الفصل السادس

العمل مع أدوات النصوص

Text Controls

تعتبر أدوات أو مربعات إظهـــار النصـــوص مـــن العناصـــر المستخدمة بكثرة داخل المربعات الحوارية، حيث نحتاج دائماً لكتابة العناوين الثابتة وذلك باستخدام أداة العنوان أو أداة النص الثابـــت كما نحتاج أيضاً لكتابة النصوص القابلة للتحرير باستخدام مربـــع النص أو مربع التحرير.

بانتهاء هذا الفصل ستتعرف على:

- ***استخدام أداة العنوان.***
- ***استخدام أدوات مربع النص.***
- ***إنشاء تصنيف جديد.***
- ***استخدام مربعات النصوص متعددة السطور.***

obeikandl.com

استخدام أداة العنوان Text Control

تعتبر أداة العنوان (Static Text control) من أدوات النصوص المستخدمة داخل المربعات الحوارية لعرض عناوين أدوات التحكم الأخرى أو وصف خيار من الخيارات الموجودة بالمربع الحوارى أو إظهار جملة أو فقرة كاملة مثل رسالة حقوق الطبع والتأليف ورقم الإصدار والبيانات الأخرى المصاحبة للبرنامج. يمكنك كتابة حتى ٢٥٥ حرف داخل أداة العنوان كما يمكنك التحكم فى شكل ونمط عرض النص داخل الأداة ليكون فى وسط السطر أو فى أحد الأجناب أو يكون بارزاً أو محفوراً كما يمكنك إضافة إطار حول النص.

الجمع بين أداة العنوان ومربع النص

تستخدم أداة العنوان بكثرة فى المربعات الحوارية لوصف مربعات النصوص. فمربعات النصوص عبارة عن أدوات تحكم تتيح للمستخدم إدخال البيانات إلى البرنامج، ولأن هذه الأدوات لا تحتوى على الخاصية **Caption** المستخدم لإعطاء عنوان الأداة، فدائماً ما يتم استخدام أداة عنوان بجوار مربع النص كعنوان لهذا المربع ويكون عنوان المربع ثابتاً وبيانات مربع النص نفسه هى التى تتغير. فمثلاً عبارة "اسم العميل" التى تظهر قبل المربع الذى تكتب فيه اسم العميل أو الذى يظهر فيه اسم العميل تكون ثابته وهى التى نقول عنها أداة العنوان أو **Static text control** أما اسم العميل الذى يتغير من سجل لآخر والذى تكتبه بنفسك أو يظهر لك مع بيانات العميل فنقول عنه مربع النص أو **Edit box control** كما ستعرف بعد قليل. كما أن هناك هدف آخر من استخدام العنوان مع مربع النص، وهو سهولة الوصول إلى مربع النص باستخدام لوحة المفاتيح. فمثلاً إذا احتوت أداة العنوان على كلمة **Football**، فهذا يعنى الوصول لمربع النص المجاور للأداة بضغط الاختصار **Ctrl+B** من لوحة المفاتيح.

تعديل أداة العنوان أثناء عمل البرنامج

نحتاج أحيانا إلى تغيير محتويات أداة العنوان أثناء تنفيذ البرنامج. ففى هذا الفصل سنقوم بإنشاء مشروع جديد يسمى **SysInfo** يظهر اسم الكمبيوتر واسم المستخدم وحجم الذاكرة الحرة المتاحة بالجهاز. ولأن حجم الذاكرة الحرة يتغير من حينٍ إلى آخر تبعاً للبرامج المحملة بالذاكرة، فستحتاج إلى تغيير محتوى أداة العنوان باستمرار. لبيان كيفية تعديل أداة العنوان أثناء تنفيذ البرنامج، قم بإنشاء مشروع حوارى جديد باسم **SysInfo** كما تعلمت من قبل ثم تابع معنا الخطوات الآتية:

١. قم بفتح المربع الحوارى **IDD_SYSINFO_DIALOG** داخل محرر الموارد ثم احذف مربع النص الموجود بالمربع الحوارى.
٢. اختر رمز أداة العنوان Aa Static Text من مربع الأدوات ثم قم بإضافة أربعةً منها بالجانب الأيسر من المربع الحوارى (انظر شكل ٦-١).
٣. قم بتعديل عنوان كل أداة من الأدوات كما هو موضح بالشكل.
٤. اختر رمز أداة العنوان من مربع الأدوات مرةً أخرى ثم قم بإضافة أربعة أدوات منها بجوار الأدوات السابقة وقم بتسمية الأربعة أدوات الجديدة كما بالجدول التالى (قمنا بتسمية هذه الأدوات دون الأدوات السابقة لأننا سنستخدمها داخل الكود):

	الاسم
الأولى	***IDC_COMPUTER_NAME***
الثانية	**IDC_TOTAL_MEMORY**
الثالثة	**IDC_FREE_MEMORY**
الرابعة	**IDC_MEMORY_LOAD**

