

## الفصل الخامس إنهاء أول تطبيق ويب

من الأمور الجيدة في Visual Basic 2008 أنك تستطيع إنشاء تطبيقات ويب يتم تشغيلها من خلال مستعرض الويب وذلك بإنشاء صفحات ASP.NET. سنقوم في هذا الفصل بإلقاء نظرة بسيطة على كيفية إنشاء مثل هذه التطبيقات على أن نتناولها بالتفصيل فيما بعد الجزء الثالث من هذه السلسلة.

بانتها هذا الفصل ستتعرف على:

- ◆ مفهوم تطبيقات الويب
- ◆ الاتصال بخادم الويب
- ◆ إنشاء تطبيق ويب بسيط

من النادر في هذه الأيام أن ترى أحد الأشخاص المهتمين بعالم الحاسب غير متقن لاستخدام الشبكة العنكبوتية WWW أو كيفية استخدام الإنترنت. ومن ثم فإن القدرة على تطوير تطبيقات ومواقع الويب أصبحت إحدى المهارات الأساسية التي لا يستغنى عنها أى مبرمج. سنقوم في هذا الفصل بإعطائك فكرة مبسطة عن كيفية تطوير تطبيقات الويب من خلال Visual Basic 2008 على أن نتعرض لهذا الموضوع بالتفصيل في الجزء الثالث من كتاب المرجع الأساسى لمستخدمى Visual Basic 2008، فكن معنا.

### مفهوم تطبيقات الويب

حينما تطلب من مستعرض الويب المثبت على حاسبك عرض صفحة ويب معينة، فإنك تحصل بعد ثواني معدودة على صفحة الويب معروضة على شاشتك. ولكن الأمر ليس بهذه السهولة، فهناك العديد من العمليات التي تتم من وراء الكواليس كى تظهر لك الصفحة بالصورة التي تراها. فإذا قمت مثلاً بالنقر على زر الشراء Buy داخل موقع الشراء المفضل لديك على الويب، يتم فى الواقع استدعاء برنامج للتأكد من صحة رقم بطاقة الائتمان الذى قمت بإدخاله وكذلك البيانات الأخرى الخاصة بك حتى لا يتمكن الآخرون من استخدام هذه البيانات فى شراء المزيد من المنتجات. ولعلك تتساءل الآن عن الفرق بين تطبيق الويب والتطبيق النوافذى المعتاد، وأجيبك بأن الاختلاف الرئيسى بينهما يكمن فى أن معظم كود تطبيق الويب يكون على خادم الويب نفسه بدلاً من الحاسب الشخصى فى حالة التطبيق النوافذى.

يمكنك استخدام لغة Visual Basic 2008 بأكثر من طريقة لإنشاء تطبيقات الويب وذلك كما يلي:

- يمكنك إنشاء صفحات الخادم النشطة (ASP) Active Server Pages التي تحتوى على كود Visual Basic يتم تنفيذه على الخادم.

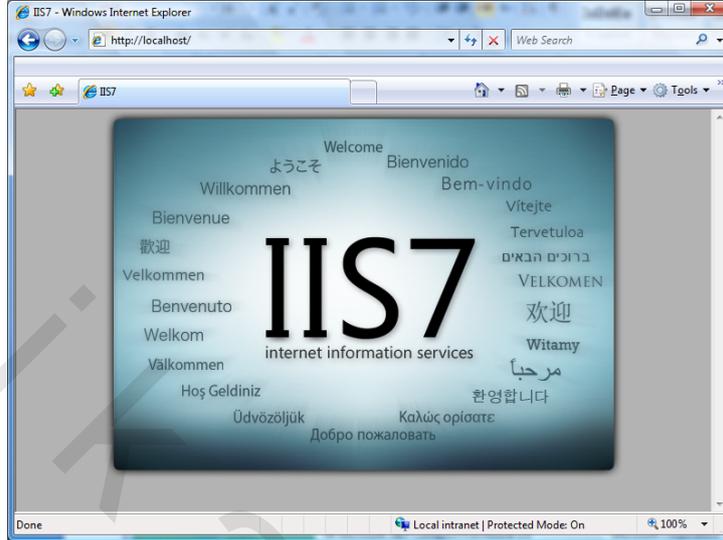
- يمكنك إنشاء التصنيفات التي يتم ترجمتها إلى ملفات DLL تخزن على الخادم ومن ثمّ يتم استدعاء هذه التصنيفات من داخل صفحات الخادم النشطة ASP أو عرضها كخدمة متاحة للجميع من خلال الإنترنت.
  - يمكنك استخدام VBScript وهي إحدى اللغات المتفرعة من اللغة الأم Visual Basic في تنفيذ الكود داخل مستعرض الويب الخاص بالمستخدم.
- يمكنك حقيقةً استخدام أي من الطرق الثلاثة السابقة بمعزلٍ عن Visual Basic 2008 في إنشاء تطبيق ويب إلا أن Visual Basic 2008 يقوم باستخدامها جميعاً في إنشاء مثل هذه التطبيقات.

