الفصل التاسع عشر

# تقنية ActiveX، الأصوات، الوسائط المتعددة

تعلمت من الفصول السابقة، أساسيات لغة فيجول بيسك. مثل إنشاء النماذج، ووضع الكائنات على النموذج، وربط نص برنامج معين بهذه الكائنات. يركز هذا الفصل على كيفية تصميم التطبيقات في ويندوز مع تقنية ActiveX.

## تقنية ActiveX

تعتبر تقنية ActiveX تقنية مهمة جداً، ويعتمد عليها ويندوز بشكل كبير، وهي بتعريف بسيط: إمكانية تشغيل وحدة نمطية تنفيذية EXE Module، التي تمثل برنامجاً كاملاً، أو جزء من برنامج، أو عنصر تحكم، داخل برنامجك الرئيسي.

بكلام آخر، تضمين برنامجٍ آخر في برنامجك، بدون كتابة سطر إضافي من نص البرنامج.

مثلاً، لو أردت تصميم برنامجٍ، يُشغل ملفات الوسائط المتعددة، مثل ملفات الصوت Wave، أو ملفات الأفلام والحركة AVI، أو ملفات الموسيقى MIDI، لاحتجت إلى وقت طويل جداً لكتابة هذا البرنامج، مع استخدام الكثير من توابع API، (هذا إذا استطعت كتابته أصلاً)، هذا بالإضافة إلى إهدار الكثير من الجهد والوقت.

بدلاً من ذلك، يمكنك الاستفادة من برنامج جاهز (قابل للدمج مع برنامجك الرئيسي)، يقوم بنفس العمل الذي تود تنفيذه. تصور الكسب في الوقت والجهد الذي اختصرته لبناء تطبيقك الرئيسي، وصرف هذا الوقت والجهد عليه.

يشبه هذا الأمر عملية بناء منزل مثلاً، وطلبت من المتعهد **فيجول بيسك**، بناء هذا المنزل تحت إشرافك الشخصي، هنا لديك احتمالين:

* إما أن تبني المنزل كاملاً بنفسك، بمساعدة المتعهد **فيجول بيسك**.
* أو شراء بعض الأشياء الجاهزة من شركات أخرى، كالأبواب والنوافذ والأثاث مثلاً.

تصور الوقت المختصر في الاحتمال الثاني، الذي ينعكس على سرعة الإنجاز. ولا بد أنك تخيلت الآن طريقة عمل تقنية ActiveX، وما توفره من ميزات كثيرة.

في الحقيقة، لقد استعملت هذه التقنية منذ بداية هذا الكتاب.

فزر الأمر Command Button مثلاً، هو عنصر تحكم ActiveX، وجميع الأدوات التي استخدمتها سابقاً هي عناصر تحكم ActiveX.تسمى هذه العناصر بالعناصر القياسية التي تأتي أصلاً مع لغة فيجول بيسك، وهي ليست ملفات منفصلة بل موجودة في نواة اللغة نفسها.

تأتي بعض العناصر الأخرى مع فيجول بيسك أيضاً، ولكنها منفصلة عنه (موجودة في ملف منفصل بامتداد OCX).

لو عدنا لزر الأمر، لوجدنا أن له عملاً معيناً، هو تنفيذ نص برنامج ما، عند النقر عليه، وهو ما فعلناه في معظم أمثلة هذا الكتاب، لكن ألم تسأل نفسك، كيف يغير هذا الزر شكله عند النقر عليه، وأين هي أسطر البرنامج المسئولة عن فعل ذلك؟. قس هذا الأمر على باقي الأدوات والعناصر الأخرى.

للإجابة على هذا السؤال نقول: تعتبر هذه العناصر في الحقيقة، عناصر تحكم تعمل بتقنية ActiveX، وهي عبارة عن وحدة نمطية تنفيذية، لا تعمل بشكل منفصل، إنما ضمن تطبيق رئيسي. (مثل الباب في مثالنا السابق عن بناء المنزل، فالباب بحد ذاته، لا يركب بدون منزل).

دعنا الآن نلخص مزايا استخدام تقنية ActiveX:

* ***تطوير أسرع للبرامج:*** يوفر استخدام عناصر تحكم ActiveX الوقت اللازم ليناء البرنامج الرئيسي، بدلاً من ضياعه في توفير مزايا موجودة أصلاً. بكلام آخر، أنت لست مضطراً لاختراع العجلة من جديد.
* ***ثقة أعلى لتطبيقاتك:*** طورت عناصر التحكم ActiveX التي تستخدمها، شركات خاصة، مهمتها توفير هذه العناصر وبيعها، تحت إشراف مبرمجين مختصين. هذا الأمر يضمن لك وثوقية عالية في تطبيقاتك، التي تعني عدم حصول أخطاء غير متوقعة، لأن هذه العناصر قد اختبرت بشكل كامل، وهي خالية تقريباً من الأخطاء والشوائب.
* ***تقليل الوقت اللازم للتعلم:*** تعمل جميع عناصر تحكم ActiveX بنفس الطريقة تقريباً، وبمجرد تعلم مبدأ عمل تقنية ActiveX، تستطيع استخدام باقي العناصر، مع القليل من معرفة خصوصية كل عنصر بالذات. بالإضافة إلى أنك قد تستخدم عنصر تحكم ما، وأنت لا تعرف بالضبط، كيفية عمل هذا العنصر داخلياً (في الحقيقة، لا يهمك كثيراً كيفية عمله). وبذلك تكون قد وفرت وقتاً طويلاً في تعلم كيف يقوم هذا العنصر بعمله، لو أردت تنفيذ هذا العمل بنفسك.
* *و****اجهة استخدام مألوفة للمستخدم:*** يجعل استخدام عناصر التحكم ActiveX القياسية، برنامجك أو تطبيقك ذو واجهة استخدام مألوفة من قبل المستخدم، وهي مشابهة لباقي واجهات التطبيقات الأخرى، التي اعتاد وتدرب المستخدم عليها. تصور لو أنك أنشأت زر أمر خاص بك، ومختلف عن زر الأمر القياسي، توقع عندئذ أن المستخدم قد لا يعرف أن هذا الشيء الذي أنشأته هو زر أمر.