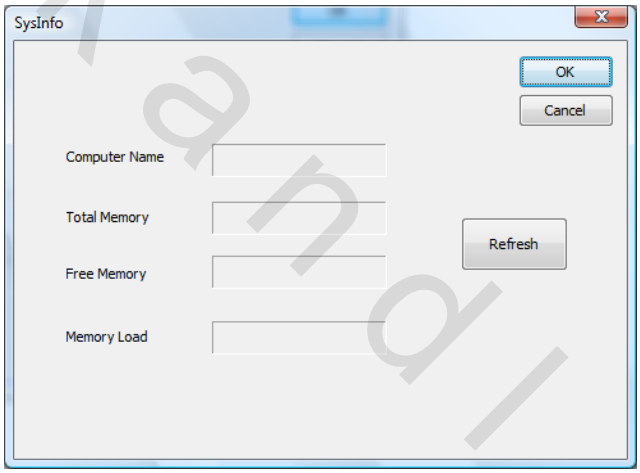
٥. اضغط مفتاح **Ctrl** من لوحة المفاتيح واختر الأربعة أدوات التى قمت بإضافتها منذ قليل ثم احذف محتويات الخاصية **Caption**.

obeikandi.com

٦. قم أيضاً بتخصيص القيمة **True** للخاصية **Sunken** حتى تظهر الأدوات محفورة داخل المربع الحوارى.

٧. اختر رمز زر الأمر من مربع الأدوات ثم قم بإضافة زر جديد بالركن الأيمن السفلى من المربع الحوارى.

٨. قم بإدخال اسم الزر إلى الخاصية **ID** وليكن **IDC_REFRESH** وعنوان الزر إلى الخاصية **Caption** وليكن **Refresh**. يجب أن يظهر المربع الحوارى الآن كما فى شكل ٦-١.



شكل ٦-١ إضافة أدوات العنوان للمربع الحوارى

وبذلك ستقوم الأربعة عناوين المحفورة بعرض معلومات النظام كلما تم نقر الزر **Refresh**.

الخطوة التالية هى تعريف متغير لكل أداة من الأدوات الأربعة، تابع معنا الخطوات الآتية:

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

٢. انقر التصنيف **CSysInfoDlg** 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 العنوان الأولى **IDC_COMPUTER_NAME** من مربع السرد **Control ID**.

٥. اختر **Value** من مربع السرد **Category** لأننا نريد قيمة أداة التحكم فقط ولا نحتاج إلى التعامل مع الخصائص الأخرى المصاحبة لهذه الأداة.

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

٧. اكتب اسم المتغير فى خانة **Variable name** ولتكن **m_strComputerName** واختر درجة المتغير من مربع السرد **Access** وهو **Public** فى هذه الحالة.

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

٩. قم بتكرار الخطوات السابقة لإنشاء المتغير **m_strTotalMemory** المرتبط بأداة العنوان **IDC_TOTAL_MEMORY** والمتغير **m_strFreeMemory** المرتبط بأداة العنوان **IDC_FREE_MEMORY** والمتغير **m_strMemoryLoad** المرتبط بأداة العنوان **IDC_MEMORY_LOAD**.

الخطوة التالية هى إنشاء دالة نقر الزر **IDC_REFRESH** . انقر الزر نقراً مزدوجاً من داخل نافذة محرر الموارد ثم قم بإدخال الكود التالى:

```
1.    void CSysInfoDlg::OnBnClickedRefresh()
2.    {
3.      TCHAR szBuffer[256];
4.      DWORD dwSize = 256;
5.      GetComputerName(szBuffer,&dwSize);
6.      m_strComputerName = szBuffer;
7.      MEMORYSTATUS mem_stat;
8.      GlobalMemoryStatus(&mem_stat);
9.      m_strTotalMemory.Format("%1d
                              KB",mem_stat.dwTotalPhys/1024);
```


obeikandl.com

```
10.     m_strFreeMemory.Format("%1d
                        KB",mem_stat.dwAvailPhys/1024);
11.   m_strMemoryLoad.Format("%d%%",
                              mem_stat.dwMemoryLoad);
12.     UpdateData(FALSE);
13.   }
```

وعن هذا الكود نوضح ما يلى:

- فى السطور ٣و٤ تم تعريف متغيرين لاستخدامهما مع الدالة **GetComputerName** فى السطر رقم ٥.
- فى السطر رقم ٦، تم نقل محتويات المتغير **szBuffer** إلى المتغير **m_strComputerName** الخاص بأداة العنوان **IDC_COMPUTER_NAME**.
- فى السطر رقم ٧، تم تعريف المتغير **mem_stat** وهو متغير من نوع التركيب **MEMORYSTATUS** الذى يحتوى على الكود التالى:

```
typedef struct _MEMORYSTATUS {
   DWORD dwLength;
   DWORD dwMemoryLoad;
   DWORD dwTotalPhys;
   DWORD dwAvailPhys;
   DWORD dwTotalPageFile;
   DWORD dwAvailPageFile;
   DWORD dwTotalVirtual;
   DWORD dwAvailVirtual;
} MEMORYSTATUS;
```

- فى السطر رقم ٨، تم تمرير عنوان المتغير **mem_stat** للدالة **GlobalMemoryStatus()** التى تقوم بملأ متغيرات التركيب **MEMORYSTATUS** ببيانات الذاكرة الحالية.
- فى السطور من ٩ إلى ١١ تم نقل البيانات من أعضاء المتغير **mem_stat** إلى المتغيرات الخاصة بأدوات العنوان التى ستحتوى على بيانات الذاكرة.

- فى السطر رقم ١٢، استخدمنا دالة **UpdateData()** لتحديث البيانات التى تقوم أدوات العنوان الأربعة بعرضها.

