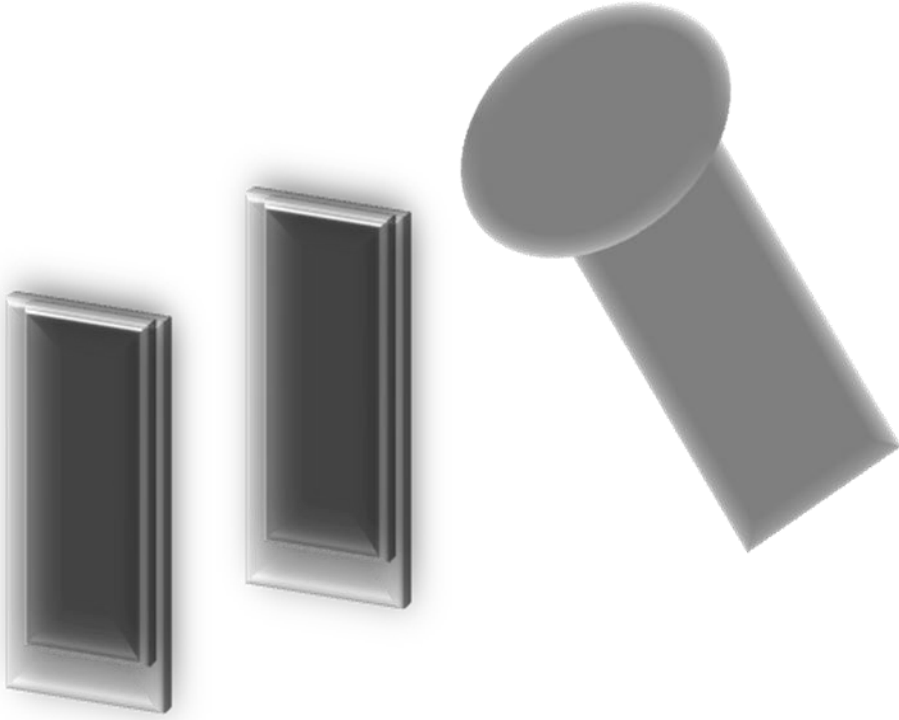


بسم الله الرحمن

شرح مبسط لمكتبة Speech SDK Library



Speech SDK Library

هذا الكتاب مسموح بنشره وتوزيعه في سبيل الله وحقوق الطبع غير محفوظة
لشخص معين لذا يسمح لك نشر الكتاب سواء بالصورة الورقية أو الالكترونية بشرط
الدعوة للمؤلف بدعوة صالحة والله ولي التوفيق

بسم الله الرحمن الرحيم

إن الهدف الرئيس من هذا الكتاب هو شرح مبسط لطريقة استخدام مكتبة Microsoft Speech SDK المتخصصة في عمليات فهم الكلام المحكي إلى المايكروفون ومن ثم تخزينه على شكل نص في البرنامج

أو تقوم المكتبة بتحويل النصوص إلى كلام Text To Speech

ولقد قمت بتأليف هذا الكتاب الصغير لما عانيت به عندما بحثت عن كتاب باللغة العربية يشرح هذه المكتبة ولكن هباءً لذا قررت كتابة هذا الكتاب ليكون شرحاً مبسطاً عن المكتبة ولنمسه في تطوير علم فهم الكلام لأنه لا يزال علم جديد وفرصتنا في تطويره كبيرة

أما عن ما تحتاج لمعرفته من لغات البرمجة

فيجب أن تكون تعرف بعض المعلومات عن لغة الفيجوال بيسك دوت نت

VB.NET

و القليل عن البرمجة الشيئية الموجهة object oriented programming

ويجب أن تملك أحد إصدارات مكتبة Microsoft Speech SDK وسأستخدم أنا إصدار Microsoft Speech SDK 5.1 وهو موجود على موقع مايكروسوفت للتحميل بالمجان

أما عن لغة البرمجة فسوف أستخدم VB.NET 2005

وأرجو أن ينال هذا الكتاب إعجابكم وأرجو عدم اللوم في حال وجدت أي أخطاء فالكمال لله

والسلام عليكم ورحمة الله وبركاته

الفهرس

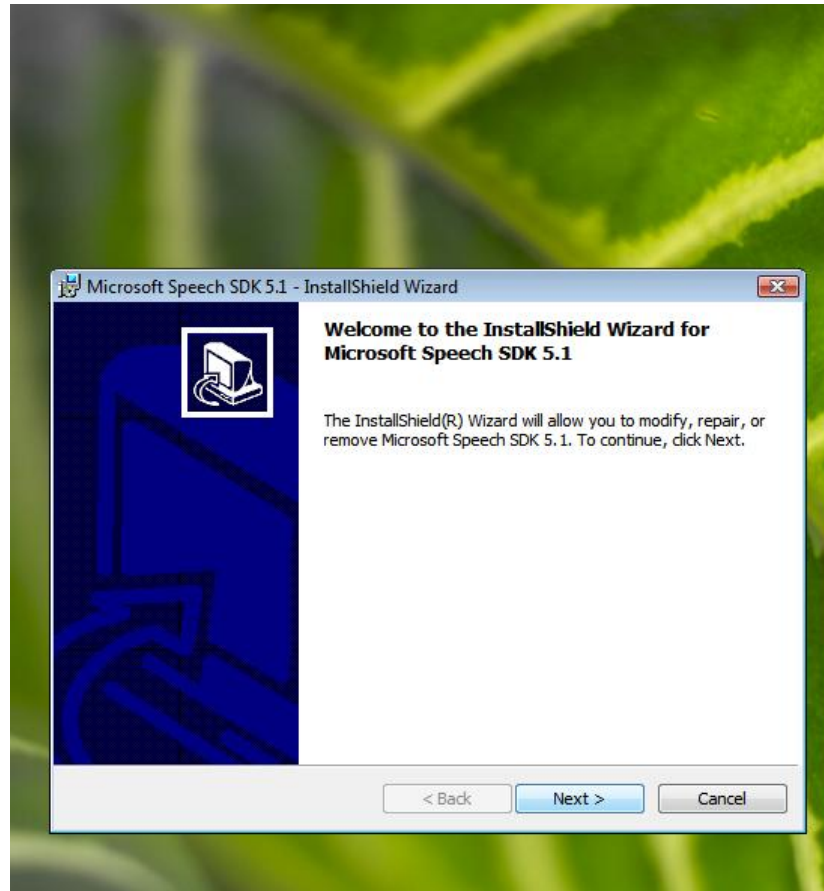
- ١ طريقة تنزيل المكتبة
- ٢ طريقة إضافة المكتبة إلى المشروع
- ٣ شرح بعض الأساسيات
- ٤ شرح طريقة عمل فهم الأوامر المستقبلية صوتيا
- ٥ شرح مكتبة الإملاء
- ٦ شرح طريقة تحويل الكتابة الى كلام
- ٧ خصائص المكتبة في لوحة التحكم
- ٨ إعداد المايكروفون
- ٩ جلسات تدريب الحاسوب
- ١٠ - بعض التطبيقات من المكتبة
- ١١ - برنامج يقوم بنطق الساعة
- ١٢ - برنامج يقوم بتسجيل الدخول لمستخدم عن طريق نطق كلمة السر
- ١٣ - برنامج يقوم برد التحية
- ١٤ - برنامج محادثة صغير مع الحاسوب
- ١٥ - آلة حاسبة باستخدام الأوامر الصوتية
- ١٦ - برنامج املاء يقوم بحفظ النصوص المحكية الى ملف نصي

- طريقة تنزيل المكتبة

أولاً قم بتحميل المكتبة من موقع مايكروسوفت من خلال هذا الرابط

<http://www.microsoft.com/downloads/details.aspx?FamilyID=5e86ec97-40a7-453f-b0ee-6583171b4530&displaylang=en>

ثم بعد ذلك قم بتنصيب المكتبة كأى برنامج عادي (next >next)

