

ABAHE

الفن والبيسك

Visual Basic

## المحتويات

### واجهة الفيجوال بيسك

ABAHE

#### شرح واجهة العمل في الفيجوال بيسك

- نافذة النموذج
- شرح شريط الأدوات (Toolbar) مع أمثلة
- أشرطة القوائم

### قواعد البيانات

- طرق إنشاء قواعد البيانات
- إعداد برنامجي الأول
- التعامل مع الفورم
- البحث في الفيجوال بيسك

## أدوات ومهارات

- إنشاء القوائم
- إطارات الرسائل المختلفة
- العبارة الشرطية IF
- الطباعة
- حلقات التكرار For
- عمل تقرير طباعة
- إظهار البيانات في جدول

## المتמידيا

- تشغيل الصوت
- تشغيل الصورة

## حزم البرنامج وتجهيز الـ Setup الخاص به

## مشروع

## الفيجوال بيسك

إن جميع لغات البرمجة تحتوي على أوامر يستخدمها المبرمجون في بناء برامجهم، فالمبرمج يقوم بتسخير هذه الأوامر لخدمته ويوظفها بطريقة معينة لكي يستفيد منها بأكبر قدر ممكن، ولكل لغة من لغات البرمجة قدرات خاصة ومزايا معينة تميزها عن غيرها من اللغات.

**كلمة البيسك بشكل عام تعني ترميز التعليمات لأغراض المبتدئين كافة.** تمتاز لغة البيسك بأنها سهلة الاستخدام وسهلة التعلم. وتستعمل البيسك في المجالات العلمية والتجارية ومجالات أخرى.

أصل لغة فجوال بيسك هي لغة بيسك التي ظهرت في كلية دارتماوث Dartmouth علي يد جون كيمني John Kemeny وتوماس كيرتز Thomas Kurtz عام 1963م و قد أصبحت بسرعة فائقة من أشهر و أسهل لغات البرمجة بل لم تقف عند هذا فحسب فأصبحت كلغة تعليمية تدرّس في المعاهد و الكليات و الجامعات و بدأ انتشارها في السبعينات ثم أخذت هذه اللغة في التطور الدائم مع المحافظة على سهولة استخدامها و بساطتها و كان الإصدار الأول لهذه اللغة عام 1991 م ثم كان الإصدار الثاني منها عام 1992 ثم كان الإصدار الثالث عام 1993 ومن ذلك الوقت تعاقبت إصدارات كثيرة منها.

وتختلف وجهات النظر بين المبرمجين حول تعريف أو تصنيف Visual Basic هل Visual Basic لغة برمجة أم هي برنامج تصميم نوافذ أم منتج متكامل؟  
تقنياً، Visual Basic عبارة عن مكتبة من المكتبات توفر عدة أوامر برمجية متكاملة وبرنامج تصميم نوافذ ومكتبات تشغيلية، أي باختصار هو منتج متكامل.

منذ بداية اكتشاف الحاسب الآلي وحتى الآن فإن لغات البرمجة تعتبر الشغل الشاغل لكل من أراد الاستفادة من الحاسب بأكبر قدر ممكن ، فلقد كان الهاجس الأكبر لعلماء الكمبيوتر بعد اختراعه هو كيفية تبسيط التعامل معه والاستفادة منه بأكبر وأسهل وأيسر وسيلة ممكنة، ولا يتم ذلك إلا بواسطة طريقة واحدة ألا وهي لغات البرمجة، حيث أن هذه اللغات هي التي تجعل الحاسب يقوم بالعمليات التي نريدها منه، فبدون هذه اللغات يصبح الحاسب كتلة هامة من المعدن.

وتنقسم لغات البرمجة إلى قسمين :

**لغات برمجة منخفضة المستوى (Low-Level Languages)**

ومن هذه اللغات لغة الآلة (Machine Language)

ولغة الاسمبلي (Assembly Language)

**لغات البرمجة عالية المستوى (High-Level Languages)**

ومن هذه اللغات لغة الفورتران (Fortran Language)

ولغة الكوبول (Cobol Language)

ولغة البيسيك (Basic) بأنواعها المختلفة.

ومع التطور والتقدم في مجال الكمبيوتر، والقفزات الهائلة في علم الحواسيب، ظهرت لغة برمجة جديدة تعتبر امتداداً للغة البيسيك ألا وهي الفيجوال بيسك، وأصبحت هذه اللغة ذات إمكانيات مذهلة، ومن أهم خصائص هذه اللغة أنها قادرة على التعامل مع قواعد البيانات بسهولة مذهلة، وأن طريقتها في التعامل مع البيانات مميزة.

ولقد بذلت شركة مايكروسوفت أقصى ما بوسعها لتزيد من قوة هذه اللغة وطورتها طبقاً للتطور الحادث في مجال الكمبيوتر وطبقاً لرغبات المبرمجين ومستخدمي البرامج إلى

أن ظهر منها الإصدار السادس (Microsoft Visual Basic 6.0)

الذي يعتبر آخر إصدارات الفيجوال بيسك، ويعتبر هذا الإصدار أقوى إصدارات هذه اللغة في التعامل مع البيانات حتى أنه تفوق على بعض لغات البرمجة الأخرى.

### ومن مميزاتهما:

- أصبح بمقدور المبرمجين أن يصمموا واجهات تطبيق جميلة حيث توفر لهم لغة الفيچوال بيسيك كائنات (Objects) وأدوات تستخدم في نظام التشغيل نفسه.
- وجاءت الميزة الأخرى والدعم الآخر الذي تلقته لغة الفيچوال بيسيك هو أيضاً من شركة مايكروسوفت حيث تم إضافة إمكانية استدعاء الكثير من الوظائف الموجودة في نظام التشغيل وإمكانية الاستفادة من بعض البرامج التي تعمل تحت بيئة نظام النوافذ (Windows).

ومع كل هذا التطور بقيت الفيچوال بيسيك ضعيفة في معالجة قواعد البيانات وفي إنشاء برامج وتطبيقات دعم الإنترنت مما دفع الشركات الأخرى لأن تستفيد من هذه السلبات الموجودة في لغة الفيچوال بيسيك وتطور لغات برمجية منافسة مثل شركة (Borland) التي طرحت عدة إصدارات من لغتها البرمجية (ديلفي) والتي تتمتع بمحرك قواعد بيانات قوي ، وقد دخلت شركة مايكروسوفت في منافسة قوية مع هذه الشركة جعلت شركة مايكروسوفت تقوم بتطوير لغة الفيچوال بيسيك ، فجاء الإصدار الخامس الذي يعتبر قوياً في معالجة البيانات وجاء بعده الإصدار السادس كما ذكرنا الذي يملك إمكانيات هائلة في التعامل مع قواعد البيانات.

## واجهة الفيجوال بيسك

ستظهر هذه الواجهة عند بدء تشغيل الفيجوال بيسك ويمكن إزالتها من بدء التشغيل عن طريق تفعيل علامة الاختيار Don't Show this dialog in the future



علامات التبويب في أعلى الواجهة:

**New** وتعني اختيار مشروع جديد

**Existing** فتح ملفات مخزنة لديك

**Recent** قائمة بأخر الملفات التي قمت بفتحها حسب التاريخ

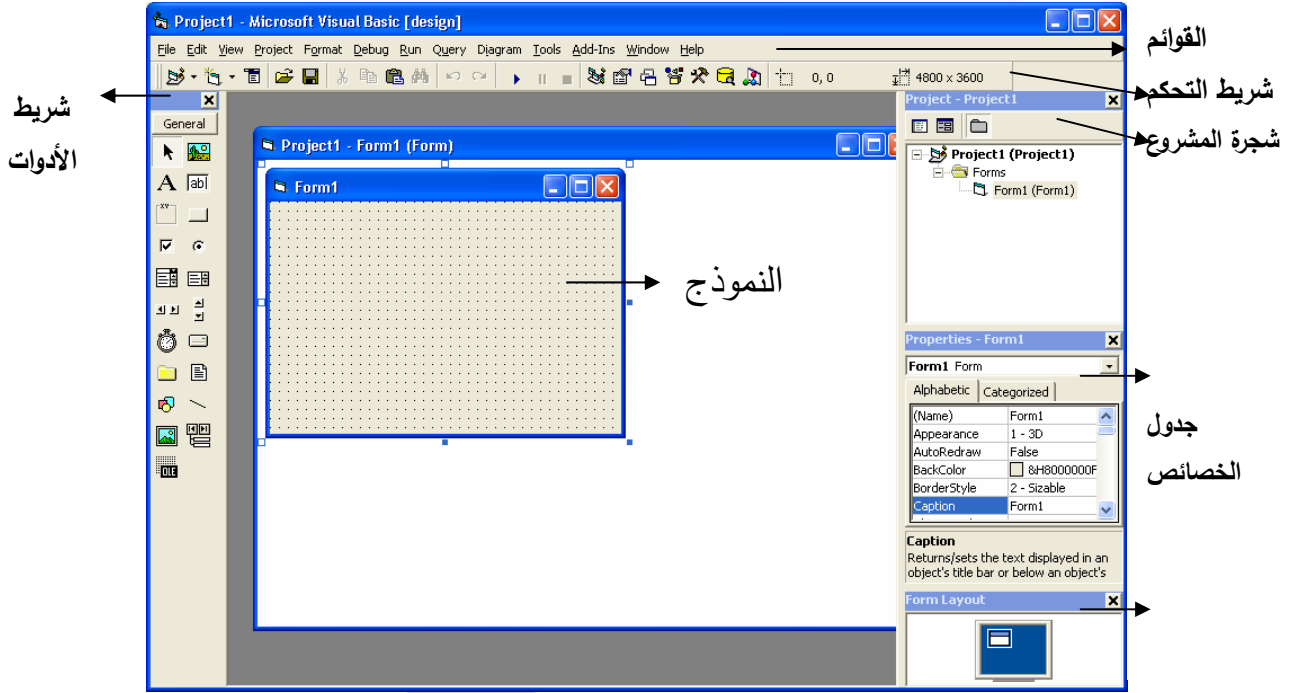
وسنشرح الآن محتويات البند **New** (جديد) نظراً لأهميته وتعدد خياراته:

التطبيق	الشرح
Standard EXE	ينشئ برنامج تنفيذي بعد طلبك لذلك من ( Make EXE )
ActiveX EXE	ينشئ ملفات لها امتداد EXE تعمل مع برامج أخرى في شكل OLE
ActiveX DLL	ينشئ ملف له الامتداد (DLL) وهو ملف ذو برامج فرعية مساعدة وهو لا يعمل بنفسه بل مع exe
ActiveX control	ينشئ ملف من نوع (ocx) وهي تساعد على احتواء واجهة مستخدم أو برامج فرعية
ActiveX Document DLL	ينشئ ملف ذو امتداد (DLL) يقوم بتشغيل برامج على الإنترنت.
ActiveX Document exe	ينشئ ملفات تظهر على الإنترنت
Add In	خاص بالفيجوال بيسك حيث يمكنك إضافة واجهة
VB Application Wizard	معالج تلقائي في البيسك يساعد على إنشاء نوافذ دون كتابة سطر واحد من النوع EXE .
ISS Application	ISS اختصار لـ (Internet Information Server) ويحضر لتصميم (web class) وهو ما يسمى صفحات HTML
Data project	تضع مشروعاً لقواعد بيانات وتهيئها في بيئة متكاملة من أدوات الربط وصنع التقارير دون الحاجة لفتح برامج أخرى .
Vb wizard manager	تبدو مشابهة لمعالجات ( Microsoft ) يمكنك استخدامه .
DHTML Application	تساعد في صنع صفحات HTML للإنترنت دون وجود خبرة في لغة الـ Java
Vb Enterprise Edition controls	تساعد على وضع جميع الأدوات المستخدمة لقواعد البيانات والاتصال وتطبيقها.



## شرح واجهة العمل في الفيجوال بيسك

قبل البدء في أي شيء لنتعرف على الأشرطة والأدوات التي سنستعملها خلال تعاملنا مع هذه اللغة. وستظهر لنا الواجهة على النحو التالي:



وستحدث عن بعض هذه الأقسام . إن صح التعبير . بشكل مفصل حسب أهمية كلاً

منها

## نافذة النموذج:

هذه النافذة تعتبر من عوامل نجاح Visual Basic وهي من اشهر النوافذ منذ الإصدار الأول وتأخذ دائماً الاسم Form1 ومن خلال هذه النافذة تقوم بعملية تصميم واجهة برنامجك أو بتعديل خصائصها أو وضع باقة أدوات عليها باستخدام الفأرة وهذه التقنية تثبت مدى قوة ونجاح فكرة لغات البرمجة المرئية وانتهاء عصر التصميم من خلال الأكواد والماكرو. وكما ذكرنا يمكنك من خلال هذه النافذة التعامل مع كافة الأدوات من خلال الماوس ولكن الأهم هو كيفية التعامل مع هذه النافذة من خلال تلك الأدوات.

ونلفت النظر إلى وجود نوعين من الأدوات التي يمكن استخدامها مع Visual Basic النوع الأول هي الأدوات الداخلية - Built-in Controls أو الجوهرية Intrinsic Controls - وهي العشرين أداة الموجودة مبدئياً.

## شريط الأدوات (Toolbar):

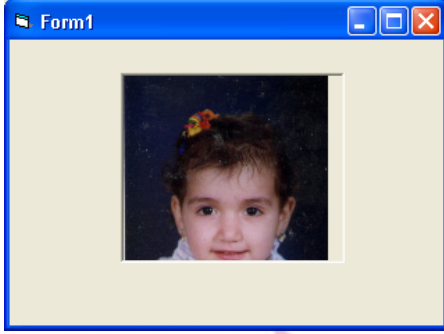
سنتعرف الآن على هذه الأيقونات التي يتألف منها هذا الشريط وعددها 20 أداة ثم سنتناول بشكل مفصل كيفية التعامل مع أهمها.

الشرح	الاسم	الأيقونة
صندوق عرض الصور	Picture Box	
أداة عرض النصوص	Label	
أداة النصوص والكتابة	Text Box	
إنشاء إطارات داخل البرنامج	Frame	
زر أمر	Command Button	
زر اختيار	Check Box	
زر خيارات	Option Box	
قائمة منسدلة	Combo Box	
أداة القائمة	List Box	
شريط التمرير الأفقي	Horizontal Scroll Bar	
شريط التمرير العمودي	Vertical Scroll Bar	
أداة التوقيت	Timer	
أداة عرض السواقات	Drive Box	
أداة عرض المجلدات	Dir Box	
أداة عرض الملفات	File Box	
أداة الأشكال	Shape	
رسم خط	Line	
أداة الصور	Image	
ربط قاعدة بيانات	Data Control	
عنصر تحكم Ole	Ole Object	

### أيقونة الصور PictureBox



يستخدم لعرض الصور بحجمها الطبيعي وله استخدامات كثيرة أخرى فهو يستخدم كإطار لأدوات التحكم وأيضاً لعرض الفيديو و للرسم ويمكن تحويله إلى أداة عرض نصوص و زر أمر، وسوف نتحدث عن ذلك لاحقاً.

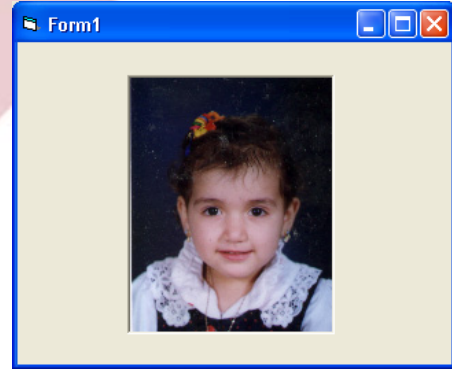


لوضع صورة: من مربع الخصائص الخاص بـ Picture سوف تجد الخاصية Picture انقر فوق الزر ثم حدد مكان وجود الصورة المطلوبة. لاحظ هنا أن الصورة ظهرت غير كاملة.

ولكن إذا فعلنا الخاصية AutoSize لظهرت لنا الصورة على الشكل التالي:

يمتاز صندوق الصور Picture عن صندوق عرض الصور Image بمايلي:

- يعرض الصورة بحجمها الطبيعي ولو صغرت إطار الصندوق فلن تصغر الصورة.

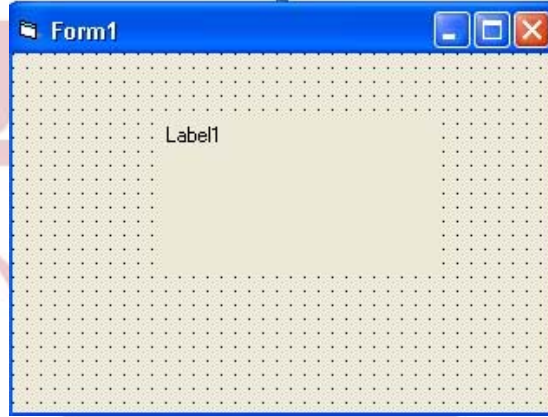
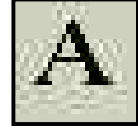


- يمكن تغيير لون الخلفية ونمط الإطار
- يستخدم في أغراض كثيرة.

### الخصائص الأساسية لـ Picture Box :

معناها	اسم الخاصية
اسم الأداة	Name
محاذاة	Align
نوع الإطار	Appearance
حجم تلقائي حسب الصورة	AutoSize
لون الخلفية	BackColor
نمط الإطار	BorderStyle
النص	Caption
الفاعلية	Enabled
نوع الخط	Font
لون الخط	ForeColor
الارتفاع	Height
ترتيب مفتاح Tab	Index
البعد عن الحافة اليسرى	Left
مؤشر الماوس	MousePointer
الأيقونة التي تمثل مؤشر الماوس	MouseIcon
الصورة	Picture
الاتجاه من اليمين لليسار	RightToLeft
ترتيب Tab	TabIndex
إيقاف الـ Tab	TabStop
ما سيظهر عند وضع الماوس فوقها	ToolTipText
Tag	Tag
النص	Caption
البعد عن حافة النافذة العليا	Top
مرئي أو مخفي	Visible
العرض	Width

**Label** تستخدم هذه الأداة لإظهار وكتابة النصوص على النموذج .  
ولكن بشكل نهائي حيث لا يمكن بعد ذلك تعديلها أو الكتابة فيها بعد إنهاء  
البرنامج والعمل عليه بصورة EXE.

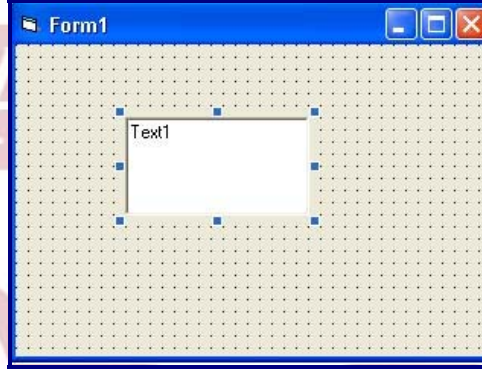


لتغيير النص الذي سيظهر بداخلها طبعاً عن طريق Caption  
وهذا الجدول يبين الخصائص الأساسية لها:

اسم الخاصية	معناها
Name	اسم الأداة
Alignment	محاذاة
Appearance	الظهور
AutoSize	حجم تلقائي حسب النص
BackColor	لون الخلفية
BorderStyle	نمط الإطار
Caption	النص
Enabled	الفاعلية
Font	نوع الخط
ForeColor	لون الخط
Left	البعد عن الحافة اليسرى

مؤشر الماوس	MousePointer
الأيقونة التي تمثل مؤشر الماوس	MouseIcon
الاتجاه من اليمين لليساار	RightToLeft
ترتيب Tab	TabIndex
إيقاف الTab	TabStop
ما سيظهر عند وضع الماوس فوقها	ToolTipText
Tag	Tag
النص	Caption
البعد عن حافة النافذة العليا	Top
مرئي أو مخفي	Visible
العرض	Width

**TextBox** تستخدم في إدخال النصوص إلى البرنامج كما يمكن استخدامها في إخراج النصوص. وهي لا تختلف كثيراً عن Label سوى أنها تمكنك من الكتابة فيها.



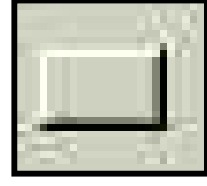
وهذا الجدول يبين الخصائص الأساسية لها:

معناها	اسم الخاصية
اسم الأداة	Name
محاذاة	Alignment
الظهور	Appearance
حجم تلقائي حسب النص	AutoSize
لون الخلفية	BackColor
نمط الإطار	BorderStyle
التصحيح	CauseValidation
النص	Text
مقفول لا يمكن الكتابة بداخلها	Locked
الحد الأقصى	MaxLength
الفاعلية	Enabled



نوع الخط	Font
لون الخط	ForeColor
البعد عن الحافة اليسرى	Left
مؤشر الماوس	MousePointer
الأيقونة التي تمثل مؤشر الماوس	MouseIcon
حرف كلمة المرور	PasswordChar
هل تستخدم لعدة أسطر أم لا	MultiLine
أشرطة التمرير	ScrollBars
الاتجاه من اليمين لليسار	RightToLeft
ترتيب Tab	TabIndex
إيقاف الـ Tab	TabStop
ما سيظهر عند وضع الماوس فوقها	ToolTipText
Tag	Tag
النص	Caption
البعد عن حافة النافذة العليا	Top
مرئي أو مخفي	Visible
العرض	Width

## زر الأمر Command Button



هو من أهم أدوات الفيجوال بيسك وإذا عملت أي برنامج فسوف تحتاج بالتأكيد إلى وضع زر أمر في برنامجك

فهو الأداة الأساسية للأحداث ومن خلال الخصائص الأساسية يمكنك القيام بالكثير من التغييرات على هذا الزر.