### الاتصال بخادم الويب

كما ذكرنا منذ قليل، يتم تخزين محتويات تطبيق الويب على خادم الويب نفسه، فإذا أردت تطوير تطبيقات الويب باستخدام لغة Visual Basic 2008، يجب أن تقوم بتعريف خادم الويب الموجود بالشبكة التي تتصل بها. أو بدلاً من ذلك يمكنك في مرحلة التطوير والاختبار تعريف خادم ويب على حاسبك المستخدم لتطوير مثل هذه التطبيقات. قبل أن تبدأ في تطوير تطبيقات الويب، تأكد أولاً من تنشيط خادم الويب وتأكد أنك تعرف العنوان URL الخاص بهذا الخادم. وأبسط الطرق للتعرف على مدى وجود خادم الويب من عدمه هو محاولة الاتصال به من خلال مستعرض الويب المثبت على حاسبك. ابدأ تشغيل مستعرض الويب ثم اكتب العنوان التالي للخادم:

**http://servername**

حيث **servername** هو اسم الخادم والذي غالباً ما يكون **localhost** في حالة استخدام الخادم الوهمي المثبت على حاسبك الشخصي، وحينئذٍ تحصل على أحد الصفحات الموجودة على الخادم (انظر شكل ٥-١).



شكل ١-٥ صفحة ASP موجودة على خادم الويب المثبت بجاسيك.

### إنشاء تطبيق ويب بسيط

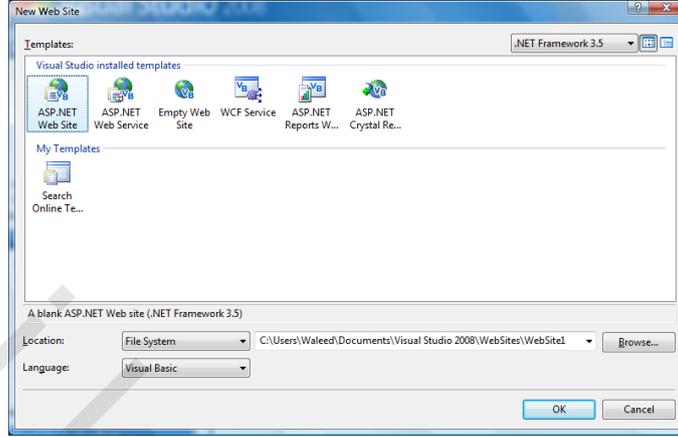
للتعرف على طريقة إنشاء تطبيق ويب من داخل بيئة تحرير Visual Studio 2008، سنقوم فيما يلي بإنشاء صفحة ويب بسيطة مشابهة للنموذج النوافذ الذي أنشأناه في الفصل السابق. لأداء ذلك، تابع معنا الخطوات الآتية:

١. من بيئة تطوير Visual Studio 2008، قم بأداء أي مما يلي:

- افتح قائمة File من شريط القوائم ثم اختر New Web Site من القائمة المنسدلة.

• من صفحة البدء Start Page، انقر الارتباط Web Site بالصف Create.

وفي الحالتين يظهر المربع الحوارى New Project (انظر شكل ١-٥-٢).



شكل ٥-٢ المربع الحوارى New Web Site.

٢. يحتوى الجزء **Templates** بالمربع الحوارى على عدد من الرموز التى تمثل برامج

وتطبيقات الويب المختلفة التى يمكنك إنشاءها من خلال **Visual Basic 2008**.

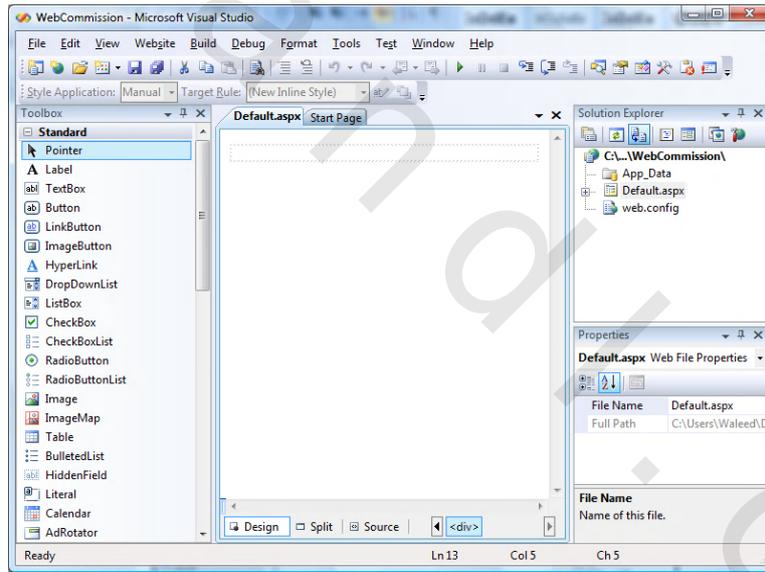
يوضح جدول ٥-١ التالى الرموز الموجودة بالمربع الحوارى واستخدام كل منها.

