الفصل السابع

استخدام أدوات السرد

List Controls

تعد طريقة تمثيل البيانات داخل البرنامج عامل أساسى فى تقبل المستخدم للبرنامج وتفهمه لخطواته وطريقة عمله، لذا يجب أن نختار الأداة المناسبة لتمثيل كل جزء من أجزاء البرنامج. سنقوم فى هذا الفصل بالتعرف على مجموعة الأدوات المستخدمة لحصر مجموعة من العناصر.

بانتهاء هذا الفصل ستتعرف على:

- عرض قوائم البيانات داخل المربعات الحوارية.
- ترتيب البيانات داخل شجرة البيانات.
- اختيار أكثر من عنصر داخل القائمة.
- الربط بين القوائم المختلفة داخل المربع الحوارى.

عند العمل مع برامج النوافذ، تلاحظ أنه كثيراً ما يتم تمثيل البيانات فى صورة قوائم مثل المجلدات والملفات وأسماء الخطوط والأنماط.ولأن الطريقة المثلى لتمثيل البيانات فى صورة قائمة تعتمد أساساً على نوعية هذه البيانات وعلى ما إذا كنت ستختار عنصراً واحداً فقط أم أكثر من عنصر من هذه القائمة، لذا هناك أربعة أنواع من أدوات السرد، لكل منها الخصائص والأنماط التى تميزه عن باقى الأنواع، وهى مربع السرد والتحرير Combo Box ومربع السرد List Box وشجرة البيانات Tree وأخيرا أداة القائمة List Control. وفى هذا الفصل سنقوم بالتعرف على كيفية إضافة هذه الأنواع والفرق بينها وكيفية عملها داخل البرنامج مع أدوات التحكم الأخرى. قم بإنشاء مشروع حوارى جديد باسم Lists بنفس الطريقة التى ذكرناها فى الفصول السابقة لاستخدامه فى شرح محتويات هذا الفصل.

استخدام مربع السرد والتحرير

تحتوى هذه الأداة أساساً على ثلاث أدوات هى مربع نص ومربع سرد وزر أمر، لذا أطلق عليها اسم مربع السرد والتحرير Combo Box والذى يقوم بعرض قائمة من الخيارات بحيث يمكنك اختيار عنصر واحد فقط منها، حيث تتميز عن باقى أدوات السرد فى ظهور العنصر المختار دائماً أعلى المربع.

يوجد ثلاثة أنواع من مربعات السرد والتحرير كما فى جدول ٧-١، حيث يتم التحكم فى هذه الأنواع من خلال الخاصية Type بمربع الخصائص المصاحب لمربع السرد والتحرير.

جدول ٧-١ أنواع مربع السرد والتحرير

النوع	المواصفات
Simple	يحتوى على مربع نص ومربع قائمة
Dropdown	يحتوى على مربع نص وزر ومربع قائمة
Drop List	يحتوى على أداة عنوان وزر ومربع قائمة

لإضافة مربع السرد والتحرير إلى المربع الحوارى **Lists**، تابع معنا الخطوات الآتية:

١. قم بفتح قالب المربع الحوارى **IDD_LISTS_DIALOG** داخل نافذة محرر الموارد ثم قم بحذف النص الموجود بالمربع الحوارى.

٢. قم أيضاً بحذف زر **Cancel** وانقل زر **Ok** من مكانه إلى الركن الأيمن السفلى من المربع الحوارى.

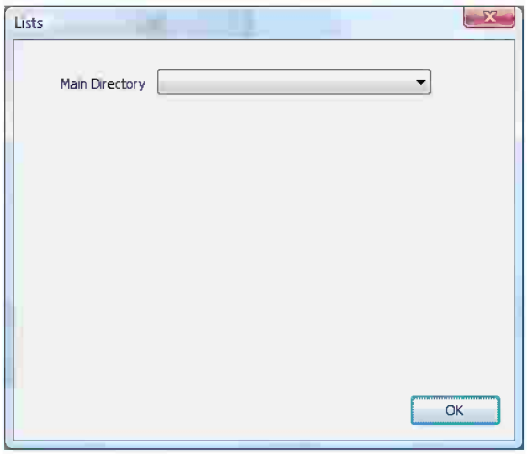
٣. اختر رمز أداة العنوان من مربع الأدوات ثم قم بإضافة عنوان إلى الركن الأيسر العلوى من المربع الحوارى. قم بتغيير عنوان الأداة إلى **Main Directory**.

٤. اختر رمز مربع السرد والتحرير

من مربع الأدوات ثم قم بإضافة المربع يمين العنوان **Main Directory**. قم بزيادة طول مربع السرد والتحرير إلى يمين المربع الحوارى.

٥. قم بإدخال اسم مربع السرد والتحرير إلى الخاصية **ID** وليكن **IDC_MAIN_DIR**.

٦. اختر **Drop List** من الخاصية **Type** بمربع الخصائص. يجب أن يظهر المربع الحوارى الآن كما فى شكل ٧-١.



شكل ٧-١ إضافة مربع السرد والتحرير إلى المربع الحوارى

بمجرد إضافة مربع السرد والتحرير إلى المربع الحوارى، تلاحظ تخصيص القيمة True للخاصية Sort داخل مربع الخصائص تلقائياً، وهذا يعنى أنه سيتم ترتيب عناصر الأداة ترتيباً أبجدياً. فإذا لم يرق لك ذلك، قم بتخصيص القيمة False للخاصية Sort من مربع الخصائص.

قد لا يكون حجم القائمة المنسدلة كافٍ لعرض البيانات كاملة. لتغيير حجم مربع السرد والتحرير، انقر السهم الموجود يمين المربع، يظهر إطار يبين حجم القائمة المنسدلة. قم بإعادة تحجيم الأداة باستخدام أزرار التحجيم.