وهذا الجدول يبين هذه الخصائص الأساسية:

اسم الخاصية	معناها
Name	اسم الأداة
Appearance	الظهور
BackColor	لون الخلفية
Caption	النص
Default	زر افتراضي عند ضغط مفتاح الإدخال
DisablePicture	الصورة عندما يكون غير متاح
DownPicture	الصورة عند الضغط لأسفل
Enabled	الفاعلية
Font	نوع الخط
ForeColor	لون الخط
MaskColor	قناع اللون
Style	النمط
Left	البعد عن الحافة اليسرى
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تمثل مؤشر الماوس
RightToLeft	الاتجاه من اليمين لليسار
TabIndex	ترتيب Tab
TabStop	إيقاف ال Tab

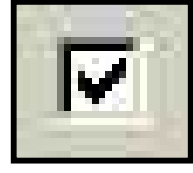
ما سيظهر عند وضع الماوس فوقها	ToolTipText
Tag	Tag
الصورة في الوضع العادي	Picture
البعد عن حافة النافذة العليا	Top
مرئي أو مخفي	Visible
العرض	Width



## Option و Check Box

الفرق بين زر الخيار وخانة الاختيار هو بالأداء وليس بالشكل فلو كان لديك فورم ووضعت فيه أكثر من خانة اختيار Check فإنك تستطيع أن تحدد كافة الخانات الموجودة على هذا الفورم مع بعضها.

ولكن لو كان لديك أكثر من زر خيار Option فلن تستطيع سوى أن تحدد واحداً منها فقط.



إن النص الذي بجانب كل أداة هو تابع لها يمكنك تغييره عن طريق خاصية Caption وعند النقر عليه تتغير قيمة الأداة



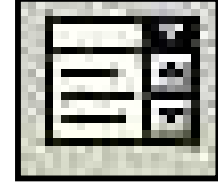
الخصائص الأساسية لكلا الأدوات:

اسم الخاصية	معناها
Name	اسم الأداة
Alignment	محاذاة
Appearance	الظهور
BackColor	لون الخلفية
Caption	النص
DisabledPicture	الصورة عند عدم الفاعلية
DownPicture	الصورة عند النقر للأسفل
Enabled	الفاعلية
Font	نوع الخط

لون الخط	ForeColor
البعد عن الحافة اليسرى	Left
مؤشر الماوس	MousePointer
الأيقونة التي تمثل مؤشر الماوس	MouseIcon
الاتجاه من اليمين لليسار	RightToLeft
ترتيب Tab	TabIndex
إيقاف الـ Tab	TabStop
ما سيظهر عند وضع الماوس فوقها	ToolTipText
Tag	Tag
الصورة	Picture
النمط	Style
القيمة	Value
البعد عن حافة النافذة العليا	Top
مرئي أو مخفي	Visible
العرض	Width

## Combo Box و List Box

هذا الأمر يستخدم لإنشاء قائمة تحتوي على قيم أو معلومات ثم يتم الاختيار منها حسب الشكل أدناه.



ويمكن إضافة الأدوات بنفس الطريقة عن طريق الخاصية List حيث يمكنك كتابة العناصر الخاصة بتلك القائمة.

كما يمكنك الربط بقاعدة بيانات جاهزة ويتم عرض السجلات بداخلها.

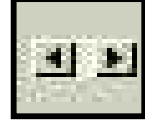
### جدول يبين الخصائص الأساسية لـ Box List

اسم الخاصية	معناها
Name	اسم الأداة
Appearance	الظهور
columns	عرض التحديد
Back Color	لون الخلفية
Enabled	الفاعلية
Font	نوع الخط
ForeColor	الخط لون
Left	البعد عن الحافة اليسرى
MultiSelect	السماح بتحديدات متعددة
Style	يحتوي على خانة اختيار
MousePointer	مؤشر الماوس

الأيقونة التي تشمل مؤشر الماوس	MouseIcon
الاتجاه من اليمين لليساار	RightToLeft
ترتيب Tab	TabIndex
إيقاف الـ Tab	TabStop
ما سيظهر عند وضع الماوس فوقها	ToolTipText
Tag	Tag
النص	Caption
عن حافة النافذة العليا البعد	Top
أو مخفي مرئي	Visible
العرض	Width

وخصائص ComboBox هي نفس الخصائص السابقة مع بعض التغييرات:  
Text وتعني النص الذي سيظهر في التحديد الأول  
Locked ممنوع من الكتابة بداخله وتختلف خاصية Style أيضاً.

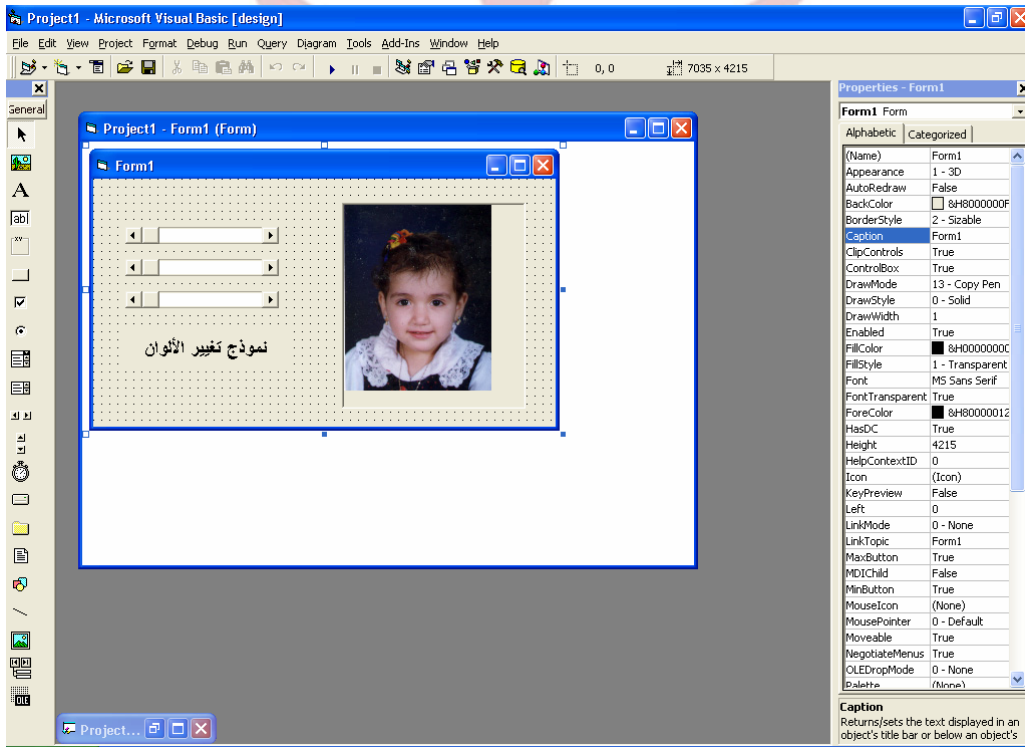
## أشرطة التمرير الأفقية والعمودية:



من المعروف والمتداول من خلال البرامج التي نتعامل معها أن هذه الأشرطة تقوم بعرض معلومات بشكل متتالي ولكن المفهوم الأعمق لهذه الأشرطة هو إمكانية تغيير قيمة ما زيادة أو نقصان.



لتوضيح هذه الفكرة سأعرض المثال التالي . أرجو أن تكونوا قد نفذتم المثال السابق الخاص بالصورة. سنضيف عليه ثلاثة أشرطة تمرير ونلغي خيار الحجم التلقائي للصور ونكبر حجم الإطار الخاص بها قليلاً. ولا ننسى طبعاً كتابة (نموذج تغيير الألوان) من خلال الأداة Label.



لنضع الآن الصيغة التالية في محرر الأكواد وذلك بالضغط المزدوج على مكان فارغ في ال form1.



## الصيغة:

HScroll1.Min = 0

HScroll1.Max = 250

HScroll1.SmallChange = 30

HScroll1.LargeChange = 50

HScroll2.Min = 0

HScroll2.Max = 250

HScroll2.SmallChange = 30

HScroll2.LargeChange = 50

HScroll3.Min = 0

HScroll3.Max = 250

HScroll3.SmallChange = 30

HScroll3.LargeChange = 50

وإذا قرأنا هذه الصيغة أو الشيفرة أو الكود (سمها ما شئت) سنفهم بأننا قد حددنا لكل شريط القيمة الصغرى الكبرى وكذلك سرعة التمرير.

بقي أن نحدد مهمة كل شريط وسأفرض أنني أرغب بتغيير ألوان خلفية الصورة فيجب أن أحدد لكل شريط علاقة لونية يستطيع من خلالها التحكم باللون ومن ثم تغييره مع مراعاة حركة الأشرطة الأخرى.

وبالضغط على الشريط الأول نكتب الكود التالي:

**Picture1.BackColor = RGB(HScroll1.Value, HScroll2.Value, HScroll3.Value)**

ثم نكرر نفس العملية للشريط الثاني والثالث.

حاول الآن تشغيل البرنامج وذلك من خلال زر Start من شريط التحكم أو من قائمة Run لتظهر لك النتيجة على الشكل التالي:



حاول الآن التلاعب بأشرطة التمرير وشاهد النتائج.

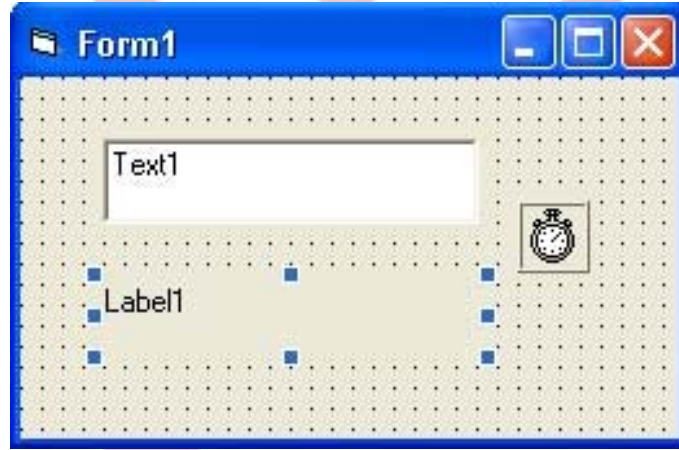
## Timer



قبل البدء في شرح هذه الأداة يجب أن نعلم بأن كلمة Timer هي أداة للوقت وليس الوقت فقط هو الساعة بل أيضاً التاريخ أي يمكننا من خلال هذه الأداة التعامل مع الوقت والتاريخ وتستخدم هذه الأداة لعمل حدث بعد وقت معين أو تكرار الحدث.

وسأشرح هذه الأداة من خلال مثال بسيط وهو تصميم فورم يحتوي على الساعة والتاريخ:

قم بتصميم فورم وأضف إليه Timer و Text و Label



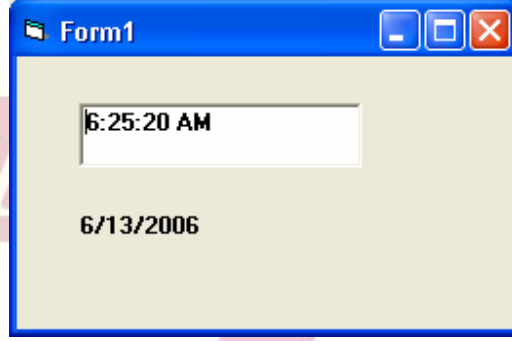
غير خاصية Interval إلى 500 أو أي قيمة تحت 1000  
أنقر فوق التايمر نقرتين وأكتب مايلي:

**Label1 = Date**

**Text1 = Time**

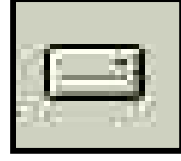
لاحظ جيداً العناصر المتوضعة على الفورم.

نفذ المثال وسوف يظهر لك الوقت والتاريخ في أدواتي عرض النصوص على الشكل التالي:

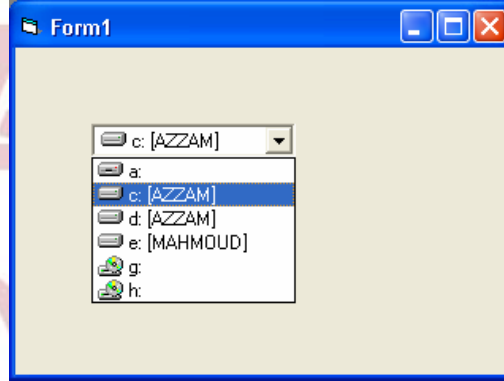


لاحظ الآن البرنامج بعد التشغيل وقارن بينه وبين الفورم الأساسي...؟! من الواضح أن أداة Timer لم تظهر في البرنامج ولكن ظهرت نتائجها.

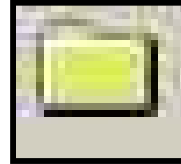
## أداة عرض السواقات DriverList Box



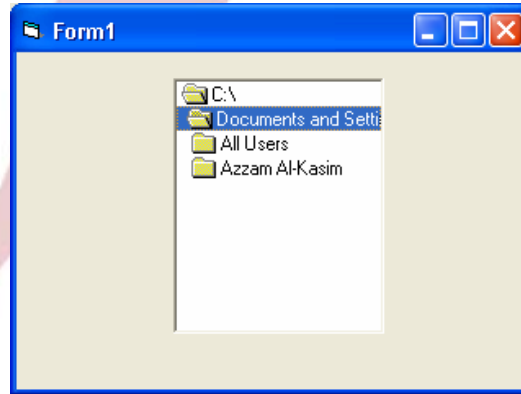
هذه الأداة رغم أهميتها إلا أنها سهلة في التعامل بشكل كبير ما عليك سوى سحبها ووضعها داخل الفورم لتظهر لك قائمة بسواقات جهازك. على الشكل التالي:



## أداة عرض المجلدات DirList Box



وهي كسابقتها من حيث سهولة الاستعمال:



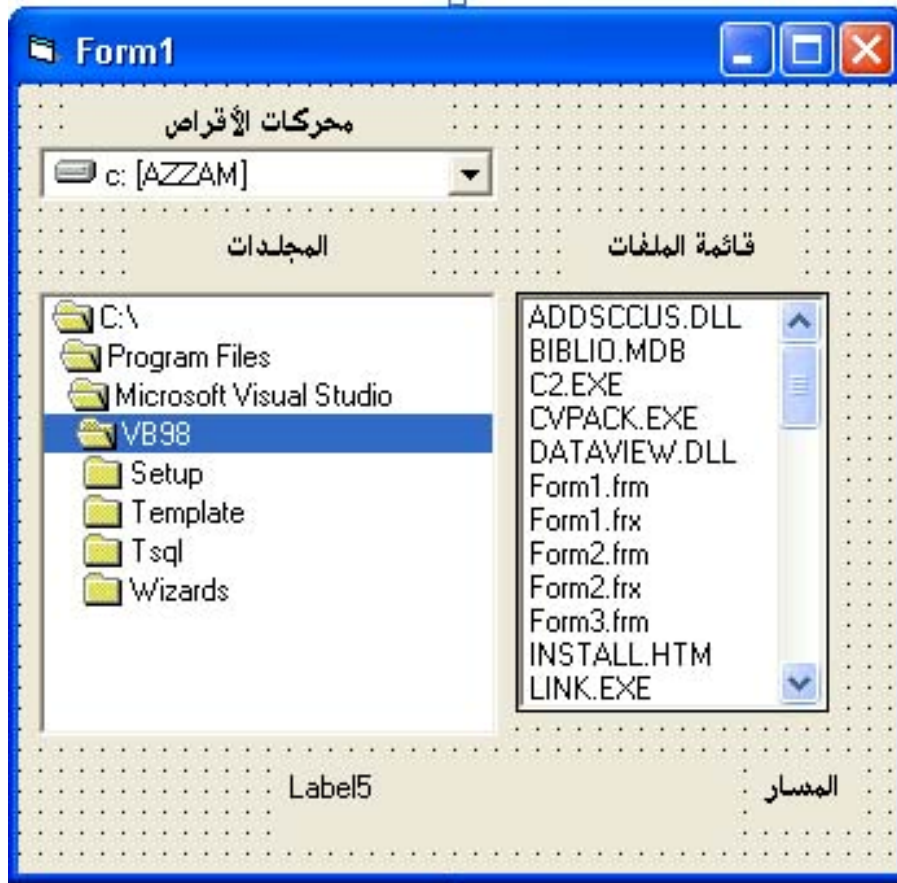
## أداة عرض الملفات FileList Box



وهي كسابقتها من حيث سهولة الاستعمال

مثال توضيحي حول الأدوات الثلاثة السابقة:

لنضع الأدوات الثلاث السابقة في فورم جديد وبترتيب مناسب ثم نضيف Label أسفل أداة عرض المجلدات وذلك على الشكل التالي:



والآن نأتي إلى عملية الربط بين هذه الأدوات وقبل ذلك علينا أن نعرف الاسم البرمجي لكل أداة وهذا سهل جداً من خلال جدول الخصائص وبند Name حيث يظهر لنا اسم الأداة عند تحديدها والأدوات هنا تكون أسمائها على الشكل التالي:

صندوق محرك الأقراص Drive1

صندوق المجلدات Dir1

صندوق الملفات File1

الآن أصبح الأمر أسهل فكل ما علينا هو أن نجعل صندوق المجلدات يُظهر محتويات محرك الأقراص وصندوق الملفات يُظهر محتويات المجلدات. ذلك عن طريق الصيغ التالية:

☒ نقرتين فوق صندوق المجلدات ثم نكتب:

**File1 = Dir1**

**Label5 = Dir1**

السطر الأول: عرض الملفات التي داخل هذا المجلد (في صندوق الملفات File1)

السطر الثاني: عرض المسار الكامل في أداة النصوص Label5

☒ نقرتين فوق صندوق محركات الأقراص ثم نكتب:

**On Error Resume Next**

**Dir1 = Drive1**

السطر الأول يعني عند الخطأ لا تتابع مثلاً لو حدد محرك القرص المرن ولم يكن هناك قرص فسوف يعطيك رسالة خطأ. أما هذا السطر يمنع الخطأ ويتابع العمل. أما السطر الثاني فيعرض المجلدات الخاصة بمحرك الأقراص الذي تم تحديده داخل صندوق المجلدات.



☒ نقرتين فوق صندوق الملفات ثم

نكتب:

**Label5 = Dir1 + "\" + File1**

ويعني عرض المسار الكامل للملف في Label5

يمكنك تحديد نوع الملفات التي تريده أن يظهر في صندوق عرض الملفات وذلك عن طريق خاصية Pattern ولعرض

كافة التنسيقات أكتب مايلي: \*.\* وعند الرغبة في التحديد يمكن الكتابة على الشكل التالي: \*.doc;\*.exe\* شغل البرنامج لأن وتأكد من النتائج. كما في الشكل.

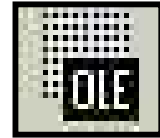
### ربط قاعدة بيانات



هنا بدأ الجد ولا أقصد أن هذا صعب ولكنه ليس سهل أيضاً حيث يجب أن نكون دقيقين جداً في تطبيق هذه الأداة فهي التي تقوم بربط قاعدة البيانات الخاصة بالبرنامج ويتم تسجيل كافة المعطيات فيها كما يمكن تسجيل كافة النتائج عليها.

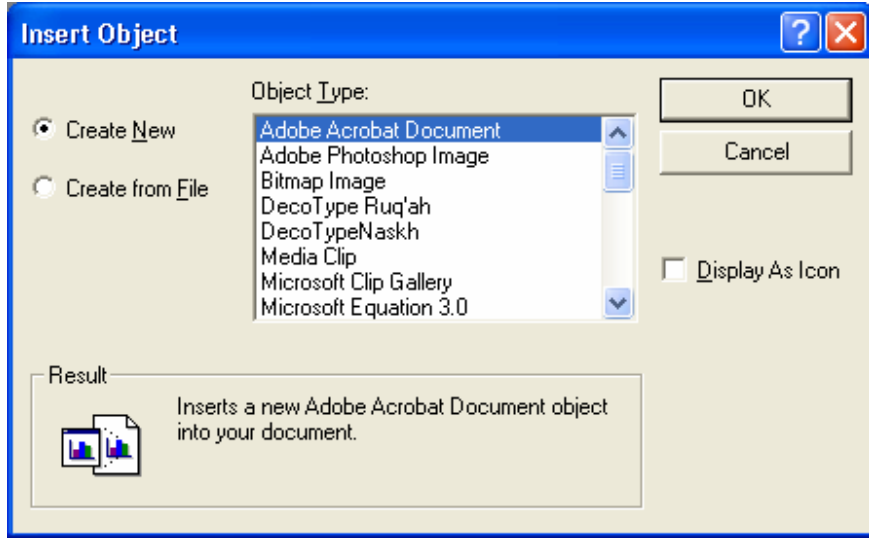
ويشكل عام يتعامل معظم المبرمجين لهذه اللغة مع قواعد بيانات الأكسس وسوف أتطرق هنا إلى ناحية بسيطة من هذا البرنامج الضخم لغرض إنشاء قاعدة بيانات مبسطة يمكن استعمالها في مثالنا القادم.

### عنصر تحكم Ole



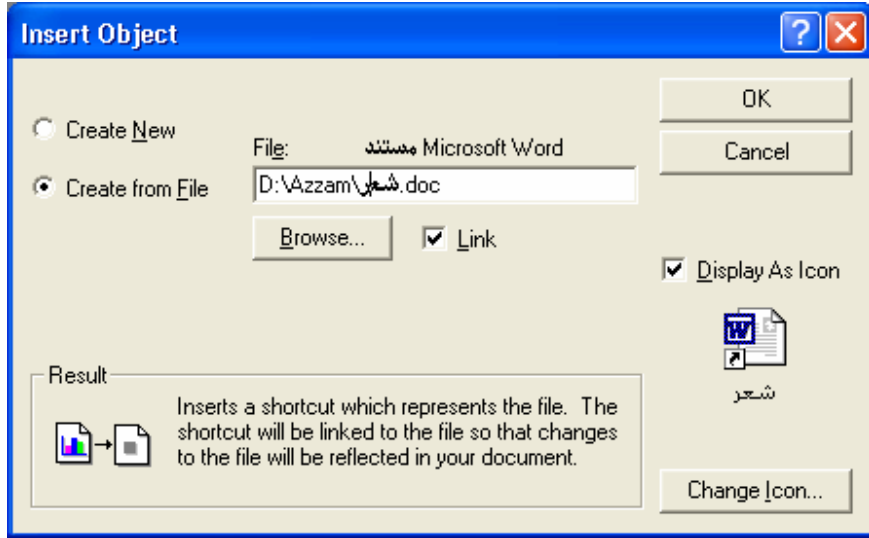
يستعمل الكائن OLE لإدراج أي تطبيق في برنامجك كإدراج صورة نقطية يمكنك التحرير والرسم فيها من داخل برنامجك عن طريق دمج الرسام معه. ويمكنك دمج الكثير من العناصر كملفات Word و Access و أي تطبيق آخر. لنأخذ مثال ملف Word انقر أداة التحكم OLE داخل الفورم بعد إفلات زر الفارة يظهر مربع حوار إدراج كائن Insert Object يعرض قائمة الكائنات التي يمكن ربطها في النموذج.