## قواعد استخدام عناصر ActiveX

هناك بعض القواعد الرئيسية التي يجب اتباعها عند استخدام أدوات ActiveX:

* لا تسرف في استخدام أدوات ActiveX، بل استخدم ما يلزمك منها فقط. لأن كثرة الأدوات في تطبيقاتك، سوف تبطئ عمل التطبيق بشكل ملحوظ، وخصوصاً عند بداية التنفيذ.
* حاول قدر الإمكان، استخدام الأدوات القياسية التي تأتي أصلاً مع لغة فيجول بيسك. وبذلك تحصل على عدة مزايا منها:
* ضمان استمرار تطوير هذه العناصر من قبل Microsoft في الإصدارات الجديدة من فيجول بيسك (وهو ما لن تستطيع ضمانه مع الشركات الأخرى، فيما لو استخدمت عناصرها).
* تقليل حجم البرنامج الكلي (أقراص الإعداد).
* التأكد أن هذه الأدوات القياسية، تعمل بشكل خال من الأخطاء تقريباً، وبتوافق عالي مع النظام ككل.
* يظهر برنامجك بالنسبة للمستخدم بشكل مألوف (سرعة في التعلم).
* هناك قاعدة تقول: إذا استطعت الاستغناء عن عنصر تحكم ActiveX بقليل من الجهد، فافعل ذلك فوراً. لأن هذا الأمر يزيد من سرعة تنفيذ وثقة تطبيقك بآن واحد.
* لا تستخدم عناصر التحكم ActiveX ذات الإصدار الدعائي أو الاستعراضي   
  Demo Version، بل استخدم العناصر الأصلية الكاملة والمسجلة Register. يسبب استخدام العناصر الدعائية الكثير من المشاكل، وهي نسخة غير قابلة للتوزيع غالباً. طبعاً، الاستخدام الذي قصدته هو الاستخدام النهائي، وليس بقصد تجربة العنصر. هذا يعني أنه يمكنك استخدام العنصر الدعائي لتجربته فقط، ولكن لا تضعه أبداً في أقراص الإعداد.

## برنامج الوسائط المتعددة

يوضح لنا برنامج الوسائط المتعددة، كيفية استخدام عنصر التحكم Microsoft Multimedia (اسم الملف هو MCI32.OCX)، لإنشاء برنامج يمكنه تشغيل ملفات الوسائط المتعددة.

يتمكن برنامج الوسائط المتعددة (بالاعتماد على العنصر MCI32.OCX)، من فتح ملفات الصوت ذات الامتداد WAV، وفتح ملفات الحركة ذات الامتداد AVI، وأيضاً فتح ملفات الموسيقى ذات الامتداد MID.

يوفر برنامج الوسائط المتعددة، واجهة استخدام سهلة، وقائمة تمكننا من فتح أحد أنواع الملفات الثلاثة السابقة، بعد فتح الملف يمكننا:

* عزف الملف Play.
* إيقاف عزف الملف Stop.
* إيقاف مؤقت، لعزف الملف Pause.
* إعادة المؤشر في أول الملف Back.
* وضع المؤشر في نهاية الملف End.
* تقديم الملف صورة للأمام Next Frame.
* ترجيع الملف صورة للخلف Previous Frame.انظر الشكل 19-1.

|  |  |
| --- | --- |
|  | الشكل 19-1  برنامج الوسائط المتعددة. |

كما يوفر برنامج الوسائط المتعددة، بعض المعلومات عن الملف الحالي تتلخص فيما يلي:

* اسم ومسار الملف الحالي المفتوح.
* طول الملف الحالي (تتغير وحدة الطول حسب نوع الملف).
* موقع العزف الحالي من الملف (إظهاره كرقم صحيح).
* مؤشر يدلنا على موقع العزف الحالي في الملف، بمساعدة شريط الإزاحة الأفقي.

### إنشاء برنامج الوسائط المتعددة

لنعمل الآن على إنشاء برنامج الوسائط المتعددة:

* أنشئ الدليل C:\VB5Prg\Ch19، لكي تحفظ عملك فيه.
* ابدأ مشروعاً جديداً بنوع Standard EXE.
* احفظ النموذج From1 باسم MCI.frm، في الدليل C:\VB5Prg\Ch19. واحفظ المشروع باسم MCI.Vbp في نفس الدليل السابق.

### إضافة عنصر تحكم OCX إلى المشروع

قبل التمكن من وضع عنصر OCX على النموذج، يجب أولاً إضافته إلى المشروع نفسه.

سنضيف العنصر MCI32.OCX إلى مشروعنا MCI.Vbp:

* اختر البند Component من قائمة **Project**.

يستجيب فيجول بيسك بإظهار مريع الحوار Component.

* اختر البند Microsoft Multimedia Control 5.0 من صفحة Controls في مربع الحوار Components (انظر الشكل 19-2).

|  |  |
| --- | --- |
|  | الشكل 19-2  إظهار مربع الحوار Components  لاختيار عنصر تحكم الوسائط المتعددة. |

* قبل الخروج من مربع الحوار Components، اختر البند   
  Microsoft Common Dialog Control 5.0 أيضاً، لأننا سنحتاجه في برنامج الوسائط المتعددة.
* انقر الزر **OK** للخروج من مربع الحوار Components.

بعد تنفيذ الخطوات السابقة، ينبغي ظهور رمزين جديدين: أحدهما لعنصر تحكم الوسائط المتعددة والثاني لعنصر تحكم مربعات الحوار الشائعة. (انظر الشكل 19-3).

|  |  |
| --- | --- |
|  | الشكل 19-3  نافذة الأدوات بعد إضافة عنصري الوسائط المتعددة ومربعات الحوار الشائعة. |

### التصميم المرئي لبرنامج الوسائط المتعددة

لنبدأ الآن عملية التصميم المرئي لبرنامج الوسائط المتعددة:

* أسند القيم الموضحة لاحقاً، للخصائص التي تتبع لها للنموذج الافتراضي Form1:

Name: frmMCI

Caption: برنامج الوسائط المتعددة

RightToLeft: True

اربط القائمة الموضحة بالجدول 19-1 مع النموذج frmMCI.

جدول 19-1. قائمة النموذج frmMCI.

|  |  |
| --- | --- |
| **Caption** | **Name** |
| &ملف | mnuFile |
| …فتح ملف &صوت | mnuOpenWave |
| …فتح ملف &حركة | mnuOpenAVI |
| …فتح ملف &موسيقى | mnuOpenMIDI |
| …- | mnuSep1 |
| **…&خروج** | **mnuExit** |

* أضف عنصر الوسائط المتعددة للنموذج frmMCI، وأسند القيمة mciMyMedia للخاصية Name.
* أضف عنصر مربعات الحوار الشائعة للنموذج frmMCI، وأسند القيمة dlgGetFile للخاصية Name.

