

## الفصل الخامس المزيد عن أدوات الويب

تعرفنا في الفصل السابق على مفهوم أدوات الويب وتقسيماها المختلفة، وفي هذا الفصل نستكمل ما بدأناه حيث نتناول عدداً من الأدوات الأخرى. بانتهاء هذا الفصل، ستتعرف على:

- إنشاء القوائم البسيطة.
- إظهار العناصر في أعمدة متعددة.
- إنشاء الجداول.
- استخدام قوائم ربط البيانات.
- استخدام أداة التقييم.

بالإضافة إلى إدخال المستخدم للبيانات من خلال مربعات النصوص، كثيراً ما يتم استقبال هذه البيانات عن طريق إظهار قائمة تحتوي على مجموعة من الخيارات التي يمكنه الاختيار من بينها. وحتى إذا لم ترغب أن يقوم المستخدم باختيار البيانات من خلال القوائم، يمكنك استخدامها في عرض البيانات في صورة منظمة وجذابة.

يحتوي النموذج البرمجي لنماذج الويب على العديد من الأدوات التي تمكنك من إظهار الأنواع المختلفة من القوائم بدايةً من عمود واحد يحتوي على القيم الحرفية وانتهاءً بشبكة البيانات DataGrid المرتبطة بقاعدة بيانات. سنقوم فيما يلي بإلقاء نظرة سريعة على أدوات تحكم خادم ASP.NET المستخدمة في إضافة القوائم المختلفة إلى تطبيقاتك، فكن معنا.

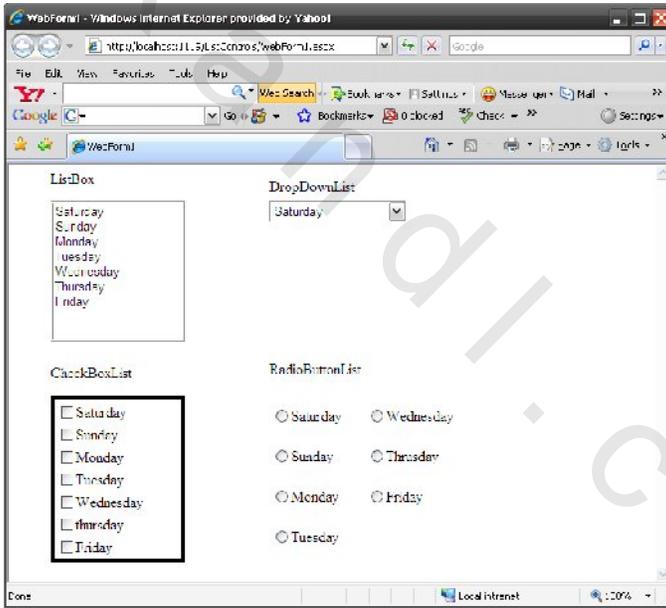
## إنشاء القوائم البسيطة

لإظهار قائمة بسيطة من البيانات الحرفية داخل تطبيقاتك، يمكنك الاختيار بين أدوات تحكم الخادم التالية:

- مربع السرد **ListBox** ويقوم بإظهار عدد من العناصر في نفس الوقت، ويمكن للمستخدم من خلاله تحديد أكثر من خيار تبعاً لإعدادات الخصائص التي تقوم بتحديدتها.
- القائمة المنسدلة **DropDownList** وتتيح هذه القائمة للمستخدم اختيار عنصر واحد فقط من قائمة من العناصر وتأخذ نفس مساحة مربع النص لأن قائمة العناصر تختفي بمجرد انتهاء المستخدم من عملية الاختيار.
- قائمة مربعات الاختيار **CheckBoxList** وتتيح هذه الأداة للمستخدم اختيار أكثر من عنصر حيث تظهر علامة اختيار بجوار كل عنصر مختار، وهي بعكس مربع السرد، يمكنها إظهار أعمدة متعددة.
- قائمة خانات الاختيار **RadioButtonList** وتعمل هذه الطريقة بنفس طريقة عمل قائمة مربعات الاختيار، الفرق الوحيد أنها تتيح للمستخدم اختيار خانة

واحدة فقط في الوقت الواحد.

وكما تتوقع فإن معظم هذه الأدوات تعمل بنفس طريقة عمل مثيلاتها في تطبيقات Windows على الرغم من تقييد وظائف هذه الأدوات في حالة تطبيقات الويب. فمربع السرد **ListBox** على سبيل المثال لا يحتوى على إمكانية الترتيب التلقائي لعناصره مثلما الحال في الأداة الموجودة في **Windows** كما أنها تستطيع فقط تخزين القيم البسيطة والسبب في ذلك كما ذكرنا من قبل يرجع إلى الإمكانيات المحدودة لمستعرضات الويب والتي لا زالت تتعامل مع كود **HTML**.  
يوضح شكل ٥-١ التالي كيف تبدو الأدوات الأربعة السابقة للمستخدم داخل صفحة الويب.



شكل ٥-١ الأشكال المختلفة لأدوات السرد كما تبدو للمستخدم داخل صفحة الويب

### استخدام التصنيف **ListControl**

تشتق أدوات السرد الأربعة السابقة من التصنيف **ListControl**، لذا فحينما نتعرف على كيفية العمل مع العناصر الموجودة بإحدى هذه الأدوات، يمكنك بسهولة تامة

استخدام الأدوات الأخرى.

