الفصل السابع:

مدخل إلي Visual Basic .Net

نتناول في هذا الفصل مفاهيم ضرورية يجب أن تعرفها قبل أن تبدأ العمل مع **Visual Basic**.

بالانتهاء من هذا الفصل ستكتسب المعارف وتتدرب على المهارات التى تجعلك قادرا على:

- *فهم تطور* **Visual Basic** *و تقنية* **.NET** ؟
- *فلسفة البرمجة باستخدام* **Visual Basic**
- *البرمجة المسيرة بالأحداث*
- *البرمجة الموجهة بالكائنات* **Object Oriented Programming (OOP)**
- *التعرف على بيئة تطوير* **Visual Studio .Net**
- *إنشاء مشروع جديد وحفظه وتسميته*
- *الفرق بين الحل والمشروع والبرنامج*

لمحة عن تطور Visual Basic

كانت Visual Basic 6 برنامجاً مستقلاً ، أما Visual Basic 2008 فهي عبارة عن لغة في نظام تطوير أكبر، فإذا عدنا إلي الوراء للتعرف علي أصول Visual Basic فقد كانت Basic لغة برمجية يتم استخدامها منذ عشرين عاماً كجزء من MS-Dos. وفي عام ١٩٨٥ تحولت Basic إلي Visual Basic كما أصبحت جزءاً من أداة إنشاء التطبيقات الخاصة بالـ Windows. أما بخصوص برنامج Visual Basic 6 فهو ليس مجرد لغة – بل كان يمكنه إنشاء النماذج الخاص به – علي سبيل المثال – أما بالنسبة لـ Visual Basic 2008 فيملك أداة إنتاج نماذج جديدة توفر له وسيلة جديدة للتفاعل مع نظام التشغيل Windows تلك هي بيئة .NET Framework. التي تحيط بلغة Visual Basic 2008.

عندما تنظر إلى Visual Basic فمن المحتمل أن تفكر فى تطبيقات Windows الخاصة بـ Microsoft. فعلى مدى خمسة عشر عاما استخدم المطورون Ruby Forms

Engine الخاصة ببرنامج Visual Basic. لكتابة تطبيقات العمل الشائعة، وعندما كان يتم تعريف برنامج كبرنامج Visual Basic، فهذا معناه أنه تطبيق Windows بشكل افتراضى.

لكن زاد مجال Visual Basic واستخدامها مع ظهور Visual Basic.NET و Visual Basic.NET هى عبارة عن لغة مثل C++ أو COBOL أو Java والتى يمكنك استخدامها لكتابة أى نوع من التطبيقات مدعم من قبل (Application programming interface) API. والآن عندما يتم تعريف برنامج كبرنامج Visual Basic، فيجب أن تطرح الأسئلة التالية : ما نوع البرنامج ؟ هل هو تطبيق Windows أو موقع ويب؟ هل هو خدمة Windows أو خدمة ويب XML؟

يمكن القول أنه ما زلت هناك حاجة كبيرة لإنشاء تطبيقات Windows – والتى يطلق عليها الآن تطبيقات Windows Forms. فبالرغم من أن تطبيقات الويب تزداد الحاجة إليها يوماً بعد يوم، فإن بيئة العمل الفعالة لأى تطبيق Windows ليست مألوفة فحسب، بل لها أيضا العديد من المزايا التى من الصعب أن تتوفر بالنسبة لأى نوع تطبيق آخر.

في بيئة NET. أصبحت اللغة البرمجية مجرد وسيلة للتفاعل مع بيئة العمل وبالتالي مع نظام تشغيل Windows فجميع البرامج تحتاج الى مجموعة من القواعد المحددة (بالنسبة لاتخاذ القرار وعمليات التكرار وما شابه) داخل البرامج للحصول على التطبيق المطلوب توفر لغة Visual Basic مثل هذه المجموعة من القواعد فى حين توفر بيئة العمل الكائنات والأحداث التي سيتم التعامل معها.

ما هى تقنية NET. ؟

أرادت مايكروسوفت من خلال تقنية NET. توفير بنية تحتية مشتركة لجميع المطورين والمستخدمين على حد سواء تهدف أساساً إلى تحويل نظرة البرمجة والبرمجيات من مفهوم الحاسبات الشخصية المستقلة أو الشبكات الصغيرة إلى مفهوم إتاحة البيانات وعرضها فى جميع أنحاء العالم من خلال شبكة الإنترنت. وقد أطلقت مايكروسوفت فى البداية على هذا النطاق الجديد المصطلح Next Generation Windows Services(NGWS) ثم عادت وأسمته NET..

يحتوى NET Framework. (نطاق NET.) حقيقةً على وفرة هائلة من التصنيفات والبنية التحتية والأدوات التى تمكن المطورين من تطوير تطبيقات عالية الدقة والجودة بسهولة منقطعة النظير مقارنةً بالإصدارات السابقة، حيث يحتوى هذا النطاق على عدد من لغات البرمجة القوية والشهيرة مثل Visual C++ 2008 ولغة Visual C# 2008 إلى جانب لغة Visual Basic 2008 موضوع هذا الكتاب، بالإضافة إلى تضمين ASP.NET وADO.NET كعناصر أساسية داخل النطاق.

