الفصل الثاني

بيئة تطوير برامج فيجوال ستوديو .نت
THE VISUAL STUDIO .NET DEVELOPMENT ENVIRONMENT

أهداف الفصل:

- استكشاف بيئة تطوير برامج فيجوال ستوديو .نت.
- تطوير أول مشروع مستخدماً لغة فيجوال بيسك .نت (VB .NET).
- ترجمة برامج فيجوال بيسك .نت وتنفيذها.
- استخدام مصمم النماذج المرئي (Visual form designer).
- استكشاف أدوات تصحيح البرامج.
- استكشاف أدوات المساعدة.

قدمنا لك من خلال الفصل الأول أساسيات تطوير البرامج المعتمدة على الكائنات (Object-oriented System Development) وخواصها، حيث تم التعرف على فوائد تطوير النظم مستخدماً هذه التقنية والتعرف أيضاً على الأدوات التي تمكن محللي النظم من نمذجة متطلبات النظم (System Requirements Modeling) مثل لغة UML، وقد قدمنا أيضاً المفاهيم الأساسية لتقنية الكائنات (Object-oriented) مثل الكائنات (Objects) والإجراءات (Methods) والتوارث (Inheritance) والكبسلة (Encapsulation).

وقد ذكرنا أيضاً أن لغة فيجوال بيسك .نت تمثل جزءاً من مايكروسوفت فيجوال ستوديو .نت، والتي نفسها تمثل جزءاً من تقنية أكبر وهي تقنية .نت التي تدعم نظم تشغيل مختلفة مثل نظام تشغيل Windows 2000، ونظام تشغيل Windows XP، ونظام تشغيل Windows NT. كما تشتمل تقنية .نت على العديد من لغات البرمجة مثل لغة C++ .NET، ولغة C# .NET، وبالطبع لغة Visual Basic .NET. ويجب التذكير أن إطار عمل .نت

(.NET framework) يتكون من المكون محرك تنفيذ اللغات المشترك Common Language Runtime (CLR)، ومكتبة أصناف .نت (.NET class library) اللذان يقدمان خدمات مشتركة لجميع لغات فيجوال ستوديو .نت.

عزيزي القارئ سوف تتعلم من خلال هذا الفصل كيفية كتابة وتنفيذ برامج فيجوال بيسك .نت (VB .NET) حيث سيتم استكشاف المعالم الرئيسية لبيئة عمل تطوير .نت واستخدام الأدوات والنوافذ الرئيسية لتطوير برامج بسيطة، هذا بالإضافة إلى تقديم كل من أدوات التصحيح وأدوات المساعدة. ولذلك سوف يصبح لديك بعد قراءة هذا الفصل أول خبرة بلغة فيجوال بيسك .نت وكيفية استخدام بيئة التطوير التي تمكنك من إنشاء البرامج وتعديلها وتنفيذها مستخدماً هذه اللغة.

استكشاف بيئة تطوير برامج فيجوال ستوديو .نت
Exploring the Visual Studio .NET Development Environment

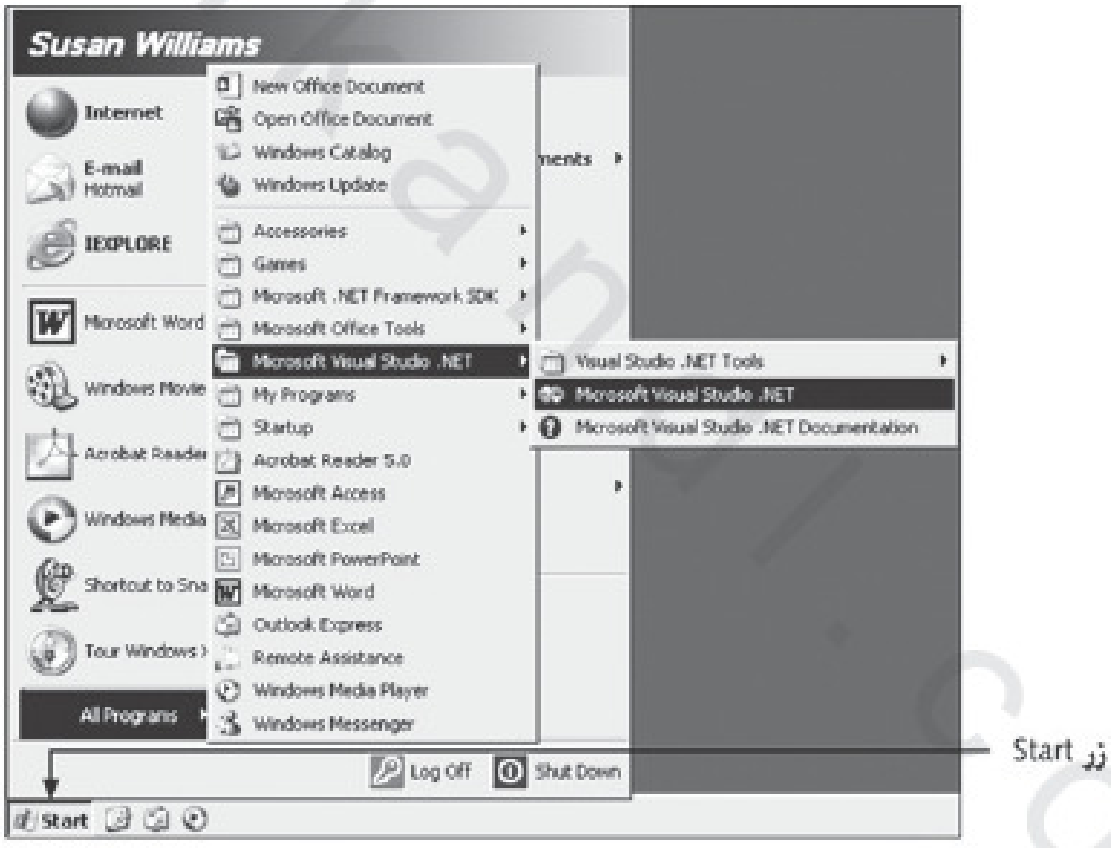
كما ذكرنا في الفصل الأول أن لغة فيجوال بيسك .نت هي واحدة من ضمن عدة لغات مدعمة بواسطة بيئة التطوير المتكاملة (Integrated Development Environment) لفيجوال ستوديو .نت التي تختصر بالاسم IDE. تتكون بيئة التطوير المتكاملة من عدة أدوات برمجية لمساعدة المبرمج على كتابة شفرة البرنامج واختبارها وتوثيقها. وسوف نرى أن بيئة تطوير فيجوال ستوديو .نت تيسر عملية تطوير أنظمة الحاسب وتطبيقات الإنترنت بسبب ما تحتويه من إمكانات هائلة، ومن ثم تقلل من زمن تطوير هذه البرامج. ومن خلال التعرف على تطوير البرامج مستخدماً لغة فيجوال بيسك .نت سوف تحصل على خبرة هائلة بهذه البيئة والتي يمكن تطبيقها مع أي لغة أخرى تعمل مع تقنية .نت.

مثل معظم بيئات تطوير البرامج المتكاملة، فإن فيجوال ستوديو .نت تحتوي على العديد من الأدوات المتنوعة بداية من محرر نصوص بسيط إلى محررات ذكية التي تميز الأوامر أثناء كتابتها ومن ثم تكملتها، كما تشمل على أدوات تطوير برمجة مرئية (Visual Programming) التي تنشئ شفرة أوامر البرنامج بناء على أشكال رسومية تم تصميمها بواسطة المبرمج، ومن ثم فإن فيجوال ستوديو .نت تيسر مهام تطوير البرامج مثل ترجمة البرامج وتنفيذها واختبارها وعزل الأخطاء البرمجية لتجعل البرامج تعمل كما هو متوقع. كما تقدم فيجوال ستوديو .نت أدوات مساعدة عديدة للمبرمج أثناء عملية كتابة شفرة البرامج. سيتم التركيز في هذا الفصل على فهم الأدوات والمفاهيم والنوافذ الضرورية لتطوير برامج فيجوال بيسك .نت.

البداية مع لغة فيجوال بيسك .نت
Getting Started with VB .NET

لكي تتمكن عزيزي القارئ من تنفيذ تمارين هذا الفصل، يجب أن يحتوي جهاز الحاسب الخاص بك على برنامج فيجوال ستوديو .نت حيث يمكنك الرجوع إلى صدر هذا الكتاب لمعرفة تعليمات تنصيب هذا البرنامج، وفي حالة الانتهاء من تنصيب البرنامج فإن تشغيل لغة فيجوال بيسك .نت أمر سهل للغاية. ولتشغيل فيجوال بيسك .نت تحت بيئة Windows XP، اتبع الخطوات التالية:

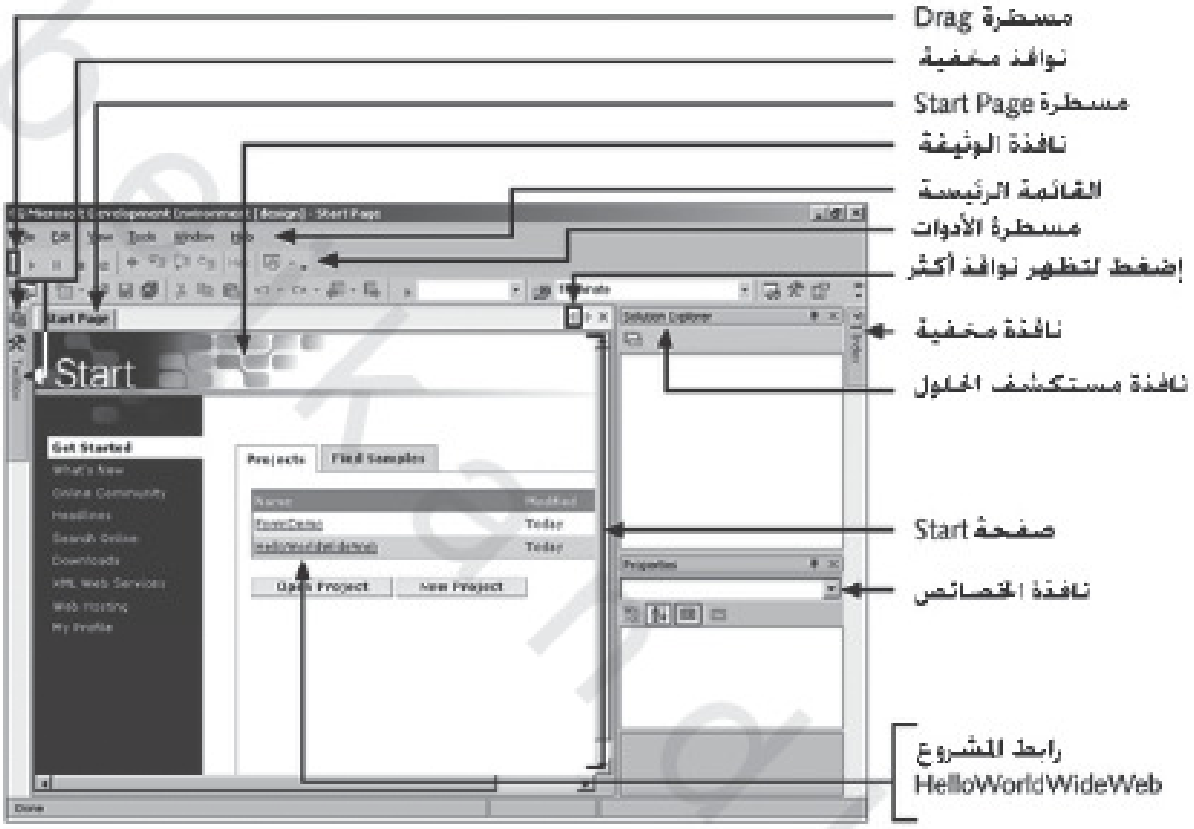
١- اضغط على قائمة Start، ثم اختر All Programs، ثم اختر Microsoft Visual Studio .NET، ثم اضغط على الأمر Microsoft Visual Studio .NET كما هو واضح في الشكل رقم (٢.١).



الشكل رقم (٢.١). تشغيل برنامج فيجوال ستوديو .نت.

٢- عندئذ ستظهر نافذة تحتوي على لغات فيجوال ستوديو .نت المتواجدة على جهازك الحاسب لمدة ثوان، ثم تظهر نافذة بيئة تطوير مايكروسوفت (Microsoft Development Environment) كما هو واضح في الشكل رقم (٢.٢) (ربما يتغير مظهر هذه النافذة قليلاً بناء على ضبط إعدادات جهازك الحاسب).

لاحظ أن الأشكال الواردة في هذا الكتاب تستخدم نظام تشغيل Windows XP ذات الرؤية الكلاسيكية والتي تشبه مظهر نظام التشغيل Windows 2000.



الشكل رقم (٢,٢). بيئة مايكروسوفت المتكاملة.

استكشاف نافذة بيئة تطوير مايكروسوفت

Exploring the MDE

كما هو واضح في الشكل رقم (٢,٢) أن نافذة بيئة تطوير مايكروسوفت (Microsoft Development Environment) التي تختصر بنافذة MDE تحتوي على قائمة رئيسة ومساطر أدوات مختلفة والعديد من النوافذ. تستخدم القائمة الرئيسة لاختيار الأوامر التي تنجز المهام أثناء عملية التطوير مثل فتح الملفات وغلقها، وفتح المشاريع وغلقها، وترجمة البرامج وتنفيذها وتصحيحها، وأخيراً التعامل مع أدوات المساعدة.

تحتوي مساطر الأدوات على العديد من أزرار الأوامر الأكثر شيوعاً، ولكي تتعرف على فائدة هذه الأزرار يمكنك أن تشاور بالفأرة عليها وتنتظر ظهور نافذة المعلومات (Screen tip). توجد العديد من مساطر الأدوات المتاحة التي يمكن إظهارها أو إخفائها عن طريق ضغط الزر الأيمن للفأرة داخل مكان فارغ داخل مسطرة الأدوات، ثم

التأشير (Clicking) أو إلغاء التأشير (Unclicking) على اسم مسطرة الأدوات، ويمكنك أيضاً اختيار القائمة الفرعية Tools من القائمة الرئيسة، ثم اختيار أمر Customize، ثم الضغط على مسطرة صناديق الأدوات Toolbars لإظهار مساطر الأدوات أو إخفائها. يمكنك أيضاً تغيير موضع أي مسطرة أدوات بواسطة استخدام رمز Drag bar (عبارة عن مجموعة من الخطوط القصيرة التي تظهر في الحافة اليسرى لمسطرة الأدوات) لتحريكها إلى موقع جديد. يحتوي باقي نافذة MDE على العديد من النوافذ المختلفة بناء على ضبط إعدادات البرنامج حيث ربما لا تظهر جميع النوافذ الموضحة في الشكل رقم (٢.٢)، كما يجب أن تلاحظ أن نافذة Start Page تشغل معظم الشاشة التي تظهر كصفحة داخل نافذة المستند حيث يمكن الانتقال إليها عن طريق نقر المسطرة الخاصة بها (Start Page Tab)، فمثلاً يحتوي الشكل رقم (٢.٢) على مستند Start Page، ومستند "HelloWorldWideWeb.vb" حيث يمكن التنقل بينهما عن طريق نقر المساطر الخاصة بهما، وعندئذ يظهر المستند النشط (المختار بخط أسود عريض)، ولاحظ أن مستند Start Page هو المستند النشط. يمكنك استخدام الأسهم اليمنى واليسرى الموجودة في مسطرة العنوان العلوي لكي تظهر المستندات الواقعة خارج رؤية البرنامج الحالية وذلك في حال وجود عدد كبير من المستندات في نفس الوقت.

يحتوي الشكل رقم (٢.٢) على كل من مستكشف الحلول (Solution Explorer window) ونافذة الخصائص (Properties window) حيث سيتم التعرف عليهم لاحقاً في هذا الفصل، كما يظهر الشكل رقم (٢.٢) مجموعة من المساطر (Tabs) بطول جانب النافذة التي تشير إلى وجود نوافذ مخفية (Hidden windows)، ومن ثم يتمكن المبرمج من إظهار ما يريد من نوافذ وأدوات عند الحاجة دون إشغال نافذة البرنامج بالعديد من النوافذ المفتوحة. لكي يظهر المبرمج نافذة مخفية، يضع مؤشر الفأرة على مسطرة هذه النافذة، فعندئذ تنزلق النافذة للخارج كما هو واضح في الشكل رقم (٢.٣)، ولإخفاء النافذة نضع مؤشر الفأرة خارج حدود النافذة، فعندئذ ترجع النافذة إلى وضعها المخفي. كما يظهر الشكل رقم (٢.٣) مجموعة من الأزرار لفتح نافذة مستكشف الحلول (Solution Explorer window) ونافذة الخصائص (Properties window) ونافذة صندوق الأدوات (Toolbox window).

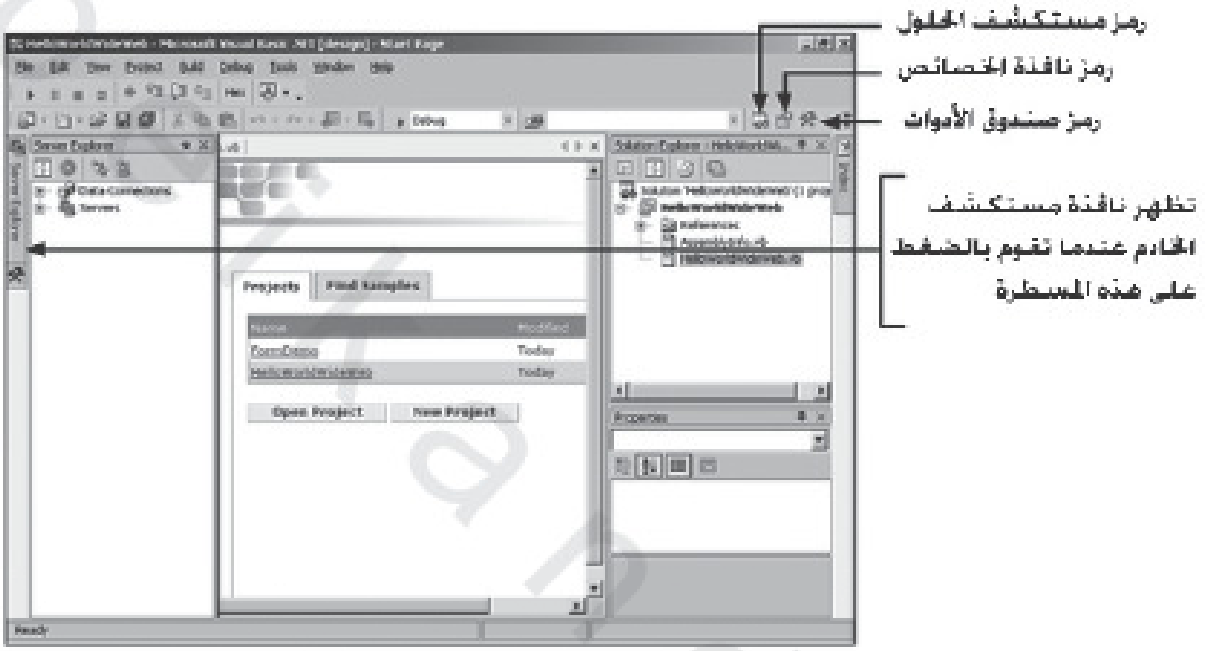
عزيزي القارئ عند استخدامك برنامج فيجوال ستوديو .نت للمرة الأولى يجب أن تتشابه نافذة MDE الخاصة بك بالنافذة الموضحة في الشكل رقم (٢.٢) بشرط قبولك للإعدادات الافتراضية عند تنصيب البرنامج. فإن لم يكن كذلك فإن الخطوات التالية تساعدك على ضبط برنامج فيجوال ستوديو .نت لكي يعمل بالإعدادات الافتراضية:

١- اختر القائمة الفرعية Tools ثم الأمر Customize، عندئذ يظهر الصندوق الحواري Customize (انظر الشكل رقم ٢.٤).

٢- اختر مسطرة Toolbars إذ لم تكن مختارة بالفعل.

٣- تأكد أن العناصر المختارة هي العنصر MenuBar والعنصر Standard والعنصر Debug فقط (يمكنك إضافة بعض مساطر الأدوات لاحقاً).

٤- اضغط على زر Close لكي تغلق الصندوق الحواري وتنفذ الإعدادات المطلوبة.



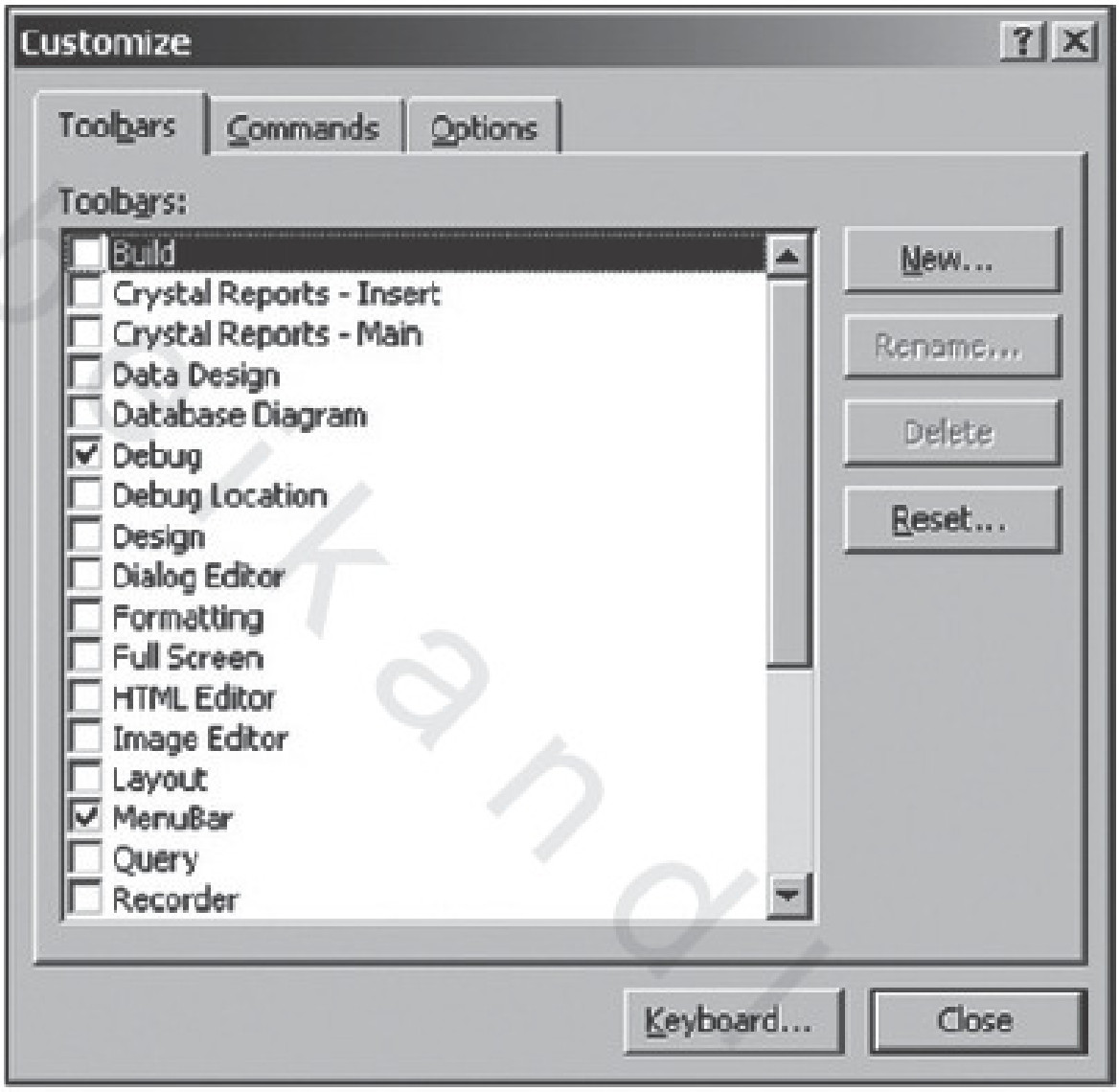
الشكل رقم (٢,٣). إظهار نافذة مخفية.

٥- أغلق أي نافذة مفتوحة عدا نافذة مستكشف الحلول (Solution Explorer window) ونافذة الخصائص (Properties window) عن طريق الزر "X" الموجود في الركن العلوي الأيمن للنافذة.

٦- افتح نافذة مستكشف الحلول (Solution Explorer window) إذ لم تكن مفتوحة عن طريق زر Solution Explorer المتواجد داخل مسطرة الأدوات، مع العلم أن هذه النافذة تظهر في الركن الأيمن العلوي.

لاحظ أنه يمكنك أيضاً فتح نافذة مستكشف الحلول (Solution Explorer window) عن طريق اختيار القائمة الفرعية View ثم الضغط على Solution Explorer.

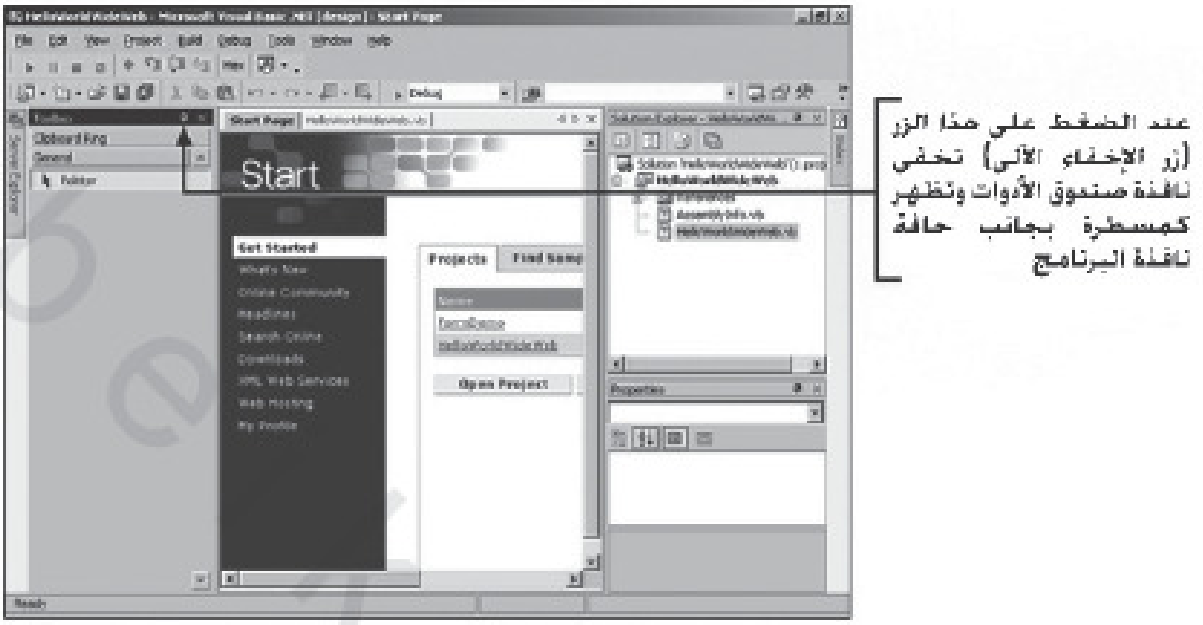
٧- افتح نافذة الخصائص (Properties window) إذ لم تكن مفتوحة عن طريق زر Properties window المتواجد داخل مسطرة الأدوات، مع العلم أن هذه النافذة تظهر في الركن الأيمن السفلي.