انقر زر خيار إنشاء من ملف Create From File في مربع حوار إدراج كائن Insert Object يظهر مربع صغير فيه مسار كائن لإضافة ملف موجود إلى البرنامج انقر مربع الاختيار ارتباط Link لإنشاء كائن مربوط في برنامجك وعندما تنقر مربع الاختيار ارتباط سيتم تحميل صورة من الملف الذي اخترته في كائن OLE انقر زر استعراض Browse للبحث عن ملف Word الذي ستربطه ثم بعد تحديده اختر زر Display As Icon.

والآن عند العودة إلى برنامجك وتشغيله ستتمكن من مشاهدة الملف فور الضغط على الرمز.

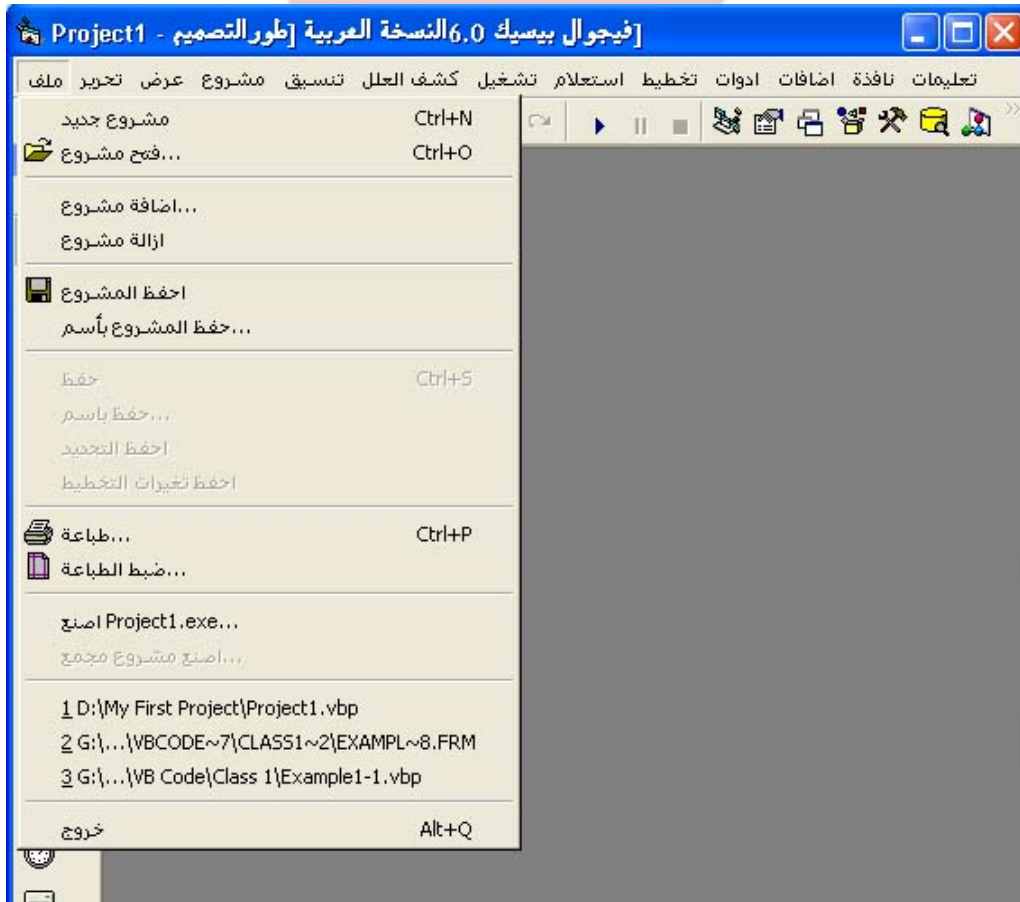


والآن نكون قد انتهينا من شرح الأدوات الأساسية في الفيچوال بيسك.

## أشرطة القوائم:

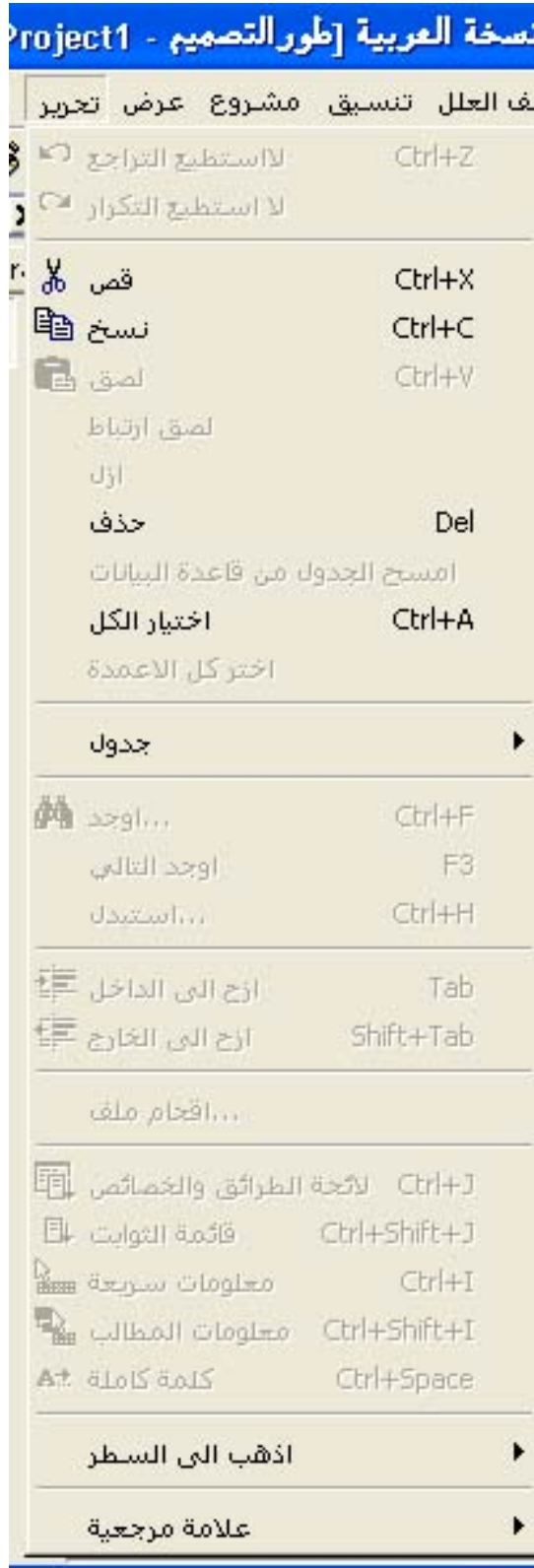
سنتحدث الآن عن القوائم التي تحتوي على أوامر هامة أيضاً.. ولكنني سأعرض هنا القوائم باللغة العربية لسهولة الشرح والبنود التي تتطلب شرح أو تفصيل سيتم شرحها في الوقت المناسب وعند تطبيقها.

**قائمة File ملف:** تحتوي هذه القائمة على أوامر أساسية خاصة للمشاريع بشكل عام، كإنشاء مشروع جديد، حفظ محتويات المشروع، طباعة محتويات المشروع وترجمة المشروع وتحويله إلى ملف ثنائي Binary الميزة التي أضيفت لـ VB6 هي إمكانية فتح أكثر من مشروع في نسخة واحدة من البيئة، وهي ميزة تعرف بالمشاريع المتعددة Multiple Projects.



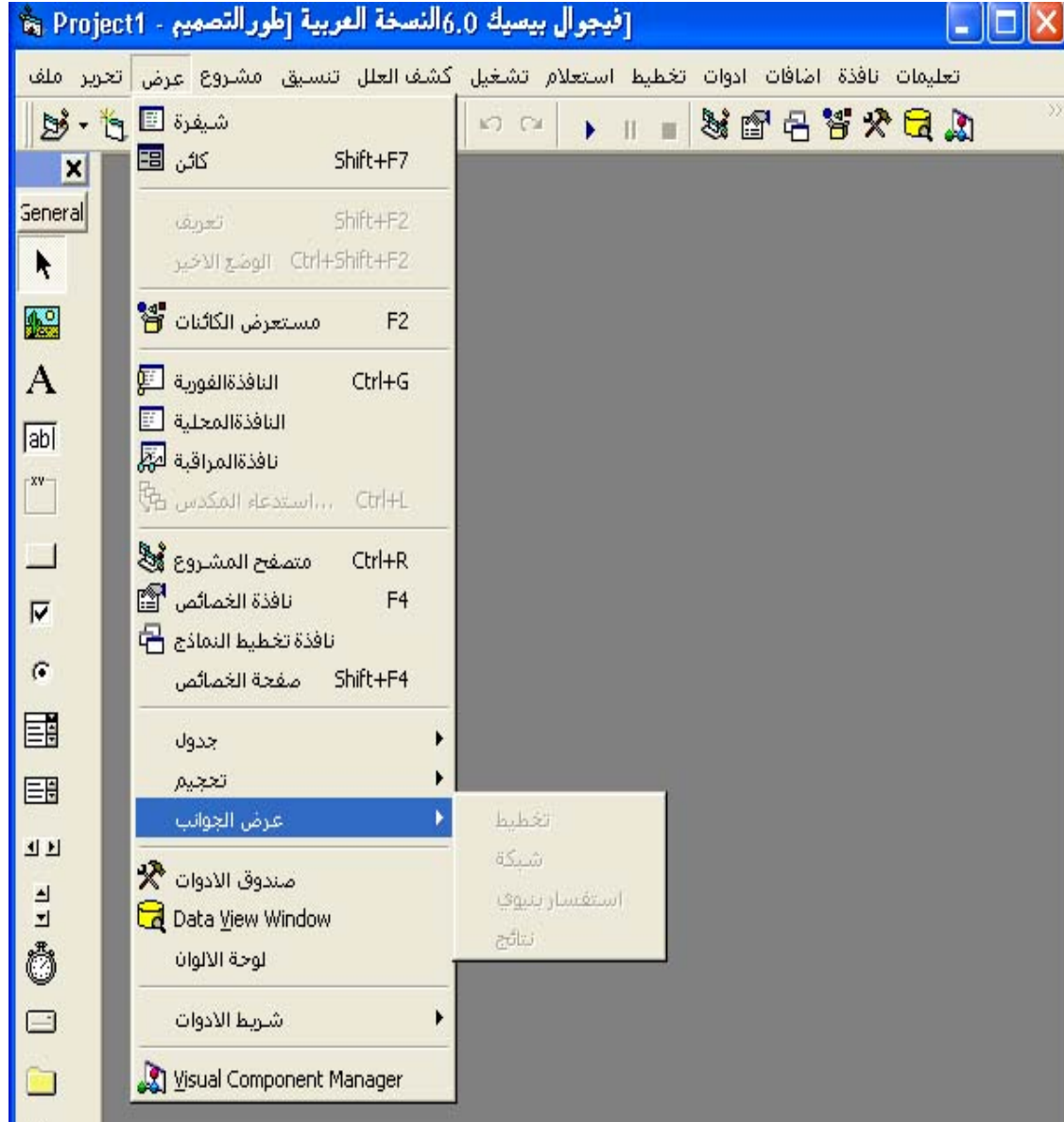
## القائمة تحرير Edit:

تحتوي هذه القائمة على أوامر التحرير القياسية كالقصاص، النسخ واللصق . بالإضافة إلى أوامر خاصة بقواعد البيانات في حالة وجود قاعدة بيانات في نافذة عرض البيانات Data View ومعظم الأوامر الواردة في أسفل هذه القائمة تستخدمها مع نافذة محرر الأكواد Code Window.



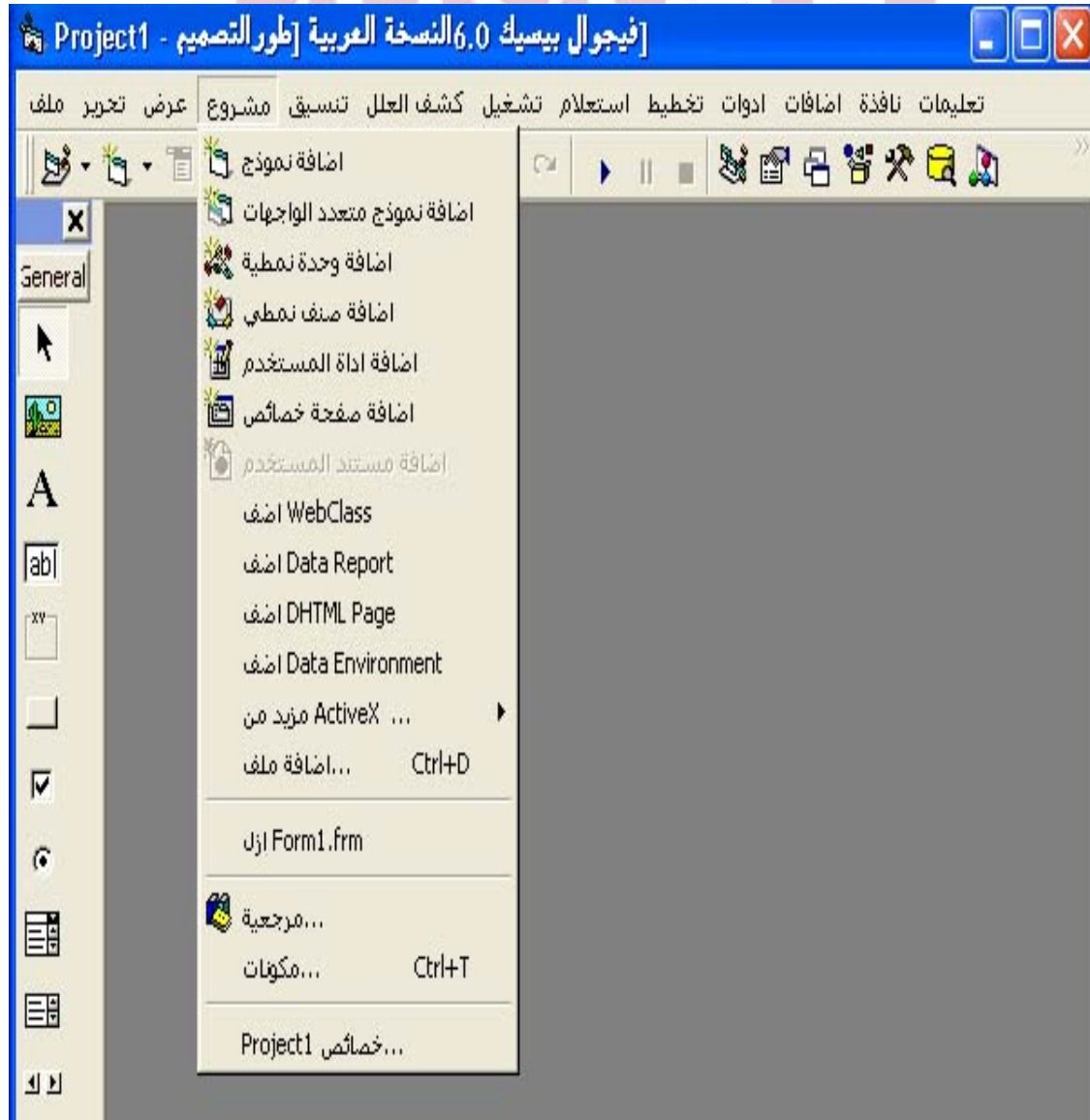
## القائمة عرض View:

معظم محتوياتها سهلة وهذه القائمة العربية منها:



## القائمة مشروع Project:

معظم أوامرها خاصة بمحتويات المشاريع، فهي تمكنك من إضافة عنصر أو عناصر من عناصر المشروع كنوافذ النماذج Forms - ملفات البرمجة Module . فئات Classes، كما يمكنك إضافة أدوات تحكم ActiveX Controls إضافية عن طريق الأمر Components أو تضمين مكتبات ActiveX DLL خارجية عن طريق الأمر References.



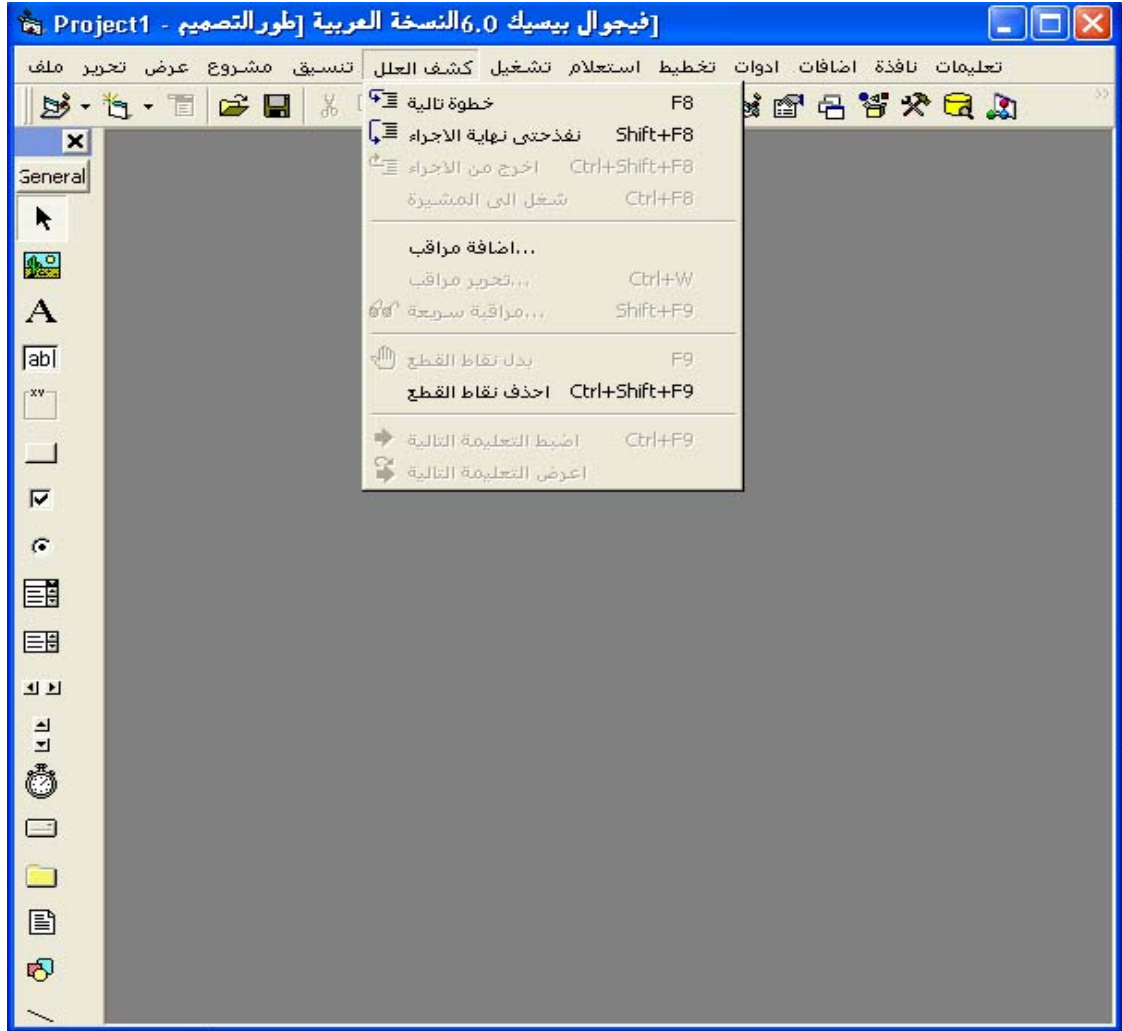
## القائمة تنسيق Format:

الأوامر الموجودة في هذه القائمة خاصة بتنسيق الأدوات التي تضعها على نافذة النموذج من ناحية موقعها على النافذة، فتوجد أوامر مرنة توفر عليك جهد محاذاة الأدوات أو توسيطها على النافذة، بالإضافة إلى تغيير ترتيب ظهور الأدوات أي وضع أداة فوق الكل أو أداة خلف الكل . الأمر الأخير Lock Controls تستخدمه إذا كنت تود منع نفسك من تغيير أحجامها أو مواقعها عن طريق الخطأ، هذا القفل تستطيع فتحه بكل بساطة باختيار نفس الأمر مرة أخرى.



## قائمة كشف العطل Debug:

معظم أوامر التنقيح وضعت اسفل هذه القائمة . من هذه الأوامر اختيار طريقة تنفيذ البرنامج، كتنفيذ سطر واحد منه Step Into إجراء كامل Step Over أمر سابق Step Out أو التنفيذ حتى الوصول إلى السطر الذي يوجد عليه مؤشر الكتابة Run to Cursor وبالنسبة لنقاط القطع BreakPoints فهي علامات تظهر مبدئياً باللون الأحمر على سطر معين بحيث تتم عملية الإيقاف المؤقت للبرنامج عند الوصول إلى هذه العلامات.