وقديماً كنت فى حاجة إلى استخدام وإتقان لغة واحدة من لغات البرمجة إذا أردت تطوير تطبيق من التطبيقات يعمل على الأجهزة الشخصية أو من خلال خادم إحدى الشبكات. أما الآن فالأمر تغير إلى حد ما، حيث يلزمك التعرف على العديد من التقنيات والمهارات مثل صفحات الخادم النشطة Active Server Pages (ASP) ومكونات COM وHTML وXML وVBScript وJScript وغيرها من التقنيات الأخرى إذا أردت تطوير تطبيق متعدد الأطراف n-tier Application. ومن حسن الحظ تضمين جميع هذه التقنيات داخل نطاق NET. (NET Framework.) كما يمكنك داخل NET. Framework (نطاق NET.) اختيار اللغة التى تشعر تجاهها بالراحة كما يمكنك إنشاء وتطوير تطبيقك باستخدام أكثر من لغة، حيث تتيح NET. التداخل بين اللغات.

وإذا كان الموطن الطبيعي للبرامج الآن هو الحاسبات الشخصية، فإن استخدام NET. في تطوير التطبيقات والبرامج يعمل على نقل موطن هذه التطبيقات أو تلك البرامج إلى أعالي البحار ليكون داخل الويب وهو ما يعنى إتاحتها للعالم بأثره. والهدف من هذا الكتاب أن تتعرف على الكيفية التي يتعامل بها Visual Basic 2008 مع بيئة عمل NET. Framework الخاصة بنظام Windows.

فلسفة البرمجة باستخدام Visual Basic

من المعروف أن بيئة Windows بيئة رسومية بينية، وتسمى Graphical User Interface وهى تختلف عن نظام DOS، ولذلك فان البرمجة تحت بيئة Windows تختلف عن البرمجة تحت نظام التشغيل DOS، وهذه نقطة هامة يجب أن تنتبه إليها خصوصا إذا كان قد سبق لك كتابة برامج بلغة بيسك تحت نظام DOS. ولكى يسهل

عليك تصميم برامج لتعمل تحت بيئة Windows بصفة عامة أو باستخدام Visual Basic 2008 بصفة خاصة يجب أن تفهم فكرة البرمجة تحت بيئة Windows.

البرمجة المسيرة بالأحداث

Programming بينما تستخدم البرمجة تحت نظام التشغيل DOS مفهوم تنفيذ التعليمات تسلسليا أى بترتيب ورودها فى البرنامج . وفيما يلى نوضح مفهوم البرمجة المسيرة بالأحداث.

تسمى البرمجة تحت بيئة Windows بـ Event Driven Programming أى البرمجة المسيرة بالأحداث. وتهدف فكرة البرمجة المسيرة بالأحداث إلى إعطاء الآخرين أو البرامج الأخرى الفرصة لكى تشارك برنامجك مصادر الكمبيوتر، وهو ما يتمشى مع فكرة تعدد المهام التى يستخدمها نظام Windows. ونقصد بكلمة الأحداث أى الأحداث التى يقوم بها المستخدم مثل اختيار أمر من قائمة أو نقر زر أمر معين، أو الضغط على لوحة المفاتيح. عندما يقوم المستخدم باختيار أمر أو نقر زر معين ينتقل التنفيذ إلى البرنامج المسئول عن ذلك الأمر أو هذا الزر وبالتالى يقوم البرنامج باتخاذ الإجراء الذى يراه مناسبا لهذا الحدث. وبعد أن ينتهى البرنامج من تنفيذ الإجراء المناسب ينتقل التنفيذ إلى نظام Windows ... وهكذا. فمثلا لتشغيل برنامج معين من نافذة مستكشف Windows، يجب أن تنقر رمز البرنامج نقرا مزدوجا، هذا النقر المزدوج عبارة عن حدث يتولى Windows رصده وإرسال رسالة بهذا المعنى إلى مدير البرامج الذى يستجيب للحدث بأن يشغل البرنامج. وهكذا تلاحظ أن البرامج لا تقوم بنفسها بالبحث عن الأوامر أو عن حركات الفأرة وإنما تنتظر حتى يخبرها Windows بذلك.

وعليك أن تضع فى اعتبارك هذا المفهوم حين تصمم برامجك، فيجب أن تعرض على المستخدم خيارات مختلفة وأن تترك له حرية التصرف والانتقال من برنامجك إلى برنامج آخر ثم العودة إلى برنامجك مرة أخرى ...وهكذا. بعبارة أخرى صمم برنامجك بحيث يستجيب للأحداث المختلفة التى يقوم بها المستخدم، وليس على أنه سلسلة من التعليمات التى تنفذ واحدة بعد الأخرى-كما هو الحال فى برامج DOS. وهذا يتطلب تجزئة البرنامج إلى عدة أجزاء، بحيث يستجيب كل جزء منه لحدث معين، فمثلا جزء

يستجيب لحدث النقر على زر الفأرة وجزء يستجيب لحدث الضغط على لوحة المفاتيح. ويسهل عليك Visual Basic هذا الأمر عن طريق تقسيم البرنامج إلى إجراءات Procedures يختص كل منها بحدث معين.

