الباب الأول

استخدام أستوديو سى++

Microsoft Visual C++ 6.0

(١-١) الأستوديو المرئى طراز ٦

إذا كنت قادماً من طراز سابق من لغة سى++ المرئية (Visual C++) ، فإنك ستتحقق من الإضافات الجديدة التى أضافها الطراز 6.0 فقد أصبح البناء شامخاً. أما إذا كنت قادما من إحدى طرازات لغة سى أو سى++ الموجهة لنظام التشغيل دوس فإنك ستنتقل إلى عالم جديد متسع الأرجاء. فبرنامج لغة سى++ المرئية متعدد الأغراض ولا يقتصر عمله على ترجمة برامج سى أو سى++ ، بل هو مصنع هائل لكل تكنولوجيات الكومبيوتر فى العصر الحالى. ويأتى مترجم اللغة كأحد أجزاء البرنامج المتكامل الأستوديو المرئى (Visual Studio 6.0) الذى يصدر فى ثلاثة أحجام:

- الطراز القياسى (Standard)
- طراز المحترفين (Professional)
- طراز المصنع (Enterprise)

وتختلف محتويات كل طراز عن الآخر ، ويمثل طراز المصنع كل ما تنتجه شركة ميكروسوفت تحت اسم الأستوديو المرئى ، أما الطراز القياسى فيمثل الحد الأدنى من المنتج.

وهذه هى محتويات طراز المصنع التى تراها عند تركيب البرنامج لأول مرة ؛ وهى بصفة عامة تنقسم إلى مجموعة من لغات البرمجة بالإضافة إلى بعض الأدوات العامة التى يمكن استخدامها من خلال اللغات المختلفة.

لغات برمجة:

لغة بيسك المرئية طراز ٦	Microsoft Visual Basic 6.0

لغة سى++ طراز ٦	Microsoft Visual C++ 6.0
لغة فوكس ــ برو المرئية طراز ٦	Microsoft Visual FoxPro 6.0
لغة إنتريف المرئية طراز ٦	Microsoft Visual InterDev 6.0
لغة جافا++ المرئية طراز ٦ (قد تأتى على قرص مستقل)	Microsoft Visual j++ 6.0

مركبات عامة:

مركبات أكتيف ــ إكس: ActiveX

عبارة عن "قطع غيار" من البرامج الصغيرة التى يمكن تداولها وإضافتها إلى البرامج.

مركبات داتا ــ آكسيس: Data Access

تتضمن أجهزة قيادة (Drivers) لقواعد البيانات والمركبات الأخرى اللازمة لبرمجة قواعد البيانات.

أدوات المصنع: Enterprise Tools

تشمل الأدوات التى تستخدمها المؤسسات الكبيرة ، مثل:

- برنامج تأمين البرامج (Visual SourceSafe Microsoft).
- المستودع (Repository).
- برنامج تصميم النماذج (Visual Modeler).
- تنظيم المركبات المرئية (Visual Component Manager) ــ VCM... ألخ.

وهذه الموضوعات تعلوعن مستوى هذا الكتاب.

أدوات الرسم: Graphics

عبارة عن مجموعة من الرسومات (Bitmaps, Metafiles) والمؤشرات (Cursors) والأيقونات التى يمكنك استخدامها فى البرامج.

أدوات عامة: Tools

وهى عبارة عن مجموعة من أدوات البرمجة والمنافع العامة وتشمل:

- برنامج عرض دوال الوصلة البينية API (API Text Viewer).
- برنامج تتبع مؤسسة الفصائل (MFC Trace Utility).
- الجاسوس (برنامج صيانة) (Spy++).
- برنامج عرض مركبات OLE (OLE/Com Object Viewer).
- برنامج البحث عن الأخطاء (VC Error Lookup).
- أدوات SDK (Win 32 SDK Tools).
- وعاء اختبار مركبات أكتيف ــ إكس (ActiveX Control Test Container).
- برنامج معلومات ميكروسوفت (MS Info).

وجميع هذه الأدوات تخرج عن موضوع الكتاب ونتركها لاجتهاد القارئ.

ويمكنك أثناء تركيب البرنامج أن تختار بعض أو كل هذه المركبات. وبطبيعة الحال فإن ما تشاهده على شاشتك أثناء التركيب يعتمد على طراز الأستوديو الذى اشتريته (القياسى/ المحترفين / المصنع).

(١-٢) محتويات مترجم لغة سى++ المرئية VC++

يأتى أيضاً مترجم اللغة فى ثلاثة أحجام (القياسى و المحترفين والمصنع). ويشمل المترجم المركبات الرئيسية الآتية:

أستوديو VC++ للمبرمجين: البيئة المجمعة التى تدير منها دفة العمل. تحتوى على المترجم (Cpmpiler) والرابط (Linker) وأدوات الصيانة (Debugger) والسحرة (Wizards) وشاشات النجدة (Help).	VC++ Developer Studio
مكتبات VC++ للتشغيل: تمثل مجموعة المكتبات القياسية للغة علاوة على الكود المصدر (Source Code) لهذه المكتبات.	VC++ Runtime Libraries
مكتبات مؤسسة ميكروسوفت للفصائل (MFC) ، والنماذج (ATL): تشمل مكتبة MFC الموجهة لبرمجة النوافذ وهى تمثل موضوعنا الأساسى فى هذا الكتاب. أما المكتبة ATL فهى موجهة لبناء المركبات (COM) ولن نتعرض لها فى هذا الكتاب.	VC++ MFC and Template Libraries
أدوات بناء VC++: تستخدم هذه الأدوات تلقائياً أثناء العمل فى البرامج.	VC++ Build Tools