جدول ٥-١ تطبيقات الويب التى يمكنك إنشاءها من خلال **Visual Basic 2008**

الرمز	الاستخدام
	إنشاء تطبيق ويب يعمل على خادم الويب. حيث يمكنك تطوير تطبيقات الويب بنفس طريقة تطوير تطبيقات <b>Windows</b> إلا أن الخرج يكون عبارة عن موقع ويب يعمل من خلال الويب وليس تطبيق تنفيذى يعمل من خلال <b>Windows</b>
	إنشاء موقع ويب على خادم الويب يقوم بتوفير خدمة معينة أو أكثر لبقية التطبيقات الموجودة على الويب
	إنشاء موقع ويب وظيفته إنشاء خدمات <b>WCF</b>
	إنشاء موقع ويب خالى بحيث تقوم بإنشاء جميع عناصره بنفسك
	إنشاء موقع ويب يحتوى على تقرير بللورى

إنشاء موقع ويب يحتوي على تقرير عادي	 ASP.NET Reports W...
-------------------------------------	---

٣. اختر الرمز **ASP.NET Web Site** من المربع **Templates** الموجود بالجزء العلوي من المربع الحوارى.
٤. على فرض أننا نرغب فى تخزين الموقع الجديد على الحاسب، تأكد من اختيار **File System** من القائمة المنسدلة **Location** ثم انتقل إلى نهاية العنوان المقترح وقم بتعيين اسم مناسب لجلد الموقع وليكن **WebCommission** بدلاً من الاسم الافتراضى للمجلد.
٥. انقر زر **Ok**، يتم إنشاء موقع الويب على موقع الويب الافتراضى، وتظهر صفحة ويب باسم **Default.aspx** لا تحتوى على أية بيانات (انظر شكل ٥-٣).



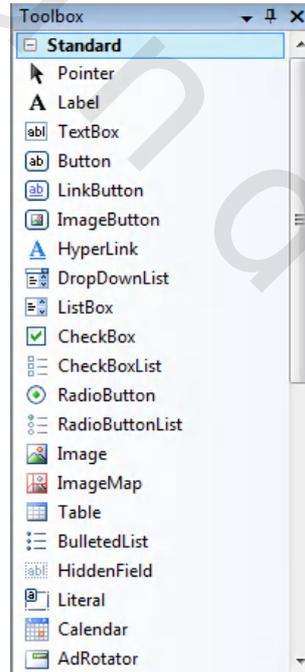
شكل ٥-٣ تطبيق الويب داخل بيئة التطوير.

والآن حان وقت تصميم تطبيق الويب الذى بين أيدينا فى هذا الفصل وهو مشابه إلى حد كبير لتطبيق حساب العمولة الذى أنشأناه فى الفصل السابق غير أنه يعمل من خلال الويب على عكس الأول الذى يعمل تحت نظام التشغيل **Windows**. سنقوم كما فعلنا فى

الفصل السابق بتصميم واجهة المستخدم بما فيها من عناصر تحكم وخصائصها المختلفة وبعد ذلك نقوم بكتابة كود **Visual Basic** اللازم لحساب العمولة وإظهار النتائج وأخيراً نقوم بتشغيل التطبيق من خلال الويب، فيلى التفاصيل.

### تصميم نموذج الويب

كما علمنا في الفصل السابق، فإن **Visual Basic 2008** يحتوى على العديد من أدوات التحكم الجاهزة التى يمكنك استخدامها فى تطبيقاتك المختلفة. والخطوة الأولى فى إضافة أدوات التحكم إلى النموذج هى إظهار مربع الأدوات. لأداء ذلك، افتح قائمة **View** من شريط القوائم ثم اختر **Toolbox** من القائمة المنسدلة أو انقر زر  بشريط الأدوات بالجزء العلوى من النافذة، وفى جميع الأحوال تظهر نافذة مربع الأدوات (انظر شكل ٥-٤).



شكل ٥-٤ مربع الأدوات Toolbox داخل بيئة التطوير **Visual Studio 2008** فى حالة تصميم مواقع الويب.

سنحتاج في تطبيق حساب العمولة الذى بين أيدينا إلى ثلاثة عناصر من عناصر التحكم الموجودة بمربع الأدوات السابق وهى:

- مربع النص **TextBox** الذى يتيح للمستخدم إدخال النصوص من لوحة المفاتيح مثل بيانات المرتب الأساسى وإجمالي المبيعات ونسبة العمولة. وسوف نحتاج إلى إضافة أربعة مربعات إلى التطبيق.
- أداة العنوان (التسمية) **Label** التى تستخدم لعرض العناوين وسوف نحتاج فى تطبيقنا أيضاً إلى أربعة عناوين.
- زر الارتباط **LinkButton** الذى يمكنك من خلاله تنفيذ حدثٍ ما ويشبه إلى حد كبير زر الأمر **Command Button** المستخدم فى التطبيقات النوافذية. وسوف نستخدم فى هذا التطبيق زر ارتباط واحد فقط يقوم المستخدم من خلاله بإجراء الحسابات الخاصة بصافى المرتب.