البرمجة الموجهة بالكائنات Object Oriented Programming (OOP)

البرمجة الموجهة بالكائنات Object Oriented Programming (OOP) أسلوب تعتمده العديد من لغات البرمجة مثل Smalltalk ، Ada ، Java ، C++ ، C# وVisual Basic، ويهدف هذا الأسلوب لفصل البرنامج إلي أجزاء منفصلة وظيفيا وشكليا تسمى بالكائنات Objects تعمل باستقلال تام وإن أرادت التعاون مع غيرها من الكائنات خاطبتها من خلال ما يسمى بواجهة التخاطب للكائن interface.

أهم ما يميز البرمجة بالكائنات، استخدام كائنات يمكن إعادة استخدامها بحيث تدعم المفاهيم التالية:

- الاحتوائية Encapsulation : لابد أن يحتوي الكائن على المعلومات الموصفة له ، مضافا إليها الأساليب المستخدمة في معالجة هذه المعلومات. تسمى هذه الخاصية أحيانا ب "إخفاء البيانات" Information Hiding.
- الوراثة Inheritance : يمكن إنشاء كائن جديد من كائن موجود ويرث الكائن الجديد كافة خصائص الكائن الأصلي والذي يسمى "والد" Parent.
- تعدد الأشكال Polymorphism : رغم السماح بوجود نفس الوظيفة Method في العديد من الكائنات (حتى بين الكائنات المشتقة من كائنات أخرى)، إلا أن كل كائن يقوم بهذه الوظيفة بشكل مختلف. مثال على ذلك العامل + يمكن استخدامه لكل من الأعداد الحقيقية والصحيحة، رغم اختلاف التمثيل الداخلي لكل منهما إلا أن البرنامج سيقوم بتنفيذ الأسلوب المناسب عند إجراء عملية الجمع الفعلية.

التعرف على بيئة تطوير Visual Studio .Net

بيئة تطوير Visual Studio .Net عبارة عن الأداة التى يمكنك استخدامها فى إنشاء أى نوع من التطبيقات باستخدام إحدى اللغات البرمجية المتوافقة والموجودة بداخل المجموعة. وتحل بيئة تطوير Visual Studio محل برنامج Visual Basic 6.0 والذى كان يعتبر فيما سبق جزءاً من مجموعة برامج Visual Studio التى كان يحمل كل عنصر من عناصرها رقم الإصدار 6.0). وعندما تقوم بكتابة برنامج جديد فى بيئة NET.، يتم

بدايةً تشغيل بيئة تطوير Visual Studio.Net ثم تحديد نوع البرنامج الذى تريد كتابته باللغة البرمجية التى ترغب فى استخدامها، فيمكنك على سبيل المثال كتابة برنامج نوافذى باستخدام لغة Visual Basic أو برنامج آخر باستخدام لغة C#.

أما Visual Basic.Net فهى اللغة البرمجية التى يتم شرحها على مدار هذا الكتاب بأكمله. فلم يعد بإمكانك الآن تشغيل لغة Visual Basic أو تحميلها ككيان منفصل، فهى تعد بمثابة إحدى اللغات البرمجية التى تعمل من خلالها بيئة عمل .NET Framework الخاصة بشركة مايكروسوفت.

ويعمل Visual Basic 2008 من خلال بيئة عمل تسمى .NET Framework. وهى عبارة عن الطبقة الموجودة بين اللغة (لغة Visual Basic فى هذه الحالة) ونظام التشغيل الذى يمكن أن يكون إصدار Windows 98 أو Windows ME أو Windows 2000 أو Windows XP أو Windows Vista أو Windows Server 2003 أو أياً من الإصدارات الفرعية الأخرى. وتعمل طبقة .NET Framework على توفير الإمكانيات الوظيفية المطلوبة اعتماداً على أسلوب عمل نظام Windows ومجموعات الملفات الخاصة بإمكانيات وظيفية أخرى.

تحتوى بيئة تطوير Visual Studio 2008 على محرر كود مشترك للغات Visual Basic.Net وVisual C# 2008 وVisual C++ 2008 كما تحتوى على محرر XML ومحرر HTML وواجهة خادم SQL.