ملاحظة:

تأتى شاشة النجدة (Help) على قرص منفصل يحمل الاسم MSDN Library. وهى تحتوى على الكتب المصاحبة لمنتج الأستوديو المرئى ، وهى تسمى "كتب على الهواء" لأنها استبدلت الكتب المطبوعة.

ملاحظة:

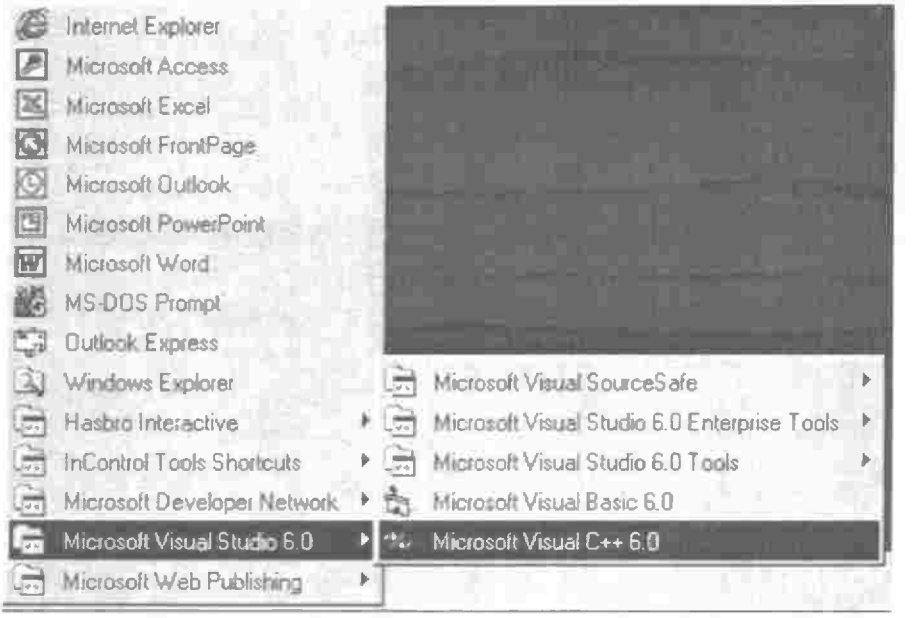
(١-٣) ترجمة وتشغيل برامج سى++ فى بيئة الأستوديو

هناك تعبيران متشابهان لا يجوز الخلط بينهما:

- الأستوديو المرئى (Visual Studio): وهو حزمة برامج لغات البرمجة التى تصدرها ميكروسوفت ، والتى تتضمن لغة سى++ المرئية.
- أستوديو VC++ للمبرمجين (VC++ Developer Studio): وهو البيئة التى نبنى فيها التطبيقات بلغة سى++. وسوف نكتفى بكلمة **الأستوديو** فى الفقرات القادمة.

تشغيل الأستوديو

- اضغط على زر البدء (Start).
- افتح قائمة البرامج (Programs).
- اختر الأستوديو المرئى (Microsoft Visual Studio) ، فتظهر قائمة فرعية على اليمين وبها أسماء برامج اللغات المختلفة التى قمت بتركيبها من قبل كما بالشكل التالى.



شكل (١-١) قائمة البرامج وبها برنامج لغة سى++ المرئية

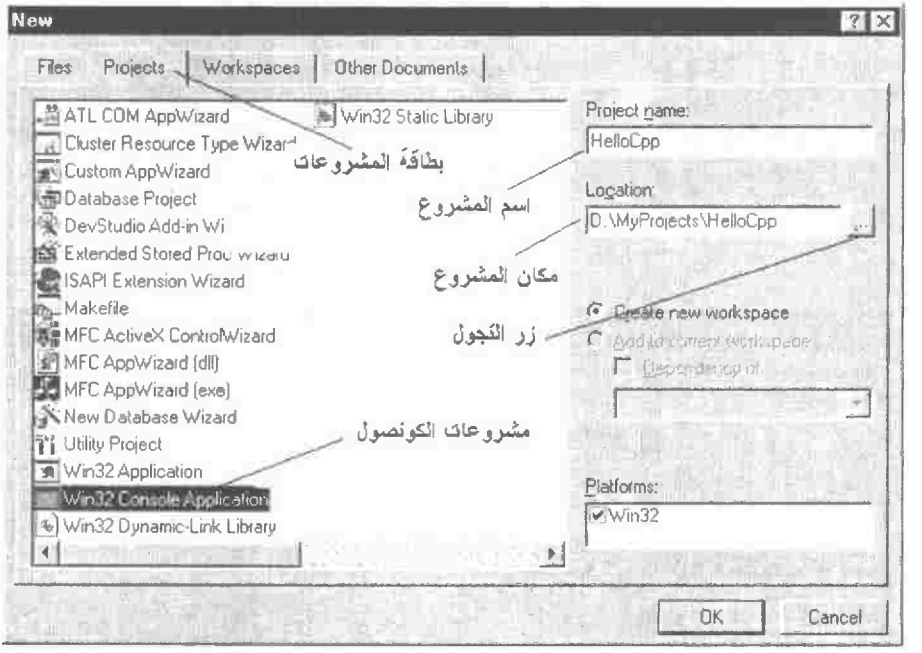
من الجائز أن يكون لديك لغة سى++ فقط ، وفى هذه الحالة لن تشاهد نفس المحتويات فى القائمة.