الشكل رقم (٢,٤). نافذة Customize.

٨- إذا لم تكن نافذة صندوق الأدوات (Toolbox window) مفتوحة كنافذة مخفية في الحافة اليسرى لنافذة MDE، اضغط على زر Toolbox المتواجد داخل مسطرة الأدوات، فعندئذ تظهر نافذة صندوق الأدوات (Toolbox window) كما هو واضح في الشكل رقم (٢.٥).

٩- اضغط على زر Auto Hide الموجود في عنوان نافذة صندوق الأدوات ثم حرك مؤشر الفأرة إلى منتصف الشاشة لكي تنزلق نافذة صندوق الأدوات إلى الداخل وتصبح نافذة مخفية.

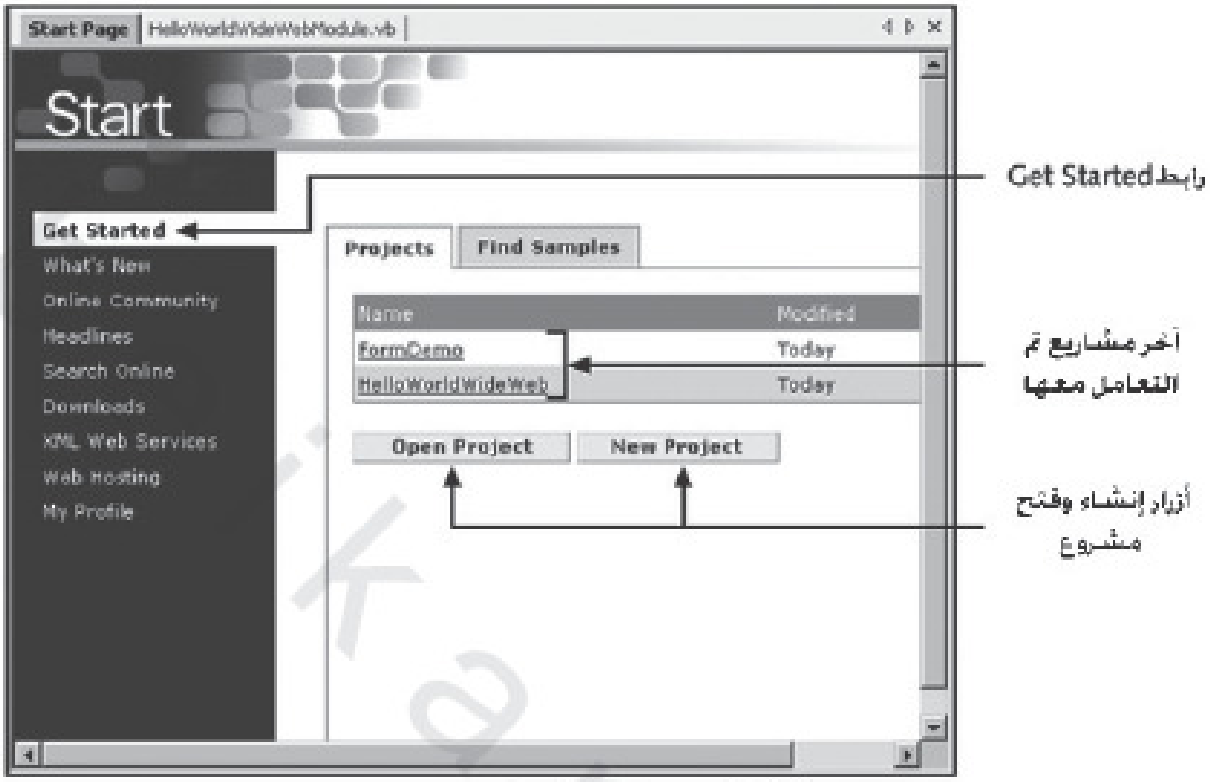


الشكل رقم (٢,٥). نافذة صندوق الأدوات.

التعرف على مستند Start Page
Understanding the Start Page

يظهر مستند Start Page بشكل تلقائي عند تشغيل برنامج Visual Studio .NET كما هو واضح في الشكل رقم (٢.٦) حيث يحتوي على العديد من روابط استدعاء موارد مساعدة هامة والتي يتطلب ظهور معظمها وجود اتصال بشبكة الإنترنت. سوف نركز في هذا الفصل على محتوى الارتباط Get Started link الذي يمكن المبرمج من إنشاء مشروع جديد أو استمرار العمل في مشروع قديم. عند الضغط على رابطة استدعاء Get Started يظهر في الجزء الأيمن للنافذة قائمة بالمشاريع التي تم التعامل معها حديثاً (Recent projects) كما هو واضح في الشكل رقم (٢.٦)، وتظهر أزرار المشاريع (Project buttons).

ولفتح أحد مشاريع القائمة يتم النقر على اسمه، أما إذا أردت العمل مع مشروع غير ظاهر في القائمة يجب الضغط على زر "Open Project" ثم التجول داخل أقراص الجهاز لاختيار المشروع المراد فتحه. أما إذا أردت إنشاء مشروع جديد فعليك الضغط على زر "New Project". ولإظهار مستند Start Page بعد غلق غير مقصود، فيتم ذلك عن طريق نقر القائمة Help ثم اختيار أمر Show Start Page.



الشكل رقم (٢,٦). صفحة البداية (Start Page).

إنشاء مشروع جديد باستخدام لغة فيجوال بيسك .نت
Creating a Project Using VB .NET

جرت العادة عند تعلم لغة برمجة جديدة أن ننشئ أول برنامج لنا يظهر الرسالة "Hello World"، وسوف نلتزم بهذا التقليد ولكن سنغير الرسالة فقط لتصبح "Hello World Wide Web". عندما نريد أن ننشئ أحد مشاريع فيجوال ستوديو .نت، يجب علينا أولاً أن نختار نوع المشروع، وبمعنى آخر يجب علينا أن نختار لغة البرمجة التي سنُستخدم في تطوير هذا المشروع، وفي هذه الحالة نختار لغة Visual Basic لإنشاء مشروع فيجوال بيسك .نت(تذكر أن فيجوال ستوديو .نت تدعم العديد من لغات البرمجة). بالإضافة إلى اختيار نوع المشروع، يجب على المبرمج أن يحدد أي قالب مشروع يريده (Template)، ويحدد أيضاً كلاً من اسم المشروع ومساره. تقدم لغة فيجوال بيسك .نت العديد من نماذج المشاريع التي تحدد أي نوع من البرامج يريد المبرمج أن يطوره، وسوف نتعامل في هذه الوحدة مع نموذجين (قالب مشروع Console Application وقالب مشروع Windows Application)، ولكي ننشئ مشروع فيجوال بيسك .نت اتبع الخطوات التالية:

١- شغل برنامج فيجوال بيسك .نت إذا لم تكن قد شغلته بالفعل، عندئذ تظهر لك نافذة MDE التي تحتوي على نافذة Start Page ونافذة متصفح الحلول (Solution Explorer window) ونافذة الخصائص (Properties window).

٢- اختر سطرة Start Page (إذا لم تكن مختارة بالفعل)، عندئذ ستظهر قائمة بالمشاريع الحديثة وزر "Open Project" وزر "New Project". إذا كانت هذه أول مرة تشغيل لبرنامج فيجوال ستوديو .نت، ستظهر قائمة المشاريع فارغة.

٣- اضغط على زر "New Project" لإنشاء مشروع جديد، وعندئذ سيظهر صندوق حواري بعنوان "New Project" كما هو واضح في الشكل رقم (٢.٧).

٤- اختر المجلد Visual Basic Projects من قائمة Project Types (إذا لم يكن مختاراً بالفعل)، ومن قائمة Templates اختر نوع البرنامج المراد إنشائه. وبما أن هذا أول برنامج لك سنختار برنامجاً من النوع Console Application والذي يعمل تحت شاشة محث الدوس بدلاً من العمل تحت بيئة الويندوز.

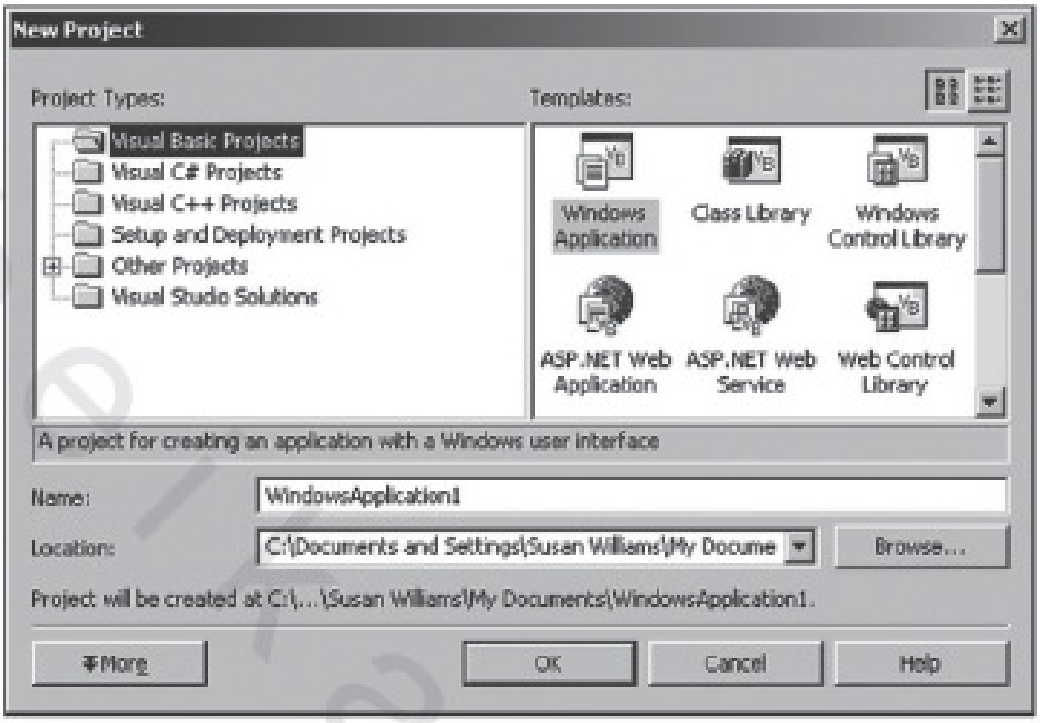
٥- حرك عمود التصفح في قائمة Template إلى أسفل ليظهر رمز Console Application، وعندئذ قم باختياره.

٦- اكتب اسم المشروع المراد إنشائه داخل صندوق النص Name، وسيقترح برنامج فيجوال ستوديو .نت اسم ConsoleApplication1 كاسم افتراضي إذا كانت هذه أول مرة إنشاء Console Application. امسح هذا الاسم ثم اكتب HelloWorldWideWeb داخل صندوق النص Name.

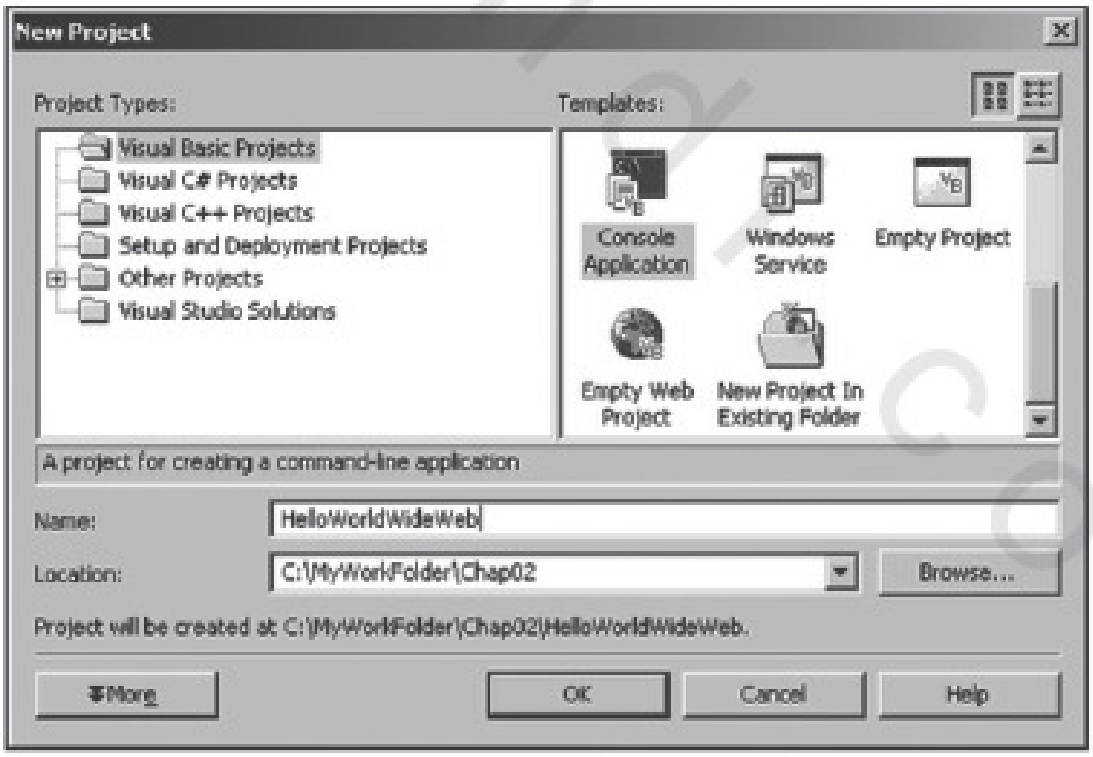
٧- يتم تحديد المجلد الأب للمشروع المراد إنشاؤه داخل صندوق النص Location حيث سيتم تخزين ملفات المشروع داخل مجلد فرعي تحت هذا المجلد، وإذا لم يكن المجلد الأب الذي كتبته غير موجود بالفعل سوف تتولى فيجوال بيسك .نت إنشاؤه بالنيابة عنك. اكتب اسم Chap02 داخل المجلد المخصص للعمل من قبلك كاسم مجلد أب لمشاريع الفصل الثاني (انظر الشكل رقم ٢.٨). تستطيع أن تجد معلومات عن إنشاء المجلدات وتسميتها في صدر هذا الكتاب.

٨- اضغط على زر OK، عندئذ سيظهر الشكل رقم (٢.٩) الذي يوضح مستند Module1.vb داخل نافذة البرنامج، كما سيظهر كلاً من نافذة متصفح الحلول ونافذة الخصائص اللذان يحتويان على معلومات عن مشروعك الجديد.

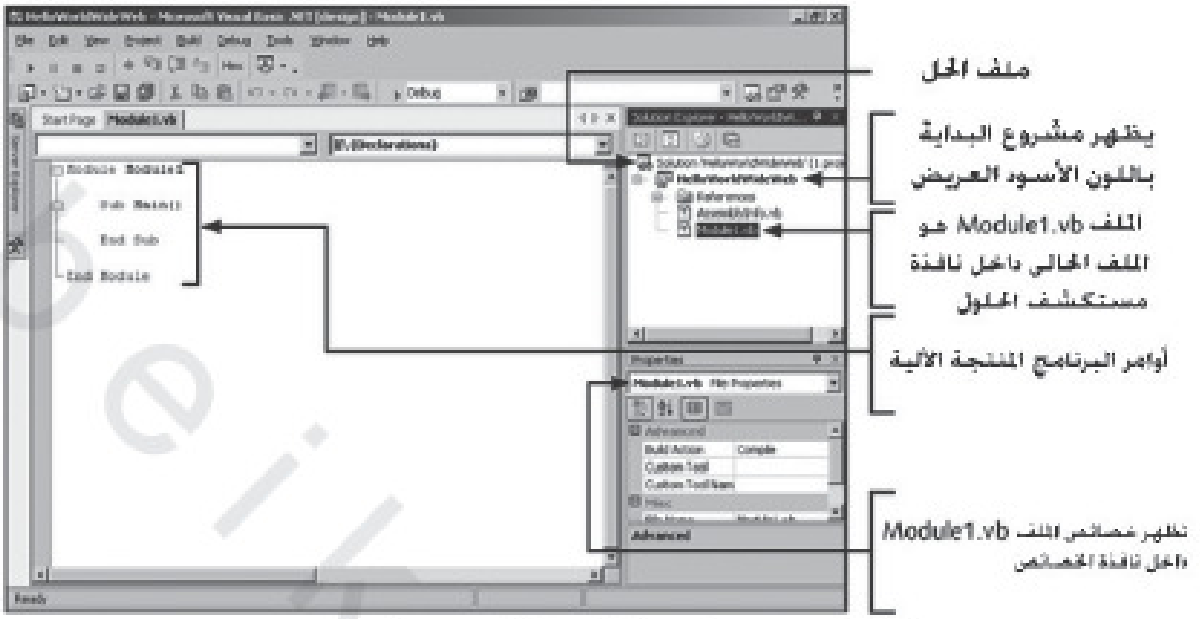
عزيزي القارئ قم بإلقاء نظرة متفحصة لمحتويات كل من مستند Module1.vb ونافذة متصفح الحلول ونافذة الخصائص، وعندئذ ستكتشف أن مستند Module1.vb يحتوي على شفرة تم إنتاجها بشكل آلي بناء على نوع البرنامج الحالي (Console Application) والتي ستعلم معناها خلال الفصل التالي. كما يجب أن تلاحظ أيضاً أن نافذة الخصائص تحتوي على معلومات حول الملف Module1.vb، وعند اختيار عنصر آخر في نافذة متصفح الحلول ستتغير محتويات نافذة الخصائص لتحتوي على معلومات حول هذا العنصر.



الشكل رقم (٢,٧). نافذة إنشاء مشروع جديد.



الشكل رقم (٢,٨). تحديد نوع المشروع ونموذجه واسمه وموقعه.



الشكل رقم (٢.٩). مستكشف الحلول ونافذة الخصائص وأوامر البرنامج.

كيف تنظم لغة فيجوال بيسك .نت البرامج ومحتوياتها

Understanding the Way VB .NET Organizes Your Programs

إن نافذة متصفح الحلول مسؤولة عن تنظيم محتويات الحلول والمشاريع في شكل شجري حيث يجب معرفة كيفية تنظيم هذه العناصر وكذلك العلاقة التي تربطها سوياً. تعطي لغة فيجوال بيسك .نت الاسم الممتد ".vb" لجميع ملفات برامجها كما تعطي الاسم الافتراضي "Module1" ، والاسم "Module2" لملفات تطبيق Console Application. سوف تتعلم كيف تغير هذه الأسماء بإسناد أسماء ملفات ذات معنى خلال هذا الفصل.

تستخدم لغة Visual Studio .NET كلاً من مفهوم المشروع (Project) والحل (Solution) لتنظيم وإدارة الملفات المتعلقة بها. يمثل المشروع أسلوباً لتجميع مجموعة من الملفات ذات العلاقة ، فعلى سبيل المثال ربما يحتوي المشروع على العديد من الملفات البرمجية (Modules) وملفات صور وملفات أخرى متنوعة. أما الحل (Solution) فيمثل حاوية لمشروع أو أكثر حيث يظهر ملف الحل في قمة الشجرة داخل نافذة متصفح الحلول ، وعندما يحتوي الحل على أكثر من مشروع فيجب أن نخصص مشروع البداية (Start Up Project) الذي ينفذ أولاً عند تشغيل الحل ، مع العلم أن معظم الحلول التي سوف تطورها خلال هذا الكتاب سوف تحتوي على مشروع واحد فقط ، ومن ثم سوف يخصص من قبل فيجوال بيسك .نت كمشروع البداية. يظهر مشروع البداية في نافذة متصفح الحلول باللون الأسود العريض كما هو واضح في الشكل رقم (٢.٩).

يحتوي مشروع HelloWorldWideWeb على المجلد References (بالإضافة إلى ملف Module1.vb) وعلى ملف برمجي آخر يطلق عليه اسم "AssemblyInfo.vb". تحتوي هذه العناصر على معلومات مطلوبة للنظام لكي يكمل التطبيق الحالي وينفذه. تُستخدم كل من علامة الطرح وعلامة الجمع الظاهرة على يسار عناصر الشجرة داخل نافذة متصفح الحلول لتوسعة أو لانكماش هذه العناصر.

استخدام محرر النصوص

Using the Text Editor

لكي نجعل التطبيق الحالي يظهر رسالة "HelloWorldWideWeb" يجب تعديل الملف البرمجي Module1.vb مستخدماً محرر النصوص فيجوال ستوديو .نت الذي يقدم العديد من إمكانات محرر النصوص المعتادة بالإضافة إلى إعطاء ألوان مختلفة للنصوص بناء على استخدامها، والتنسيق الآلي لشفرة البرنامج، والتكملة الآلية لأوامر البرنامج. عادة تظهر الكلمات المحجوزة (Keywords) للغة باللون الأزرق والتعليقات (Comments) باللون الأخضر الغامق، وتظهر بقية النصوص باللون الأسود. وكما هو ملاحظ من الشكل رقم (٢.٩) أن أوامر البرنامج تم تنسيقها آلياً بداية من حافة النافذة اليسرى (Code indentation)، مع العلم أن هذا التنسيق ليس ضرورياً لتنفيذ البرنامج ولكنه يحسن مظهر شفرة البرنامج ويسهل قراءتها.

يحتوي محرر النصوص أيضاً على خاصية تكملة الشفرة تلقائياً والتي تسمى IntelliSense التي تساعدك على تكملة أوامر البرنامج بشكل تلقائي من خلال مطابقة الأجزاء الأولى من الكلمات، حيث يميز محرر النصوص الجزء المدخل من اسم الصنف أو من اسم الإجراء، ثم يقترح محرر النصوص أسماء ممكنة تضاهي الجزء المدخل، وعندئذ يستطيع المبرمج تكملة اسم الصنف أو الإجراء من القائمة المقترحة.

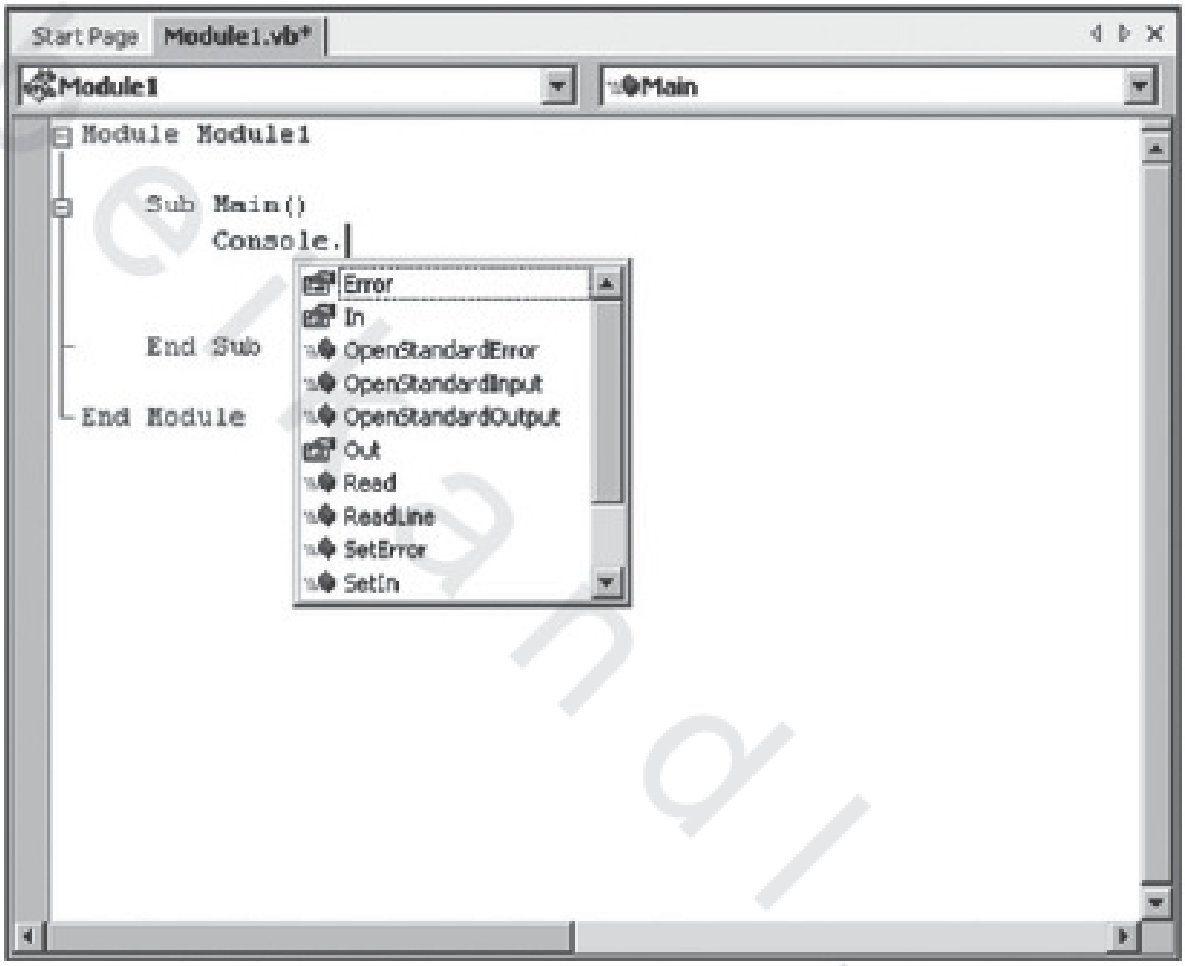
ومثل العديد من محررات النصوص، فإن هذا المحرر يُمكّن المبرمج من إصدار أوامر عن طريق استخدام المفاتيح المختصرة (Shortcut keys) مثل اختيار نص لتنفيذ عمليات النسخ، والقص، واللصق، والبحث عن نص، ولتعديل محتويات الملف البرمجي Modul1.vb اتبع الخطوات التالية:

١- ضع مؤشر الكتابة عند نهاية السطر "Sub main ( )"، ثم اضغط على مفتاح الإدخال الذي يؤدي على إدراج سطر فارغ (لاحظ أن مؤشر الكتابة انتقل بشكل تلقائي أسفل كلمة Main).

٢- اكتب هذه الفقرة "Console." وتأكد أن تكتب نقطة في نهاية الفقرة.

٣- عند كتابة النقطة في نهاية الكلمة Console تظهر قائمة فرعية (انظر الشكل رقم ٢.١٠) تحتوي على جميع الإجراءات (Methods) والخصائص (Properties) للصنف Console الذي سوف تتعرف عليه لاحقاً. والآن يجب عليك فقط فهم أن اختيارك لأحد هذه العناصر سوف يكمل هذا الأمر (هذا ما يسمى بخاصية IntelliSense).

لاحظ أنك تستطيع أيضاً الضغط على كل من مفتاح Ctrl والمفتاح I في آنٍ واحد لكي تعرض عناصر أي صنف أثناء كتابته.