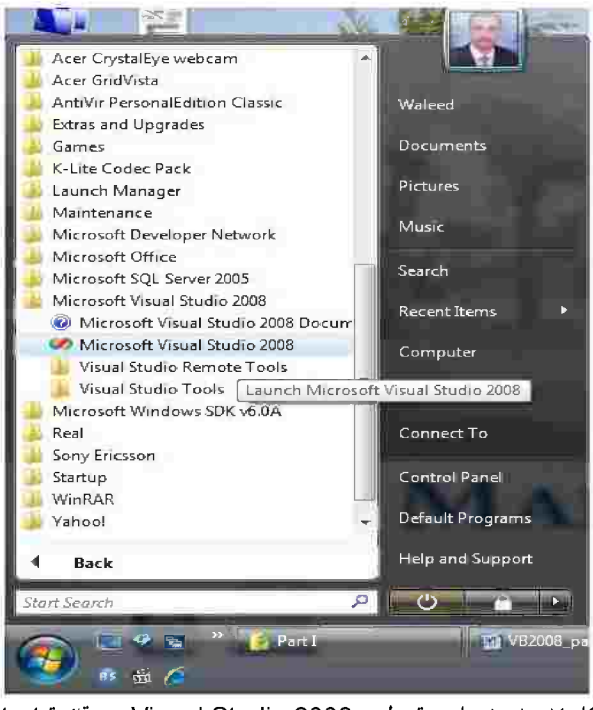
عندما تبدأ تشغيل Visual Studio.Net لأول مرة، ستحصل على بيئة تطوير متكاملة تسمى IDE اختصاراً للعبارة Integrated Development Environment تشتمل على العديد من النوافذ مثل نافذة الخصائص Properties التى تظهر على اليمين، كما تشتمل أيضاً على نافذة "مربع الأدوات" Toolbox الذى يظهر على اليمين أيضاً، بالإضافة إلى عدد من التبويبات Tabs ويتسبب نقر أى تبويب فى ظهور نافذته المصاحبة. وعندما تقف بمؤشر الفأرة على أى تبويب لمدة ثوانى ستظهر النافذة التى تخصه. لإخفاء النافذة، حرك مؤشر الفأرة بعيداً عنها.

فتح بيئة تطوير Visual Studio 2008

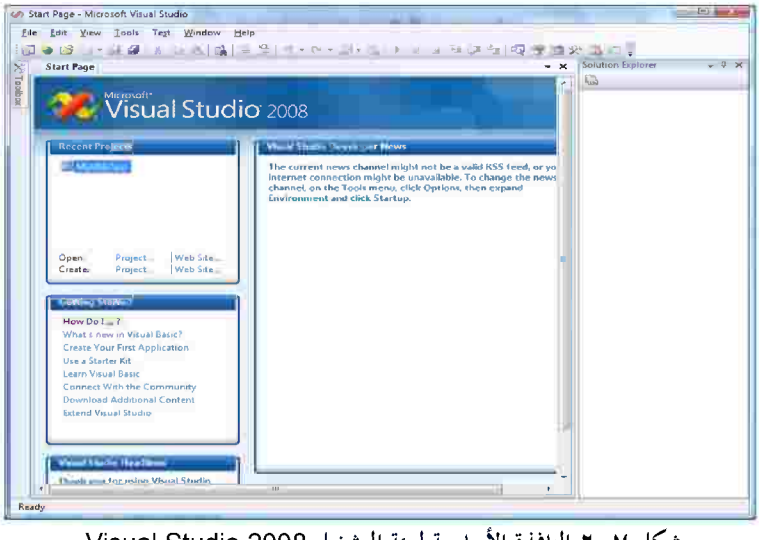
تعتبر بيئة التطوير المتكاملة Visual Studio 2008 آخر بيئات التطوير التى طرحتها مايكروسوفت والمستخدمة لتطوير جميع تطبيقات مجموعة Visual Studio 2008. وبيئة التطوير هذه مبنية أساساً على بيئات التطوير السابقة مع إضافة العديد من السمات المفيدة والذكية فى نفس الوقت والتى تساعدك على الوصول إلى ما تريد بسرعة وسهولة متناهيين وخاصةً تلك السمات المتعلقة بالويب.

لفتح بيئة تطوير Visual Studio 2008 كى تتمكن من تطوير تطبيقاتك سواءً باستخدام لغة Visual Basic 2008 أو أيٍ من اللغات الأخرى، تابع معنا الخطوات الآتية:

١. انقر زر Start ثم اختر All Programs من القائمة التابعة (أو اختر Programs إذا كنت تعمل على نظام التشغيل Windows 2000 أو Windows NT).
٢. اختر مجموعة Microsoft Visual Studio 2008 من قائمة البرامج التابعة.
٣. انقر Microsoft Visual Studio 2008 مرةً أخرى من القائمة التابعة (انظر شكل ٧-١).
٤. تبدأ بيئة تطوير Visual Studio 2008 فى العمل وبعد ثوانٍ معدودة تظهر النافذة الأساسية فى بيئة التشغيل محتويةً على صفحة البدء Start Page (انظر شكل ٧-٢).



شكل ٧–١ تشغيل بيئة تطوير Visual Studio 2008 من قائمة Start.



شكل ٧–٢ النافذة الأساسية لبيئة التشغيل Visual Studio 2008.