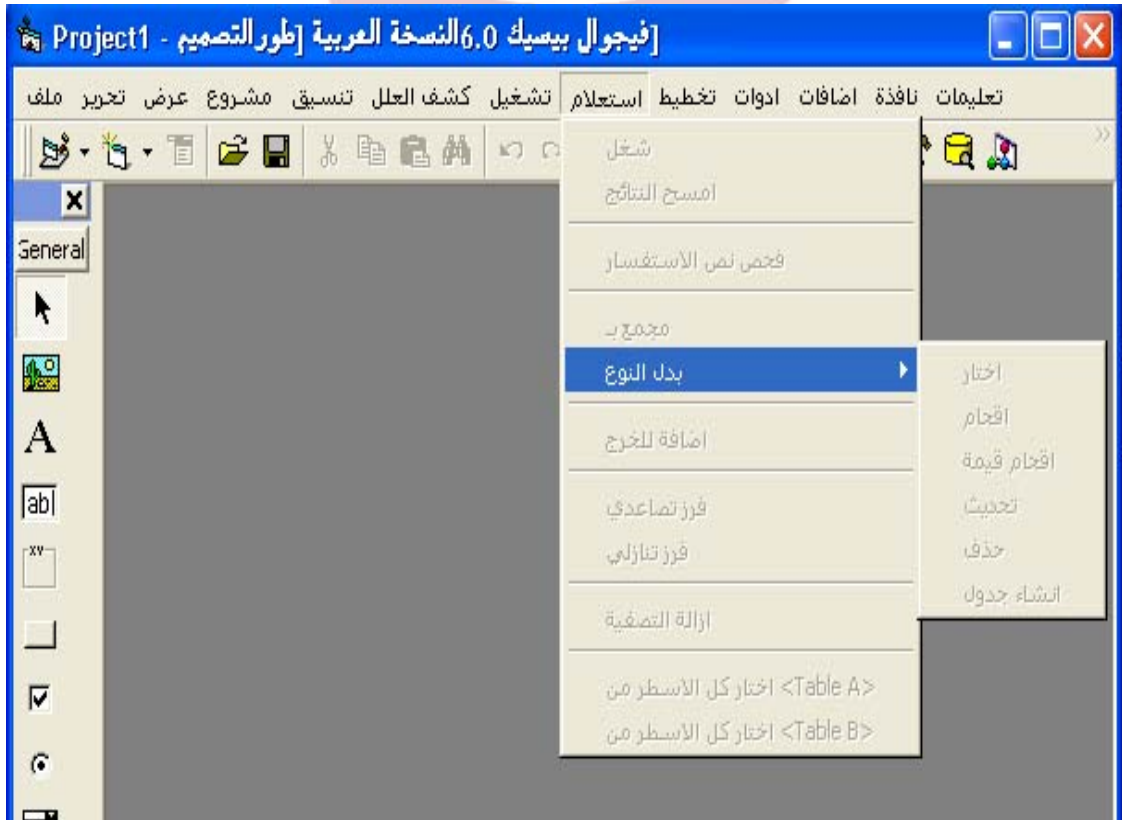


### القائمة تشغيل Run:

تستطيع تشغيل البرنامج عن طريق هذه النافذة البسيطة وتمكنك من اختيار الأوامر الأخرى كالإيقاف المؤقت Break. أو إنهاء عملية تنفيذ البرنامج End . بالنسبة للأمر Start with Full Compile هو مشابه لأمر التنفيذ Start ولن تحتاجه إلا في حالات نادرة ستجدها لاحقاً في هذا الكتاب.

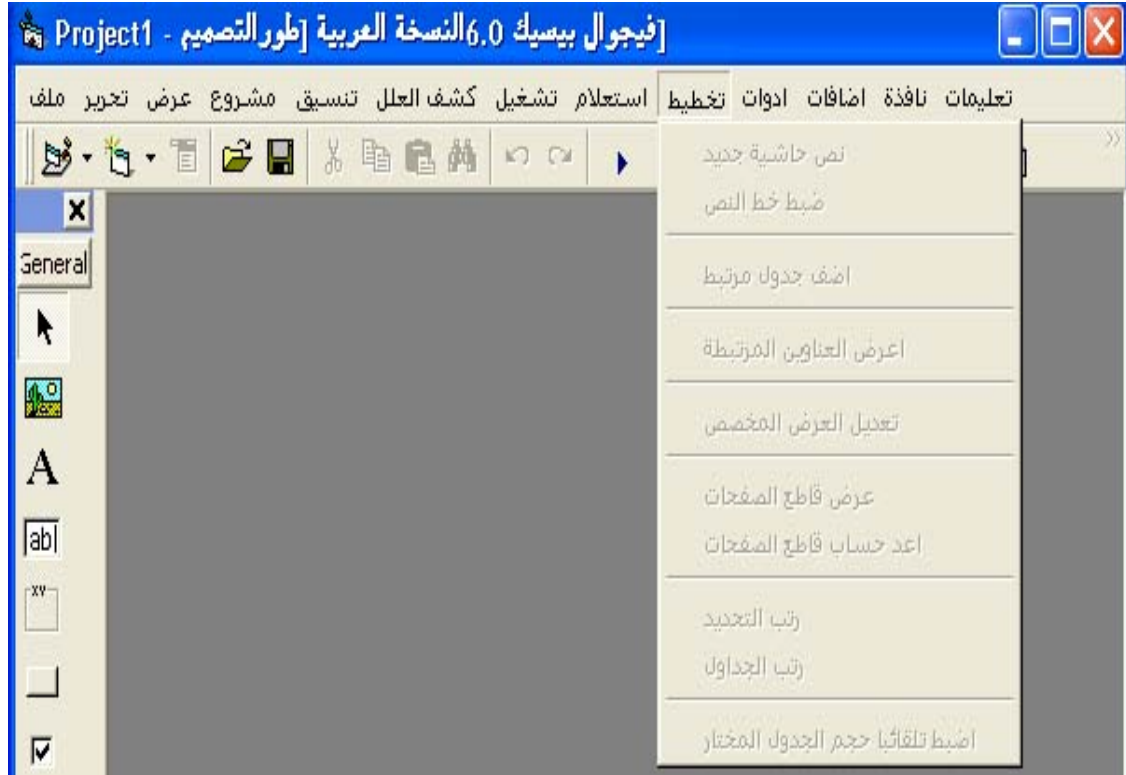
### القائمة استعلام Query:

هذه القائمة جديدة وللإصدار السادس VB6. وأوامر هذه القائمة غير ممكنة حتى تنشئ جملة استعلام SQL باستخدام الأداة Microsoft Query Builder.



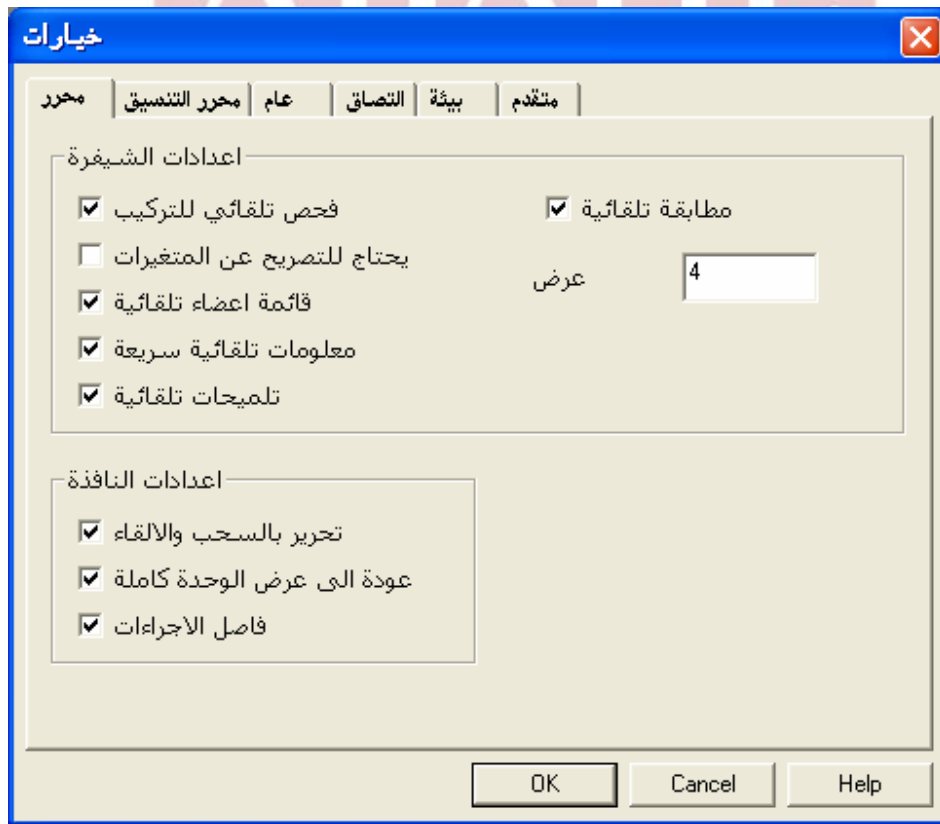
## القائمة تخطيط Diagram:

هذه قائمة أيضاً جديدة وأوامر هذه القائمة غير ممكنة إلا في حالة تعاملك مع قاعدة بيانات SQL Server أو ORACLE.



## القائمة أدوات Tools:

تحتوي على أوامر مختلفة التصانيف كمحرر القوائم Menu Editor ومسهل كتابة الإجراءات Add Procedure وغيرها... إذا أدت تخصيص بيئة التطوير المتكاملة IDE فالأمر Options يمكنك من الوصول إلى صندوق الحوار Options الذي يوفر لك عشرات الخيارات والخاصة بتغيير إعدادات بيئة التطوير IDE.



## القائمة إضافات Add-Ins:

قد تعاملنا مع هذه القائمة عندما قمنا بتصميم قاعدة البيانات والأوامر الموجودة فيها عبارة عن برامج مستقلة تسمى الإضافات Add-Ins هدفها توفير خدمات إضافية لبيئة التطوير تزيد من مرونتها.

## قواعد البيانات

المشروع الأول: تصميم برنامج ذاتية موظفين بسيط.

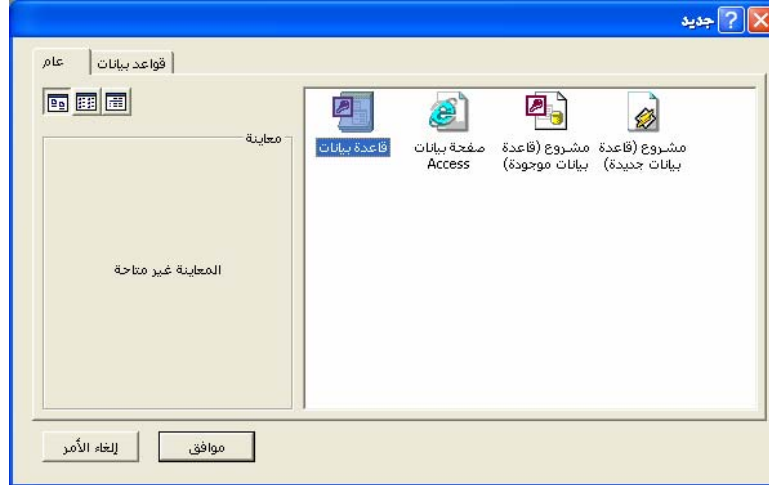
خطوات العمل:

أنشئ مجلداً وسمه My First Project حيث سنستعمله لحفظ كافة الملفات الخاصة بهذا المشروع.

لنبدأ الآن بتصميم قاعدة بيانات أكسس وذلك بطريقتين السبب في ذلك هو اختلاف الإصدارات الخاصة بكلا البرنامجين فالإصدارات الحديثة من فيجوال بيسك تتقبل تقريباً كافة إصدارات أكسس والعكس صحيح أنا شخصياً أفضل أن تقوم بتصميم قاعدة بياناتك من خلال برنامج فيجوال بيسك في هذه المرحلة على الأقل خصوصاً إذا كنت غير خبير في الأكسس إلا أنني سأقوم بشرح الطريقتين للفائدة.

## الطريقة الأولى لإنشاء قاعدة بيانات:

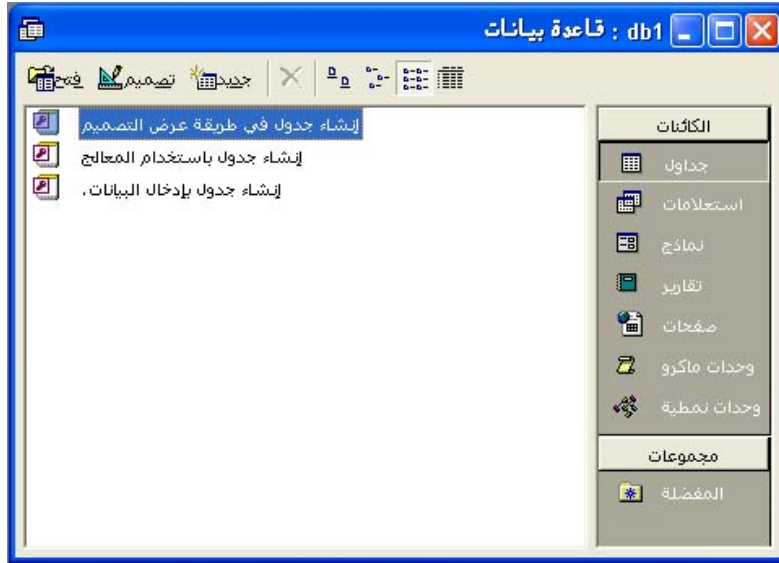
✘ بعد تشغيل برنامج الأكسس نختار قاعدة بيانات جديدة حسب الشكل أدناه.



بعد الضغط على قاعدة بيانات تظهر لنا لائحة لنحدد فيها المسار الخاص الذي سنحفظ به هذه القاعدة وأعود أذكر بأننا أنشأنا مجلد خاص للعمل.



✘ فور الحفظ ستظهر لنا خيارات وأدوات إنشاء قاعدة البيانات على الشكل التالي:



نختار إنشاء جدول في طريقة عرض التصميم، ثم نكتب أسماء الحقول التي نرغب بتضمينها في برنامجنا...

وهنا (في هذا المثال) سننشئ 4 حقول فقط وهي:

الاسم . العمل . العمر . الهاتف

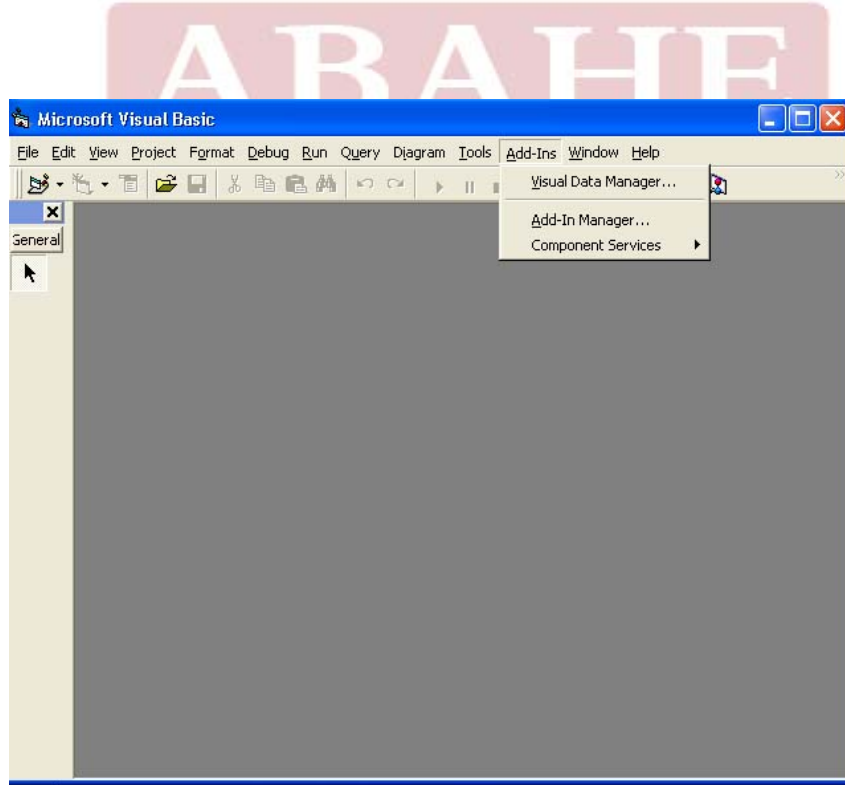
وذلك على الشكل التالي:



عند إغلاق هذه اللائحة ستظهر رسالة تطلب منا حفظ الجدول (احفظه باسم job)

الطريقة الثانية لإنشاء قاعدة بيانات:

وذلك من خلال برنامج الفيجوال بيسك نفسه أرجو إتباع هذه الخطوات بدقة.  
✕ نختار من قائمة Add-Ins الخيار Visual Data Manager.



عند ضغط هذا الخيار ستظهر لنا الشاشة التالية:



الشكل السابق واضح وقد تعدت أن أجعل اللوائح باللغة العربية لسهولة الشرح والفهم.



☒ ما عليك في هذه المرحلة سوى تطبيق الخيارات كما في الشكل السابق وسوف يظهر لديك مربع حوار لاختيار اسم لقاعدة البيانات الخاصة بالأكسس.

نسميها Jobs ونحفظها في نفس المجلد الذي أنشأناه سابقاً.

☒ بعد حفظ قاعدة البيانات تبدأ هنا مرحلة إنشاء الجداول داخلها وهي مرحلة دقيقة إلا أنها سهلة وسلسة إذا ما توخينا الدقة. فور الحفظ ستظهر لنا اللاتحة أدناه وهي خاصة بإنشاء قاعدة البيانات نختار منها جدول جديد حسب الشكل:





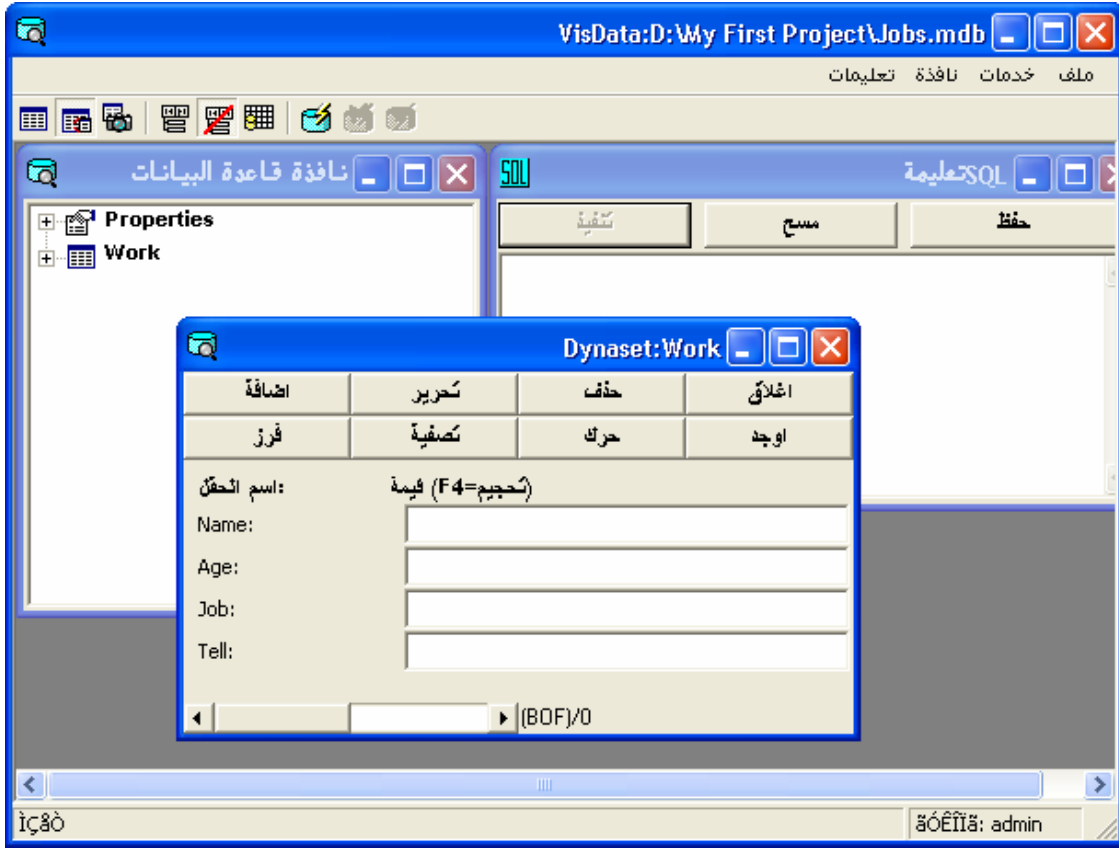
✘ فور الضغط على جدول جديد ستظهر لنا لائحة بناء جدول حسب الشكل أدناه نقوم هنا بوضع اسم للجدول في الخانة المخصصة لذلك.

✘ بعد تحديد اسم الجدول نضغط على إضافة حقل.

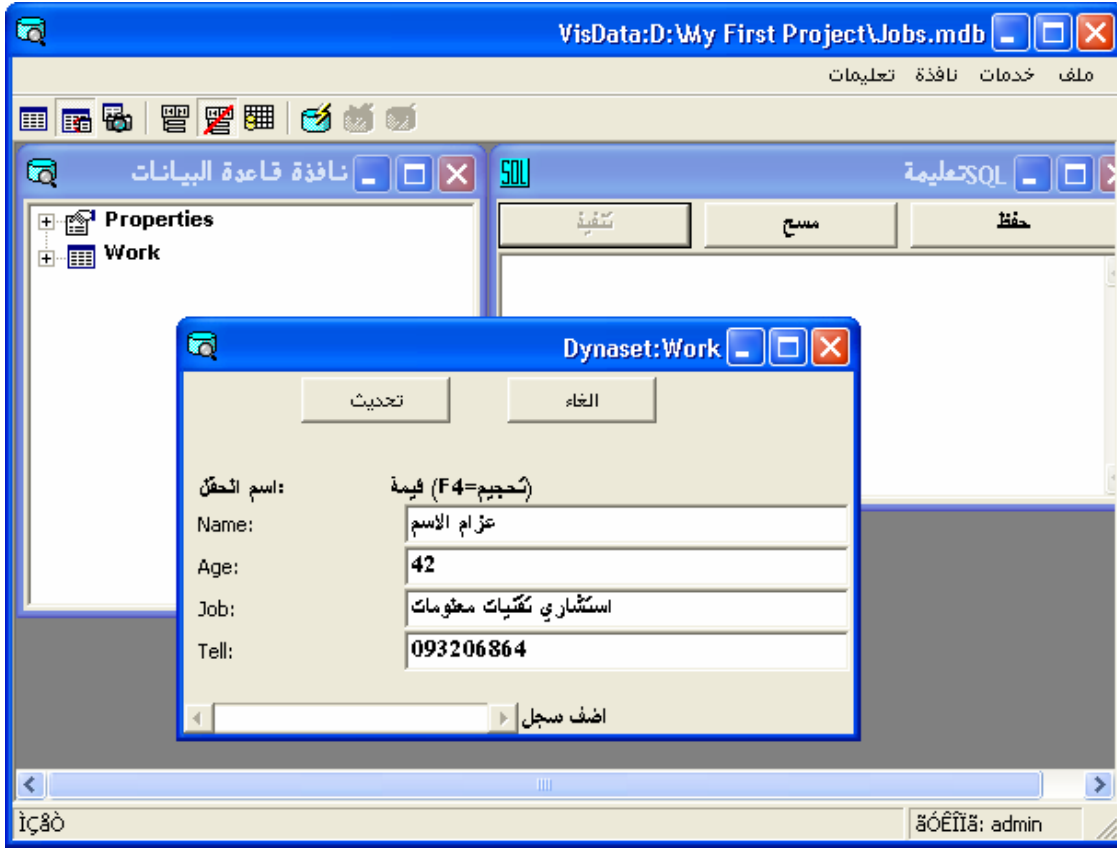
✘ نكتب اسم الحقل كما نريده وليكن Name ثم نضغط موافق. في هذه المرحلة يستحسن ترك الخيارات الخاصة بالحقل كما هي.

