الباب الحادى عشر

جولة فى إستوديو دوت.نت

(Microsoft Visual Studio .NET)

محتويات الباب:

- استخدام البيئة المجمعة للإستوديو
- بدء تطبيق جديد
- استخدام تطبيقات الكونصول
- معالم البيئة المجمعة
- ترجمة وتشغيل البرامج
- استخدام تطبيقات النوافذ
- استخدام تطبيقات الوب
- فتح تطبيق موجود (قديم)
- ترجمة وبناء مشروع مكتبة
- استخدام برنامج المكتبة ضمن التطبيق الحالى

استخدام البيئة المجمعة للإستوديو

فـــى هـــذا الـــباب نقـــدم جولـــة يحـــتاجها من كان جديداً على إستوديو ميكروسوفت المرئى الطراز 7.0 الذى صدر تحت اسم ميكروسوفت إستوديو دوت.نت (Microsoft Visual Studio .NET). ونقدم فى هذه الجولة أهم الملامح الأساسية للإستوديو وأنواع التطبيقات المختلفة التى يمكن برمجتها من خلاله. ولا ندعـــى أن هذا العرض سيكون كاملاً لأن الموضوع أكبر من لأن يغطيه فصـــل فى كتاب. ولكن ما نقدمه هنا يعتبر ملخصاً سريعا للملامح ويعطى فكـــرة عامـــة عـــن الإستوديو ، كما يقدم أهم الإرشادات فى الاستخدام مثل خطوات الترجمة والتنفيذ للبرامج المختلفة.

بدء تطبيق جديد

يحفـــظ الإستوديو التطبيق فى صورة حل (Solution) وقد يحتوى الحل الواحد على مشروع (Project) أو أكثر.

لكى تبدأ تطبيقاً جديداً اتبع الآتى:

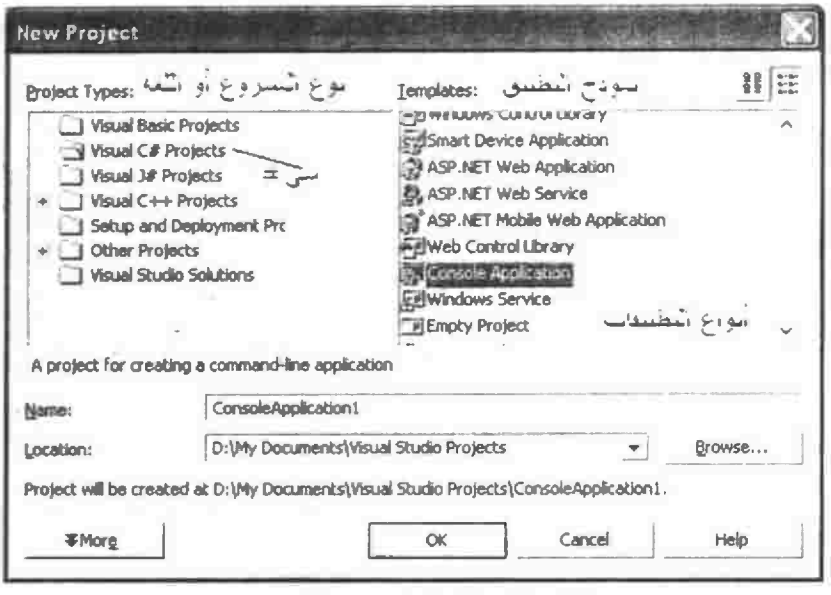
- ❖ افتح قائمة الملفات (Files)
- ❖ اختر New
- ❖ اختر Project

عندئذ تلتقى بالشاشة الموضحة بالشكل (1) ، وفيها تحدد الآتى:

- نوع المشروع أو اللغة التى ترغب فى استخدامها (القسم الأيسر من النافذة: Project Types). وقـــد اخـــترنا مشـــروعات سى# (Visual

C# Projects).

- نموذج التطبيق (فى القسم الأيمن من النافذة: Templates) وهو يحدد نوع التطبيق. وأهم أنواع التطبيقات هى:
  - تطبيقات النوافذ (Windows Applications)
  - تطبيقات الوب (ASP .NET Web Applications)
  - خدمات الوب (ASP .NET Web Services)
  - المكتبات (Class Libraries)
  - تطبيقات الكونسول (Console Applications) ؛ وهذه هى البرامج المستخدمة فى هذا الكتاب. (لكى ترى هذا الاختيار حرك المنزلق إلى أسفل فى القسم الأيسر.)