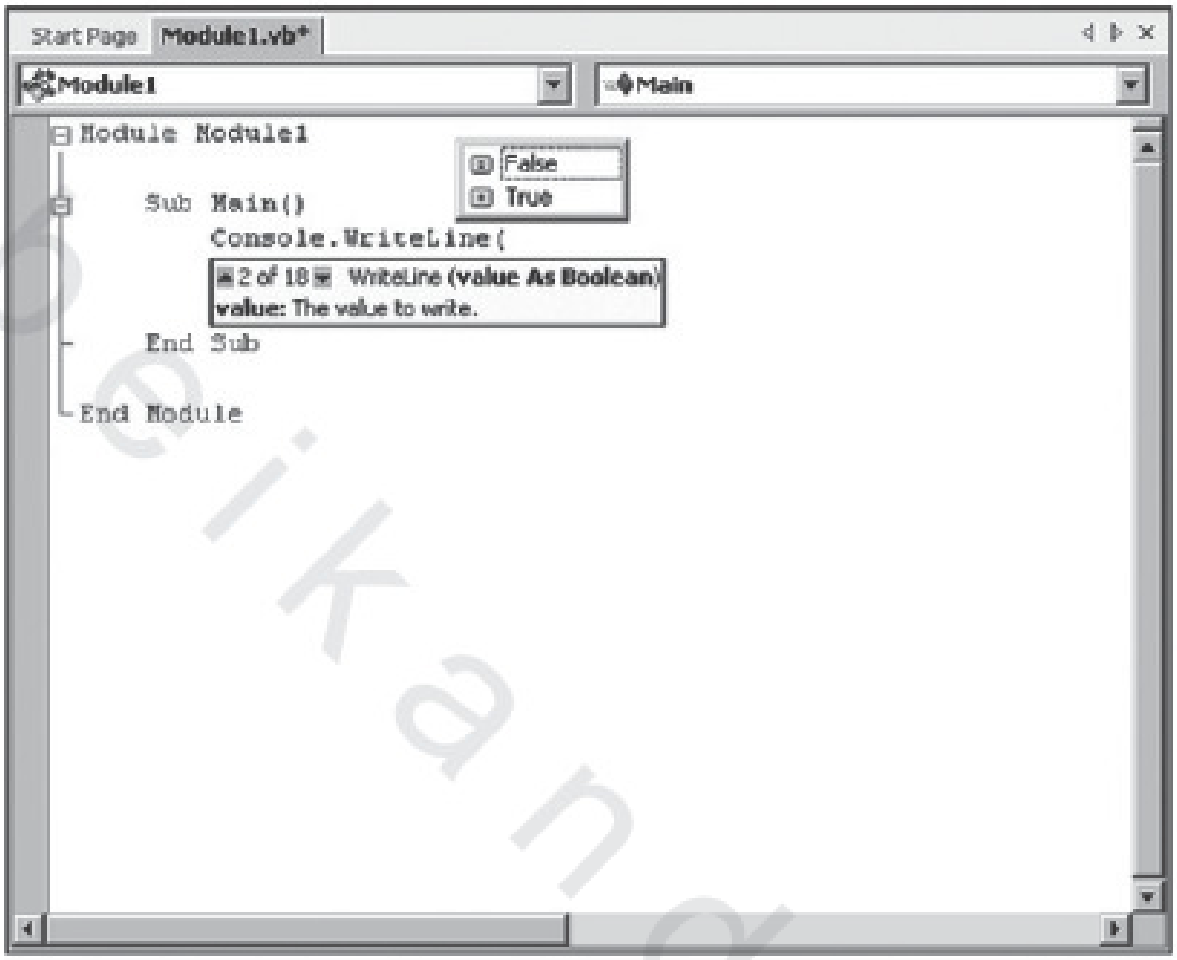


الشكل رقم (٢,١٠). إمكانية كتابة الأوامر تلقائياً (IntelliSense).

٤- تصفح القائمة المنسدلة لأسفل حتى ترى الإجراء ( )WriteLine ثم اضغط عليه ضغطاً مزدوجاً، فعندئذ تلاحظ أن هذا الإجراء تم إلحاقه مباشرة بالعبارة "Console.".

٥- اكتب العبارة "Hello World Wide Web" داخل أقواس الإجراء ( )WriteLine، وستظهر لك نوافذ ورسائل أثناء كتابة العبارة تعرض المساعدة لتكملة الأمر كما واضح في الشكل رقم (٢,١١)، وسوف نتعرض لهذه الرسائل والنوافذ لاحقاً في هذا الكتاب.

٦- خزن عملك بواسطة الضغط على زر Save المتواجد داخل مسطرة الأدوات العلوية.



الشكل رقم (٢,١١). إظهار رسائل ألناء كتابة الأوامر.

تعديل إعدادات محرر النصوص
Modifying the Text Editor Settings

يمكنك عزيزي القارئ تعديل إعدادات محرر النصوص ليتوافق مع احتياجاتك، فعلى سبيل المثال ربما تحتاج لتغيير نوع الخط أو حجم الخط أو حتى لون النص داخل نافذة محرر النصوص. ولتغيير حجم خط نص شفرة البرنامج اتبع التالي:

١- اختر القائمة Tools، ثم اختر الأمر Options، فعندئذ سيظهر الصندوق الحواري Options.

٢- اختر المجلد Environment إذا لم يكن مختاراً بالفعل.

٣- اختر المجلد Environment مرة ثانية لظهور محتوياته إذا لم تكن ظاهرة بالفعل.

٤- اختر الخيار "Fonts and Colors".

٥- اختر Text Editor (إذا كان ضرورياً) من إعدادات Show داخل القائمة.

٦- من القائمة Size، اختر ١٢ ثم اضغط على زر OK، فعندئذ سيظهر حجم الخط داخل نافذة محرر شفرة البرنامج إلى حجم ١٢ نقطة.

غالباً ما يحتاج المبرمج أن يظهر أرقام سطور البرنامج ليسهل عملية الإشارة إلى خط محدد داخل البرنامج، ولكي تظهر أرقام سطور شفرة البرنامج نتبع الآتي:

١- اختر القائمة Tools من القائمة الرئيسية، ثم اختر الأمر Options وسوف يظهر الصندوق الحواري Options.

٢- اختر المجلد Text Editor إذا لم يكن مختاراً بالفعل.

٣- اختر المجلد Text Editor (إذا كان ضرورياً مرة أخرى) لظهور محتوياته، ثم اضغط على المجلد الفرعي Basic الذي يظهر أسفل المجلد Text Editor.

٤- أشّر على الخيار Line Numbers ثم اضغط على زر OK. الآن تحتوي شفرة البرنامج داخل نافذة محرر النصوص على أرقام السطور.

عزيزي القارئ يمكنك ضبط العديد من الإعدادات داخل بيئة تطوير مايكروسوفت متبعاً نفس هذا الأسلوب.

لاحظ أن جميع أشكال هذا الفصل تحتوي على الإعدادات الافتراضية لمحرر النصوص (حجم الخط يساوي ١٠ نقاط، كما لا تظهر أرقام السطور).

إعادة تسمية ملف Module1.vb

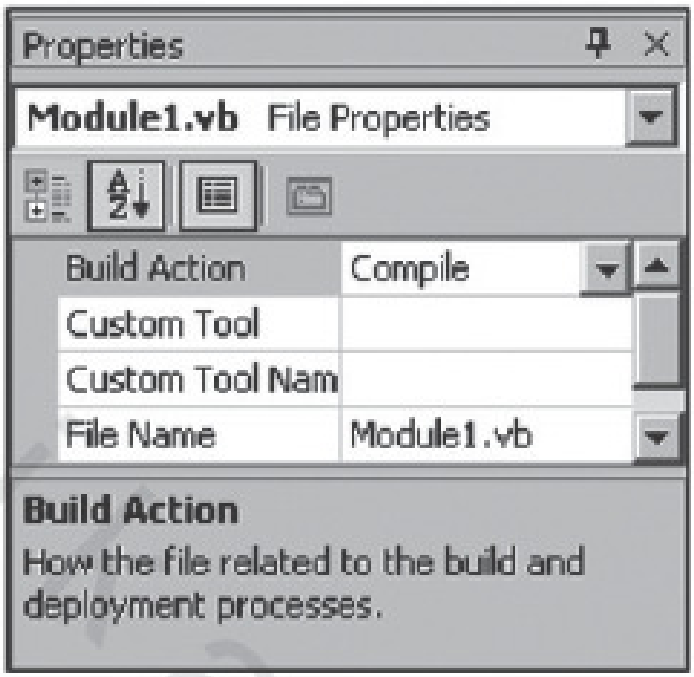
Renaming Module1.vb

إنه من المفضل والشائع عزيزي القارئ أن تسند أسماء ذات معنى لبرامجك. تذكر أن الأسماء الافتراضية للغة VB .NET هي Module1.vb وهكذا، ولكي تعيد تسمية هذا الملف اتبع التالي:

١- اختر الملف Module1.vb من نافذة متصفح الحلول. عندئذ ستظهر صفات الملف Module1.vb داخل نافذة الخصائص (انظر الشكل رقم ٢,١٢).

٢- تصفح نافذة الخصائص للوصول إلى الصفة "File Name"، ثم احذف الاسم الافتراضي Module1.vb (داخل صندوق النص المجاور للصفة)، ثم اكتب HelloWorldWideWeb.vb ثم اضغط مفتاح الإدخال، عندئذ سيظهر اسم الملف كما هو واضح في الشكل رقم (٢,١٣).

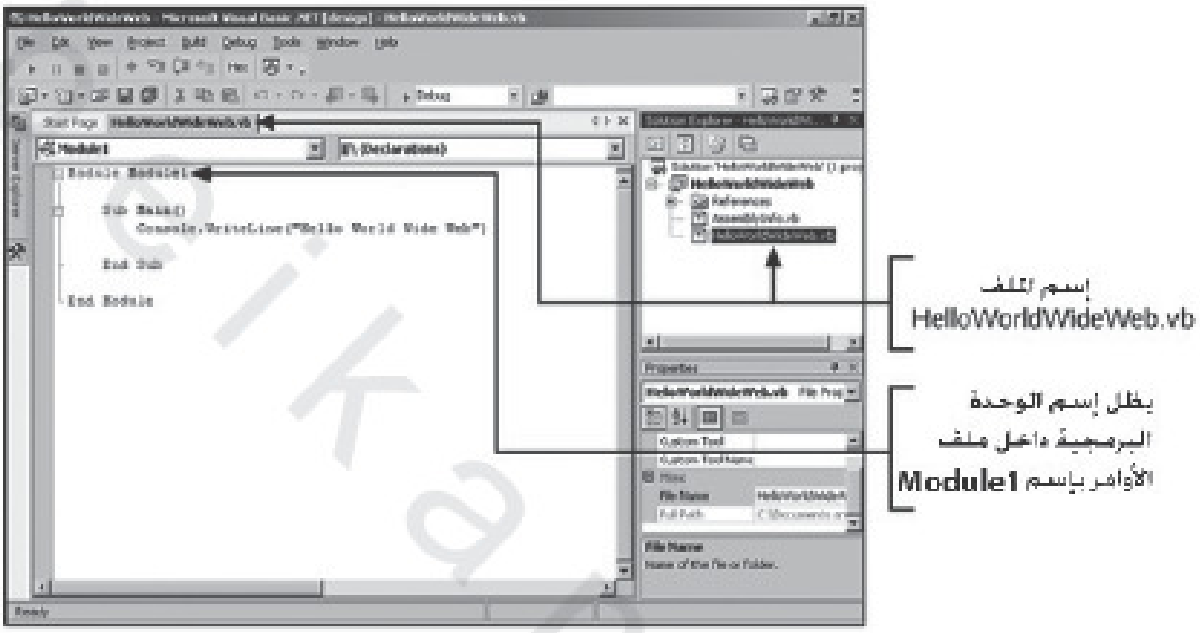
لاحظ أن هناك طريقة أخرى لتغيير اسم الملف وهي كالتالي: اضغط بالزر الأيمن للفأرة على اسم الملف في نافذة متصفح الحلول، ثم اختر الأمر Rename لكي تغير اسم الملف مباشرة.



الشكل رقم (٢,١٢). خصائص ملف Module1.vb.

الشكل رقم (٢,١٣). إعادة تسمية الملف Module1.vb.

٣- سوف ترى أن اسم الملف تغير إلى HelloWorldWideWeb.vb في كل من نافذة الخصائص ومتصفح الحلول، وتغير أيضاً عنوان نافذة البرنامج إلى HelloWorldWideWeb.vb (انظر الشكل رقم ٢.١٤).

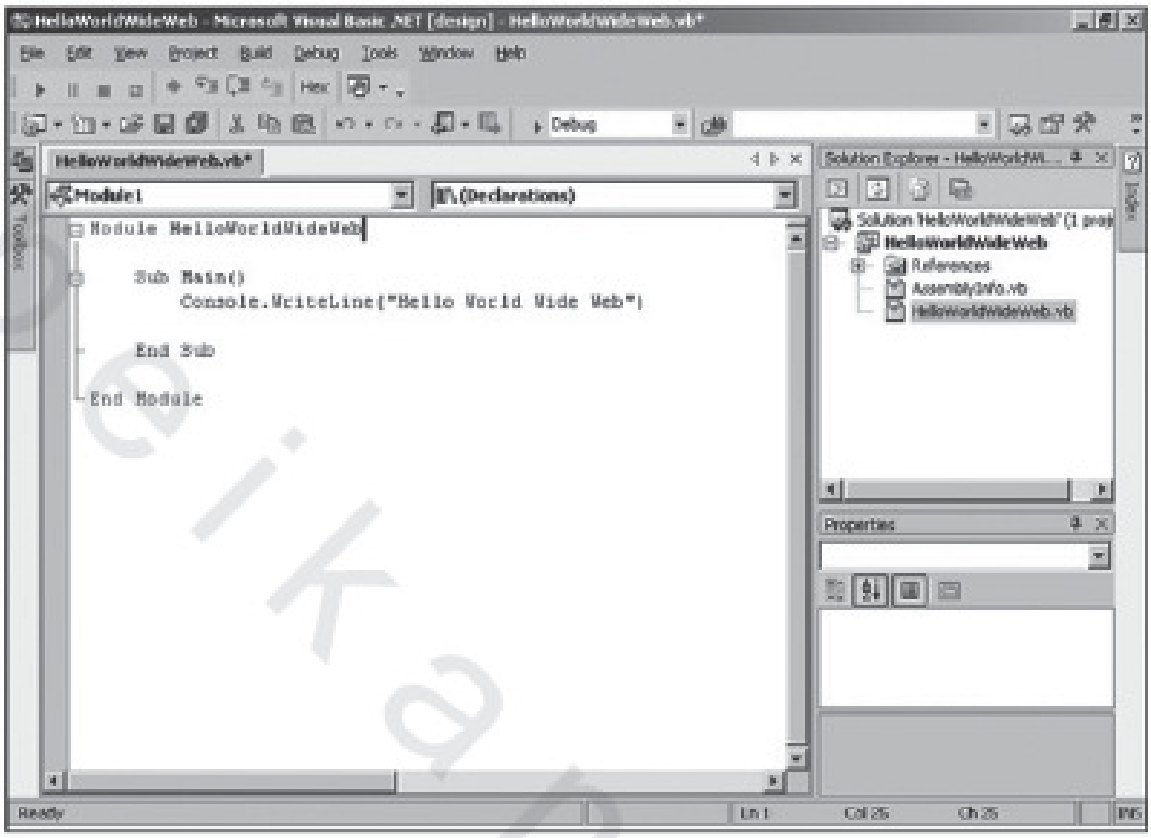


الشكل رقم (٢.١٤). بيئة التطوير المتكاملة بعد تغيير اسم الملف Module1.vb.

٤- يجب أن تلاحظ أن تغيير اسم الملف لا يؤثر على محتويات شفرة البرنامج داخل نافذة HelloWorldWideWeb.vb، وما زال اسم الوحدة البرمجية Module1 كما هو لم يتغير، وبالرغم أنه ليس من المطلوب تغيير اسم الوحدة البرمجية، ولكن من المفضل برمجياً أن تعطي له اسماً ذات معنى داخل شفرة البرنامج.

٥- ولتغيير اسم الوحدة البرمجية، يتم الضغط على نافذة شفرة البرنامج والذهاب إلى السطر الأول، ثم حذف الاسم القديم Module1، ثم كتابة الاسم الجديد HelloWorldWideWeb كما هو واضح في الشكل رقم (٢.١٥).

٦- الضغط على زر Save المتواجد في سطرة الأدوات لحفظ التغييرات التي أجريتها على الملف HelloWorldWideWeb.vb.



الشكل رقم (٢,١٥)- إعادة تسمية الوحدة Module1 إلى HelloWorldWideWeb.

إعداد كائن البداية
Setting the Startup Object

لقد وصلنا الآن إلى آخر مرحلة قبل تنفيذ البرنامج وهي مرحلة ترجمة البرنامج، ولكن يجب إعداد كائن البداية (Startup Object) داخل خصائص المشروع الذي يمثل نقطة بداية تنفيذ البرنامج الخاص بك، وبما أننا قمنا بتغيير اسم الوحدة البرمجية، فيجب من تغييرها داخل خصائص المشروع، ولعمل ذلك اتبع التالي:

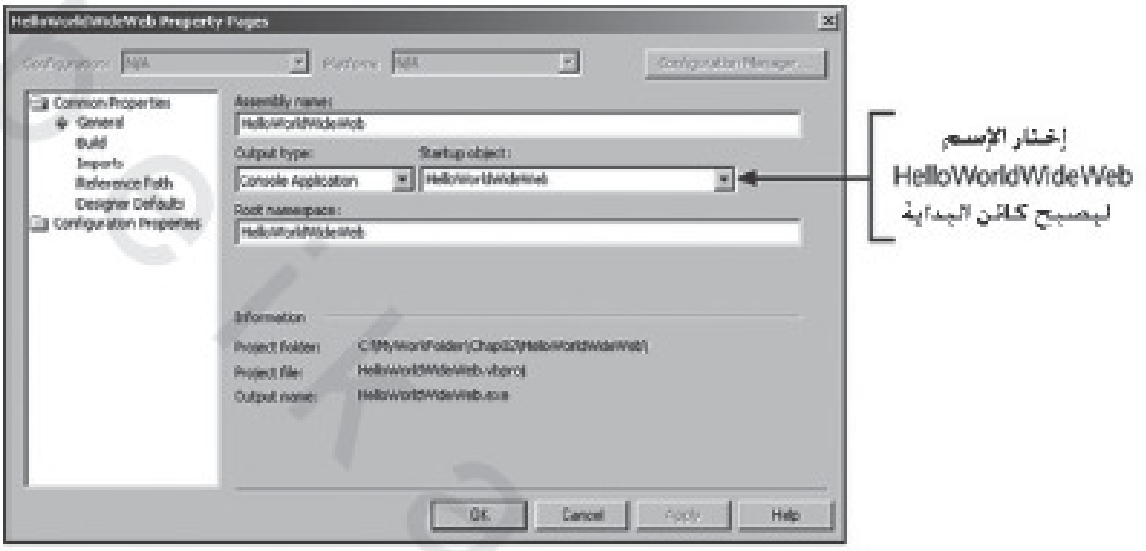
١- اضغط بالزر الأيمن للفأرة على رمز المشروع HelloWorldWideWeb داخل نافذة متصفح الحلول.

٢- اختر الأمر Properties من القائمة الموضحة، وسيتم فتح صندوق حواري بعنوان HelloWorldWideWeb Property Pages كما هو واضح في الشكل رقم ٢,١٦.

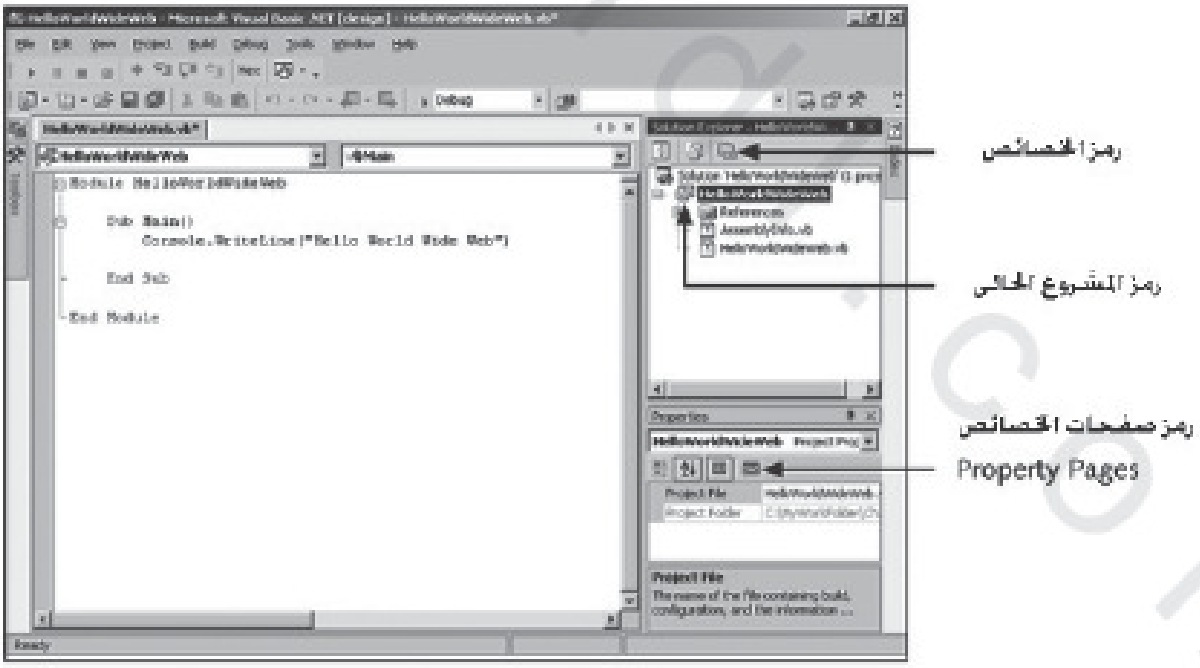
٣- اختر HelloWorldWideWeb من قائمة Start Object ثم اضغط على زر OK.

لاحظ أن هناك أسلوباً آخر لفتح الصندوق الحواري HelloWorldWideWeb Property Pages وهي كالتالي: اختر المشروع من متصفح الحلول، ثم اضغط على رمز Properties المتواجد داخل متصفح الحلول أو اضغط على زر Properties المتواجد داخل نافذة الخصائص (انظر الشكل رقم ٢,١٧).

والآن يأتي دور آخر مرحلة في تطوير البرنامج وهي مرحلة ترجمة البرنامج ، ومن ثم تنفيذ أول برنامج لك بلغة فيجوال بيسك .نت.



الشكل رقم (٢,١٦). نافذة صفات المشروع HelloWorldWideWeb.



الشكل رقم (٢,١٧). أسلوب آخر لإظهار صفات المشروع HelloWorldWideWeb.

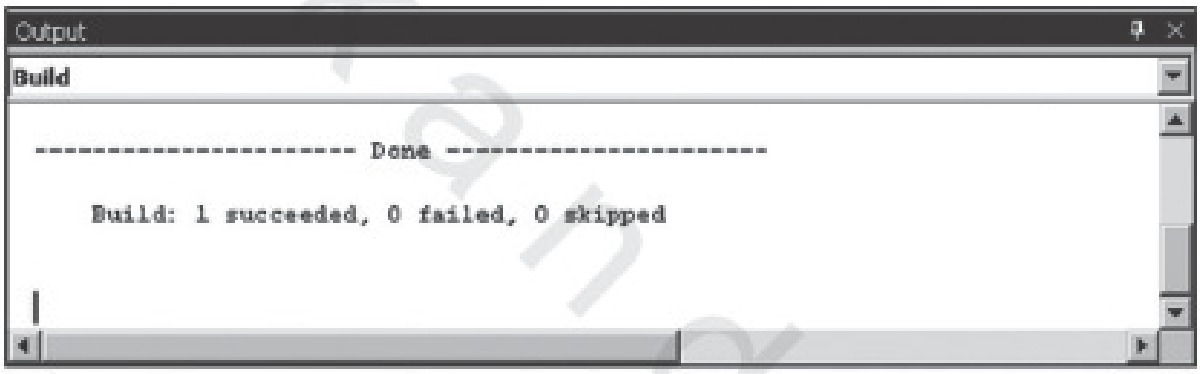
ترجمة برامج لغة فيجوال بيسك .نت وتنفيذها

Compiling and Executing a VB .NET Program

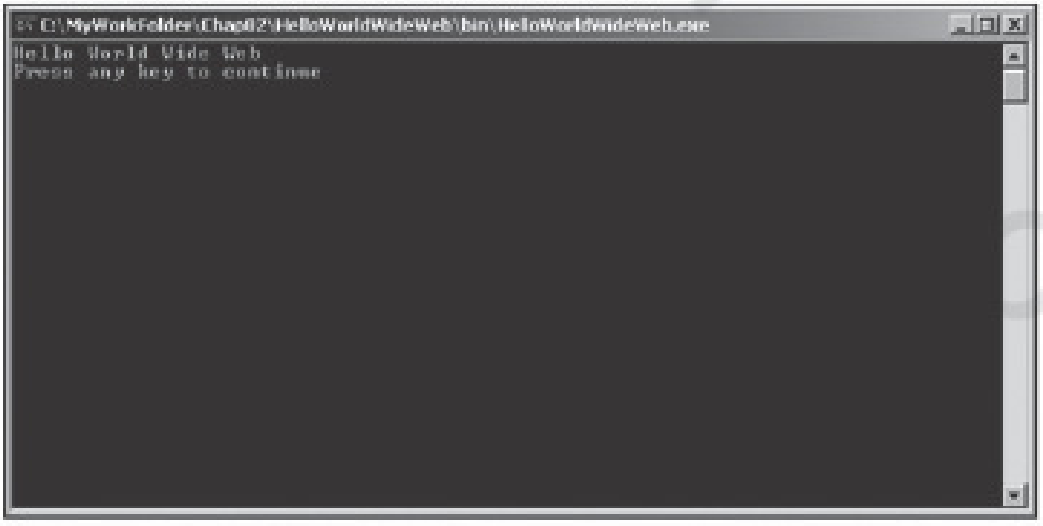
يمكنك عزيزي القارئ ترجمة برنامجك وتنفيذه مستخدماً أكثر من طريقة حيث يمكنك استخدام أوامر القائمة Build وأوامر القائمة Debug، أو استخدام مسطرة الأدوات، أو استخدام المفاتيح السريعة، ولكن دعنا الآن نستخدم أوامر القائمة Debug. لترجمة برنامجك وتنفيذه اتبع الآتي:

١- اختر القائمة Debug، ثم اختر الأمر Start Without Debugging.

٢- إذا لم توجد أخطاء داخل البرنامج ستظهر نافذة الإخراج في الجزء السفلي من الشاشة لتعلمك أنه قد تمت الترجمة بنجاح (انظر الشكل رقم ٢,١٨). بعد ذلك ستظهر نافذة وتعرض لك رسالة "Hello World Wide Web" كما هو واضح في الشكل رقم (٢,١٩).



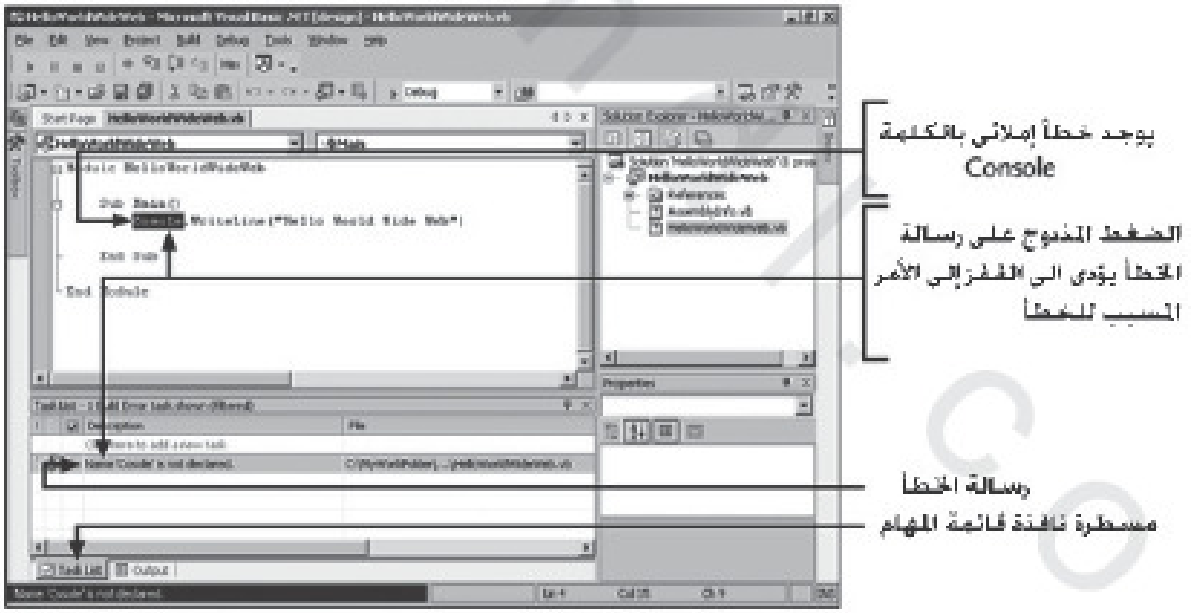
الشكل رقم (٢,١٨). نافذة المخرجات.