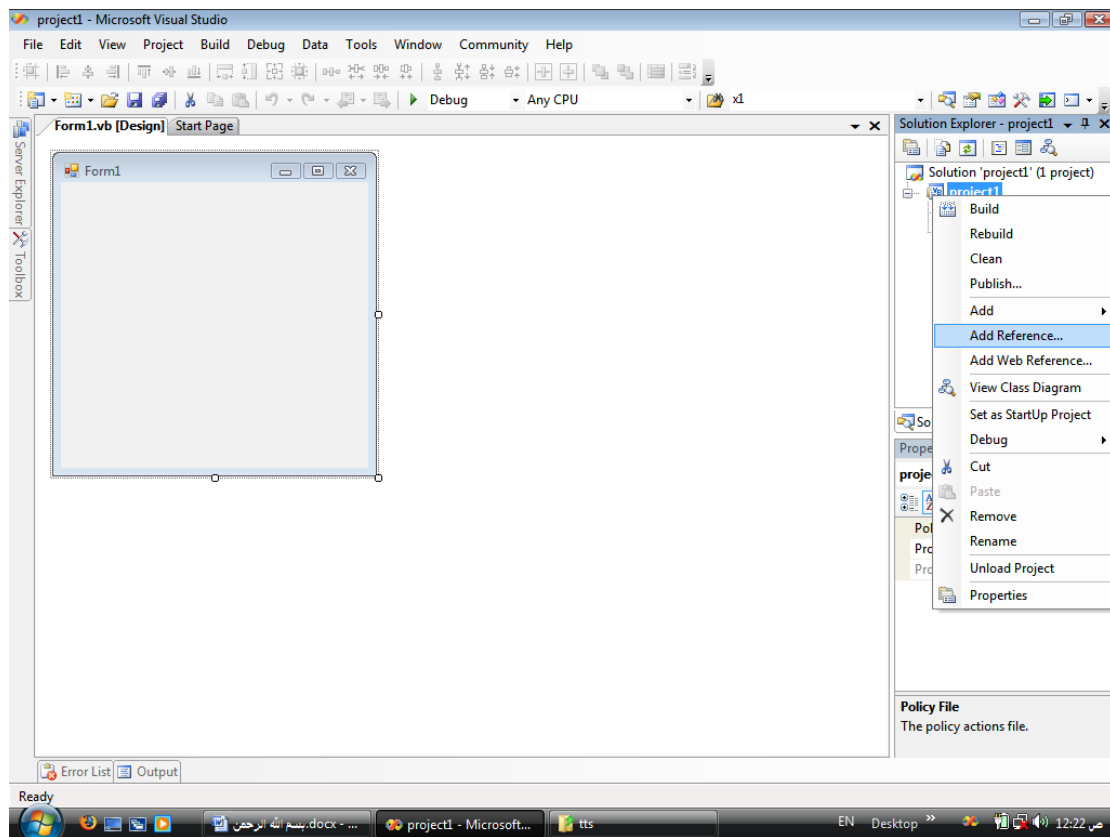


ومن ثم تكون المكتبة قد جهزت للاستخدام من قبل المستخدم

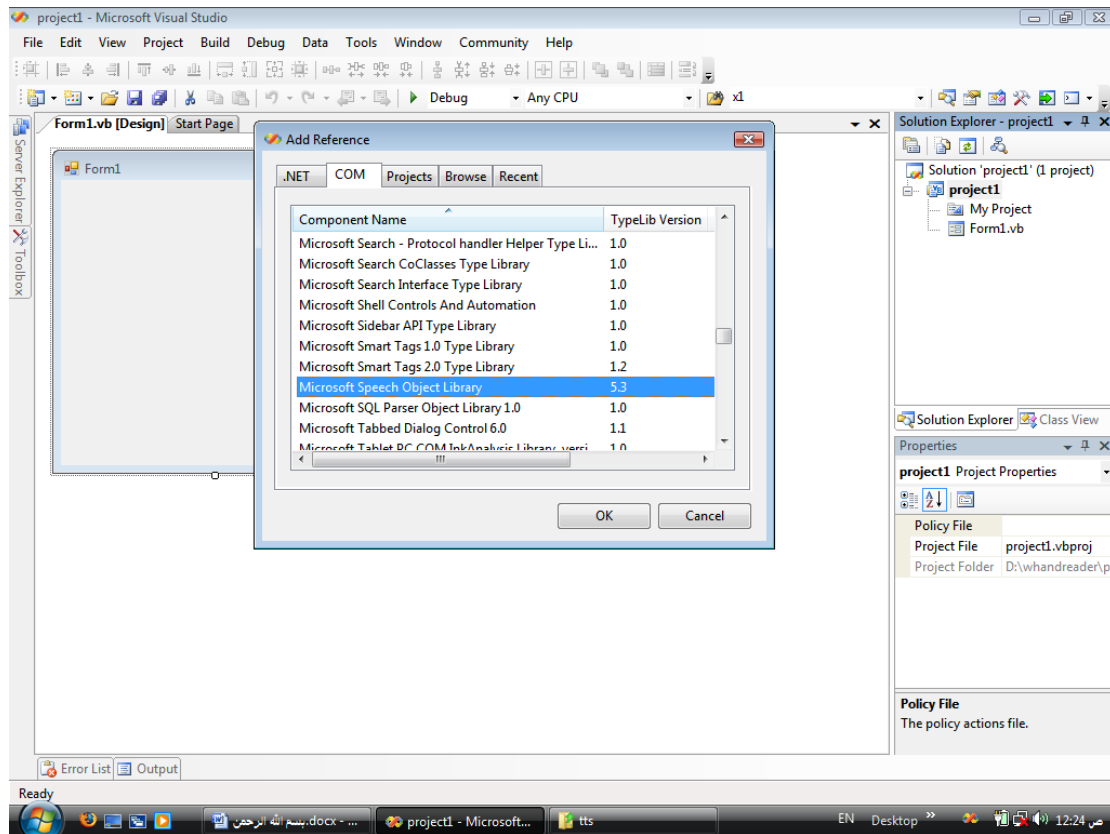
ولابد من تنزيل هذه الملفات على الجهاز الهدف ليتمكن من تشغيل البرنامج الناتج بعد تنزيل هذه المكتبة تستطيع الآن الحصول على الأوامر الموجودة في المكتبة واستخدامها في برنامجك وأرجو المتعة في قراءة هذا الكتاب

- طريقة اضافة المكتبة الى المشروع

أولاً قم بإنشاء مشروع جديد باستخدام Microsoft Visual Studio 2005 وبإمكانك استخدام أي لغة برمجة تتقنها ولك سوف انا سوف أستخدم لغة الفيجوال بيسك نظراً لسهولة استخدامها ويسر لها كلغة برمجة بعد إنشاء المشروع اذهب الى مسكشف الحلول solution explorer واضغط على أكونة المشروع ثم بالزر اليمين قم باختيار add reference



يظهر لك مربع حوار اختر تبويب COM ومن ثم اختر Microsoft Speech object library من القائمة



ثم اضغط موافق

وبهذا المشروع الآن جاهز لاستقبال الأوامر والدوال من المكتبة

- بعض الأساسيات

قبل الشروع في البرمجة لابد من تذكير بعض الأساسيات كتذكير للبرمجة الشيئية
object oriented programming وشرح مبدأ الترجمة وأنواعها قبل البدء
باستخدام المكتبة

- البرمجة الموجهة

يقصد بالبرمجة الموجهة لأن تجعل كل شيء في البرنامج عبارة عن كائن له
خصائص كأن تقوم بتغيير اسمه أو طوله أو حجمه
ويحتوي الكائن أيضا على أحداث Events كأن تأمر البرنامج القيام بعمل ما
عند استدعاء العنصر أو تحريك المؤشر عليه وتسمى كل هذه الأمور أحداث
وتحتوي البرمجة الموجهة على الطرق Method وهي تمثل الدوال
والإجراءات التي ترتبط في الكائن الأساسي كمثال في إظهار النافذة form
في فيجوال بيسك نستخدم الأمر form.show() هذا عبارة عن إجراء وهو
من ضمن methods لل Form

هذا بشكل عام تذكير لما هو OOP ولكن ما سنحتاجه هنا هو طرق استدعاء هذه الخصائص والأحداث والطرق من مكتبة Speech SDK وسنذكر بشكل عام كيف يتم ذلك خلال مرور هذه علينا وسوف نعمل مع المكتبة عن طريق انشاء كائن للمكتبة ونقوم باستدعاء الاجراءات اللازمة لتحضير المكتبة ومن ثم تحضير الحدث الذي سيمكننا من استقبال الكلام الملتقط من المايكروفون على شكل كتابة مخزنة في متغير نصي

ومن ثم تستطيع استخدامه في تطبيقك كوحدة ادخال وسنقوم بانشاء كائن يسمح لك باستخدام مكتبة تحويل النصوص الى كلام وسوف أقوم بعد ذلك بوضع بعض الأمثلة على البرامج التي يمكن عملها باستخدام هذه المكتبة ويمكنك استخدام أي لغة برمجة تدعم OOP لتتمكن من استخدام المكتبة أما أنا فسوف أستخدم الفيجوال بيسك 2005 وذلك لأن معظم القراء يعرفون استخدامه كونه لغة سهلة وبسيطة

- شرح طريقة استخدام المكتبة لفهم كلمة من عدة كلمات مختارة صوتيا

وأقصد هنا أن تقوم بتعريف البرنامج على عدة كلمات حيث يقوم البرنامج بمعرفة الكلمة التي نطقها المستخدم ومن ثم تخزينها في متغير نصي