- اختر مترجم لغة سى++ المرئية (Microsoft Visual C++ 6.0) فيتم تشغيل الأستوديو.
- تظهر الشاشة خالية إلا من القائمة وسطر الأدوات فى انتظار بدء مشروع جديد.

بدء مشروع جديد

يتم بناء التطبيقات فى بيئة الأستوديو كمشروعات. وحتى تبدأ مشروعا جديدا اتبع الآتى:

- من قائمة الملفات (File) اختر الأمر New ، فتظهر الشاشة الموضحة بالشكل التالى. لا تتزعج! فإننا لن نتعرض لكل هذه الأنواع من المشروعات.



شكل (١-٢) نافذة بدء مشروع جديد (New)

- كنظرة سريعة إلى هذه النافذة ، فهى تحتوى على أربع بطاقات (Tabs):

١. الملفات (Files): تستخدم لفتح الملفات ، وإضافتها إلى المشروع الحالى.

٢. المشروعات (Projects): البطاقة سابقة التعريف التى تفتح عليها النافذة. وهى تستخدم لبناء مشروعات التطبيقات.

٣. حيز العمل (Workspaces): عندما يتم بناء مشروع فإنه ينتمى إلى حيز عمل معين كما سنرى. وفى إمكانك أن تبدأ حيزا خاليا أو تفتح حيز سبق استخدامه.

٤. وثائق أخرى (Other Documents): لفتح ملفات خاصة مثل الجداول الإليكترونية (Excel files) أو ملفات وورد (Word) إلى آخره.

- اختر Win32 Console Application من بطاقة المشروعات التى نصطلح عليها باسم مشروعات الكونصول وهى تناظر برامج دوس (DOS) التقليدية.

أنواع المشروعات:

سوف نتعرض فى هذا الكتاب إلى أنواع المشروعات الآتية:

١. Win32 Console Application: وهى مشروعات الكونصول (خط الأوامر) التقليدية. وسوف نستخدمها فى مراجعة لغة سى++.

٢. Win32 Application: وهى تطبيقات برمجة النوافذ باستخدام مكتبة الدوال API. وسوف نتعرف بها تعارفا سريعا لتكوين فكرة عن برمجة النوافذ.

٣. MFC Wizard Application (exe): وهى تطبيقات برمجة النوافذ بالاستعانة بمكتبة مؤسسة الفصائل MFC. وهذه هى الموضوع الأساسى للكتاب.

- استخدم الصندوق Location لاختيار مكان المشروع على القرص بتحديد اسم الممر الكامل مثل "D:\MyProjects\" كما يمكنك استخدام زر التجول الموجود على اليمين لاختيار الدوسيه من شجرة الفهارس.
- اكتب اسم المشروع فى الصندوق Project name وليكن:

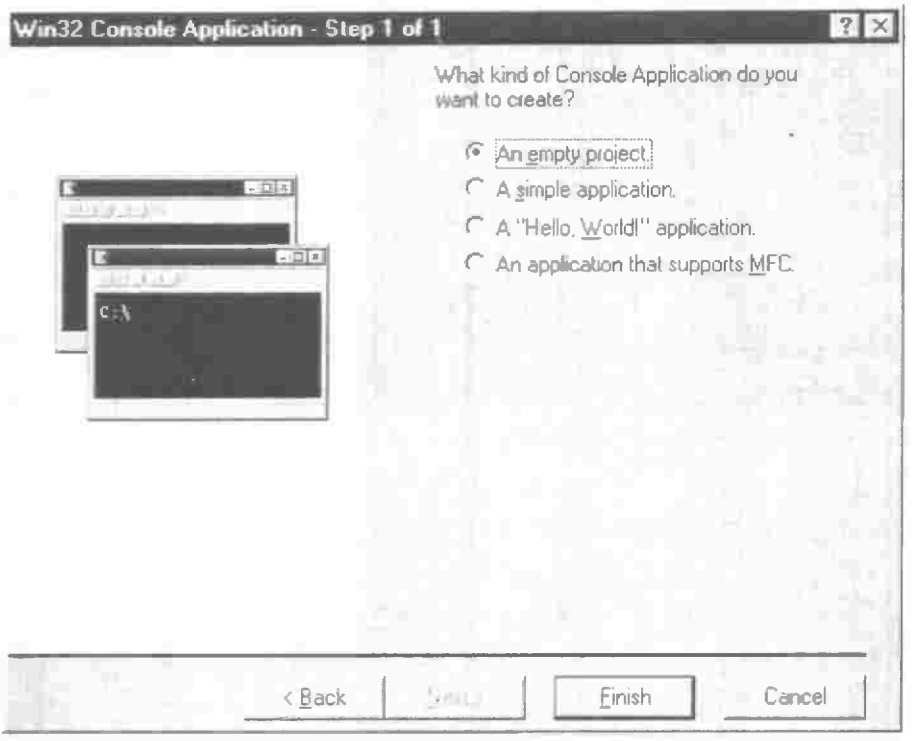
HelloCpp

وسوف تلاحظ أن البرنامج سوف ينسخ هذا الاسم فى الصندوق Location فيصبح "D:\MyProjects\HelloCpp".