الشكل رقم (٢,١٩). نافذة مخرجات البرنامج: رسالة "Hello World Wide Web".

لاحظ أنه إذا كان هناك خطأ داخل البرنامج ستظهر رسالة تعلمك بوجود أخطاء وتطلب منك هل تريد الاستمرار في ترجمة البرنامج وبنائه أم لا؟ (الشكل رقم ٢,٢٠). تظهر رسالة الخطأ داخل قائمة المهام (Task List pane) في الجزء السفلي من الشاشة كما هو واضح في الشكل رقم (٢,٢١). يمكنك القفز مباشرة إلى الخطأ داخل شفرة البرنامج وذلك بالضغط المزدوج داخل قائمة المهام.

الشكل رقم (٢,٢٠). نافذة وجود خطأ عند ترجمة البرنامج.



الشكل رقم (٢,٢١). رسائل الخطأ داخل نافذة قائمة المهام.

٣- أغلق نافذة الرسالة "Hello World Wide Web" بواسطة الضغط على زر الإغلاق "X".

٤- أغلق نافذة الإخراج.

استخدام مصمم النماذج المرئي
Using the Visual Form Designer

سوف تتعلم كيفية تطوير تطبيقات ويندوز بالإضافة إلى تطوير برامج Console. تعمل تطبيقات الويندوز مع بيئة نظام التشغيل ويندوز التي تطلب تصميم واجهة استخدام رسومية (GUI)، وبمعنى آخر تطلب تصميم نماذج (Forms) مستخدماً مصمم النماذج المرئي (Visual Form Designer) ومحرر النصوص.

يسمح لك مصمم النماذج المرئي اختيار رموز مكونات النماذج المختلفة مثل الأزرار (Buttons)، وصناديق النصوص (Text Boxes) والقوائم (Lists)، ثم وضعها وتنسيقها على النموذج لتصميمه. تنتج لغة فيجوال بيسك .نت الأوامر البرمجية المطلوبة لبناء النموذج أثناء مرحلة التصميم الذي تراه أمامك، وعندما تعيد ترتيب المكونات على النموذج أو تغيير صفات هذه المكونات ينعكس ذلك بشكل تلقائي داخل شفرة البرنامج. يمكن مصمم النماذج المرئي المبرمج من إنشاء نماذج تطبيقات الويندوز بسرعة فائقة مثل تلك الموضحة في الشكل رقم (٢,٢٢).

الشكل رقم (٢,٢٢). نموذج تطبيق الويندوز.

يطلق على مصمم النماذج اسم "مصمم نماذج ويندوز" (Windows Forms Designer)، ويحتوي صندوق الأدوات (Tool Box) على مكونات مرئية مثل صناديق النصوص (Text Boxes)، والأزرار (Buttons)، والعناوين

(Labels) التي تستخدم لتصميم نماذج الإدخال، كما يمكنك استخدام نافذة الخصائص لضبط صفات هذه المكونات مثل تغيير لون الزر أو تغيير لون الخط داخل صندوق النص، وهكذا. عندما يتم تنظيم هذه العناصر ومعالجتها، تنشئ لغة VB .NET الشفرة المطلوبة لعمل النموذج. سوف تتعلم عزيزي القارئ كيف تطور تطبيقات ويندوز بسيطة مستخدماً مصمم نماذج الويندوز خلال هذا الفصل، وسوف تتعلم الكثير عن تطوير تطبيقات ويندوز أكثر تعقيداً، وكذلك مهام مصمم نماذج الويندوز داخل الفصل العاشر.

إنشاء تطبيق ويندوز
Creating a Windows Application

سوف تتعلم الآن إنشاء مشروع بلغة VB .NET من النوع Windows والذي من خلاله سوف نستخدم مصمم النماذج المرئي لإنشاء نموذج البرنامج. تذكر عند تطوير مشروع جديد يجب عليك اختيار نوع المشروع ونموذجه واسمه ومسار تخزينه. ولإنشاء تطبيق ويندوز اتبع الخطوات التالية:

١- اختر مسطرة Start Page من نافذة برنامج فيجوال ستوديو .نت.

٢- اختر الرابط Get Started إذا لم يكن مختاراً.

٣- اختر الأمر New Project، فعندئذ سوف يظهر الصندوق الحواري New Project.

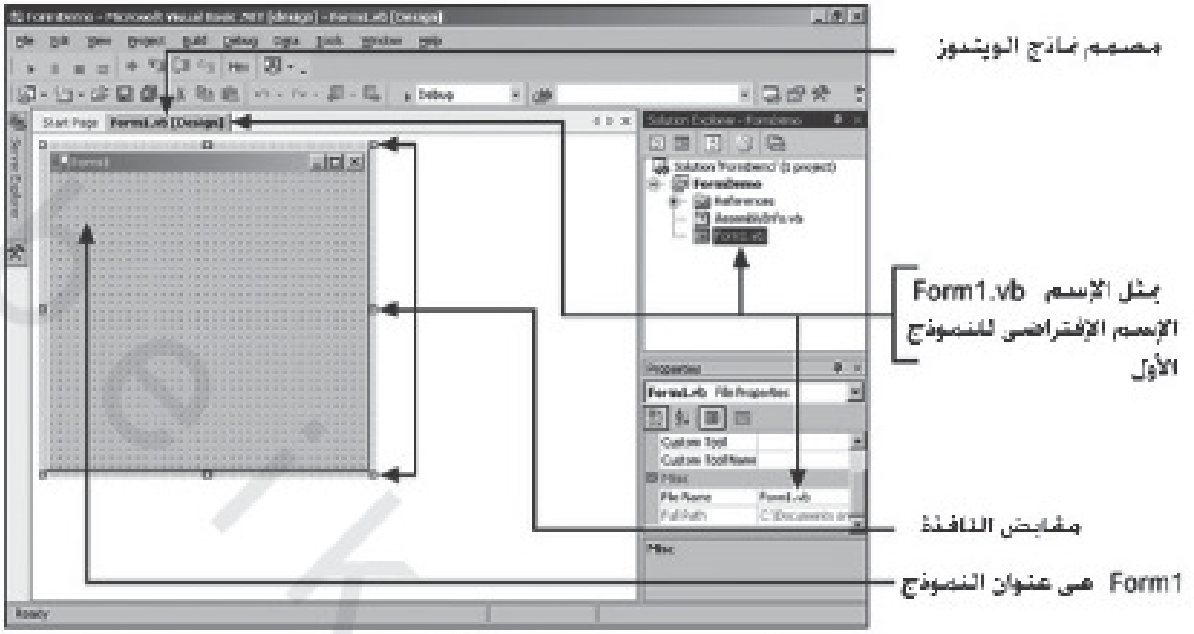
٤- اختر المجلد Visual Basic Properties من قائمة Projects Types، ثم اختر الخيار Windows Application إذا لم يكن مختاراً.

٥- إذا كان هذا هو أول تطبيق ويندوز لك سوف يحتوي صندوق النص "Name" على اسم المشروع الافتراضي "WindowsApplication1". احذف هذا الاسم ثم اكتب اسم المشروع الجديد "FormDemo".

٦- حدد المجلد الرئيس للمشروع وذلك في صندوق النص "Location" حيث يمكنك قبول المسار الافتراضي أو الضغط على زر "Browse" لاختيار مسار بديل.

٧- اختر الزر OK، عندئذ سيظهر مصمم النماذج ويندوز كما هو واضح في الشكل رقم (٢,٢٣) في نافذة منفصلة بعنوان "Form1.vb [Design]" حيث يعطي مصمم نماذج ويندوز أسماء افتراضية للنماذج (الاسم Form1.vb، والاسم Form2.vb، وهكذا).

لاحظ أن نافذة متصفح الحلول تحتوي على معلومات حول المشروع الحالي وحول عناصره، وأن نافذة الخصائص تظهر خصائص الملف Form1.vb، وأن عنوان النموذج يأخذ نفس الاسم الرئيس للملف "Form1"، ولاحظ أيضاً أن خلفية النموذج تحتوي على شبكة نقطية التي تساعد المبرمج على تصميم مكونات النموذج وتنسيقها حيث تظهر هذه الشبكة أثناء التصميم فقط وتختفي أثناء تنفيذ البرنامج.



الشكل رقم (٢,٢٣). مصمم نماذج الويندوز.

يوجد على حدود النموذج مربعات بيضاء اللون يطلق عليها مقابض (Handles) والتي تمكن المبرمج من ضبط حجم النموذج، وعندما يتحرك مؤشر الفأرة على أحد هذه المقابض يتغير شكله إلى سهم مزدوج، ومن ثم يمكن الضغط على الزر الأيسر للفأرة وإعادة ضبط حجم النموذج.

والآن سوف يتم تنفيذ الخطوات المطلوبة لتصميم النموذج ليظهر في وضعه النهائي كما هو واضح في الشكل رقم (٢,٢٢). يتم ضبط الحجم أولاً، ثم لون الخلفية، ثم عنوان النموذج. وبعد ذلك يتم إضافة رسالة ترحيبية وزر إلى النموذج. وأخيراً سوف تكمل الخطوات المطلوبة لإظهار رسالة شكر عند تشغيل البرنامج والضغط على زر النموذج (كما هو واضح في الشكل رقم ٢,٢٤).

الشكل رقم (٢,٢٤). رسالة شكر نتيجة الضغط على زر النموذج السابق.

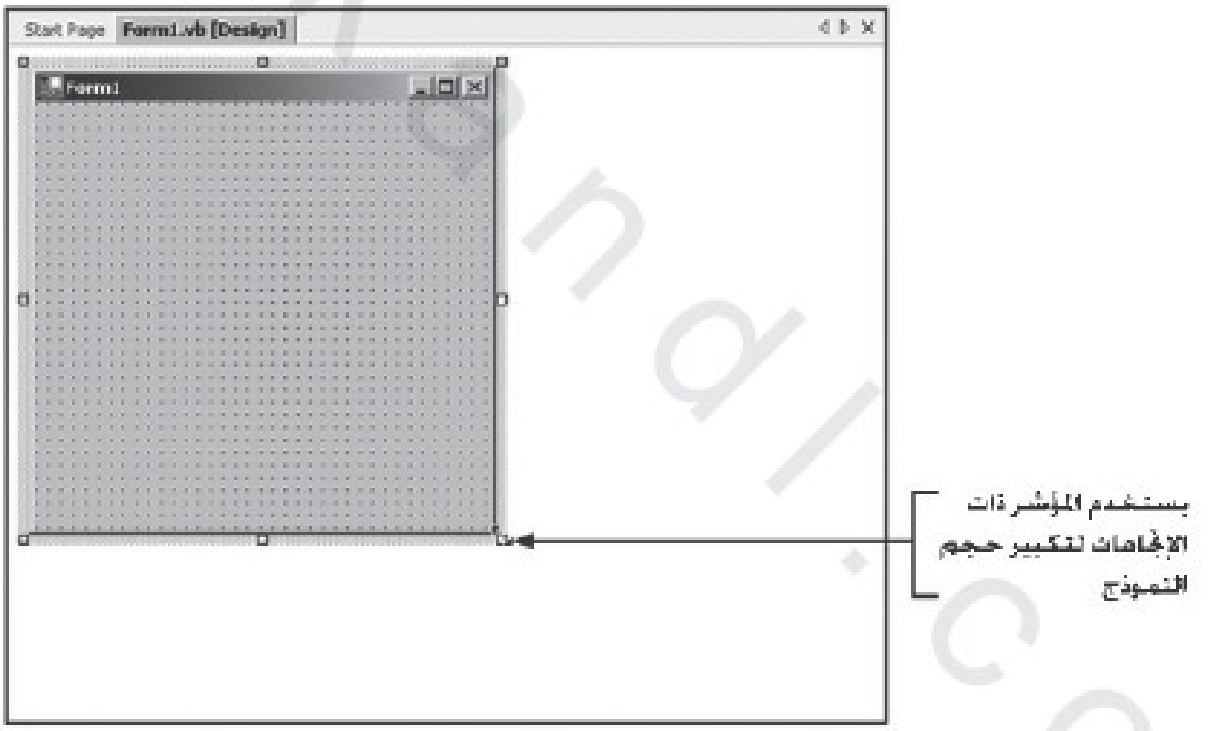
تعديل مظهر النموذج

Customizing the Appearance of a Form

توجد العديد من السمات التي يمكن تغييرها لتغيير مظهر النموذج، وسوف نتطرق في هذا الفصل إلى كل من تعديل سمة حجم النموذج ولون خلفيته وعنوانه. أما في الفصل العاشر فسوف نتعرف على أساليب عديدة لتصميم نماذج شديدة التعقيد. ولتغيير حجم النموذج اتبع التالي:

١- اضغط بالزر الأيسر للفأرة في منتصف النموذج لإظهار مقابض النموذج إذا لم تكن ظاهرة بالفعل.

٢- حرك مؤشر الفأرة إلى الركن السفلي الأيمن للنموذج إلى أن يتغير شكل المؤشر إلى مؤشر مزدوج، وعندئذ اضغط المؤشر واسحبه لأسفل وإلى أعلى لتكبير وضبط حجم النموذج كما هو واضح في الشكل رقم (٢,٢٥).



الشكل رقم (٢,٢٥). تكبير النموذج.

ولتغيير كل من سمة عنوان النموذج ولون خلفيته اتبع التالي:

١- تأكد أن النموذج مختار بالفعل، وإذا لم يكن كذلك اضغط بزر الفأرة الأيسر في منتصفه لاختياره.

لاحظ أن علامة اختيار أي عنصر داخل النموذج (أو النموذج نفسه) تعني ظهور مقابض هذا العنصر.

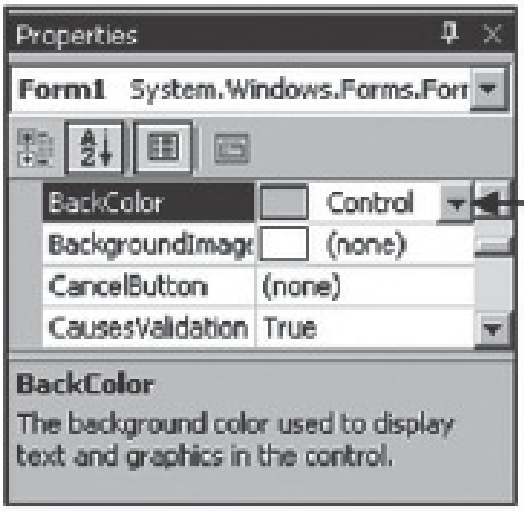
٢- تجول داخل نافذة الخصائص (Properties windows) إلى أن تصل إلى الصفة Text ، عندئذ احذف القيمة الافتراضية Form1 ، ثم اكتب FormDemo واضغط المفتاح Enter كما هو واضح في الشكل رقم (٢,٢٦). سيتغير عنوان النموذج إلى FormDemo.

٣- تجول داخل نافذة الخصائص (Properties windows) إلى أن تصل إلى الصفة BackColor ، عندئذ اضغط عليها ليظهر سهم قائمة منسدلة بجوار القيمة الحالية للصفة كما هو موضح في الشكل رقم (٢,٢٧).

٤- اضغط على سهم القائمة المنسدلة لتنسدل قائمة من الألوان المقسمة إلى ثلاثة مجموعات (System, Web, and Custom) (انظر الشكل رقم ٢,٢٨). اختر المجموعة Custom ، ثم اختر اللون الأبيض (White) ، وعندئذ ستتغير خلفية النموذج إلى اللون الأبيض.

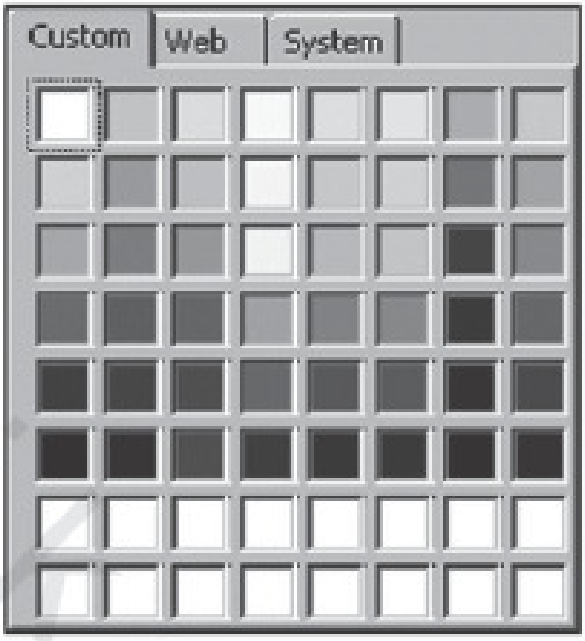
قم بتغيير عنوان النموذج من Form 1 إلى Form Demo

الشكل رقم (٢,٢٦). تغيير عنوان النموذج.



إضغط هنا لتظهر قائمة الألوان

الشكل رقم (٢,٢٧). تغيير خلفية النموذج.



الشكل رقم (٢,٢٨). اختيار اللون من المجموعة Custom.

إضافة المكونات إلى النموذج
Adding Components to a Form

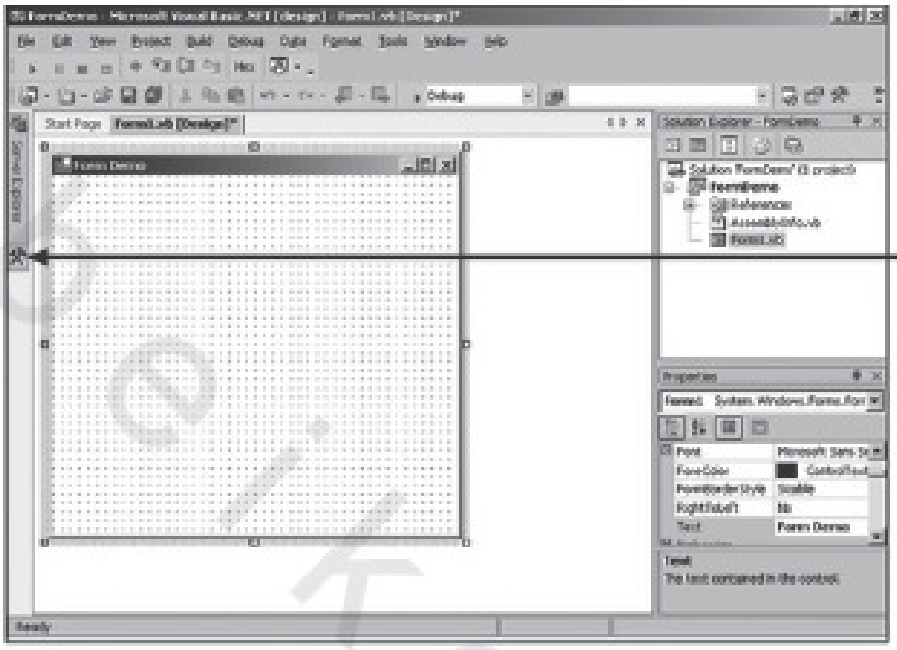
يستخدم صندوق الأدوات (Toolbox) عند إضافة المكونات (Components) إلى النموذج أثناء مرحلة التصميم. سيظهر صندوق الأدوات (Toolbox) كنافذة مخفية على الجانب الأيسر من الشاشة في حال اتباع الخطوات المذكورة سابقاً داخل هذه الوحدة. أما إذا لم يكن صندوق الأدوات (Toolbox) ظاهراً، يتم الضغط على رمزه في مسطرة الأدوات لفتح صندوق الأدوات.

لاحظ أنه لفتح صندوق الأدوات بأسلوب آخر، اختر القائمة الفرعية View من القائمة الرئيسة، ثم اضغط على أمر Toolbox. يمكنك الآن إخفاء نافذة صندوق الأدوات بواسطة الضغط على رمز إخفاءها الآلي (Auto Hide icon) لتظهر كما في الشكل رقم (٢,٢٩).

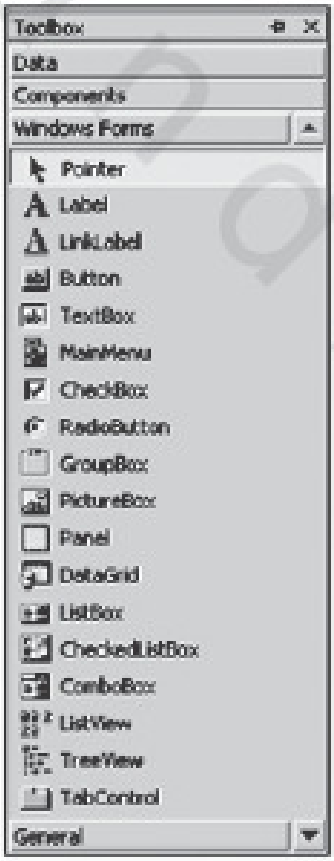
ولإضافة رسالة ترحيبية للنموذج، اتبع الخطوات التالية:

١- ضع مؤشر الفأرة على رمز صندوق الأدوات لتظهر محتوياته، عندئذ يظهر العديد من المكونات مثل العناوين (Labels)، والأزرار (Buttons)، وصناديق النصوص (Text Boxes) كما هو واضح في الشكل رقم (٢,٣٠).

٢- اضغط ضغطاً مزدوجاً على المكون Label لكي تضيف عنواناً إلى النموذج، ثم حرك المؤشر تجاه منتصف الشاشة بعيداً عن صندوق الأدوات لكي يرجع ثانية إلى وضع الإخفاء، وعندئذ سترى المكون Label على النموذج بعنوان افتراضي "Label1".



الشكل رقم (٢,٢٩). إخفاء نافذة صندوق الأدوات.

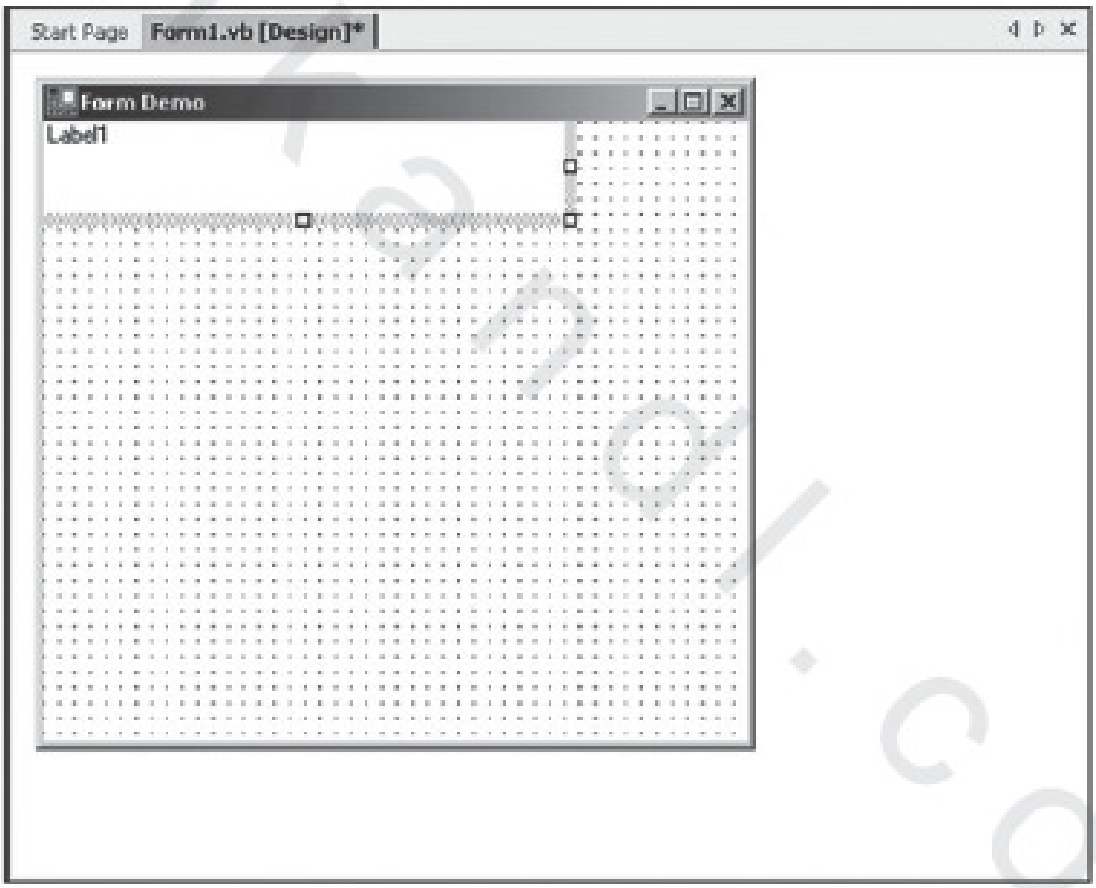


الشكل رقم (٢,٣٠). محتويات نافذة صندوق الأدوات.

٣- اسحب مقبض الركن الأيمن السفلي للمكون Label إلى أسفل وإلى يمين لتكبير حجم العنوان كما هو واضح في الشكل رقم (٢,٣١).

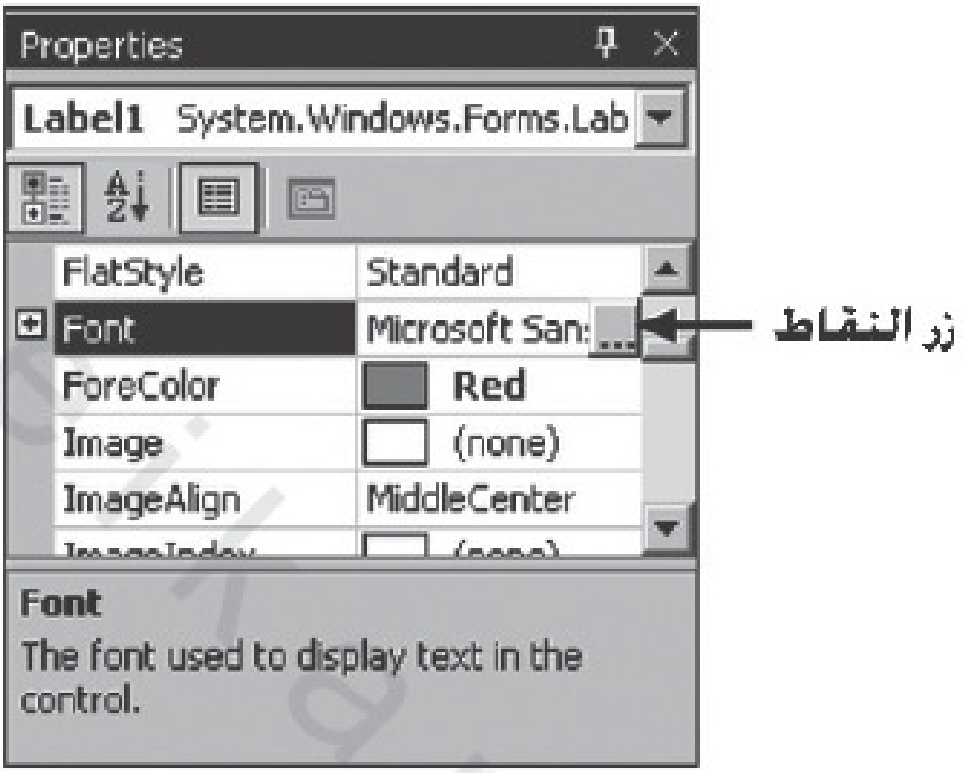
٤- تجول داخل نافذة الخصائص حتى تجد الصفة Text، وعندئذ غير القيمة الافتراضية Label1 إلى القيمة "Welcome to VB .NET" ثم اضغط على المفتاح Enter.