✘ تذكر أننا أنشأنا سابقاً أربعة حقول هي الاسم . العمل . العمر . الهاتف. نقوم هنا بإنشائها كما في المرحلة السابقة.

✘ بعد إتمام إنشاء الحقول نضغط إغلاق فنعود إلى اللائحة السابقة ونضغط إضافة فهرس/جدول (Build The Table). فتظهر لنا اللائحة أدناه وعند الضغط على جدول Work سيظهر لنا الشكل Dynaset:Work. والذي نستطيع من خلاله إضافة البيانات للحقول بالضغط على زر إضافة.



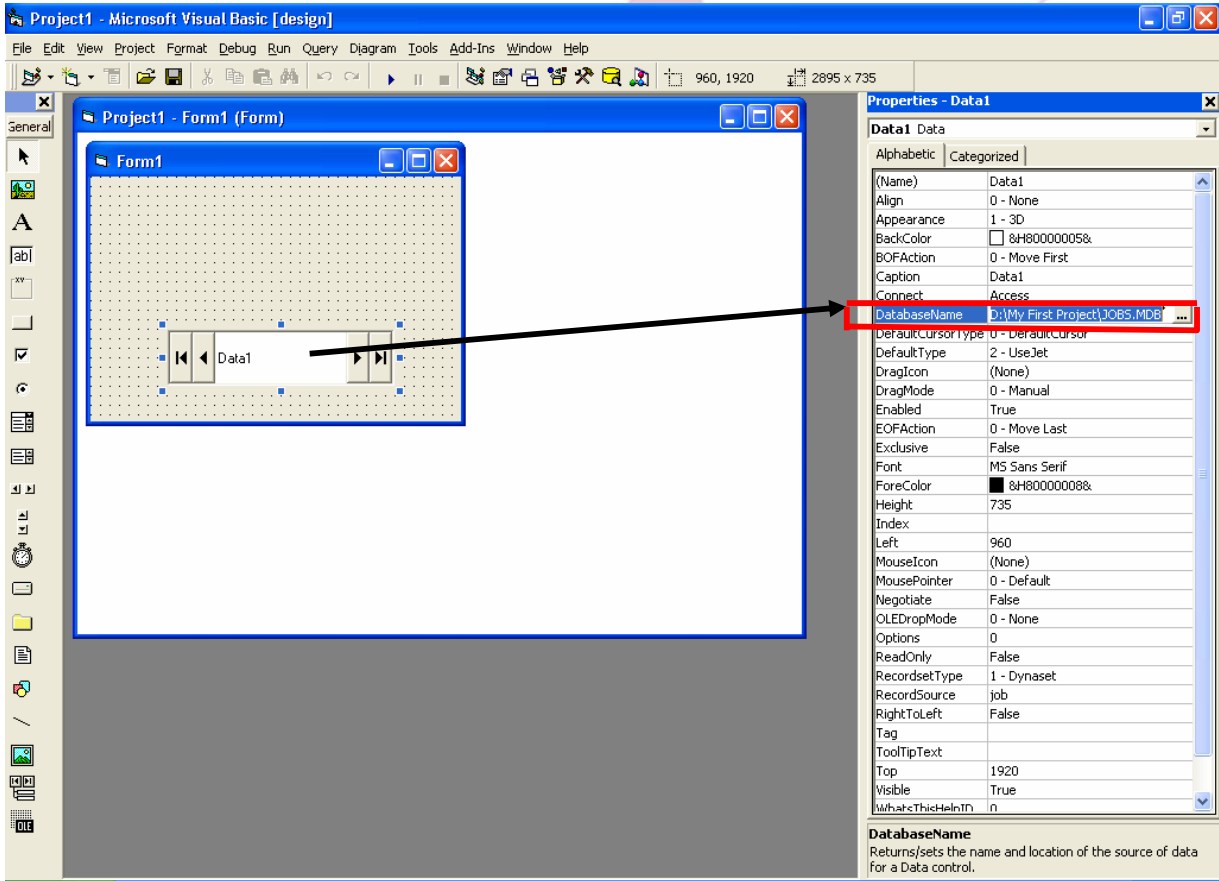
✘ وفور الضغط على إضافة ستظهر لنا لائحة فارغة يمكننا من خلالها تعبئة البيانات المطلوبة.



- ✘ بعد الانتهاء نضغط تحديث لتثبيت هذه البيانات.
- ✘ الآن قاعدة البيانات جاهزة، نغلق كافة الإطارات ونعود إلى البرنامج لننشئ فورم جديد ونبدأ العمل.

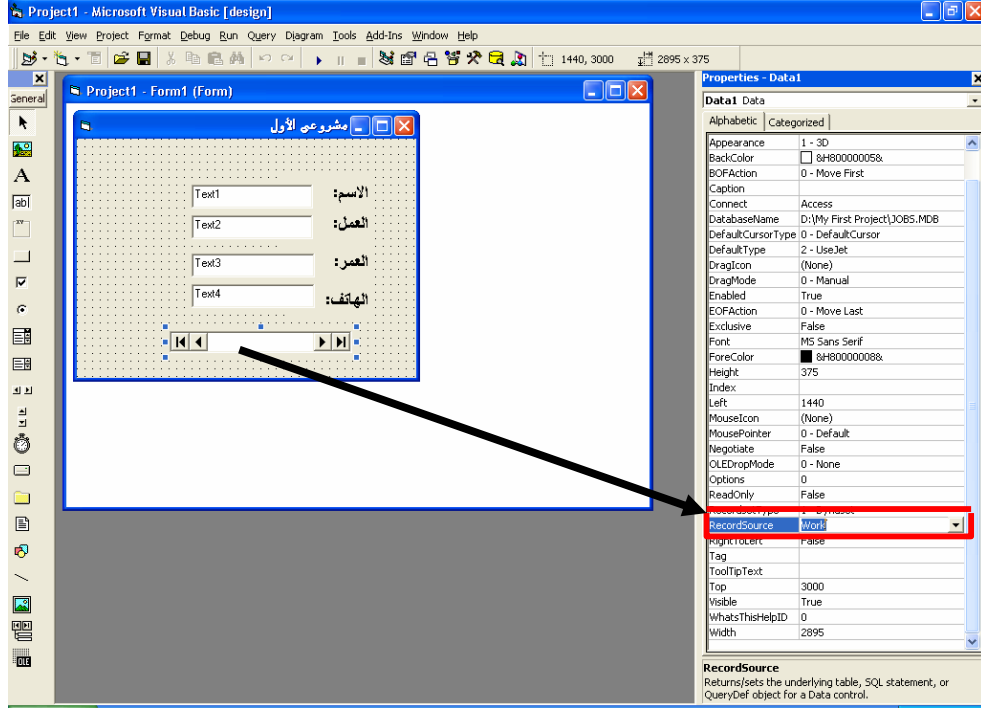
## إعداد برنامجي الأول

☒ نعود الآن إلى برنامج الفيچوال بيسك ونفتح مشروع جديد ثم نضع أداة ربط قاعدة بيانات داخل الفورم ومن الخصائص نختار DATABASE NAME ونحدد مكان قاعدة البيانات (ولا يوجد اختلاف في تعريف قاعدة البيانات في حال استخدمت أي واحدة من طرق الإنشاء السابقة).  
لاحظ الشكل أدناه...

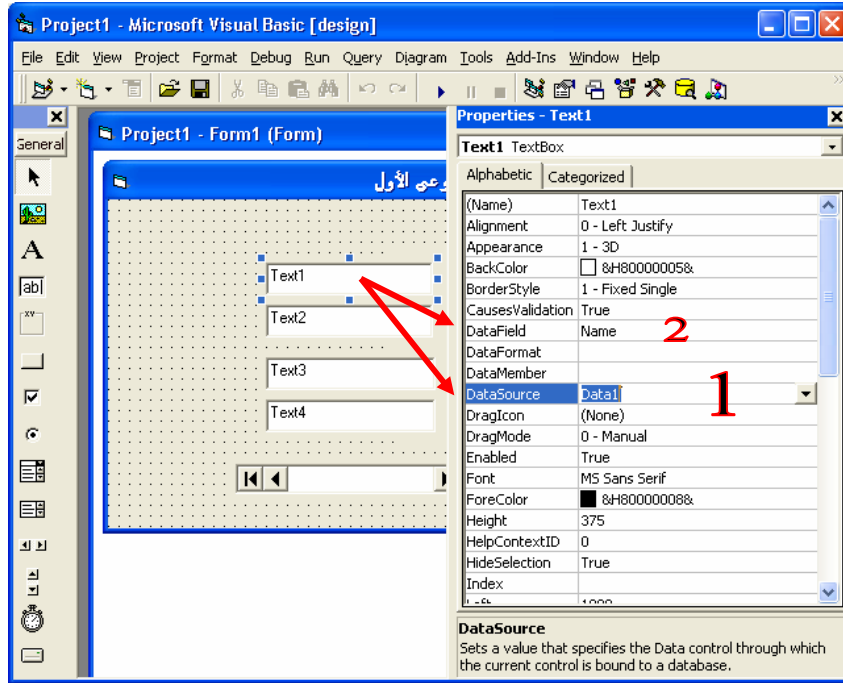


هنا نكون قد عرفنا قاعدة البيانات ولكن يبقى أن نحدد الجدول الذي سنتعامل معه (طبعاً من الممكن أن يكون هناك أكثر من جدول داخل قاعدة البيانات)

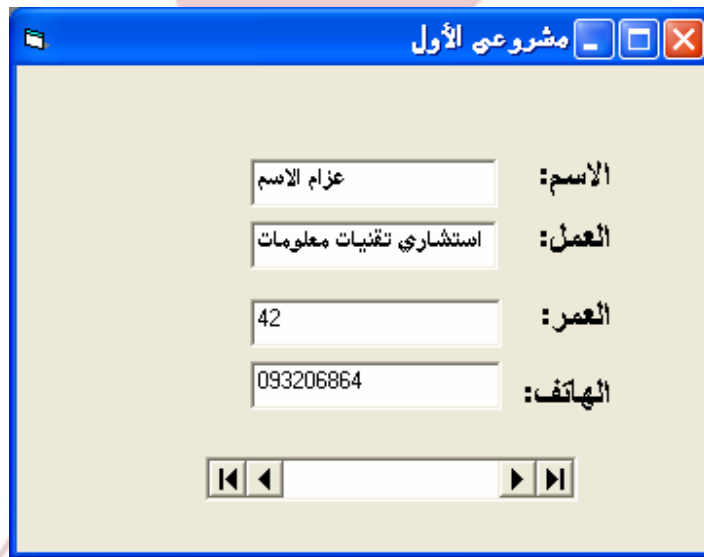
✘ لتحديد الجدول المطلوب العمل عليه نختار من الخصائص Record Source  
ومن ثم اسم الجدول المطلوب. ومن ثم ننشئ 4 محررات نصوص TextBox  
و 4 Label. حسب الشكل أدناه.



✘ نبدأ الآن بربط كل مربع نص مع الحقل المناسب له في قاعدة البيانات وذلك من  
خلال الخاصية **DataSource** حيث نقوم من خلالها بالتعارف بين قاعدة  
البيانات ومربع النص ومن ثم الخيار **data field** في خصائص مربع النص  
لتحديد الحقل المناسب على النحو التالي:



✘ الآن أصبح برنامجك جاهز تقريباً شغل البرنامج لاحظ النتائج



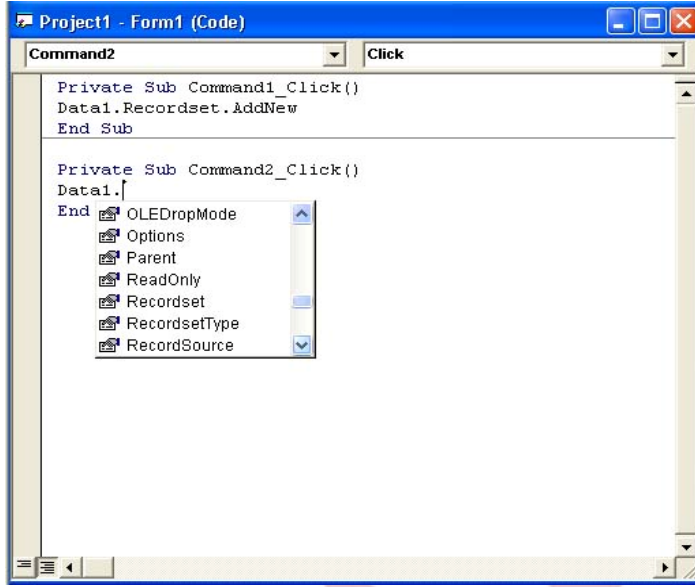
✘ لاحظ أنك الآن تستعرض ما سبق وأدخلته أما عن إدخالات جديدة فيجب أن

نضيف أمر خاص لذلك وذلك من خلال زر الأوامر.

✘ أدخل زر أوامر إلى الفورم ومن ثم عدل اسمه إلى سجل جديد ثم اضغط مرتين

عليه للوصول إلى محرر الصيغة وكتب الصيغة التالية:

**Data1.Recordset.AddNew**



```
Private Sub Command1_Click()  
Data1.Recordset.AddNew  
End Sub  
  
Private Sub Command2_Click()  
Data1.[  
End
```

- OLEDropMode
- Options
- Parent
- ReadOnly
- Recordset
- RecordsetType
- RecordSource

إذا كنت تخشى من الأخطاء  
الإملائية أو الأخطاء في  
الصيغة اكتب Data1. ثم لاحظ  
ماذا سيحدث.....

ستظهر لك قائمة تتضمن  
كافة الأوامر التي يمكنك  
إضافتها وفور الضغط على  
الأمر المطلوب سيثبت في  
صفحة الشيفرة.

وبعد أن تضع . ستظهر لك هذه القائمة لاختيار جديد....



✘ بعد تثبيت الصيغة السابقة أنت بحاجة إلى زر لتسجيل هذه البيانات ويمكن ذلك باتباع الخطوات السابقة الخاصة بزر السجل الجديد إلا أن الصيغة هنا هي: Data1.Recordset.Update.

✘ كما يمكنك إضافة عد أزرار تراها مناسبة وبنفس الطريقة. مثال: زر حذف سجل الصيغة هي:

Data1.Recordset.Delete  
Data1.Refresh

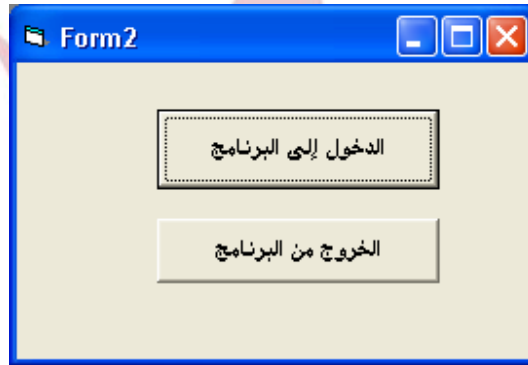
✘ الآن يمكنك إدخال البيانات وحفظها ومتابعتها أيضاً من خلال الأسهم.

بهذا نكون قد أنهينا جزء بسيط من التعامل مع قواعد البيانات وسنتعرض في الأجزاء القادمة إلى أمور عديدة تتعلق بقواعد البيانات وطرق البحث و....  
والآن بعد أن صممت برنامجاً على الفيجوال بيسك فإنك سوف تحفظه ولكن الأهم هو تحويله إلى ملف تنفيذي EXE وطبعاً يمكنك ذلك فبعد أن حفظنا المشروع (البرنامج) باسم project1 اذهب قائمة File ثم اختر منها make project1.exe.  
الآن اذهب للمسار الذي كنت حفظت فيه البرنامج تراه قد عمل أيقونة لبرنامجك أوتوماتيكياً وعند النقر على الأيقونة يفتح برنامجك الرائع.

## التعامل مع الفورم:

حتى نتمكن من شرح هذا التعامل الذي هو هام جداً في التطبيقات البرمجية. دعوني أوضح لكم في البداية كيف يمكننا ربط أكثر من نموذج (فورم) مع بعضهما البعض من خلال زر الأوامر..

لنعود الآن إلى برنامجنا السابق... My First Project. اطلب البرنامج داخل الفيچوال بيسك. وأنشئ فيه نموذج جديد من خلال قائمة المشروع. إضافة نموذج. ليصبح على الشكل التالي:



نكتب الشيفرة أدناه للزر الأول:  
Unload Me لإغلاق نفسه  
Form1.Show لطلب النموذج  
وللزر التالي نكتب:  
End أي خروج من البرنامج

وبذلك نكون قد أعدنا زرین واحد لطلب برنامجنا والثاني للخروج من البرنامج:

في النموذج القديم أضف زر جديد على الشكل التالي:

ونضع الشيفرة التالية على الزر الجديد:

Unload Me لإغلاق نفسه

Form2.Show لطلب النموذج

ولكن تبقى الآن مشكلة وهي عند تشغيل البرنامج لن يظهر لنا النموذج الجديد في بدء التشغيل... فما العمل...!!؟؟

الحل بسيط من قائمة مشروع نختار خصائص فتظهر لنا اللائحة التالية:

من هذه اللائحة يمكننا تغيير النموذج الذي سيبدأ به البرنامج وفي مثالنا سنجعله  
.Form2

شغل الآن البرنامج وشاهد ما حدث....



## البحث في الفيچوال بيسك:

قد لا يكون هناك أي أهمية لأي برنامج لا نستطيع من خلاله القيام بعمليات البحث عن معلومات معينة، حتى أننا لا يمكن أن نعتبر أي برنامج جيد دون وجود هذه الميزة... وبالطبع فالفيچوال بيسك لا تتقصه هذه الميزة.

ولنحاول الآن إضافة بعض الميزات لبرنامجنا السابق من خلال إضافة بعض أزرار الأوامر الهامة له إضافة إلى ميزة البحث... هل أنت جاهز... لنبدأ العمل..

جهاز نموذجك على الشكل التالي:

كما تلاحظ فقد أضفنا بعض الأزرار والأدوات...

Combo1 : لقد تعرفنا على هذه الأداة سابقاً. وسنقوم هنا بوضع عناصر داخلها وذلك حتى نتيح لنا خيارات البحث مثل البحث حسب الاسم أو حسب العمل أو العمر أو رقم الهاتف... وذلك حسب برنامجنا موضوع العمل..

### الهدف من المثال وطريقة العمل:

الهدف واضح وهو إمكانية إجراء البحث، أما الآلية فسوف تتم على النحو التالي:

- من Combo1 سنضيف تصنيف البحث حسب...

- في Text5 سنكتب القيمة المطلوب البحث عنها، والتي تتوافق مع تصنيف البحث..
- نضغط زر بحث فتظهر النتائج (إن وجدت في عناصر النموذج). وفي حال وجود قيم متطابقة سوف يمكننا مشاهدتها من خلال الضغط على أسهم الشريط .Data1

والآن إلى العمل...

ونضيف الكود التالي على النموذج حتى يتمكن من تحميله عند التشغيل أي في Load

```
Combo1.AddItem "بحث عن الاسم"
Combo1.AddItem "بحث عن العمل"
Combo1.AddItem "بحث عن العمر"
Combo1.AddItem "بحث عن الهاتف"
On Error GoTo data_error
Data1.DatabaseName = App.Path & "\JOBS.mdb"
Data1.Refresh
Data1.Recordset.AddNew
Exit Sub
data_error:
MsgBox "لم يتم العثور على قاعدة البيانات", vbCritical,
```

لاحظ أنني قمت هنا بوضع الجمل التي ستظهر داخل Combo1 ضمن " ".  
أظن هذا الكود ليس بحاجة إلى شرح فهو سيقوم بتوليد القائمة داخل Combo1.  
على الشكل التالي:

أما بالنسبة لباقي الأزرار فالكود الخاص بها ليس صعباً عليك...  
ولكني سأشرح الكود الخاص بزر البحث:

```
Select Case Combo1.ListIndex
Case 0
Data1.RecordSource = (" select * from work where name=" + "" + Text5.Text + """)
Data1.Refresh
Case 1
Data1.RecordSource = (" select * from work where job=" + "" + Text5.Text + """)
Data1.Refresh
Case 2
Data1.RecordSource = (" select * from work where age=" + "" + Text5.Text + """)
Data1.Refresh
Case 3
Data1.RecordSource = (" select * from work where tell=" + "" + Text5.Text + """)
Data1.Refresh
End Select
```

لاحظ هنا أننا ضمنا اسم الجدول Work في الشيفرة... ولكن ماذا تعني...؟؟  
تعني: البحث في الحقل داخل الجدول عن أي بيانات متطابقة مع القيمة الموجودة في Text4.  
فالرمز \* يعني كافة الحقول.  
نفذ ما سبق بدقة وستظهر لك النتائج على النحو التالي:

مشروعى الأول

أزرار الأوامر

سجل جديد

التعديل

حذف

حفظ سجل

سجل المعلومات

الاسم: أمين القاسم

العمل: طالب

العمر: 18

الهاتف: 092300286

بحث حسب الاسم أمين القاسم

بحث


عودة للوحة الرئيسية

شريط العرض



## أدوات ومهارات

### إنشاء القوائم:

لا يمكننا الاستغناء بأي حال من الأحوال عن القوائم في أي برنامج على الرغم من إمكانية التعامل من خلال أزرار الأوامر (كما مر معنا) إلا أن للقوائم سحرها الخاص في البرنامج لما توفره من إمكانيات ومكان ضمن البرنامج. من شريط الأدوات نختار الزر  فتظهر لنا اللائحة التالية وهي المخصصة لتحرير القوائم:

وضعت لكم القائمة المعربة وذلك لفهم العبارات ووضوح التعامل معها. الآن سأشرح لكم باختصار محتويات هذه اللائحة. **العنوان:** ويكتب به اسم القائمة الرئيسية التي ستظهر في البرنامج **الاسم:** وهو الاسم الذي سيتعامل معه الفيجوال بيسك مع هذه اللائحة. **الاختصار:** ويمكنك هنا وضع مفتاح اختصار من لوحة المفاتيح للبند المحدد.