١٦ بت ــ ٣٢ بت:

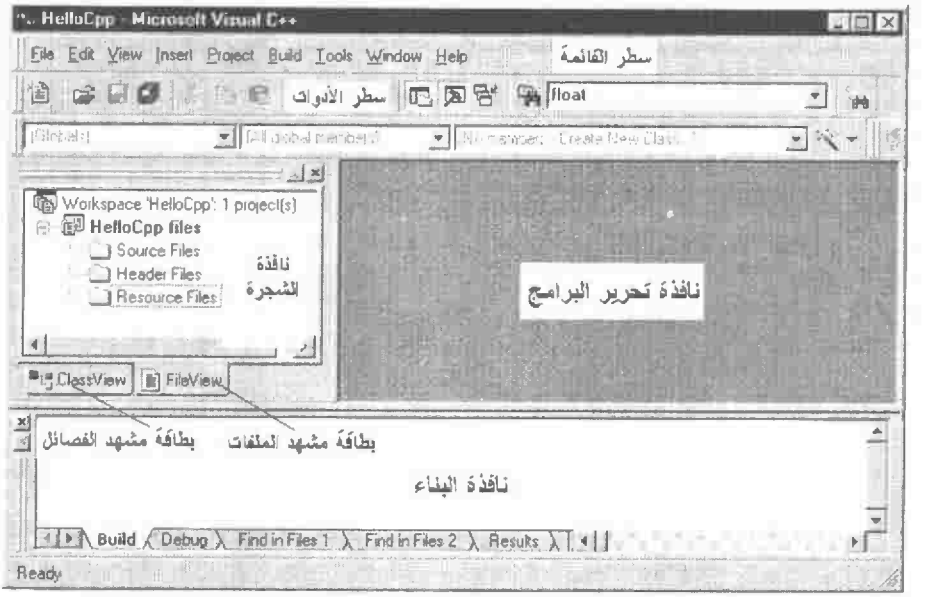
إن المشروعات التى نكتبها فى بيئة نوافذ 3.x أو فى بيئة نظام التشغيل دوس ، يطلق عليها اسم التطبيقات ١٦ بت. أما البرامج التى نكتبها باستخدام سى++ فى بيئة نوافذ ٩٥ (أو ما بعدها) فيطلق عليها اسم التطبيقات ٣٢ بت. والرقم ١٦ أو ٣٢ يمثل سعة المسجّلات (Registers) التى يستخدمها المعالج. ولذلك فإنك لا تستطيع تنفيذ البرامج ٣٢ بت فى بيئة دوس أو نوافذ 3..x ولكن العكس جائز.

- اضغط الزر OK فتظهر النافذة الموضحة بعد وبها مزيد من الاختيارات. وسوف نكتفى بالاختيار الأول الذى يمنحنا حيز عمل خالٍ نكتب فيه البرنامج (أما الاختيارات الأخرى فهى تمنحك برنامجاً جاهزاً تستطيع الإضافة إليه ، ويمكنك تجربتها بنفسك ومشاهدة النتائج).



شكل (١-٣) نافذة أنواع مشروعات الكونسول

- اضغط الزر Finish لإتمام العملية ، ثم اضغط الزر OK بالشاشة التى تتبع هذه الشاشة ، فتحصل على المشروع الموضح بالشكل التالى.

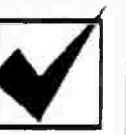


شكل (١-٤) نافذة المشروع الابتدائية

وكما نرى أن النافذة تحتوى على سطر للقائمة (Menu bar) ، وسطر للأدوات (Toolbar) ، كما تحتوى على ثلاث نوافذ فرعية:

١. **نافذة تحرير البرامج**: وهى تحتوى على محرر للنصوص ، وهى تظهر خالية مبدئيا ، وسوف نكتب فيها برامج سى++.

٢. **نافذة الشجرة**: وتحتوى على بطاقتين لتغيير المشهد ، مابين مشهد الملفات (File View) ، ومشهد الفصائل (Class View).



إن نافذة الشجرة تشبه شجرة الملفات فى بيئة الكشاف (Windows Explorer) ويمكنك فتح الدوسيهات بالضغط على العلامة + وإغلاقها بالضغط على العلامة -. وفى المشهد السابق فإن جميع الدوسيهات مفتوحة.

٣. **نافذة البناء**: وهى مخصصة للصيانة ولعرض تسلسل الأحداث أثناء الترجمة (Compilation) والربط (Linking).