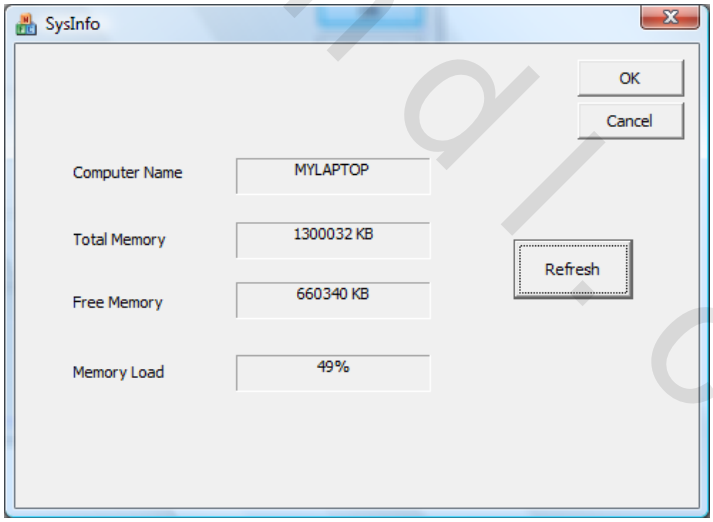
بالإضافة لاستخدام الدالة GetComputerName()، يمكنك استخدام الدالة GetUserName لمعرفة المستخدم الذى يقوم بتنفيذ التطبيق والدالة GetVersionEx() لمعرفة نظام التشغيل المستخدم لتشغيل التطبيق.

لاختبار المربع الحوارى بعد التعديلات الأخيرة، تابع معنا الخطوات الآتية:

١. قم ببناء المشروع وتنفيذه.

٢. انقر زر **Refresh**، تلاحظ ظهور بيانات داخل أدوات العنوان الموجودة بالجانب الأيمن من المربع الحوارى (انظر شكل ٦-٢).

٣. قم بفتح وإغلاق بعض البرامج، وفى كل مرة انقر زر **Refresh**، تلاحظ تغير البيانات التى تظهر داخل أدوات العنوان كلما قمت بفتح أو إغلاق برنامج آخر.



شكل ٦-٢ عرض بيانات الجهاز عند نقر زر **Refresh**

استخدام مربع النص Edit box Control

يعتبر مربع النص (Edit box Control) أحـــد أدوات النصـــوص المســـتخدمة لتمكين المستخدم من إدخال البيانات وعرضها مرة أخرى.حيث تختلف عن أداة العنـــوان فى قابليتها للتعديل مباشرةً أثناء تنفيذ البرنامج بالنسخ والقص واللصق وباقى العمليـــات التى تتم على النصوص باستخدام الفأرة ولوحة المفاتيح.كما تحتوى مربعات النصوص على أنماط متعددة يمكنك الاختيار من بينها لتتناسب مع المهام التى يؤديها كل مربع على حدة ، وهذا ما سنتعرف عليه فى الجزء المتبقى من هذا الفصل.

إضافة مربع النص إلى المربع الحوارى

للتعرف على كيفية إضافة مربعات النصوص إلى المربعات الحوارية، قم بإنشاء مشروع حوارى جديد باسم **Edits** ثم تابع معنا الخطوات الآتية:

١. قم بفتح المربع الحوارى **IDD_EDITS_DIALOG** داخل محرر الموارد، ثم احذف مربع النص الموجود بالمربع الحوارى.
٢. اختر رمز أداة العنوان من مربع الأدوات ثم قم بإضافة أداة عنوان بالجزء الأيســـر العلوى من المربع الحوارى (انظر شكل ٦-٣).
٣. تأكد من اختيار أداة العنوان التى قمت بإضافتها ثم انقر الأداة بزر الفأرة الأيمـــن واختر **Properties** من القائمة الموضعية، تظهر نافذة الخصائص.
٤. أدخل عنواناً للأداة داخل الخاصية **Caption** وليكن **&First**.
٥. اختر رمز مربع النص ab| Edit Control من مربع الأدوات ثم قم بإضافة مربـــع نص يمين أداة العنوان.
٦. قم بإدخال اسم مربع النص إلى الخاصية **ID** وليكن **IDC_EDIT_FIRST**.
٧. اختر رمز الزر من مربع الأدوات ثم قم بإضافة زر جديد يمين مربع النص الـــذى قمت بإضافته منذ قليل (انظر شكل ٦-٣).

obeikandl.com

٨. أدخل عنوان الزر داخل الخاصية **Caption** وليكن **Get Text** واسم الزر داخل الخاصية **ID** وليكن **IDC_GET_TEXT**.

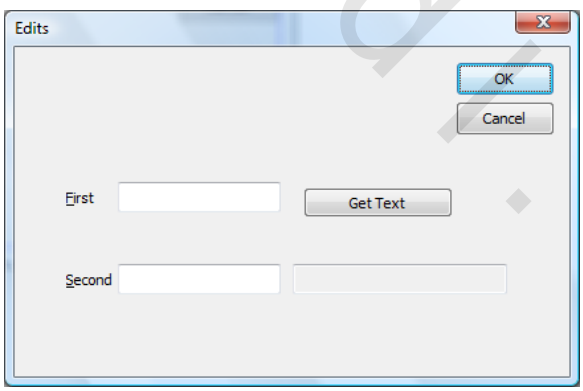
٩. اختر رمز أداة العنوان من مربع الأدوات ثم قم بإضافة أداة عنوان جديدة أسفل الأداة **First** وأعطها العنوان **&Second**.