#### إضافة مربع النص إلى نموذج النافذة

لكي تضيف أداة تحكم إلى تطبيقك، لابد أولاً أن تضيفها إلى نموذج النافذة. أول أداة سنضيفها إلى تطبيق حساب العمولة هي مربع النص. لإتمام ذلك، تابع معنا الخطوات الآتية:

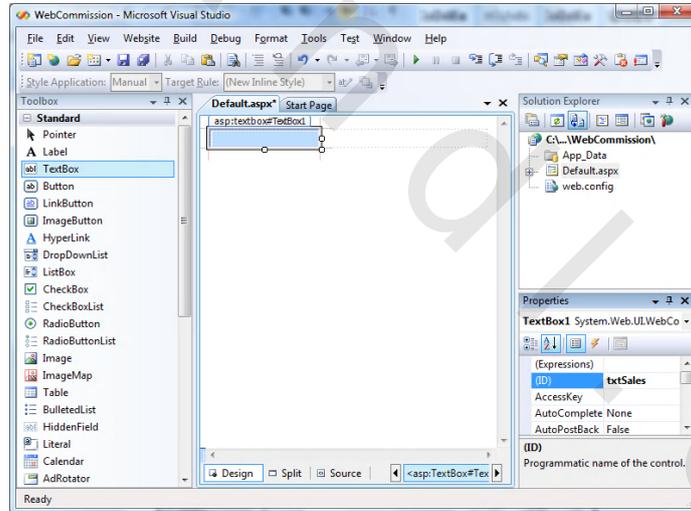
١. انقر أداة مربع النص **Label TextBox** من مربع الأدوات نقرأ مزدوجاً لإضافتها إلى النموذج، حيث يتم فى هذه الحالة إضافة مربع النص إلى الركن الأيسر العلوى من النموذج.
٢. استخدم أزرار التحجيم الموجودة حول الأداة فى ضبط طول الأداة وعرضها إن أحببت.

#### ضبط خصائص مربع النص

بعد إضافة مربع النص إلى نموذجك، تحتاج إلى ضبط بعض خصائصه. تذكر أن الخصائص تتحكم فى مظهر عنصر التحكم أو فى مظهر الكائن بوجه أعم وفى سلوكه. بالنسبة لمربع

النص السابق نحتاج إلى ضبط خاصيتين فقط من خصائصه وهما ID و Text . يقوم Visual Basic بإعطاء عنصر التحكم اسماً مميزاً، فمثلاً لو وضعت مربع نص؟ سيطلق عليه TextBox1 وإذا أضفت مربع نص آخر سيسميه TextBox2 وهكذا، ويمكنك استخدام هذه الأسماء الافتراضية ولكنها عادة سيئة في البرمجة، حيث أن الاسم المعبّر سهل الحفظ والمتابعة عند حدوث الأخطاء، فمثلاً لو أن لديك ثلاثة مربعات نصوص لإدخال First name و Last name و Address في أحد البرامج فإن أسماء مثل txtFname و txtLname و txtAddress أفضل من TextBox1 و TextBox2 و TextBox3.

لتغيير اسم مربع النص في حالة نماذج الويب، اذهب إلى خاصية ID في نافذة الخصائص وفي المساحة المقابلة لها اكتب txtSales وهو المربع الذي سيحتوي المبيعات (انظر شكل ٥-٥).



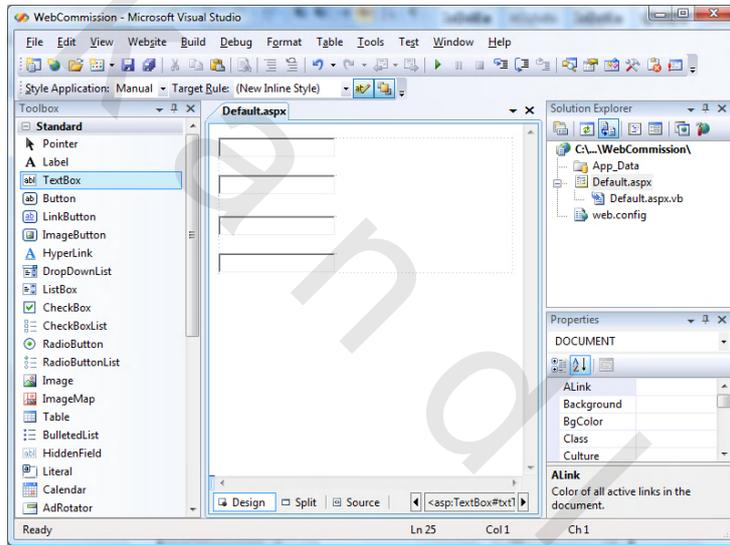
شكل ٥-٥ إضافة مربع النص إلى النموذج وتغيير اسمه من خلال مربع الخصائص. ومن الملاحظ أن الخاصية Text التي تحتوي على نص مربع النص تكون افتراضياً خالية تماماً كما في مربع النص في حالة النماذج النوافذية عند إضافة مربع النص إلى النموذج، لذا فلننا في حاجة إلى تغيير هذه الخاصية.

تذكر أننا ما نزال حتى الآن في طور التصميم وما أن نبدأ تشغيل التطبيق لن نستطيع تغيير هذه الخصائص إلا من خلال الكود.