تحتوى كل أداة من أدوات السرد على الخاصية **Items** المستخدمة لاحتواء عناصر الأداة، كما يحتوى كل عنصر من هذه العناصر على الخاصية **Text** التى تحتوى على النص الذى يظهر للمستخدم داخل صفحة الويب، بالإضافة إلى الخاصية **Value** المرتبطة بالعنصر المعروض إلا أنها لا تظهر على الشاشة.

يمكنك إضافة العناصر أثناء التصميم بنفس طريقة إضافتها فى الأدوات المصاحبة فى **Windows**. إلا أننا سنقوم هنا بتوضيح دالة مستخدمة فى ملئ أداة السرد بالعناصر من خلال الكود وذلك كما يلي:

```
Private Sub ListWeekDays(ByVal MyList As ListControl)
Dim DayNames() As String = {"Saturday", "Sunday", "Monday",
"Tuesday ", "Wednesday", "Thursday", "Friday"}
Dim DayValues() As Integer = {1,2,3,4,5,6,7}
Dim i As Integer
For i = 0 To 6
    MyList.Items.Add(New ListItem(DayNames(i), DayValues(i)))
Next
End Sub
```

تقوم هذه الدالة كما ترى بتعيين قيم الخصائص **Value** و **Text** من خلال دالة إنشاء التصنيف **ListItem**، حيث تعمل هذه الدالة مع أدوات السرد سألفة الذكر.

لست فى الحقيقة مطالباً بملء قيمة الخاصية **Value** المصاحبة لأدوات السرد، حيث يتم تلقائياً تخصيصها بنفس قيم الخاصية **Text** المناظرة.



تتيح جميع الأدوات المشتقة من التصنيف **ListControl** للمستخدم اختيار مدى من العناصر، حيث تم تصميم الأدوات **DropDownList** و **RadioButtonList** خصيصاً للمواقف التى يجب على المستخدم فيها اختيار عنصر واحد فقط من القائمة المتاحة، بينما تتيح الأداة **CheckBoxList** للمستخدم اختيار عنصر واحد أو أكثر أو ربما عدم اختيار أى عنصر على الإطلاق. أما الأداة **ListBox** فتحتوى على نمطين من أنماط الاختيار والتى تحدد إذا ما كان المستخدم يستطيع اختيار عنصر واحد فقط أم أكثر من عنصر فى

نفس الوقت. وفي جميع الأحوال يحتوى كل عنصر من العناصر الموجودة بالقائمة على الخاصية **Selected** المصاحبة التي يمكنك استخدامها داخل الكود للتعرف على مدى اختيار العنصر من عدمه أثناء تشغيل البرنامج كما في الكود التالي:

```
Dim Templtem As ListItem
For Each Templtem In MyList.Items
If Templtem.Selected Then
lblInfo.Text &= "You Selected " & Templtem.Text & vbCrLf
End If
Next
```

يستخدم هذا الكود دواراة **For** لاختبار كل عنصر من العناصر الموجودة بالقائمة ، وهذا على فرض إمكانية اختيار أكثر من عنصر. أما إذا كانت القائمة لا تسمح باختيار أكثر من عنصر في الوقت الواحد، فيمكنك استخدام كود أبسط من السابق كما يلي:

```
If Not IsNothing(MyList.SelectedItem) Then
lblInfo.Text = "You selected " & MyList.SelectedItem.Text
Else
lblInfo.Text = "No item selected!"
End If
```

وكما ترى، تقوم الخاصية **SelectedItem** بإرجاع العنصر المختار داخل القائمة.

إذا قمت باستخدام الخاصية **SelectedItem** مع أداة تدعم اختيار أكثر من عنصر في نفس الوقت، يتم الحصول على أقل عنصر في الترتيب بين العناصر المختارة.



لإنشاء الاختيارات الافتراضية التي تظهر بمجرد تشغيل التطبيق، قم بتخصيص القيمة **True** للخاصية **Selected** المصاحبة للعناصر التي ترغب في اختيارها.



يمكنك أيضاً الوصول إلى عناصر القائمة من خلال أرقام الدليل الرقمية المخصصة لهذه العناصر، حيث يتم تخصيص الدليل 0 للعنصر الأول والدليل 1 للعنصر الثاني وهكذا. يمكنك باستخدام هذه الأدلة تخصيص القيم للعناصر أو استرجاعها منها كما في الكود التالي:

**MyList.Items(2).Text = "Mohamed"**

كما يمكنك معرفة دليل العنصر المختار من خلال الخاصية **SelectedIndex** والتي تحتوي على الرقم 1- في حالة عدم اختيار أى عنصر.

يمكنك كذلك معرفة عدد العناصر الموجودة بالقائمة من خلال الخاصية **Count** مع الأخذ في الاعتبار أن أدلة العناصر الموجودة بالقائمة تبدأ دائماً من الرقم 0، فإذا قامت الخاصية **Count** بإرجاع القيمة 20 على سبيل المثال، فهذا يعنى احتواء القائمة على عشرين عنصراً تبدأ من الدليل 0 وتنتهى بالدليل 19.