بهذا يصبح المشروع جاهزا على استقبال أول برنامج بلغة سى++.

إضافة ملفات البرامج إلى المشروع

يمكنك أن تضم أحد الملفات الجاهزة إلى المشروع ، كما يمكنك أن تستخدم الأستوديو نفسه فى كتابة كود البرنامج.

لكتابة ملف جديد ، اتبع كالآتى:

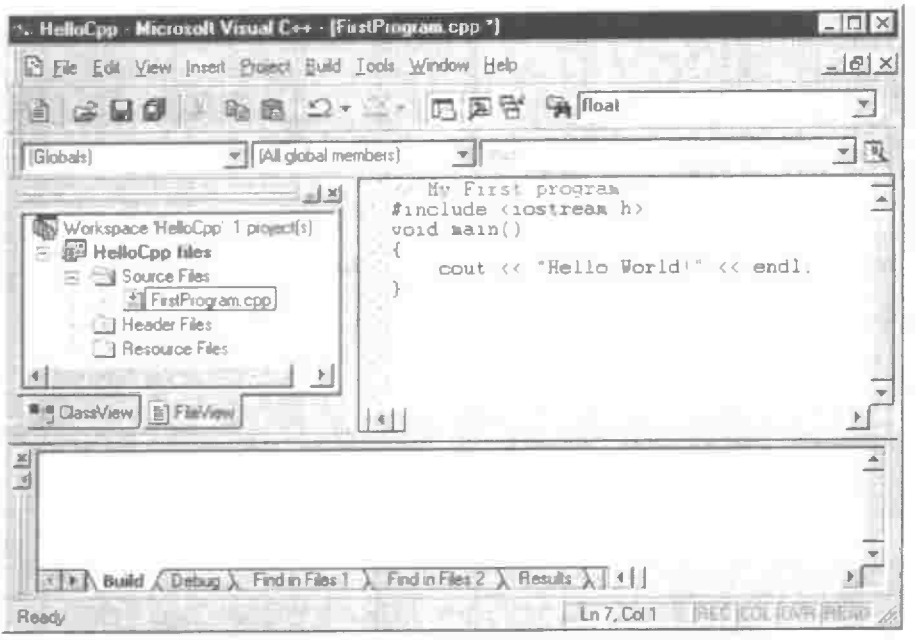
- استخدم أمر القائمة File - New (بمعنى افتح قائمة الملفات (File) ثم اختر منها الأمر New).
- اضغط على بطاقة الملفات (Files) فتحصل على الشاشة الموضحة بالشكل التالى.
- اختر من النافذة C++ Source File كما بالشكل التالى.

شكل (١-٥) إضافة ملف خال إلى المشروع

- اكتب اسما للملف في الصندوق File name وليكن "FirstProgram".
- اختر صندوق الاختبار Add to project (بحيث تظهر بداخله علامة صح √).
- اضغط الزر OK. بهذا ينضم إلى شجرة الملفات ملف فارغ بالاسم FirstProgram.cpp. اضغط على دوسيه هذا الملف بنافذة الشجرة ، ثم اضغط بالفأر على أى نقطة فى نافذة البرامج وابدأ فى الكتابة.
- اكتب البرنامج التالى فى نافذة البرامج:

```
// My First Program
#include <iostream.h>
void main()
{
        cout << "Hello World!" << endl;
}
```

وبهذا يصبح المشهد على شاشتك كما هو موضح بالشكل التالى.



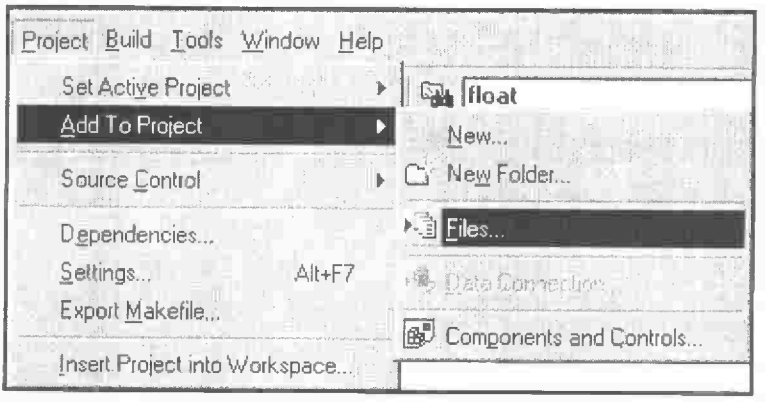
شكل (١-٦) الأستوديو بعد كتابة البرنامج

كما يمكنك كتابة البرنامج باستخدام أى برنامج من برامج تحرير النصوص مثل برنامج النوتة (Notepad) ، وحفظه بالامتداد .cpp ، ثم إضافته إلى المشروع ، أو إضافة أى ملف قد تكون كتبته من قبل.

لاستخدام ملف موجود أصلا ، اتبع الآتى:

١.بعد فتح المشروع الجديد ، اختر أمر المشروع (Project) من القائمة.

٢.اختر الأمر Add To Project. (انظر الشكل التالى.)



شكل (١-٧) إضافة ملف إلى المشروع

٣.من القائمة الفرعية ، اختر Files ، فتنتقل إلى نافذة اختيار الملف.