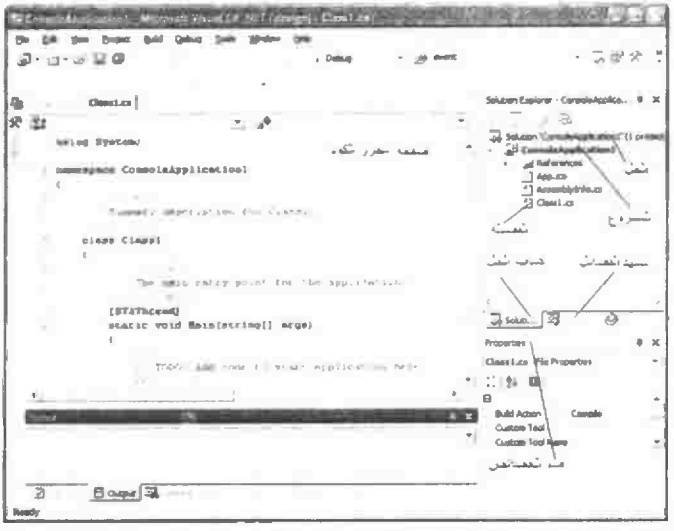
شكل (١) نافذة المشروع بالإستوديو

استخدام تطبيقات الكونصول

يمكـــنك تشـــغيل جميع برامج الكتاب باستخدام تطبيقات الكونصول أو باســـتخدام مترجم بيئة الأوامر. وعندما تختار تطبيقات الكونصول وتبدأ مشروعاً جديداً فإن المشروع الجديد سوف يحتوى على جزء من الكود يشمل:

- التوجيه using System
- حـــيز الاســـم ConsoleApplication1 ـــ وهـــذا هو الاسم سابق التعريف ، ويمكنك تغييره باستخدام اسم مختلف للتطبيق
- الفصيلة Class1
- الأسلوب Main() فارغاً

وتسمى الصفحة التى تكتب فيها الكود "مُحرر الكود". انظر الشكل (2).



شكل (2) مشروع كونصول جديد موضح به معالم البيئة المجمعة

ومـــن البديهى أنه يمكنك ترجمة وتشغيل هذا البرنامج الفارغ طالما أنه يحـــتوى علـــى الأسلوب Main(). يمكنك أيضاً أن تضيف ما تشاء إلى الكـــود الموضـــح بالشكل السابق. كما يمكنك أن تمسحه وتلصق مكانه برنامجاً كاملاً (مكتوب ببرنامج النوتة "Notepad" مثلاً).

ملاحظة:

لاحظ أن ملف الحل يأخذ أيضاً نفس الاسم كما ملف المشروع. ولو أنك أضـــفت مشـــروعات أخـــرى إلـــى التطبـــيق فســـوف تأخذ الأسماء ConsoleApplication1 و ConsoleApplication2 وهكذا.

معالم البيئة المجمعة

كمـــا نرى فى الشكل (2) أن البيئة المجمعة تحتوى على الأقسام الآتية (لاحـــظ أنـــه يمكنك إضافة المزيد من الأقسام عن طريق قائمة المشهد View):

- قسم الكتابة (النافذة بالركن الأيسر العلوى): وهو عبارة عن محرر الكود الذى يستخدم فى كتابة البرنامج.
- قســـم الخرج (النافذة بالركن الأيسر السفلى): وهو يحتوى على عدة بطاقات لمشاهدة نتيجة الترجمة أو الأخطاء.
- قســـم المشـــاهدة (الـــنافذة بالركـــن الأيمن العلوى): ويحتوى على البطاقات الآتية:

- كشاف الحل (Solution Explorer): وهو يعرض محتويات "الحل" الـتى تشمل المشروع ، والفصائل والمكتبات. وقد يحتوى الحل على أكثر من مشروع كما ذكرنا من قبل.
- مشـهد الفصـائل (Class View): وهو عبارة عن واجهة مختلفة لمشـاهدة الحـل حيـث ترى فيها الأهداف فى صورة فصائل وأساليب. أنظر الشكل (3).
- قسـم الـنجدة (Contents): ويحـتوى على وثائق النجدة لجميع محتويات الإستوديو.