STRING

ويكون ذلك عن طريق انشاء كائن من نوع SpSharedRecoContext

ومن ثم انشاء كائن ليشكل القاعدة التي سوف يستخدمها البرنامج

ومن ثم انشاء كائن لقوم بتحديد شروط المعرفة

ومن ثم كائن لتحديد خصائص القاعدة

ومن ثم اضافة الكلمات المرادة

ومن ثم تحديد طريقة النطق

وبالنهاية نقوم بتعريف حدث عند الحصول على الكلمة ليقوم بتخزينها في متغير نصي

لنصغ هذا الكلام برمجيا

كود

```
Imports SpeechLib 'استيراد الكودات من المكتبة
Public WithEvents objrec As SpSharedRecoContext 'الكائن الرئيس للمكتبة ويحتوي الأحداث
'والمكتبات الرئيسة التي سوف نستخدمها في البرنامج
Public grammer As ISpeechRecoGrammar 'الكائن الذي سوف نستخدمه لتحديد نوع الادخال والقاعدة
Dim rule As ISpeechGrammarRule 'الكائن الذي سنستخدمه في تحديد الشروط المنطبقة على
'القاعدة وإضافة الكلمات
Dim propvalue As Object = "" 'إنشاء كائن لتحديد خصائص الكائن
Public Sub SPReco_Create()
    objrec = New SpSharedRecoContext() 'انشاء كائن جديد من نوع الكائن الرئيس لتتمكن كم استخدامه
    grammer = objrec.CreateGrammar(0) 'انشاء قاعدة استخدام جديدة
    rule = grammer.Rules.Add("MenuCommand", SpeechRuleAttributes.SRADynamic Or
SpeechRuleAttributes.SRATopLevel, 0) 'تحديد نوع الادخال أنه من نوع فهم عدد من الكلمات
'المحددة
    rule.InitialState.AddWordTransition(Nothing, "new", "",
SpeechGrammarWordType.SGLexical, wrd, 0, propvalue, 1.0) 'اضافة كلمة new للبرنامج
    rule.InitialState.AddWordTransition(Nothing, "old", "",
SpeechGrammarWordType.SGLexical, wrd, 0, propvalue, 1.0) 'اضافة كلمة old للبرنامج
    rule.InitialState.AddWordTransition(Nothing, "help", "",
SpeechGrammarWordType.SGLexical, wrd, 0, propvalue, 1.0) 'اضافة كلمة help للبرنامج
```



```

    grammar.Rules.Commit() ' اعطاء الأمر بالاستعداد للترجمة
    grammar.CmdSetRuleState("MenuCommand", SpeechRuleState.SGDSActive) '
    اعطاء الأمر للمترجم ببدء عملية الاستماع للمايكروفون والتعرف على الكلام
End Sub
الان سوف نقوم بإنشاء اجراء يتم تنفيذه عند حدث قيام المايكروفون بالتقاط صوت وقيام المكتبة '
    بفهمه
Public Sub Reco_Event(ByVal streamnumber As Integer, ByVal streamposition As
Object, ByVal Recognitiontype As SpeechRecognitionType, ByVal Result As
ISpeechRecoResult) Handles objrec.Recognition
    هنا تضيف الأوامر التي تريد أن يكون الكلام مخزنا هنا في خاصية لمتغير قمنا بتسميته ' Result
    تنفذ عندما تقوم المكتبة بفهم الكلام من المايكروفون
    حيث يكون الكلام مخزنا فيه وتمن الحصول عليه من خلا استدعاء الأمر في الكود التالي
    Dim str As String ' قمنا بإنشاء متغير نصي
    Str= Result.PhraseInfo.GetText(0, -1, True) ' هذا الأمر يستخدم لجعل المتغير الموجود يسار الكود
    قيمته ما التقط في المايكروفون
    والان تستطيع أن تفعل ما تشاء بالمتغير الناتج فعلى سبيل المثال في مثالنا سنقوم باظهار النتيجة على شكل
    رسالة تظهر للكمبيوتر المحكية
    MsgBox(str)

End Sub
الان بقي عليك استدعاء الاجراء الذي قمنا بإنشائه وسنقوم بذلك عن طريق اجراء استدعاء النافذة '
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        هنا قمنا باستدعاء الاجراء ' SPReco_Create()

    End Sub
    والان لايقاف عملية الاستماع نقوم بإنشاء اجراء يحتوي على امر ايقاف عملية الاستماع
Public Sub stoprec()
    امر ' grammar.CmdSetRuleState("MenuCommand", SpeechRuleState.SGDSInactive)
    ايقاف عملية التسجيل
End Sub

```

والان وبهذا الكود تكون قد أنشأت برنامج يقوم بالتسجيل من المايكروفون والتعرف على الكلام ومن ثم مقارنته بالكلمات الموجودة ثم يقوم بخزن الكلام في متغير Ruslt ومن ثم عرضه على الشاشة وهذا البرنامج فقط يسمع الكلمات المضافه له وهي (New، Old، help) وتستطيع اضافة الكلمات التي تريد بتكرار الامر الذي أضفنا به هذه الكلمات مع تغير الكلمة فيه

- شرح طريقة عمل الاملاء

شرحنا في الدرس السابق طريقة عمل فهم لعدد معين من الكلمات وهذه الطريقة مفيدة جدا اذا اردت عمل برنامج يقوم بتنفيذ أوامر محددة كأن يدخل كلمة سر بنطقها ككتابتها أو الوصول الى امر في قائمة محددة بنطقها بدلا من استخدام الفأرة

ولكن ماذا اذا أردت عمل برنامج لترجمة النصوص المحكية من لغة لأخرى أو برنامج ليقوم بكتابة الرسالة المحكية الى نص ومن ثم ارسالها الى بريد الكتروني معين في برامج مثل ذلك هل ستستطيع اضافة جميع كلمات اللغة ومن ناحية أخرى ماذا سيكون حجم البرنامج الناتج وعدد الأسطر فيه من هذا المنطلق وفرت المكتبة ميزة الإملاء بحيث تكون الكلمات المفهومة للبرنامج غير محددة وتكون معرفة في النظام والكود التالي يوضح طريقة عمل الاملاء وهو كود أسهل بكثير من الكود السابق

كود

```
Imports SpeechLib 'استيراد الكودات من المكتبة '
الكائن ' Public WithEvents objrec As New SpSharedRecoContext
الرئيس للمكتبة ويحتوي الأحداث والمكتبات الرئيسة التي سوف نستخدمها في البرنامج
الكائن الذي سوف نستخدمه ' Public grammar As ISpeechRecoGrammar
لتحديد نوع الادخال والقاعدة
Public Sub Dic()
    تحديد نوع القاعدة املاء ' grammar=objrec.CreateGrammar(1)
    GRAMMER.DictationLoad(
    أمر تجهيز الاملاء ' SpeechLib.SpeechLoadOption.SLOStatic)
    GRAMMER.DictationSetState(SpeechLib.SpeechRuleState.SGDS
    Active) 'أمر البدء بالاملاء
End Sub
والان ما علينا القيام به هو تحديد اجراء يقوم البرنامج باستدعائه عند حدوث '
الاملاء
Private Sub RecoContext_Reognition(ByVal StreamNumber
As Integer, ByVal StreamPosition As Object, ByVal
RecognitionType As SpeechLib.SpeechRecognitionType, ByVal
Result As SpeechLib.ISpeechRecoResult) Handles
OBJREC.Recognition
    هنا ويكون الكلام مخزنا هنا في خاصية لمتغير قمنا بتسميته ' Result
```

```

تضيف الأوامر التي تريد أن تنفذ عندما تقوم المكتبة بفهم الكلام من
المايكروفون
حيث يكون الكلام مخزنا فيه وتمن الحصول عليه من خلا استدعاء الأمر في '
الكود التالي
Dim str As String ' قمنا بانشاء متغير نصي
Str= Result.PhraseInfo.GetText(0, -1, True)' هذا الأمر يستخدم لجعل
المتغير الموجود يسار الكود قيمته ما التقط في المايكروفون
والان الكود الذي يظهر الترجمة على رسالة على الشاشة
MsgBox(str)
End Sub
والان لايقاف الاملاء نستخدم الأمر التالي'
Public Sub stopdic()
GRAMMER.DictationSetState(SpeechRuleState.SGDSInactive)'
أمر ايقاف الاملاء
End sub
وحتى تستطيع البدء بالاملاء قم باستدعاء الاجراء Dic()'

```

وبهذا البرنامج قمنا بانشاء برنامج يقوم بتحويل الكلام المحكي على شكل
املاء ومن ثم ارساله الى البرنامج
وبالنهاية يقوم البرنامج بعرضة على الشاشة
وبشرح هذا البرنامج نكون قد انتهينا من شرح مبسط لما تحتاجه من طرق
الادخال التي تلزمك

- شرح طريقة تحويل النصوص الى كلام

قد شرحنا في الدروس السابقة من المكتبة الجزء الذي يتعلق بوحدة الادخال
المايكروفون وقد يأل البعض هل يمكن للمكتبة القيام بالعملية العكسية وتحويل الكلام
الى نصوص

و الجواب نعم وهذا ما سيشرح في هذا الدرس وطريقة هذا الدرس سهلة على خلاف
الدرسين السابقين حيث كل ما ستحتاجه هنا هو انشاء كائن لمكتبة الصوت Voice
ومن ثم استدعاء دالة الكلام

كود

```
Public Sub talk()
```

```
Dim SP As New SpeechLib.SpVoice 'انشاء كائن الصوت
```

```
sp.Speak("hello") 'الأمر بنطق الكلمة المطلوبة
```

```
End sub
```

بعد ذلك سوف تحتاج للقيام باستدعاء الأمر الذي أنشأناه Talk وبذلك تحصل على الكلام من السماع

وبهذا نكون قد أنهينا شرح طرق الادخال والاخراج في المكتبة

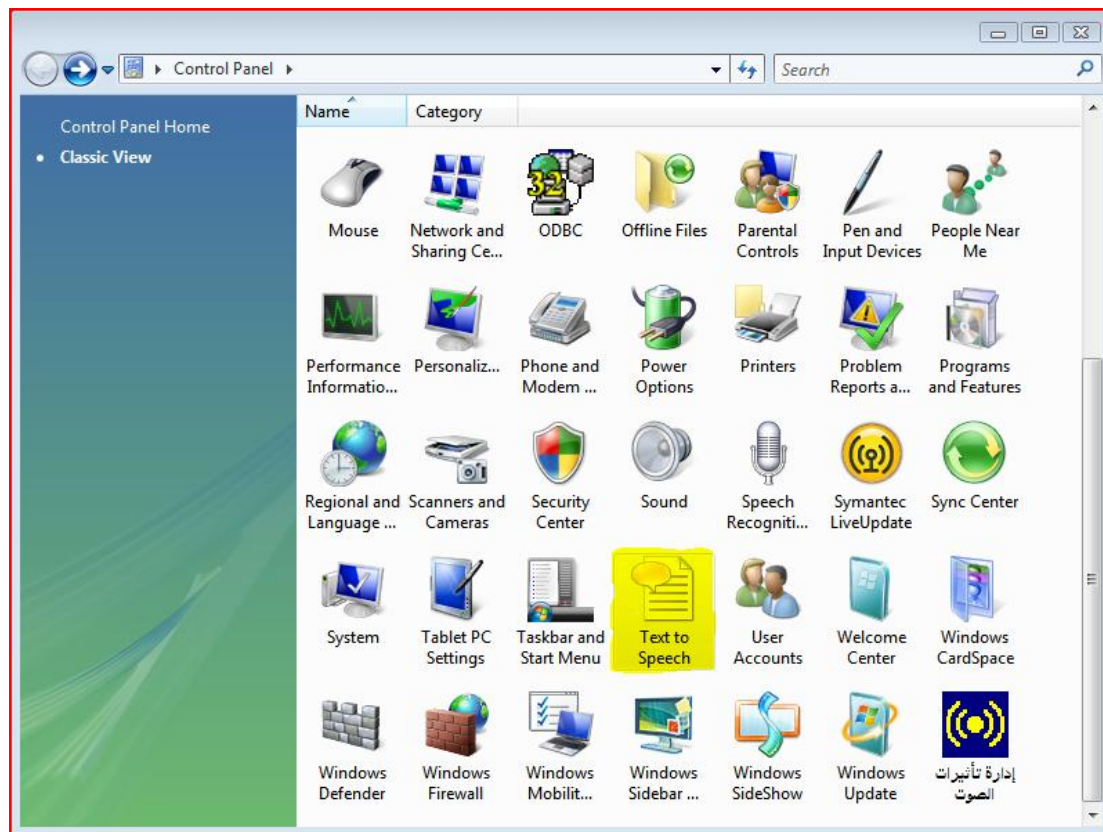
وفي الدروس الباقية سنقوم أولاً بتعلم خصائص المكتبة في لوحة التحكم ومن ثم بعض التطبيقات التي سوف تفيدك عزيزي القارئ من كل ما تعلمناه وذلك بإنشاء بعض البرامج التي يمكن استخدامها في هذه المكتبة