بعد الانتهاء من الخطوات السابقة، ينبغي أن يصبح النموذج frmMCI، كما في الشكل 19-4.

|  |  |
| --- | --- |
|  | الشكل 19-4  النموذج frmMCI بعد إضافة القائمة وعنصري الوسائط المتعددة ومربعات الحوار الشائعة. |

لنكمل الآن عملية بناء النموذج frmMCI:

* أضف عنصر تحكم الوقت Timer للنموذج frmMCI.
* أسند له الخصائص التالية:

Name: tmrGetPosition

Interval: 50

Enabled: False

* أضف باقي العناصر وفقاً للجدول 19-2 (انتبه لأنك أضفت بعض العناصر الموجودة في الجدول 19-2).

جدول 19-2 العناصر المكونة لبرنامج الوسائط المتعددة وخصائصها.

|  |  |  |
| --- | --- | --- |
| **الكائن** | **الخاصية** | **القيمة** |
| **Form** | **Name** | **frmMCI** |
|  | Caption | برنامج الوسائط المتعددة |
|  | RightToLeft | True |
| **MMControl** | **Name** | **mciMyMedia** |
| **Common Dialy** | **Name** | **dlgGetFile** |
| **Timer** | **Name** | **tmrGetPosition** |
|  | Interval | 50 |
|  | Enabled | False |
| **Label** | **Name** | **lblTitle1** |
|  | Caption | اسم الملف الحالي |
| **Label** | **Name** | **lblTitle2** |
|  | Caption | الموقع الحالي |
| **Label** | **Name** | **lblTitle3** |
|  | Caption | طول الملف |
| **الكائن** | **الخاصية** | **القيمة** |
| **Label** | **Name** | **lblCurrentFile** |
|  | Caption | فارغ |
| **Label** | **Name** | **lblCurPosition** |
|  | Caption | فارغ |
| **Label** | **Name** | **lblLastPosition** |
|  | Caption | فارغ |
| **HscrollBar** | **Name** | **hsbPosition** |

بعد الانتهاء من إضافة جميع العناصر الموجودة في الجدول السابق، ينبغي أن يظهر النموذج frmMCI كما في الشكل 19-5.

|  |  |
| --- | --- |
|  | الشكل 19-5  النموذج frmMCI بعد الانتهاء من عملية تصميمه. |

#### كتابة نص برنامج الوسائط المتعددة

بعد الانتهاء من تصميم النموذج frmMCI، علينا كتابة نص البرنامج:

* اكتب الأسطر التالية في الإجراء mnuOpenWave\_Click():

Private Sub mnuOpenWave\_Click()

Dim mmFileName As String

'filename تهيئة الخاصية

dlgGetFile.filename = ""

'تحديد نوع الملف الذي سيفتحه مربع الحوار

dlgGetFile.Filter = "ملفات الصوت(\*.wav)|\*.wav"

'إظهار مربع الحوار فتح ملف

dlgGetFile.ShowOpen

'mmFileName إسناد الملف الذي اختاره المستخدم للمتحول

mmFileName = dlgGetFile.filename

'إذا لم يختر المستخدم ملفا، فاخرج من الإجراء فورا

If mmFileName = "" Then

Exit Sub

End If

'إغلاق الجهاز السابق في حال وجود جهاز مفتوح

mciMyMedia.Command = "Close"

'تحديد نوع الجهاز بأنه جهاز ملفات الصوت

mciMyMedia.DeviceType = "WaveAudio"

'تحديد الملف المراد فتحه

mciMyMedia.filename = mmFileName

'إعطاء أمر فتح للجهاز الصوتي

mciMyMedia.Command = "Open"

'تحديد أكبر موقع ممكن لشريط الإزاحة الأفقي

hsbPosition.Max = mciMyMedia.Length

'عرض قيمة طول الملف في أداة العنونة

lblLastPositiom.Caption = mciMyMedia.Length

'عرض الملف الحالي في أداة العنونة

lblCurrentFile.Caption = mmFileName

End Sub

* اكتب الأسطر التالية في الإجراء mnuOpenAVI\_Click():

Private Sub mnuOpenAVI\_Click()

Dim mmFileName As String

'filename تهيئة الخاصية

dlgGetFile.filename = ""

'تحديد نوع الملف الذي سيفتحه مربع الحوار

dlgGetFile.Filter = "ملفات الحركة(\*.avi)|\*.avi"

'إظهار مربع الحوار فتح ملف

dlgGetFile.ShowOpen

'mmFileName إسناد الملف الذي اختاره المستخدم للمتحول

mmFileName = dlgGetFile.filename

'إذا لم يختر المستخدم ملفا، فاخرج من الإجراء فورا

If mmFileName = "" Then

Exit Sub

End If

'إغلاق الجهاز السابق في حال وجود جهاز مفتوح

mciMyMedia.Command = "Close"

'تحديد نوع الجهاز بأنه جهاز ملفات الحركة

mciMyMedia.DeviceType = "AVIVideo"

'تحديد الملف المراد فتحه

mciMyMedia.filename = mmFileName

'إعطاء أمر فتح للجهاز الحركي

mciMyMedia.Command = "Open"

'تحديد أكبر موقع ممكن لشريط الإزاحة الأفقي

hsbPosition.Max = mciMyMedia.Length

'عرض قيمة طول الملف في أداة العنونة

lblLastPositiom.Caption = mciMyMedia.Length

'عرض الملف الحالي في أداة العنونة

lblCurrentFile.Caption = mmFileName

End Sub

* اكتب الأسطر التالية في الإجراء mnuOpenMIDI\_Click():

Private Sub mnuOpenMIDI\_Click()

Dim mmFileName As String

'filename تهيئة الخاصية

dlgGetFile.filename = ""

'تحديد نوع الملف الذي سيفتحه مربع الحوار

'dlgGetFile.Filter = "All Files|(\*.rmi)"

'إظهار مربع الحوار فتح ملف

dlgGetFile.ShowOpen

'mmFileName إسناد الملف الذي اختاره المستخدم للمتحول

mmFileName = dlgGetFile.filename

'إذا لم يختر المستخدم ملفا، فاخرج من الإجراء فورا

If mmFileName = "" Then

Exit Sub

End If

'إغلاق الجهاز السابق في حال وجود جهاز مفتوح

mciMyMedia.Command = "Close"