٥- لتغيير لون نص الرسالة، اختر الصفة ForeColor من نافذة الخصائص، وعندئذ سيظهر سهم القائمة المنسدلة بجوار القيمة الحالية للون. اضغط على هذا السهم لترى خيارات الألوان، ثم اختر المجموعة Custom، ثم اختر اللون الأحمر، وسيتغير لون العنوان إلى اللون الأحمر.



الشكل رقم (٢,٣١). تكبير العنصر Label1.

٦- لتغيير نوع خط الرسالة، اختر الصفة Font من نافذة الخصائص، وعندئذ سيظهر زر يحتوي على النقاط (...) بجوار القيمة الافتراضية للصفة Font (انظر الشكل رقم ٢,٣٢).



الشكل رقم (٢,٣٢). العامل مع خاصية Font.

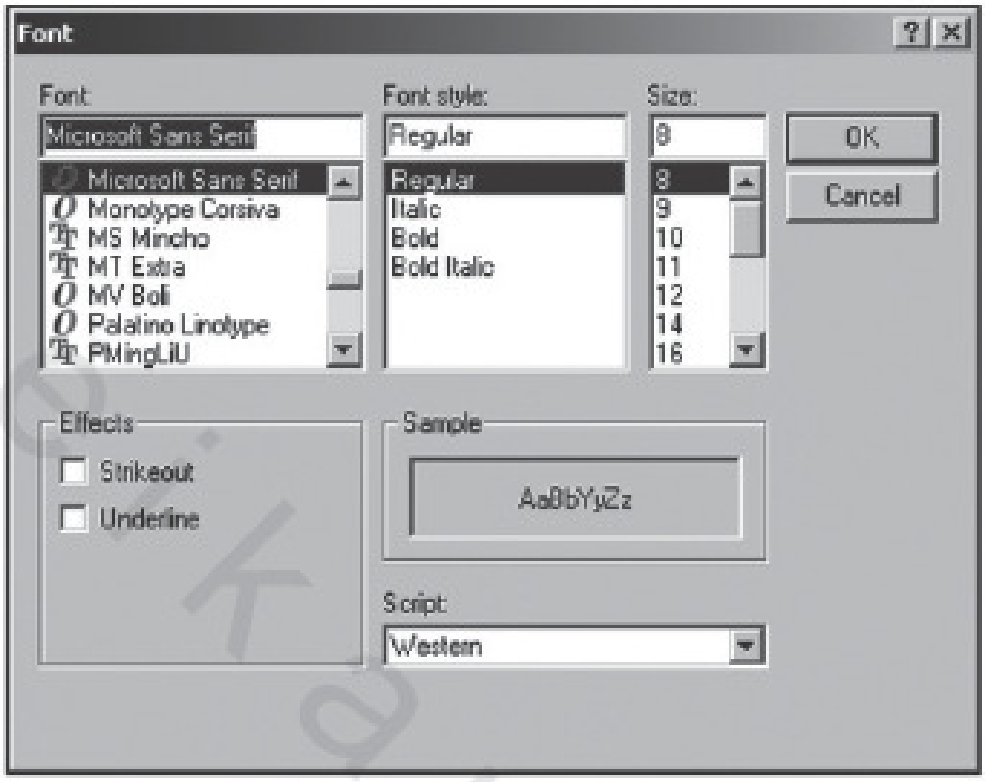
٧- عند الضغط على الزر (...) ellipsis، سيظهر صندوق حواري بعنوان Font كما هو واضح في الشكل رقم (٢,٣٣). اختر نوع الخط Times New Roman من قائمة Font، ثم اختر الحجم ١٨ من نافذة Size. اضغط الزر OK لتطبيق التغييرات المطلوبة وغلق الصندوق الحواري، وعندئذ ستظهر هذه التعديلات على المكون Label.

٨- لضبط نص الرسالة في منتصف مساحة المكون Label، اختر الصفة TextAlign من نافذة الخصائص، ثم اختر سهم القائمة الكائن على يمين الصفة، وستظهر تسعة مستطيلات (انظر الشكل رقم ٢,٣٤) لتمثيل جميع خيارات تنسيق النص الممكنة داخل المكون Label (الركن الأيسر العلوي، والركن الأيمن العلوي، إلخ)، واختر المستطيل الأوسط للإشارة إلى توسيط لنص داخل حجم المكون Label.

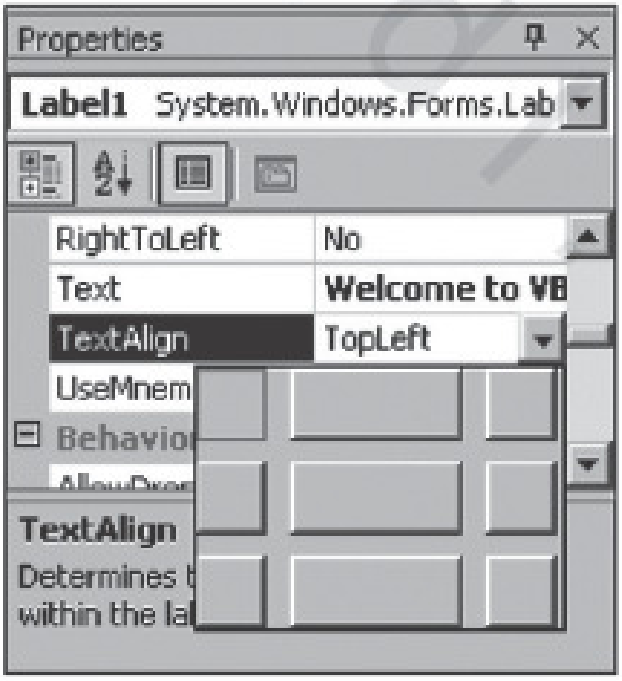
٩- شاور بالفأرة على العنوان وسيتغير مؤشر الفأرة إلى سهم رباعي كما هو واضح في الشكل رقم (٢,٣٥). وعندما يظهر هذا المؤشر تستطيع أن تقوم بسحب العنصر إلى المكان الجديد.

١٠- اسحب العنوان إلى المكان الموضح في الشكل رقم (٢,٣٦).

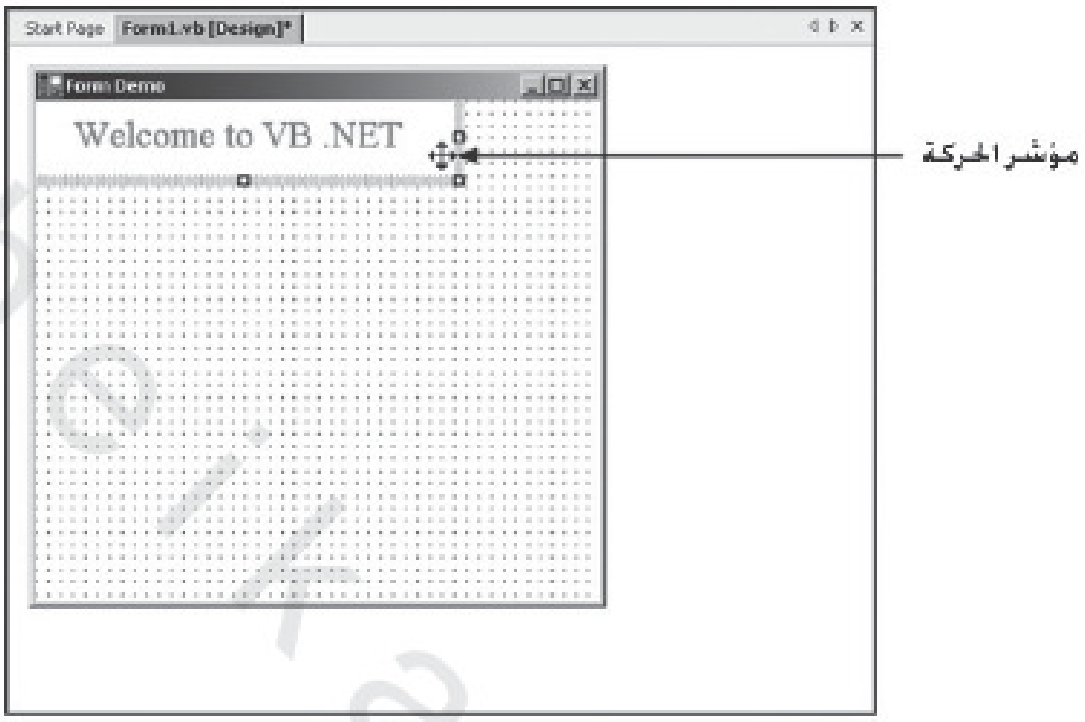
لاحظ أنه لكي تضع العنوان في المنتصف بشكل أفقي قم باختيار العنوان ثم اضغط القائمة Format داخل مسطرة القائمة، ثم شاور على الرمز Center في النموذج، ثم اختر "Horizontally".



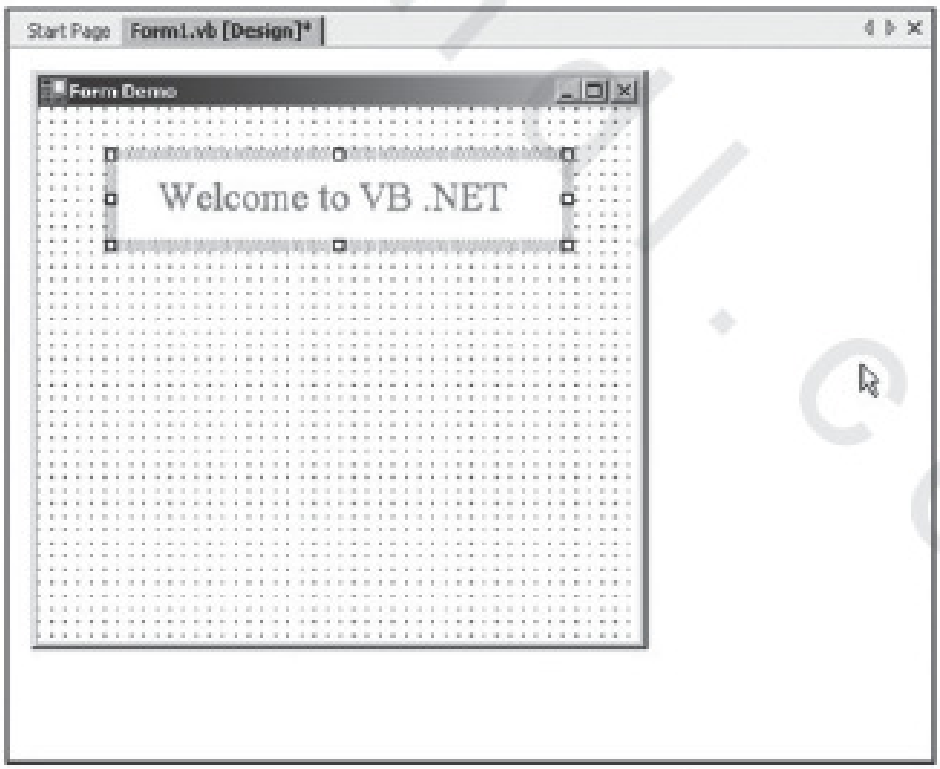
الشكل رقم (٢,٣٣). نافذة تغيير صفات الخط (Font).



الشكل رقم (٢,٣٤). ضبط قيمة الخاصية TextAlign.



الشكل رقم (٢,٣٥). سحب العنوان (Label) إلى موضع جديد.



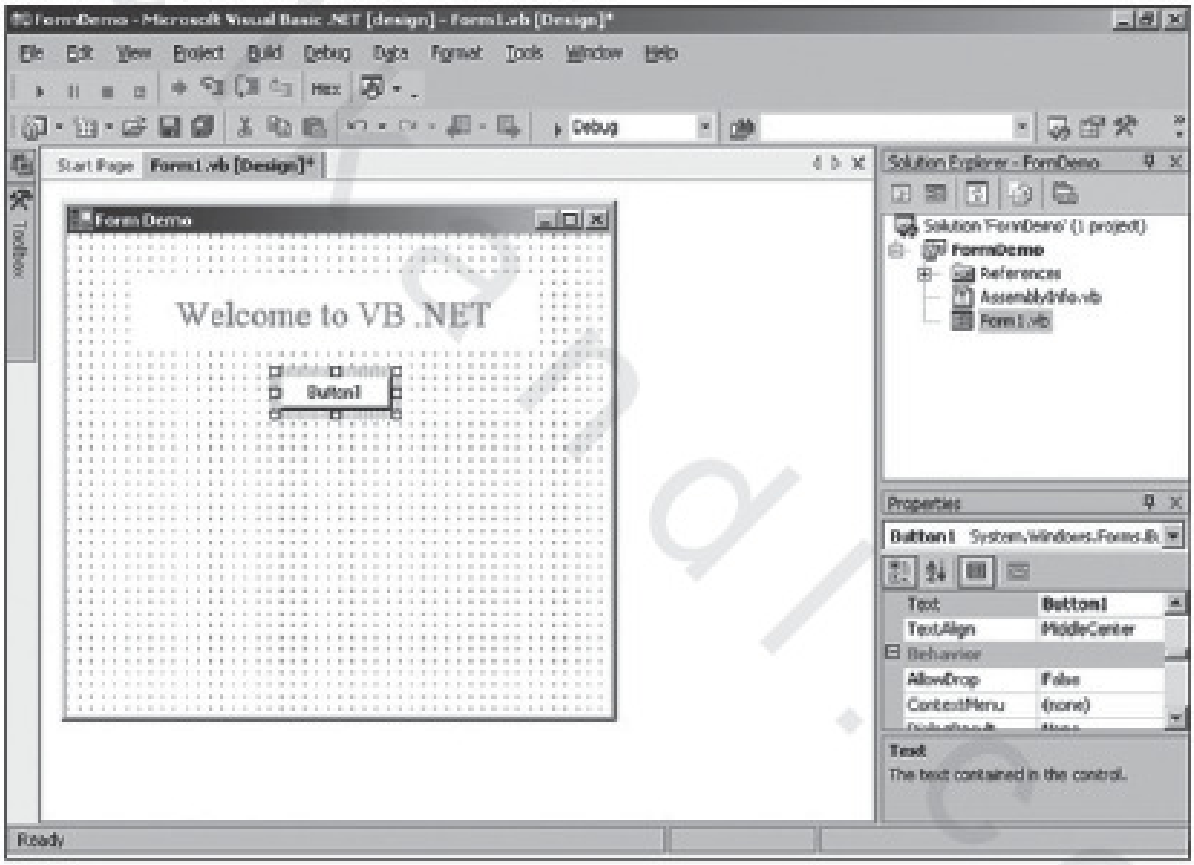
الشكل رقم (٢,٣٦). العنوان (Label) في موضعه الجديد.

لإضافة زر (Button) إلى النموذج، اتبع الخطوات التالية:

١- أشر بالفأرة على مسطرة صندوق الأدوات لإظهارها، ثم اضغط ضغطاً مزدوجاً على المكون Button، ثم حرك مؤشر الفأرة تجاه منتصف الشاشة لكي يعود صندوق الأدوات إلى وضع الإخفاء.

لاحظ أنه لإضافة أي مكون إلى النموذج، يمكنك أيضاً اختيار هذا المكون بالضغط عليه ثم سحبه إلى النموذج.

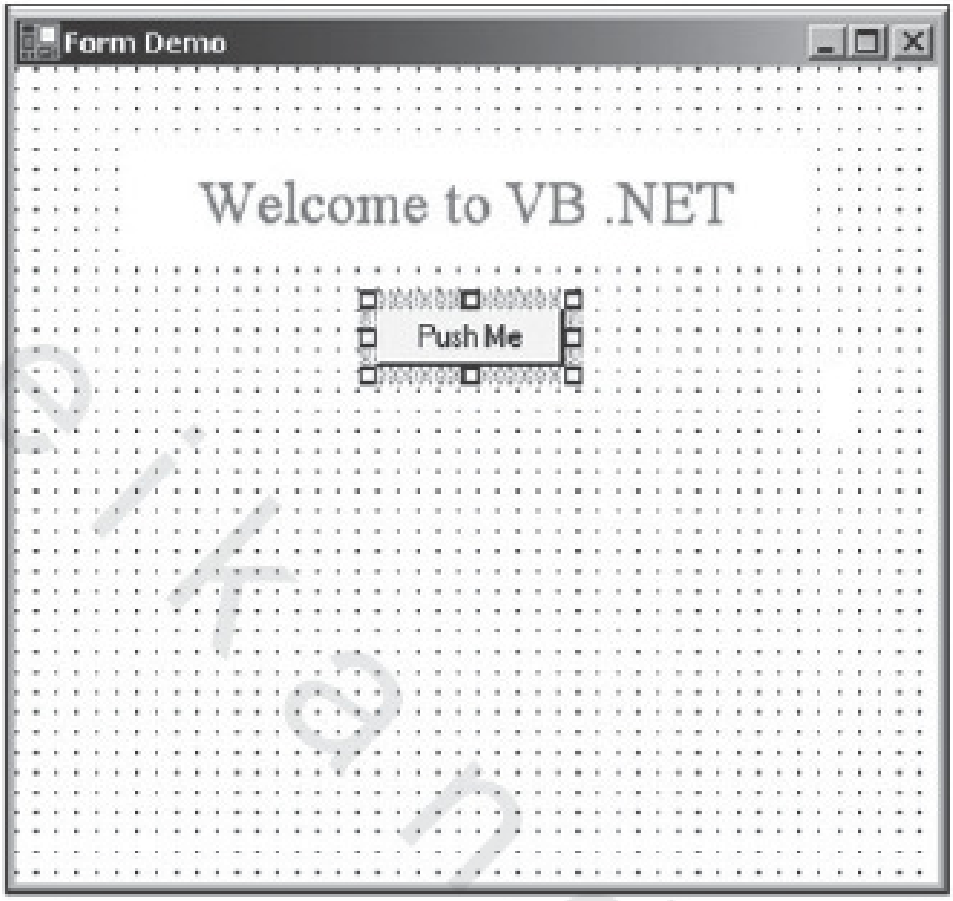
٢- اسحب الزر Button إلى منتصف النموذج ليكون أسفل رسالة الترحيب (انظر الشكل رقم ٢,٣٧).



الشكل رقم (٢,٣٧): إضافة زر جديد إلى النموذج.

٣- بما أن الزر هو المكون المختار الحالي (لاحظ ظهور مقابضه)، تظهر خصائصه الآن داخل نافذة الخصائص. اختر الخاصية Text من نافذة الخصائص، ثم قم بتغيير قيمتها لتصبح "Push Me"، ثم اضغط المفتاح Enter.

٤- اختر الخاصية BackColor من نافذة الخصائص، ثم غير قيمتها إلى اللون الأصفر أو أي لون آخر من اختيارك. والآن اكتمل التصميم المبدئي للنموذج كما هو واضح في الشكل رقم (٢,٣٨).



الشكل رقم (٢,٣٨). إنهاء تصميم النموذج.

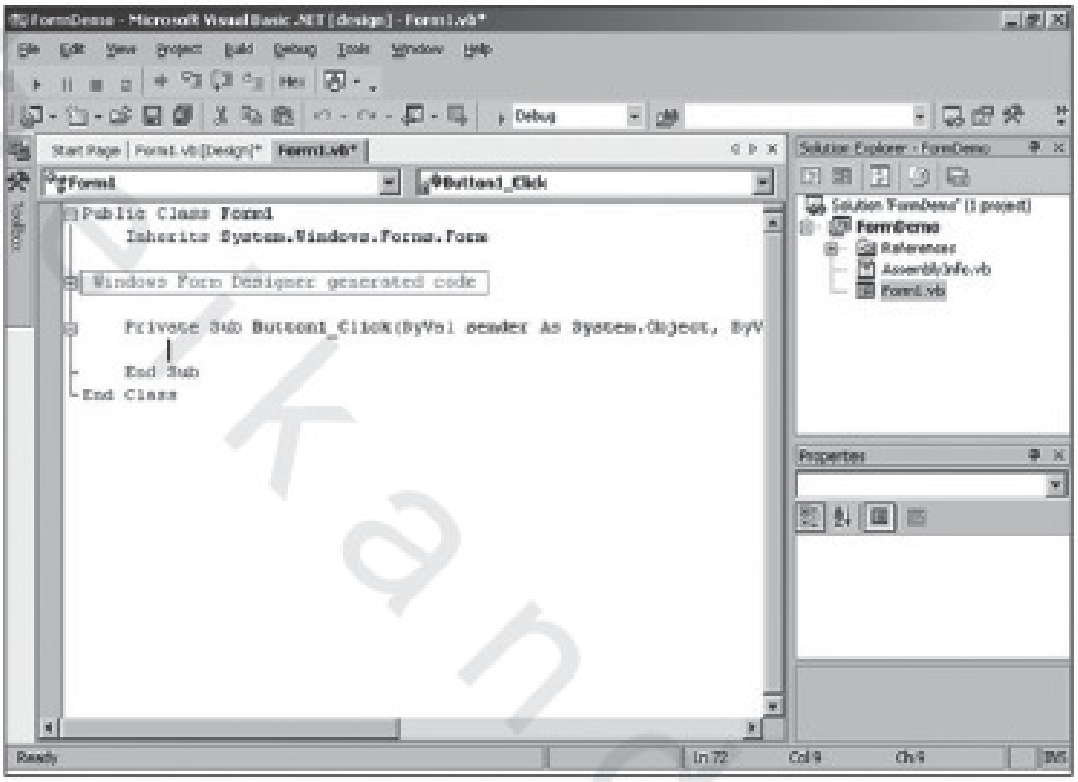
عند تشغيل البرنامج في هذه المرحلة، يظهر النموذج محتوياً على رسالة ترحيبية وزر بعنوان "Push Me"، ولكن عندما تضغط على الزر لن يستجيب البرنامج، وذلك لأننا لم نكتب إلى الآن الأوامر البرمجية التي ستنفذ عند الضغط على الزر، ومن ثم ظهور رسالة شكر للمستخدم. ولإضافة هذه الوظيفة للبرنامج، يجب استخدام محرر النصوص لإضافة الشفرة المطلوبة، ولعمل ذلك اتبع الخطوات التالية:

١- اختر صفحة النموذج Form1.vb [Design] إذا لم تكن مختارة بالفعل.

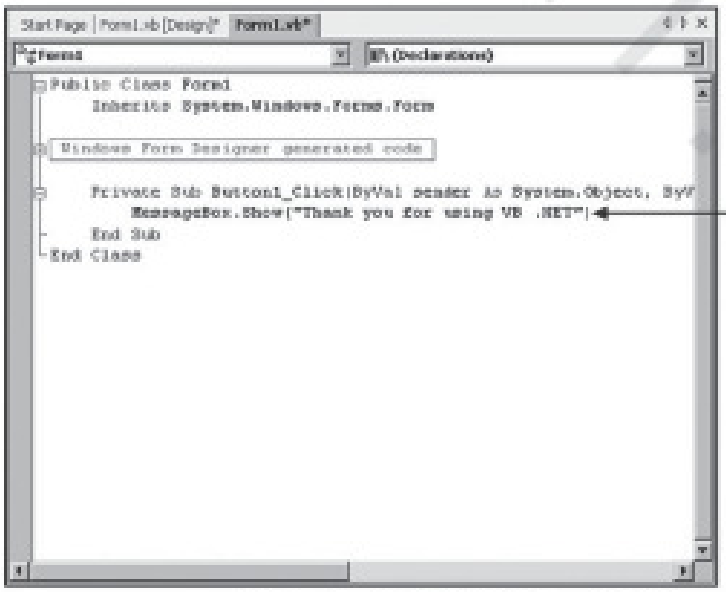
٢- اضغط ضغطاً مزدوجاً على الزر "Push Me"، وسيظهر محرر النصوص محتوياً على شفرة برمجية تم إنتاجها آلياً بواسطة لغة VB .NET بناء على تصميم النموذج الحالي (انظر الشكل رقم ٢,٣٩). سوف يتضح معنى هذه الشفرة خلال فصول هذا الكتاب القادمة.

لاحظ أن مؤشر الكتابة (Insertion point) يوجد بين سطري Private Sub...End Sub اللذين يمثلان بداية الإجراء الفعلي ونهايته المناظر للضغط على الزر والذي يتم استدعاءه تلقائياً عند الضغط على الزر "Push Me".

٣- اكتب الأمر MessageBox.Show("Thank you for using VB .NET") كما هو واضح في الشكل رقم (٢,٤٠).
لاحظ خاصية الكتابة التلقائية (IntelliSense) عند كتابة الأوامر.



الشكل رقم (٢,٣٩). الأوامر التي أنشأها مصمم النماذج آلياً.



الشكل رقم (٢,٤٠). إضافة أوامر إجراء معالجة الزر.

٤- لاحظ أن كلاً من مسطرة نافذة محرر الشفرة ومسطرة سهم نماذج الويندوز يحتويان على رمز النجمة (*) والتي تشير إلى تغيير محتويات هاتين النافذتين بعد آخر تخزين لهما على القرص الصلب، ولتخزين محتويات النافذتين اضغط على زر "Save All" المتواجد داخل مسطرة الأدوات.

ولترجمة البرنامج وتنفيذه اتبع الخطوات التالية :

١- اختر القائمة الفرعية Debug، ثم اختر الأمر "Start Without Debugging"، عندئذ ستظهر نافذة المخرجات (Output window) لتعلمك هل تم ترجمة البرنامج وبنائه بنجاح أم لا.

٢- إن لم توجد هناك أي أخطاء برمجية، سيظهر النموذج داخل نافذة بعنوان FormDemo (انظر الشكل رقم ٢,٤١).

٣- اضغط على الزر "Push Me"، وعندئذ ستظهر رسالة شكر لاختيارك لغة VB .NET (انظر الشكل رقم ٢,٤٢).

الشكل رقم (٢,٤١). مخرجات برنامج FormDemo.

الشكل رقم (٢,٤٢). رسالة الشكر.

٤- اضغط على الزر OK لإغلاق نافذة رسالة الشكر، ثم اضغط على زر إغلاق النموذج الرئيس (X) لإغلاق النموذج، ثم أغلق أيضاً نافذة المخرجات.

عزيزي القارئ لقد أصبح لديك فكرة لا بأس بها عن إمكانيات لغة VB .NET ومرونة التطوير بها حيث تعرفت كيف تستخدم كلاً من نافذة الخصائص ومصمم نماذج الويندوز لتطوير التطبيقات بسرعة ويسر. سوف تتعرف أثناء بقية هذا الفصل على إمكانات أخرى للغة VB .NET (أداة تصحيح الأخطاء (Debugging tool)، وأدوات المساعدة (Help facility)).