- قسـم الخصـائص (النافذة بالركن الأيمن السفلى): وهى مخصصة لعـرض خصـائص الهدف الذى يقع عليه المؤشر فى قسم كشاف الحل أو مشهد الفصائل.

شكل (3): مشهد الفصائل لمشروع كونصول

ولا يوجـــد اختلاف كبير فى معالم البيئة المجمعة عند تشغيل التطبيقات المخـــتلفة ، فـــيما عدا أن تطبيقات النوافذ وتطبيقات الوب تحتوى على أقسام أخرى لتصميم الواجهة.

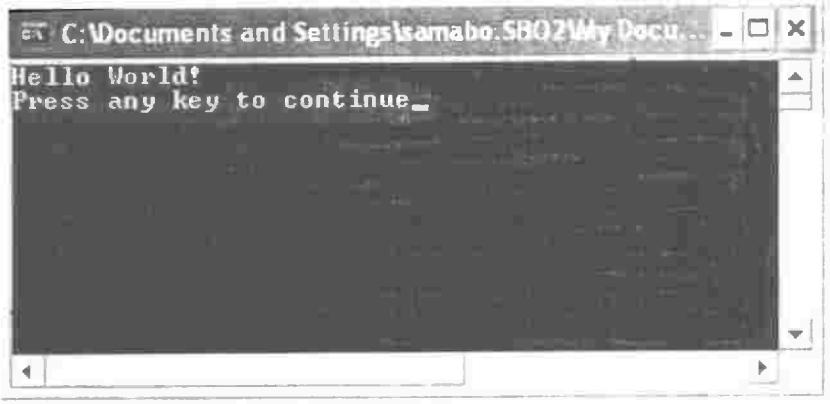
ترجمة وتشغيل المشروعات

يمكنك ترجمة وتنفيذ المشروعات من جميع الأنواع فيما عدا مشروعات المكتـــبات فإنهـــا تـــترجم فقط ولكنها لا تنفذ ، فالمكتبة تستخدم ضمن مشروع آخر.

لترجمة وتشغيل المشروع اتبع الآتى:

- للترجمة: استخدم الاختيار Build Solution أو Rebuild Solution من القائمة Build. (أو استخدم مجموعة الأزرار Ctrl+Shift+B).
  (إذا كـــان الحـــل يتضمن أكثر من مشروع فيمكنك ترجمة مشروع معيـــن باستخدام نفس القائمة. اختر المشروع ثم استخدم أمر البناء الخاص به من نفس القائمة مثل Build "*project-name*").
- للترجمة والتنفيذ معاً: استخدم الاختيار Start من القائمة Debug (أو اضغط الزر F5).
- لـــترجمة وتنفيذ برامج الكونصول: يفضل استخدام الاختيار Start Without Debugging مـــن القائمـــة Debug (أو اضـــغط المجموعة Ctrl+ F5). وهذا الأمر يعلق تشغيل البرنامج حتى تضغط على الزر

Enter كما هو موضح بالشكل (4). أما لو استخدمت الاختيار Start فإن البرنامج قد يبدأ وينتهى دون أن تلاحظ النتيجة.

