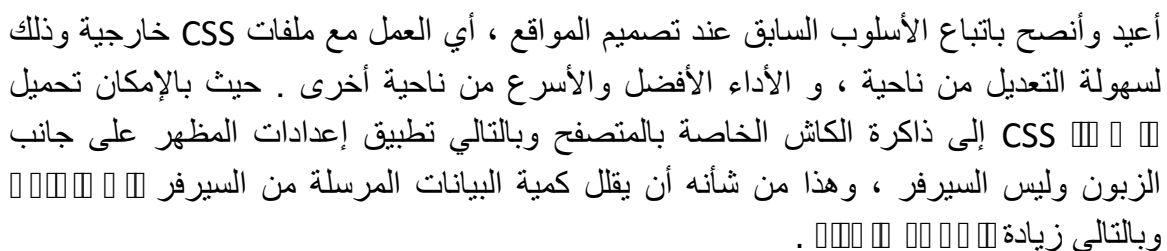
■


```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default6.aspx.cs"
Inherits="Default6" Theme="StyleTheme" %>

<!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>
</head>
<body>
    <form id="form1" runat="server">
        <div class="content" >
            <asp:Label ID="Label1" runat="server" Text="First Name : " />
            <br />
```

وبالتالي سوف يتم تطبيق الإعدادات المحددة في ملف CSS ضمن التيم وتكون نتيجة تنفيذ الصفحة السابقة مشابهة للتالي :



كما أنه بالإمكان تنظيم فقرات الصفحات وعمل الكثير من التأثيرات الرائعة عبر مزايا الـ CSS
 ، ولرؤية مايمكن عمله بهذه التقنية ، أنصح بزيارة الموقع : www.dreamw...

كما أن هنالك الكثير من المواقع التي توفر ملفات وأكواد CSS مجانية .

سنقوم في هذه المثال بتطبيق ملف CSS على صفحة وبشكل مباشر دون تطبيق ثيم ، حيث سنقوم بتنظيم فقرات الصفحة ضمن ملف الـ CSS ، ومن ثم ربطه مع الصفحة كرابط `Link` `App_Theme` واكتب فيه `Float.css` وليكن اسمه `Float.css` وضعه داخل المجلد `App_Theme` واكتب فيه :

```
html
{
    background-color:Silver;
    font:14px Arial,Sans-Serif;
}
#div3
{
    position:absolute;
    left:15px;
    top:15px;
    width:200px;
    padding:10px;
    background-color:White;
}
#div2
{
    position:absolute;
    left:250px;
    top:65px;
    width:200px;
    padding:10px;
    background-color:White;
}
#div1
{
    position:absolute;
    left:485px;
    top:115px;
    width:200px;
    padding:10px;
    background-color:White;
}
```

الاعدادات السابقة على كل منها .
 aspخ جديدة ، حيث سنقوم بإضافة ثلاث فقرات (div) عليها ليتم تطبيق

ASP.NET 4.5

```

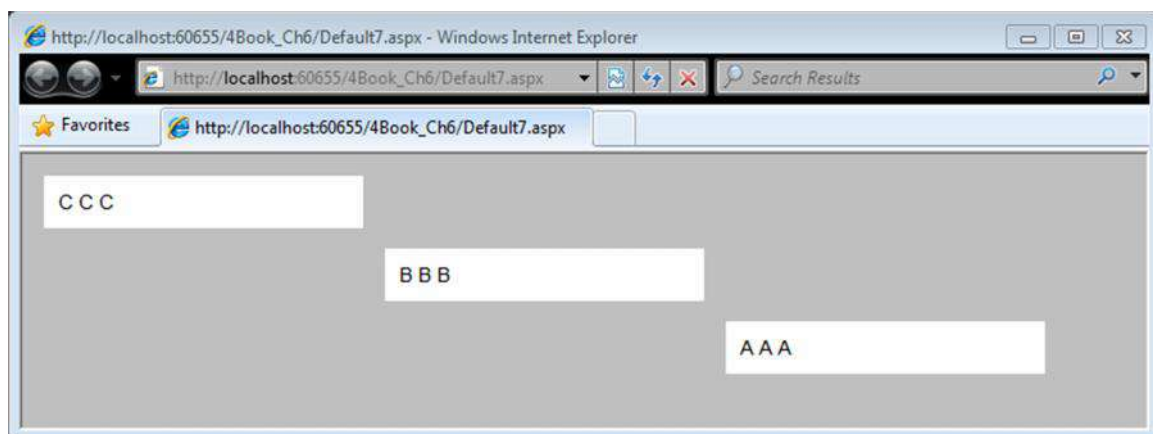
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default7.aspx.cs"
Inherits="Default7" %>

<!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>
<link href="App_Themes/Float.css" rel="stylesheet" type="text/css" />
</head>
<body>
<form id="form1" runat="server">
<div id="div1">
A A A
</div>
<div id="div2">
B B B
</div>
<div id="div3">
C C C
</div>
</form>
</body>
</html>