- خصائص المكتبة في لوحة التحكم

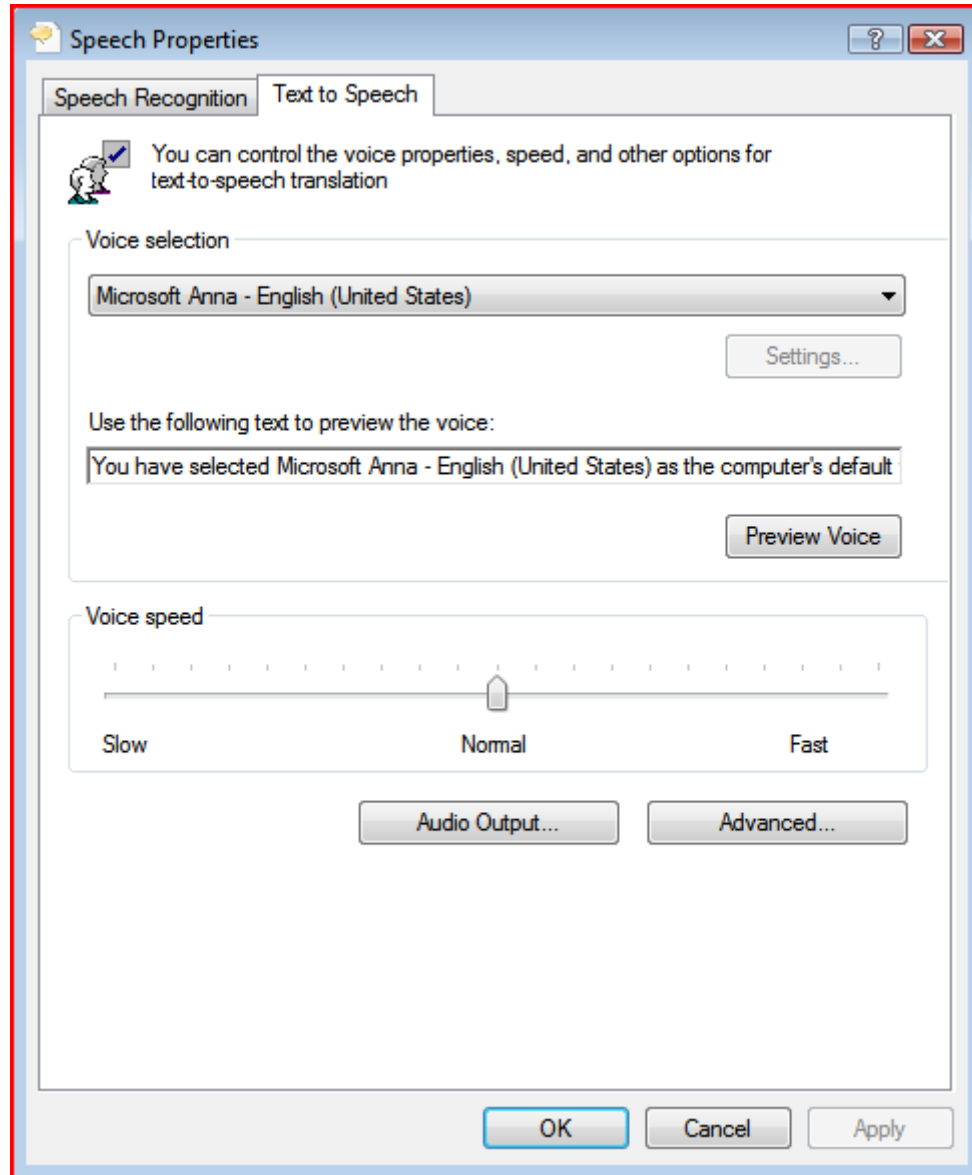
قمنا سابقاً بشرح طرق الادخال والاخراج في مكتبة Microsoft Speech SDK بشكل عام

ولكن للحصول على نتائج أفضل سواء في عملية فهم الكلام أو تحويل النصوص الى كلام وجد لهذا المكتبة نصيب في لوحة التحكم ليتمكن المستخدم من تحديد الخصائص التي يريد

للوصول الى الخصائص قم باختيار أيقونة Speech الموجودة في لوحة التحكم



تظهر لك النافذة التالية

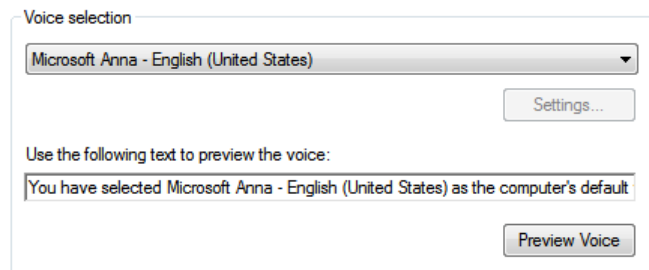


تتكون نافذة الخصائص المكتبة م تبويين الأول تبويي التعرف على الصوت speech recognition والثاني Text To speech

وسأبدأ أولاً بتبويب تحويل النصوص الى كلام Text To speech لسهولة

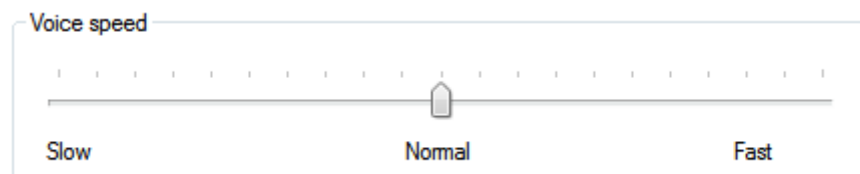
يتكون جزء خيارات التحويل من النص الى كتابة على الخيارات التالية

١ صوت المتكلم : وهو يمكنك من اختيار شكل الصوت الذي سيقوم بقراءة الكلام الذي ترسله الى المكتبة



حيث يمكنك هذه الخاصية من اختيار ان كنت تريد أن يُقرأ بصوت رجل أو امرأة و يوجد مربع نصي يمكنك من تجربة الصوت المختار

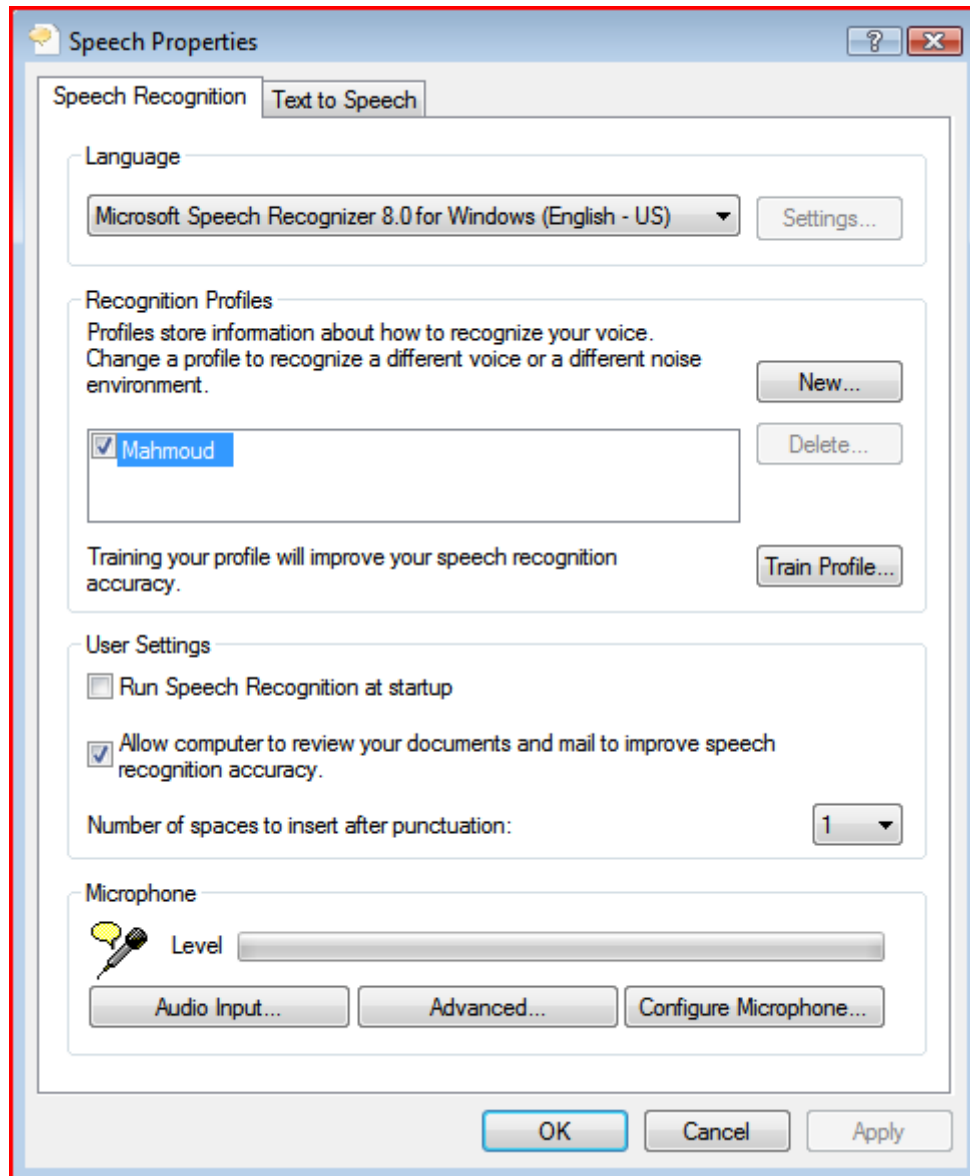
٢ - خاصية تحديد سرعة الكلام وتستخدم لتقوم بتسريع القراءة أو تبطيئها حسب ما يناسبك