'تحديد نوع الجهاز بأنه جهاز ملفات الموسيقى

mciMyMedia.DeviceType = "Sequencer"

'تحديد الملف المراد فتحه

mciMyMedia.filename = mmFileName

'إعطاء أمر فتح للجهاز الصوتي

mciMyMedia.Command = "Open"

'تحديد أكبر موقع ممكن لشريط الإزاحة الأفقي

hsbPosition.Max = mciMyMedia.Length

'عرض قيمة طول الملف في أداة العنونة

lblLastPositiom.Caption = mciMyMedia.Length

'عرض الملف الحالي في أداة العنونة

lblCurrentFile.Caption = mmFileName

End Sub

* اكتب الأسطر التالية في الإجراء mnuExit\_Click():

Private Sub mnuExit\_Click()

End

End Sub

* اكتب الأسطر التالية في الإجراء trmGetPosition\_Timer():

Private Sub tmrGetPosition\_Timer()

lblCurPosition.Caption = mciMyMedia.Position

hsbPosition.Value = mciMyMedia.Position

End Sub

* اكتب الأسطر التالية في الإجراء mciMyMedia\_PlayClick():

Private Sub mciMyMedia\_Click()

End Sub

* اكتب الأسطر التالية في الإجراء Form\_Unload():

Private Sub Form\_Unload(Cancel As Integer)

mciMyMedia.Command = "Close"

End Sub

* احفظ عملك الآن.

### تنفيذ برنامج الوسائط المتعددة

بعد الانتهاء من كتابة نص البرنامج، يصبح جاهزاً للتنفيذ:

* نفّذ البرنامج بضغط المفتاح F5.

يستجيب فيجول بيسك بإظهار نافذة برنامج الوسائط المتعددة.

* اختر البند **فتح ملف صوتي** من قائمة ملف.

يستجيب برنامج الوسائط المتعددة بإظهار مربع الحوار فتح.

* اختر أحد الملفات الصوتية التي لها الامتداد Wav، ثم انقر الزر **فتح** في مربع الحوار فتح.

تجد أمثلة عن ملفات الصوت في القرص المدمج المرفق مع الكتاب، في دليل الوسائط المتعددة.

* يعرض برنامج الوسائط المتعددة اسم الملف الذي اخترته ومساره الكامل، كما يعرض أيضاً طول الملف بوحدة جزء من الألف من الثانية.

استعد برنامج الوسائط المتعددة لعزف الملف الذي اخترته:

* انقر الزر **Play** الموجود على عنصر تحكم الوسائط المتعددة.

يستجيب برنامج الوسائط المتعددة بعزف الملف الذي اخترته، ويظهر لك الموقع الحالي للعزف بطريقتين:

* رقم يحدد الزمن الذي تم عزفه إلى الآن.
* استخدام مؤشر شريط الإزاحة الأفقي.
* لإعادة عزف الملف مرة ثانية، يجب نقر الزر **Back** (السهم الموجود فى أقصى اليسار)، لإعادة مؤشر العزف إلى بداية الملف مرة أخرى، ثم نقر الزر **Play**.
* أثناء عزف الملف، تستطيع نقر الزر **Stop** (مربع غامق)، لإيقاف العزف نهائياً، أو زر **Pause** (خطين شاقوليين) لإيقاف العزف مؤقتاً.
* لعزف نوع آخر من الملفات وليكن ملفات الحركة، اختر البند **فتح ملف حركة** من قائمة **ملف**، واختر أحد الملفات التي لها الامتداد AVI (ملفات حركة).

تجد أمثلة عن ملفات الحركة، في القرص المدمج المرفق مع الكتاب، في دليل الوسائط المتعددة.

يعرض البرنامج أيضاً معلومات عن الملف الذي اخترته مثل اسم الملف ومساره الكامل، وطول الملف بالثواني.

* انقر الزر **Play** لعرض الملف الذي اخترته.

يستجيب برنامج الوسائط المتعددة بفتح نافذة، يعرض الفيلم (ملف AVI) عبرها. وعنوان النافذة عبارة عن اسم الملف الذي يتم عرضه فيها.

* انقر الزر **Next Frame** أو الزر **Previous Frame**، لعرض الملف صورة صورة (Frame by Frame).
* تستطيع إغلاق نافذة عرض الفيلم بنقر أيقونة الإغلاق في النافذة نفسها، أو ببساطة فتح ملف آخر (مهما كان نوعه). لأن فتح ملف جديد يتسبب في إغلاق الملف الحالي آلياً.

لفتح النوع الثالث من ملفات الوسائط المتعددة:

* اختر البند **فتح ملف موسيقي** من قائمة **ملف**، واختر أحد الملفات التي لها الامتداد MID.

تجد أمثلة عن ملفات الموسيقى في القرص المدمج المرفق مع الكتاب، في دليل الوسائط المتعددة.

* انقر الزر **Play** لعزف الملف الموسيقي الذي اخترته.
* تمرن على البرنامج بجميع أقسامه ثم اختر البند **خروج** من قائمة **ملف** لإنهاء البرنامج.

### كيف يعمل برنامج الوسائط المتعددة

يعتمد برنامج الوسائط المتعددة اعتماداً أساسياً على عنصر تحكم الوسائط المتعددة mciMyMedia. وتنحصر وظيفة البرنامج في تحديد الخصائص، وإعطاء الأوامر لهذا العنصر، ليقوم بفتح وعزف ملفات الوسائط المتعددة المختلفة.

#### نص البرنامج للإجراء mnuOpenWave\_Click()

ينفذ هذا الإجراء عند اختيار المستخدم البند **فتح ملف صوتي** من قائمة **ملف** في برنامج الوسائط المتعددة. يسمح هذا الإجراء للمستخدم، باختيار ملف صوتي (WAV)، ومن ثم فتحه وعزفه.