كما تعودنا فى الفصول السابقة ، للعمل مع أى أداة تحكم يجب ربطها بمتغير يحتوى على قيمة الأداة. لربط مربع السرد والتحرير الذى أضفناه منذ قليل بمتغير، تابع معنا الخطوات الآتية:

١. نشط التبويب Class View من نافذة عمل المشروع.
٢. انقر التصنيف CListsDlg بزر الفأرة الأيمن ثم اختر Add ➢ Add Variable من القائمة الموضعية، يظهر معالج إضافة متغير جديد.
٣. قم بتنشيط مربع الاختيار Control variable لأننا نرغب فى إنشاء متغير مرتبط بإحدى أدوات التحكم الموجودة بالمربع الحوارى.
٤. اختر أداة مربع السرد IDC_MAIN_DIR من مربع السرد Control ID.
٥. تأكد من اختيار Control من مربع السرد Category لأننا نحتاج إلى التعامل مع الخصائص الأخرى المصاحبة لهذه الأداة.
٦. تأكد من اختيار نوع البيانات CComboBox بمربع السرد والتحرير Variable Type.
٧. اكتب اسم المتغير فى خانة Variable name وليكن m_cbMainDir واختر درجة المتغير من مربع السرد Access وهو Public فى هذه الحالة.
٨. انقر زر Finish لإغلاق نافذة المعالج.

تحتوى لغة Visual C++ على مربع سرد وتحرير محسَّن يسمى Extended ComboBox يختلف عن مربع السرد والتحرير المعتاد فى إمكانية إضافة الصور. سنتعرض بالتفصيل لهذه الأداة فى مكان آخر من الكتاب.

استخدام شجرة البيانات

تتميز شجرة البيانات بعرض العناصر فى صورة هرمية تبدأ من اليسار متجهةً إلى اليمين. يوجد العنصر الرئيسى للشجرة فى أقصى اليسار ويسمى Root node. كما يوجد العنصر الأخير الذى لا يحتوى على عناصر فرعية فى أقصى اليمين ويسمى Leaf node ، بينما تسمى العناصر الموجودة بين العنصر الأول والعنصر الأخير Branch nodes.

يمكنك افتراضياً اختيار عنصر واحد فقط من الشجرة، فإذا أردت اختيار أكثر من عنصر فى نفس الوقت، قم بكتابة الكود اللازم لذلك كما سنرى فى الجزء الثانى من هذا الفصل.

لإضافة شجرة بيانات إلى المربع الحوارى Lists، تابع معنا الخطوات الآتية:

١. قم بفتح قالب المربع الحوارى IDD_LISTS_DIALOG داخل نافذة محرر الموارد.

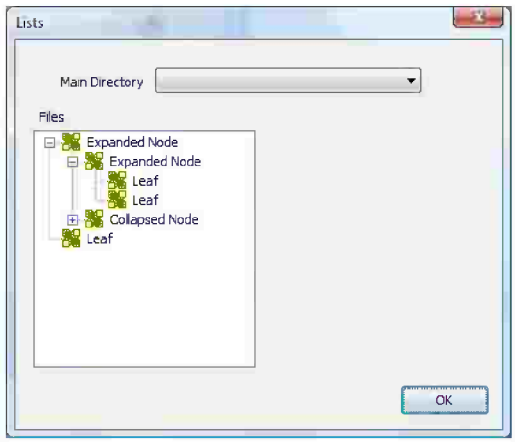
٢. اختر رمز أداة العنوان من مربع الأدوات ثم قم بإضافة عنوان للشجرة أسفل العنوان Main Directory وقم بتغيير العنوان إلى Files (انظر شكل ٧-٢).

٣. اختر رمز شجرة البيانات

من مربع الأدوات ثم قم بإضافة شجرة يسار المربع الحوارى وأسفل Files.

٤. قم بإدخال اسم الشجرة إلى الخاصية ID بمربع الخصائص وليكن IDC_FILES_TREE.

٥. قم بتخصيص القيمة True للخصائص Has buttons وHas lines وLines at root وScroll داخل مربع الخصائص. يجب أن يظهر المربع الحوارى الآن كما فى شكل ٧-٢.



شكل ٧-٢ إضافة شجرة البيانات إلى المربع الحوارى

لربط شجرة البيانات بمتغير، تابع معنا الخطوات الآتية:

١. نشط التبويب **Class View** من نافذة عمل المشروع.

٢. انقر التصنيف **CListsDlg** بزر الفأرة الأيمن ثم اختر **Add** ➢ **Add Variable** من القائمة الموضعية، يظهر معالج إضافة متغير جديد.

٣. قم بتنشيط مربع الاختيار **Control variable** لأننا نرغب فى إنشاء متغير مرتبط بإحدى أدوات التحكم الموجودة بالمربع الحوارى.

٤. اختر أداة شجرة البيانات **IDC_FILES_TREE** من مربع السرد **Control ID**.

٥. تأكد من اختيار **Control** من مربع السرد **Category** لأننا نحتاج إلى التعامل مع الخصائص الأخرى المصاحبة لهذه الأداة.

٦. تأكد من اختيار نوع البيانات **CTreeCtrl** بمربع السرد والتحرير **Variable Type**.

٧. اكتب اسم المتغير فى خانة **Variable name** ولـيكن **m_treeFiles** واختـر درجة المتغير من مربع السرد **Access** وهو **Public** فى هذه الحالة.

٨. انقر زر **Finish** لإغلاق نافذة المعالج.

استخدام مربع السرد

يشتمل مربع السرد List Box في أبسط صوره على قائمة من العناصر، إلا أنها؛ بعكس شجرة البيانات أو مربع السرد والتحرير، تتيح للمستخدم اختيار أكثر من عنصر داخل القائمة. يوجد أربعة أنواع من الخيارات المتاحة عند العمل مع مربع السرد كما بالجدول ٧-٢ التالى والتى يمكنك اختيار أى منها من الخاصية Selection بمربع الخصائص.

جدول ٧-٢ أنواع مربع السرد

النوع	المواصفات
Single	يتيح اختيار عنصر واحد فقط
Multiple	يتيح اختيار أكثر من عنصر باستخدام الفأرة ومفتاح Ctrl أو Shift
Extended	يتيح اختيار أكثر من عنصر باستخدام الفأرة فقط
None	لا يتيح اختيار أى عنصر

كما فى مربع السرد والتحرير، بمجرد إضافة مربع سرد إلى المربع الحوارى، يتم تخصيص القيمة True للخاصية Sort تلقائياً، وهذا يعنى ترتيب عناصر مربع السرد ترتيباً أبجديا. فإذا لم يرق لك ذلك، يمكنك تخصيص القيمة True لهذه الخاصية. لإضافة مربع سرد إلى المربع الحوارى، تابع معنا الخطوات الآتية:

١. قم بفتح قالب المربع الحوارى IDD_LISTS_DIALOG داخل نافذة محرر الموارد.