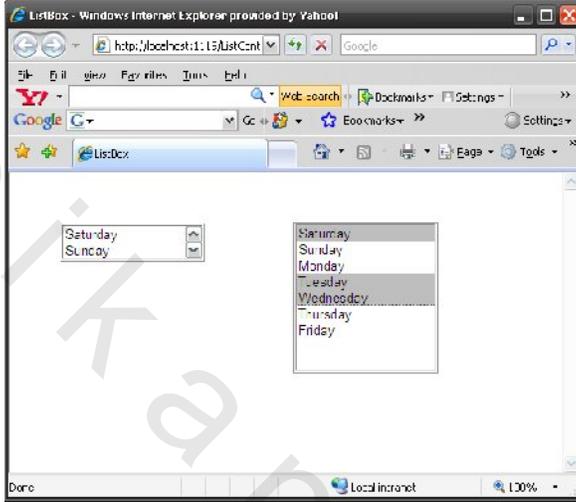
تحتوى القوائم أيضاً على العديد من الوظائف المفيدة المستخدمة للتحكم فى القائمة عن طريق إجراء العمليات المختلفة كالبحث والحذف، حيث يوجد وظيفتان للبحث إحداهما الوظيفة **FindByText()** المستخدمة فى البحث عن العناصر عن طريق محتويات الخاصية **Text**، والوظيفة **FindByValue()** المستخدمة للبحث عن العناصر عن طريق محتويات الخاصية **Value**. وهذا بالإضافة إلى الوظيفة **Remove()** المستخدمة فى حذف العناصر من القائمة كما فى الكود التالى:

```
Templtem = MyList.Items.FindByText("Monday")
If Not IsNothing(Templtem) Then
    MyList.Items.Remove(Templtem)
End If
```

**استخدام مربع السرد ListBox**

يتيح مربع السرد للمستخدم عرض أكثر من عنصر على الشاشة فى نفس الوقت كما يقوم تلقائياً بإظهار شريط تمرير إذا زاد عدد العناصر المعروضة عن المساحة المحددة. وكما هو الحال مع أدوات التحكم الأخرى، يمكنك تحديد حجم مربع السرد أثناء رسمه على النموذج أو من خلال الخاصيتين **Height** و **Width** بمربع الخصائص، إلا أن مربع السرد يتيح لك التحكم فى ارتفاع الأداة طبقاً لعدد العناصر التى ترغب فى إظهارها على الشاشة فى الوقت الواحد وذلك من خلال الخاصية **Rows** والتى يجب لاستخدامها حذف القيمة الموجودة بالخاصية **Height** التى تستخدم بدورها فى تحديد ارتفاع الأداة بغض النظر عن

عدد العناصر التي تظهر على الشاشة. يوضح شكل ٥-٢ التالي مربعي سرد، في الأول تم تخصيص القيمة 2 للخاصية Rows لعرض صفين فقط على الشاشة (المربع الأيسر)، بينما تم تحديد الارتفاع في مربع السرد الأيمن من خلال الخاصية Height.



شكل ٥-٢ تعيين ارتفاع مربع السرد

من الشكل السابق، تلاحظ اختيار أكثر من عنصر داخل مربع السرد الأيمن وهذا يرجع إلى تغييرنا لقيمة الخاصية SelectionMode من Single إلى Multiple. وعلى ذلك يمكنك اختيار العناصر المختلفة بنفس طريقة الاختيار في Windows.

تأكد من تعيين الحجم المناسب لمربع السرد تبعاً لعدد العناصر التي يحتويها. فعلى سبيل المثال من الصعب أن ينتقل المستخدم بين قائمة تحتوي على ما يقرب من مائة عنصر إذا تم تحجيم مربع السرد بحيث يقوم بإظهار ثلاثة عناصر في الوقت الواحد. ففي هذه الحالة من الأفضل التفكير في تقسيم القائمة إلى أكثر من قائمة فرعية وذلك لتقليل عدد الخيارات.



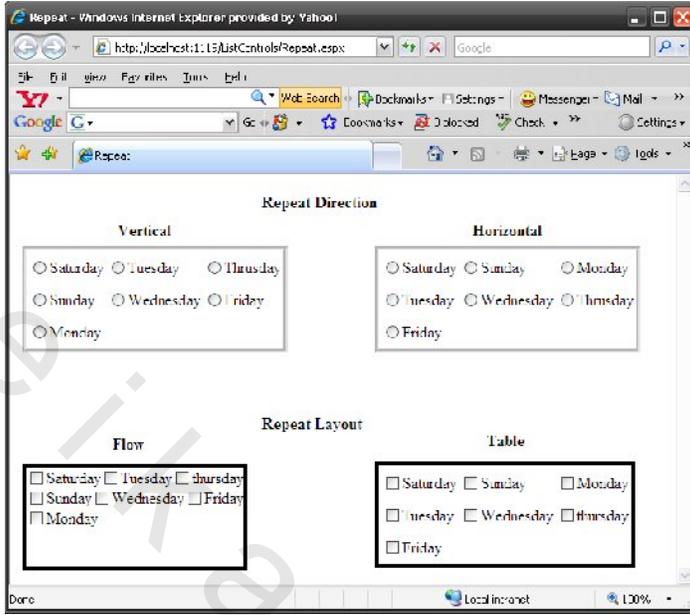
يمكنك استخدام لوحة المفاتيح للتنقل بين عناصر مربع السرد وذلك من خلال المفاتيح PageUp و PageDown و Home و End ومفاتيح الأسهم. كما يمكنك في القوائم المرتبة كتابة الحرف الأول من الكلمة التي ترغب في عرضها ليتم الذهاب إليها مباشرة.