٤.باختيار الملف والضغط على الزر OK ينضم الملف إلى المشروع.

٥.إذا كان المشروع يستوعب أكثر من ملف واحد ، فيمكنك استخدام نفس الطريقة لإضافة أكثر من ملف إلى نفس المشروع.

ملخص الخطوات:

١.ابدأ مشروعا جديدا.

٢. افتح ملف مصدر (C++ Source File) جديد ، ثم اكتب البرنـــامج فى نافذة تحرير البرامج.

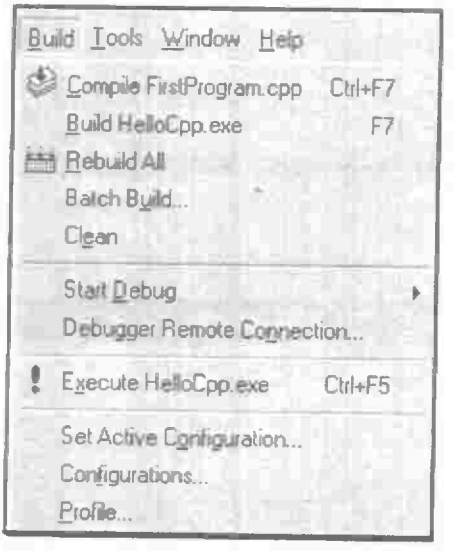
أو

أضف إلى المشروع أحد الملفات الموجودة على القرص.

بناء وتنفيذ البرنامج

تتضمن عملية البناء عمليتى الترجمة والربط. وتتم كالآتى:

- افتح القائمة Build الموضحة بالشكل التالى.



شكل (١-٨) قائمة البناء (Build)

- لاحظ وجود هذه الإمكانات بالقائمة:

١. ترجمة ملف المصدر: Compile FirstProgram.cpp

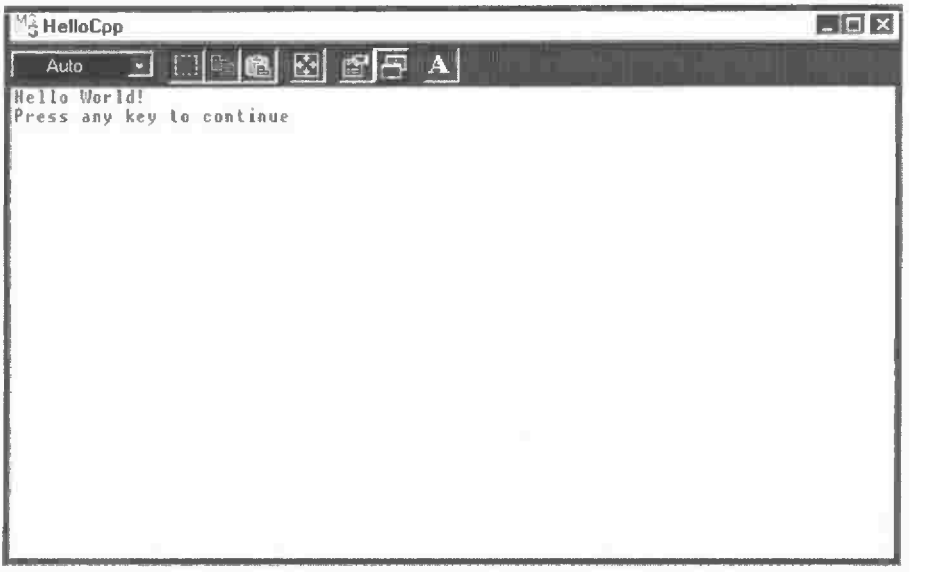
٢. بناء المشروع: Build FirstProgram.exe

٣. تنفيذ المشروع: Execute FirstProgram.exe

- يمكنك استخدام هذه الأوامر بالترتيب لإتمام عملية التنفيذ ، كما يمكنك اختيار الأمر الثالث مباشرة ، ويتولى البرنامج من نفسه عملية البناء ، بعد توجيه رسالة بهذا المضمون.

عندما يكون البرنامج مكونا من عدة ملفات ، فإن المترجم يتعرف تلقائيا على الملفات التى جرى عليها أى تعديل ، ويعيد ترجمتها ، بينما لا يترجم الملفات التى لم تتغير. فإذا أردت أن ترغم المترجم على ترجمة جميع الملفات ، ما تغير منها وما لم يتغير ، فإنك تستخدم أمر البناء: Build All.

بتنفيذ البرنامج السابق ترى النتيجة على الشاشة التالية ، وهى تسمى بيئة خط الأوامر أو الكونصول (وهى تشبه بيئة دوس التقليدية).



شكل (١-٩) تنفيذ البرنامج في بيئة الكونصول
(لاحظ أن الخلفية تظهر سوداء على شاشتك)

إذا كان البرنامج محتويا على أخطاء فى قواعد اللغة فإنك ترى تقريرا بهذه الأخطاء فى نافذة الصيانة أثناء عملية الترجمة. وعندما تضغط ضغطة مزدوجة على السطر المحتوى على رسالة الخطأ ، فإن البرنامج يرشدك إلى مكان الخطأ فى نافذة تحرير البرنامج. أنظر الشكل التالى ، حيث حذفنا علامة الفاصلة المنقوطة من نهاية البرنامج ، فتلقينا الرسالة الموضحة بالشكل التالى.