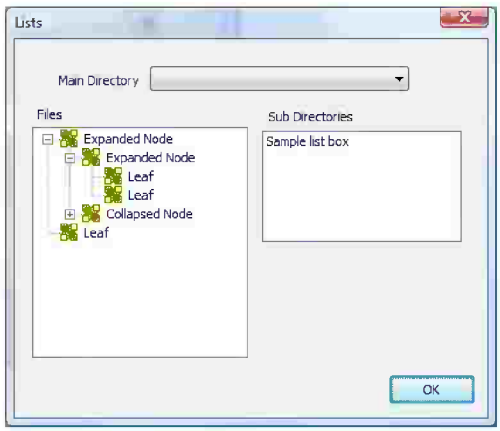
٢. اختر رمز أداة العنوان من مربع الأدوات ثم قم بإضافة عنوان لمربع السرد بمحاذاة العنوان Files وقم بتغيير العنوان إلى Sub Directories.

٣. اختر رمز مربع السرد List Box من مربع الأدوات ثم قم بإضافة مربع سرد أسفل العنوان Sub Directories (انظر شكل ٧-٣).

٤. قم بإدخال اسم مربع السرد إلى الخاصية ID بمربع الخصائص وليكن IDC_SUB_DIR.

٥. قم بتخصيص القيمة **Extended** للخاصية **Selection** بمربع الخصائص.

٦. قم بتخصيص القيمة **False** للخاصية **No Integral Height** حتى يتم حساب ارتفاع مربع السرد تبعاً للعناصر الموجودة فيه. يجب أن يظهر المربع الحوارى الآن كما فى شكل ٧-٣.



شكل ٧-٣ إضافة مربع السرد إلى المربع الحوارى

لربط مربع السرد بمتغير، تابع معنا الخطوات الآتية:

١. نشط التبويب **Class View** من نافذة عمل المشروع.

٢. انقر التصنيف **CListsDlg** بزر الفأرة الأيمن ثم اختر **Add** ➢ **Add Variable** من القائمة الموضعية، يظهر معالج إضافة متغير جديد.

٣. قم بتنشيط مربع الاختيار **Control variable** لأننا نرغب فى إنشاء متغير مرتبط بإحدى أدوات التحكم الموجودة بالمربع الحوارى.

٤. اختر أداة شجرة البيانات **IDC_SUB_DIRS** من مربع السرد **Control ID**.

٥. تأكد من اختيار **Control** من مربع السرد **Category** لأننا نحتاج إلى التعامل مع الخصائص الأخرى المصاحبة لهذه الأداة.

٦.

٧. تأكد من اختيار نوع البيانات CListBox بمربع السرد والتحرير Variable Type.

٨. اكتب اسم المتغير فى خانة Variable name وليكن m_lbSubDir واختر درجة المتغير من مربع السرد Access وهو Public فى هذه الحالة.

٩. انقر زر Finish لإغلاق نافذة المعالج.

استخدام أداة القائمة

تعتبر أداة القائمة أكثر الأنواع الأربعة تعقيداً، كما تستخدم بكثرة داخل نوافذ العرض أكثر منها فى المربعات الحوارية، كما أن لها القدرة على إظهار الصور إلى جانب النص ولعل مستكشف Windows وجهاز الكمبيوتر خير مثال لأداة القائمة. يوجد أربعة أنماط عرض لأداة القائمة والموضحة بجدول ٥-٣ والتى يمكنك اختيار أيٍ منها من الخاصية View بمربع الخصائص.

جدول ٥-٣ أنواع أداة القائمة

النوع	المواصفات
Icon	تقوم بإظهار أيقونات كبيرة ويكون النص أسفل الأيقونة ويكون الترتيب بالعرض ثم لأسفل
Small Icon	تقوم بإظهار أيقونات صغيرة ويكون النص يمين الأيقونة ويكون الترتيب بالعرض ثم لأسفل
List	مثل النوع السابق إلا أن الترتيب يكون لأسفل ثم بالعرض
Report	تظهر معلومات العناصر داخل أعمدة لها رؤوس

عندما تفقد أداة القائمة التركيز، يتم إلغاء العنصر المختار. فإذا أردت أن يكون العنصر المختار مرئى دائماً، قم بتخصيص القيمة True للخاصية Show Selection always داخل مربع الخصائص.

لإضافة أداة القائمة إلى المربع الحوارى، تابع معنا الخطوات الآتية:

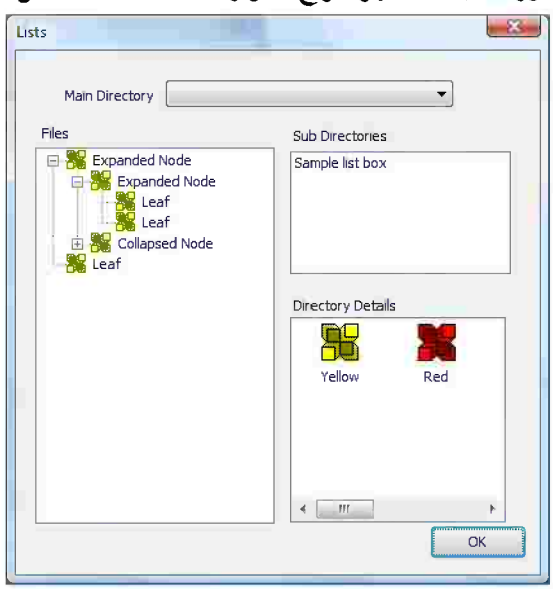
١. قم بفتح قالب المربع الحوارى **IDD_LISTS_DIALOG** داخل نافذة محرر الموارد.

٢. اختر رمز أداة العنوان من مربع الأدوات ثم قم بإضافة عنوان لأداة القائمة أسفل مربع السرد وقم بتغيير العنوان إلى **Directory Details**.

٣. اختر رمز أداة القائمة List Control من مربع الأدوات ثم قم بإضافة أداة قائمة أسفل العنوان **Directory Details** (انظر شكل ٧-٤).