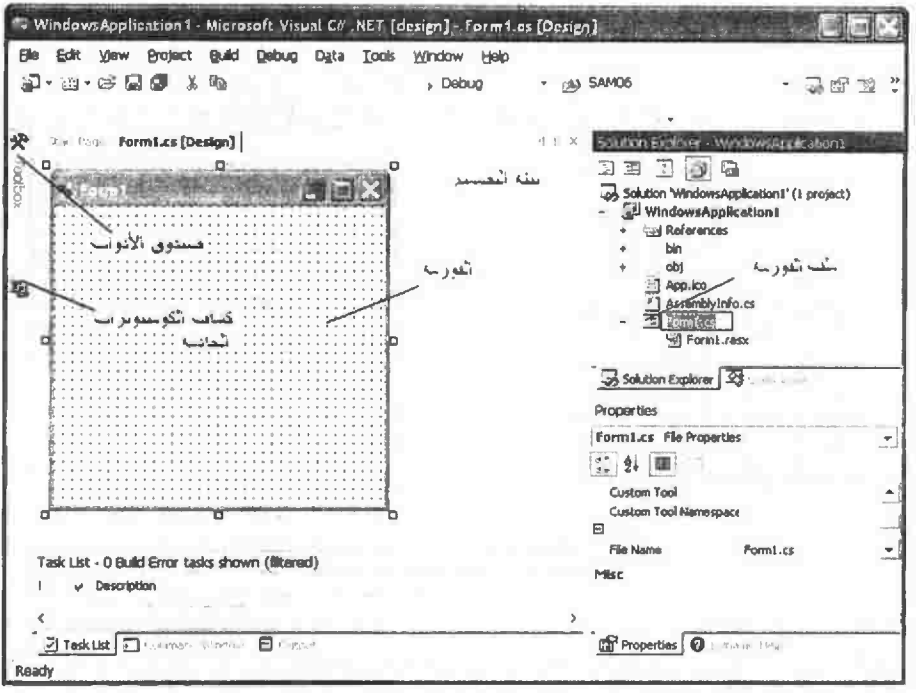


شكل (4): تنفيذ برنامج كونصول مع تعليق التنفيذ لحين الضغط على ENTER

استخدام تطبيقات النوافذ

عندما تختار تطبيقات النوافذ فإنك تبدأ بقسم الواجهة الذى يحتوى على فورمة النافذة (Windows Form). ويمكنك أن تبنى الواجهة بسحب أدوات التحكم من صندوق الأدوات ولصقها على الفورمة ، فيقوم المترجم بتسجيل حركاتك وتوليد الكود اللازم فى الخلفية.

والشكل (5) يوضح فورمة النافذة ومعالم بيئة تصميم التطبيقات النوافذية.



شكل (5) بيئة تصميم التطبيقات النوافذية

ومـــن الجدير بالذكر أنك تستطيع استخدام أكثر من فورمة واحدة فى نفس التطبيق.

أمـــا الوجـــه الآخـــر لبيئة التصميم فهو مشهد الكود الذى يكتب فى الخلفـــية. وبطبـــيعة الحـــال يمكـــنك أن تضيف إليه ما تشاء أثناء بناء التطبيق.

للتوصل إلى مشهد الكود قم بإجراء الخطوات التالية:

- اضـــغط على أيقونة الفورمة بالزر الأيمن فتظهر النافذة الموضحة بالشـــكل التالى. أختر منها View Code. وللعودة إلى مشهد التصميم اختر View Designer من نفس القائمة.

أو

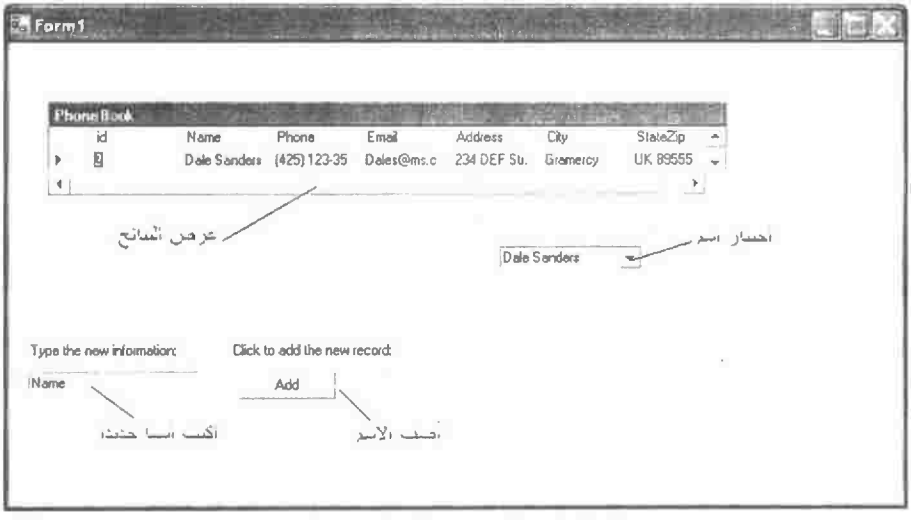
- افتح القائمة View واختر منها Code (أو اضغط على F7).

كمـــا يمكنك الانتقال إلى مشهد التصميم باختيار Designer من نفس القائمة (أو اضغط على Shift+F7).