### إظهار العناصر في أعمدة متعددة

يتم دائماً وضع عناصر مربع السرد أو القائمة المنسدلة رأسياً في عمود واحد فقط، بينما تحتوي قائمة مربعات الاختيار وقائمة خانات الاختيار على إمكانية وضع العناصر في أكثر من عمود وذلك من خلال خاصيتين هامتين هما:

- الخاصية RepeatColumns ويمكنك من خلالها تحديد عدد الأعمدة التي تحتاجها داخل القائمة.
  - الخاصية RepeatDirection وتحدد اتجاه ملئ القائمة، أفقياً أو رأسياً.
  - الخاصية RepeatLayout وتحدد نمط وضع العناصر داخل القائمة سواءً في صورة جدول Table أو بصورة متتالية Flow.
- يوضح شكل ٥-٣ التالي استخدام الخصائص الثلاثة السابقة.



شكل ٥-٣ استخدام خصائص عرض عناصر أدوات السرد

إذا أردت ملء أداة السرد متعددة الأعمدة من خلال الكود، من الأفضل في هذه الحالة حذف القيمة الموجودة بالخاصية **Width** أثناء عملية التصميم.



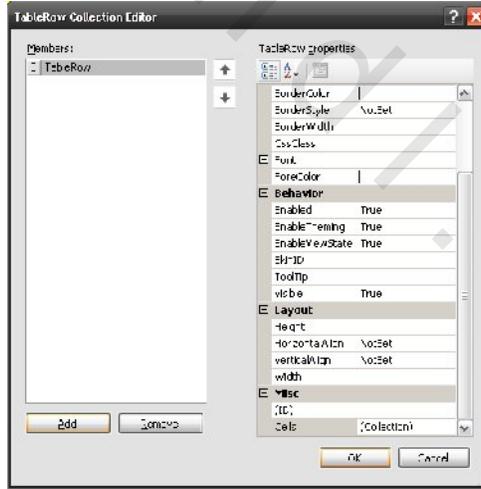
## إنشاء الجداول

على الرغم من استخدام أدوات السرد لعرض المعلومات والاختيار من بينها إلا أنها تفتقر كثيراً إلى سمات التنسيق التي ربما أردت استخدامها في تحسين واجهة هذه البيانات لعرضها بصورة جذابة. وعلى الجانب الآخر فإن جداول **HTML** من العناصر المعتادة في صفحات الويب المختلفة والتي تستخدم لعرض البيانات في صورة جدول يحتوي على العديد من التنسيقات كإظهار البيانات في صورة أعمدة محتوية على رؤوس لتوضيح محتويات كل عمود على حده. يمكنك إنشاء الجداول داخل نماذج الويب من خلال الأداة **Table** الموجودة بشرط الأدوات، حيث يتكون الجدول من مجموعة من الصفوف **Rows** التي تحتوي بدورها على مجموعة من الخلايا **Cells**. وعلى ذلك يمكنك استخدام التصنيفات

**Table** و **TableRow** و **TableCell** للحصول على الوظائف المعتادة عند إنشاءك للجدول من خلال HTML.

للتعرف على كيفية استخدام الأداة **Table** في إضافة جدول إلى النموذج، تابع معنا الخطوات الآتية:

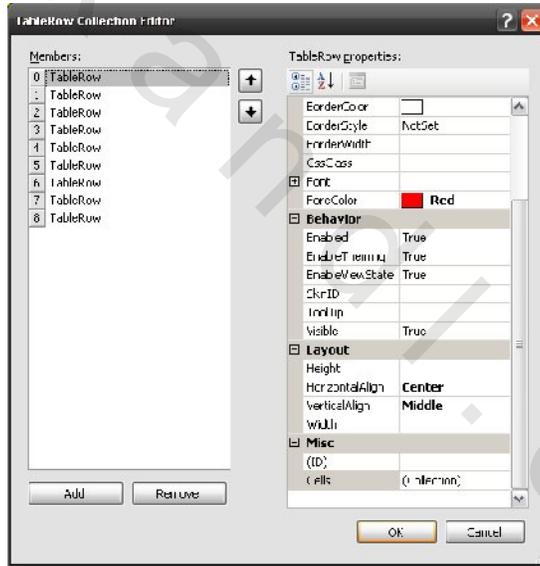
١. قم بإنشاء تطبيق ويب جديد أو قم بإضافة نموذج جديد إلى التطبيق الحالي مع جعل هذا النموذج هو صفحة البدء.
٢. قم بإدراج الأداة **Table** من مربع الأدوات إلى النموذج.
٣. من مربع الخصائص، قم بتعيين اسم مناسب للجدول وليكن **NamesTable**.
٤. لإضافة الصفوف والحلایا إلى الجدول، انقر الزر المجاور للخاصية **Rows** داخل مربع الخصائص، تظهر النافذة المعتادة لإضافة العناصر (انظر شكل ٥-٤).
٥. انقر زر **Add** لإضافة الصف الأول بالجدول وهو الصف الذى يحتوى على رؤوس الأعمدة.