٤. قم بإدخال اسم أداة القائمة إلى الخاصية **ID** بمربع الخصائص وليكن **IDC_DIR_DET**.

٥. قم بتخصيص القيمة **Report** إلى الخاصية **View** وذلك باختيارها من مربع السرد والتحرير المجاور. يجب أن يظهر المربع الحوارى الآن كما فى شكل ٧-٤.



شكل ٧-٤ إضافة أداة القائمة إلى المربع الحوارى

لربط أداة القائمة بمتغير، تابع معنا الخطوات الآتية:

١. نشط التبويب Class View من نافذة عمل المشروع.
٢. انقر التصنيف CListsDlg بزر الفأرة الأيمن ثم اختر Add Add Variable من القائمة الموضعية، يظهر معالج إضافة متغير جديد.
٣. قم بتنشيط مربع الاختيار Control variable لأننا نرغب فى إنشاء متغير مرتبط بإحدى أدوات التحكم الموجودة بالمربع الحوارى.
٤. اختر أداة القائمة IDC_DIR_DET من مربع السرد Control ID.
٥. تأكد من اختيار Control من مربع السرد Category لأننا نحتاج إلى التعامل مع الخصائص الأخرى المصاحبة لهذه الأداة.
٦. تأكد من اختيار نوع البيانات CListCtrl بمربع السرد والتحرير Variable Type.
٧. اكتب اسم المتغير فى خانة Variable name وليكن m_lbDirDetails واختر درجة المتغير من مربع السرد Access وهو Public فى هذه الحالة.
٨. انقر زر Finish لإغلاق نافذة المعالج.

إضافة العناصر لأدوات السرد

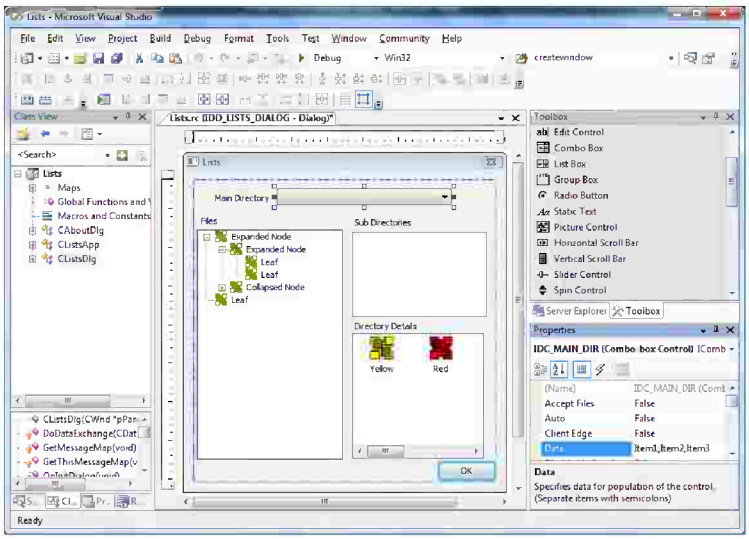
تعتبر عملية إدخال العناصر إلى أدوات السرد أولى خطوات برمجة تلك الأدوات. ورغم التقارب الشديد بين الطرق المتبعة لإدخال العناصر إلى أدوات السرد الأربعة، إلا أن لكل أداة بعض المتطلبات الخاصة التى تختلف فيها عن باقى الأدوات وهو ما سنتعرف عليه من خلال الجزء المتبقى من هذا الفصل.

ملء مربع السرد والتحرير بالبيانات

يعد مربع السرد والتحرير هو أداة السرد الوحيدة التى يمكن إدخال عناصرها أثناء التصميم من خلال مربع الخصائص وذلك كما يلى (انظر شكل ٧-٥):

١. قم بتنشيط مربع السرد والتحرير الموجود بالمربع الحوارى.
٢. استخدم شريط التمرير للوصول إلى الخاصية Data داخل مربع الخصائص.

٣. قم بكتابة عناصر مربع السرد واحداً تلو الآخر باستخدام الفاصلة المنقوطة ";" لفصل هذه العناصر.



شكل ٧-٥ ملء مربع السرد والتحرير باستخدام مربع الخصائص

وهذه الطريقة نادرة الاستخدام، حيث يتم غالباً إدخال العناصر أثناء تنفيذ البرنامج من خلال الدالة **OnInitDialog()** التى يتم استدعاؤها بمجرد فتح المربع الحوارى وقبل ظهور مربع السرد والتحرير.

يطلق على التصنيف الذى يحتوى على وظائف مربع السرد والتحرير داخل مكتبة MFC اسم **CComboBox** ويحتوى على مجموعة من الدوال التى تختص بإضافة وحذف العناصر والموضحة بجدول ٧-٤.

جدول ٧-٤ الدوال المستخدمة لإضافة العناصر لمربع السرد والتحرير

الدالة	الاستخدام
AddString()	إضافة عنصر إلى نهاية المربع أو فى ترتيبه الصحيح
DeleteString()	حذف عنصر من المربع
InsertString()	إضافة عنصر لمكان معين داخل المربع

الدالة	الاستخدام
ResetContent()	حذف جميع عناصر المربع
Dir()	إضافة أسماء الملفات كعناصر عادية داخل المربع