استكشاف أداة تصحيح الأخطاء
Exploring the Debugging Tool

تحتوي فيجوال ستوديو .نت على أداة قوية لتصحيح أخطاء أوامر البرنامج (يطلق عليها اسم Debugger) والذي تُستخدم لعزل الأخطاء مما يجعل البرنامج يعمل كما هو مخطط له. تحتوي أداة Debugger على العديد من الإمكانيات، وسنعرض واحدة فقط من هذه الإمكانيات خلال هذا الفصل، أما البقية فسيتم التعرض لها لاحقاً.

سوف نعرض في هذا الفصل استخدام أداة Debugger لوضع نقاط توقف (Breakpoints) والتي تستخدم لوقف تنفيذ البرنامج عند موضع محدد، وبذلك يتمكن المبرمج من رؤية معلومات الذاكرة (قيم المتغيرات) للمساعدة على اكتشاف مصدر المشكلة، وسوف نتعرف في الفصل الرابع على الأساليب المستخدمة لاكتشاف المشكلة وحلها عند توقف البرنامج عند نقطة التوقف.

البداية مع أداة تصحيح الأخطاء
Getting Started with the Debugger

عزيزي القارئ يجب عليك قبل استكشاف أداة تصحيح الأخطاء أن تتعرف أولاً على دورها الأساسي. صممت هذه الأداة لمساعدتك على التعرف على الأخطاء البرمجية التي تحدث أثناء تنفيذ البرنامج (الأخطاء المنطقية) وليس دورها هو معرفة الأخطاء النحوية التي تمنع ترجمة برنامجك من البداية. وبمعنى آخر يجب عليك أولاً أن تقوم ببناء برنامجك وترجمته بشكل صحيح قبل استخدام هذه الأداة.

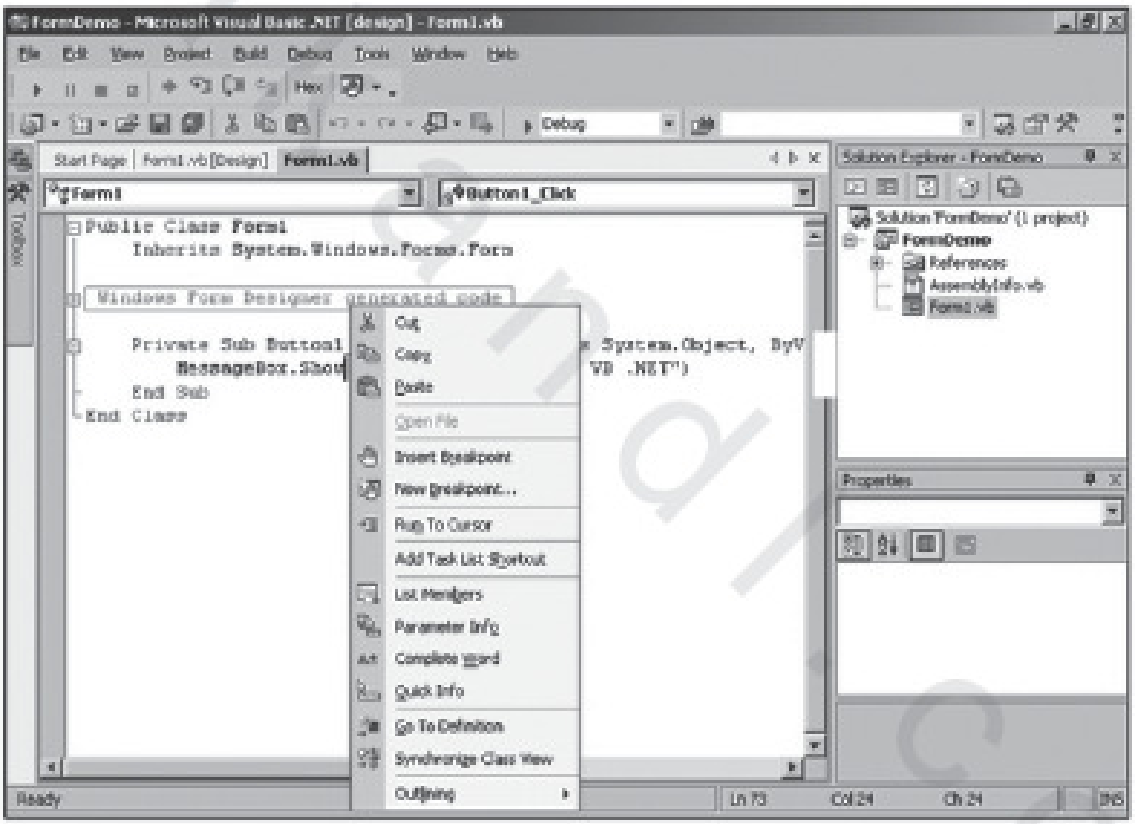
وضع نقاط التوقف
Setting Breakpoints

تذكر أن نقطة التوقف هي علامة يتم وضعها في مكان ما داخل شفرة البرنامج والتي تجعل أداة Debugger تعمل على توقيف البرنامج عندها، والآن دعنا نضع نقطة توقف عند الأمر المستخدم لإظهار رسالة شكر في البرنامج السابق. لوضع نقطة توقف:

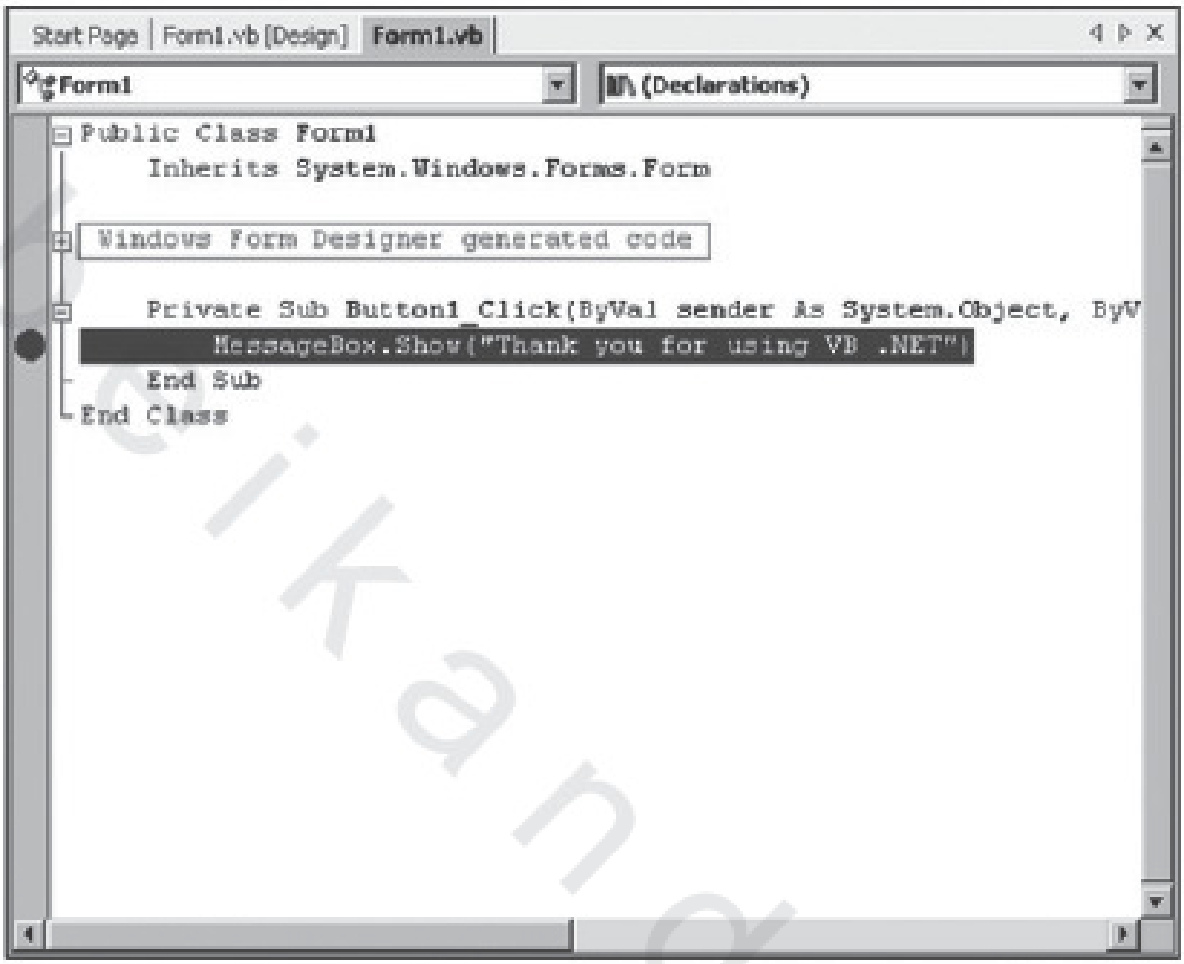
١- اضغط بالزر الأيمن للفأرة عند الأمر MessageBox.Show، وعندئذ ستظهر قائمة فرعية كما هو واضح في الشكل رقم (٢,٤٣).

٢- اختر أمر "Insert Breakpoint"، فعندئذ تظهر نافذة محرر شفرة البرنامج كما هو واضح في الشكل رقم (٢,٤٤). لاحظ أن الأمر MessageBox.Show يظهر باللون البني مما يعني وضع نقطة توقف عند هذا الأمر.

سوف يتوقف تنفيذ البرنامج قبل تنفيذ هذا الأمر عند تنفيذ البرنامج في وضع تصحيح الأخطاء (Debug Mode).



الشكل رقم (٢,٤٣). قائمة إدراج نقطة توقف.



الشكل رقم (٢,٤٤). إدراج نقطة توقف داخل البرنامج.

لتنفيذ البرنامج في وضع تنفيذ الأخطاء، اتبع الخطوات التالية:

١- اختر القائمة الفرعية Debug، ثم اختر الأمر Start، عندئذ تظهر العديد من النوافذ بواسطة أداة Debugger أسفل الشاشة، ثم تظهر نافذة النموذج.

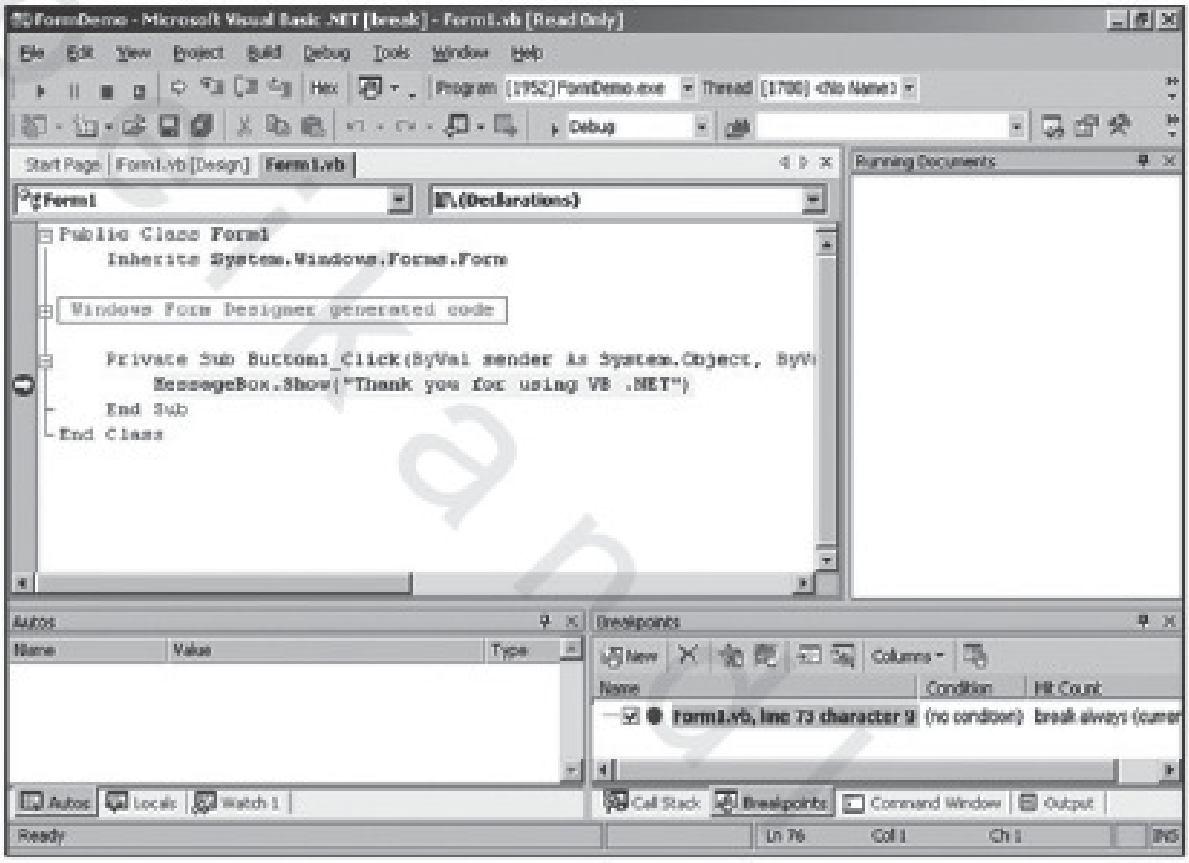
٢- عند الضغط على زر "Push Me" تظهر نافذة محرر شفرة البرنامج وفيها يظهر أمر MessageBox.Show محدداً باللون الأصفر مشيراً إلى الوصول إلى نقطة التوقف (انظر الشكل رقم ٢,٤٥) التي عندها توقف البرنامج عن التنفيذ. سوف تتعلم معرفة القيام بالخطوات اللازمة لعزل أخطاء البرنامج أثناء توقف تنفيذ البرنامج لاحقاً في هذا الكتاب.

٣- لاسترجاع تنفيذ البرنامج، اختر القائمة الفرعية Debug ثم اختر الأمر Continue، وسوف ينفذ الأمر MessageBox.Show ومن ثم تظهر رسالة الشكر.

٤- اضغط على زر OK لغلق رسالة الشكر.

٥- قم بإغلاق نافذة المخرجات (Output window).

٦- اختر الأمر "Close Solution" من القائمة الفرعية File وذلك لغلق المشروع.



الشكل رقم (٢,٤٥). توقف البرنامج عند نقطة التوقف.

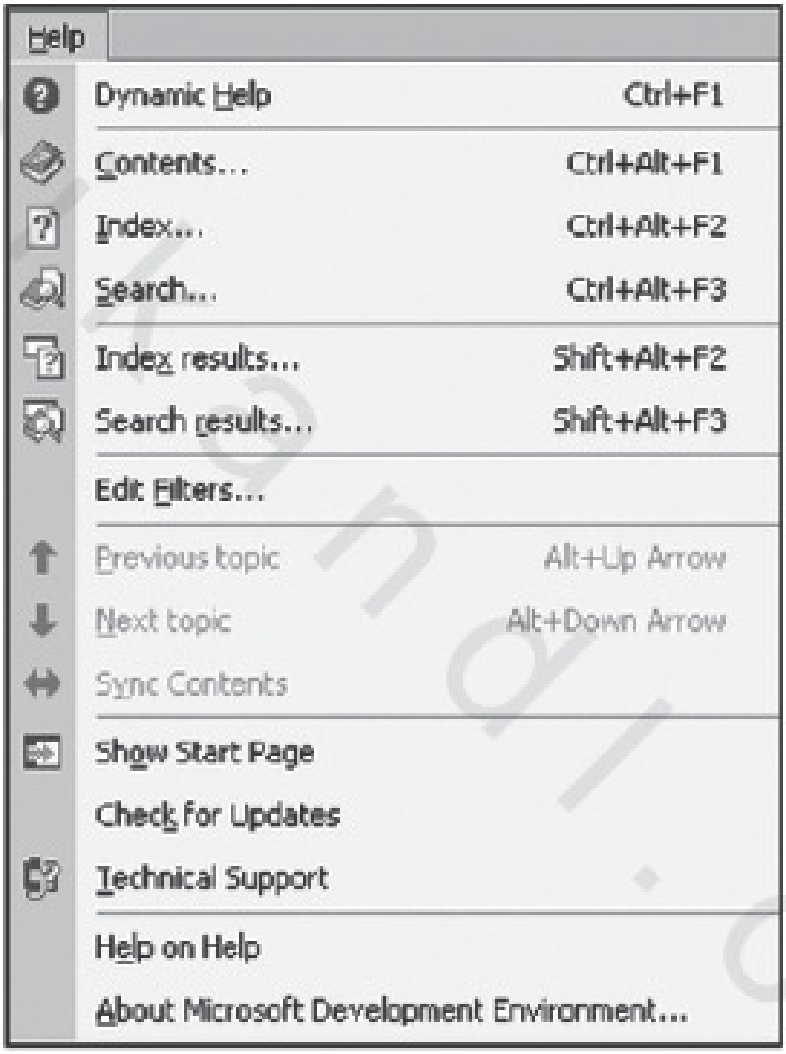
استكشاف أدوات المساعدة

Exploring the Help Facility

تقدم لغة فيجوال بيسك .نت العديد من أدوات المساعدة التي تمكن المبرمج من الوصول إلى معلومات مساعدة بأكثر من أسلوب مثلها مثل معظم تطبيقات الويندوز التي تقدم للمستخدم أداة للبحث عن معلومة ، وأداة تصفح جدول المحتويات ، والتصفح من خلال قائمة مواضيع مرتبة أبجدياً. هذا بالإضافة أن لغة VB .NET تشتمل على أدوات فعالة جداً للمساعدة وهي : أداة المساعدة الديناميكية (Dynamic help) ، وأداة مساعدة السياق الحالي (Context-sensitive help). وسوف نتعرف في هذه الفقرة على جميع إمكانات المساعدة المقدمة من قبل لغة فيجوال بيسك .نت.

الوصول إلى أدوات المساعدة
Accessing Help

تستطيع أن تصل إلى معظم أدوات المساعدة من خلال أوامر القائمة الفرعية Help الموضحة في الشكل رقم (٢,٤٦). عند اختيار أحد أوامر قائمة Help، يتم فتح نافذة مناظرة داخل نافذة بيئة التطوير المتكاملة (IDE window).

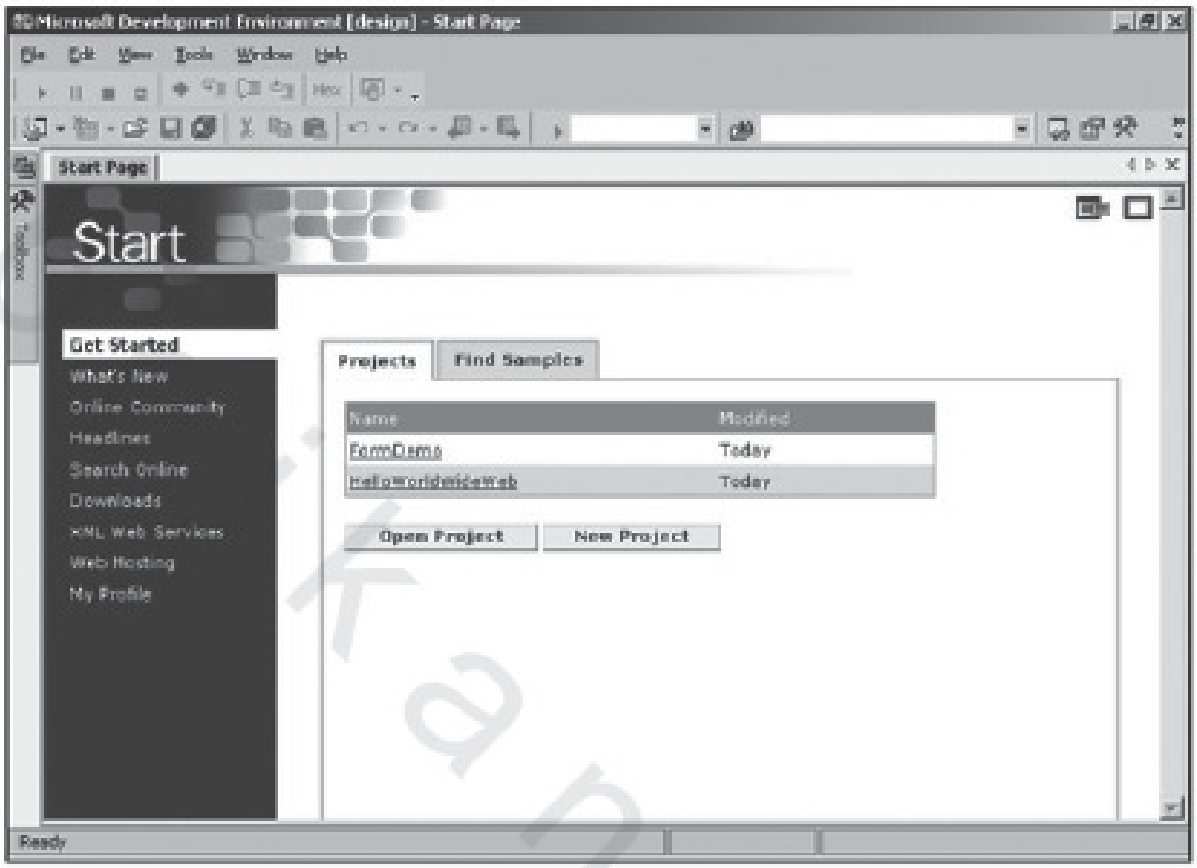


الشكل رقم (٢,٤٦). قائمة المساعدة (Help Menu).

لكي تتأكد أن نوافذ فيجوال ستديو .نت الخاصة بك تشبه الشكل الموجود في هذا الكتاب، اتبع التالي:

١- أغلق جميع النوافذ عدا نافذة صفحة Start Page.

٢- أغلق جميع نوافذ الأدوات (مثل نافذة متصفح الحلول، ونافذة الخصائص)، فعندئذ يجب أن تتشابه شاشة جهازك الحاسب مع الشكل رقم (٢,٤٧).



الشكل رقم (٢,٤٧). فتح نافذة صفحة البداية فقط داخل بيئة التطوير المتكاملة.

استكشاف الخيار Contents للقائمة الفرعية Help

Exploring the Contents Option on the Help Menu

يعرض الخيار Contents المتواجد داخل القائمة الفرعية Help قائمة بمواضيع مساعدة على شكل جدول محتويات، ولكي تفتح نافذة المحتويات هذه اتبع التالي:

١- اختر القائمة الفرعية Help، ثم اختر Contents، وعندئذ ستظهر نافذة المحتويات.

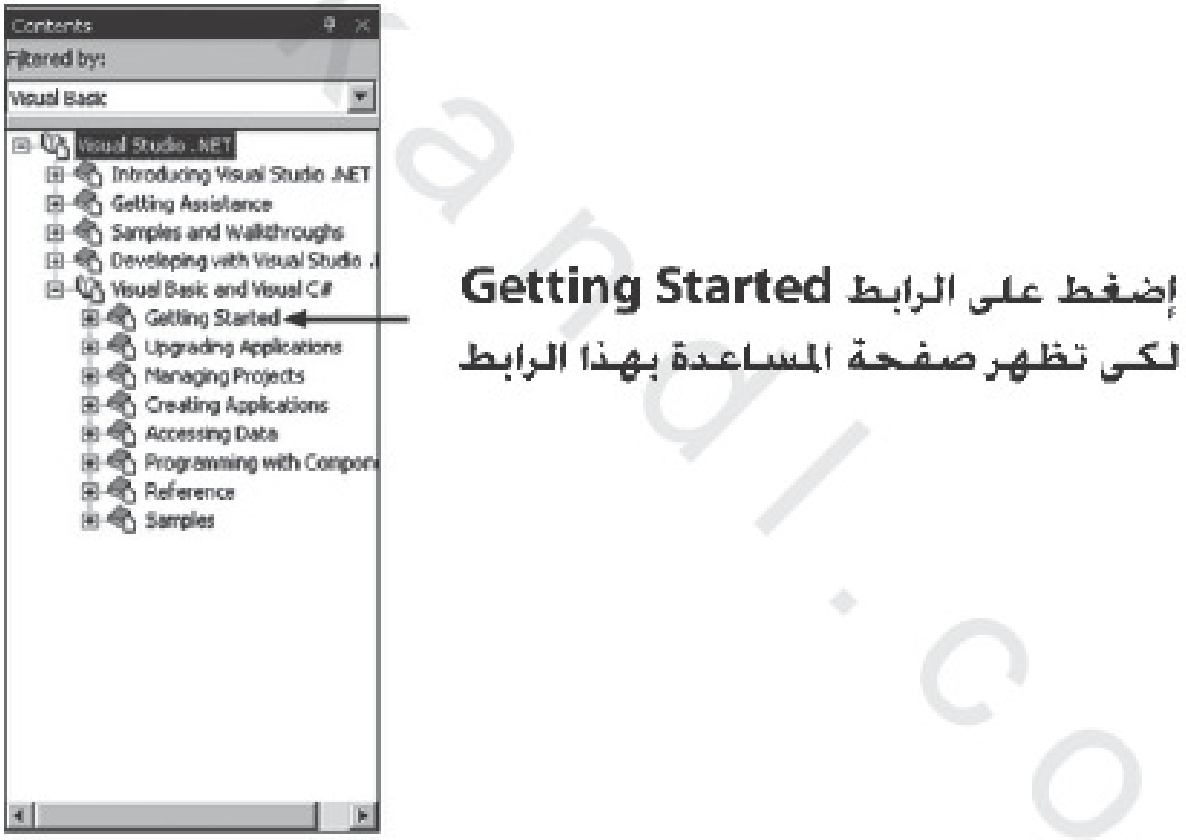
٢- يمكنك إظهار مواضيع لغة VB .NET وذلك باختيار Visual Basic من قائمة Filter.

لاحظ أن فيجوال ستوديو .نت تشتمل على لغات أخرى غير لغة VB .NET، وعند اختيار Visual Basic من قائمة Filter ستظهر المواضيع المتعلقة فقط بلغة VB .NET، كما يمكنك أيضاً إظهار ما يتعلق بلغة أخرى عند اختيار هذه اللغة من هذه القائمة.