١٠. اختر رمز مربع النص من مربع الأدوات ثم قم بإضافة مربع نص جديد يمين أداة العنوان **Second** وقم بإدخال اسم لمربع النص داخل الخاصية **ID** وليكن **IDC_EDIT_SECOND**.

١١. اختر رمز مربع النص من مربع الأدوات ثم قم بإضافة مربع نص يمين مربع النص الذى قمت بإضافته منذ قليل.

١٢. قم بإدخال اسم لمربع النص إلى الخاصية **ID** وليكن **IDC_EDIT_SHOW** ثم قم بتخصيص القيمة **True** للخاصية **Read-only**.

١٣. قم بزيادة طول مربع النص الأخير إلى اليمين قليلاً. يجب أن يظهر المربع الحوارى الآن كما فى شكل ٦-٣.



شكل ٦-٣ إضافة مربع النص للمربع الحوارى

لاحظ أننا استخدمنا الاختصارات عند تسمية العناوين الموجودة بجوار مربعات النصوص. فإذا قمت بضغط الاختصار **Alt+F** من لوحة المفاتيح، يتم وضع مؤشر الكتابة داخل مربع

النص المجاور للعنوان **First**. أما إذا قمت بضغط الاختصار **Alt+S**، فيتم وضع مؤشر الكتابة داخل مربع النص المجاور للعنوان **Second**.

عند استخدام الاختصارات مع مربعات النصوص، يجب أن يتبع مربع النص أداة العنوان المجاورة مباشرةً فى ترتيب الجدولة Tab Order.

العمل مع النص الموجود بمربع النص

عندما تقوم بإدخال نص داخل مربع النص ثم تقوم بنقر زر **Ok** الموجود بالمربع الحوارى، فغالبا ما تحتاج لهذه النصوص لتخزينها داخل حقول قواعد البيانات أو إظهارها فى مكان آخر بالبرنامج. لأداء ذلك، يجب أن تقوم أولاً بتعريف المتغيرات التى سيتم نقل هذه النصوص إليها، لذا تابع معنا الخطوات الآتية:

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

٢. انقر التصنيف **CEditsDlg** 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 النص **IDC_EDIT_FIRST** من مربع السرد **Control ID**.

٥. اختر **Value** من مربع السرد **Category** لأننا نريد قيمة أداة التحكم فقط ولا نحتاج إلى التعامل مع الخصائص الأخرى المصاحبة لهذه الأداة.

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

٧. اكتب اسم المتغير فى خانة **Variable name** وليكن **m_strFirst** واختر درجة المتغير من مربع السرد **Access** وهو **Public** فى هذه الحالة.

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

obeikandl.com

عند نقر الزر **Get Text**، نريد أن يقوم البرنامج بعكس النص الموجـــود بمربـــع **First** وكتابته مرةً أخرى داخل المربع. لأداء ذلك، انقر زر **IDC_GET_TEXT** نقراً مزدوجاً ثم قم بإدخال الكود التالى لدالة احتواء حدث نقر هذا الزر:

```
1.    void CEditsDlg::OnBnClickedGetText()
2.    {
3.       UpdateData();
4.       if (m_strFirst.IsEmpty() == FALSE)
5.          {
6.             AfxMessageBox(m_strFirst);
7.             m_strFirst.MakeReverse();
8.             UpdateData(FALSE);
9.          }
10.   }
```

وعن هذا الكود، نوضح ما يلى:

- فى السطر رقم ٣، تم استدعاء الدالة **()UpdateData** التى تقوم بنقل البيانـــات الموجودة بمربع النص **First** إلى المتغير **m_strFirst**.
- فى السطر رقم ٤، تقوم الدالة **()IsEmpty** بإرجاع القيمـــة **False** إذا لم يكـــن المتغير فارغاً.
- فى السطر رقم ٧، تقوم الدالة **()MakeReverse** بعكس الحـــروف الموجـــودة بالمتغير **m_strFirst**.
- فى السطر رقم ٩، تم استدعاء الدالة **()UpdateData** مرةً أخرى مع تمرير المعامل **FALSE** لنقل محتويات المتغير **m_strFirst** إلى مربـــع الـــنص **First** ولـــيس العكس كما فى سطر ٣.

قم ببنـــاء البرنـــامج وتنفيـــذه ثم قـــم بكتابـــة نـــص بســـيط داخـــل مربـــع الـــنص **IDC_EDIT_FIRST** وانقر زر **Get Text**، يظهر مربع رسالة يحتوى على النص الذى أدخلته. انقر زر **Ok** لإغلاق مربع الرسالة، تلاحظ عكس حروف النص داخل مربع النص (انظر شكل ٦-٤).

شكل ٦-٤ تظهر رسالة بالنص المدخل ثم عكس هذا النص داخل مربع النص

الاستجابة لأحداث مربع النص

عندما تقوم بتعديل النص الموجود داخل مربع النص، يقوم **Windows** بإرسال رسالة إلى البرنامج تحدد له نوعية هذا التعديل. يوضح جدول ٦-١التالى أهم الرسائل المصاحبة لأداة مربع النص.