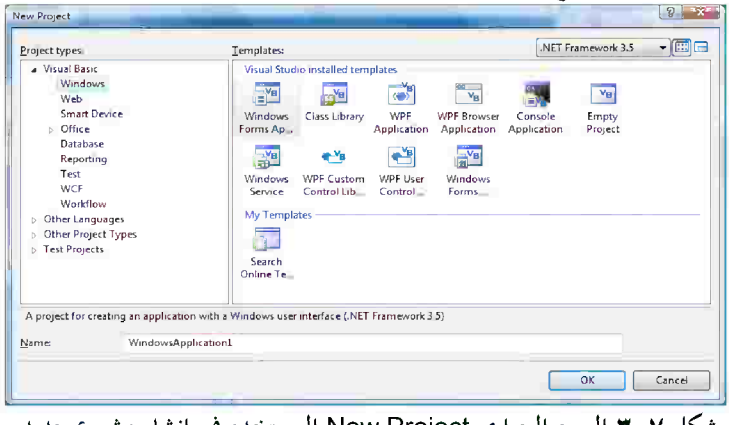
تحتوى صفحة البدء Start Page على قائمة بأسماء المشروعات التى قمت بفتحها مؤخراً داخل بيئة التطوير المتكاملة سواءً باستخدام Visual Basic 2008 أو باستخدام أيٍ من اللغات الأخرى الموجودة بالنطاق، فإذا كنت تقوم بفتح Visual Studio للمرة الأولى، فستظهر هذه القائمة خالية. كما تحتوى الصفحة أيضاً على العديد من الارتباطات التشعبية التى يمكنك استخدامها فى فتح مشروعات أخرى أو إنشاء مشروعات جديدة وهى الخيارات التى يمكنك بالطبع تنفيذها من داخل شريط القوائم. هذا إلى جانب العديد من المهام الأخرى المستخدمة لفتح موارد الويب والحصول على المساعدة وهو ما سنتعرف عليه فيما بعد.

إنشاء مشروع جديد

للتعرف على مكونات بيئة التطوير المتكاملة Visual Studio 2008 بشيءٍ من التفصيل، سنقوم فيما يلى بإنشاء مشروع بسيط باستخدام لغة Visual Basic 2008. لأداء ذلك، تابع معنا الخطوات الآتية:

١. تأكد أنك داخل بيئة التطوير المتكاملة وإلا تابع الخطوات السابقة لفتح بيئة التطوير من قائمة Start.

٢. من صفحة البدء، انقر الارتباط Project الموجود بجوار كلمة Create أو افتح قائمة File من شريط القوائم واختر New Project من القائمة المنسدلة، وفى الحالتين يظهر المربع الحوارى New Project (انظر شكل ٧-٣).



شكل ٧-٣ المربع الحوارى New Project المستخدم فى إنشاء مشروع جديد.

٣. يحتوى المربع Project Types الموجود بالجزء الأيسر من المربع الحوارى New Project على الأدوات التى يمكنك استخدامها فى إنشاء مشروعك. تأكد من اختيار Windows داخل المجموعة Visual Basic لأننا نرغب فى إنشاء مشروع باستخدام لغة Visual Basic 2008.

٤. يحتوى المربع Templates الموجود بالجزء الأيمن من المربع الحوارى New Project على العديد من قوالب المشروعات التى يمكنك الاختيار من بينها، حيث يعتمد اختيارك بالطبع على نوع التطبيق الذى ترغب فى إنشائه. يحتوى جدول ٣-١ التالى على أنواع القوالب (المشروعات) التى يمكنك إنشائها من خلال المربع الحوارى New Project.

جدول ٣-١ أنواع قوالب (مشروعات) لغة Visual Basic 2008

نوع المشروع (القالب)	الاستخدام
Windows Forms Application	إنشاء التطبيقات التى تحتوى على واجهة رسومية وهو مشابه إلى حد كبير للتطبيق التنفيذى الموجود بالإصدارات القديمة من Visual Basic
Class Library	إنشاء ملف DLL يحتوى على تصنيف أو أكثر يمكن الوصول إليه من قِبَل التطبيقات الأخرى
Windows Forms Control Library	إنشاء أداة تحكم مخصصة تشبه إلى حد كبير أدوات ActiveX الموجودة بالإصدارات القديمة من Visual Basic
WPF Application	إنشاء تطبيق من النوع WPF (اختصار Windows Presentation Foundation)
WPF Custom Control Library	إنشاء أداة تحكم للاستخدام داخل تطبيقات WPF
Console Application	إنشاء تطبيق نوافذى لا يحتوى على نماذج وإنما يقوم

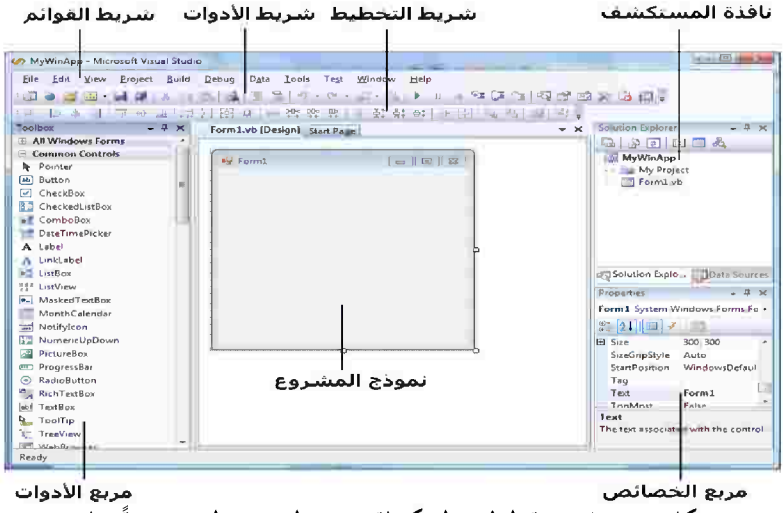
نوع المشروع (القالب)	الاستخدام
	بإجراء جميع عمليات الإدخال والإخراج من خلال نافذة محث الأوامر
Windows Service	إنشاء مشروع يحتوى على خدمة معينة أو أكثر يتم استخدامها من قبل التطبيقات الأخرى الموجودة على شبكة تعمل بنظام التشغيل Windows
Empty Project	إنشاء مشروع نوافذى خالى بحيث تقوم بإنشاء جميع عناصره بنفسك
WPF Browser Application	إنشاء تطبيق مستعرض WPF