شكل (١-١٠) رسائل الأخطاء أثناء الترجمة

وبالطبع فإن الأستوديو يحتوى على وسائل كثيرة لصيانة البرامج ولكننا سوف نغفلها فى الوقت الحالى ، وسوف نقدم بعضها عند الحاجة إليها فى فصول الكتاب.

طريقة سريعة لتشغيل برامج الكونصول

إن أمثلة برامج الكونصول المقدمة فى هذا الكتاب (بالجزء الثانى) مكونة من ملف واحد يحمل الامتداد cpp. وفى بيئة النوافذ ، يرتبط هذا الامتداد ببرنامج الأستوديو. ومعنى ذلك أنك عندما تضغط على اسم الملف ضغطة مزدوجة يتم تشغيل الأستوديو (إذا لم يكن عاملا من الأصل) ويظهر الملف فى نافذة تحرير البرامج. ولذلك فإنك تستطيع تشغيل جميع البرامج بالطريقة الآتية:

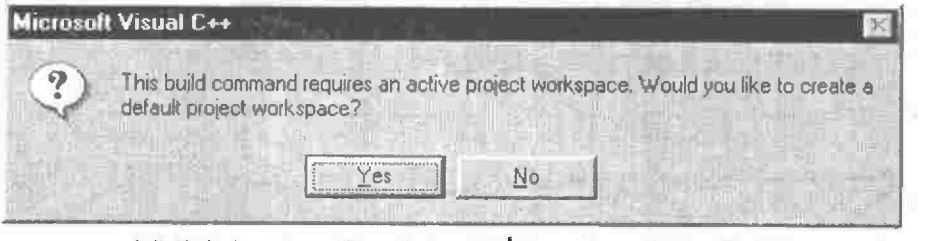
- اعرض الملف فى شاشة الكشاف.
- اضغط على اسم الملف ضغطة مزدوجة.
- استخدم أمر قائمة البناء:

Build

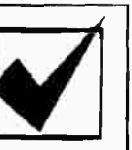
أو

Build All

- سوف تظهر نافذة مراجعة كما بالشكل التالى ، تخبرك بأن هذا الملف لا ينتمى إلى حيز عمل (Workspace) معين وأن الأستوديو يمكنه خلق حيز عمل سابق التعريف لهذا الملف حتى يمكن ترجمته. اضغط على الزر Yes فتتم عملية البناء.



شكل (١-١١) رسالة من الأستوديو: يلزم خلق حيز عمل لهذا الملف



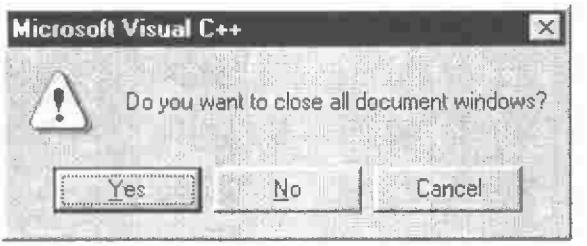
ربما لا يتضح لك تماما مفهوم حيز العمل (Workspace) حتى الآن ، لأن مشروعنا يتكون من ملف واحد. ولكننا عندما ننتقل إلى بيئة برمجة النوافذ فسوف تتحقق من أن أى برنامج نوافذى صغير (بما فى ذلك البرنامج "Hello World" النوافذى) يحتاج إلى موارد مختلفة يمدنا بها الأستوديو ولا يقتصر على سطور الكود التى نكتبها بلغة سى++.

إغلاق المشروع

يظل حيز العمل للمشروع الحالى مفتوحا حتى يتم إغلاق الأستوديو أو إغلاق المشروع تمهيدا لبدء مشروع جديد. ومن البديهى أنك لو حاولت تشغيل أى ملف (بضغطة مزدوجة عليه) قبل إغلاق المشروع فإنه سوف ينضم إلى المشروع المفتوح (بعد رسالة تنبيه بالطبع).

لكى تغلق المشروع الحالى اتبع الآتى:

- احفظ حيز العمل بأمر قائمة الملفات Save Workspace
- اختر أمر قائمة الملفات Close Workspace. (إذا لم تنفذ الخطوة الأولى فسوف يقوم الأستوديو بحفظ المشروع مع إرسال الرسالة المناسبة).
- سوف تظهر الرسالة الموضحة بالشكل على الشاشة ، حيث يسألك البرنامج إذا كنت ترغب فى إغلاق جميع نوافذ الوثائق. والإجابة المعتادة هى نعم (Yes). وبعدها يعود الأستوديو خاليا كما بدأ.