شكل ٥-٤ إضافة الصفوف إلى الجدول

٦. قم بتعيين الخصائص المختلفة للصف من العمود الأيمن بالنافذة مثل لون الخط المستخدم في الصف ومواصفات هذا الخط ولون الخلفية وغيرها من الخصائص

- المتعادة لك والتي سنتعرف على جزءٍ منها بعد قليل.
٧. لإضافة الخلايا إلى الصف الجديد، انقر الزر المجاور للخاصية **Cells** حيث تظهر نافذة أخرى شبيهة يمكنك منها إضافة الخلايا إلى الصف وتعيين نص كل خلية ومحاذاتها بالإضافة إلى إمكانية تخصيصها بلون وخلفية مميزة عن الصف بالكامل (انظر شكل ٥-٥).
٨. قم بتكرار الخطوات الثلاث السابقة لإضافة عدد من الصفوف والخلايا ثم أعد تحجيم أداة الجدول لتظهر كما تريد داخل مستعرض الويب.
٩. قم بتشغيل التطبيق لمعاينة النموذج، تلاحظ ظهور الجدول بالإعدادات التي حددتها (انظر شكل ٥-٦).



شكل ٥-٥ إضافة الخلايا إلى الصف

The screenshot shows a web browser window titled 'TableControl - Windows Internet Explorer provided by Yahoo!'. The address bar shows 'http://localhost:11976/...' and the search bar contains 'Google'. The main content area displays a table with the following data:

First Name	Last Name	Age	Position
Sami	Alshibshishi	39	Manager
Waeed	Abdrazek	27	ITM Assistant
Nagat	Ismael	27	IS Assistant
Sobhy	Omrin	31	Sales Manager
Mohamed	Nageeb	21	Maint. Assistance
Giada	Farook	27	Secretary
Mokhtar	Sary	28	Accountant
Mohamed	Eman	25	Store Assistant

شكل ٥-٦ مظهر الجدول داخل مستعرض الويب

يمكنك أيضاً إضافة العناصر إلى الجدول من خلال الكود كما في الكود التالي:

**Private Sub Page\_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load**

**Dim rowTemp As TableRow**

**Dim cellTemp As TableCell**

**Dim i As Integer**

**rowTemp = New TableRow()**

**For i = 0 To 3**

**cellTemp = New TableCell()**

**rowTemp.Cells.Add(cellTemp)**

**cellTemp = Nothing**

**Next**

**rowTemp.Cells(0).Text = "First Name"**

**rowTemp.Cells(1).Text = "Last Name"**

**rowTemp.Cells(2).Text = "Age"**

**rowTemp.Cells(3).Text = "Last Name"**

**NamesTable.Rows.Add(rowTemp)**

**End Sub**

التحكم في مظهر الجدول

تستخدم الخاصية **Text** لتعيين النص الذي يظهر داخل الخلية، هذا إلى جانب العديد من

الخصائص الأخرى المستخدمة للتحكم في مظهر كل خلية على حده أو كل صف على حده. فعلى سبيل المثال هناك الخاصية **BackColor** المستخدمة لتعيين خلفية الخلية أو الصف والتي يمكنك تعيينها من خصائص نافذة إضافة الخلية أو الصف (راجع شكلي ٥-٤ و ٥-٥) أو من خلال الكود كما يلي:

```
CurRow.BackColor = Color.FromName("#AAAADD")
```

يمكنك بنفس الطريقة تعيين العديد من الخصائص والتي أهمها ما يلي:

- يمكنك تعيين الإطارات **Borders** الخيطة بكل خلية داخل الجدول من خلال الخصائص **GridLines** و **BorderStyle** و **BorderWidth**.
- يمكنك تعيين محاذاة نص الخلايا **Alignment** من خلال الخاصيتين **HorizontalAlign** و **VerticalAlign**.
- يمكنك التحكم في حجم الخلايا من خلال الارتفاع **Height** والعرض **Width** وإذا ما كان النص قابل للانتقال إلى سطر جديد من خلال الخاصية **Wrap**.
- يمكنك تعيين خصائص الخط والألوان سواءً على الجدول بالكامل أو على مستوى كل صف أو خلية على حده. كما يمكنك تعيين صورة تظهر في خلفية الجدول من خلال الخاصية **BackImageURL**.

#### استخدام HTML داخل الخاصية Text

أحد ميزات استخدام النموذج البرمجي لنماذج الويب أنك كمبرمج غير مطالب بالتعامل مع HTML. ولكن لأن كود HTML هو الذى يصل بالفعل إلى المستعرض، فكل ما تقوم بإنشائه يتحول عند المستعرض إلى كود HTML، فربما تحتاج أحياناً إلى كتابة هذا الكود بنفسك لإجراء المزيد من التحكم. فعلى سبيل المثال، إذا سبق لك إنشاء الجداول في HTML من خلال الرمز **<TABLE>** فلا شك أنك تعرف أنه بإمكانك وضع ما تريد داخل خلايا الجدول مثل الارتباط التشعبي أو حقل الإدخال. يمكنك بالتأكيد أداء ذلك داخل **ASP.NET** من خلال الأداة **Table** بإدخال كود HTML داخل الخاصية **Text** سواءً أثناء التصميم أو وقت التشغيل كما يلي:

```
CurrCell.Text = "<A HREF=detail.aspx?id=" "</A>"
```

وبالتالي بمجرد ظهور الجدول داخل المستعرض، يظهر الارتباط `detail.aspx` داخل الخلية المحددة.