جدول ٦-١رسائل مربع النص

الرسالة	*وقت إرسالها*
EN_CHANGE	**عند تغير النص بعد عرضه**
EN_UPDATE	**عند تغير النص قبل عرضه**

obeikandl.com

الرسالة	وقت إرسالها
EN_SETFOCUS	عندما تستقبل الأداة محث التركيز
EN_KILLFOCUS	عندما تفقد الأداة محث التركيز
EN_MAXTEXT	عندما يكون النص المدخل أكثر من المتاح
EN_HSCROLL	عند نقر قضيب الانزلاق الأفقى
EN_VSCROLL	عند نقر قضيب الانزلاق الرأسى
EN_ERRSPACE	عند فشل الأداة فى حجز مكان بالذاكرة

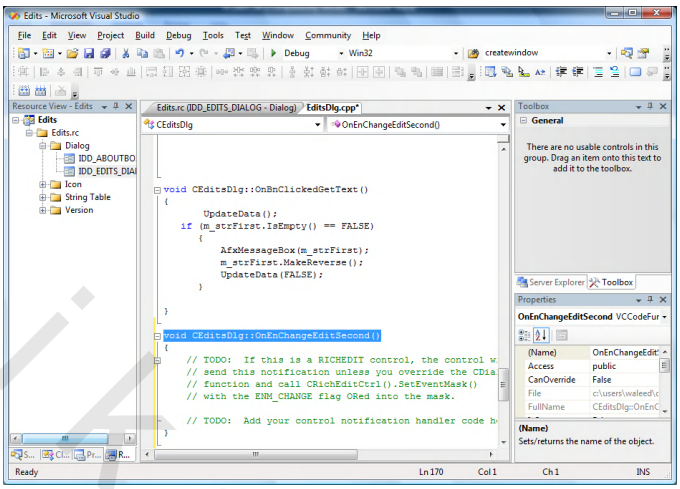
ولكى تستخدم رسالة من هذه الرسائل، يجب أن تقوم بتعريف دالة جديدة تحتوى على الكود الذى تريد تنفيذه عند حدوث التعديل. وفيما يلى سنقوم بإنشاء دالة جديدة تعبر عن الحدث EN_CHANGE الخاص بمربع النص Second حيث سيتم نقل النص الموجود به إلى مربع النص المجاور كلما حدث تعديل.

لإنشاء دالة احتواء رسالة (Message Handler) تابع معنا الخطوات الآتية:

١. قم بتنشيط مربع النص المجاور للعنوان Second من نافذة تحرير الموارد.

٢. انقر زر الأحداث ⚡ من شريط أدوات مربع الخصائص، تظهر قائمة بالأحداث (الرسائل) المصاحبة لمربع النص.

٣. انقر الحدث EN_CHANGE ثم اختر Add OnEnChangeEditSecond من مربع السرد المجاور، يتم إنشاء هيكل دالة الحدث (الرسالة) داخل نافذة الكود (انظر شكل ٦-٥).



شكل ٦–٥ إضافة دالة جديدة

٤. قم بإدخال الكود التالى إلى الدالة OnEnChangeEditSecond() التى سيتم استدعاؤها كلما قام المستخدم بتغيير النص الموجود بمربع النص IDC_EDIT_SECOND سواءً بالإضافة أو الحذف.

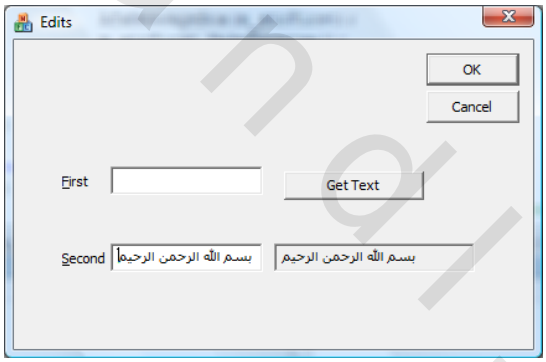
```
1.  void CEditsDlg::OnEnChangeEditSecond()
2.  {
3.    // TODO:  If this is a RICHEDIT control, the control will not
4.    // send this notification unless you override the
        CDialog::OnInitDialog()
5.    // function and call CRichEditCtrl().SetEventMask()
6.    // with the ENM_CHANGE flag ORed into the mask.

7.    // TODO:  Add your control notification handler code here
8.   CEdit* pEdit = (CEdit*)GetDlgItem(IDC_EDIT_SECOND);
9.   CEdit* pEditShow = (CEdit*)GetDlgItem(IDC_EDIT_SHOW);
10.  CString strText;
11.   pEdit -> GetWindowText(strText);
12.   pEditShow ->SetWindowText (strText);
13.  }
```

وعن هذا الكود، نوضح ما يلى:

- تحتوى السطور من ٣ إلى ٧ على نص المساعدة الذى يمدك به البرنامج كى تتمكن من كتابة كود الرسالة (الحدث).
- فى السطور ٨ و٩ قمنا باستدعاء الدالة **GetDlgItem()** لتخصيص مؤشر لكلٍ من مربعى النص التى سنتعامل معها.
- فى السطر رقم ١١ تم استدعاء الدالة **GetWindowText()** لنقل محتويات مربع النص **Second** إلى المتغير **strText**.
- فى السطر رقم ١٢ تم استدعاء الدالة **SetWindowText()** لنقل محتويات المتغير **strText** إلى مربع نص القراءة فقط المجاور لمربع النص **Second**.

قم ببناء البرنامج وتنفيذه، تلاحظ أنك كما قمت بالكتابة داخل مربع النص **Second** ، يظهر نفس النص داخل مربع النص المجاور (انظر شكل ٦-٦).