٣ - وأما الزر Advance و Output device فيستخدمان لتحديد الأجهزة التي يتم اخراج الكلام بواسطتها

وبهذا نكون قد انهينا تبويب Text To speech

لنبدأ بشرح تبويب التعرف على الصوت speech recognition



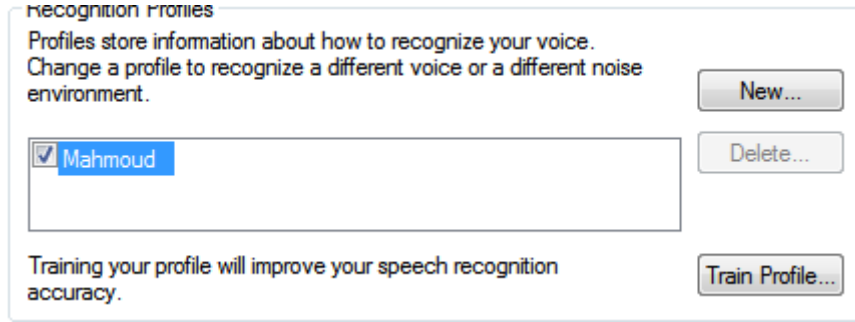
يتكون هذا التبويب من عدة خيارات

١ خيار تغير معرف الكلام speech recognizer والهدف من هذه الخاصية تغير متعرف الكلام حسب الخبرات المكتسبة واللهجة التي يستخدمها المستخدم في الوضع الافتراضي

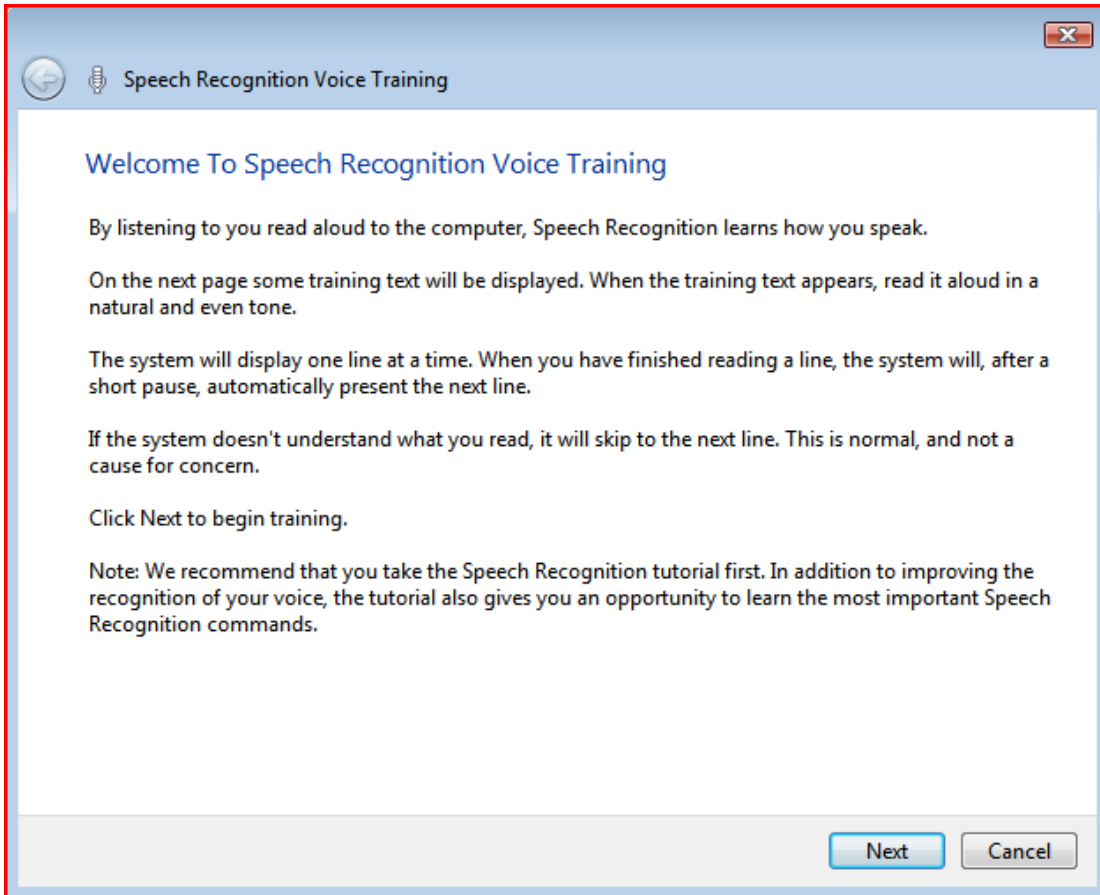
Microsoft Speech Recognizer 8.0 for Windows (English - US) Settings...

وأما بقية الخيارات فتختلف حسب نظام التشغيل المستخدم و لمزيد من المعلومات تستطيع الرجوع الى برنامج المساعدة التابع لنظام التشغيل لكن ما يهمنا في هذا التبويب خياران الأول recognition profile - عملية تدريب الحاسوب

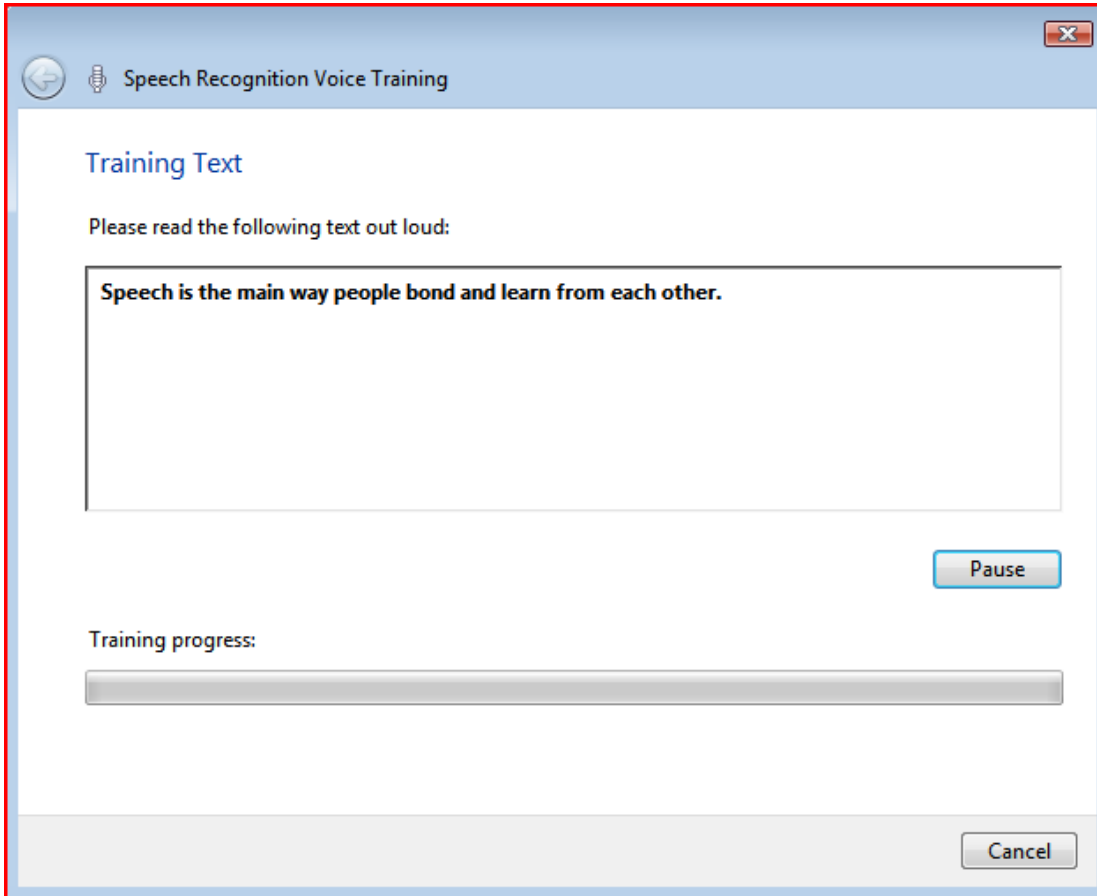
وهو يحتوي على الخبرات التي حصل عليها الحاسوب في عملية فهم صوت المتكلم صاحب الصوت أي أصبحت الدقة في التعرف على كلام صاحب البروفایل عالية بعد استخدام المكتبة أكثر من مرة وعند تنصيب المكتبة لأول مرة يفضل تدريب الحاسوب على لهجة المستخدم وذلك من خلال أمر تدريب train profile