فى المثال الذى بين أيدينا، تقوم الدالة OnInitDialog() بملء مربع السرد والتحرير ببعض المجلدات الرئيسية. قم بإضافة دالة جديدة إلى التصنيف CListsDlg باسم PopulateCombo() وهى من النوع void وذلك باستخدام معالج إضافة الدالة ثم قم بتنشيط التبويب Class View وانقر الدالة OnInitDialog() الموجودة أسفل التصنيف CListDlg نقراً مزدوجاً وقم بتعديل كود هذه الدالة وإنشاء كود الدالة PopulateCombo() كما يلى (اختر التصنيف CListDlg من الجزء العلوى من التبويب ثم انقر الدالة OnInitDialog() من الجزء السفلى نقراً مزدوجاً):

```
1.  BOOL CListsDlg::OnInitDialog()
2.  {
3.    CDialog::OnInitDialog();

4.    // Add "About..." menu item to system menu.
5.    // IDM_ABOUTBOX must be in the system command
        range.
6.    ASSERT((IDM_ABOUTBOX & 0xFFF0) ==
                                        IDM_ABOUTBOX);
7.    ASSERT(IDM_ABOUTBOX < 0xF000);
8.    CMenu* pSysMenu = GetSystemMenu(FALSE);
9.    if (pSysMenu != NULL)
10.     {
11.       CString strAboutMenu;
12.       strAboutMenu.LoadString(IDS_ABOUTBOX);
13.       if (!strAboutMenu.IsEmpty())
14.         {
15.           pSysMenu->AppendMenu(MF_SEPARATOR);
16.           pSysMenu->AppendMenu(MF_STRING,
                          IDM_ABOUTBOX, strAboutMenu);
17.         }
18.     }
```

```
19.     // Set the icon for this dialog.  The framework does this
          automatically
20.     //  when the application's main window is not a dialog
21.     SetIcon(m_hIcon, TRUE);                    // Set big icon
22.     SetIcon(m_hIcon, FALSE);          // Set small icon
23.     // TODO: Add extra initialization here
24.     PopulateCombo();
25.     return TRUE;  // return TRUE  unless you set the focus to
                          a control
26.   }

27.   void CListsDlg::PopulateCombo(void)
28.   {
29.     TCHAR szBuffer[MAX_PATH];
30.     GetWindowsDirectory(szBuffer,MAX_PATH);
31.     m_cbMainDir.AddString (szBuffer);

32.     szBuffer[2] = 0;
33.     m_cbMainDir.AddString(szBuffer);

34.     GetSystemDirectory(szBuffer,MAX_PATH);
35.     m_cbMainDir.AddString (szBuffer);

36.     GetCurrentDirectory(MAX_PATH,szBuffer);
37.     m_cbMainDir.AddString(szBuffer);
38.   }
```

وعن هذا الكود، نوضح ما يلى:

- الـــــــــدوال GetWindowsDirectory() وGetSystemDirectory وGetCurrentDirectory الموجودة بالسطور ٣٠، ٣٤ ، ٣٦ هى دوال عامة تقوم بملء المصفوفة szBuffer باســـم المســـار المناســـب.كمـــا يقـــوم المـــتغير MAX_PATH بتعيين أكبر طول للمسار والذى غالبا ما يكون ٢٦٠.
- بعد الحصول على المسار داخل مصفوفة szBuffer، يتم تمرير المصفوفة إلى الدالة AddString() الخاصة بالمتغير m_cbMainDir والذى يقوم بإضافة المســـار إلى

مربع السرد والتحرير.ولأن قيمة الخاصية Sort هى True، يتم ترتيب العناصر المضافة لمربع السرد والتحرير ترتيباً أبجدياً.

يجب أن تقوم باختبار القيم المرجعة من الدوال AddString() وInsertString() والتى تقوم بإرجاع الرسائل CB_ERR أو CB_ERRSPACE عند حدوث خطأ.

الاستجابة لأحداث مربع السرد والتحرير

الغرض من استخدم مربع السرد والتحرير هو السماح للمستخدم باختيار عنصر واحد من العناصر الموجودة فيه، وحينئذٍ يجب أن يتعرف البرنامج على هذا الاختيار باستقبال الحدث CBN_SELCHANGE من مربع السرد والتحرير إلى المربع الحوارى كلما تم تغيير العنصر المختار.

قم بإضافة دالة الحدث CBN_SELCHANGE لمربع السرد والتحرير IDC_MAIN_DIR من خلال مربع الخصائص باسم OnCbnSelChangeMainDir() ثم قم بإضافة متغير عضو من النوع CString باسم m_strMainDir وقم بإضافة الكود التالى للدالة:

```
1.    void CListsDlg::OnCbnSelchangeMainDir()
2.    {
3.      // TODO: Add your control notification handler code here
4.      int nIndex = m_cbMainDir.GetCurSel();
5.      if(nIndex != CB_ERR)
6.         {
7.            m_cbMainDir.GetLBText(nIndex,m_strMainDir);
8.            PopulateTree();
9.         }
10.     PopulateListBox();
11.   }
```

وعن هذا الكود نوضح ما يلى:

- فى السطر رقم ٤ تم استدعاء الدالة GetCurSel() التى تقوم بإرجاع ترتيب العنصر المختار إلى المتغير nIndex.

- فى السطر رقم ٥، إذا لم يتم اختيار أى من عناصر مربع السرد والتحرير، يتم إرجاع القيمة **CB_ERR** وهى قيمة خاصة متعارف عليها.
- فى السطر رقم ٧، إذا كانت قيمة **nIndex** صحيحة، يتم تمريرها إلى الدالة **()GetLBText** بحيث يتم تخزين النص المختار فى المتغير **m_strMainDir**.
- فى السطر رقم ٨ تم استدعاء الدالة **PopulateTree** التى ستقوم بملء شجرة البيانات والتى سنقوم بإنشائها فى البند التالى.
- فى السطر رقم ١٠تم استدعاء الدالة **()PopulateListBox** التى ستقوم بملء مربع السرد وسنتعرض لها أيضاً بعد قليل.

ملء شجرة البيانات

عند إضافة العناصر إلى شجرة البيانات، يجب تخزين معلومات عن تركيب الشجرة حتى يتم وضع كل عنصر فى المكان المناسب. يحتوى التصنيف **CTreeCtrl** على وظائف شجرة البيانات كما يحتوى على مجموعة من الدوال التى تختص بإضافة وحذف العناصر والموضحة بجدول ٧-٥ التالى.