شكل ٦-٦ ظهور النص بمربع القراءة فقط بمجرد الكتابة فى المربع المجاور

لاحظ أنه إذا قمت بنقر مربع نص القراءة فقط، يظهر مؤشر الكتابة داخل مربع النص إلا أنك لن تستطيع الكتابة فيه. للتغلب على هذه المشكلة، قم بإنشاء دالة الحدث **EN_SETFOCUS** داخل مربع النص **IDC_EDIT_SHOW** كما سبق ثم قم بإضافة السطر التالى داخل الدالة **OnEnSetfocusEditShow()**:

```
GetDlgItem(IDC_EDIT_SECOND)->SetFocus();
```

قم ببناء البرنامج وتنفيذه مرةً أخرى ثم انقر مربع نص القراءة فقط، تلاحظ ظهور مؤشر الكتابة داخل مربع النص **Second** بدلاً من المربع الأصلى.

إنشاء تصنيف جديد

أحيانا نريد إدخال النص داخل مربع النص بتنسيق معين أو بنمط معين. لعمل ذلك، لابد من إنشاء تصنيف متفرع من تصنيف مربع النص الرئيسى. يستخدم هذا التصنيف الجديد الصفات والوظائف الموجودة بالتصنيف الرئيسى بالإضافة إلى وظائف أخرى إضافية. لتوضيح هذا المفهوم، سنقوم فيما يلى بإنشاء تصنيف فرعى باسم **CInitials** من التصنيف الأساسى **CEdit**. حيث يقوم هذا التصنيف بالتحكم فى بدايات أسماء الأشخاص بحيث تظهر بصورة معينة وذلك بالسماح بإدخال الحروف الأبجدية فقط وتحويل الحروف الصغيرة إلى حروف كبيرة تلقائياً وإضافة نقطة بعد كل بداية بحيث يتم حذف هذه النقطة إذا ضغط المستخدم مفتاح **Backspace**. سنقوم أولاً بإضافة بعض مربعات النصوص للمربع الحوارى **Edits** ثم نقوم بعد ذلك بإنشاء التصنيف الفرعى.

إضافة مربعات نصوص إلى المربع الحوارى

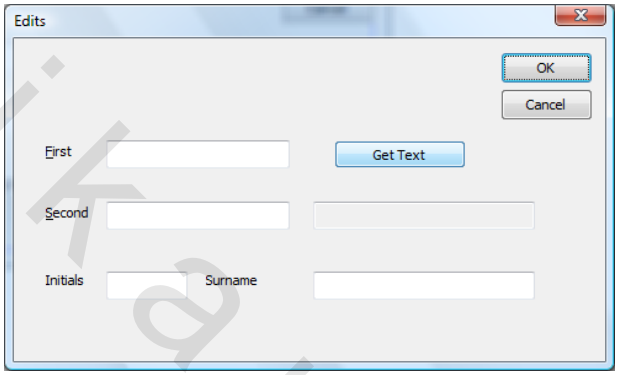
لإضافة مربعات النصوص إلى المربع الحوارى، تابع معنا الخطوات الآتية:

١. افتح قالب المربع الحوارى **IDD_EDITS_DIALOG** داخل نافذة محرر الموارد.
٢. اختر رمز أداة العنوان من مربع الأدوات ثم قم بإضافة عنوان بالجزء السفلى الأيسر من المربع الحوارى (انظر شكل ٦-٧).
٣. قم بإدخال عنوان الأداة داخل الخاصية **Caption** وليكن **Initials**.
٤. اختر رمز مربع النص من مربع الأدوات ثم قم بإضافة مربع نص يمين العنوان **Initials**.
٥. قم بإدخال اسم مربع النص إلى الخاصية **ID** وليكن **IDC_EDIT_INITIALS**.
٦. اختر رمز أداة العنوان من مربع الأدوات ثم قم بإضافة عنوان يمين مربع النص الذى قمت بإضافته منذ قليل (انظر شكل ٦-٧).
٧. قم بإدخال عنوان الأداة إلى الخاصية **Caption** وليكن **Surname**.

obeikandl.com

٨. اختر رمز مربع النص من مربع الأدوات ثم قم بإضافة مربع نص يمين العنوان **Surname**.

٩. قم بزيادة طول مربع النص الأخير إلى اليمين. يجب أن يظهر المربع الحوارى الآن كما فى شكل ٦-٧.



شكل ٦-٧ إضافة أدوات تحكم جديدة إلى المربع الحوارى

إنشاء التصنيف الجديد

لإنشاء التصنيف الفرعى الجديد باستخدام معالج التصنيفات، تابع معنا الخطوات الآتية:

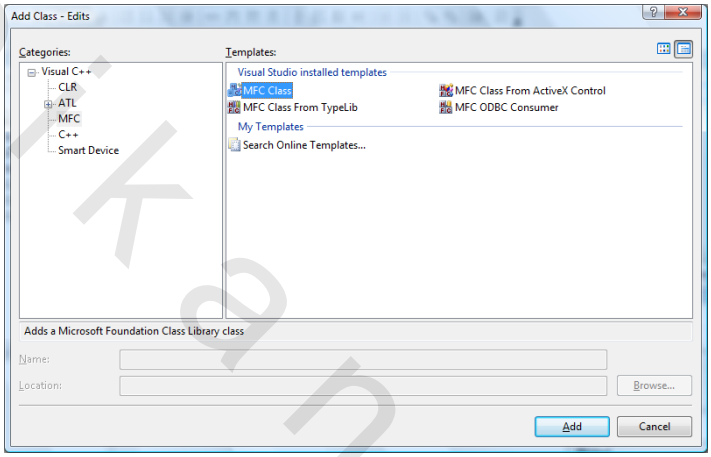
١. نشط التبويب **Class View** ثم انقر اسم المشروع **Edits** بزر الفأرة الأيمن واختر **Add**➢**Class** من القائمة الموضعية الناتجة، يظهر المربع الحوارى **Add Class** (انظر شكل ٦-٨).