مما يزيد الدقة في فهم صوت المستخدم الجديد بعد الضغط على train profile يظهر المعالج



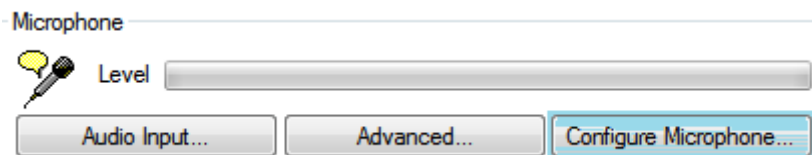
قم بالضغط على Next ثم ابدأ بقراءة النصوص التي تظهر على الشاشة



عند نهاية المعالج يكون الحاسوب قد اكتسب الخبرة في طرق فهم لهجتك وطريقة نطقك للحروف مما يزيد من دقة الترجمة - طريقة اعداد المايكروفون

وأما الامر الثاني الذي يهمننا من تبويب speech recognition فهو أمر اعداد المايكروفون microphone configure حيث يستخدم لاعداد وضع المايكروفون في البيئة المحيطة وذلك لمحاولة التأقلم مع الضجيج ومعرفة نبرة صوت المتحدث وشدة صوته ليتمكن المترجم من التعرف على الكلام في البيئة الجديدة للمايكروفون

وللوصول الى هذا الزر اختر الزر configure microphone



ومن ثم يظهر المعالج

ويختلف طريقة التعامل مع المعالج من نظام لآخر لكن يتم الاتفاق على المبدأ حيث
لأبد للمستخدم في النعاية من نطق مجموعة من الكلمات حيث يقوم النظام بتقليل
الصوت في النظام الى حد يصبح به تقريبا يستمع فقط صوت المتكلم
وبهذا نكون قد شرحنا خصائص المكتبة في لوحة التحكم

تطبيقات من استخدام المكتبة

عزيزي المستخدم بمعرفة هذه المعلومات أنت الآن تتقن استخدام مكتبة Speech SDK ولكن قد يخطر في بالك سؤال ما فائدة كل ما تعلمته وكيف سوف أستفيد منه في برامجي مستقبلا ؟ لذلك ستكون باقي الدروس هي عبارة عن برامج تطبيقية على ما قمنا بتعلمه خلال الدروس السابقة

هناك بعض الملاحظات التي سوف نذكرها:

- ستكون معظم برامجنا التي سنكتبها من النوع console application والسبب هو فقط اختصارا للوقت وبالتأكيد إن الإجراءات التي قمنا بها سابقا لا تقتصر على console Application بل انها قد تكون أفضل في windows application ولا فرق باستخدامها هنا وهناك فالمبدأ واحد ولكن الفرق هو أن الاجراءات التي نكتبها في برامجنا ستقوم بكتابتها بالحدث المناسب كحدث استدعاء النافذة form_load
- سأقوم بعرض خوارزمية البرنامج ومن ثم الكود وسأشرح فقط الكودات الجديدة ومبدأ العمل
- ستكون الطرق التي تعلمناها سابقا اما وحدات إدخال أو إخراج
- اختصارا لوقتك وجهدك سوف نتفق على تعريف تسمية ثابتة لبعض الكائنات المهمة والإجراءات الضرورية
- وسوف نقوم بإضافتها إلى وحدة نمطية module بحيث تصبح معمة في اللغة وسوف يكفيك مستقبلا لاستخدام هذه الكائنات والإجراءات فقط إضافة الوحدة الى مشروعك
- لنتفق على محتويات الوحدة قبل البدء في تصميمها
- سنعتبر الكائن objrec كناية عن الكلاس SpSharedRecoContext
- سنعتبر grammer كناية عن العنصر ISpeechRecoGrammar
- سنعتبر الاجراء sprec_cr() هو اجراء عمل عملية التعرف على الكلمات وينتهي عند الاجراء الذي يسبق اجراءات اضافة الكلمات
- سنعتبر addwrd(wrd) اجراء اضافة الكلمات حيث wrd هي الكلمة التي تريد اضافتها
- سنعتبر sprec_bg() كناية عن بدأ عملية الاستماع

سنعتبر `sprec_st()` كناية عن ايقاف عملية الاستماع

سنعبر عن `Sp` كناية عن الكائن `SpeechLib.SpVoice`

سنعتبر `dic()` كناية عن إجراءات الاملاء

سنعتبر `stopdic()` كناية عن اجراء توقف الاملاء

وفي هذه الوحدة سنضيف ما يلزم من متغيرات

اذا ستحتوي هذه الوحدة كل الأوامر التي تعلمناها مع اختلاف التنظيم حيث كما علمت سنقوم بتقسيم الاجراءات حيث يزيد من مرونة الكود وعدم جمودة مع ما سنحتاجه وهذا ما ستلاحظه عند العمل و بالوجه المقابل ستري كم سيكون العمل أسهل

ولنبدأ بصياغة الكود

```
Imports SpeechLib
Public Module Spmdle
Public WithEvents objrec As New SpSharedRecoContext
Public grammer As ISpeechRecoGrammar
Public SP As New SpVoice
Dim rule As ISpeechGrammarRule
Dim propvalue As Object = ""
Public Sub SPRec_cr()
    objrec = New SpSharedRecoContext()
    grammer = objrec.CreateGrammar(0)
    rule = grammer.Rules.Add("MenuCommand",
    SpeechRuleAttributes.SRADynamic Or SpeechRuleAttributes.SRATopLevel,
    0)
End Sub
Public Sub addwrd(ByVal wrd As String)
    rule.InitialState.AddWordTransition(Nothing, wrd, " ",
    SpeechGrammarWordType.SGLexical, wrd, 0, propvalue, 1.0)
End Sub

Public sub SPRec_BG()
    grammer.Rules.Commit()
```

```

    grammar.CmdSetRuleState("MenuCommand",
SpeechRuleState.SGDSActive)
End Sub

Public Sub SpRec_st()

grammar.CmdSetRuleState("MenuCommand",SpeechRuleState.SGDSInactive)
End Sub

Public Sub Dic()
grammar=objrec.CreateGrammar(1)
GRAMMER.DictationLoad(, SpeechLib.SpeechLoadOption.SLOStatic)
GRAMMER.DictationSetState(SpeechLib.SpeechRuleState.SGDSActive)
End Sub
Public Sub stopdic()
GRAMMER.DictationSetState(SpeechRuleState.SGDSInactive)
End sub
    Public Sub Reco_Event(ByVal streamnmber As Integer, ByVal
streamposition As Object, ByVal Recognitiontype As SpeechRecognitionType,
ByVal Result As ISpeechRecoResult) Handles objrec.Recognition

    End Sub
End Module

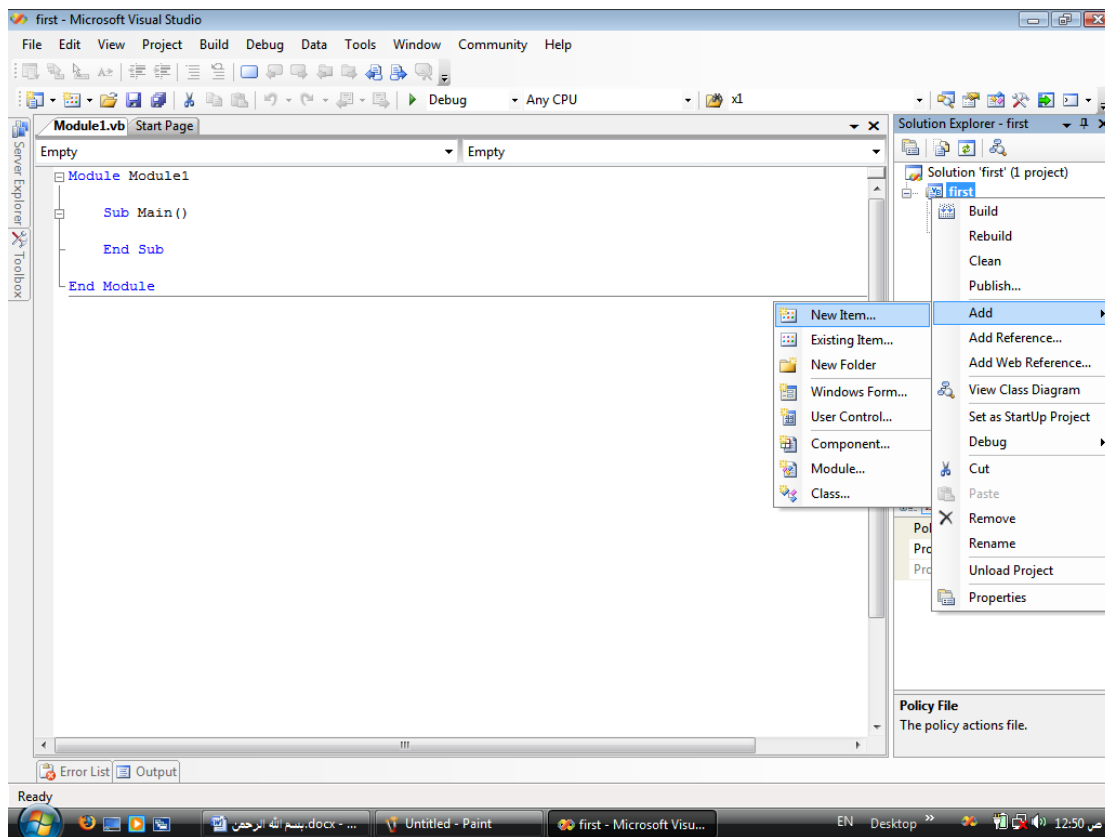
```