جدول ٧-٥ الدوال المستخدمة من قِبَل شجرة البيانات

الدالة	*الاستخدام*
InsertItem()	إضافة عنصر إلى شجرة البيانات
DeleteItem()	حذف عنصر من شجرة البيانات
DeleteAllItems()	حذف جميع عناصر شجرة البيانات

فى المثال الذى بين أيدينا، تقوم شجرة البيانات بعرض ملفات المجلد المختار بمربع السرد والتحرير فى ترتيب أبجدى. قم أولاً بإنشاء دالة عضو جديدة باسم **()PopulateTree** من النوع **void** والتى تقوم بإنشاء عنصر أبوى لكل حرف من الحروف الأبجدية اللاتينية وعددها ٢٦ عنصر ثم تقوم بإنشاء العنصر رقم ٢٧ والذى سيحتوى على الملفات التى لا تبدأ بحرف من الحروف الأبجدية. وفى سبيل ذلك تستخدم الدالة بعض دوال **Windows**

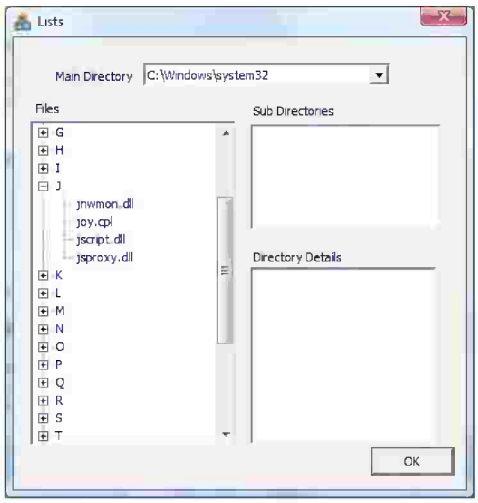
اللازمة لمعرفة محتويات المجلد ووضع كل عنصر فى مكانه المناسب.إذا كنت قد انتهيت الآن من إنشاء الدالة PopulateTree()، قم بإدخال الكود التالى إلى الدالة:

```
1.    void CListsDlg::PopulateTree(void)
2.    {
3.      m_treeFiles.DeleteAllItems();
4.      HTREEITEM hLetter[27];
5.      for(int nChar = 'A' ; nChar <= 'Z' ; nChar++)
6.      hLetter[nChar - 'A'] = m_treeFiles.InsertItem
                                          ((TCHAR*)&nChar);
7.      hLetter[26] = m_treeFiles.InsertItem("Other");
8.      HANDLE hFind;
9.      WIN32_FIND_DATA dataFind;
10.     BOOL bMoreFiles = TRUE;
11.     CString strFile;
12.     hFind = FindFirstFile(m_strMainDir + "\\*.*" , &dataFind);
13.     while(hFind != INVALID_HANDLE_VALUE && bMoreFiles
                                                == TRUE)
14.     {
15.       if(dataFind.dwFileAttributes ==
                                    FILE_ATTRIBUTE_ARCHIVE)
16.         {
17.             int nChar = dataFind.cFileName[0];
18.             if(islower(nChar))
19.                 nChar -= 32;
20.             if(isalpha(nChar))
21.                 nChar -= 'A';
22.             else
23.                 nChar = 26;
24.             m_treeFiles.InsertItem(dataFind.cFileName,
25.                                          hLetter[nChar]);
26.         }
27.       bMoreFiles = FindNextFile(hFind,&dataFind);
28.     }
29.     FindClose(hFind);
30.   }
```

وعن هذا الكود نوضح ما يلى:

- فى السطر رقم ٣ تم استدعاء الدالة DeleteAllItems() لحذف جميع عناصر الشجرة وذلك لأننا نقوم بإعادة ملء الشجرة كلما تم اختيار مجلد جديد من مربع السرد والتحرير.
- فى السطر رقم ٤ تم تعريف مصفوفة hLetter[ ] والتى ستحتوى على الحروف الأبجدية وهى من النوع HTREEITEM .
- فى السطور ٦ و٧ يتم استدعاء الدالة InsertItem() داخل الدوارة For لإضافة الحروف الأبجدية وكلمة Other كعناصر أساسية فى الشجرة وذلك لأننا لم ندخل معاملات أخرى تحدد مكان هذه العناصر كما سنرى بعد قليل.
- فى السطر رقم ١٢يتم استدعاء الدالة FindFirstFile() وذلك بتمرير معامل يحتوى على جميع الملفات الموجودة بالمجلد.
- فى السطر رقم ١٥يتم اختبار صفات الملف للتأكد من أنه ملف وليس مجلد.
- فى السطر رقم ١٧يتم تخزين الحرف الأول من اسم الملف وتحويله فى السطر ١٩ إلى حرف كبير فى حالة الضرورة.
- فى السطر ٢٠ يتم اختبار الحرف الأول من اسم الملف. فإذا كان الملف يبدأ بحرف أبجدى، قام البرنامج بوضعه فى المكان المناسب داخل المصفوفة hLetter[ ] وإلا قام بوضعه فى آخر المصفوفة.
- فى السطر ٢٤، يتم وضع الملف فى المكان الصحيح داخل الشجرة باستخدام دالة InsertItem().

والآن قم ببناء البرنامج وتنفيذه.اختر مجلد من مربع السرد والتحرير Main Directory، تلاحظ ظهور شجرة بالملفات الموجودة فى هذا المجلد داخل شجرة البيانات Files (انظر شكل ٧-٦).



شكل ٧-٦ عرض الملفات داخل شجرة البيانات

ملء مربع السرد

تشبه طريقة ملء مربع السرد إلى حد كبير الطريقة المستخدمة فى ملء مربع السرد والتحرير وذلك لأن عناصر مربع السرد والتحرير هى بالفعل عناصر مربع السرد، إلا أن الاختلاف يكمن فى طريقة العرض وإمكانية اختيار أكثر من عنصر فى حالة مربع السرد.

يحتوى التصنيف **CListBox** على وظائف مربع السرد والتى تتمثل فى مجموعة من الدوال التى تختص بإضافة وحذف العناصر وهى مشابهة تماماً لتلك الموجودة بجدول ٧-٤ السابق.