شكل (6) الانتقال إلى مشهد الكود فى بيئة النوافذ

وعـــند تشـــغيل الـــبرنامج النوفذى فإنك ترى على الشاشة الفورمة بما تحتويه من أدوات تحكم جاهزة على الاستعمال. وفى الشكل التالى نرى

فورمـــة تنفيذ برنامج دليل تليفونات (Phone Book). وكما نرى فى هذا المثال أنه يمكنك أن تختار اسماً ما من صندوق الاسم (انظر الشكل 7) فـــتظهر المعلومات (id, Name, Phone, Address, City, StateZi p) فى صـــندوق القائمـــة (النـــتائج). كما يمكنك إضافة اسم جديد بكتابته فى صندوق النصوص بأسفل الشاشة.



شكل (7): تطبيق نوافذى لدفتر تليفون

استخدام تطبيقات الوب

عـــندما تختار تطبيقات الوب فإنك تبدأ بقسم الواجهة الذى يحتوى على فورمـــة الـــوب (Web Form). ويمكنك أن تبنى الواجهة بسحب أدوات التحكم من صندوق الأدوات ولصقها على الفورمة. وكما هو الحال مع تطبـــيقات الـــنوافذ ، فـــإن المترجم يقوم بتسجيل حركاتك وتوليد الكود اللازم فى الخلفية.

للتوصل إلى مشهد الكود قم بإجراء الخطوات التالية:

- اضغط على أيقونة الفورمة بالزر الأيمن ثم اختر من القائمة السريعة الأمر View Code. ولكى تعود إلى مشهد التصميم اختر View Designer من نفس القائمة.

أو

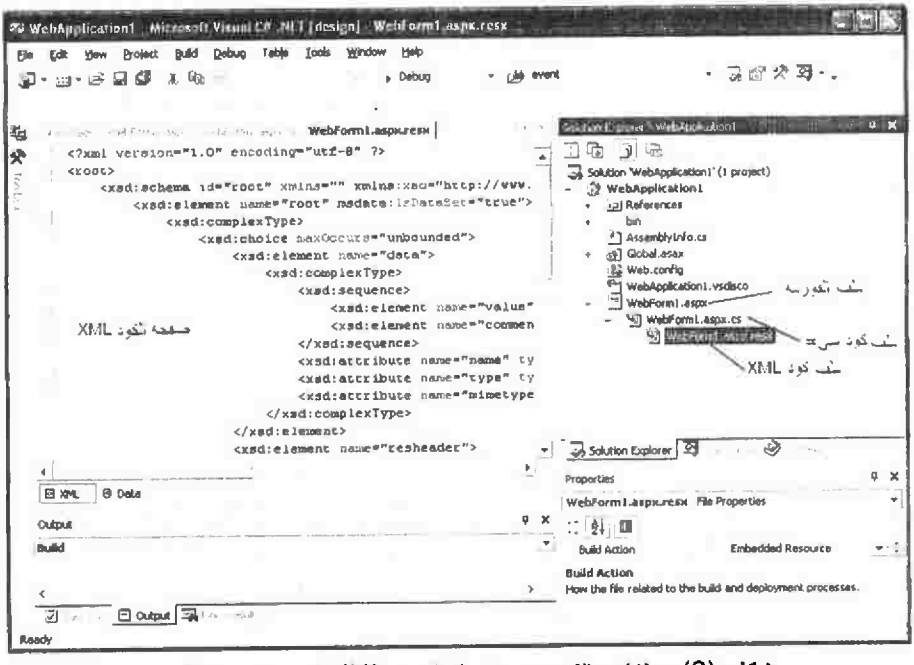
- باستخدام القائمة View اختر الأمر Code لمشهد الكود أو Designer لمشهد التصميم.

أو

- اضغط ضغطة مزدوجة على ملف الكود فتنتقل إلى مشهد الكود أو على ملف الفورمة فتنتقل إلى مشهد التصميم. أنظر الشكل (8) الذى يوضح فورمة الوب ومعالم بيئة التصميم.

