

إنشاء متصفح للويب بلغة V B. NET

٨٠ تهييد

ما رأيك - عزيزي القارئ - بعد أن تعرفنا على هذا الكم الكبير من أساسيات البرمجة بلغة Visual Basic أن نقوم الآن بإنشاء برنامج كامل يعمل في بيئة الويندوز.

فكرة هذا البرنامج عبارة عن متصفح للويب، يمكنك من خلاله كتابة عنوان الموقع URL في صندوق نص، والضغط على زر أمر للذهاب إلى الموقع؛ بشرط توفر الاتصال بالإنترنت على الكمبيوتر. وسوف أشرح لك بالتفصيل خطوات بناء واجهة التعامل، وكذلك ضبط خصائص جميع أدوات التحكم، وأيضا الكود المطلوب.

إن أفضل الطرق لتعلم البرمجة بلغة معينة، هي أن تقوم بإنشاء برامج مختلفة. ومن ثم، فإن هذا التدريب يأخذك خطوة بخطوة نحو تصميم وبناء برنامج يعرض صفحات الويب المختلفة.

١.٨ خطوات إنشاء البرنامج

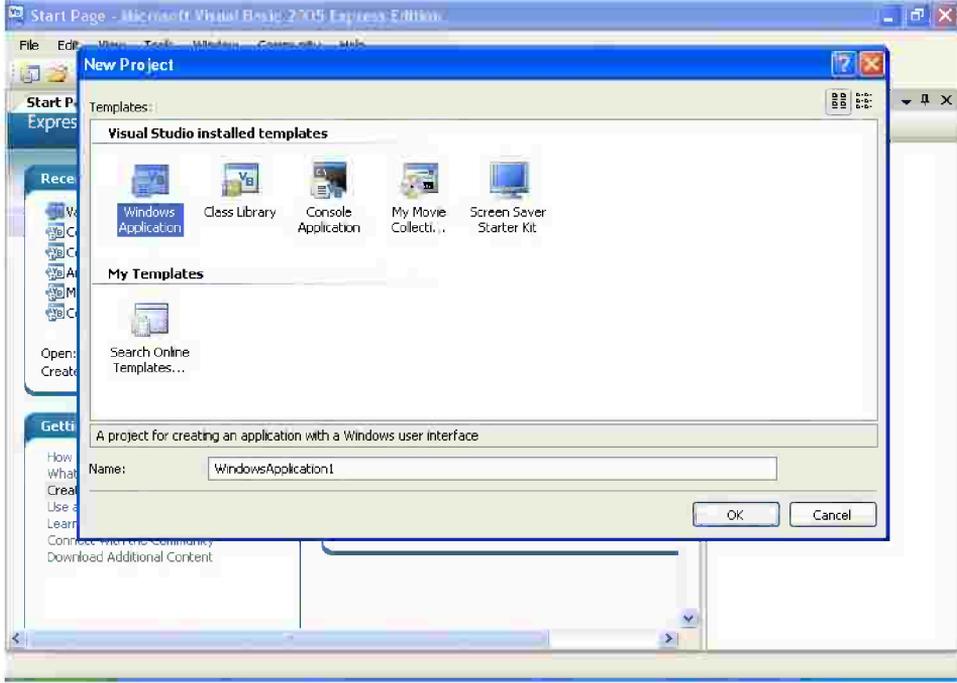
١. فتح مشروع في Visual Basic
٢. بناء واجهة التعامل مع المستخدم User Interface
٣. ضبط خصائص واجهة التعامل.
٤. إضافة الكود إلى عناصر التحكم.
٥. تشغيل واختبار البرنامج.

١.٨.١ فتح مشروع في Visual Basic

الخطوة الأولى في بناء برنامج بلغة VB هي فتح بيئة التطوير المتكاملة VB IDE وإنشاء مشروع project جديد.

١. من قائمة Start اختر All Programs ثم افتح Microsoft Visual Basic 2005 Express Edition.
٢. تظهر أمامك نافذة الترحيب "Welcome to Visual Basic Express". هذه هي واجهة التعامل في لغة VB، وتعرف أيضا، باسم واجهة التطبيق المتكاملة Integrated Development Environment IDE.
٣. من قائمة File اختر New Project.
٤. اختر Windows Application ثم اضغط Ok.
٥. يظهر أمامك نافذة نموذج form جديد، مع إضافة جميع الملفات الضرورية

للمشروع إلى نافذة مستكشف الحلول Solution Explorer. ولعلك لاحظت أن VB قد أعطى للمشروع الجديد الاسم الافتراضي WindowsApplication1



شكل (١.٨) إنشاء مشروع جديد في Visual Basic Express

في الخطوات السابقة قمنا بإنشاء مشروع لبرنامج تصفح الويب. والمشروع project في لغة VB عبارة عن حاوية تشمل جميع الملفات الخاصة بالبرنامج في مرحلة الإنشاء.