## استخدام قوائم ربط البيانات

في الأمثلة السابقة قمنا بملء القوائم المختلفة عن طريق إضافة عنصر واحد فقط في الوقت الواحد. يمكنك بالطبع إضافة مجموعة من العناصر مرة واحدة، حيث تدعم معظم أدوات خادم **ASP.NET**، مثل مربع السرد **ListBox**، عملية ربط البيانات **Data Binding** وهو ما يعنى ارتباط البيانات الموجودة بالقائمة بمصدر للبيانات. فبدلاً من أن تقوم بتعريف بيانات القائمة بنفسك، يتم تعيين مصدر يحتوى على هذه البيانات ومن ثم يتم ملء القائمة تلقائياً من خلال هذا المصدر.

سنشرح فيما بعد بالتفصيل كيفية استخدام تقنية ربط البيانات مع **ADO.NET** كما سنشرح كيفية استخدام أداة شبكة البيانات **DataGrid** والتي تتيح بدورها للمستخدم تعديل العناصر الموجودة بالقائمة وإرسالها مرة أخرى إلى مصدر البيانات المصاحب ومن ثم يمكنك استخدام هذه الشبكة ليس فقط لاختيار البيانات وإنما لإدخال البيانات الجديدة أيضاً. وحتى نأتى إلى هذا الجزء، سنقوم فيما يلي بتوضيح مفهوم ربط البيانات من خلال إحدى أدوات خادم **ASP.NET** وهى أداة المكرر **Repeater**.

تستخدم أداة التحكم **Repeater** إذا أردت التحكم التام في القائمة التي تظهر داخل مستعرض المستخدم، حيث يمكنك من خلال هذه الأداة إنشاء قائمة **HTML** مخصصة من خلال استخدام القوالب **Templates** حيث يتم تعريف قالب لتحديد كيفية ظهور كل عنصر من عناصر القائمة ثم تتولى الأداة **Repeater** بعد ذلك ربط القالب مع كل عنصر من عناصر القائمة. وأسهل الطرق لتعريف القالب هي كتابته في عرض **HTML** داخل مصمم النموذج. لأداء ذلك، تابع معنا الخطوات الآتية:

١. قم بإنشاء تطبيق ويب جديد أو قم بإضافة نموذج جديد إلى التطبيق الحالي مع جعل هذا النموذج هو صفحة البدء.

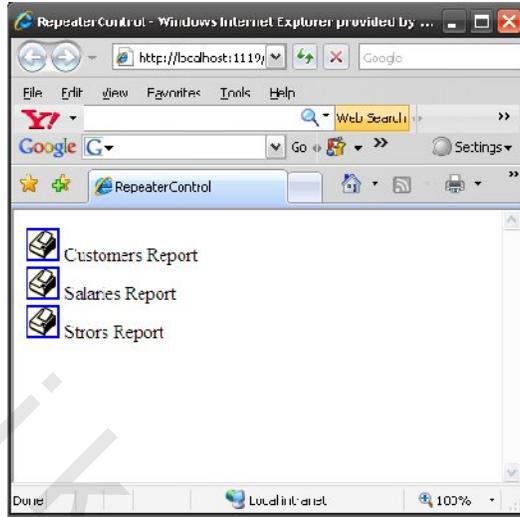
٢. قم بإدراج الأداة **Repeater** من الجزء **Data** بمربع الأدوات إلى النموذج.
٣. انقر زر **HTML** للانتقال إلى طريقة عرض **HTML**، تلاحظ إضافة السطر التالي:
- ```
<asp:Repeater id="Repeater1" runat="server"></asp:Repeater>
```
٤. لإنشاء قالب عناصر القائمة الموجودة بالأداة، قم بإدخال كود القالب داخل العنصر **<ItemTemplate>** كما يلي:

```
<asp:Repeater id="Repeater1" runat="server">
  <ItemTemplate >
    <a href="displayreport.aspx?id=%#
      Container.DataItem.ReportID %">>
    </a>
    <%# Container.DataItem.ReportName %> Report <br>
  </ItemTemplate>
</asp:Repeater>
```

٥. قم بإضافة الكود التالي إلى النموذج:

```
Private Sub Page_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
  Dim ReportList As New ArrayList()
  ReportList.Add(New ReportInfo("Customers", 1))
  ReportList.Add(New ReportInfo("Salaries", 2))
  ReportList.Add(New ReportInfo("Stors", 3))
  Repeater1.DataSource = ReportList
  Repeater1.DataBind()
End Sub
Private Class ReportInfo
  Public ReportName As String
  Public ReportID As Integer
  Public Sub New(ByVal Name As String, ByVal ID As Integer)
    Me.ReportName = Name
    Me.ReportID = ID
  End Sub
End Class
```