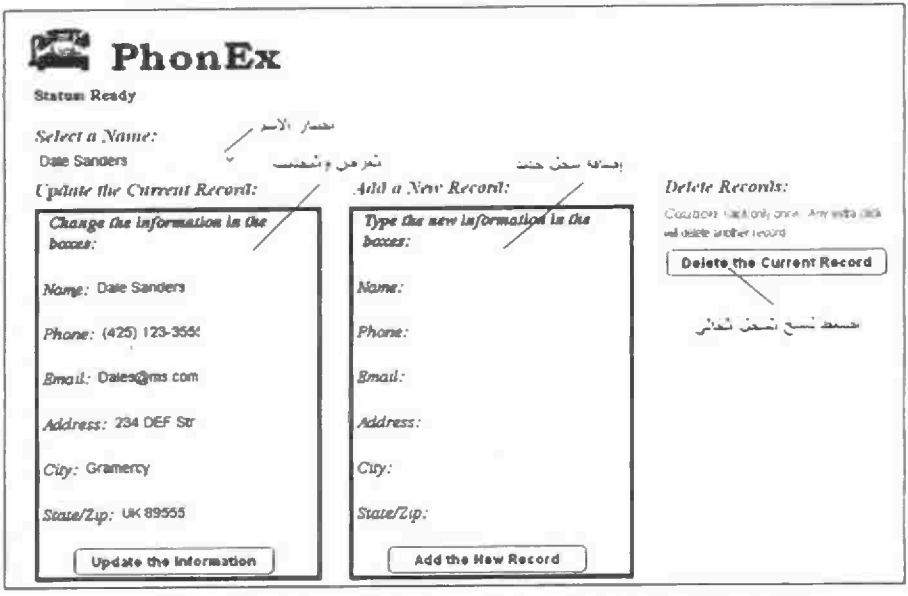
شكل (8): بيئة التصميم ونرى بها الفورمة والملفات

ونلاحـــظ فى تطبيقات الوب أن هناك صفحتين للكود واحدة بلغة سى# والأخـــرى بلغـــة XML وتـــتعاون الصفحتان على إنجاز التطبيق. أنظر الشكل (9).



شكل (9): ملفات الكود وصفحة الكود XML فى مشروع الوب

وعند تشغيل برنامج الوب فإن الفورمة تظهر بداخل كشاف الإنترنت (Internet Explorer) بما تحتويه من أدوات تحكم جاهزة على الاستعمال. وفى الشكل (10) نرى فورمة تنفيذ برنامج دليل تليفونات (Phone Book) وبهـــا مجموعة من أدوات التحكم لعرض وإضافة وحذف السجلات. ويمكنك ربط تطبيقات الوب وتطبيقات النوافذ بقاعدة بيانات مثل " SQL Server" لاسترجاع وعرض وتعديل البيانات كما فى هذا المثال.



شكل (10): مثال لتطبيق صفحة وب كما يظهر بداخل كشاف الإنترنت

فتح تطبيق موجود (قديم)

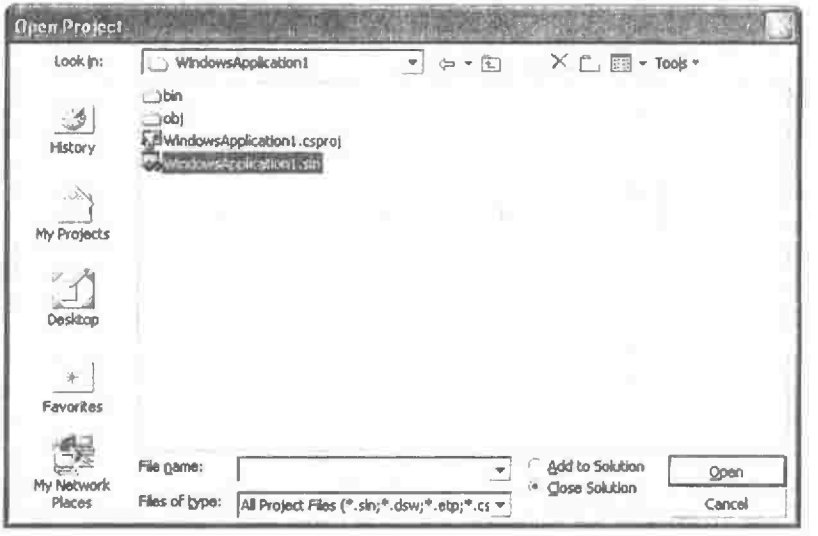
لكى تفتح تطبيقاً من المشروعات الموجودة على القرص استخدم قائمة الملفات كما هو موضح بعد.

فتح تطبيق موجود على القرص:

- ❖ افتح قائمة الملفات (Files)
- ❖ اختر Open
- ❖ اختر Project ــ أنظر الشكل (11).