عند إنشاء مشروع جديد، يظل فقط في ذاكرة الكمبيوتر العشوائية. فإذا قمت بغلاق المشروع، فسوف يطلب منك VB إما أن تحفظه Save، وإما أن تتجاهله Discard. عند اختيار حفظ المشروع، يمكنك إعطاؤه اسماً أكثر دلالة من WindowsApplication1.

كما تلاحظ أيضاً - عزيزي القارئ - عند فتح صندوق حوار مشروع جديد، أن هناك أنواعاً مختلفة من المشروعات يستطيع VB أن ينشئها. وقد اخترنا تطبيق نوافذ

Windows Application لأن البرنامج الذى نحن بصدد إنشائه عبارة عن برنامج يعمل فى بيئة الويندوز، ويتم فتحه من خلال قائمة Start.

وبعد أن فتحنا مشروعنا جديدا، ظهر نموذج form (يعرف أيضا بمصمم النماذج form designer) فى وسط نافذة بيئة التطوير المتكاملة IDE. يمثل هذا النموذج نافذة البرنامج الجديد عند تشغيله. ونظرا إلى أن الكثير من البرامج قد تحتوى على نوافذ متعددة، فإن المشروع الواحد قد يحتوى على أكثر من نموذج.

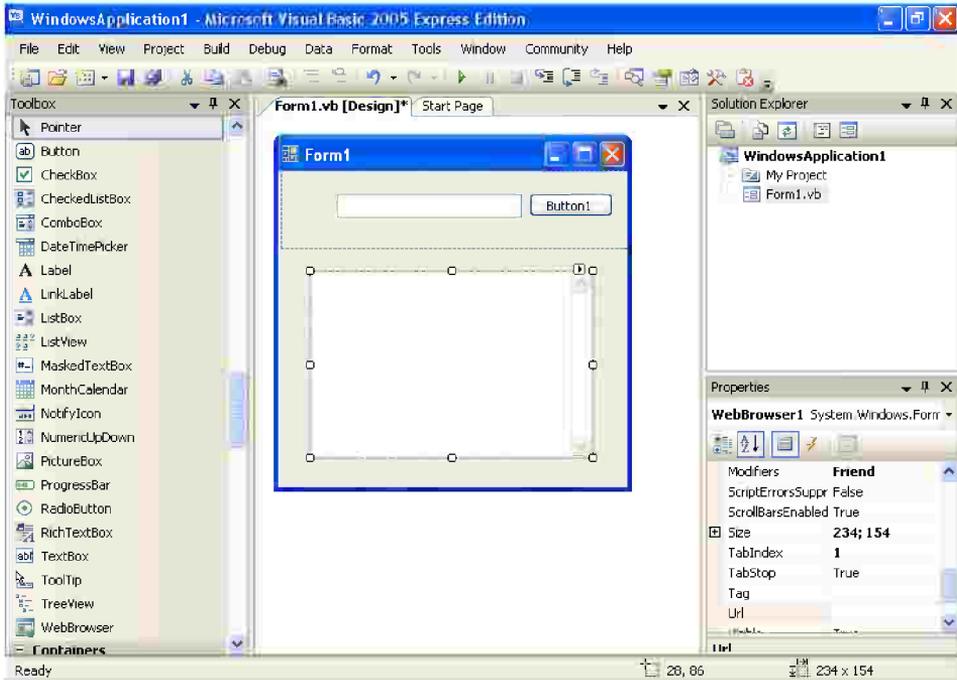
٢-١-٨ بناء واجهة التعامل مع المستخدم

حان الوقت الآن - عزيزى القارئ - للبدء فى بناء برنامج متصفح الويب. سوف نستخدم برنامج Microsoft Visual Basic 2005 Express Edition لبناء واجهة التعامل user interface وهى عبارة عن الجزء المرئى الذى يتعامل معه المستخدم عند تشغيل البرنامج. يتم بناء هذه الواجهة من خلال إضافة أدوات التحكم controls من صندوق الأدوات Toolbox إلى سطح النموذج.