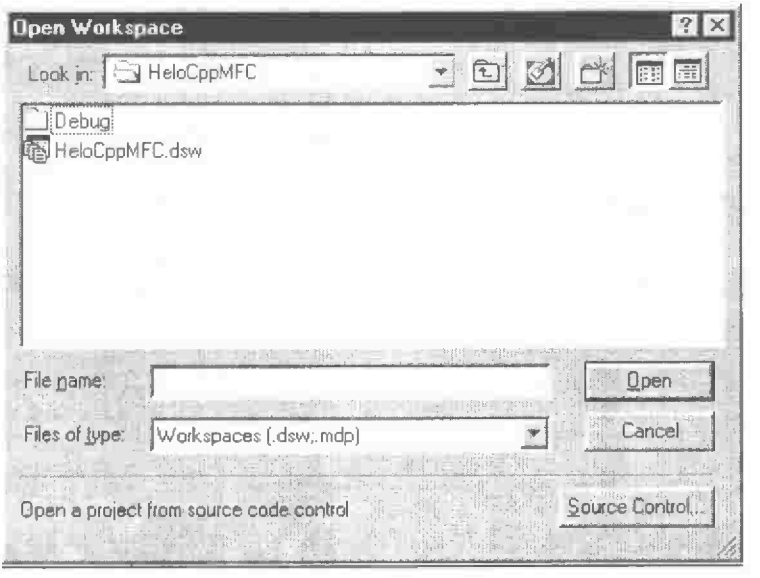


شكل (١-١٢) هل ترغب في إغلاق جميع الوثائق؟

أما إذا كنت ترغب فى استخدام الملفات المفتوحة فى مشروع أخـــر فيمكنك الإجابة بالنفى ، وفى هذه الحالة تصبح الملفات بلا مشـــروع وبلا حيز عمل ، ويمكنك أن تنشئ لها حيز عمل جديد عند بنائها.

فتح المشروعات القديمة

- استخدم أمر القائمة File - Open Workspace فتظهر نافذة فتـــح حيز العمل (Open Workspace) الموضحة بالشكل التالى.



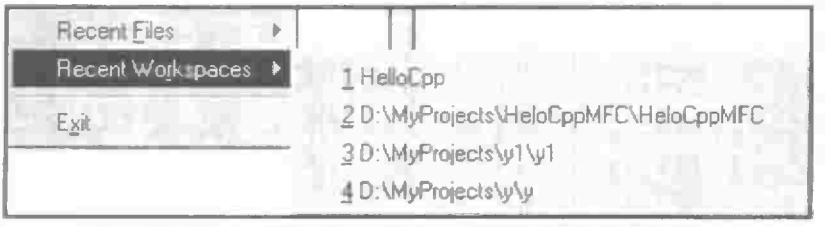
شكل(١-١٣) فتح مشروع قديم

- اختر المشروع من النافذة ، وسوف تلاحظ أن المشـــروعات تحمل الامتداد dsw أو mdp ، أما عدا ذلك فـــلا يظـــهر بـــهذه النافذة.
- اضغط الزر Open فيتم تحميل المشروع.

لاحظ انك لا تستطيع تحميل مشروع قديم باستخدام أمر لقائمـــة File-Open. إن هذا الأمر مخصص لفتح الملفات فقط.

طريقة سريعة لفتح المشروعات القديمة

تحتفظ قائمة الملفات بقائمة تاريخ (History List) لكل مـــن الملفـــات والمشروعات. ويمكنك فتح المشروعات ، وكذلك الملفات ، بسرعة من هذه القوائم بمجرد اختيار اسم المشروع أو الملف. أنظر الشـكل التالى.



شكل(١-١٤) قائمة التاريخ تحتوى على المشروعات القديمة

- لفتح مشروع قديم استخدم أمر القائمة:

File-Recent Workspaces

- لفتح ملف قديم استخدم أمر القائمة:

File-Recent Files

تذكر هذه المصطلحات

Developer Studio	الأستوديو / أستوديو المبرمجين
Microsoft Visual Studio	الأستوديو المرئى
Building	البناء (الترجمة والربط)
Compilation	الترجمة
Linking	الربط
Win32 Console Application	تطبيقات الكونصول ٣٢ بت
Win32 Applications	تطبيقات النوافذ ٣٢ بت
MFC Wizard Applications	تطبيقات ساحر "MFC"
Workspace	حيز عمل
Command Line	خط الأوامر
Application Programmer's Interface	دوال الوصلة البينية API
Console	كونصول
Visual C++ / VC++	لغة سى++ المرئية
Microsoft Foundation Class	مؤسسة الفصائل MFC
Project	مشروع

الجزء الثانى

دورة مكثفة فى لغة سى++

مفتتح

فى هذه الدورة المكثفة سوف نقدم ملخصاً للغة سى ++ بهدف المراجعة العامة لمبادئ اللغة قبل الدخول فى مجال برمجة النوافذ. كما أنها تمدك بحصيلة من المصطلحات المعربة التى سوف نستخدمها فى الكتاب. والتى قد تختلف قليلاً عن حصيلتك من المصطلحات العربية إذا كنت قادماً من أرض مختلفة!

وليكن معلوما أن هذه الدورة المكثفة لا تكفى من كان جديداً على لغة سى++. فإذا كان هذا هو الحال ، فلابد لك من قراءة الجزء الأول من كتاب *ثلاث قمم فى كتاب واحد*" للمؤلف. أما إذا كنت تجيد لغة سى++ فلتتخطى هذا الجزء ولتبدأ بالجزء الثانى مباشرة.