لكن لا تنس أن تضيف الإجراءات والدوال التي تحتاج في بلوك الحدث event
 للتعرف على الصوت وقد تركته فارغا هنا لأن هذه الوحدة ستستخدم مرارا وتكرارا
 لذا أسهل إبقاء مكان الكود فارغ حاليا حتى نحتاج ملئه

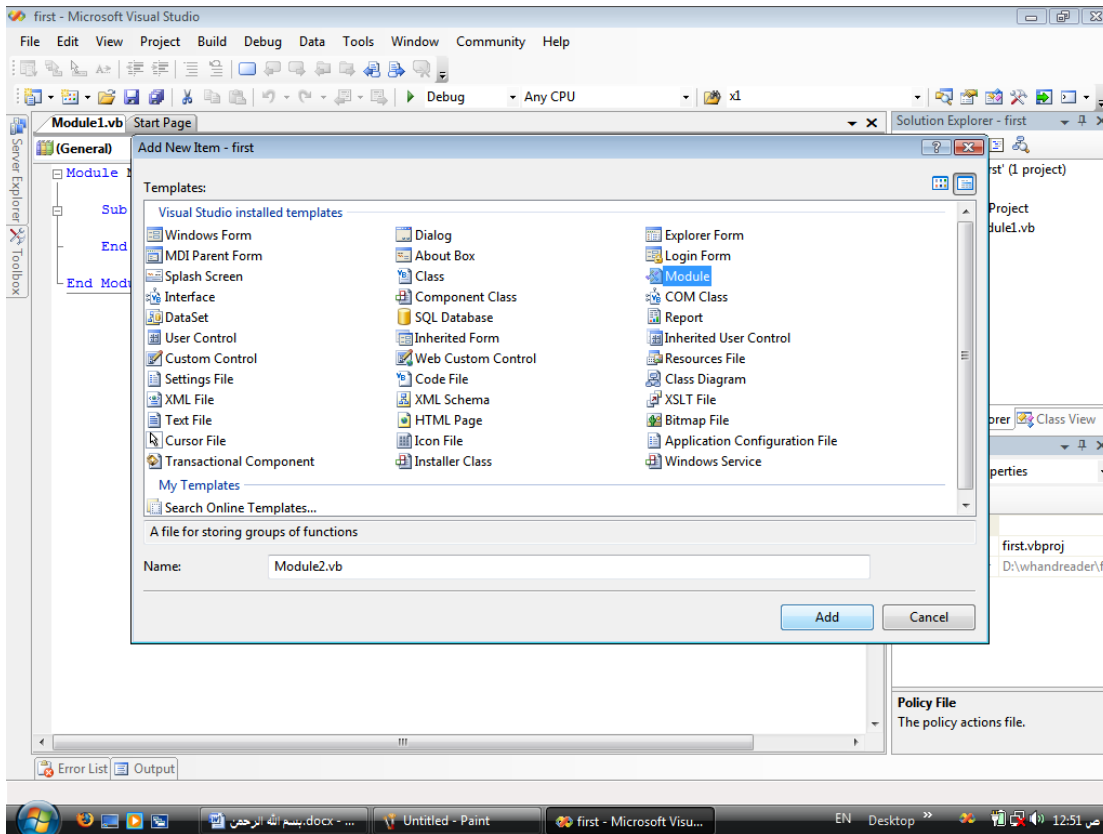
لإضافة الوحدة النقطية الى مشروعك

١ - قم بالضغط بالزر الأيمن من الفأرة على ايقونة المشروع في مستكشف
 الحلول

٢ - اختر اضافة عنصر new → add item



٣ - اختر module من القائمة ثم Add



إذا أردت إعادة استخدام الوحدة في مشروع آخر فما عليك الا اختيار من قائمة Add → exiting item
ومن ثم اخيار الملف المطلوب واضغط add
وبذلك تحصل على الوحدة النقطية المطلوبة
يكفي شرحا ولنبدأ بالتطبيقات
التطبيق الأول: برنامج يقوم بنطق الساعة
من أجمل وأسهل التطبيقات على المكتبة حيث تكون فكرة هذا البرنامج على
الحصول على الوقت ومن ثم نطقه باستخدام أمر تحويل النص الى كلام
وقد احببت أن أبدأ بهذا المثال لسهولة فهمه لتبين كم استخدام هذه المكتبة
سهلة

خوارزمية هذا البرنامج تكون بالشكل التالي

- ١ - بداية
- ٢ - قم بالحصول على الوقت من النظام و احفظه في متغير نصي
- ٣ - قم باستخدام الدالة sp.speak لنطق الوقت
- ٤ - النهاية

كود

Module mainmd

Sub Main()

Dim time As String

time = TimeOfDay.Hour & "-" & TimeOfDay.Minute

SP.Speak("its now-" & time)

End Sub

End Module

لنعد الى البرنامج كما لاحظت البرنامج يقوم فقط بنطق الساعة والدقيقة وبنظام اليوم الكامل(24 ساعة) لنقم بتحسين الكود وذلك باضافة جملة شرطية تقوم بالحويل الى نظام (12 ساعة)

الان ما رأيك بتعديل الجملة

time = TimeOfDay.Hour & "-" & TimeOfDay.Minute & " AM"

الى

If TimeOfDay.Hour > 12 then time = TimeOfDay.Hour - 12 & "-" &
TimeOfDay.Minute & " PM"

كود

Module mainmd

Sub Main()

Dim time As String

time = TimeOfDay.Hour & ":" & TimeOfDay.Minute & " am"

If TimeOfDay.Hour > 12 Then time = TimeOfDay.Hour - 12 & ":" &
TimeOfDay.Minute & " pm"

SP.Speak("its now-" & time)

End Sub

End Module

والآن ننتقل الى تطبيق آخر

التطبيق الثاني :برنامج تسجيل الدخول باستخدام أذخال كلمة السر صوتيا

برنامج يقوم بتسجيل الدخول لإعطاء المعلومات المطلوبة عن نوع سيارة عند قيام المستخدم بنطق كلمة السر معينة

في المثال السابق قمنا بعمل مثال على استخدام الاخراج من السماعة حيث يقوم البرنامج باستخدام التاريخ المخزن في النظام ومن ثم خزنه في نص ومن ثم تحويل النص الى كلام

وأما في هذا المثال سنقوم باستخدام أوامر التعرف على الكلام بين كلمات معينة لتحديد كلمة السر ثم نطق معلومات السيارة

الخوارزمية

١- بداية

٢- اضافة اوامر تجهيز المكتبة للتعرف على الكلام

٣- اضافة عدد من الكلمات الى البرنامج وبينها كلمة السر

٤- اطلب من المستخدم نطق كلمة السر

٥- عمل حلقة تكرارية لا نهائية تقوم بطلب النظام بالقيام بتفقد الأحداث

٦- في بلوك حدث التعرف على الكلام اذا كانت قيمة results.gettext=

password فقم بعض المعلومات وإذا لم تكن اطلب من المستخدم اعادة

المحاولة

٧- نهاية

قد يخطر في بالك من الخوارزمية أن الكود صعب وانا معك في ذلك ولكن باستخدام الأوامر في الوحدة النمطية ستكتشف ان الخوارزمية سهلة جدا لذا لا تنس اضافه الوحدة النمطية

قبل اضافة الكود ستحتاج الى اضافة مكتبة system.windows.form

وهي بنفس طريقة اضافة مكتبة speech

كود

```
Imports SpeechLib
Module mainmdle
```

```

Sub Main()
    SPRec_cr()
    addwrd("car")
    addwrd("red")
    addwrd("that")
    addwrd("name")
    Console.WriteLine("please Say the password in the
microphone")
    SPRec_BG()
    Do
        System.Windows.Forms.Application.DoEvents()
    Loop
End Sub

End Module

```

كود في بلوك الحدث في الوحدة النمطية

```

Public Sub Reco_Event(ByVal streamnmber As Integer, ByVal
streamposition As Object, ByVal Recognitiontype As
SpeechRecognitionType, ByVal Result As ISpeechRecoResult)
Handles objrec.Recognition
Dim str As String
str = Result.PhraseInfo.GetText(0, -1, True)
If str = "name" Then Console.WriteLine("Welcome to car reader"
& vbCrLf & " you have chosed BMW 1999 Model thank you for
register")
If Not str = "name" Then Console.WriteLine("Welcome to car
reader" & vbCrLf & "sorry you have said inviald passowrd")
End Sub

```

نلاحظ ان الكود في الدالة الرئيسة قد انتهى بالكود التالي

```

Do
    System.Windows.Forms.Application.DoEvents()
Loop

```

والهدف منه هو انهاء الكود من الدالة الرئيسية بحلقة تكرارية غير منتهية
تطلب من النظام استدعاء الأحداث وذلك لأن برنامج الدوس كما نعلم في
الوضع الطبيعي لا يستقبل تعدد المهام
أما اذا كنت تطبق الأمثلة على نظام windows application فلن تحتاج
الى اضافة هذه الدالة
وأن الكلمات المضافة الى المترجم car ,Red ,That , name

كما انه عليل كالا تنسى اضافة المكتبة System.Windows.Forms
وأخر ملاحظة هي أنه عليك عدم نسيان ملئ بلوك الحدث

وكما لاحظنا في هذا المثال قمنا بإدخال كلمة المرور شفويا وقد قبلها النظام
وهذا يعتبر مثال على الإدخال
وفي التطبيق التالي سوف ندرس برنامج يستخدم دالة الإدخال والإخراج وهو برنامج
يقوم برد التحية على شكل كلام بعد نطق التحية صوتيا
التطبيق الثالث: برنامج يقوم برد التحية
خوارزمية البرنامج
١ - بداية
٢ - قم بتجهيز طريقة التعرف على كلمات
٣ - أضف كلمة hey و hello و hi
٤ - قم بتشغيل الطريقة
٥ - في بلوك حدث التعرف قم بوضع كود يقوم بنطق الكلمة التي قالها المستخدم ومن ثم
في يقول how are you
٦ - نهاية