وعندما يفتح ملف المشروع فإنك ترى ملفين:

- ملف المشروع ، ويتميز بالامتداد csproj.
- ملف الحل ، ويتميز بالامتداد sln.

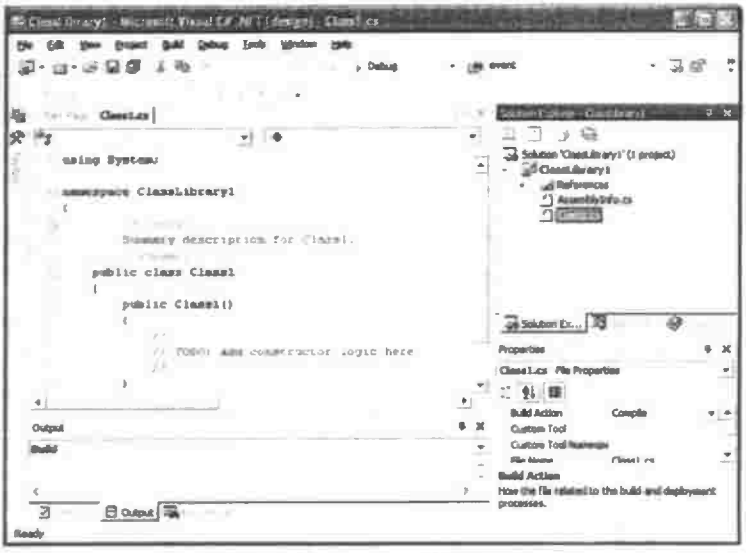


شكل (11) فتح مشروع موجود على القرص

وعـــادة فـــإن المؤشر يكون جاهزاً فوق ملف الحل. اختر ملف الحل ؛ ومـــع ذلـــك لو أنك أردت استخدام ملف المشروع فيمكنك أن تفعل ذلك أيضـــاً وسوف يتعرف الإستوديو على ملف الحل أتوماتيكياً. أما إذا لم يكـــن ملـــف الحل موجوداً فى الدوسيه فإن الإستوديو سوف يسألك أن تحفظ المشروع فى "حل" ما وسوف يطلب منك إدخال اسم هذا الحل.

ترجمة وبناء مشروع مكتبة

إن الفارق الأساسى بين مشروع الكونصول ومشروع المكتبة أن الكود الجاهـــز لمشروع المكتبة لا يحتوى على الأسلوب ()Main وبالتالى فلا يجوز تنفيذه. أنظر الشكل (12).



شكل (12): مشروع مكتبة DLL

لترجمة وبناء مشروع مكتبة اتبع الآتى:

- من القائمة Build اختر الأمر Build Solution أو Rebuild Solution إذا كان الحل يحتوى على مشروع مكتبة واحد.

أو

- إذا كان الحل يحتوى على أكثر من مشروع فيمكنك ترجمة كل مشروع على حدة باختيار الأمر Build ClassLibrary1, 2, ... (أو Rebuild ClassLibrary1, 2, ...). كما يمكنك بناء الحل كله كما فى الخطوة السابقة.

بهذا يتم ترجمة وبناء المشروع ، ويوضع الملف الناتج فى الدوسيه Debug للمشروع (لاحظ أن الاسم ClassLibrary1, 2, ... هو الاسم سابق التعريف وأنه يمكن تغييره) .

استخدام برنامج المكتبة ضمن المشروع الحالى

لنفرض أنك بنيت مكتبة بالاسم Factorial.dll تحتوى على أسلوب لإيجاد مضـــروب العدد الممر كبارامتر. ولنفرض أن نص المشروع كما يلى (لاحظ أن حيز الاسم يحمل اسم المكتبة تلقائياً.):