### إضافة مربعات النصوص الباقية

باستخدام الخطوات السابقة أضف إلى نموذج النافذة ثلاثة مربعات نصوص أخرى بأسماء **txtPercent** و **txtSalary** و **txtTotal**. انقر إلى اليمين من مربع النص الأول ثم اضغط مفتاح الإدخال مرتين للانتقال إلى سطر جديد في كل مرة وقم بإضافة مربع النص التالي. بعد قيامك بالخطوات السابقة سيبدو النموذج كما ترى في شكل ٥-٦.



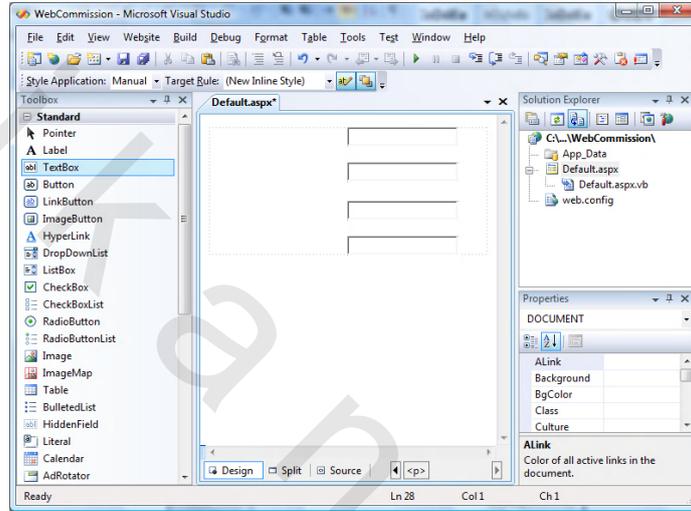
شكل ٥-٦ نموذج النافذة بعد إضافة مربعات النصوص الأربعة.

### تحريك وتغيير حجم مربعات النصوص

لتحريك أي من مربعات النصوص السابقة قم بالنقر عليها ثم اسحبها (مستمرًا في ضغط زر الفأرة) إلى المكان الذي تريده أو استخدم مسطرة المسافات في إزاحة مربع النص إلى المكان المطلوب.

تغيير حجم مربع النصوص مماثل لتغيير حجم أي نافذة في **Windows** وذلك بنقره أولاً ثم سحب أحد المقابض الثلاثة (المربعات الصغيرة) الموجودة على أضلاع وأركان مربع النص.

الآن، وبعد أن رسمنا مربعات النصوص ينبغي أن نضع عنوان لكل مربع حتى يدرك المستخدم المحتويات المطلوبة في كل منهم، ولنفسح مكانا للعناوين فلنقم بتحريك المربعات إلى اليمين قليلا، استخدم مسطرة المسافات في إزاحة كل مربع نص على حده حتى تحصل على النموذج قريبا من شكل ٧-٥ التالي.



شكل ٧-٥ تحريك مجموعة من عناصر التحكم مرة واحدة بعد تحديدها.

### إضافة العناوين

كما أشرنا سابقا، فإننا بحاجة إلى إشارة ظاهرة للمستخدم تخبره بنوعية المعلومات المرادة في كل مربع نصي (لا تنس أن الأسماء المعطاة سابقا للمربعات تستخدم فقط في كود التطبيق ولا تظهر على الشاشة ولا يعلم عنها المستخدم شيئا)، أسهل الطرق أن نضع أداة عنوان **Label** ملاصقة للمربع النصي بها الكلمات المراد الإشارة بها إلى المحتوى.

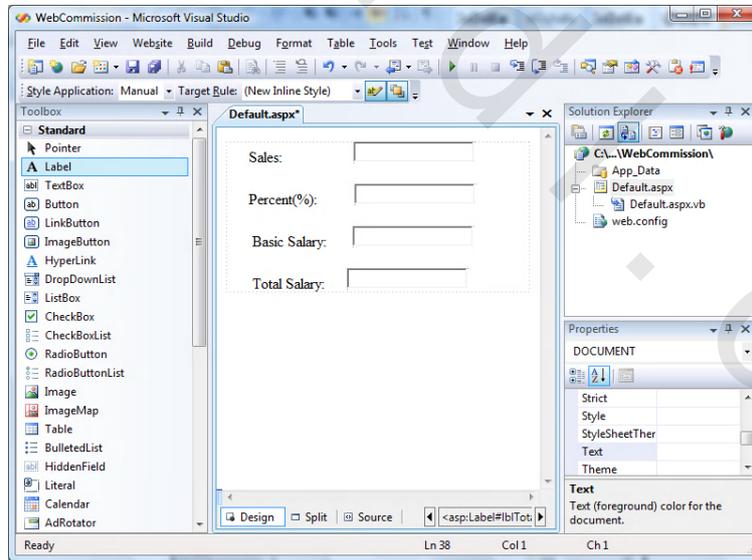
الفرق الأساسي بين مربع النص والعنوان أن العنوان لا يستطيع المستخدم تعديله أثناء تشغيل التطبيق كما يفعل مع محتويات مربع النص.