يوجد صندوق الأدوات فى الجزء الأيسر من البرنامج، ويتكون من عدة تبويبات tabs، مثل Data وهو التبويب الخاص بأدوات التحكم التى نستخدمها عند التعامل مع قواعد البيانات، Components وهو التبويب الذى يحتوى على أدوات مساعدة مثل المؤقت Timer ومزود المساعدة HelpProvider، وأيضا التبويب المهم جدا All Windows Forms وهو الذى سوف نستخدمه بشكل أساسى فى برنامج متصفح الويب. من أمثلة أدوات الموجودة داخل هذا التبويب: صندوق النص TextBox، زر الأمر Button، صندوق الاختيار CheckBox، وكلها تمثل أدوات يمكنك إضافتها للبرامج التى تقوم بإنشائها، عن طريق سحبها ووضعها على سطح النموذج.

١. اختر أداة التحكم Panel تحت التبويب All Windows Forms فى صندوق الأدوات Toolbox. اسحب هذه الأداة وضعها فى الجزء العلوى من النموذج، ناحية اليسار.

٢. ضع زر أمر Button فوق أداة Panel.
 ٣. ضع صندوق نص TextBox أيضا فوق أداة Panel لكن بمحاذاة زر الأمر.
 ٤. وأخيرا ضع أداة WebBrowser على سطح النموذج، وتحت أداة Panel.
- في الخطوات السابقة، وضعنا أربع أدوات تحكم على سطح النموذج. وفائدة أدوات التحكم هذه هي احتواء الكود الذي يحدد وظيفتها في البرنامج. على سبيل المثال: زر الأمر Button هو أداة تحكم لا يخلو منها برنامج.
- وعلى الرغم من أنه يمكنك كتابة الكود الذي يقوم برسم زر أمر على سطح النموذج، مع تحديد شكل هذا الزر وخصائصه المختلفة؛ إلا أن Visual Basic قد وفر عليك الكثير من الوقت والجهد عزيزي القارئ؛ حيث يقدم لك ذلك الكود جاهزا، وما عليك سوى تحديد ما الذي يجب أن يفعله هذا الزر في برنامج.



شكل (٢.٨) بناء واجهة التعامل User Interface

وكما ترى، فإن صندوق الأدوات Toolbox يحتوي الكثير من أدوات التحكم، لكل منها وظيفته الخاصة. فالأداة Panel تستخدم لاحتواء أدوات تحكم أخرى، كتلك التي أضفناها إلى برنامج متصفح الويب الذي نحن بصدد إنشائه. أما أداة زر الأمر Button فعادة ما تستخدم لأداء بعض المهام عند ضغط click المستخدم عليها بزر الفأرة؛ كغلق البرنامج مثلاً. ويستخدم صندوق النص TextBox لإدخال نص بواسطة لوحة المفاتيح أثناء تشغيل البرنامج (الحصول على مدخلات من المستخدم). في حين تقدم أداة WebBrowser إمكانات تصفح الويب بشكل مشابه لبرنامج Internet Explorer على سبيل المثال. وبالتأكيد لا تريد أن تكتب بنفسك الكود اللازم للقيام بكل ذلك. وتلك إحدى سمات أو مزايا لغات البرمجة الحديثة التي تندرج تحت مظلة التطوير السريع للتطبيقات Rapid Application Development RAD الذي بدأ في تسعينيات القرن العشرين للتغلب على مشكلات مناهج التصميم الهيكلية Structured Design في إعداد برمجيات الحاسوب.

٣-١-٨ ضبط خصائص واجهة التعامل

بعد أن انتهينا من بناء واجهة التعامل، بإضافة أدوات التحكم إلى التطبيق، نكون قد وصلنا إلى مرحلة ضبط خصائص هذه الأدوات من خلال نافذة الخصائص Properties Window.