```

وعند تنفيذ الصفحة السابقة يكون الناتج كالتالي :



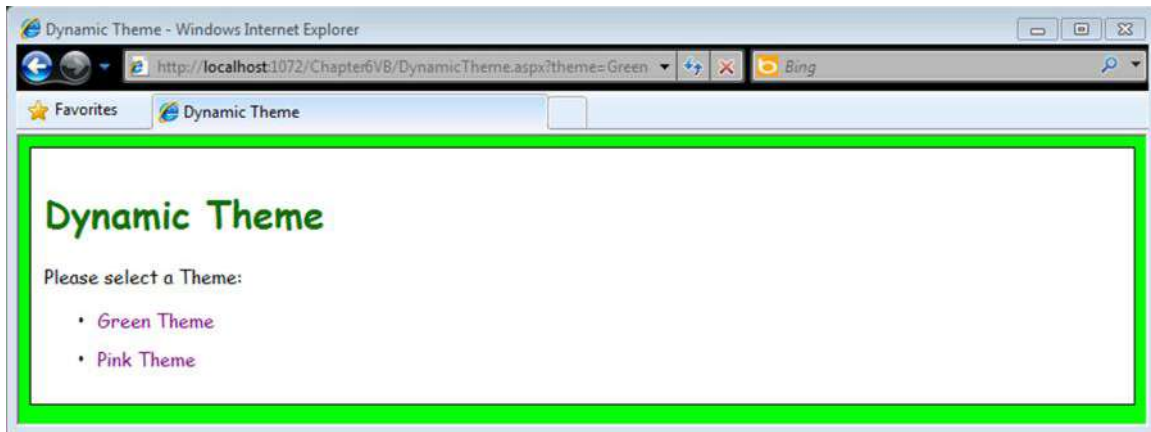
اختيار الـ Theme بحسب رغبة المستخدم

بالإمكان إنشاء أكثر من Theme للموقع والسماح لكل زائر أن يختار الـ Theme الذي يرغب بتطبيقه وبالتالي
 يمكن توفير الـ Theme الذي سيتم تطبيقه عند إطلاق

أول حدث لبدء تحميل الصفحة (PreInit) ، وبالتالي لتطبيق ثيم بشكل برمجي وقت التنفيذ فإننا نضيف الكود المطلوب إلى الحدث السابق .

تطبيق :

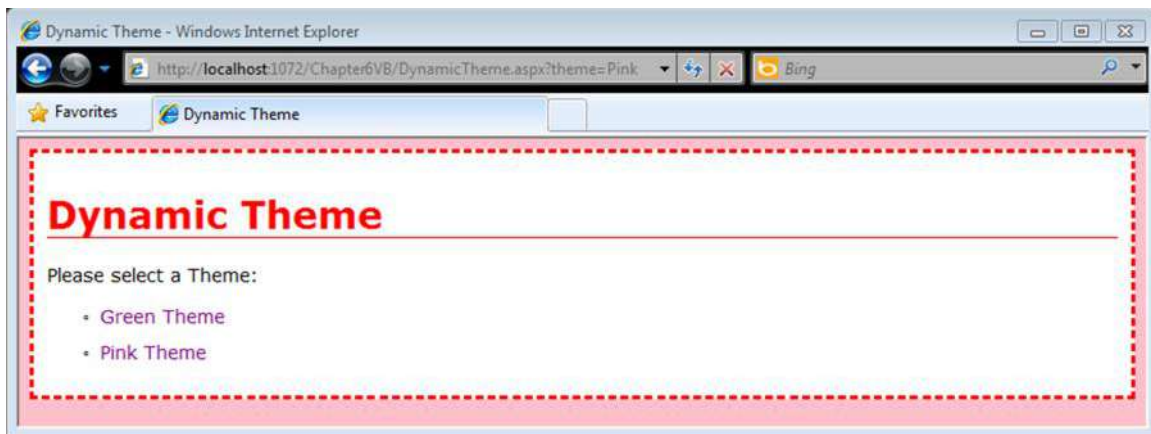
نريد من خلال هذا التطبيق إنشاء صفحة شبيهة بالتالي :



سنقوم بإنشاء ثيمين ، حيث يقوم المستخدم باختيار الثيم الذي يريد تطبيقه :

✓ GreenTheme : حيث يجعل الخلفية □ □ □ □ □ □ □ □ □ □ ، ويحدد تنسيقات معينة □ □ □ □ الأدوات كما تظهر الصورة السابقة .