لكي تضيف أحد العناوين إلى نموذجك، تابع معنا الخطوات الآتية:

١. انقر في المكان الذي ترغب في إضافة أداة العنوان إليه داخل النموذج.

٢. انقر أداة العنوان **A Label** داخل مربع الأدوات نقرأ مزدوجاً، وحينئذٍ يتم إضافة الأداة في المكان المحدد داخل النموذج.
  ٣. غير اسم العنوان ( أي غير خاصية ID ) إلى **lblSales**.
  ٤. غير التعليق الخاص بالعنوان (خاصية Text) من **Label** (القيمة الافتراضية) إلى **"Sales:"**، لا تنس النقطتان في نهاية الكلمة.
- باتباع نفس الخطوات يمكننا أن نضيف العناوين الثلاثة الباقية بجوار مربعات النصوص المناظرة مع اختلاف الاسم والتعليق الخاص بكل منها. يحتوى الجدول ٥-٢ على خصائص العناوين الأربع التي سننشئها. بعد الانتهاء ستبدو النافذة كما ترى في شكل ٥-٨.
- جدول ٥-٢ خصائص العناوين التي توضح مدخلات التطبيق

الاسم ID	العنوان Text
<b>lblSales</b>	<b>Sales :</b>
<b>lblPercent</b>	<b>Percent(%) :</b>
<b>lblBasic</b>	<b>Basic salary :</b>
<b>lblTotal</b>	<b>Total Salary :</b>

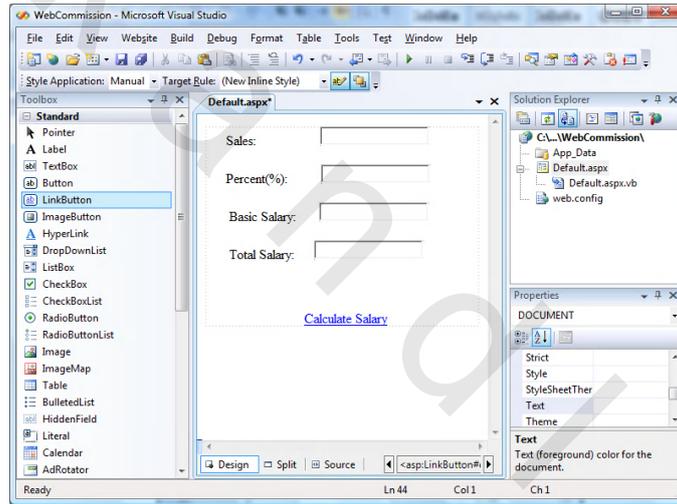


شكل ٥-٨ نموذج النافذة بعد إضافة العناوين لتعريف مربعات النصوص.

### إضافة زر الارتباط *LinkButton*

آخر ما نحتاج من العناصر هو زر الارتباط الذي يتشابه في طريقة عمله كثيراً مع زر الأمر المستخدم في النماذج النوافذية، وهو مسئول عن بدء عملية ما.

اضغط مفتاح الإدخال بعد آخر مربع نص عدة مرات ثم انقر الأداة *LinkButton* بمربع الأدوات نقراً مزدوجاً لرسم الزر على النافذة وامنحه اسماً مميزاً وليكن *cmdCalculate* كما فعلت بالعناصر السابقة. أيضاً للزر خاصية *Text* مثل العنوان وتحتوي الكلمات التي ستظهر على سطح الزر وغالباً ما تعبر عما يقوم به التطبيق عند الضغط على الزر، ضع التعليق *Calculate Salary* في خاصية *Text* للزر (انظر شكل ٩-٥).



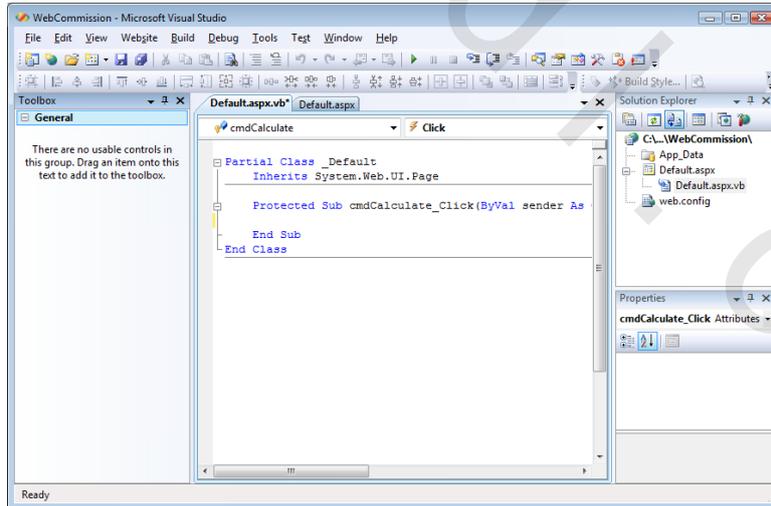
شكل ٩-٥ نموذج النافذة بعد إضافة زر الارتباط.