كود

```
Imports SpeechLib
Module mainmdle
```

```
Sub Main()
    SPreC_cr()
    addwrd("hey")
    addwrd("hello")
    addwrd("hi")
```

```
SPRec_BG()
```

```
Do
```

```
System.Windows.Forms.Application.DoEvents()
```

```

Loop
End Sub

End Module

```

كود في بلوك الحدث في الوحدة النمطية

```

Public Sub Reco_Event(ByVal streamnumber As Integer, ByVal
streamposition As Object, ByVal Recognitiontype As
SpeechRecognitionType, ByVal Result As ISpeechRecoResult)
Handles objrec.Recognition
SP.Speak(Result.PhraseInfo.GetText(0, -1, True) & ", How Are
You?")
End Sub

```

التطبيق الرابع: برنامج محادثة مع الحاسوب
لننتقل الى كود أعقد وهو كود برنامج محادثة صغير حيث يقوم المستخدم بالتحدث
الى الحاسوب ويقوم الحاسوب بفهم الكلام ومن ثم يقوم بالرد عليه وسنحدد عدد من
الكلمات التي سنتعامل معها وعدد ما من الردود
خوارزمية البرنامج
١ - تحضير طريقة الكلام للتعرف على الأصوات
٢ - اضافة الجمل التي نريد للحاسوب فهمها
وستكون بمثالنا

```

Hey
Hello
Hi
Im fine
How are you?
Thank you
Do you like milk?
Yes
No

```

٣ - تشغيل الطريقة

٤ - اضافة في بلوك التعرف جمل شرطية بحيث عندما يلتقط المترجم

Hi /hello /hey → (hey/hello / hey) ,how are you

Im fine → that's good

Thank you → Welcome

How are you → im fine
 Do you like milk → yes ,Do you
 Yes → cool
 No → maybe
 ٥ - نهاية

كود

```
Imports SpeechLib
Module mainmdle

Sub Main()
    SPreC_cr()
    addwrd("hey")
    addwrd("hello")
    addwrd("hi")
    addwrd("hey")
    addwrd("im fine")
    addwrd("how are you?")
    addwrd("thank you")
    addwrd("do you like milk?")
    addwrd("yes")
    addwrd("no")

    SPreC_BG()
    Do
        System.Windows.Forms.Application.DoEvents()
    Loop
End Sub

End Module
```

كود في بلوك الحدث في الوحدة النمطية

```
Public Sub Reco_Event(ByVal streamnumber As Integer, ByVal
streamposition As Object, ByVal Recognitiontype As
SpeechRecognitionType, ByVal Result As ISpeechRecoResult) Handles
objrec.Recognition
```

```
Dim x As String = Result.PhraseInfo.GetText(0, -1, True)
```

```
If x = "hey" Or x = "hello" Or x = "hi" Then
SP.Speak(Result.PhraseInfo.GetText(0, -1, True) & ", How Are You?",
SpeechVoiceSpeakFlags.SVSFlagsAsync)
```

```
If x = "im fine" Then SP.Speak("thats good")
```

```
If x = "how are you?" Then SP.Speak("im fine")
```

```
If x = "thank you" Then SP.Speak("welcome")
```

```
If x = "do you like milk?" Then SP.Speak("yes ,do you?")
```

```
If x = "yes" Then SP.Speak("cool")
```

```
If x = "no" Then SP.Speak("maybe")
```

```
End Sub
```

لاحظ أننا عندما وضعنا الكلمات كانت كل الحروف وضعناها صغيرة وهذا ليس شرط ولكن يفضل فعل ذلك لتسهيل على نفسك عملية المقارنة بين الكلمة المحيكة والمخزنة في الذاكرة

التطبيق الخامس: اله حاسبة باستخدام الاوامر الصوتية

البرنامج التالي يوضح طريقة عمل برنامج يقوم بجمع الأرقام أو طرحها باستخدام الأوامر الصوتية

خوارزمية البرنامج

- ١ - بداية
- ٢ - اضافة أوامر تجهيز مكتبة التعرف لى الكلام
- ٣ - اضافة الأرقام من 1-1000
- ٤ - اضافة اسماء العمليات
- ٥ - بدأ الاستماع
- ٦ - الطلب من المستخدم ادخال الرقم الذي يريد صوتيا
- ٧ - انتظار حتى يقوم المستخدم بادخال الرقم
- ٨ - الطلب من المستخدم ادخال العملية
- ٩ - الانتظار حتى يقوم بالادخال
- ١٠ - الطلب من المستخدم ادخال الرقم الثاني
- ١١ - الانتظار حتى يقوم بالادخال
- ١٢ - اجراء العملية المختارة على الأرقام المدخلة
- ١٣ - نهاية

كود

```
Imports SpeechLib
Module mainmdle

    Sub Main()
        SPRec_cr()
        Dim num As Integer
        For num = 1 To 1000
            addwrd(num.ToString)
        Next

        addwrd("add")
        addwrd("subtract")
        addwrd("multiple")
        addwrd("divide")

        SPRec_BG()

        SP.Speak("Please say the first number")
        Do
            System.Windows.Forms.Application.DoEvents()
        Loop

    End Sub

End Module
```

كود الحدث في الوحدة النمطية

```
Public Sub Reco_Event(ByVal streamnumber As Integer, ByVal
streamposition As Object, ByVal Recognitiontype As
SpeechRecognitionType, ByVal Result As ISpeechRecoResult)
Handles objrec.Recognition
```



```

Dim r As String = Result.PhraseInfo.GetText(0, -1, True)
    Static x As Integer ' سنستخدم هذا المتغير لحفظ تسلسل الادخال بقيمته يحدد
    الموقع اهو عند الرقم 1 ام العملية ام الرقم 2
    Static n1, n2 As Double ' هذا المتغير سيحفظ الأعداد المدخلة
    Static op As Int16 ' هذا المتغير سيحفظ لنا العملية

    If x = 0 And (Val(r) > 0) Then ' تحديد الرقم الأول

        n1 = Val(r)
        x = 1

        SP.Speak("Please say the operation")
        Exit Sub
    End If
    If x = 1 And (r = "add" Or r = "subtract" Or r = "multiple" Or
r = "divide") Then ' تحديد العملية

        If r = "add" Then
            op = 1

            x = 2
        End If
        If r = "subtract" Then
            op = 2

            x = 2
        End If
        If r = "multiple" Then
            op = 3

            x = 2
        End If
        If r = "divide" Then
            op = 4

            x = 2
        End If
        SP.Speak("Please say the second number")
        x = 2
        Exit Sub
    End If
    If x = 2 And (Val(r) > 0) Then ' تحديد الرقم الثاني واجراء العملية

```

المختارة ونطق الجواب

```

n2 = Val(r)
Dim total As Double
If op = 1 Then total = Val(n1 + n2)
If op = 2 Then total = Val(n1 - n2)
If op = 3 Then total = Val(n1 * n2)
If op = 4 Then total = Val(n1 / n2)
SP.Speak("the total is - " & total)
SP.Speak("Please say the first number")
x = 0

End If

End Sub

```

لاحظ أننا عن استبدالنا كلمة Dim بكلمة static وذلك حتى تبقى القيمة مخزنة في الذاكرة بعد تحديدها وانتهاء تنفيذ الاجراء

وبهذا المثال نكون قد أنهيان التطبيقات على هذه المكتبة من حيث الادخال والاخراج
 بقي هذا المثال الأخير أقوم بوضعه كمثال على عملية الاملاء وتكون بهذا قد اتقنت استخدام هذه المكتبة

التطبيق السادس ولأخير: برنامج املاء يقوم بتخزين الكلام الى ملف وطباعته على الشاشة
 وهو برنامج يقوم بعرض الكلام المنقل على الشاشة و من ثم حفظه الى ملف
 وهو برنامج بسيط جدا ولكنه يفيدك بأن تعلمك طريقة الاملاء
 خوارزمية البرنامج

- ١ - بداية
- ٢ - أدخل الموقع الذي تريد حفظ الملف فيه
- ٣ - جهز مكتبة الاملاء
- ٤ - قم باضافة أمر الكتابة الى الملف عند حدوث املاء جديدة
- ٥ - اطبع على الشاشة الكلام
- ٦ - نهاية

كود

Module Module1

```
Public path As String
Sub Main()
    Console.WriteLine("Please Enter The Path you want to save to>")
    path = Console.ReadLine
    Dic()
    Do
        Windows.Forms.Application.DoEvents()
    Loop
End Sub

End Module
```

كود في الوحدة النمطية

```
Public Sub Reco_Event(ByVal streamnumber As Integer, ByVal
streamposition As Object, ByVal Recognitiontype As
SpeechRecognitionType, ByVal Result As ISpeechRecoResult) Handles
objrec.Recognition
    Console.WriteLine(Result.PhraseInfo.GetText(0, -1, True))
    Dim file As New IO.StreamWriter(path, True)
    file.WriteLine(Result.PhraseInfo.GetText(0, -1, True))
    file.Close()
End Sub
```

لقد استخدمنا في هذا الكود اوامر من مكتبة IO لنتمكن من الكتابة على الملف و حفظ الاملاء

وبهذا المثال تنتهي دروسنا ومبارك لقد أتقنت مكتبة Microsoft SpeechSDK الى اللقاء مع كتاب آخر وأرجو لكم بدعوة صالحة منكم فهذا كل ما ابغ من كتابة هذا الكتاب وليس أي شيء اخر والسلام عليكم