أما باقي الأزرار فأراها واضحة وأترك لك تجربتها...

والآن لننشئ قائمة لبرنامج ما...

عنوان القائمة الرئيسية ملف تحتوي على (جديد . فتح . بحث . خروج).

▪ اكتب اسم القائمة في العنوان (ولا تنسى بند الاسم). ثم اضغط التالي.

محرر القوائم

العنوان: ملف موافق

الاسم: menu1 الغاء

الدليل: الاختصار (None)


معرف المساعدة 0 وضع الانجاز 0 - None

إختيار  ممكنة  ورثية  قائمة النوافذ

← → ↑ ↓ التالي أقحم حذف

ملف

▪ نكتب اسم البند الأول في القائمة (وذلك بعد أن ضغطنا التالي). في مربع

العنوان وأيضاً الاسم... كما في السابق ولكن نضغط  بدل التالي:

- تطبيق المرحلة السابق على كافة البنود المطلوبة. فتظهر على الشكل التالي:

▪ نضغط موافق ونرى النتيجة...



نلاحظ في البرامج الاحترافية أو حتى العادية وجود مفاتيح اختصار بجانب البنود كما نلاحظ وجود إشارة \_ تحت حرف معين مما يسمح لنا بطلب ذلك الأمر من خلال الضغط على زر Alt+الحرف الذي تحته خط. وبالطبع يمكننا ذلك باستعمال خيار الاختصار من اللوحة السابقة وبوضع إشارة & قبل الحرف الذي نريد أن تظهر تحته الإشارة \_.

ويمكنك مراقبة الشكل أدناه لمعرفة ذلك.

محرور القوائم

العنوان: ملف

الاسم: menu1

الدليل: الاختصار (None)

معرفة المساعدة 0 وضع الانجاز 0 - None

إختيار  ممكنة  مرئية  قائمة النوافذ

التالي أفحم حذف

ملف  
جديد  
فتح  
بحث  
خروج

محرور القوائم

العنوان: جديد

الاسم: New

الدليل: الاختصار Ctrl+N

معرفة المساعدة 0 وضع الانجاز 0 - None

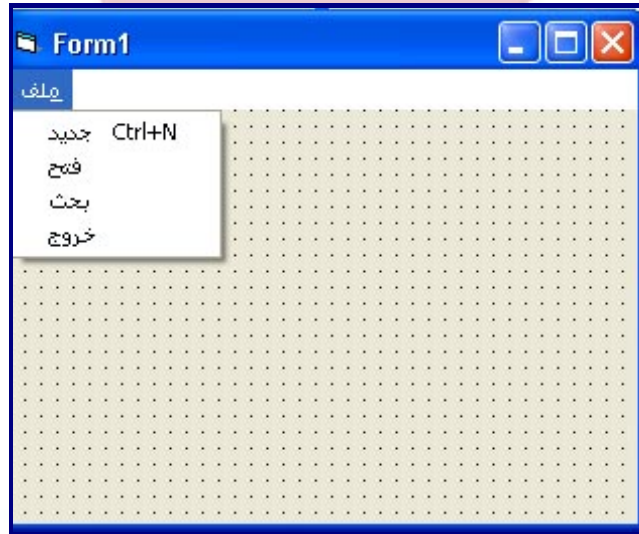
إختيار  ممكنة  مرئية  قائمة النوافذ

التالي أفحم حذف


ملف  
جديد Ctrl+N  
فتح  
بحث  
خروج

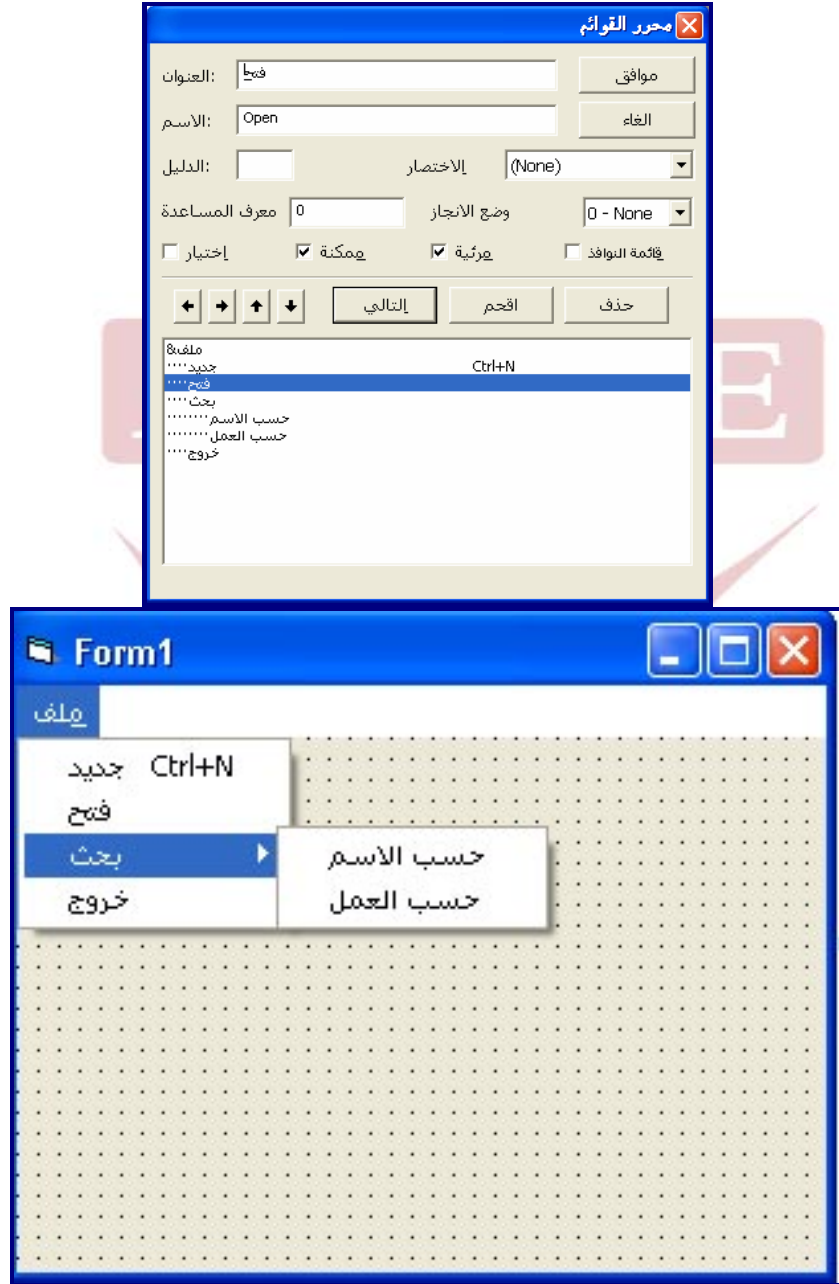
ABAHE

فتظهر النتائج على النحو التالي:



**القوائم الفرعية:**

وتكونها بسيط جداً كل ما عليك فعله هو الوقوف في البند الذي يلي البند المطلوب تكوين قائمة فرعية له ونضغط إقام ونسجل المعلومات ومن ثم  فنرى اختلاف موقع البند وحسب الشكل التالي:



إظهار القائمة بزر الماوس الأيمن:

ضع الشيفرة التالية في محرر شيفرة الفورم في الحدث Mouse Down للـ Form1:

```
Then If Button = 2  
menu1 Form1.PopupMenu  
End If
```

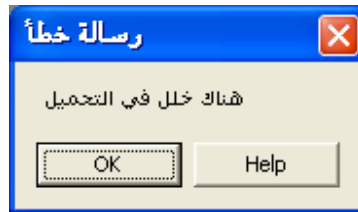
شغل البرنامج واضغط زر الماوس الأيمن وشاهد...



### إطارات الرسائل المختلفة:

الرسائل على اختلاف أنواعها تشكل لغة حوار وتفاهم بين المستثمر والبرنامج... وهي يجب أن تكون واضحة ودقيقة وفي المكان المناسب لها. الشيفرة بشكل عام وتوضع مباشرة في الفورم...

الكلمات باللون الأزرق خاصة بالمبرمج يمكنه تغييرها كما يريد... الشيفرة السابقة تنتج:



وتأخذ هنا هذه الإطارات الشكل العام من نظام Windows بينما نضمنها التفاصيل الخاصة التي نريد...



## العبارة الشرطية IF:

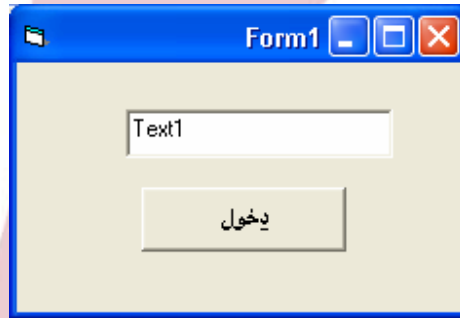
هذه العبارة هي من أساسيات البرمجة وقد تكون من أساسيات التعامل والحياة بشكل عام، فهناك دائماً شرط مهما كان بسيط لحدوث الأفعال وبالتالي فوجود شرط سيؤدي إلى نتيجتين الأولى سلبية إذا لم يتحقق هذا الشرط والأخرى إيجابية في حال تحققه...

وهنا الأمر كذلك...

ولنأخذ مثال لتوضيح الفكرة وأترك لك الخيال العلمي لاستعمال تطبيقات هذه العبارة في نواحي متعددة...

مثال:

أنشئ فورم يحتوي على مربع نص وزر أوامر على الشكل التالي:



الآن نضيف الصيغة التالية إلى زر الأوامر:

`If Text1 = "Ammar" Then Unload Me Else MsgBox "كلمة مرور خاطئة حاول مرة ثانية"`

المعنى: إذا Text1 يساوي القيمة Ammar اخرج من البرنامج وإذا كانت القيمة لا

تساوي Ammar أظهر رسالة الخطأ.

جرب الصيغة وتأكد من النتيجة...

## الطباعة:

إن الضروري في كل برنامج طباعة البيانات المدخلة أو النتائج وسوف نتعرض هنا إلى طباعة البيانات التي ندخلها على الفورم...  
وسنعود الآن إلى برنامجنا الأول ونحاول طباعة نتائج البحث...  
لنضيف زر أمر (طباعة النتائج).. حسب الشكل أدناه..

نضغط على زر (طباعة النتيجة) ونكتب الشيفرة التالية:

```
Printer.FontSize = 40  
Printer.Print Text1.Text  
Printer.Print Text2.Text  
Printer.Print Text3.Text  
Printer.Print Text4.Text  
Printer.EndDoc
```

السطر الأول لتحديد حجم الحرف، أما الأسطر التالية فهي طلب طباعة النص الموجود في مربعات النصوص ومن ثم أمر إغلاق الطباعة...  
جرب الآن تنفيذ البرنامج وأجري البحث ثم اضغط طباعة النتائج وشاهد النتيجة على الورق....

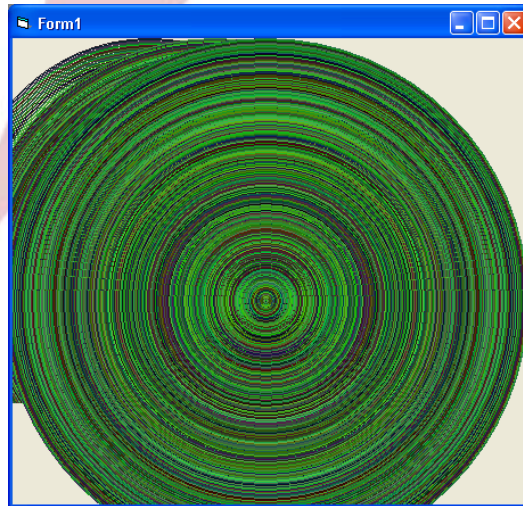
## حلقات التكرار :For

يمكننا فهم هذه الخاصة من خلال الأمر التالي والذي يقوم بتكرارات معينة حسب الرقم المحدد في For وذلك للحدث المحدد:  
ضع الكود التالي في أداة التوقيت:

```
Dim h
Dim w
h = Me.Height / 2
w = Me.Width / 2
For i = 1 To w
Me.Circle (h, w), i, RGB(Rnd * 100, Rnd * 200, Rnd * 90)
Next
```

عدل الخاصة Interval من 0 إلى 100.

شغل البرنامج وراقب النتيجة وقارنها مع الشيفرة...

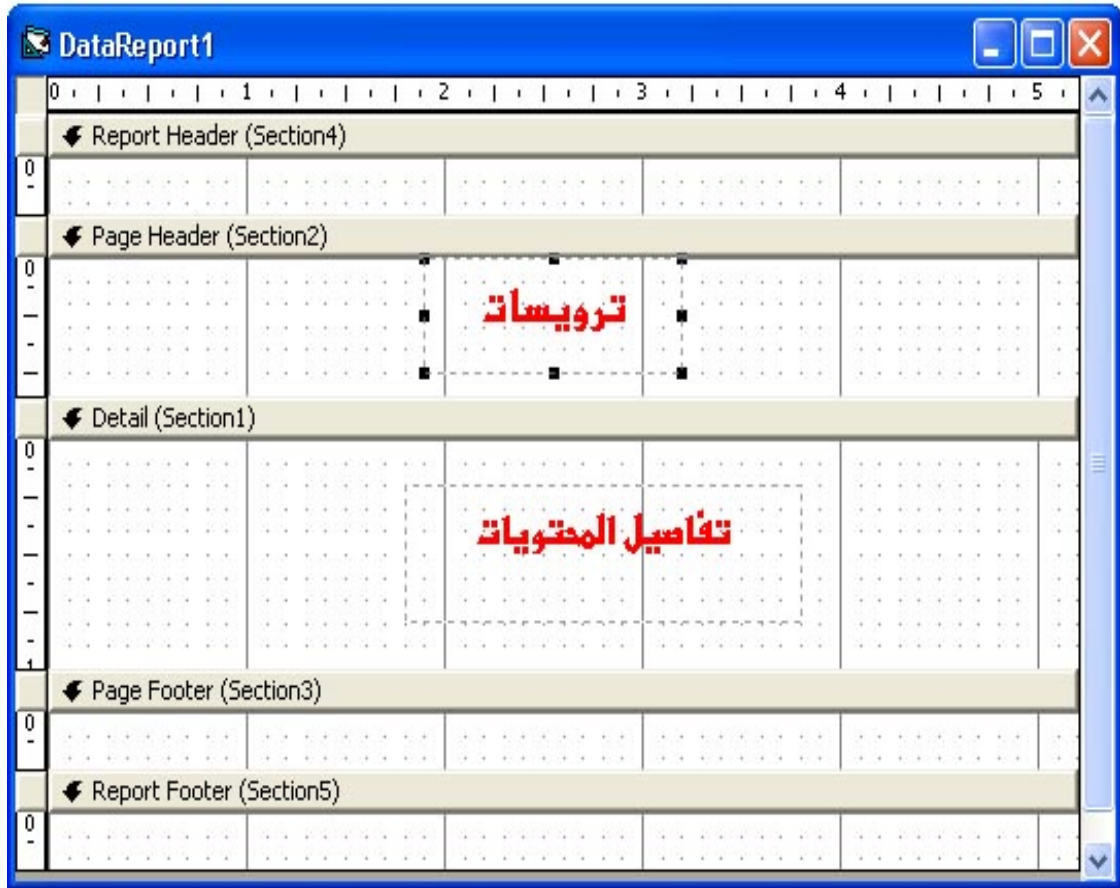


## عمل تقرير طباعة:

إن كافة البيانات التي تنشئ عن قواعد البيانات لن تكون لها فائدة كبيرة ما لم تترجم وتصنف نتائجها في قوائم أو بالأصح إلى تقارير يمكن من خلالها تقديم النتائج المطبوعة والمنظمة للمدراء والزبائن و.....

ويمكن ذلك من خلال تقرير الطباعة الخاصة بالفيجوال بيسك والذي يمكننا التعامل معه حسب المراحل التالية:

من القائمة Project اختر Data Report. فتظهر لك على الشكل التالي:



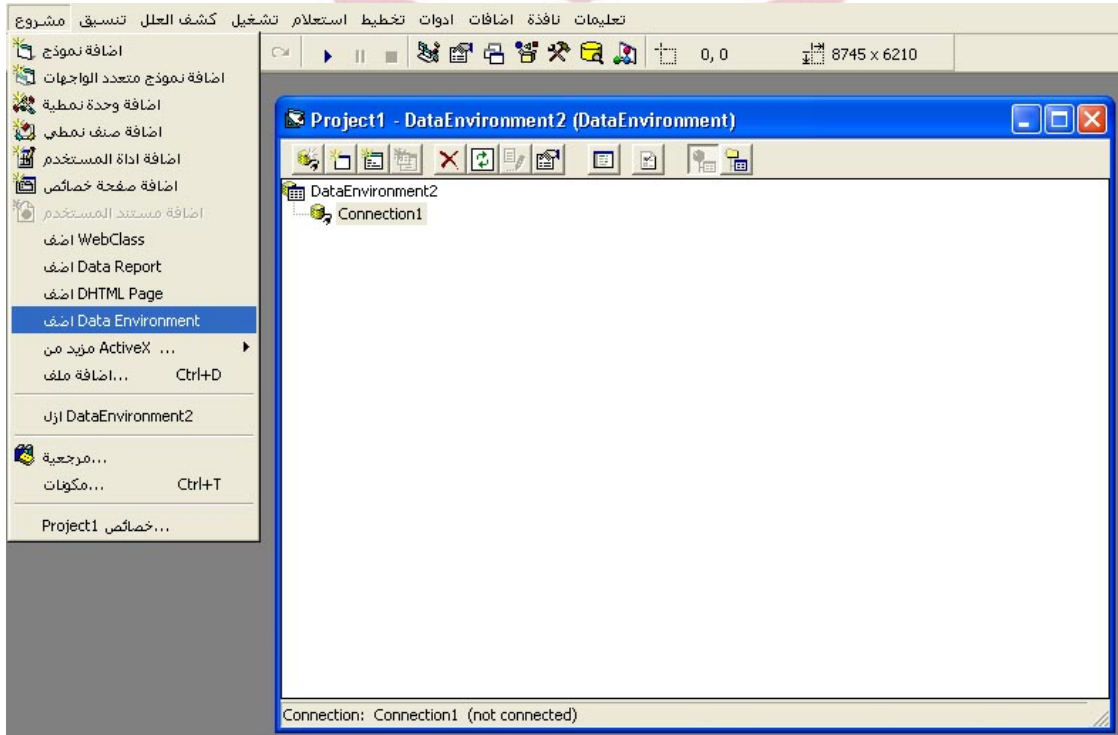
نلاحظ أن لدينا التقرير مقسم إلى ثلاثة أجزاء رئيسية:

- الترويسات وهي قسمين ترويسة التقرير العامة وترويسات الصفحة.
- منطقة إظهار البيانات (تفاصيل المحتويات).
- تذييل التقرير والصفحة.

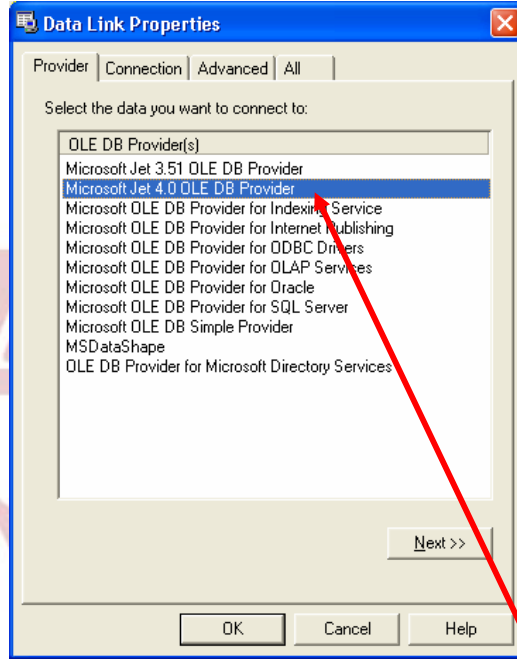
نقوم الآن بتعريف تقرير الطباعة على قاعدة البيانات وذلك من خلال وسيط بين قاعدة البيانات والتقرير. وإنشاء عنصر ربط وهو **Data Environment**.

من قائمة **Add Data Environment . Project**.