٥. من المربع Templates، اختر الرمز Windows Forms Application لأننا نرغب فى إنشاء مشروع نوافذى جديد.

٦. قم بتعيين اسم مميز للمشروع الجديد داخل مربع النص Name وليكن MyWinApp.

٧. يتم إنشاء المشروع الجديد تلقائياً باستخدام كود الإصدار 3.5 من .NET Framework. فإذا أردت استخدام أحد الإصدارات السابقة، اختر هذا الإصدار من القائمة المنسدلة الموجودة بالركن الأيمن العلوى من المربع الحوارى.

٨. انقر زر Ok، وحينئذٍ تقوم بيئة التطوير بإغلاق المربع الحوارى New Project وإنشاء مشروع جديد يحتوى على نموذج واحد باسم Form1.vb (انظر شكل ٧-٤).



شكل ٧-٤ تقوم بيئة التطوير المتكاملة بإنشاء المشروع الجديد نيابةً عنك

فهم مكونات نافذة المشروع

فيما يلي نوضح أهم العناصر التى تشتمل عليها بيئة التطوير المتكاملة والتي تراها في شكل ٧-٤ السابق، وفيما يلي من فصول الكتاب ستتعرف بالتفصيل على كل عنصر من هذه العناصر وطريقة استخدامه.

شريط القوائم Menu Bar

يحتوى على مجموعة من القوائم التى يحتوى كل منها على مجموعة من الخيارات المختلفة. فمثلا تستخدم قائمة File للتعامل مع ملفات المشروعات المختلفة، بينما تستخدم قائمة Edit في كتابة البرامج وتصميم النماذج، وتستخدم قائمة View للتحكم في عرض أو إخفاء عناصر بيئة التطوير، وتستخدم القائمة Debug لاكتشاف وتصحيح الأخطاء وتنفيذ أو إيقاف البرنامج، كما تتيح قائمه Tools التحكم فى اختيارات عامة تختص بالشكل العام لبيئة التطوير، بينما تستخدم القائمة Window للتحكم في طريقة عرض نوافذ بيئة التطوير، أما قائمة Help فتستخدم للحصول على تعليمات المساعدة.

وحتى لا تلتبس عليك المصطلحات فى هذا المستوى من الدراسة نوضح ما يلى:

لقد كانت الإصدارات القديمة للغة Visual Basic تمثل بيئات تطوير كاملة. أما فى الإصدار الأخير منها، فهى تعد جزءاً واحداً من استراتيجية برمجة تتكون من:

- اللغة: فى هذا الكتاب هى Visual Basic 2008. وتتضمن اللغات الأخرى الممكنة C# وVisual C++وغيرها.
- بيئة التطوير المتكاملة Integrated Development Environment (IDE): وهى بيئة تطوير Visual Studio 2008 فى هذا الكتاب. أما عن IDEs الممكنة، فتتضمن Borland وMacromedia والعديد من الأدوات الأخرى.
- المشروع: فى هذا الكتاب، سيتم تناول أربعة أنواع أساسية من المشروعات، وهى تطبيقات Windows Forms وتطبيقات Web Forms ومجموعات ملفات التصنيفات وخدمات ويب XML، كما يمكنك استخدام لغة Visual Basic لإنشاء خدمات ويب وتطبيقات معتمدة على وحدة تحكم (Console applications) وتطبيقات Smart Device وMobile Web Forms والعديد من أنواع المشروعات الأخرى.

النموذج Form

يظهر النموذج فى منتصف الشاشة تقريباً ويستخدم لتصميم واجهات التطبيق، حيث يوضع عليه جميع الأدوات المستخدمة (أدوات كتابة – صور – الخ ...) ويحتوى التطبيق على واحد أو أكثر من هذه النماذج. وقد يستخدم كنموذج أو تتفرع عنه عدة نماذج أخرى. وقد يحتوى النموذج على شبكة من النقاط تساعد على تصميم الواجهة وضبط مواقع الأدوات والعناصر المختلفة (راجع شكل ٧-٤).

شريط الأدوات Toolbar

يحتوى شريط الأدوات على مجموعة من الأزرار التى يقوم كل منها مقام وظيفة داخل شريط القوائم، حيث يتم استدعاء هذه الوظيفة مباشرةً بمجرد النقر عليها بدلا من فتح القوائم واختيار الأوامر منها. فمثلا أول زر من ناحية اليسار فى شريط الأدوات القياسي

هو New Project ويعنى فتح المربع الحوارى New Project بمجرد النقر عليه بالفأرة وهو إجراء بديل لفتح قائمة File ثم اختيار أمر New Project .