من الأفضل الآن في هذه المرحلة أن تقوم بحفظ عملك تحسباً لانقطاع التيار أو حدوث خطأ ما. لأداء ذلك، افتح قائمة *File* من شريط القوائم ثم اختر *Save All* من القائمة المنسدلة أو اضغط الزر  مباشرةً من شريط الأدوات.

## كتابة الكود

عند هذه النقطة اكتملت واجهة التطبيق، إلا أنك لو شغلته لن يقوم بأي وظيفة، صحيح أنك تستطيع كتابة ما تريد في مربعات النصوص من أرقام ولكن لن يقوم التطبيق بأية حسابات لإيجاد النواتج المرغوبة وهنا تكمن وظيفة كود التطبيق. يتيح لك **Visual Basic** كتابة إجراء ما **Procedure** يتم تنفيذه عند النقر على زر معين أو أي حدث لأي كائن آخر. يسمى هذا النوع من الإجراءات بالإجراءات الحدثية **Event Procedures**.

في **Visual Basic** يقوم التطبيق بالأفعال عادةً استجابة لحدث ما. والحدث هو ما يطرأ على كائن نتيجةً لفعل المستخدم أو لفعل من قِبَل جزء آخر من التطبيق أو لفعل من قِبَل نظام التشغيل. من أمثلة الأحداث **Events** أن يضغط المستخدم مفتاحاً أو يجرّك الفأرة، أو يغير قيمة في أحد أدوات التحكم، أو مرور فترة زمنية محددة. أسهل الطرق لإدخال الكود أن تنقر نقراً مزدوجاً على أداة التحكم التي تريدها أن تستجيب لأحد الأحداث. ستفتح لك نافذة الكود حيث تقوم باختيار الأداة أوتوماتيكياً من المربع الأيسر كما يبدو من شكل ١٠-٥.



شكل ١٠-٥ فتح نافذة الكود بمجرد النقر المزدوج على زر الارتباط cmdCalculate .

يمكنك أيضا فتح نافذة الكود بضغط مفتاح F7 أو بنقر زر إظهار الكود من نافذة المستكشف Solution Explorer أو بفتح قائمة View واختيار Code من القائمة المنسدلة.



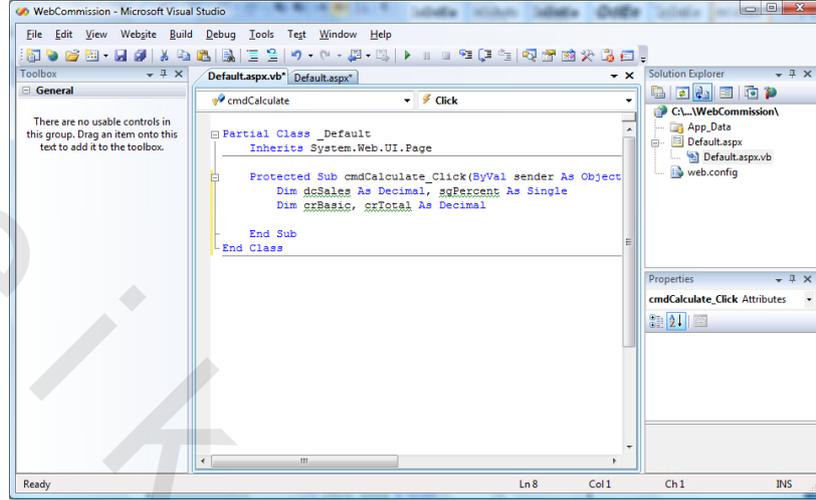
الإجراء الحدتي التالي الذي سنكتبه هو الخاص بإجراء الحسابات و هو الخاص بالنقر على زر Calculate Salary وهو الإجراء cmdCalculate\_Click وتلاحظ أنه مكون من اسم أداة التحكم والحدث المراد الاستجابة له. هذا الكود هو لب التطبيق. وقبل أن نجري حساباتنا علينا أن نجرب Visual Basic عن ماهية المتغيرات التي سنجري عليها الحسابات، إخبار مترجم اللغة بأنماط المتغيرات Variable types تسمى عملية "الإعلان عن المتغيرات Variable declaration" ولا بد أن تسبق كتابة عبارات التطبيق التنفيذية.

#### الإعلان عن المتغيرات

يعني الإعلان عن المتغيرات إخبار Visual Basic بأسماء المتغيرات التي ستستخدمها وأنماط البيانات التي ستحتويها هذه المتغيرات مثل integer, Date, string... وهذه الخطوة هامة للغاية و إجبارية في كافة لغات الكمبيوتر الهامة وسيبين فائدتها فيما يلي. سيتكون مقطع الإعلان من الأسطر التالية:

```
Dim dcSales As Decimal, sgPercent As Single  
Dim crBasic , crTotal As Decimal
```

ومن المفترض أن تبدو على نافذة البرمجة كما في الشكل ٥-١١.



شكل ١١-٥ نافذة الكود عند إضافة مقطع الإعلان عن المتغيرات.

وكما تلاحظ تتكون صيغة الإعلان عن متغير من الكلمة **Dim** يليها اسم اختياري للمتغير يليها **As** ثم نمط المتغير، ويمكن الإعلان عن المتغيرات في أسطر منفصلة أو في نفس السطر كما ترى.