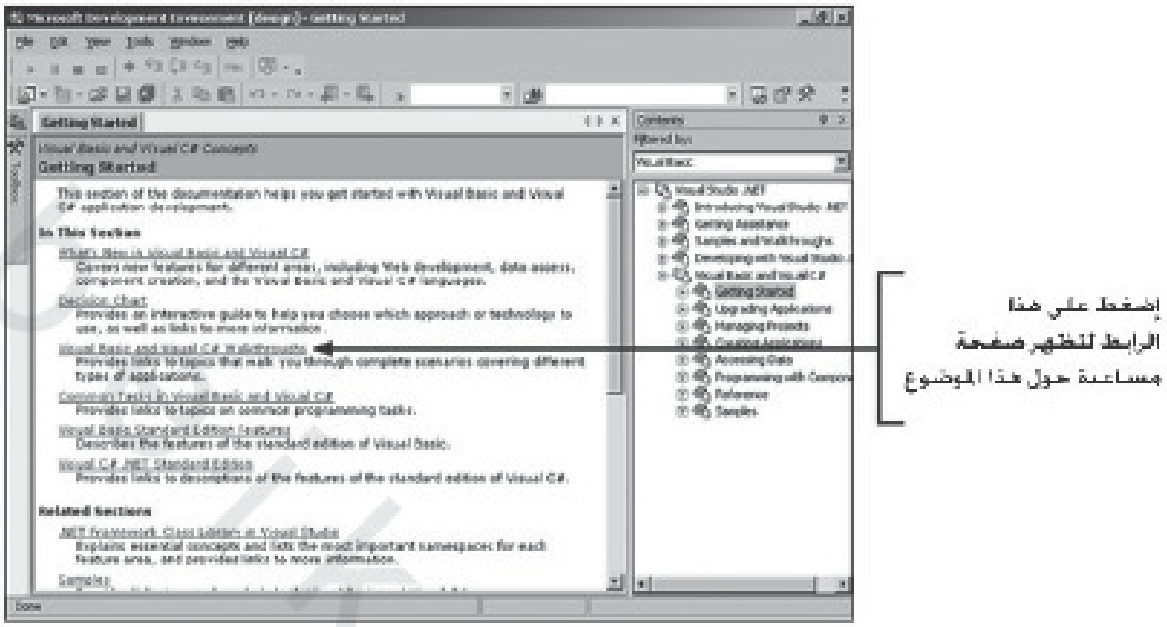
٣- لاحظ أن المحتويات داخل نافذة المحتويات تظهر في شكل شجري حيث توجد كل من علامة "+" وعلامة "-" على الجانب الأيسر لعقد الشجرة مما يفيد إمكانية توسعة مسار العقدة وانكماشها. إن توسعة مسار عقدة ما

بواسطة الضغط على علامة "+" يؤدي إلى ظهور مواضيع فرعية، أما الضغط على علامة "-" فيؤدي إلى إخفاء المواضيع الفرعية، فعلى سبيل المثال اضغط على علامة "+" للموضوع "Visual Studio .NET"، ثم اضغط على الموضوع الفرعي "Visual Basic and Visual C#" لكي ترى قائمة بالمواضيع الفرعية الأخرى الموضحة في الشكل رقم (٢.٤٨). ربما يختلف قليلاً الشكل الشجري الظاهر أمامك عن الشكل الموضح في الشكل رقم (٢.٤٨) وذلك بناء على إعدادات جهاز الحاسب الخاص بك.

٤- يمكنك الضغط على عنصر داخل الشجرة لإظهار محتويات تتعلق بهذا العنصر في نافذة منفصلة. فعلى سبيل المثال، اختر الموضوع "Getting Started"، فعندئذ ستظهر معلومات حول هذا الموضوع في نافذة منفصلة كما هو واضح في الشكل رقم (٢.٤٩).



الشكل رقم (٢.٤٨). توسعة شجرة المحتويات.

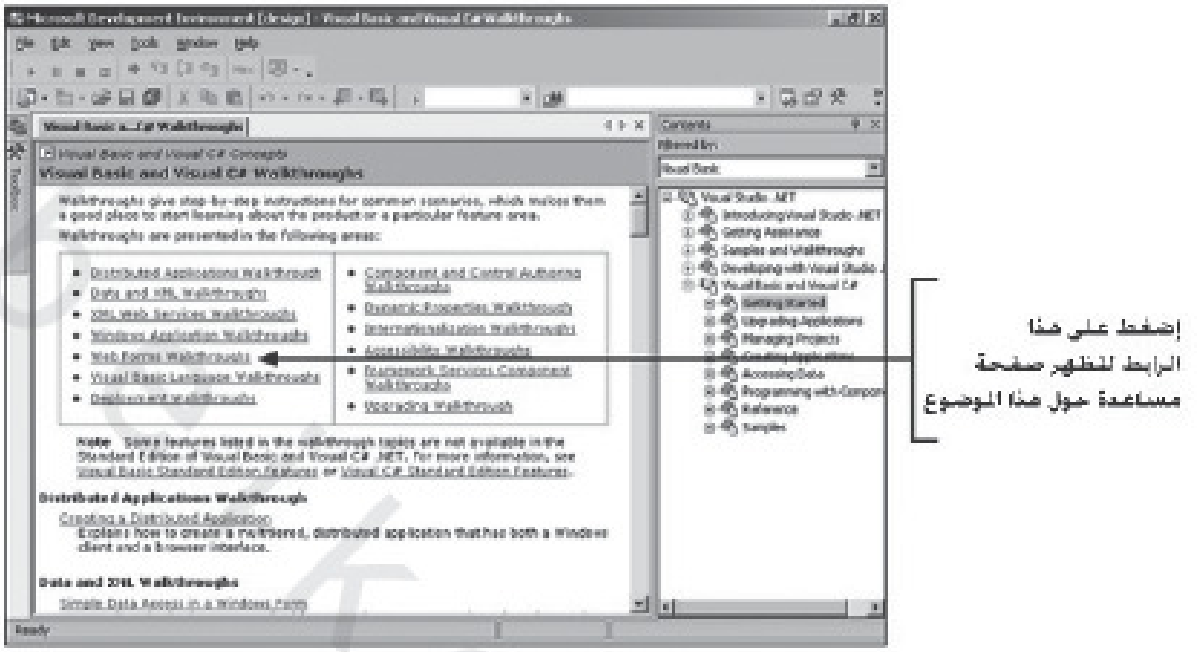


الشكل رقم (٢,٤٩). نافذة مساعدة الموضوع **Getting Started**.

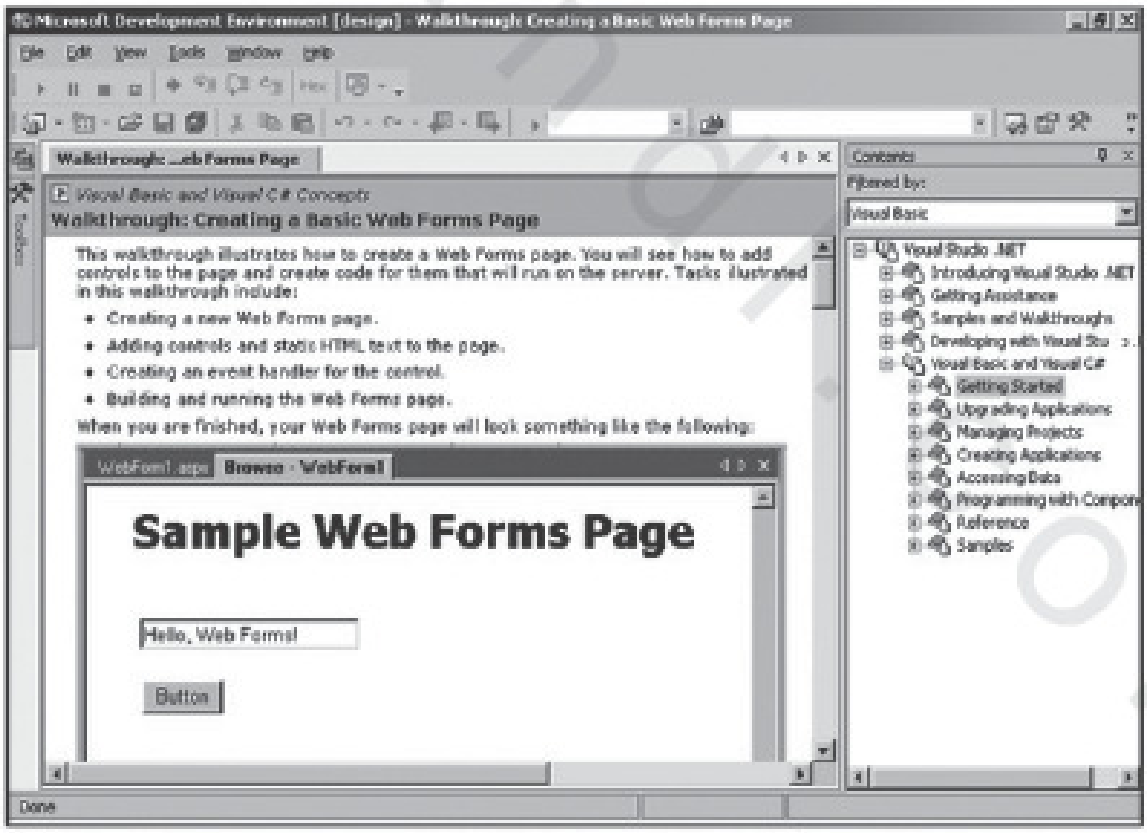
٥- يمكنك الضغط على أحد روابط هذه النافذة لكي تتجول داخل صفحات أخرى التي ستظهر معلومات أكثر تفصيلاً. فعلى سبيل المثال، اختر الرابط "Visual Basic and Visual C# Walkthroughs"، فعندئذ ستظهر صفحة أخرى داخل النافذة الحالية (انظر الشكل رقم ٢,٥٠).

٦- اختر الرابط "Web Forms Walkthroughs"، ثم اختر الرابط "Creating a Basic Web Forms Page" لكي تظهر معلومات حول هذا الموضوع (انظر الشكل رقم ٢,٥١).

٧- أغلق نافذة المحتويات (Contents window) بالضغط على زر (X).



الشكل رقم (٢,٥٠). صفحة مساعدة الموضوع Visual Basic and Visual C# Walkthroughs.



الشكل رقم (٢,٥١). صفحة مساعدة نماذج الويب الأساسية.

استكشاف الخيار Index (أحد أوامر قائمة Help)

Exploring the Index Option on the Help Menu

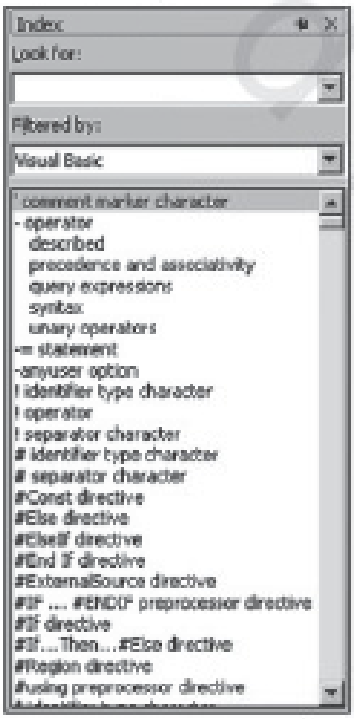
يظهر الخيار Index قائمة مواضيع مرتبة أبجدياً، ولكي تفتح نافذة Index اتبع الآتي:

١- اختر الأمر Index من قائمة Help، وعندئذ ستظهر نافذة Index كما هو واضح في الشكل رقم (٢.٥٢).

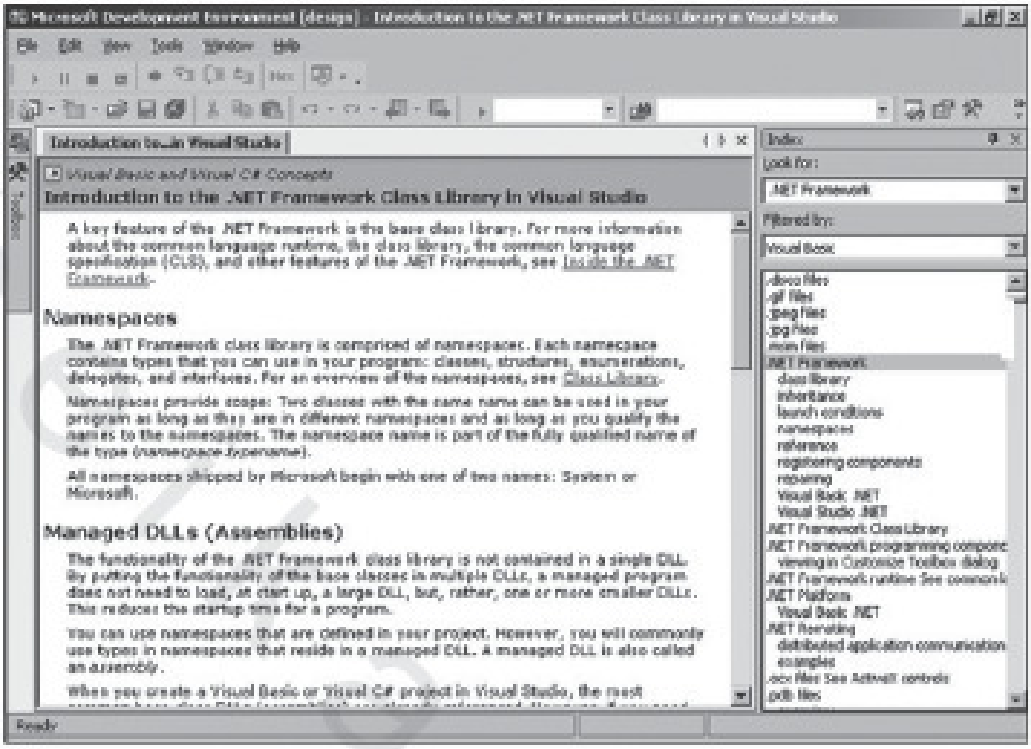
٢- اختر Visual Basic من قائمة Filtered.

٣- يمكنك إيجاد عناصر القائمة بواسطة التصفح خلال الفهرس، فعلى سبيل المثال قم بتصفح القائمة لأسفل، ثم اختار الخيار ".NET Framework"، وعندئذ ستظهر صفحة معلومات حول هذا الموضوع كما هو واضح في الشكل رقم (٢.٥٣).

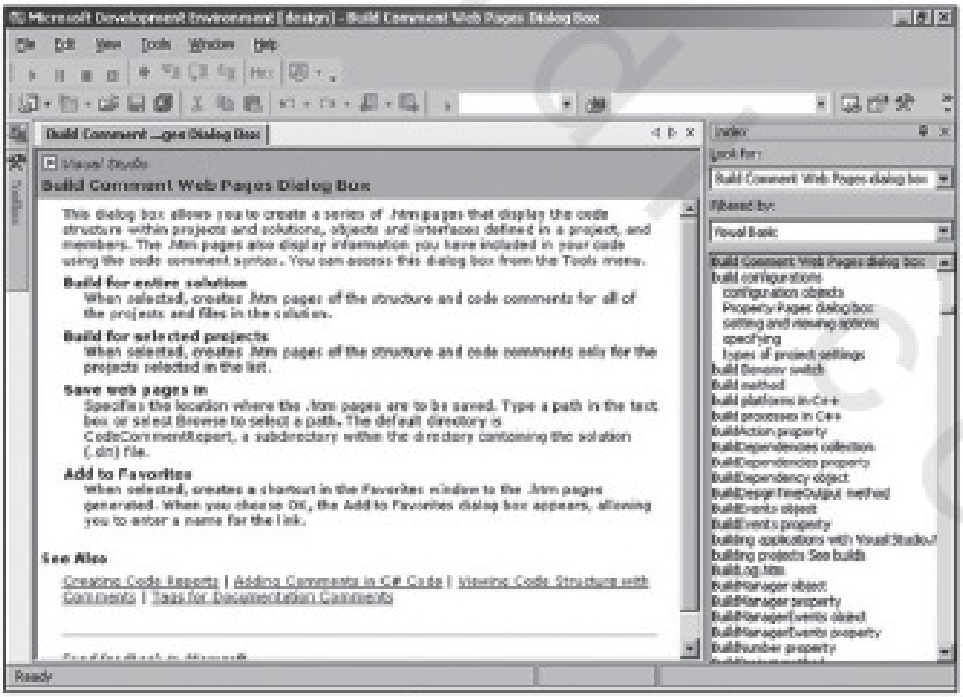
٤- توجد هناك طريقة أخرى لإيجاد موضوع محدد داخل مواضيع الفهرس وهو كتابة الموضوع داخل صندوق النص Look for. يبحث النظام بشكل تلقائي أثناء كتابة الموضوع على المواضيع التي تتطابق مع ما هو مكتوب، وعند اختيار موضوع من الفهرس تظهر نافذة معلومات هذا الموضوع. فعلى سبيل المثال للبحث عن الموضوع Build، ضع مؤشر الكتابة داخل صندوق النص Look for، ثم اكتب Build (لاحظ أن النظام يظهر المواضيع التي تبدأ بالحروف المكتوبة بشكل تلقائي)، ثم اضغط على مفتاح Enter لكي تظهر صفحة المساعدة للموضوع الحالي (انظر الشكل رقم ٢.٥٤).



الشكل رقم (٢.٥٢). نافذة Index.



الشكل رقم (٢,٥٣). مستند إطار عمل .نت.



الشكل رقم (٢,٥٤). نافذة استخدام إمكانية البحث (Look for).

لاحظ أنه يوجد في بعض الأحيان عدة صفحات مساعدة لموضوع ما، وعند اختيار مثل هذا الموضوع يظهر الفهرس روابط هذه الصفحات داخل بيئة IDE، وعندئذ يمكنك اختيار هذه الروابط لإظهار صفحة المساعدة المناظرة.

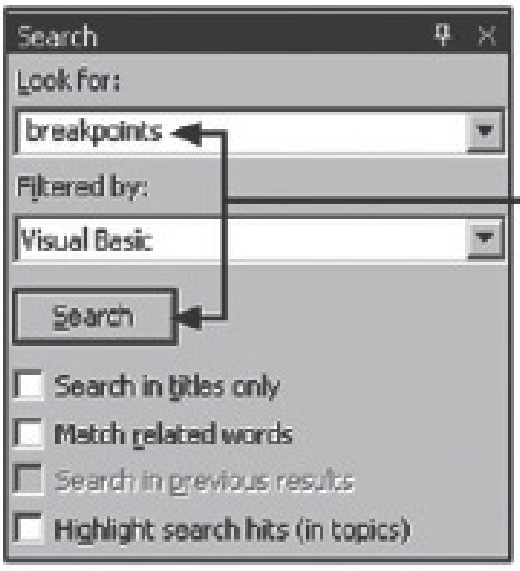
لاحظ أن فهرس المواضيع يحتوي أيضاً على عناصر ليس لها صفحات مساعدة ولكن تستخدم في عملية تنظيم مواضيع الفهرس وذلك لتجميع مجموعة من صفحات المساعدة ذات العلاقة. وعند الضغط على مثل هذه العناصر تظهر صفحة فارغة تطلب منك اختيار أحد روابط هذا الموضوع.

٥- أغلق كلاً من نافذة الفهرس ونافذة نتائج البحث داخل الفهرس.

استكشاف الخيار Search (أحد أوامر قائمة Help)
Exploring the Search Option on the Help Menu

يُستخدم الخيار Search للبحث عن كلمة أو عن عبارة محددة داخل قاعدة بيانات صفحات المساعدة، وعند تنفيذ عملية البحث تظهر قائمة بصفحات المساعدة التي تحتوي على هذه الكلمة أو العبارة داخل نافذة نتائج البحث (Search Results)، وعند اختيار أحد عناصر هذه القائمة تظهر صفحة المساعدة المناظرة. ولفتح نافذة البحث اتبع الخطوات التالية:

١- اختر القائمة الفرعية Help، ثم اختر Search، عندئذ تظهر نافذة البحث Search كما هو واضح في الشكل رقم (٢,٥٥).



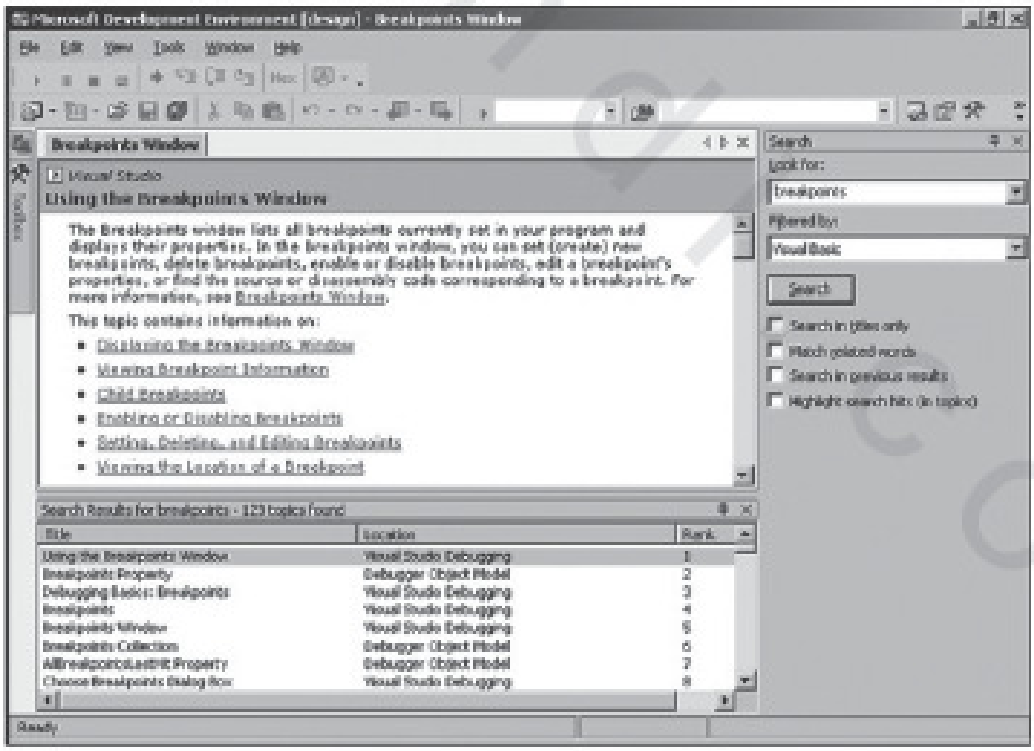
الشكل رقم (٢,٥٥). نافذة البحث.

٢- وللقيام بالبحث عن كلمة ما ولتكن العبارة "Breakpoints"، اكتب كلمة Breakpoints داخل صندوق النص Look for، ثم اضغط على زر Search، فعندئذ تظهر نافذة نتائج البحث التي تحتوي على ١٢٣ موضوعاً. اضغط ضغطاً مزدوجاً على أول موضوع في القائمة، عندئذ تظهر صفحة المساعدة المناظرة (انظر الشكل رقم ٢,٥٦).

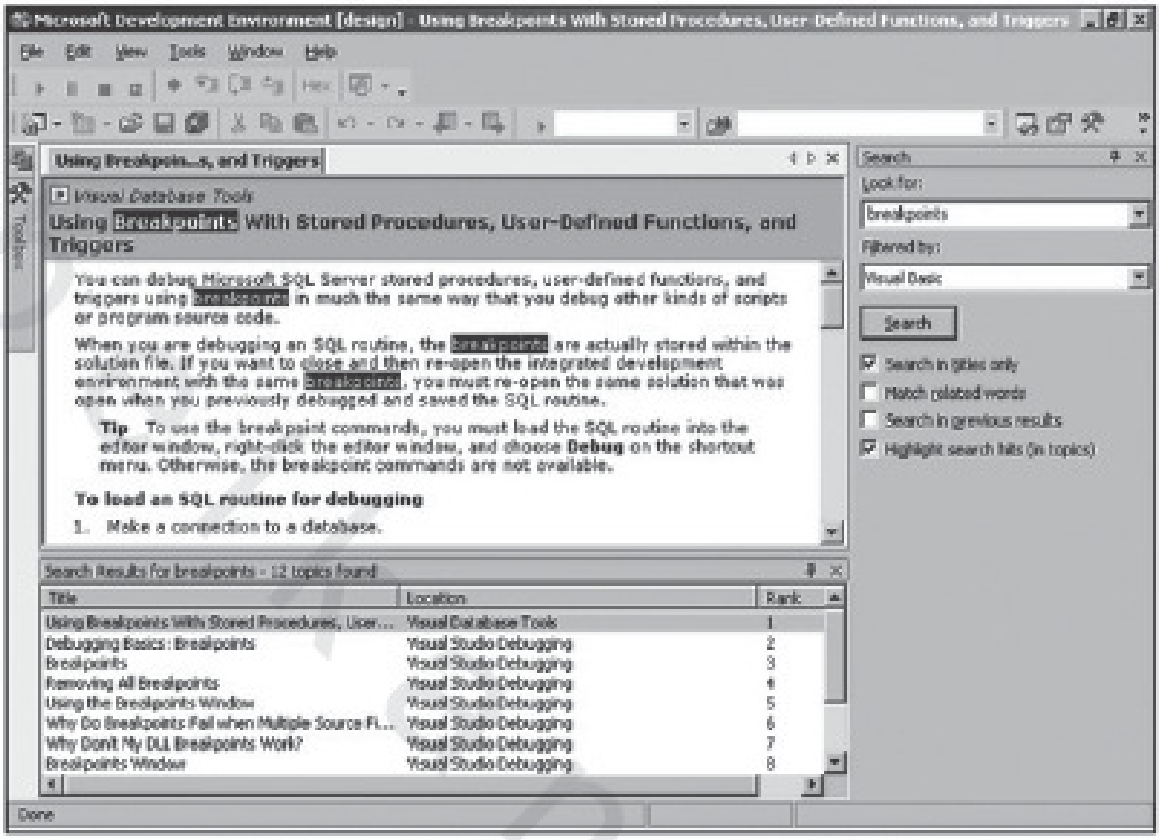
٣- تستطيع أن تبحث في العناوين فقط (Search in titles only)، أو تبحث في الكلمات الرئيسية (Match related words)، أو تبحث في نتائج البحث السابقة. هذا بالإضافة إلى تحديد الكلمات المستخدمة في البحث داخل صفحات النتائج وذلك مستخدماً صناديق التأشير الموجودة في نافذة البحث (الشكل رقم ٢,٥٥). والآن قم بإعادة عملية البحث السابقة ولكن أشر على Search in titles only الذي يؤدي إلى البحث في العناوين فقط، ثم أشر على Highlight search hits in topics والذي يؤدي إلى تحديد كلمة Breakpoints في نتائج البحث ثم اضغط على زر Search.

٤- لاحظ أن نافذة نتائج البحث تحتوي فقط على نتيجة البحث. اضغط على أول عنصر في القائمة، فعندئذ ستظهر صفحة المساعدة المناظرة لهذا الموضوع، ولاحظ أيضاً أن كلمة Breakpoints قد تم تحديدها في جميع أنحاء الصفحة (انظر الشكل رقم ٢,٥٧).

٥- أغلق نافذ البحث ونافذة نتائج البحث ونافذة صفحة المساعدة.



الشكل رقم (٢,٥٦). استخدام نافذة مساعدة نقاط التوقف.



الشكل رقم (٢,٥٧). نتائج البحث عن كلمة Breakpoints.

اكتشاف الخيار Dynamic Help (أحد اختيارات قائمة Help)
Exploring the Dynamic Help Option on the Help Menu

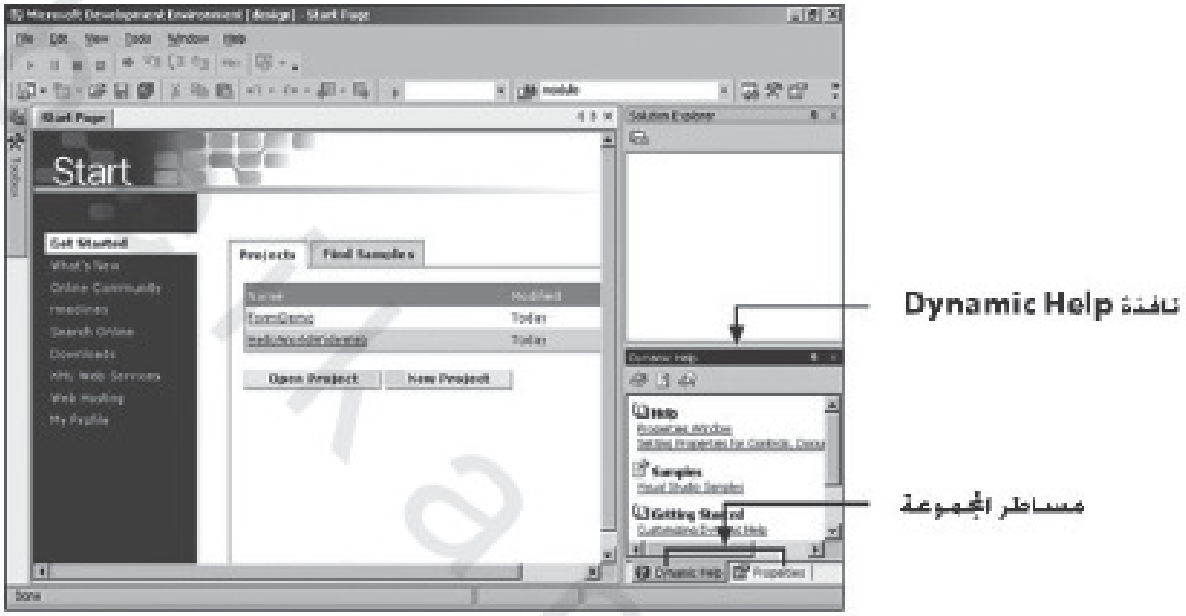
تمثل أداة المساعدة الديناميكية (Dynamic Help) أداة مساعدة فعالة التي تؤدي إلى إظهار مواضيع متعلقة بالأحداث التي تقوم بها الآن بشكل تلقائي. تعال لنرى سوياً كيف تعمل أداة المساعدة الديناميكية:

١- افتح نافذة Start Page إذا كانت مغلقة (اختر قائمة Help ثم اختر Show start page)، ثم افتح كلاً من نافذة متصفح الحلول ونافذة الخصائص.