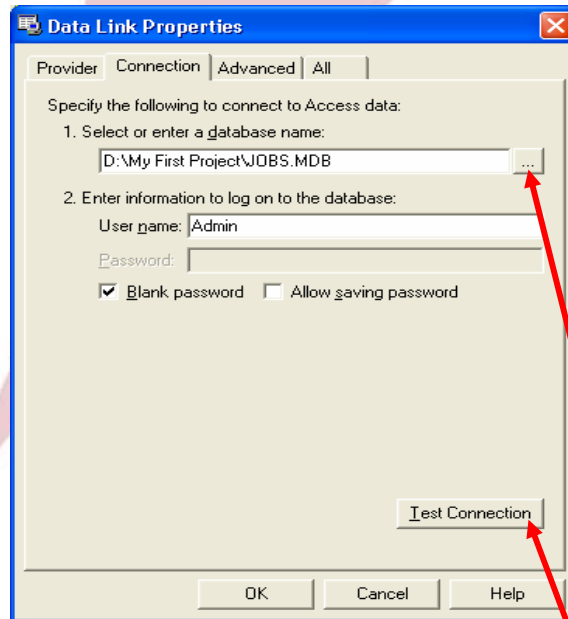
فتظهر لنا اللائحة التالية:




من خصائص Connection1 ستظهر لنا اللائحة التالية:

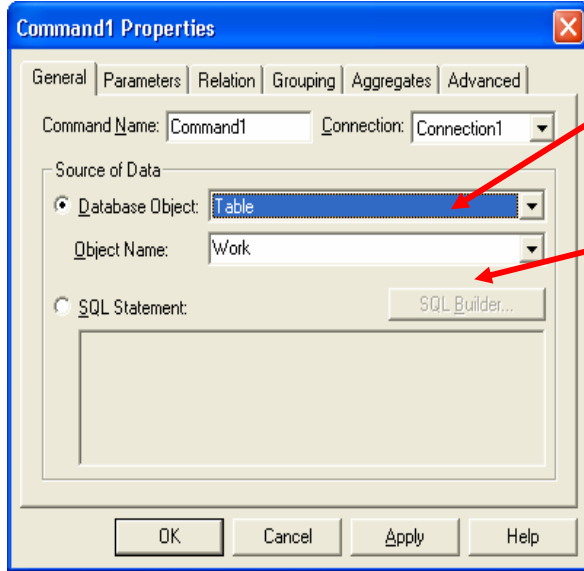


بعد اختيار البند المُشر عليه نضغط Next:



من اللائحة أعلاه نحدد قاعدة البيانات التي سبق وأنشأناها.  
ثم نقوم باختبار الاتصال مع قاعدة البيانات المحددة.

بعد ضغط موافق نعود إلى اللائحة الأولى ونختار من أشرطة الزر Add Command ثم فوق زر الخصائص  فيظهر لك الشكل التالي :



ثم نقوم بتحديد نوع المشروع المطلوب.

وتحديد اسم الجدول في قاعدة البيانات.

وبعد ضغط OK سوف تنتهي عملية الربط بين قاعدة البيانات وتقرير الطباعة.


وبالعودة إلى تقرير الطباعة ومن خلال

الأداة  نقوم بكتابة الترويسات التي تتناسب مع بياناتنا... في قسم الترويسات.

كما نقوم بتعديل خصائص التقرير على النحو التالي:

. DataEnvironment1 : **Data Source**

.Command1 : **Data Member**

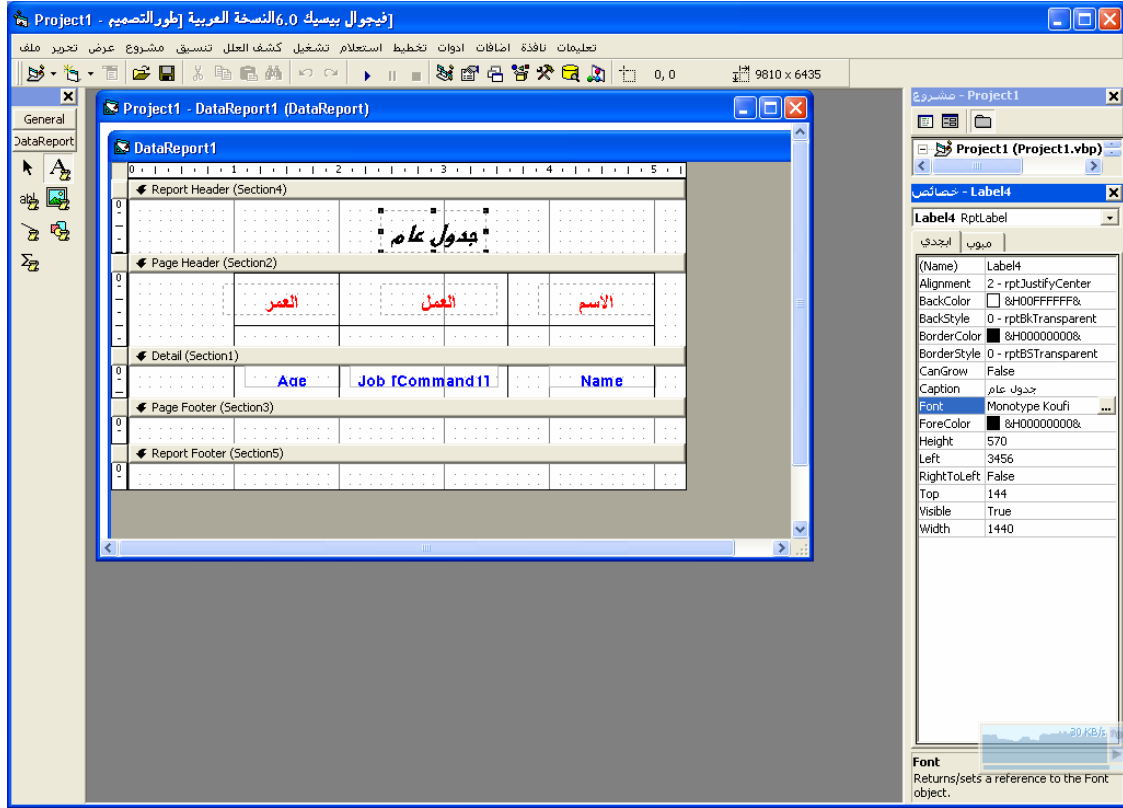
ومن خلال الأداة  نقوم بتحديد البيانات التي ستظهر في التقرير داخل قسم المحتويات... بعد أن نقوم بتعديل الخصائص بها في كل مرة على النحو التالي:

**DataMember** : الجدول

**DataField** : الحقل

**CanGrow** : تستخدم هذه الخاصية لتحديد عرض سجل واحد أو كافة السجلات سوية داخل التقرير.

أعتقد الآن أن التقرير أصبح جاهزاً وسيظهر في صفحة التصميم على الشكل التالي:



بقي أن نربط هذا التقرير مع البرنامج ويتم ذلك في الفورم الأساسي بوضع زر أمر ونكتب `DataReport1.Show`.

طبعاً سيظهر الجدول بدون تخطيط ويمكنك إجراء التخطيطات المناسبة له من خلال



وحسب برنامجنا السابق والذي أصبح على الشكل التالي:



مشروعى الأول

أزرار الأوامر

سجل المعلومات

الاسم:

العمل:

العمر:

الهاتف:

Text5

Combo1

طباعة النتيجة

عودة للوحة الرئيسية

بحث

طباعة تقرير

شريط العرض

اضغط على طباعة تقرير وشاهد النتائج

DataReport1

Zoom 100%

جدول عام

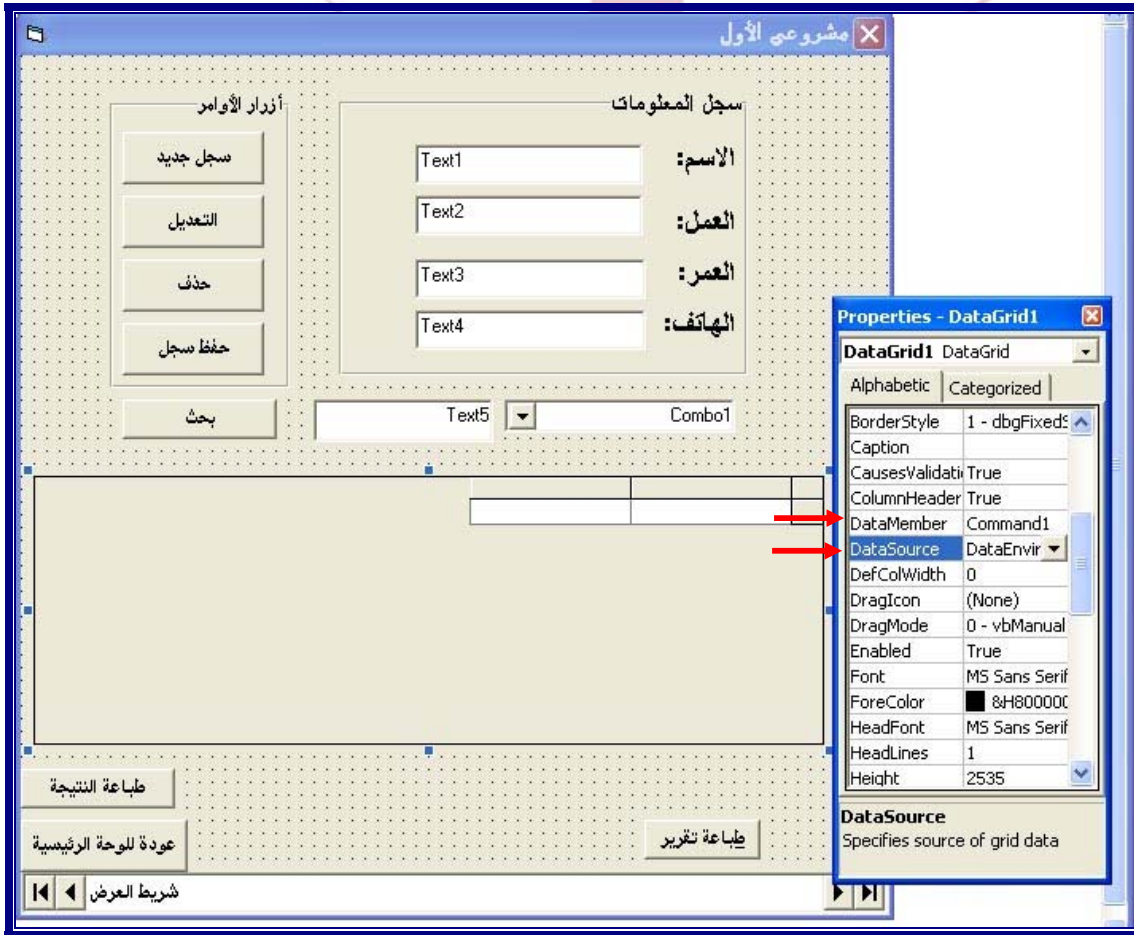
العمر	العمل	الاسم
25	خبير شبكات	يزن حرفوش
25	خبير صيانة حواسب	محمود المصري
42	استشاري تقنيات معلومات	عزام القاسم
18	طائب	أمين القاسم

Pages: 1

## إظهار البيانات في جدول:

لإظهار البيانات في سجل أو جدول داخل الفورم بشكل مباشر... نتبع الخطوات التالية...

- نجلب الأداة Microsoft Data Bound Grid Control إلى شريط الأدوات من خيارات شريط الأدوات... (وأنوه هنا على أن النسخة التي يجب أن تعمل عليها يجب أن تكون كاملة مع كافة الأدوات المساعدة).
- نضع هذه الأداة في الفورم ونغير الخصائص الخاصة بها كما يلي:



- بعد التحديد حسب ما سبق نعود ونقف على DataGridView1 داخل الفورم ونضغط الزر الأيمن للماوس ونختار Retrieve Fields والآن شغل البرنامج وشاهد...

مشروعى الأول

أزرار الأوامر

سجل المعلومات

الاسم:

العمل:

العمر:

الهاتف:

سجل جديد

التعديل

حذف

حفظ سجل

بحث

Text5

Combo1

Tell	Job	Age	Name
3719081	خبير شبكات	25	بزن حرفوش
092365638	خبير صيانة حواسب	25	محمود المصري
	استشاري تقنيات معلومات	42	عزام القاسم
092300286	طالب	18	أمين القاسم

طباعة النتيجة

عودة للوحة الرئيسية

طباعة تقرير

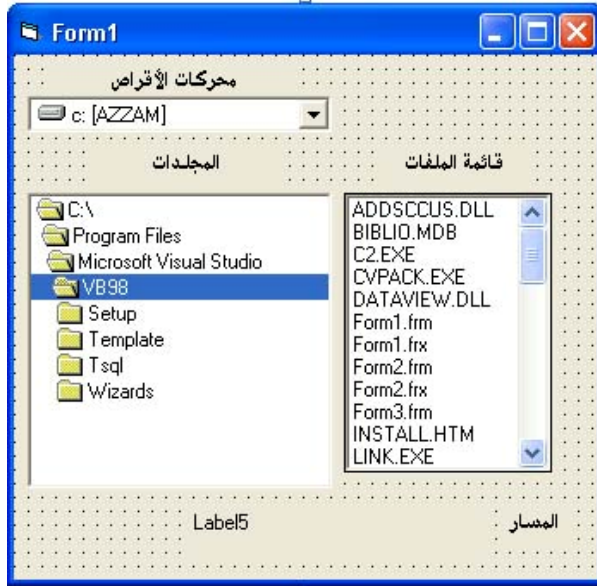
شريط العرض

## المتديا

### تشغيل الصوت:

سنقوم هنا بتشغيل الصوت في البرنامج مع عناصر تحكم الصوت حيث يمكننا إظهار شريط أدوات التحكم بالصوت وكذلك اختيار الملفات.

هل تتذكر مثالنا السابق:



سوف نعود إليه ولكن سوف نعدل الكود الخاص بقائمة الملفات ليصبح على الشكل التالي:

```
Label1 = Dir1.Path & "\" & File1  
MMControl1.FileName = Label1  
MMControl1.Command = "open"  
MMControl1.Command = "play"
```

السطر الأول يقوم بعرض المسار الكامل للملف المحدد في صندوق

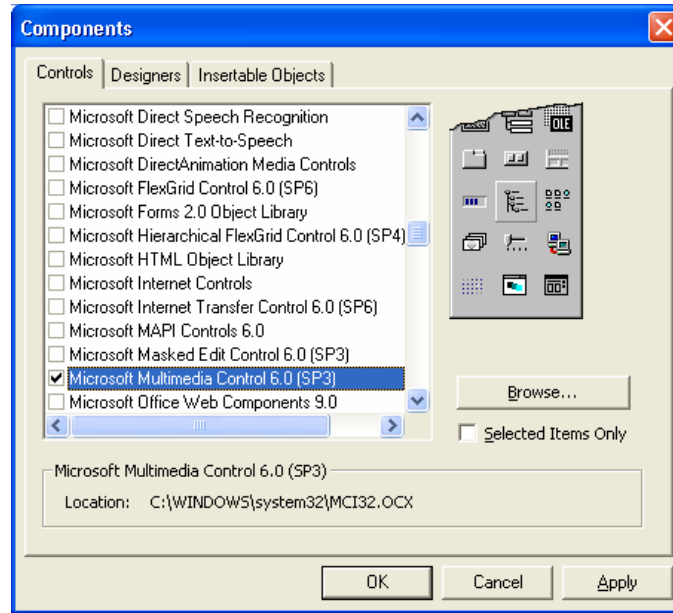
الملفات داخل الأداة Label1

السطر الثاني يقوم بتحديد الملف لأداة

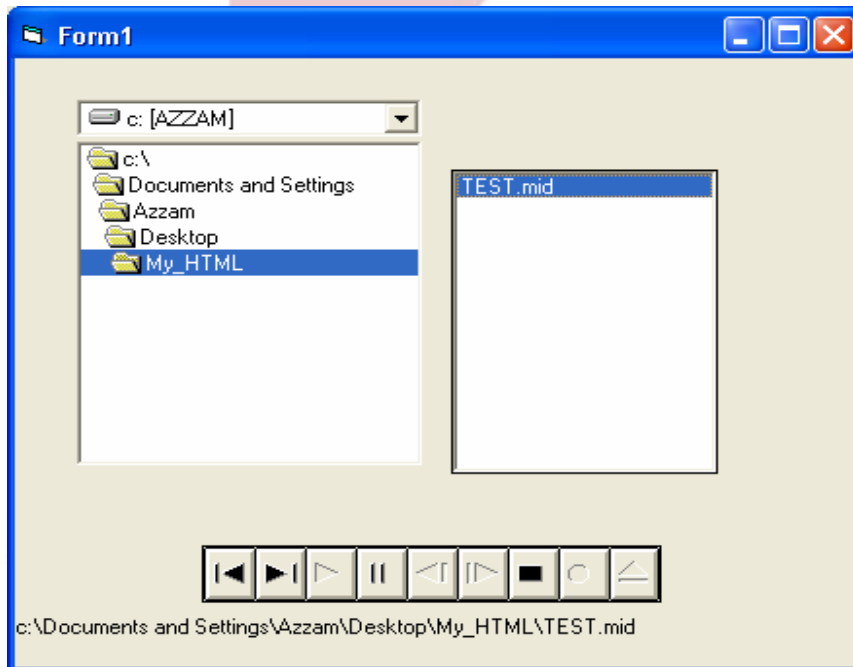
المتديا وهو الملف المكتوب داخل الأداة Label1

السطرين الأخيرين يقومان بتشغيل الملف

سوف نضيف الآن شريط التحكم الخاص بالصوت وهذا يمكنك الحصول عليه من خلال النقر بالزر اليمين للماوس فوق صندوق الأدوات . فتظهر قائمة تختار منها Components فيظهر لك مربع حوار يحوي على الأدوات سوف تجد فيها العنصر Microsoft Multimedia Control وأنقر فوق الزر Ok.



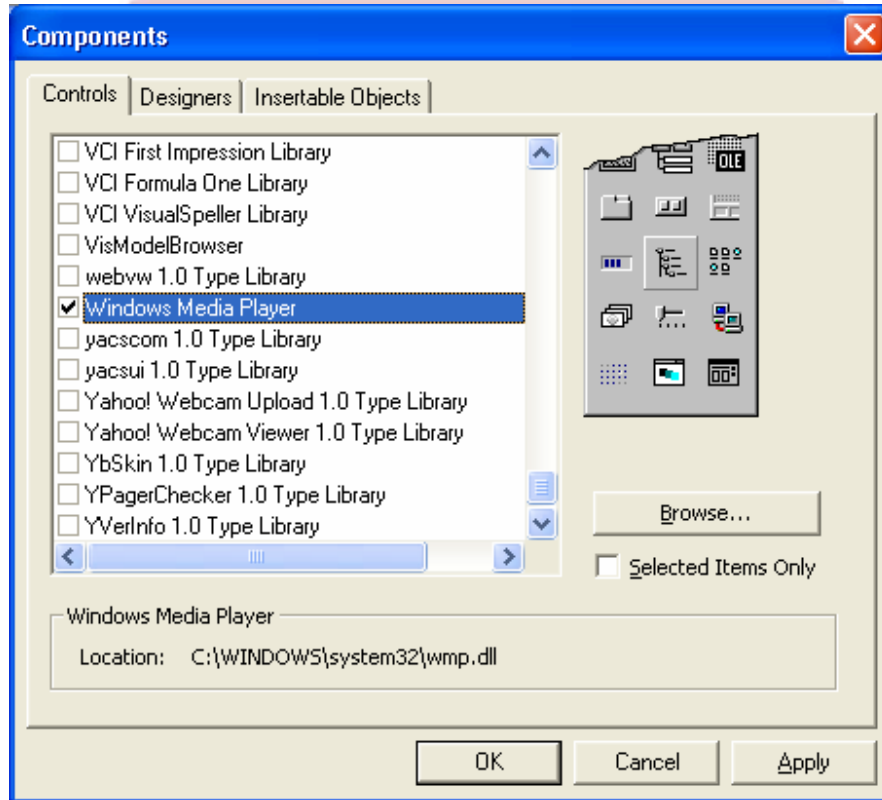
فيظهر رمز جديد في شريط الأدوات هو : MMControl.



جرب الآن وسترى هذه النتيجة بنفسك..

## تشغيل الصورة:

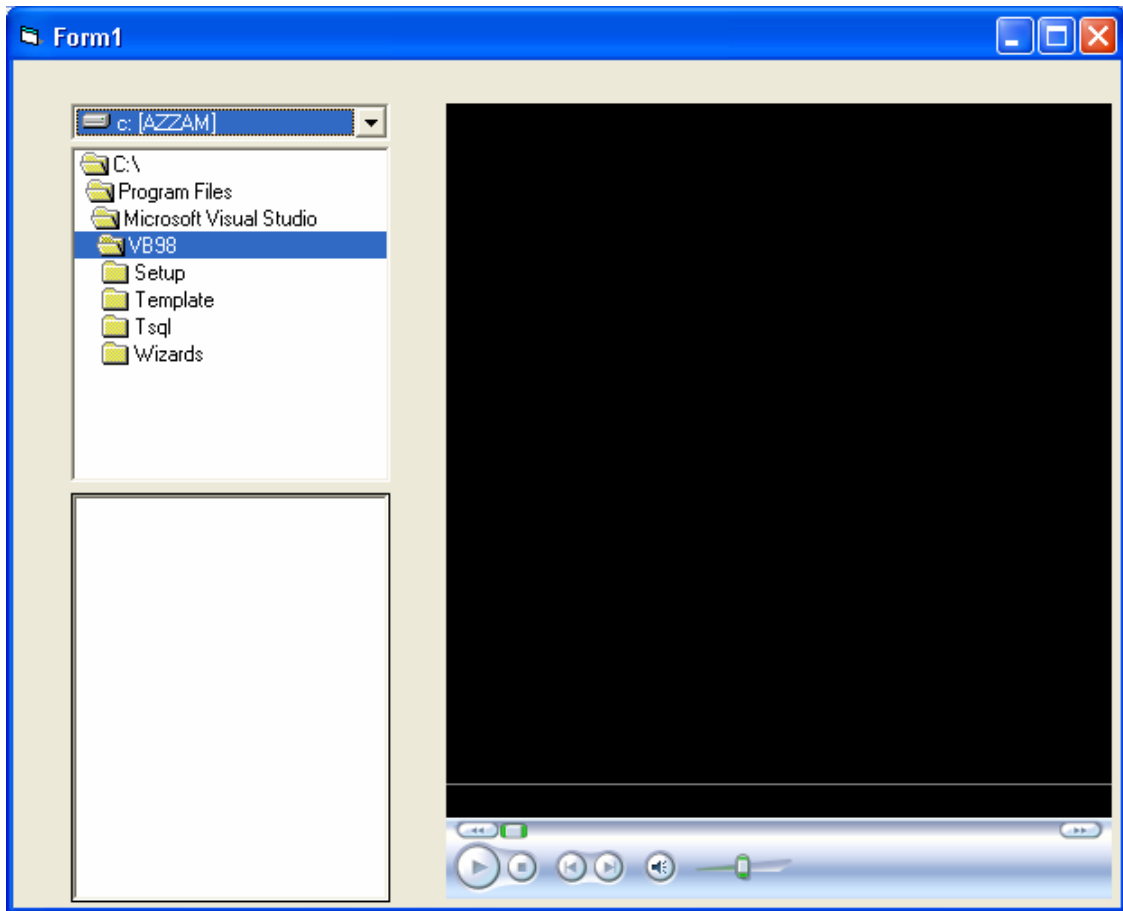
نعود إلى مثالنا السابق ونقوم بحذف شريط التحكم بالصوت ونختار عنصر تحكم جاهز هو Windows Media Player كما في السابق.