فى المثال الذى بين أيدينا، نريد لمربع السرد أن يحتوى على المجلدات الفرعية الموجودة تحت المجلد الرئيسى بمربع السرد والتحرير. قم بإنشاء دالة جديدة باسم **PopulateListBox()** من النوع **void** ثم قم بإضافة الكود التالى إلى الدالة.

```
1.    void CListsDlg::PopulateListBox(void)
2.    {
3.      m_lbSubDir.ResetContent();
4.      HANDLE hFind;
5.      WIN32_FIND_DATA dataFind;
6.      BOOL bMoreFiles = TRUE;
7.      hFind = FindFirstFile(m_strMainDir + "\\*.*" , &dataFind);
8.      while(hFind != INVALID_HANDLE_VALUE && bMoreFiles
```

```
                                                    == TRUE)
9.      {
10.       if(dataFind.dwFileAttributes ==
                                        FILE_ATTRIBUTE_DIRECTORY)
11.       {
12.           if(strcmp(dataFind.cFileName,"."))
13.           if(strcmp(dataFind.cFileName,".."))
14.             m_lbSubDir.AddString(dataFind.cFileName);
15.       }

16.       bMoreFiles = FindNextFile(hFind,&dataFind);
17.     }
18.   FindClose(hFind);
19. }
```

وعن هذا الكود نوضح ما يلى:

- فى السطر رقم ٣ تم استدعاء الدالة ResetContent() لإزالة محتويات مربع السرد وذلك لأننا نقوم بملء مربع السرد كلما تم تغيير اسم المجلد من مربع السرد والتحرير.
- فى السطر رقم ١٠، يتم اختبار العنصر للتأكد من أنه مجلد فرعى وليس ملف.
- فى السطور ١٢ و١٣ يتم تجاهل المجلدات الفرعية "." و ".." .
- فى السطر رقم ١٤ يتم إضافة اسم المجلد الفرعى إلى مربع السرد باستخدام الدالة AddString().

يجب أن تقوم باختبار القيم المرجعة من الدوال AddString() وInsertString() والتى تقوم بإرجاع الرسائل CB_ERR أو CB_ERRSPACE عند حدوث خطأ.

الاستجابة لأحداث مربع السرد

عندما يقوم المستخدم بأداء حدث من الأحداث داخل مربع السرد، كأن يقوم باختيار عنصر أو أكثر من العناصر الموجودة بالمربع، يقوم Windows بإرسال رسالة إلى المربع الحوارى تسمى LBN_SELCHANGE.

فى المثال الذى بين أيدينا، يتيح لك مربع السرد اختيار أكثر من عنصر، وعلى هذا يتم تخزين العناصر المختارة داخل متغير جديد وليكن m_strList وهو من النوع CStringList. قم بإضافة المتغير الجديد إلى تصنيف المربع الحوارى باستخدام المعالج Add Member Variable Wizard ثم قم بإنشاء دالة الحدث LBN_SELCHANGE لمربع السرد IDC_SUB_DIR من خلال مربع الخصائص واترك اسمها الافتراضى OnLbnSelchangeSubDir() كما هو. تقوم هذه الدالة بالمهام الآتية:

١. تحدد عدد العناصر المختارة.

٢. تتعرف على نص كل عنصر ثم تقوم بإضافته إلى المتغير m_strList.

٣. تقوم باستدعاء دالة جديدة باسم PopulateListControl() التى تقود المتغير m_strList لإضافة التفاصيل لأداة القائمة كما سنرى بعد قليل.

والآن قم بإضافة الكود التالى إلى الدالة OnLbnSelchangeSubDirs() كما يلى:

```
1.  void CListsDlg::OnLbnSelchangeSubDir()
2.  {
3.     // TODO: Add your control notification handler code here
4.     int nSelCount = m_lbSubDir.GetSelCount();
5.     m_strList.RemoveAll();
6.     if(nSelCount)
7.     {
8.        CString str;
9.        LPINT pItems = new int[nSelCount];
10.       m_lbSubDir.GetSelItems(nSelCount,pItems);
11.       for(int i= 0 ; i < nSelCount; i++)
12.       {
13.          m_lbSubDir.GetText(pItems[i],str);
14.          m_strList.AddTail(str);
15.       }
16.       delete [] pItems;
17.    }
18.    PopulateListControl();
19. }
```

وعن هذا الكود، نوضح ما يلى:

- فى السطر رقم ٤ يتم استدعاء الدالة GetSelCount() التى تقوم بإرجاع عدد العناصر المختارة وتستخدم فى حالة الاختيار المتعدد فقط.
- فى السطر رقم ٥ يتم استدعاء الدالة RemoveAll() لإزالة جميع العناصر الموجودة بالمتغير m_strList وذلك لإعادة بناء العناصر المختارة من جديد.
- فى السطر رقم ٩ يتم حجز مصفوفة من الأرقام الصحيحة لتحتوى على أرقام العناصر المختارة بحيث يكون حجم المصفوفة هو عدد العناصر المختارة nSelCount.
- فى السطر رقم ١٠تقوم الدالة GetSelItems() بإرجاع قائمة بالعناصر المختارة وتخزينها داخل المصفوفة.
- فى السطر رقم ١١يتم استخدام الدوارة For بعدد العناصر المختارة وفى كل مرة يتم استدعاء الدالة GetText() لتخزين النص المختار فى المتغير str.
- فى السطر رقم ١٤يتم إضافة محتويات المتغير str إلى نهاية المتغير m_strList الذى سيحتوى على جميع العناصر المختارة.
- فى السطر رقم ١٦ يتم حذف المصفوفة pItems لأننا لم نعد فى حاجة إليها.
- فى السطر رقم ١٨يتم استدعاء الدالة PopulateListControl() المستخدمة فى ملء أداة القائمة والتى سنتعرف عليها بعد قليل.

ملء أداة القائمة

تختلف طريقة ملء أداة القائمة نوعاً ما عن تلك المستخدمة مع باقى أنواع السرد التى ذكرناها فى هذا الفصل.كما أنها تختلف باختلاف نوع أداة القائمة نفسها، حيث يوجد كما ذكرنا أربعة أنواع من أداة القائمة وهى الرمز Icon والرمز الصغير Small Icon والقائمة List والتقرير Report والتى قمنا بشرحها قبل ذلك. ويعد التقرير أكثر هذه الأنواع استخداماً وفيه يتم إظهار البيانات فى أعمدة رأسية. فمثلاً طريقة عرض التقرير فى

مستكشف Windows تقوم بعرض أعمدة للاسم Name والحجم Size والنوع Type وآخر تاريخ للتعديل Modified وعلى هذا فكل صف من الصفوف يعبر عن بيانات ملف معين.

يحتوى التصنيف CListCtrl على وظائف أداة القائمة ممثلةً فى مجموعة من الدوال التى تختص بإضافة وحذف العناصر والموضحة بجدول ٧-٦ التالى.