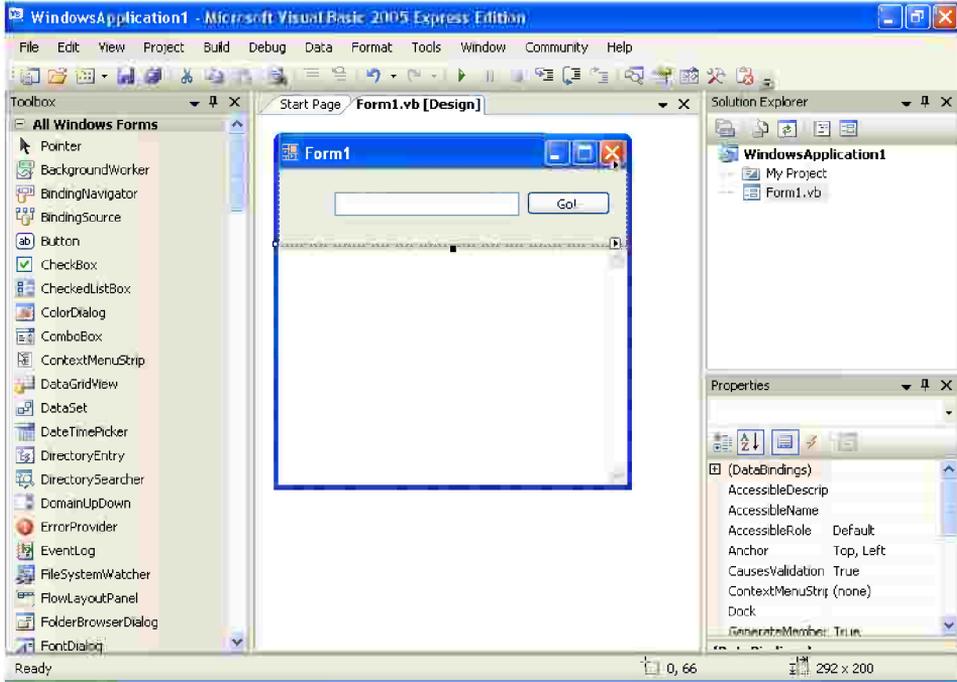
١. اختر أداة Panel الموجودة على سطح النموذج. تلاحظ أن نافذة الخصائص في الجزء السفلي على يمين شاشة IDE تعرض جميع الخصائص المتعلقة بهذه الأداة Panel1.

٢. في نافذة الخصائص اختر الخاصية Dock ثم اضغط السهم الموجود على اليمين. تلاحظ ظهور نافذة صغيرة تحتوي على عدة صناديق للاختيار من بينها.

٣. اختر الصندوق العلوي الموجودة ضمن نافذة Dock تلاحظ أن أداة Panel تتمدد لتملأ الجزء العلوي من النموذج.

٤. اختر أداة التحكم WebBrowser من على سطح النموذج، وفي نافذة الخصائص اضبط الخاصية Dock لكي تملأ هذه الأداة بقية مساحة النموذج أسفل Panel وذلك عن طريق اختيار الصندوق الأوسط في نافذة اختيارات الخاصية Dock.

٥. اختر زر الأمر Button واضبط خاصية Text على Go!



شكل (٣-٨) ضبط الخصائص في واجهة التعامل

في الخطوات السابقة، قمنا بضبط الكثير من خصائص أدوات التحكم في واجهة البرنامج. والخاصية في لغة Visual Basic بمثابة صفة لكائن object ما، والكائن هنا هو أداة التحكم. على سبيل المثال، إحدى خواص زر الأمر هو النص text الذي يظهر عليه. ففي الحالة السابقة، قمنا بضبط خاصية text إلى Go!.

وإلى جانب النص، يمكن للخصائص أن تأخذ أنواعاً مختلفة من القيم. فقد رأينا - على سبيل المثال - خاصية Dock والتي تعرض نافذة اختيارات. كما يمكن لقيمة الخاصية أن تكون رقماً، قائمة اختيارات، أو صحاً true وخطأ false.

بالإضافة إلى ما سبق، يجب أن نؤكد أيضا، على أنه عندما تقوم بتغيير حجم / موقع أداة تحكم معينة، فأنت في الواقع تغير خاصية من خواص هذه الأداة. وأخيرا تجدر الإشارة إلى أنه يمكن ضبط الخصائص من خلال الكود أيضا، إلى جانب استخدام نافذة الخصائص.

٤-٨-١ إضافة الكود إلى أدوات التحكم

في هذا الجزء، سوف نكتب الكود اللازم للتحكم في عناصر واجهة التعامل مع المستخدم.

١. اضغط مرتين على زر الأمر Button
٢. تفتح أمامك نافذة يطلق عليها محرر الكود Code Editor. وهذه هي النافذة التي تكتب فيها كود لغة Visual Basic.
٣. اكتب الكود التالي في إجراء الحدث Button1.Click

```
WebBrowser1.Navigate(Textbox1.Text)
```

سوف يتم تنفيذ هذا الكود عندما يضغط المستخدم بزر الفأرة على زر الأمر.