نضع الأداة الجديدة داخل الفورم بالطريقة التي تناسبنا وثم نغير الكود الخاص بقائمة الملفات إلى..

```
Label1 = Dir1.Path & "\" & File1  
MediaPlayer1.pause  
MediaPlayer1.play  
MediaPlayer1.Stop
```

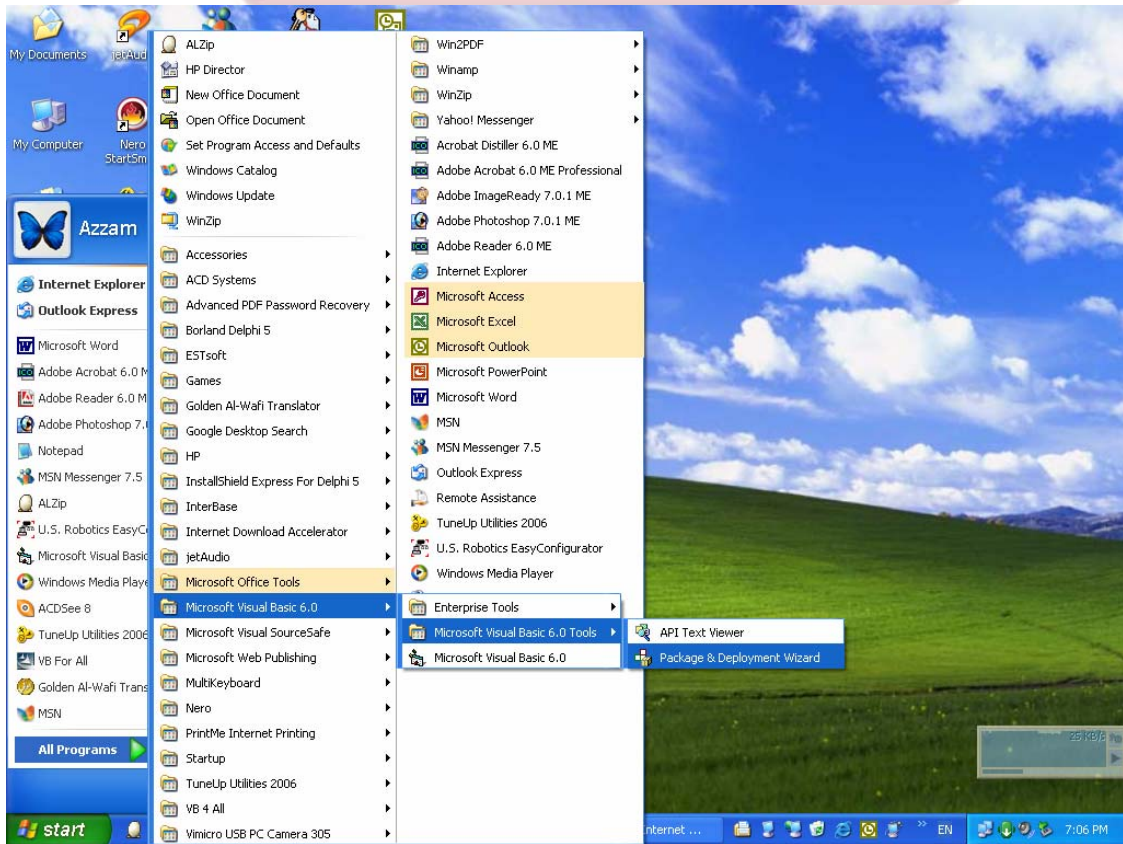
أصبح لديك الآن برنامج تشغيل فيديو يمكنك تنسيقه بالشكل الذي تريد.



## حزم البرنامج وتجهيز الـ Setup الخاص به

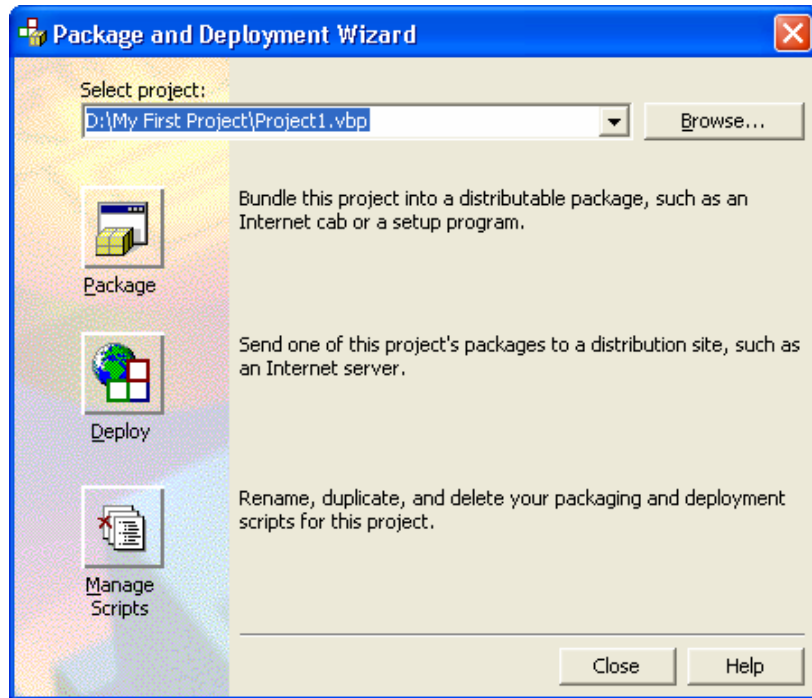
لكي نتمكن من العمل على أي برنامج قمن بتصميمه فلا بد لنا من تجميع ملفاته لنقله لجهاز آخر وهذا غير ممكن بسهولة إلا من خلال البرنامج المرافق للفيجوال بيسك وهو: deployment wizard & package وهو من ضمن أدوات الفيجوال وتجده كما في

الشكل:

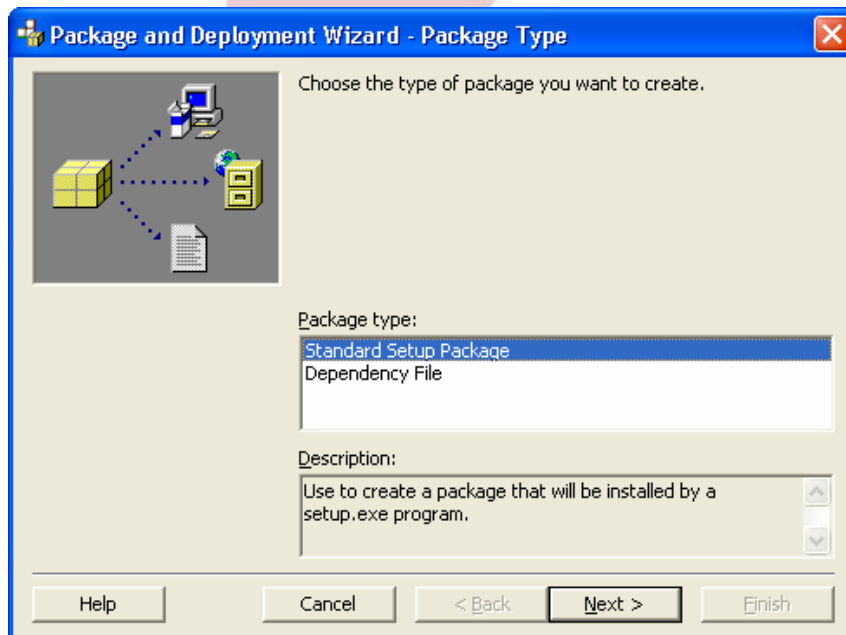


فور تشغيل البرنامج سيظهر لنا مربع الحوار والذي من خلاله سنتمكن خطوة خطوة من تجهيز برنامجنا وحزمه بشكل جيد وأنيق ويمكن تنصيبه في أي مكان من خلال برنامج الـ Setup الذي سنقوم بإعداده.

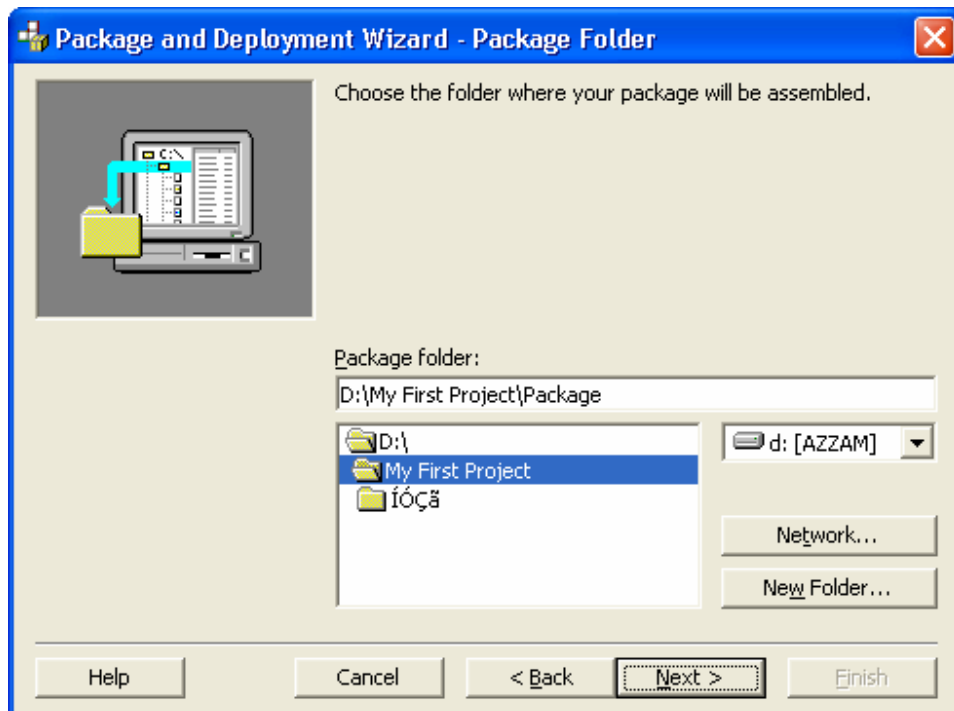




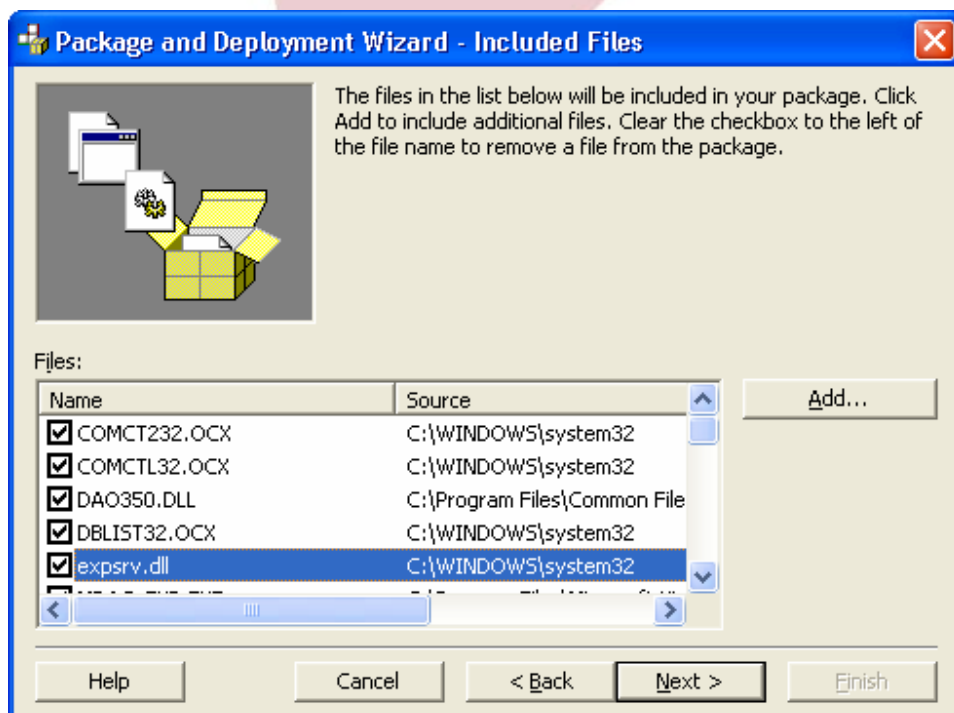
نحدد مكان البرنامج الذي قمنا بإعداده ثم نضغط Package.



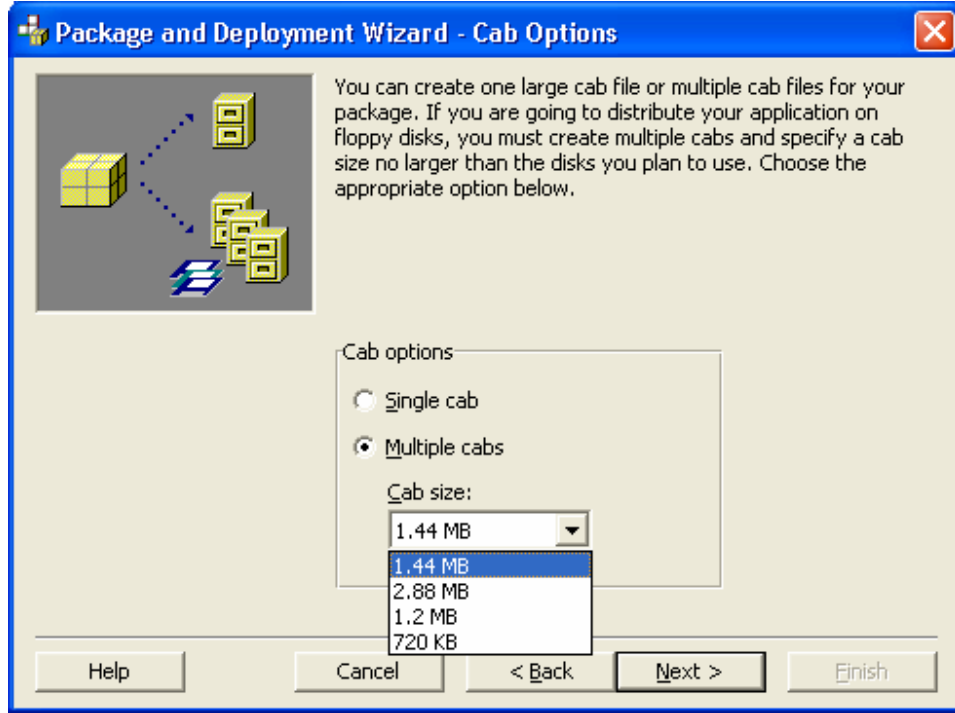
نختار نوع الحزمة عادي. ونضغط Next.  
لتظهر لائحة جديدة لتحديد المسار الذي سيتم تسجيل البرنامج به...



ويفضل اختيار مجلد جديد...

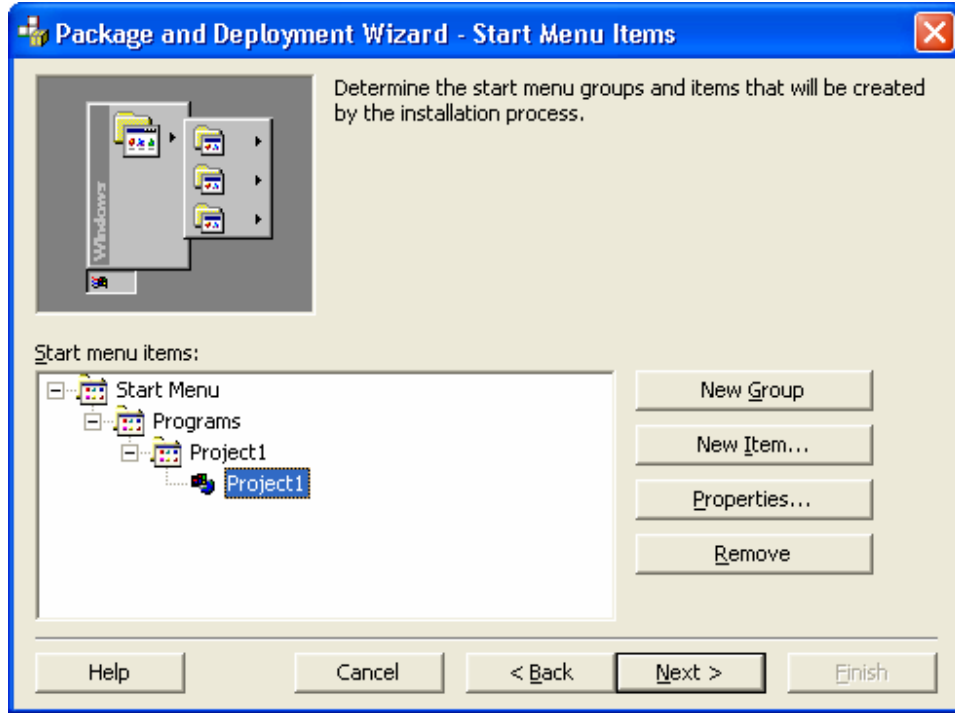


قائمة الملفات أعلاه هي مهمة جداً لتشغيل البرنامج بعد أن تتسخه من جهازك فهي تحتوي على كل ما يلزم لإعادة تركيبه وتشغيله على أي جهاز بشكل جيد (يُنصح بعدم التعديل فيها أو الإضافة إلا بعد وصولك إلى درجة متقدمة من التطبيقات العملية).



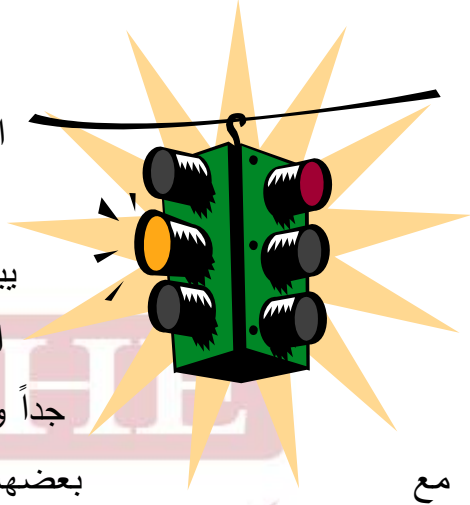
واللائحة السابقة تحدد لنا الحجم الذي نرغب بتجزئة البرنامج له حتى نتمكن من إعادة تحميله.. وبعد ضغط **Next** ستظهر لائحة جديدة نسجل فيها عنوان أو اسم البرنامج الذي نرغب أن يظهر (الاسم) عند عملية التنصيب.

والآن تظهر لائحة جديدة وهي هامة حيث يمكننا من خلالها تحديد المكان الذي سيتوضع به البرنامج بعد عملية التنصيب... وذلك حسب الشكل أدناه.



بعدها تظهر لائحة لتحديد الدليل الذي نرغب بأن تتم عملية تسجيل البرنامج به...  
وبعد تحديد اسم الحزمة أيضاً سيكون عندك برنامجك الخاص والذي أصبح يمكنك الآن  
أن تثصبه على أي جهاز بكل ثقة..

أخيراً... أرجو أن نكون قد وضعناكم على أول الطريق في عالم البرمجة، ومهما حاول الأساتذة والباحثون أن يكتبوا ويشرحوا عن أي لغة برمجة يبقوا قاصرين عن تغطية كافة نواحي اللغة موضوع البحث وذلك بسبب الإمكانيات والخيارات الضخمة جداً والتي ينتجها تداخل الأدوات الخاصة باللغة البرمجية بعضها البعض...



مع

وها نحن قد تعرفنا معاً إلى أبجديات لغة الفيچوال بيسك ويبقى أمامك عزيزي الطالب إلا أن تغني معرفتك بها من خلال الأمثلة والنماذج التطبيقية التي تنفذها إضافة إلى البحث المستمر عن آلية تطبيق أي علاقة برمجية فبقناعتي لا يوجد حالة برمجية لم تغطيها لغات البرمجة ولكن علينا الابتكار والتفكير السليم واسمحوا لي بهذا التعبير (الاحتياي) على أدوات اللغة للوصول للنتائج التي نريدها.

## مشروع...

شركة تعمل في توزيع الأقراص الليزرية بمختلف أنواعها...  
أنشئ برنامج أرشيف لهذه الشركة يتضمن:

1. بطاقة عامة للإدخال تحتوي على كافة المعلومات الخاصة بكل نوع من أنواع الأقراص (أوديو . Mp3 . أفلام....).
2. أنشئ خيارات بحث متنوعة مع إظهار النتائج.
3. صمم تقرير طباعة للموجودات.
4. ضمن البرنامج موسيقى مناسبة.
5. ومن المستحسن تزيين البرنامج برسومات وأيقونات مناسبة.
6. يترك للطالب حرية إضافة ما يراه من ميزات تتناسب مع طبيعة البرنامج.
7. أعد برنامج Setup خاص بالبرنامج.

أرسل البرنامج بشكل Setup إضافة إلى النسخة البرمجية منه إلى أستاذك المشرف.

مع تمنياتنا لكم بالنجاح والتميز  
والنجاح والتميز