وتشتمل بيئة تطوير Visual Studio 2008 على عدد من أشرطة الأدوات المختلفة. لإظهار أو إخفاء شريط/أشرطة الأدوات، اختر أمر Toolbars من قائمة View ثم اختر اسم الشريط المطلوب من القائمة التابعة.

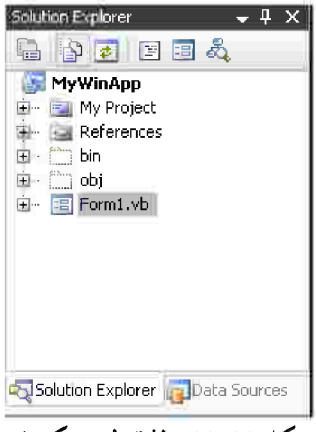
نافذة مستكشف الحل Solution Explorer

إذا كنت ممن استخدموا الإصدارات السابقة من Visual Basic، فلعلك اعتدت على مصطلح "المشروع" Project والذى يحتوى دائماً على الأجزاء المختلفة للتطبيق الذى تقوم بتطويره حيث يتم تخزين كل جزء داخل ملف مستقل ثم تجميع هذه الملفات داخل مشروع واحد. وابتداءً من Visual Studio.NET تم استحداث مصطلح جديد وهو "الحل" Solution الذى قد يحتوى على مشروع واحد أو أكثر.

وحينما تقوم بإنشاء مشروع جديد، يتم إنشاء مشروع جديد داخل حل جديد. أما إذا أردت إضافة مشروع جديد إلى الحل الحالى، اختر New Project من القائمة الفرعية Add المتفرعة من قائمة File، حيث يظهر فى هذه الحالة المربع الحوارى Add New Project الذى يشبه إلى حدٍ كبير المربع الحوارى New Project السابق.

تحتوى "نافذة مستكشف الحل" Solution Explorer Window أو ما نطلق عليه اختصاراً "نافذة المستكشف" على قائمة بجميع الملفات والعناصر الموجودة داخل الحل الحالى الذى يحتوى كما ذكرنا على مشروع واحد أو أكثر، وهى بذلك تشبه إلى حد كبير نافذة المشروع الموجودة بالإصدارات السابقة من Visual Basic.

ويتم ترتيب العناصر والملفات داخل نافذة المستكشف فى شكل هرمى بحيث يظهر اسم الحل فى أعلى النافذة ثم أسماء المشروعات الموجودة داخل الحل بحيث يحتوى كل مشروع على ملفاته مستقلة ثم المراجع References المستخدمة من قِبَل هذا المشروع وبعد ذلك تأتى العناصر التى لا تنتمى إلى مشروع معين (انظر شكل ٧-٧).



شكل ٧-٧ نافذة المستكشف

مربع الأدوات Toolbox

يشتمل مربع الأدوات على مجموعة من أدوات التحكم التى تسهل التعامل مع Visual Basic وتستخدم فى إعداد المشروعات. كما يمكن نقل مربع الأدوات إلى أي مكان داخل الشاشة حتى يسهل عليك التعامل معه ومع بقية عناصر الشاشة (راجع شكل ٧-٤).

مربع الخصائص Properties Window

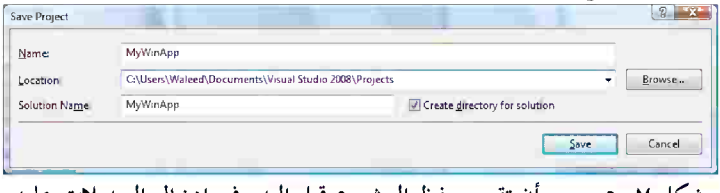
يحتوى مربع الخصائص على مجموعة الخصائص التي تخص كائن معين موجود في بيئة التطوير ويمكن عن طريقه تغيير الخصائص التى تخص الكائن المختار (انظر شكل ٧-٨).

شكل ٧–٨ مربع الخصائص يبين خصائص النموذج Form1

حفظ المشروع وتسميته

لحفظ المشروع على القرص الصلب كى تتمكن من استرجاعه فى أى وقت والتعديل فيه اتبع الآتي:

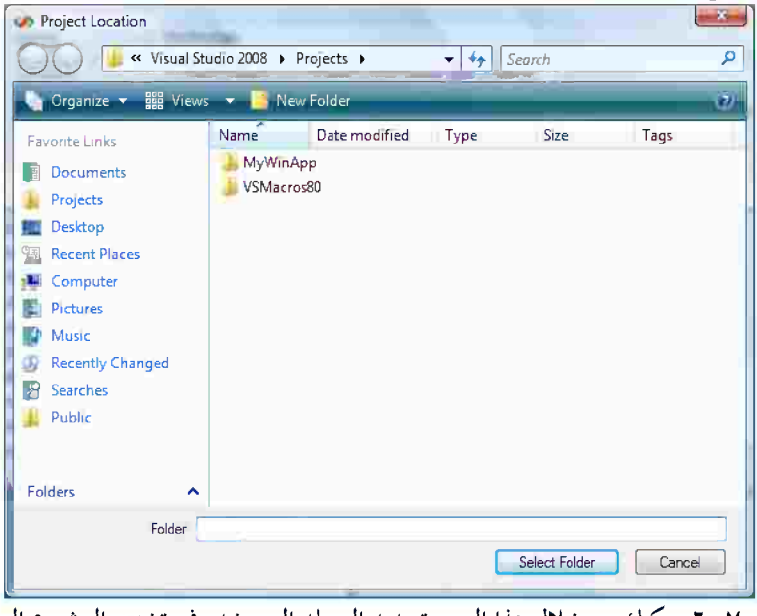
١. افتح قائمة File بشريط القوائم ثم اختر Save All من القائمة المنسدلة الناتجة، وحينئذٍ يظهر المربع الحوارى Save Project (انظر شكل ٧–٥). قم بتعيين بيانات المشروع كما يلى:



شكل ٧–٥ يجب أن تقوم بحفظ المشروع قبل البدء فى إدخال التعديلات عليه

- قم بتغيير اسم المشروع إن أحببت من خلال مربع النص Name.
- قم إن أحببت بتغيير مكان المشروع من خلال القائمة المنسدلة Location التى تحتوى بدورها على المجلد الافتراضى لتخزين المشروعات وهو C:\Users\username\Documents\Visual Studio 2008\Projects (مع

استبدال Username باسم المستخدم الخاص بك) إلى جانب أى مجلدات قمت باختيارها من قبل. فإذا أردت اختيار مجلد آخر لحفظ المشروع، انقر زر Browse المجاور للقائمة المنسدلة ثم قم بتحديد المجلد من المربع الحوارى الناتج Project Location (انظر شكل ٧-٦).



شكل ٧-٦ يمكنك من خلال هذا المربع تحديد المجلد المستخدم فى تخزين المشروع الجديد

- يتم تخزين المشروع داخل حل (Solution) الذى قد يحتوى بدوره على أكثر من مشروع، حيث يتم افتراضياً تسمية الحل بنفس اسم المشروع. فإذا أردت تعيين اسم آخر للحل، قم بإدخال الاسم الجديد إلى مربع النص Solution Name. ويمكنك التحكم فى اسم الحل فى حالة تنشيط مربع الاختيار المجاور Create directory for solution والذى يتم تنشيطه إذا ما أردت إنشاء مجلد مستقل للحل لتمييزه عن الحلول الأخرى الموجودة بنفس المجلد.

٢. بمجرد تعيين الخيارات المختلفة للمشروع والحل، انقر زر Save لإغلاق المربع الحوارى Save Project وحفظ المشروع الجديد والعودة مرةً أخرى إلى النافذة الرئيسية لبيئة التطوير.

قد تختلف بيئة التطوير أو بعض عناصرها عن تلك التى تظهر لديك، يعتمد ذلك على ما إذا كنت تشغل بيئة التطوير Visual Studio 2008 لأول مرة أم لا وعلى ما إذا كان سبق لك ضبط وتعديل بيئة التطوير. كما تختلف أيضاً بيئة Visual Studio تبعاً لنوع الإصدار الذى قمت بتثبيته عندك. سنتعرف بعد قليل على كيفية ضبط وتهيئة بيئة التطوير كى نستخدم سوياً بيئة تطوير متشابهة طوال عرض هذا الكتاب بأجزائه المختلفة.

وبالنظر إلى شكل ٧-٤ السابق، تلاحظ استخدام Visual Basic 2008 لنفس العناصر التى تستخدمها معظم برامج Windows ، فشريط الأدوات وشريط القوائم ومربع الأدوات تشبه إلى حد كبير تلك التى تستخدمها برامج Microsoft Office كبرنامج Microsoft Access على سبيل المثال.

الفرق بين الحل والمشروع والبرنامج

عند العمل داخل بيئة تطوير Visual Studio 2008 باستخدام لغة برمجة Visual Basic 2008، يتم إنشاء نوعين من البرامج وهما:

- البرنامج Program هو النهاية التى تصل إليها بعد الانتهاء من المشروع، وهو النسخة المترجمة من المشروع حيث يمكنك توزيع البرنامج لأشخاص آخرين، وعلى أجهزة أخرى. ويمكنك تنفيذه وتشغيله بدون حاجة إلى وجود Visual Basic 2008 على هذه الأجهزة.
- المشروع Project هو ما تقوم بإنشائه فعليا عندما تعمل داخل بيئة تصميم Visual Studio 2008، وعلى هذا يمكن القول أن مشروع Visual Basic 2008 هو مجموعة من الملفات، فعندما تشير إلى مشروع من المشروعات، فإنك تشير إلى مجموعة متكاملة من الملفات التى يتكون منها المشروع .
- الحل Solution: يحتوى الحل على مشروع أو أكثر ويتم افتراضياً تسمية الحل بنفس اسم المشروع.