✓ PinkTheme : يقوم هذا الثيم بجعل لون الخلفية □ □ □ □ □ □ □ □ □ □ ، ويحدد تنسيقات معينة لباقي الأدوات كما تظهر الصورة التالية .



يتم تطبيق نفس الثيم عندما يقوم نفس المستخدم بزيارة الموقع مرة أخرى في المستقبل (أي الاحتفاظ بخياره المفضل) . وذلك عبر تخزين خيار المستخدم في ملف خاص به (profile) □ □ السيرفر .

لنقم بدايةً بإنشاء الـ Theme السابقين ، الـ Theme الأول GreenTheme ونضيف بداخله ملف CSS سمه الـ GreenTheme.css واكتب فيه الكود التالي :

GreenTheme CSS

```
html
{
    background-color:#00FF00;
    font:14px Comic Sans MS;
}

a { text-decoration:none; }

h1 { color:#16700a; }

li { margin-bottom:10px; }

.content
{
    background-color:White;
    padding:10px;
    border:solid 1px black;
}
```

الـ Theme الثاني PinkTheme ونضيف بداخله ملف CSS نكتب فيه الكود التالي :

PinkTheme CSS

```
html
{
    background-color:Pink;
    font:14px Verdana;
}

a { text-decoration:none; }

h1 { color:red; border-bottom:solid 1px red; }

ul { list-style-type:circle; }

li { margin-bottom:10px; }

.content
{
    background-color:White;
    padding:10px;
    border:Dashed 3px red;
}
```

وبهذا نكون قد انتهينا من إعداد الثيمات .

لنقم الآن بإضافة صفحة جديدة للموقع Default.aspx ونقوم بتصميمها لتظهر كما تم توضيحه بالأعلى ، ويكون كود الصفحة كالتالي :

ASP.net

```
<form id="form1" runat="server">
  <div class="content">

    <h1>Dynamic Theme</h1>

    Please select a Theme:
    <ul>
      <li>
        <a href="DynamicTheme.aspx?theme=Green">Green Theme</a>
      </li>
      <li>
        <a href="DynamicTheme.aspx?theme=Pink">Pink Theme</a>
      </li>
    </ul>

  </div>
</form>
```

نلاحظ أننا قمنا بتمرير اسم الثيم كبارامتر ضمن عنوان الصفحة URL ، حيث ستم الاعتماد على القيمة URL التي يختار فيها المستخدم ثيم ، أما في المرات اللاحقة فنقوم بتحديد خياره السابق من خلال الملف Profile الخاص به لإعداد هذا الملف نفتح الملف web.config ونضيف عليه الكود التالي :

XML

```
<?xml version="1.0"?>
<configuration>
  <system.web>
    <profile>
      <properties>
        <add name="UserTheme"/>
      </properties>
    </profile>
    <compilation debug="true"/>
  </system.web>
</configuration>
```

```
: 00000000 00000000 00000000 0000 Page_PreInit 00000000
```

```
void Page_PreInit(object s, EventArgs e)
{
    if (!String.IsNullOrEmpty(Request["theme"]))
    {
        switch ((Request["theme"]))
        {
            case "Green":
                Profile.UserTheme = "GreenTheme"; break;
            case "Pink":
                Profile.UserTheme = "PinkTheme"; break;
        }
    }
    Theme = Profile.UserTheme;
}
```

```
Sub Page_PreInit(ByVal s As Object, ByVal e As EventArgs)
    If Not IsNothing(Request("theme")) Then
        Select Case (Request("theme"))
            Case "Green"
                Profile.UserTheme = "GreenTheme"
            Case "Pink"
                Profile.UserTheme = "PinkTheme"
        End Select
    End If
    Theme = Profile.UserTheme
End Sub
```

[illegible]

إلى هنا نأتي إلى نهاية هذا الفصل ، والذي استعرضنا فيه بعض `UI` المتبعة في تنسيق تصاميم مواقع الانترنت عبر الثيمات ، حيث تعلمنا في البداية كيفية استخدام الـ `Skin` المسماة ، كما تعلمنا كيفية تطبيق الثيم عبر الخصائص `Theme` `StyleSheetTheme` بينهما ، كما استعرضنا لاحقاً كيفية إضافة ملفات الـ `CSS` إلى الثيمات وكيفية تطبيقها بشكل مباشر



: 

سوريا - □ □ □ □ □ □ □ □ □ □