جدول ٧-٦ الدوال المستخدمة من قِبَل أداة القائمة

الدالة	الاستخدام
InserColumn()	إضافة عمود جديد فى مكان معين داخل القائمة
DeleteColumn()	حذف عمود من القائمة
InsertItem()	إضافة عنصر جديد للقائمة
DeleteItem()	حذف عنصر من القائمة
DeleteAllItems()	حذف جميع عناصر القائمة
SetItemText()	إضافة نص لعنصر فرعى فى القائمة

فى المثال الذى بين أيدينا، نستخدم أداة سرد من نوع التقرير تحتوى على ثلاثة أعمدة، الأول اسم المجلد Directory والثانى عدد الملفات الموجودة بالمجلد Files والأخير حجم هذه الملفات Size KB. الخطوة الأولى هى إنشاء الأعمدة الثلاثة لأنها لن تتغير أثناء سير البرنامج وذلك باستخدام الدالة OnInitDialog(). قم بإضافة الكود التالى للدالة فى السطور من ٧ إلى ٩.

```
1.    BOOL CListDlg::OnInitDialog()
2.    {
3.    ……
4.
5.      // TODO: Add extra initialization here
6.      PopulateCombo();
7.      m_lbDirDetails.InsertColumn(0,"Directory",
                                          LVCFMT_LEFT,70);
8.      m_lbDirDetails.InsertColumn(1,"Files",LVCFMT_RIGHT,
```

```
                                                    50);
9.      m_lbDirDetails.InsertColumn(2,"Size
                                          KB",LVCFMT_RIGHT, 60);
10.   return TRUE;  // return TRUE  unless you set the focus to
                       a control
11.   }
```

لاحظ إضافة الأعمدة الثلاثة فى السطور من ٧ إلى ٩ باستخدام الدالة InsertColumn() والتى تحتوى على أربعة معاملات:الأول ترتيب العمود والثانى عنوان العمود والثالث محاذاة النص والرابع عرض العمود الذى يمكنك تغييره بالفأرة أثناء تنفيذ البرنامج.

الخطوة التالية هى إدخال عناصر أداة القائمة وذلك من خلال الدالة PopulateListControl() .قم بإنشاء الدالة باستخدام المعالج ثم قم بإدخال الكود التالى لهذه الدالة:

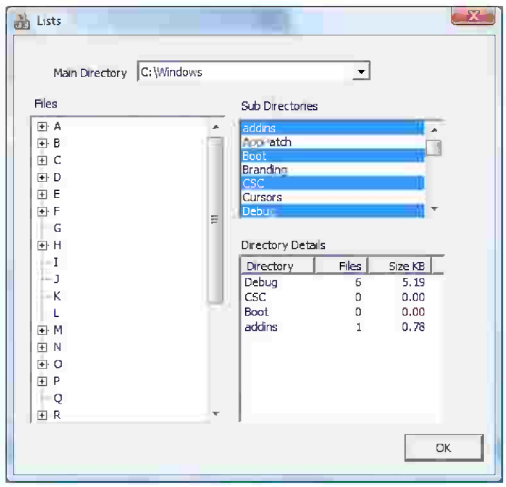
```
1.    void CListsDlg::PopulateListControl(void)
2.    {
3.      m_lbDirDetails.DeleteAllItems();
4.      POSITION pos;
5.      for(pos = m_strList.GetHeadPosition();pos != NULL;)
6.      {
7.        int nItem;
8.        HANDLE hFind;
9.        WIN32_FIND_DATA dataFind;
10.       BOOL bMoreFiles = TRUE;
11.       CString str;
12.       CString strFind;
13.       str = m_strList.GetAt(pos);
14.       nItem = m_lbDirDetails.InsertItem(0,str);
15.       strFind = m_strMainDir + "\\" + str + "\\*.*";
16.       hFind = FindFirstFile(strFind,&dataFind);
17.       int nFileCount = 0;
18.       double nFileSize = 0;
19.       while(hFind != INVALID_HANDLE_VALUE &&
                                          bMoreFiles == TRUE)
20.       {
21.          if(dataFind.dwFileAttributes ==
```

```
                                                    FILE_ATTRIBUTE_ARCHIVE)
22.          {
23.             nFileCount++;
24.             nFileSize += (dataFind.nFileSizeHigh *
                            MAXDWORD) + dataFind.nFileSizeLow;
25.          }
26.       bMoreFiles = FindNextFile(hFind,&dataFind);
27.       }
28.     FindClose(hFind);
29.     str.Format("%1d",nFileCount);
30.     m_lbDirDetails.SetItemText(nItem,1,str);
31.     str.Format("%-1.2f",nFileSize /1024.0);
32.     m_lbDirDetails.SetItemText(nItem,2,str);
33.     m_strList.GetNext(pos);
34.   }
35. }
```

وعن هذا الكود نوضح ما يلى:

- فى السطر رقم ٣ يتم حذف جميع العناصر الموجودة بأداة القائمة عدا الأعمدة الرئيسية باستخدام الدالة DeleteAllItems().
- فى السطر رقم ٥ يتم استخدام الدوارة for بعدد المجلدات المختارة بمربع السرد والمخزنة داخل المتغير m_strList.
- فى السطر رقم ١٤ يتم إضافة اسم المجلد بالعمود الأول Directory فى أداة القائمة باستخدام الدالة InsertItem().
- فى السطور ٢٣ و٢٤ يتم زيادة عدد الملفات الموجودة بالمجلد Directory والحجم الكلى لهذه الملفات باستخدام الدالة SetItemText().
- فى السطر رقم ٣٠ يتم إضافة عدد ملفات المجلد فى العمود الثانى Files باستخدام الدالة SetItemText().
- فى السطر رقم ٣٢ يتم إضافة حجم ملفات المجلد بالعمود الثالث Size KB باستخدام الدالة SetItemText().

وفى النهاية، قم ببناء البرنامج وتنفيذه ثم قم باختبار الاختيار المتعدد من مربـــع الســـرد، لتحصل على نتائج مشابهة لتلك الموجودة بشكل ٧-٧ التالى.



شكل ٧-٧ اختيار أكثر من مجلد وعرض بياناتها فى القائمة