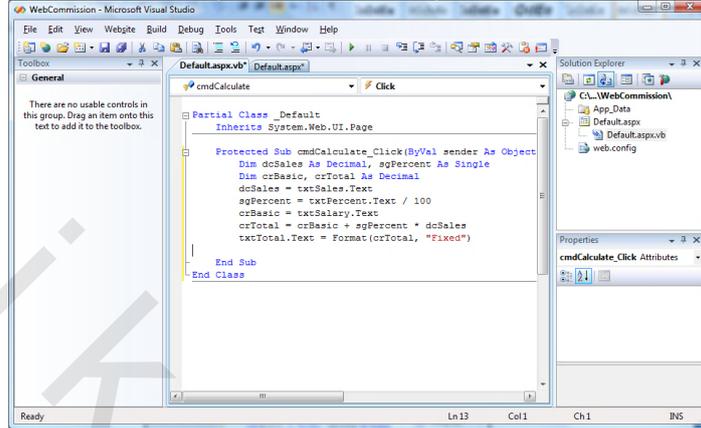
كما في تسمية عناصر التحكم يستحسن أن تتبع الطريقة القياسية في تسمية المتغيرات، هذه الطريقة القياسية تتكون من حرفان كبادئة تعبر عن نمط المتغير ثم اسم المتغير، في مثالنا السابق يتكون **dcSales** من **dc** وهي اختصار لـ **Decimal** ثم **Sales** بما يعني أنه متغير من نمط القيم العشرية (المالية) يحمل قيمة المبيعات.

### كود الإجراء

كود الإجراء هو العبارات التي تقوم بالحسابات الفعلية، سوف يشتمل هذا الإجراء على قراءة المدخلات من مربعات النصوص الثلاثة الأولى ثم حساب المرتب الإجمالي بعد إضافة العمولة وعرضه في المربع المحدد. انظر نص الإجراء فيما يلي وشكله عند كتابته في نافذة الكود في شكل ١٢-٥.

```
dcSales = txtSales.Text
sgPercent = txtPercent.Text / 100
crBasic = txtSalary.Text
```

**crTotal = crBasic + sgPercent \* dcSales**  
**txtTotal.Text = Format(crTotal,"Fixed")**



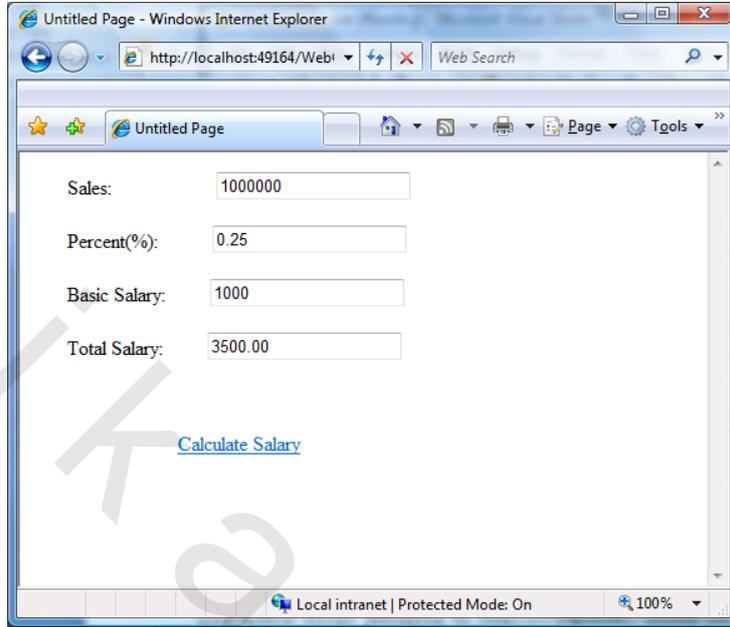
شكل ٥-١٢ شكل نافذة الكود عند كتابة إجراء الحسابات.

### تشغيل التطبيق

لتشغيل التطبيق انقر زر  من شريط الأدوات، أو افتح قائمة **Debug** من شريط القوائم ثم اختر **Start Debugging** من القائمة المنسدلة أو اضغط المفتاح **F5** مباشرة من لوحة المفاتيح، تقوم بيئة التطوير في البداية بترجمة التطبيق لاكتشاف الأخطاء وإنشاء النسخة التنفيذية منه داخل مجلد التطبيق، ومن ثم يتم تشغيل التطبيق حيث يظهر النموذج داخل مستعرض الويب المثبت على حاسبك. للاختبار أدخل القيم الآتية في المربعات:

**Sales**            **1000000**  
**Percent**        **0.25**  
**B. Salary**       **1000**

ثم اضغط الارتباط **Calculate Salary** ينبغي أن تكون قيمة **Total Salary** المعروضة مساوية **٣٥٠٠** (انظر شكل ٥-١٣). بعد اختبار التطبيق بعدة قيم يمكنك العودة لبيئة التطوير بنقر زر إغلاق مستعرض الويب بالركن الأيمن العلوي.



شكل ١٣-٥ نموذج التطبيق داخل مستعرض الويب.