لقد كتبت النص التالي في الإجراء mnuOpenWave\_Click():

Private Sub mnuOpenWave\_Click()

Dim mmFileName As String

'filename تهيئة الخاصية

dlgGetFile.filename = ""

'تحديد نوع الملف الذي سيفتحه مربع الحوار

dlgGetFile.Filter = "ملفات الصوت(\*.wav)|\*.wav"

'إظهار مربع الحوار فتح ملف

dlgGetFile.ShowOpen

'mmFileName إسناد الملف الذي اختاره المستخدم للمتحول

mmFileName = dlgGetFile.filename

'إذا لم يختر المستخدم ملفا، فاخرج من الإجراء فورا

If mmFileName = "" Then

Exit Sub

End If

'إغلاق الجهاز السابق في حال وجود جهاز مفتوح

mciMyMedia.Command = "Close"

'تحديد نوع الجهاز بأنه جهاز ملفات الصوت

mciMyMedia.DeviceType = "WaveAudio"

'تحديد الملف المراد فتحه

mciMyMedia.filename = mmFileName

'إعطاء أمر فتح للجهاز الصوتي

mciMyMedia.Command = "Open"

'تحديد أكبر موقع ممكن لشريط الإزاحة الأفقي

hsbPosition.Max = mciMyMedia.Length

'عرض قيمة طول الملف في أداة العنونة

lblLastPositiom.Caption = mciMyMedia.Length

'عرض الملف الحالي في أداة العنونة

lblCurrentFile.Caption = mmFileName

End Sub

صرحت العبارة الأولى عن المتحول mmFileName، ووظيفته تخزين اسم الملف الذي اختاره المستخدم. أما العبارة:

dlgGetFile.filename = ""

تهيئ الخاصية FileName التابعة لمربع الحوار dlgGetFile وإسناد القيمة "لا شيء" لها.

بعد ذلك، جاء دور معرفة نوع الملف الذي سيفتحه المستخدم (معرفة امتداد الملف) عن طريق العبارة التالية:

dlgGetFile.Filter = "ملفات الصوت(\*.wav)|\*.wav"

يتم إسناد أنواع الملفات التي يستطيع مربع الحوار فتحها، إلى الخاصية Filterالتابعة لمربع الحوارdlgGetFile.

بعد معرفة نوع الملفات التي نريد فتحها، نعطي الأمر بالظهور لمربع الحوار فتح، عن طريق الأمر ShowOpen وذلك كالتالي:

dlgGetFile.ShowOpen

تتوقف عملية التنفيذ عند هذا السطر، لحين انتهاء المستخدم من اختيار ملف ما، ونقر الزر **فتح** الموجود في مربع الحوار.

هنا لدينا احتمالان:

* اختار المستخدم ملفاً صوتياً.
* لم يختر المستخدم أي ملف، أو نقر الزر **إلغاء الأمر** في مربع الحوار.

لمعرفة نوع الاحتمال الحاصل، لجأنا إلى فحص اسم الملف الذي اختاره المستخدم، عن طريق العبارة If … Then التالية:

If mmFileName = "" Then

Exit Sub

End If

يفحص هذا الشرط قيمة المتحول mmFileName، الممثل لاسم الملف الذي اختاره المستخدم، إذا كانت قيمته تساوي "لا شيء"، (لم يختر المستخدم ملفاً)، فإنه ببساطة يخرج بدون إكمال تنفيذ باقي الإجراء.

لا بد أن تكون لاحظت الآن، لماذا قمنا بتهيئة الخاصية FileName التابعة لمربع الحوار، وإسناد القيمة "لا شيء" لها، قبل إظهار مربع الحوار. لأننا سنفحص هذه الخاصية بعد الخروج من مربع الحوار، ونرى هل أصبح لها قيمة أخرى غير قيمة "لا شيء"، والتي تعني أن المستخدم لم يختر ملفاً كما ذكرنا سابقاً.

ينفذ باقي الإجراء، عند اختيار ملف صوتي، ويبدأ من العبارة التالية:

mciMyMedia.Command = "Close"

تُغلق هذه العبارة، أي جهاز مفتوح مسبقاً، (يقصد بالجهاز هنا: أحد ملفات الوسائط المتعددة المفتوحة والجاهزة للعزف).

بعد إغلاق الجهاز المفتوح (في حال وجوده)، يُحدد نوع الجهاز المطلوب فتحه أو تشغيله (وهو في هذه الحالة، الجهاز "Wave Audio"). يحدد نوع الجهاز حسب نوع الملف المطلوب فتحه.

إذا كان نوع الملف صوتياً، يحدد الجهاز على أنه من النوع "Wave Audio"، أما في حال كونه موسيقياً، يكون نوع الجهاز الصحيح هو "Sequencer"، وفي حال كونه نوع الجهاز فيلماً متحركاً، يكون نوع الجهاز "AVI Video".

نُسند نوع الجهاز المطابق، للخاصية DeviceType التابعة لعنصر الوسائط المتعددة كالتالي:

mciMyMedia.DeviceType = "WaveAudio"

بعد تحديد نوع الجهاز المطلوب، نسند اسم الملف الذي اختاره المستخدم إلى الخاصية FileName التابعة لعنصر الوسائط المتعددة mciMyMedia، حتى يتعرف عنصر الوسائط المتعددة على اسم ومسار الملف المراد فتحه.

بعد معرفة اسم الملف، نُعطى لعنصر الوسائط المتعددة، الأمر Open عن طريق العبارة التالية:

mciMyMedia.Command = "Open"

ليعمل على فتح الملف وتهيئته للعزف، ومعرفة كل المعلومات الخاصة بهذا الملف.

أحد هذه المعلومات، هي طول الملف الصوتي، (بوحدة الملي ثانية)، حيث نستطيع الحصول عليها عن طريق الخاصية Length التابعة لعنصر الوسائط المتعددة mciMyMedia.

استفدنا من قيمة طول الملف الصوتي، في تحديد أكبر مجال ممكن لشريط الإزاحة الأفقي، عن طريق إسناد هذه القيمة للخاصية Max التابعة لشريط الإزاحة الأفقي كالتالي:

hsbPosition.Max = mciMyMedia.Length

أظهرنا قيمة طول الملف الصوتي في اللافتة، حتى يعرف المستخدم طول الملف المفتوح، بأجزاء من الألف من الثانية، وذلك كالتالي:

lblLastPosition.Caption = mciMyMedia.Length

أيضاً، أظهرنا اسم الملف ومساره الكامل، (حتى يعرف المستخدم اسم الملف المفتوح الحالي) في لافتة أخرى، عن طريق العبارة التالية:

lblCurrentFile.Caption = mmFileName

بعد الانتهاء من تنفيذ كامل عبارات هذا الإجراء، يصبح الملف الذي اختاره المستخدم جاهزاً للعزف.