٢- اختر الأمر Dynamic Help من القائمة Help، فعندئذ ستظهر نافذة Dynamic Help في الركن الأيمن السفلي من الشاشة (مع نافذة الخصائص) والتي تكون مع نافذة الخصائص ما يسمى Tab Group (ظهور النوافذ بشكل متبادل) كما هو واضح في الشكل رقم (٢,٥٨).

لاحظ أنه إذا ظهرت نافذة Dynamic Help في مكان مختلف فيمكنك أن تضعها في المكان المناسب. اضغط على عنوان النافذة بالزر الأيمن للفأرة للتأكد أن الوضع Dockable مختاراً، ثم اسحب النافذة عبر الشاشة (إلى الركن الأيمن)، وعندما ترى مسطرة النافذة تظهر في الركن الأيمن السفلي، قم بتحرير النافذة.

٣- نشط نافذة Start Page باختيارها، وعندئذ ترى أن نافذة Dynamic Help تظهر روابط صفحات تساعدك في مرحلة البداية (انظر الشكل رقم ٢,٥٩).



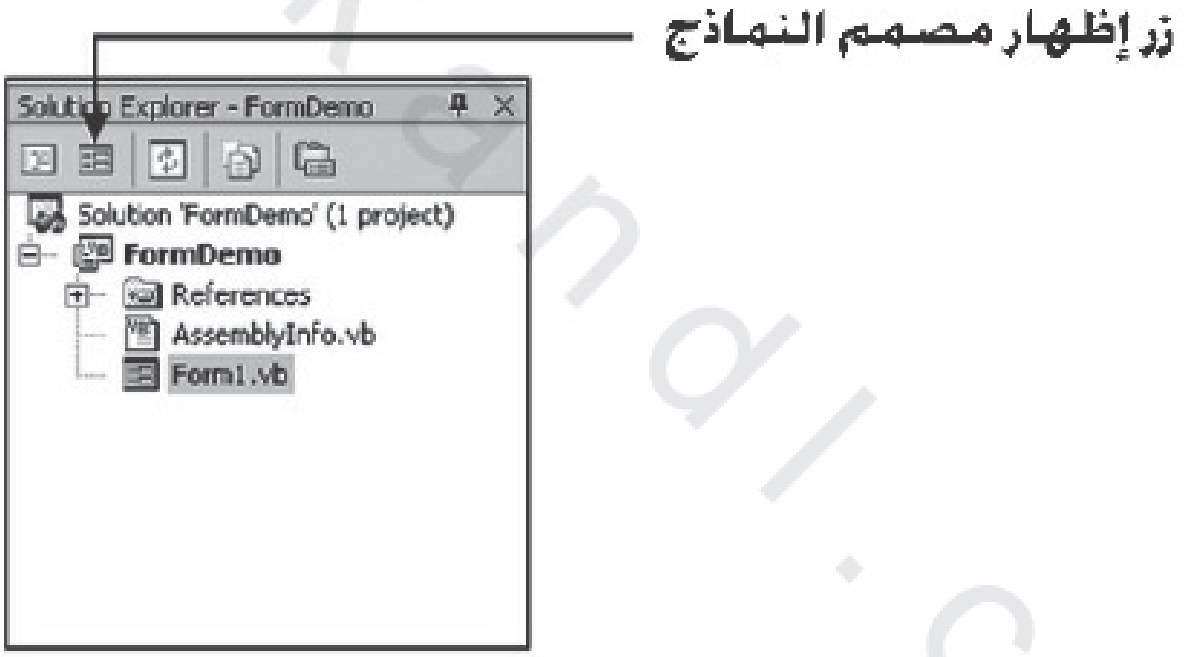
الشكل رقم (٢,٥٨). نافذة Dynamic Help.

الشكل رقم (٢,٥٩). تظهر نافذة Dynamic Help مواضيع حول Start Page.

٤- نشط نافذة متصفح الحلول بواسطة الضغط على عنوان النافذة ثم لاحظ أن محتويات نافذة Dynamic Help تغيرت لكي تمدك بروابط صفحات مساعدة حول متصفح الحلول.

٥- افتح مشروع FormDemo (إذا كان مغلقاً)، ثم اضغط على مسطرة نافذة Form1.vb [Design] في أعلى الشاشة لكي يظهر نموذج الإدخال المصمم سابقاً، ومن ثم ستتغير محتويات نافذة Dynamic Help لكي تعكس موضوعات مساعدة حول مصمم نماذج الويندوز (Windows Form Designer).

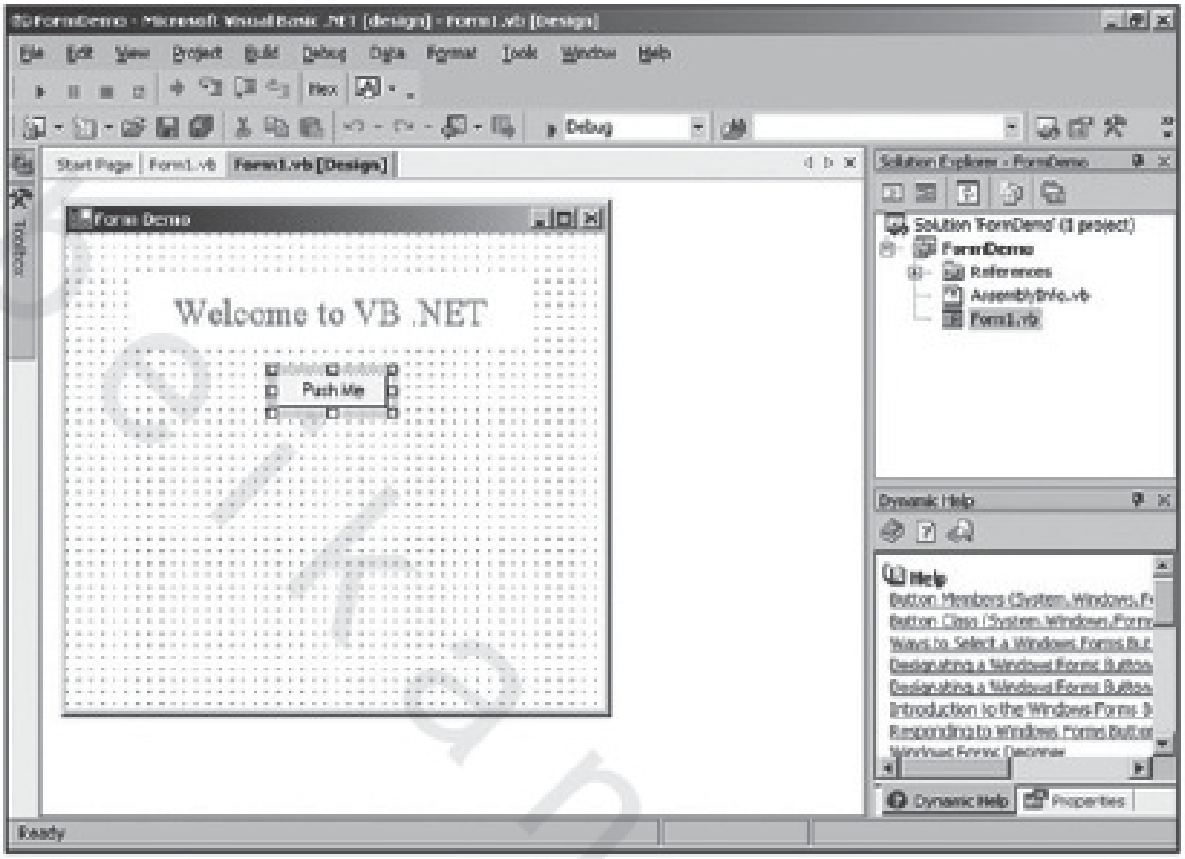
لاحظ أنه يمكنك فتح نافذة Form1.vb [Design] إذا كنت قد أغلقتها بالخطأ عن طريق الضغط المزدوج على الملف Form1.vb المتواجد داخل نافذة متصفح الحلول. يمكنك أيضاً فتح النافذة عن طريق اختيار الملف فقط من نافذة متصفح الحلول، ثم الضغط على رمز View Designer (انظر الشكل رقم ٢.٦٠).



الشكل رقم (٢.٦٠). يظهر زر مصمم النماذج داخل نافذة مستكشف الحلول.

٦- اختر الزر "Push Me" المتواجد داخل النموذج، فعندئذ ستلاحظ أن محتويات نافذة Dynamic Help ستتغير لتعكس موضوعات حول الصنف Button (انظر الشكل رقم ٢.٦١).

٧- اضغط الزر "Push Me" داخل النموذج Form1.vb [Design]. لاحظ أن محتويات نافذة المساعدة المتغيرة تتغير لتعكس موضوعات المساعدة التي لها علاقة بالأزرار كما هو واضح في الشكل رقم (٢.٦١).



الشكل رقم (٢,٦١). تظهر نافذة Dynamic Help معلومات حول الصنف Button.

اكتشاف أمر Context Sensitive

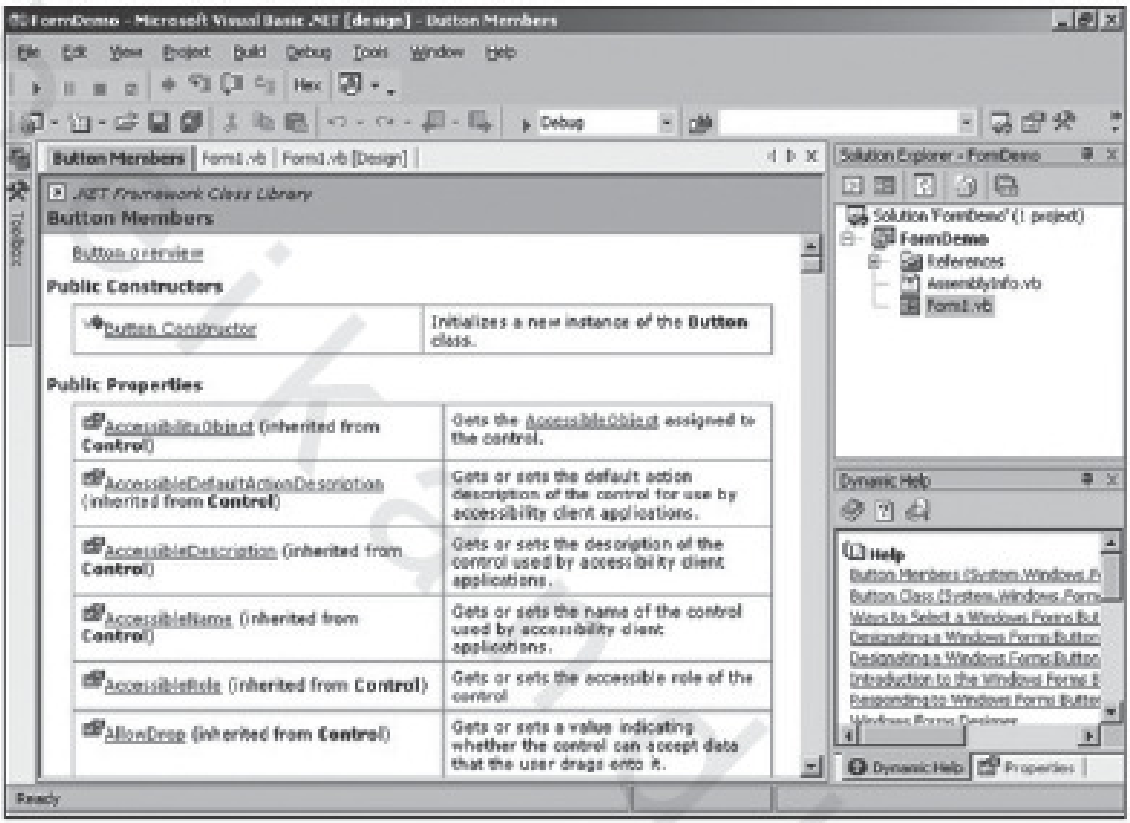
Exploring Context-Sensitive Help

توجد لدى لغة VB .NET أداة مساعدة أخرى فعالة نطلق عليها أداة السياق الحالي (Context Sensitive Help)، ولكي نستدعي هذه الأداة، نضغط على مفتاح F1 (لا يوجد اختيار لهذه الأداة داخل القائمة Help). ولكي ترى كيف تعمل أداة Context-Sensitive Help، اتبع الخطوات التالية:

١- استمراراً من آخر مرحلة تم التوصل إليها في الموضوع السابق (الزر Push Me مختاراً)، اضغط على المفتاح F1، عندئذ ستظهر نافذة مساعدة تحتوي على معلومات حول الصنف Button كما هو واضح في الشكل رقم (٢,٦٢). سوف تتعلم الكثير حول الصنف Button والمعلومات الموضحة في هذه الصفحة لاحقاً في هذا الكتاب.

٢- انقر مسطرة النافذة Form1.vb [Design] لكي تعود إلى نافذة مصمم نماذج الويندوز، ثم اضغط ضغطاً مزدوجاً على الزر Push Me لكي تفتح نافذة شفرة البرنامج التي تحتوي على نقطة التوقف (Breakpoint) المعرفة سابقاً.

٣- ضع مؤشر الكتابة في بداية الأمر MessageBox عند نقطة التوقف ، فعندئذ تتغير محتويات نافذة Dynamic Help لتقدم روابط صفحات مساعدة حول الأمر MessageBox.



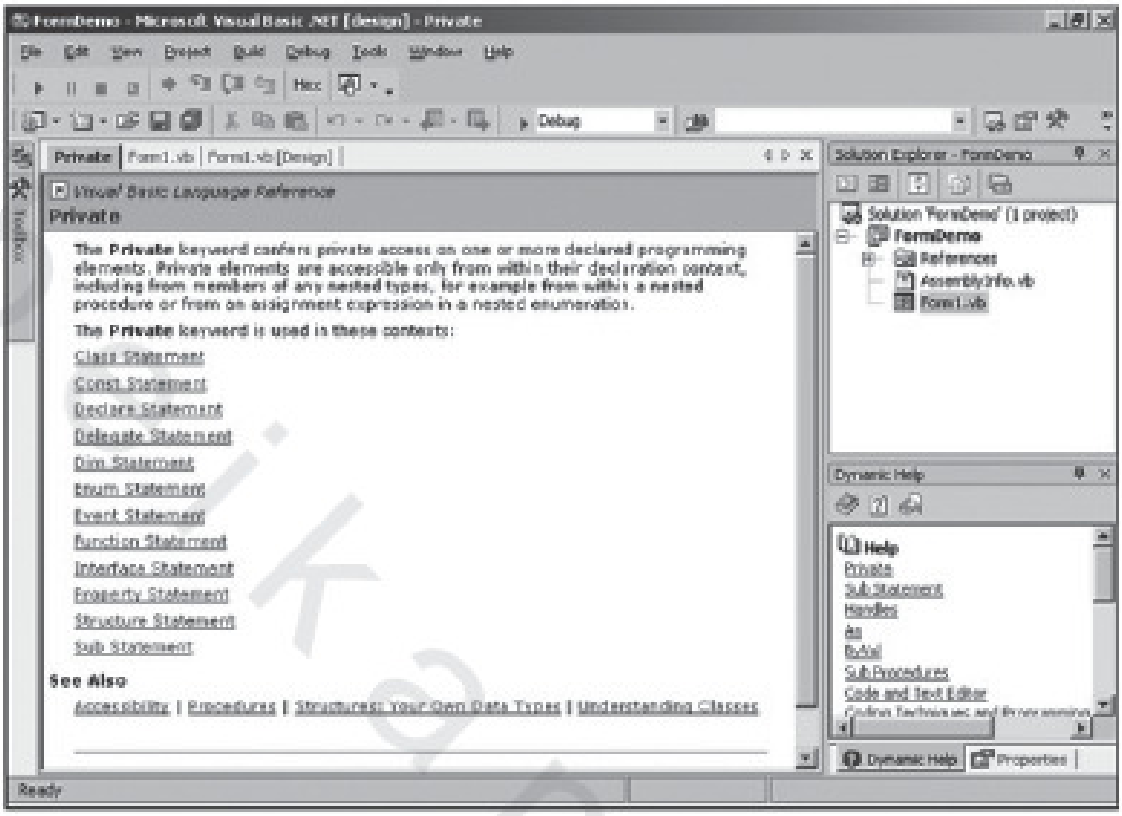
الشكل رقم (٢,٦٢). معلومات حول الصنف Button.

٤- اضغط مفتاح F1 لتظهر صفحة تحتوي على معلومات حول الصنف MessageBox.

٥- اضغط على مسطرة النافذة Form1.vb لكي تعود إلى نافذة شفرة البرنامج ، ثم ضع مؤشر الكتابة على الكلمة Private ، ثم اضغط على مفتاح F1 ، فعندئذ تظهر صفحة معلومات مساعدة حول الكلمة المحجوزة Private (انظر الشكل رقم ٢,٦٣).

٦- أغلق جميع النوافذ ، ثم أغلق برنامج فيجوال بيسك .نت.

تستطيع أن تضغط على زر F1 لكي تحصل على معلومات مساعدة حول أي شيء مثل الكلمات المحجوزة ، أو مكون ما ، أو نافذة ما ، أو أي عنصر آخر من عناصر لغة فيجوال بيسك .نت.



الشكل رقم (٢,٦٣). معلومات مساعدة حول الكلمة المحجوزة Private.

ملخص الفصل

Chapter Summary

- تمثل فيجوال ستوديو .نت بيئة تطوير متكاملة (Integrated Development Environment) التي تتكون من عدة أدوات برمجية لمساعدة المبرمج على كتابة شفرة البرنامج واختباره وتوثيقه.
- تمثل لغة فيجوال بيسك .نت واحدة من ضمن عدة لغات مدعمة بواسطة بيئة التطوير المتكاملة (Integrated Development Environment) لفيجوال ستوديو .نت.
- تعتبر تقنية فيجوال ستوديو .نت جزءاً من تقنية أكبر لمايكروسوفت حيث يتكون إطار عمل .نت (.NET Framework) من المكون محرك تنفيذ اللغات المشترك (CLR)، ومكتبة أصناف .نت (.NET Class Library) اللذان يقدمان خدمات مشتركة لجميع لغات فيجوال ستوديو .نت.
- تشمل بيئة التطوير المتكاملة لفيجوال ستوديو .نت على عدة أدوات برمجية لمساعدة المبرمج على كتابة شفرة البرنامج وتصميم نماذجه وترجمته واختباره وتوثيقه.

- يقدم محرر نصوص لغة فيجوال بيسك .نت العديد من إمكانات محرر النصوص المعتادة بالإضافة إلى إعطاء ألوان مختلفة للنصوص بناء على استخدامها، والتنسيق الآلي لشفرة البرنامج، والتكملة الآلية لأوامره.
- يسمح لك مصمم النماذج المرئي اختيار رموز مكونات النماذج المختلفة ووضعها وتنسيقها على النموذج لتصميمه، ومن ثم تنتج لغة فيجوال بيسك .نت الأوامر البرمجية المطلوبة لبناء النموذج أثناء مرحلة التصميم الذي تراه أمامك.
- تستخدم لغة فيجوال بيسك .نت تنظيماً شجرياً لتنظيم كل من الحلول والمشاريع. يمكن تشبيه كل من المشاريع والحلول بالحاويات حيث يحتوي كل مشروع على الملفات الخاصة به، كما يحتوي الحل على مشروع أو أكثر، وعادة يحتوي الحل في الأنظمة الكبيرة على عدة مشروعات.
- تقدم لغة فيجوال بيسك .نت الخيارات التي تسمح لك بتنسيق واجهة التطوير بالشكل المناسب لك.
- يحتوي فيجوال ستوديو .نت على أداة قوية لتصحيح أخطاء أوامر البرنامج (يطلق عليها اسم Debugger) والذي يُستخدم لعزل الأخطاء مما يجعل البرنامج يعمل كما هو مخطط له.
- تُستخدم نقاط التوقف (Breakpoints) لتوقف تنفيذ البرنامج عند موضع محدد وبالتالي يتحكم المبرمج من رؤية معلومات الذاكرة (قيم المتغيرات) للمساعدة على اكتشاف مصدر المشكلة.
- تقدم لغة فيجوال بيسك .نت العديد من أدوات المساعدة التي تمكن المبرمج من الوصول إلى معلومات مساعدة بأكثر من أسلوب مثلها مثل معظم تطبيقات الويندوز التي تقدم للمستخدم أداة للبحث عن معلومة، وأداة تصفح جدول المحتويات، والتصفح خلال قائمة مواضيع مرتبة أبجدياً.

المصطلحات الأساسية

Key Terms

نقطة توقف (Breakpoint)	كائن البداية (Startup object)
أداة تصحيح الأخطاء (Debugging Tool)	تطبيق ويندوز (Windows Application)
بيئة التطوير المتكاملة (Integrated Development Environment)	برنامج محث الدوس (Console Application)
مقبض (Handle)	قالب (Template)
نافذة مخفية (Hidden window)	خاصية الكتابة التلقائية (IntelliSense)
مشروع البداية (Startup project)	مشروع (Project)
	حل (Solution)

أسئلة المراجعة

Review Questions

١- ما هي بيئة التطوير المتكاملة IDE؟ ولماذا تعتبر هي الأكثر فائدة؟

٢- ما هي الأدوات الرئيسة في بيئة تطوير فيجوال ستوديو .نت المتكاملة؟ وما هي الميزات الأساسية لكل منها؟

٣- ما الفرق بين محرر النص ومحرر النموذج المرئي؟

٤- ما فائدة مستكشف الحلول؟

٥- ما فائدة صندوق الأدوات؟

٦- ما فائدة نافذة الخصائص؟

٧- ما فائدة من نافذة الوثيقة؟

٨- ماذا يعني إضافة نقطة توقف إلى البرنامج؟ وكيف تكون نقطة التوقف هذه مفيدة؟

٩- ما هي أنواع المساعدة الأساسية المتاحة في الفيجوال بيسك .نت؟ وماذا تفضل من هذه الأنواع؟ ولماذا؟

١٠- اشرح الفرق بين أداة المساعدة الديناميكية وأداة مساعدة السياق الحالي.

١١- ماذا يعني التعبير "مشروع البداية"؟ وكيف تميز مشروع البداية؟

١٢- ماذا يعني التعبير "كائن البداية"؟ ومتى يكون من الضروري تهيئة كائن البداية؟ وكيف تهيئ كائن البداية؟

١٣- ما الغرض من خاصية اللون الأمامي للعنوان؟

١٤- ما الغرض من خاصية النص للزر؟

١٥- كيف تضع نافذة الأدوات في الوضعية "pin"؟

١٦- ما هي النافذة المخفية؟ ولماذا النوافذ المخفية مفيدة؟

١٧- ما هو الفرق بين اختيار المحتويات واختيار الفهرس في قائمة المساعدة؟

١٨- ما الغرض من تصحيح الأخطاء؟ ومتى يمكن استعمال واحد منها؟

١٩- ما الفرق بين فيجوال ستوديو .نت و فيجوال بيسك .نت؟

٢٠- ما هو تطبيق الكونسول Console Application؟ وما الذي يختلف فيه تطبيق الكونسول عن تطبيق الويندوز؟

أسئلة المناقشة

Discussion Questions

١- ما هي مميزات استخدام بيئة التطوير المتكاملة لفيجوال ستوديو .نت وعيوبها؟

٢- بعد التعرف على بيئة التطوير المتكاملة لفيجوال ستوديو .نت، صف كيف يزيد استخدامك لها من إنتاجك كمبرمج؟

obeikandi.com

٣- قارن بين المصطلحات الثلاثة : حل، ومشروع، وبرنامج.

٤- هل تعتقد أن استخدام أدوات بيئة التطوير المتكاملة مثل أداة إنتاج شفرة البرنامج بشكل آلي يمكن أن يلغي دور المبرمجين؟ لماذا ولماذا لا؟

٥- ما هي أهمية استخدام كل من نافذة الخصائص وصندوق الأدوات في إنشاء تطبيقات الويندوز؟ ناقش كيفية استخدام هاتين الأداتين معاً أثناء تطوير نماذج المشروع؟

مشاريع الفصل

Projects

١- قم بإنشاء تطبيق ويندوز باسم MyFirstProgram وقم بتخزينه على المجلد Chap02\Projects على القرص C:، ثم قم بتغيير خلفية النموذج الأساسي إلى الأخضر الفاتح وقم بإضافة العنوان "This is my first program" على النموذج، ثم قم بتغيير لون العنوان إلى اللون الأزرق الغامق ولون خلفية العنوان إلى الأصفر. ثم قم بتغيير خط العنوان إلى Times New Roman والحجم ٢٠ أسود عريض، وإذا لزم الأمر قم بتغيير حجم النموذج ليظهر العنوان جيداً، وأخيراً ضع العنوان أسفل النموذج بحيث يتوسط النموذج وقم بتخزين النموذج.

٢- قم بإضافة زر إلى النموذج التي أنشأته في المشروع الأول، ثم اجعل عنوان الزر "What is my name?"، ثم قم بتغيير الخط ولونه والخلفية بالإعدادات التي تروق لك. وأخيراً اضبط مكان الزر ليتوسط النموذج وخزن المشروع.

٣- قم بتكملة المشروع رقم ٢ وذلك بإضافة شفرة البرنامج المطلوبة لعمل زر البرنامج الذي يجب أن يظهر الرسالة "My name is *your name*" حيث إن "*your name*" هو اسمك الفعلي، ثم أظهر رسالة أخرى "VB .NET is fun!"، وبعد ذلك قم بتخزين المشروع وتشغيله.

٤- قم بإعادة تسمية المشروع من Project 3 إلى MyFirstModule ثم قم بتغيير كائن البداية طبقاً لهذا التغيير، ثم قم بتخزين المشروع وتشغيله.

٥- استمر في عمل المشاريع السابقة، ثم قم بإضافة نقطة توقف بعد الرسالة الأولى وقبل ظهور الرسالة الثانية، وبعد ذلك قم بتخزين المشروع وتشغيله. وعندما يتوقف البرنامج استجابة لنقطة التوقف قم باستعادة تنفيذه وتأكد أنه يعمل كما هو مخطط له.