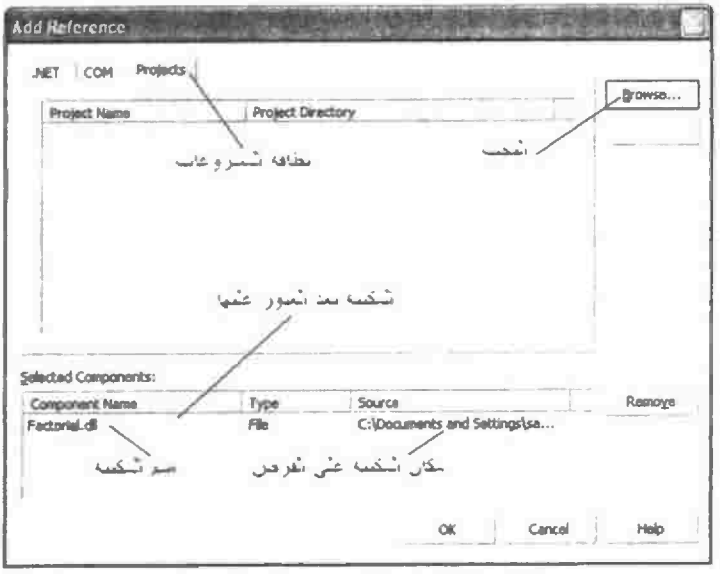
```
namespace Factorial
{
  public class Class1
  {
    public static long Fac(long i)
    {
      return ((i <= 1) ? 1 : (i * Fac(i    -1)));
    }
  }
}
```

ولـــنفرض أنـــك تريد استخدام هذا الأسلوب فى مشروع كونصول آخر بالاسم Client مثلاً ، بحيث لا تحتاج إلى برمجة الأسلوب ()Fac فى كل مرة تحتاج إليه ، بل تستدعيه مباشرة باسمه. لتحقيق ذلك افتح المشروع Client ثـــم أضـــف مـــرجعاً إلى المكتبة Factorial.dll باتباع الخطوات الوضحة بعد.

إضافة مرجع إلى المكتبة:

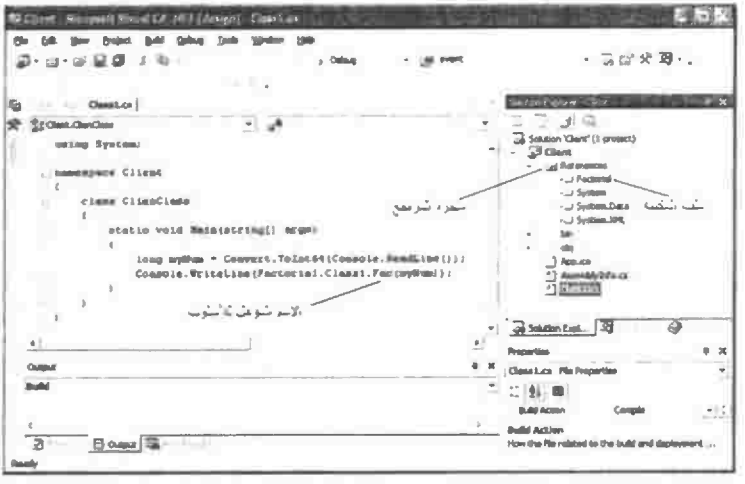
- ❖ مـــن القائمـــة Project اخـــتر Add Reference فتحصل على الشاشة الموضحة فى شكل (13).
- ❖ اضـــغط الـــبطاقة Projects ثـــم الزر Browse وابحث عن المكتبة Factorial.dll على القرص.

❖ اضغط على اسم المكتبة فتظهر فى النافذة كما هو موضح بالشكل.



شكل (13) إضافة مرجع إلى المكتبة

بعد إضافة مرجع المكتبة إلى المشروع سوف تلاحظ ظهور ملف المكتبة ضمن الشجرة References كما هو موضح بالشكل التالى.



شكل (14): برنامج المشروع بعد إضافة مرجع إلى المكتبة

اكتـــب الكـــود اللازم فى برنامجك لاستدعاء الأسلوب من المكتبة كهذا المثال:

```
using System;

namespace Client
{
  class ClienClass
  {
    static void Main(string[] args  )
    {
      long myNum = Convert.ToInt64(Console.ReadLine());
      Console.WriteLine(Factorial.Class1.Fac(myNum));
    }
  }
}
```

لاحـــظ فـــى هـــذا الكود أننا قد استخدمنا الاسم المؤهل للأسلوب Fac() بحيث يبدأ باسم الحيز (وهو نفسه اسم المكتبة) ثم الفصيلة ثم الأسلوب ، أى:

```
Factorial.Class1.Fac(myNum)
```

ولـــو أنك أنشأت المرجع قبل كتابة الكود (كما فى هذه الحالة) فسوف تـــتلقى بعض المساعدة من الإستوديو فى كتابة الكود باستخدام الخاصية IntelliSense التى تبحث عن أعضاء الفصائل أتوماتيكياً.