#### نص البرنامج للإجراء mnuOpenMIDI\_Click ()

ينفذ هذا الإجراء عند اختيار المستخدم البند **فتح ملف موسيقى** من قائمة ملف:

Private Sub mnuOpenMIDI\_Click()

Dim mmFileName As String

'filename تهيئة الخاصية

dlgGetFile.filename = ""

'تحديد نوع الملف الذي سيفتحه مربع الحوار

dlgGetFile.Filter = "ملفات الموسيقى(\*.mid)|\*.mid"

'إظهار مربع الحوار فتح ملف

dlgGetFile.ShowOpen

'mmFileName إسناد الملف الذي اختاره المستخدم للمتحول

mmFileName = dlgGetFile.filename

'إذا لم يختر المستخدم ملفا، فاخرج من الإجراء فورا

If mmFileName = "" Then

Exit Sub

End If

'إغلاق الجهاز السابق في حال وجود جهاز مفتوح

mciMyMedia.Command = "Close"

'تحديد نوع الجهاز بأنه جهاز ملفات الموسيقى

mciMyMedia.DeviceType = "Sequencer"

'تحديد الملف المراد فتحه

mciMyMedia.filename = mmFileName

'إعطاء أمر فتح للجهاز الصوتي

mciMyMedia.Command = "Open"

'تحديد أكبر موقع ممكن لشريط الإزاحة الأفقي

hsbPosition.Max = mciMyMedia.Length

'عرض قيمة طول الملف في أداة العنونة

lblLastPositiom.Caption = mciMyMedia.Length

'عرض الملف الحالي في أداة العنونة

lblCurrentFile.Caption = mmFileName

End Sub

يعمل هذا الإجراء، بنفس الأسلوب الذي شرحناه سابقاً مع وجود اختلافين اثنين هما:

* تحديد نوع الملف الذي يستطيع المستخدم اختياره عن طريق مربع الحوار، على أنه النوع ذو الامتداد (MID). والعبارة المسئولة عن ذلك هي:

dlgGetFile.Filter = "ملفات الموسيقى(\*.mid)|\*.mid"

* تحديد نوع جهاز آخر يمكنه فتح الملفات الموسيقية، وهو الجهاز ذو النوع "Sequencer"، عن طريق العبارة التالية:

mciMyMedia.DeviceType = "Sequencer"

#### نص البرنامج للإجراء mnuOpenAVI\_Click()

* ينفذ هذا الإجراء آلياً، عند اختيار البند **فتح ملف حركة** من قائمة ملف:

Private Sub mnuOpenAVI\_Click()

Dim mmFileName As String

'filename تهيئة الخاصية

dlgGetFile.filename = ""

'تحديد نوع الملف الذي سيفتحه مربع الحوار

dlgGetFile.Filter = "ملفات الحركة(\*.avi)|\*.avi"

'إظهار مربع الحوار فتح ملف

dlgGetFile.ShowOpen

'mmFileName إسناد الملف الذي اختاره المستخدم للمتحول

mmFileName = dlgGetFile.filename

'إذا لم يختر المستخدم ملفا، فاخرج من الإجراء فورا

If mmFileName = "" Then

Exit Sub

End If

'إغلاق الجهاز السابق في حال وجود جهاز مفتوح

mciMyMedia.Command = "Close"

'تحديد نوع الجهاز بأنه جهاز ملفات الحركة

mciMyMedia.DeviceType = "AVIVideo"

'تحديد الملف المراد فتحه

mciMyMedia.filename = mmFileName

'إعطاء أمر فتح للجهاز الحركي

mciMyMedia.Command = "Open"

'تحديد أكبر موقع ممكن لشريط الإزاحة الأفقي

hsbPosition.Max = mciMyMedia.Length

'عرض قيمة طول الملف في أداة العنونة

lblLastPositiom.Caption = mciMyMedia.Length

'عرض الملف الحالي في أداة العنونة

lblCurrentFile.Caption = mmFileName

End Sub

* يعمل هذا الإجراء أيضاً، بنفس أسلوب عمل الإجراءين السابقين، ومع وجود نفس الاختلافين السابقين وهما:
* تحديد نوع الملفات التي نريد اختيارها من مربع الحوار، على أنها من النوع AVI، وذلك كالتالي:

dlgGetFile.Filter = "ملفات الحركة(\*.avi)|\*.avi"

* تحديد نوع الجهاز القادر على عرض الملف الحركي، وهو الجهاز ذو النوع "AVIVideo"، عن طريق العبارة التالية:

mciMyMedia.DeviceType = "AVIVideo"

ملاحظة

يتطلب برنامج الوسائط المتعددة وجود بطاقة صوت، لإصدار الأصوات والموسيقى من خلاله، أما إظهار الملفات الحركية، فهي لا تتطلب بطاقات إضافية.

## التقنية المستخدمة لعرض الموقع الحالي خلال عزف الملف

بعد اختيار أحد ملفات الوسائط المتعددة، وفتحه عن طريق القائمة **ملف**، يصبح البرنامج جاهزاً لعزف هذا الملف، عن طريق النقر على الزر **Play،** يظهر لك الموقع الحالي للعزف خلال عزفه، بأسلوبين:

* إظهار الموقع الحالي للعزف، كرقم يمثل الوقت الذي انقضى منه، في اللافتة lblCurPosition.
* إظهار الموقع الحالي للعزف، عن طريق مؤشر شريط الإزاحة الأفقي hsbPosition.

### كيف يعلم البرنامج الموقع الحالي للعزف ويظهره للمستخدم؟

نستطيع معرفة الموقع الحالي للعزف، عن طريق قراءة قيمة الخاصية Position التابعة لعنصر الوسائط المتعددة mciMyMedia.

لكننا نريد معرفة الموقع الحالي للعزف، خلال مرحلة العزف نفسها، وعلى فترات صغيرة جداً. لذلك استخدمنا لهذه الغاية، عنصر تحكم الميقاتية Timer، وجعلناه يقرأ الموقع الحالي للعزف، وإظهاره كل 1/20 من الثانية.

أسندنا القيمة 50 للخاصية Interval التابعة للميقاتية trmGetPosition، خلال مرحلة التصميم، وتعني تنفيذ الحادثة Timer التابعة للميقاتية، كل 50 ميلي ثانية.