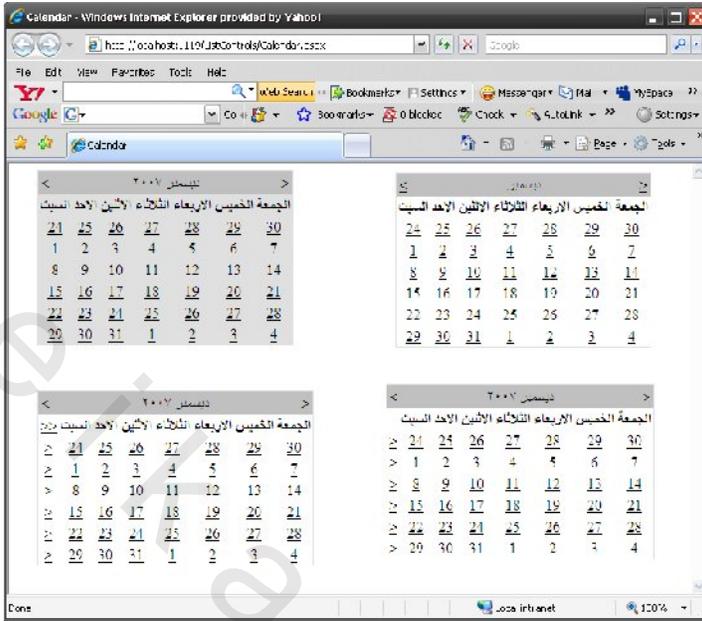
يوضح شكل ٥-٧ التالي كيف يتم دمج القالب مع عناصر القائمة داخل مستعرض الويب.



شكل ٧-٥ استخدام الأداة Repeater

## استخدام أداة التقويم

تعتبر أداة التقويم الشهري **Calendar** إحدى أدوات خادم **ASP.NET** المفيدة، وهي كمشيلتها في **Windows** تستخدم لعرض التقويم على الشاشة كي يتمكن المستخدم من اختيار التاريخ بسهولة منقطعة النظير. فمن الممكن أن تحتوى صفحة الويب على حقل تاريخ داخل مربع نص وأداة تقويم شهري بحيث يتمكن المستخدم من كتابة التاريخ أو استعراضه من خلال التقويم. يوضح شكل ٨-٥ التالي بعض الأمثلة على أداة التقويم الشهري.



شكل ٥-٨ بعض الأمثلة على أداة التقويم

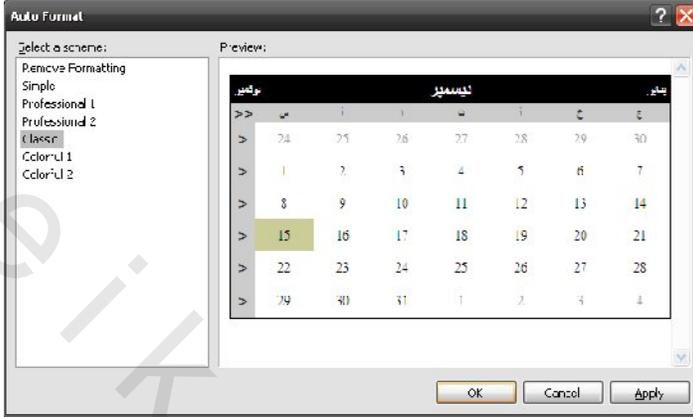
### تجميل أداة التقويم

يوجد العديد من الخصائص المستخدمة في تجميل مظهر أداة التقويم كي تظهر بالشكل الذي يروق لك. فهناك الخاصية **ShowGridLines** على سبيل المثال والتي تستخدم في إظهار الخطوط الرأسية والأفقية بين الأيام الموجودة داخل التقويم، هذا بالإضافة إلى العديد من خصائص الأنماط التي تتحكم في ظهور التاريخ حيث يحتوى النمط على خصائص اللون والخط والإطار والارتفاع والعرض وكذلك إمكانية استخدام صفحة أنماط متتابعة **CSS**. يمكنك تعريف خصائص كل نمط على حده حيث تحتوى أداة التقويم على خصائص الأنماط الآتية:

- النمط **DayHeaderStyle** ويستخدم في تعيين مظهر رؤوس أيام الأسبوع داخل التقويم. كما يمكنك اختيار تنسيق اليوم من خلال الخاصية **DayNameFormat** بحيث يظهر أول حرف أو أول حرفين أو أول ثلاثة حروف أو اسم اليوم بالكامل.

- النمط **DayStyle** ويستخدم لتعيين نمط الأيام في الشهر الحالي.
  - النمط **NextPrevStyle** ويقوم بتعيين نمط أزرار التنقل التي تتيح للمستخدم اختيار الشهر السابق أو الشهر التالي. يمكنك إخفاء هذه الأزرار بتخصيص القيمة **False** للخاصية **ShowNextPrevMonth** وفي هذه الحالة يظل المستخدم قادراً على التنقل من شهرٍ لآخر بنقر أى يوم في الشهر السابق أو الشهر التالي.
  - النمط **OtherMonthDayStyle** ويقوم بتعيين نمط الأيام الموجودة بالشهور المجاورة. لإخفاء هذه الأيام، قم بتخصيص نفس لون الخلفية للخاصية **ForColor**.
  - النمط **SelectedDayStyle** ويقوم بتعيين نمط التاريخ أو اليوم المختار بواسطة المستخدم.
  - النمط **SelectorStyle** ويقوم بتعيين نمط اختيار أكثر من يوم بالتقويم.
  - النمط **TitleStyle** ويقوم بتعيين نمط العنوان الموجود أعلى التقويم. يمكنك إخفاء عنوان التقويم بتخصيص القيمة **False** للخاصية **ShowTitle** كما يمكنك التحكم في النص الذي يظهر بالعنوان من خلال الخاصية **TitleFormat**.
  - النمط **TodayDayStyle** ويقوم نمط اليوم الحالي تبعاً لإعدادات حاسبك.
  - النمط **WeekendDayStyle** ويقوم بتعيين نمط أيام العطلات بحيث تستطيع التفرقة بينها وبين أيام العمل. يمكنك أيضاً تعيين اليوم الأول في الأسبوع من خلال الخاصية **FirstDayofWeek**.
- يمكنك من خلال الخصائص السابقة التحكم في مظهر أداة التقويم. فإذا أردت أن تظهر الأداة بمظهر أفضل من مظهرها الافتراضى دون أن تقوم بضبط كل هذه الإعدادات من خلال اختيار أحد التنسيقات الجاهزة، انقر أداة التقويم بزر الفأرة الأيمن ثم اختر **Autoformat** من القائمة الموضوعية أو انقر الارتباط **Auto Format** مباشرةً من أسفل نافذة الخصائص، يظهر المربع الحوارى **Calendar Auto Format** الذى يمكنك من