٢. اختر **MFC** من مربع السرد **Categories** الموجود بالجانب الأيسر من المربع الحوارى ثم اختر **MFC Class** من قائمة أنواع التصنيفات الموجودة داخل مربع السرد **Templates** بالجانب الأيمن ثم انقر زر **Add**، تظهر نافذة معالج التصنيفات **MFC Class Wizard** (انظر شكل ٦-٩).

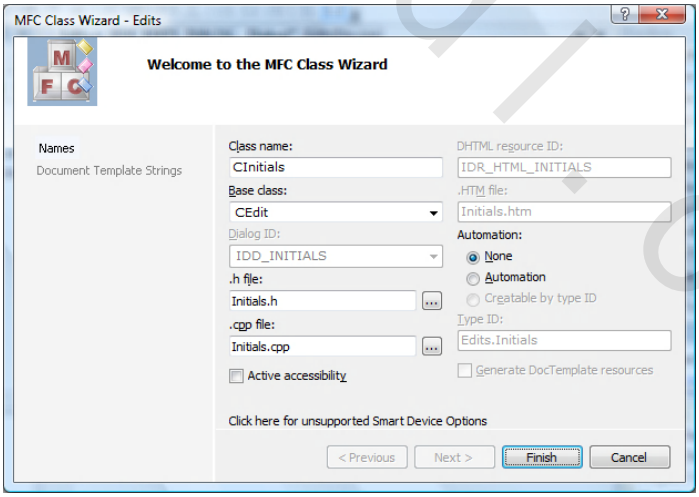
٣. قم بإدخال اسم التصنيف الفرعى إلى مربع النص **Class name** وليكن **CInitials**.

٤. اختر اسم التصنيف الأساسى من مربع السرد **Base Class** وهو **CEdit** فى هذه الحالة.

٥. انقر زر **Finish** لإغلاق نافذة المعالج وإضافة التصنيف الجديد إلى المشروع حيث تلاحظ إضافة التصنيف إلى نافذة عرض التصنيفات **Class View**.



شكل ٦–٨ نافذة أنواع التصنيفات



شكل ٦–٩ إنشاء تصنيف فرعى باستخدام معالج التصنيفات

والآن سنقوم بإنشاء متغير جديد مرتبط بمربع النص **IDC_EDIT_INITIALS**. تابع معنا الخطوات الآتية:

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

٢. انقر التصنيف **CEditsDlg** 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 النص **IDC_EDIT_INITIALS** من مربع السرد **Control ID**.

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

٦. قم بإدخال اسم التصنيف الجديد **CInitials** إلى مربع السرد والتحرير **Variable Type**.

٧. اكتب اسم المتغير فى خانة **Variable name** وليكن **m_EditInitials** واختر درجة المتغير من مربع السرد **Access** وهو **Public** فى هذه الحالة.

٨. انقر زر **Finish** لإغلاق نافذة المعالج وإضافة المتغير الجديد.

حينما قام معالج التصنيفات بإنشاء التصنيف الجديد، تم إنشاء ملفين جديدين هما **Initials.h** و **Initials.cpp**. لذلك لابد من إضافة الموجه **#include** داخل ملف التصنيف الرئيسى **EditsDlg.h** وذلك لأن تصنيف **EditsDlg** يحتوى الآن على متغير من نوع التصنيف الجديد **Initials** والمعرف داخل الملف **Initials.h** .

لإضافة الموجه **#include** الخاص بالتصنيف الجديد، تابع معنا الخطوات الآتية:

١. نشط التبويب **Class View** من نافذة عمل المشروع، تلاحظ وجود التصنيف الجديد **CInitials**.

٢. انقر التصنيف **CEditsDlg** نقراً مزدوجاً، يظهر الملف الرئيسى **Header File** الخاص بالتصنيف داخل نافذة المحرر.

obeikandl.com

٣. قم بإضافة سطر الموجه الجديد قبل تعريف التصنيف مباشرة بحيث يصبح الملف على الصورة الآتية:

```
#include "Initials.h"
Class CEditsDlg : public  Cdialog
{
.....
}
```

لكى يقوم التصنيف الجديد بأداء المهام السابق ذكرها، لابد من إنشاء دالة Handler Function من النوع WM_CHAR وهو عبارة عن رسالة تُرسل إلى البرنامج من قِبَل نظام التشغيل Windows كلما تم الضغط على مفتاح من لوحة المفاتيح ومؤشر الكتابة داخل مربع النص. لتعريف كود هذه الرسالة، تابع معنا الخطوات الآتية:

١. نشط تبويب عرض التصنيفات Class View إذا لم يكن هو التبويب النشط.

٢. تأكد من ظهور نافذة الخصائص أمامك وإلا افتح قائمة View ثم اختر Other Windows ثم Properties Window من القائمة المنسدلة.

٣. انقر التصنيف الجديد CInitials لاختياره.

٤. انقر زر الرسائل من شريط أدوات مربع الخصائص واستخدام شريط التمرير للوصول إلى الرسالة WM_CHAR.