نص البرنامج المكتوب في الحادثة Timer هو:

Private Sub tmrGetPosition\_Timer()

lblCurPosition.Caption = mciMyMedia.Position

hsbPosition.Value = mciMyMedia.Position

End Sub

كما قلنا سابقاً، يُنفذ هذا الإجراء آلياً كل 50 ميلي ثانية، ويُظهر قيمة الخاصية Position التابعة لعنصر الوسائط المتعددة (الممثلة لموقع العزف الحالي)، في اللافتة lblCurPosition كالتالي:

lblCurPosition.Caption = mciMyMedia

يُسند أيضاً قيمة الخاصية Position التابعة لعنصر الوسائط المتعددة، للخاصية Value التابعة لشريط الإزاحة الأفقي، لتنقل مؤشر شريط الإزاحة الأفقي للموقع الحالي للعزف، كالتالي:

hsbPosition.Valve = mciMyMedia.Position

وبذلك نضمن تحديث دائم، لموقع العزف الحالي الظاهر للمستخدم.

لكن توجد مشكلة مع التوقيت، هل عرفتها؟.

يعمل التوقيت دائماً خلال مرحلة عزف الملف، وخلال فترة التوقف أيضاً، وهذا الأمر غير مقبول برمجياً (عمل التوقيت الدائم، خلال فترة تنفيذ البرنامج).

إذاً، يجب تشغيل الميقاتية خلال فترة عزف الملف فقط، وإيقافه في الفترات الأخرى، وذلك عن طريق الخاصية Enabled، التي تعطّله إذا أُسندت القيمة False إليه، أو تجعله يعمل إذا أُسندت القيمة True إليها.

أسندنا القيمة False للخاصية Enabled التابعة للتوقيت، خلال مرحلة تصميم البرنامج، لأننا نريد تعطيل التوقيت عند بداية تشغيل البرنامج. لذلك، يجب تفعيل التوقيت فقط، خلال فترة عزف الملف، وتعطيلها خارج هذه الفترة.

يوفّر عنصر تحكم الوسائط المتعددة، الحادثة Play\_Click()، التي تنفذ آلياً عند نقر المستخدم للزر **Play** (بدء العزف)، وهي أنسب مكان لوضع العبارة التي تفعّل التوقيت، وذلك كالتالي:

Private Sub mciMyMedia\_PlayClick(Cancel As Integer)

tmrGetPosition.Enabled = True

End Sub

بعد الانتهاء من العزف، ينفذ عنصر تحكم الوسائط المتعددة الحادثة Done (نهاية العزف) آلياً. لذلك، هذه الحادثة تعتبر الموقع المناسب لوضع العبارة التي تعطّل التوقيت مرة ثانية، (لأن العزف انتهى، ولا داعي لبقاء التوقيت تعمل في هذه الفترة). وذلك كالتالي:

Private Sub mciMyMedia\_Done(NotifyCode As Integer)

tmrGetPosition.Enabled = False

End Sub

#### نص البرنامج للإجراء Form\_Unload ()

ينفذ الإجراء Form\_Unload() آلياً، عند إنهاء البرنامج وإزالته من ذاكرة ويندوز. كما أن الحادثة Form\_Load() تنفذ آلياً، عند بداية تشغيل البرنامج ووضع النافذة في الذاكرة.

لضمان إغلاق كافة الأجهزة المفتوحة من قِبل عنصر تحكم الوسائط المتعددة، وضعنا العبارة التالية في الإجراء Form\_Unload() كالتالي:

Private Sub Form\_Unload(Cancel As Integer)

mciMyMedia.Command = "Close"

End Sub

## تطوير برنامج الوسائط المتعددة

سنطور الآن برنامج الوسائط المتعددة، ليصبح قادراً على قراءة الأقراص الليزرية الموسيقية.

* أضف بنداً جديداً للقائمة **ملف**، باسم mnuOpenCDAudio، وعنوان **فتح قرص ليزري موسيقي**.

بعد إضافة البند الجديد، ينبغي أن يصبح البرنامج كما في الشكل 19-6.

|  |  |
| --- | --- |
|  | الشكل 19-6  برنامج الوسائط المتعددة  بعد إضافة البند **فتح قرص**  **ليزري موسيقى** له. |

* اكتب الأسطر التالية في الإجراء mnuOpenCDAudio\_Click():

Private Sub mnuOpenCDAudio\_Click()

'إغلاق الجهاز السابق في حال وجود جهاز مفتوح

mciMyMedia.Command = "Close"

'تحديد نوع الجهاز بأنه جهاز القرص الموسيقي

mciMyMedia.DeviceType = "CDAudio"

'إعطاء أمر الفتح لجهاز القرص الليزري الموسيقي

mciMyMedia.Command = "Open"

hsbPosition.Max = mciMyMedia.Length

'عرض قيمة طول المسار في أداة العنونة

lblLastPositiom.Caption = mciMyMedia.Length

lblCurrentFile.Caption = "فتح القرص الليزري الموسيقي"

End Sub

لا يعتبر القرص الليزري الموسيقي ملفاً، لذلك لا حاجة لوجود مربع حوار اختيار ملف.

ينفذ هذا الإجراء آلياً عند اختيار المستخدم البند **فتح قرص ليزري موسيقي** من قائمة **ملف**.

يُغلق الجهاز السابق أولاً، ثم يُفتح الجهاز "CDAudio" الخاص بالأقراص الليزرية الموسيقية، ‏كما هو واضح من عبارات الإجراء السابق.

ملاحظة

لكي تستطيع تشغيل هذا الجزء من البرنامج، يجب تواجد قرص ليزري موسيقي لديك. وأن يكون مشغل الأقراص الليزرية موصول بسلك مباشر مع بطاقة الصوت. أو يمكنك سماع الأقراص الليزرية الموسيقية، من زر سماعة الرأس الموجود في واجهة مشغل الأقراص الليزرية.

* بعد فتح الجهاز الليزري الموسيقي، يصبح عزف المسار الأول من القرص الليزري الموسيقي ممكناً، للانتقال إلى المسار اللاحق من القرص، انقر الزر NextTrack (هو نفس الزر الذي ينقلنا لنهاية الملف). للانتقال إلى المسار السابق،
* انقر الزر PreviousTrack (هو نفس الزر الذي ينقلنا لبداية الملف).
* انقر الزر **Eject** لفتح علبة مشغل الأقراص الليزرية.

لاحظ أن الأزرار التابعة لعنصر تحكم الوسائط المتعددة، تختلف معانيها تقريباً، حسب نوع الجهاز المفتوح.