خلاله اختيار أحد التنسيقات الجاهزة مثل **Classic** أو **Simple** أو **Professional** (انظر شكل ٥-٩).



شكل ٥-٩ اختيار أحد التنسيقات الجاهزة

تظهر معظم عناصر الاختيار والانتقال داخل أداة التقويم في صورة علامة أكبر من (<) وأقل من (>). ولأن HTML كما تعلم يستخدم هذه العلامات في تحديد حدود رموز HTML، لذا يتم تحويلها إلى الحروف الخاصة &lt; و &gt; لتمثيل أكبر من وأصغر من على الترتيب. يمكنك التحكم في هذه العناصر من خلال إدخال النص الخاص بك.



### اختيار تاريخ من التقويم

يمكنك استخدام الخاصية **SelectedDate** المصاحبة لأداة التقويم لتعيين أو استرجاع التاريخ المختار حالياً داخل التقويم، حيث يمكنك تعيين هذه الخاصية من خلال متغير من النوع **DateTime** أو من خلال تاريخ ثابت كما في الكود التالي:

استرجاع التاريخ المختار'

```
Dim dtSelDate AS DateTime  
dtSelDate = calendar1.SelectedDate
```

تعيين التاريخ المختار'

`calendar1.SelectedDate = #8/12/2008#`

إظهار التقويم على شهر أغسطس سنة ١٩٧٦

`calendar1.TodaysDate = #8/1/1976#`

لاحظ أننا استخدمنا في السطر الأخير الخاصية `TodaysDate`، وهذه إحدى الطرق التي تختلف فيها هذه الأداة على مثلتها في `Windows` حيث لا يتم تلقائياً تحديث التقويم لإظهار الشهر المحدد داخل الخاصية `.SelectedDate`.

حينما تقوم باختيار تاريخ من داخل التقويم، يتم تنفيذ الحدث `SelectionChange` على الخادم ومن ثم يمكنك استخدام هذا الحدث لتحديث مربع نص أو أي أداة أخرى مرتبطة بالتقويم.



### اختيار أكثر من تاريخ

يعتبر اختيار أكثر من تاريخ في نفس الوقت أحد السمات الهامة لأداة تقويم الويب. فعلى الرغم من تقييد هذه الوظيفة مقارنةً بمثلتها في `Windows`، يستطيع المستخدم اختيار أسبوع أو شهر كامل من خلال نقر بعض علامات الاختيار الخاصة. ولكن للأسف لا يستطيع المستخدم اختيار أيام متفرقة أو أكثر من أسبوع من خلال واجهة الأداة. يمكنك تحديد طريقة اختيار أداة التقويم لأكثر من تاريخ من خلال الخاصية `SelectionMode` عن طريق استخدام أي من القيم التالية:

- القيمة `None` والتي لا تمكن المستخدم من إجراء أي اختيارات داخل التقويم وبالتالي لن يحتوي التقويم على أي ارتباطات ولن يتم تنفيذ أي أحداث.
- القيمة `Day` والتي تمكن المستخدم من اختيار تاريخ واحد فقط. وهي القيمة الافتراضية لهذه الخاصية.
- القيمة `DayWeek` والتي تتسبب في إضافة ارتباطات اختيار داخل التقويم لاختيار أسبوع بالكامل. ولا يزال المستخدم قادراً على اختيار الأيام المفردة داخل الأسبوع.
- القيمة `DayWeekMonth` والتي تتسبب في إضافة ارتباطات اختيار داخل التقويم

لاختيار الأسابيع والشهر بالكامل. ولا يزال المستخدم قادراً على اختيار الأيام المفردة داخل الأسبوع.

على الرغم من أن عملية الاختيار من خلال واجهة أداة التقويم مقيدة بعض الشيء إلا أنك تستطيع اختيار أى مجموعة من التواريخ عن طريق تخصيص قيمها للخاصية **SelectedDates** من خلال الكود.



عند العمل مع الاختيارات المتعددة للتواريخ، يتم استخدام الخاصية **SelectedDates** بدلاً من الخاصية **SelectDate** حيث تعتبر الأولى تجمع من النوع **SelectedDatesCollection** الذى يحتوى على قيمة أو أكثر من النوع **SelectedDatesCollection** كما فى الكود التالى:

```
Dim DesiredDates As SelectedDatesCollection
Dim i As Integer
DesiredDates = calendar1.SelectedDates
For i = 0 To DesiredDates.Count -1
    call Func(DesiredDates(i))
Next
```