٥. انقر مربع السرد والتحرير المجاور للرسالة ثم اختر OnChar Add، يظهر هيكل الرسالة داخل نافذة الكود.

٦. قم بإدخال الكود التالى للدالة OnChar() التى سيتم استدعاؤها كلما قام المستخدم بإدخال حرف إلى مربع النص Initials.

```
1.    void Initials::OnChar(UINT nChar, UINT nRepCnt, UINT
      nFlags)
2.    {
3.      if (isalpha(nChar))
4.        {
5.          if (islower(nChar))
6.              nChar -= 32;
7.          DefWindowProc(WM_CHAR,nChar,MAKELONG
```


obeikandl.com

```
                                        (nRepCnt,nFlags));
8.          nChar = '.' ;
9.          DefWindowProc (WM_CHAR,nChar,MAKELONG
                                        (nRepCnt,nFlags));
10.       }
11.      if(nChar == VK_BACK)
12.       {
13.             CEdit::OnChar(nChar, nRepCnt, nFlags);
14.             CEdit::OnChar(nChar, nRepCnt, nFlags);
15.       }
16.   }
```

وعن هذا الكود، نوضح ما يلى:

- فى السطر رقم ١ تم استخدام المتغير **nChar** كمعامل للدالة يحتوى على القيمة الرقمية للحرف المدخل.
- فى السطر رقم ٣ تم استدعاء الدالة **()isalpha** للتأكد من أن القيمة المدخلة من لوحة المفاتيح تعبر عن حرف أبجدى وإلا تم الذهاب مباشرة إلى السطر رقم ١١.
- فى السطر رقم ٥ تم اختبار الحرف المدخل، فإذا كان الحرف المدخل كبير، تم الذهاب مباشرة إلى السطر رقم ٧، وإلا تم تحويل الحرف من صغير إلى كبير بطرح القيمة ٣٢ من قيمة الحرف (حيث أن الحرف الكبير أقل من الحرف الصغير بمقدار٣٢ فى منظومة حروف **ANSI**).
- فى السطر رقم ٧ تم كتابة الحرف المدخل باستخدام الدالة الافتراضية **()DefWindowProc**.
- فى السطر رقم ٨ تم استبدال الحرف الموجود بالمتغير **nChar** بالنقطة (.) وفى السطر ٩ يتم كتابة النقطة بعد الحرف باستخدام دالة **()DefWindowProc** مرة أخرى.
- فى السطر رقم ١١يتم اختبار الحرف المدخل.فإذا كان الحرف يمثل مفتاح **Backspace**، يتم استدعاء الدالة **()OnChar** مرتين لحذف النقطة والحرف

المدخل، حيث يعبر المتغير VK_BACK عن مفتاح Backspace، ويوضح الجدول ٦-٢ التالى تعريفات بعض المفاتيح الشهيرة.

جدول ٦-٢ تعريفات بعض المفاتيح الشهيرة

المفتاح	التعريف
Backspace	VK_BACK
Tab	VK_TAB
Left arrow	VK_LEFT
Right arrow	VK_RIGHT
Enter	VK_RETURN
Esc	VK_ESCAPE
Ctrl	VK_CONTROL

والآن قم ببناء البرنامج وتنفيذه ثم قم بإدخال بعض الحروف إلى مربع النص Initials، تلاحظ كتابة الحرف متبوعاً بنقطة هكذا مثلاً A.B. . اضغط مفتاح Back Space من لوحة المفاتيح تلاحظ حذف الحرف والنقطة المصاحبة فى آنٍ واحد.

استخدام مربعات النصوص متعددة السطور

يمكنك كتابة أكثر من سطر داخل مربع النص حيث يتم إضافة أشرطة تمرير رأسية أو أفقية أو كلاهما معاً تبعاً لحجم السطور الموجودة داخل مربع النص. كما أن هناك العديد من الدوال الموجودة فى أى تصنيف حوارى مثل التصنيف CEdit أو التصنيف CInitials الذى أنشأناه منذ قليل، حيث يمكنك استخدام هذه الدوال مع مربع النص متعدد السطور. يوضح جدول ٦-٣ التالى أهم هذه الدوال:

جدول ٦-٣ دوال مربعات النصوص متعددة السطور

الدالة	الاستخدام
GetLineCount()	الحصول على عدد سطور مربع النص
GetLine()	الحصول على سطر من النص
FmtLines()	تمكين أو تعطيل فواصل السطور
SetTabStops()	تعيين مكان معين داخل النص

obeikandl.com

لكى يكون مربع النص **Surname** الذى أضفناه إلى المربع الحوارى منذ قليـــل متعـــدد السطور، تابع معنا الخطوات الآتية:

١. قم بزيادة ارتفاع مربع النص فى قالب المربع الحوارى.
٢. من مربع الخصائص، قم بتخصيص القيمة **True** للخاصية **Multiline**.
٣. قم بتخصيص القيمة **True** للخاصية **Want Return** إذا أردت إضـــافة ســـطر جديد عند ضغط مفتاح الإدخال **Enter**.

يوجد نوع آخر من أنواع مربعات النصوص وهو **Rich Edit Control** والذى يمكنك من خلاله تنسيق كل من الحروف والفقرات كل على حده. وهذا النوع نادراً ما يستخدم داخل المربعات الحوارية، لذا سيتم شرحه فى موضع آخر من الكتاب إن شاء الله.

obeikandl.com