* أضف الآن الأسطر التالية للإجراء mciMyMedia\_NextClick():

Private Sub mciMyMedia\_NextClick(Cancel As Integer)

hsbPosition.Max = mciMyMedia.Length

lblLastPositiom.Caption = mciMyMedia.Length

End Sub

ينفذ هذا الإجراء آلياً، عند نقر الزر **NextTrack** (مسار لاحق)، حيث يقوم بإظهار طول المسار اللاحق الذي تم الانتقال إليه.

* أضف الآن الأسطر التالية للإجراء mciMyMedia\_PrevClick():

Private Sub mciMyMedia\_PrevClick(Cancel As Integer)

hsbPosition.Max = mciMyMedia.Length

lblLastPositiom.Caption = mciMyMedia.Length

End Sub

ينفذ هذا الإجراء آلياً، عند نقر الزر PrevTrack (مسار لاحق)، وله نفس وظيفة الإجراء السابق.

## لمحة عن أنواع ملفات الوسائط المتعددة

سنشرح الآن بعض أهم أنواع ملفات الوسائط المتعددة، مع ذكر أهم الفروقات بين هذه الأنواع، وسبب وجود هذه الأنواع أصلاً:

### الملفات الصوتية Wave

تملك هذه الملفات الامتداد (wav)، ويخزن الصوت فيها بشكل رقمي Digital.تستطيع هذه الملفات، تخزين جميع أنواع الأصوات، مثل الموسيقى والغناء والكلام البشري أيضاً. بكلام آخر، يمكن اعتبار ملف الصوت wav، مثل جهاز التسجيل المنزلي Recorder.

يمكن تخزين الصوت في هذه الملفات على عدة تنسيقات، أهمها:

صوت 8 كيلو بايت/الثانية 8 بت أحادي 8000 هرتز

صوت 16 كيلو بايت/الثانية 8 بت استريو 8000 هرتز

صوت 43 كيلو بايت/الثانية 16 بت أحادي 22050 هرتز

صوت 86 كيلو بايت/الثانية 16 بت استريو 22050 هرتز

صوت 86 كيلو بايت/الثانية 16 بت أحادي 22050 هرتز

صوت 172 كيلو بايت/الثانية 16 بت استريو 44100 هرتز

يعتبر تخزين الصوت بتنسيق أحادي 8000 هرتز، هو أدنى جودة ممكنة، وهي مشابهة لجودة صوت سماعة التلفون.

أما تخزين الصوت بتنسيق 22050 هرتز، فهي جودة متوسطة تشابه جودة صوت الراديو.

وتخزين الصوت بتنسيق 44100 هرتز، فهي جودة عالية جداً تشابه جودة صوت القرص الليزري الموسيقي.

يأتي تخزين الصوت بجودة عالية، على حساب حجم الملف، فمثلاً، تخزين خمسة دقائق من الصوت بجودة عالية جداً وصوت ستيريو، يحتاج لحجم مقداره 172 × 5 × 60 = 51600 كيلو بايت، أو 51.6 ميجا بايت.

ولتخزين نفس الملف بجودة متوسطة مثلاً، وصوت أحادي، يحتاج لحجم مقداره 43 × 5 × 60 = 12900 كيلو بايت أو 12.9 ميغا بايت. وهكذا.

إذاً، توصلنا لنتيجة مفادها: كلما زادت جودة الصوت المسجل، كلما زاد حجم الملف الذي نسجل فيه الصوت.

### الملفات الموسيقية MIDI

تصور أنك تريد تخزين الموسيقى أو النوتات الموسيقية في ملف ما، فليس من المعقول أبداً تخزين الموسيقى في ملف Wave، لأنه في هذه الحالة، سيأخذ حجماً كبيراً جداً، لذلك يمكنك تخزين النوتات الموسيقية، مع جميع الأوامر الموسيقية الأخرى، داخل الملف من النوع MID.

يكون حجم الملف MID صغيراً جداً، مقارنة مع أحجام الملفات Wave.

عند عزف الملف الموسيقي MID، يتعرف ويندوز على نوع النوتة الموسيقية التي يجب عزفها، ويرسلها كأمر إلى بطاقة الصوت (أو جهاز الأورج مثلاً)، ليصدر الصوت الخاص بهذه النوتة.

نستطيع تخيل الملف الموسيقي MID، على أنه تسجيل لضربات المفاتيح التي يقوم بها العازف على الأورج.

لا بد أنك توصلت إلى نتيجة أنه لا يمكنك تخزين صوت آخر غير النوتات الموسيقية في الملف MID، مثل صوت الكلام البشري أو الغناء.

### الملفات الحركية AVI

تخزن هذه الملفات صوراً متتابعة، يتم عرضها بشكل متتالي لتشكل فيلماً متحركاً. في الحقيقة فإن هذه الملفات قادرة على تخزين الصوت أيضاً مع الصور المتحركة. وتستطيع تخيل هذه الملفات على أنها جهاز فيديو منزلي.

تأخذ هذه الملفات حجماً كبيراً جداً على وسائط التخزين، يعتمد الحجم على دقة الصور التي تخزنها، بالإضافة لعدد الألوان وجودة الصوت المخزن مع الصورة المتحركة.

هناك تنسيقات مختلفة جداً لهذا الملف منها:

تنسيق 15 صورة بالثانية الواحدة، أو 24 صورة بالثانية الواحدة، أو 30 صورة بالثانية الواحدة.

كلما زادت عدد الصور بالثانية الواحدة، زاد حجم الملف النهائي.

تعرفنا مما سبق على بعض أهم أنواع ملفات الوسائط المتعددة القياسية وكيفية تخزين المعلومات بداخلها. وأنشأت برنامجاً يستطيع عزف هذه الملفات للمستخدم.

الخلاصة

تعلمت في هذا الفصل عن تقنية OCX ActiveX، وكيف تتعامل مع كائنات ActiveX المختلفة. وتعلمت أيضاً قواعد استخدام هذه العناصر، وكيفية إضافة أحد هذه العناصر إلى مشروعك، لتتمكن من استخدامها.

تعلمت أيضاً كيفية بناء برنامج الوسائط المتعددة، الذي يتيح لك إمكانية فتح الملفات القياسية للوسائط المتعددة، مثل ملفات الصوت Wave، والموسيقى MIDI، والحركة AVI، وأيضاً تشغيل القرص الليزري الموسيقى.

أخيراً تعلمت عن الملفات القياسية للوسائط المتعددة بشكل أعمق وكيفيةة تخزين المعلومات بداخلها وتنسيقاتها المختلفة وأحجامها التقريبية.