ربما لاحظت - عزيزي القارئ - أنه عند فتح نافذة الكود، فإنها تحتوي بالفعل على بعض أسطر الكود، والتي تبدو بالشكل التالي:

```
Private Sub Button1_Click(ByVal sender As System.Object...  
End Sub
```

يطلق على هذا الكود event handler أو منفذ الحدث Click بالنسبة إلى زر الأمر. وقد يطلق عليه أيضا Sub procedure أي إجراء فرعي، أي يتعلق فقط بها الكائن Button1. وبعبارة أخرى، فإن أية تعليمات برمجية داخل هذا الإجراء (بين Sub و End Sub) سوف يتم تنفيذها فقط عندما يضغط المستخدم على زر الأمر. كما قد تكون لاحظت - عزيزي القارئ - أن المؤشر يقف داخل الإجراء الفرعي، وكل ما هو مطلوب منك أن تكتب الكود اللازم.

إن الكود الذى كتبناه (`WebBrowser1.Navigate(textBox1.Text)`) يطلب من البرنامج استخدام طريقة `Navigate method` الخاصة بأداة التحكم `WebBrowser1` للذهاب إلى عنوان الموقع المبين فى `textBox1`، وبمعنى أدق قيمة الخاصية `Text` فى أداة التحكم `textBox1`.

الآن أصبح البرنامج جاهزا. وفى الجزء التالى، سوف نقوم بتشغيل البرنامج.

٨-٥ تشغيل واختبار البرنامج

الآن بعد اكتمال البرنامج، حان الوقت لتشغيله واختباره. بالنسبة إلى البرامج الكبيرة والمعقدة قد يكون الاختبار `testing` عملية طويلة وصعبة. لكن لحسن الحظ لا يحتاج البرنامج الحالى إلى اختبار، ومن ثم كل ما يجب عليك أن تفعله الآن هو تشغيله.

١. تأكد من اتصال الكمبيوتر بالإنترنت.

٢. فى قائمة `Debug` اختر `Start Debugging`.

ملحوظة

المفتاح `F5` يقوم أيضا بتشغيل البرنامج.

٣. فى صندوق النص اكتب `www.ala.org` ثم اضغط زر `Go!`.

٤. تلاحظ عرض موقع جمعية المكتبات الأمريكية فى أداة `WebBrowser` ويمكنك التجول فى الموقع كيفما تريد.

٥. لإيقاف البرنامج اختر `Stop Debugging` من قائمة `Debug`.

ملحوظة

يمكنك كذلك إيقاف تشغيل البرنامج بالضغط على علامة `X` الموجودة فى الركن الأيمن العلوى فى نافذة برنامج متصفح الويب.



شكل (٤-٨) تشغيل برنامج متصفح الويب

في الخطوات السابقة، قمنا بتشغيل البرنامج للتأكد من قيامه بتنفيذ المطلوب بنجاح. وفي أغلب الحالات، يحتاج المبرمج تكرار هذه العملية مرات كثيرة. وكل مرة يتم فيها إضافة كود جديد أو وظائف جديدة للبرنامج، يجب عليك تشغيل البرنامج للتأكد من أنه يعمل بنجاح. وإذا لم يكن الأمر كذلك، يجب عليك تصحيح الأخطاء الموجودة. وتعرف هذه العملية باسم `debugging` أي إزالة أية أخطاء موجودة بالبرنامج.

وأخيرا .. ما رأيك - عزيزي القارئ - في هذا البرنامج؟ عن طريق سطر واحد فقط من الكود قمنا بإنشاء متصفح للويب، يساعدك على الإبحار بين مواقع الويب المختلفة. وهذا هو مكنم القوة في لغة `Visual Basic`. فالكود الضروري كان

موجودا ضمن أداة WebBrowser وبالتالي وفر علينا VB الكثير من الوقت والعناء. فقد كان الأمر - من دون وجود تلك الأداة - يحتاج منا إلى مئات بل آلاف السطور من الكود.

أهنتك - عزيزى القارئ - لوصولك إلى هذه المرحلة المهمة فى تعلم البرمجة بلغة Visual Basic. هل تتفق معنا أن البرمجة بالفعل عالم رائع!.

أيا كانت إجابتك، دعنا نتقدم للأمام أكثر وأكثر فى هذا العالم الرحب. فى الفصل التالى نذهب معا إلى استكشاف أراض جديدة فى هذا العالم.