الفصل الخامس

Practice

التطبيق العملي

مقدمـــــة

في هذا الجزء سيجد القارئ شرح مفصل لكيفية تصميم تطبيق خاص بموظفي شركة ، ولسوف يجد القارئ في هذا الكتاب ما يمكنه من تصميم برنامج يمكن تسويقه ، وهو برنامج حقق نجاحاً كبيراً ، فيما يتعلق بتلبية حاجات المستخدمين ، وتسهيل الخطوات المتعلقة بهذا الشأن .

فسوف نستعرض مشروع النظام الإلكتروني الخاص بالمتابعة لموظفي الشركة وشركات الأدوية والمستشفيات . وهذا البرنامج أنتجته شركة نانو سوفت للبرمجيات ، وقد حقق نجاحاً كبيراً في سوق البرمجيات .

وتجدر الإشارة – هنا – إلى أن هذا البرنامج لم يتم تسويقه في مصر بشكل كامل، بل اقتصر التسويق على محافظة واحدة من محافظاتها ، كما لم يتم تسويقه في أي بلد عربي آخر ، وهذا معناه أن فرصتك لتسويق هذا البرنامج كبيرة .

غير أنني أنصح القارئ ، ألا يكون هدفه الأول تسويق البرنامج ، بل أن يتخذه – أولاً – نموذجاً لإنشاء برامج أخرى ، مثل برنامج إدارة المخازن ، أو محلات الأجهزة الكهربائية .
المهم..

- أن تبدأ بنفسك ..
- وأن تستغل التقنية ، التي تعلمتها من خلال هذا الكتاب في إنشاء تطبيقات من ابتكارك..
- وأن تتمتع بميزة التفرد ..

فإذا فعلت ذلك ، فإن فرصتك في تسويق برنامج من ابتكارك تكون أكبر من فرصة تسويق برنامج وجدت فكرته من قبل .
وعلى أية حال فإن الأمر مرده إليك ..

برمجة الحاسب
Computer Programmin

الحاسب أو(الكمبيوتر) لا يعلم أي شيء على الإطلاق ، إنه لا يعرف كيف يمكن إظهار المعلومات والبيانات على الشاشة ، ولا الوصول إلى البيانات المسجلة على القرص الصلب .. لا يعرف كيف يتصل بالشبكة الدولية العنكبوتية (الإنترنت) ، لا يعرف كيفية تشغيل ملف صوتي .. إنه لا يعرف شيئا على الإطلاق !

لكن الحاسبات تعمل بشكل جيد للغاية في تنفيذ الأوامر والتعليمات دون خلل وبسرعة كبيرة ، وهذا ما جعل تلك الآلات قوية .

فلابد أن يقوم شخص ما بكتابة التعليمات (الشفرة - الكود) لكي يقوم الحاسب بتنفيذ التعليمات ، التي يعتقد البعض أنها من عمل الحاسب ، وأنها لا تحتاج إلى مساعدة خارجية من شخص آخر هو (المبرمج) .

إن تشغيلك ملف صوتي واحد على الحاسب قد يحتاج لتنفيذ ملايين من العمليات والأوامر ، حتى تسمع الصوت الخارج من السماعة المتصلة به . فبرمجة الحاسب – إذن – هي فن كتابة (الكود) ، الذي يقوم الحاسب بتفسيره ، ثم يقوم – بناء على ذلك – بتنفيذ مجموعة من العمليات المتكررة .

والحواسب لا تقع في أي خطأ ، إلا إذا كان هناك خطأ مادي ، أي في مكونات الجهاز نفسه ، إنها تفعل ما تؤمر به بشكل دقيق . وربما تقول : أن هناك بعض الأحوال التي يخطئ فيها حاسبك الآلي ويعطيك رسالة خطأ . والحقيقة أن حاسبك لم يخطئ ، بل الخطأ جاء من المبرمج ، الذي أعطى الحاسب تعليمات ناقصة ، لأنه لو قام بكتابة التعليمات بشكل دقيق وكامل لما أظهر الحاسب رسالة الخطأ هذه ، ولنفذ العمليات المطلوبة بكل دقة .

إن ما يحدث أمامك على الشاشة شيء عجيب حقاً ؛ فهو نتيجة عشرات الآلاف مما يسمونها بـ **Tiny on and off switches**. وأعتقد أن كثيرين منا قد رأوا الكود 01010101011100010110001 ، فهذا الكود مسئول عن إحدى الوظائف ، مثل لون النقطة أمامك على الشاشة حسبما يقرر المعالج ذلك ؛ ومثل نسخ ملف في ذاكرة النصوص ، أو الاستماع إلى ملف صوتي .

هناك عشرات الآلاف من عمليات الانتقال الصغيرة من التشغيل **ON** والإيقاف **OFF** تحدث ؛ لكي تتم معالجة البيانات في الذاكرة ، ولكي يتم إرسال إشارة إلى بطاقة الشاشة أو الصوت أو لاستعادة (بايت) من على القرص الصلب .

ولحسن حظي وحظك أننا لا يجب علينا كتابة مثل هذا الكود الذي يتكون من عشرات الآلاف من عمليات الانتقال ، التي يرمز لها بـ 01 .

ولغات البرمجة الحديثة تعد لغة اتصال سهلة بيننا وبين الحاسب ، وأصبح (التقييد) الذي كان يواجه المبرمج في الماضي غير موجود ، فهناك العديد من الأدوات ، التي تقوم الآن بكثير من الوظائف عنك ، إذ يكفي أن تدرج أي أداة منها داخل مشروعك ، لتجد أن الوظيفة التي تريدها من هذه الأداة بدأت في العمل ، دون ضبط أية إعدادات ، أو كتابة أي كود .

ووظيفة المبرمج تغذية الحاسب بالأوامر ، من خلال استخدام لغة معينة من لغات البرمجة وهو ما سنقوم به في هذا الكتاب .

فإذا كنت جديداً في عالم البرمجة فإنا ننصحك أن تبدأ بلغة (**Visual Basic**) ؛ لأنها تتميز بالسهولة ، بينما لغة الـ (**C#**) تعبر عن تطور للغات برمجة أخرى . ويمكنك أن تفضل وتعلم اللغة الأخيرة ؛ إذا كانت لك بإحدى لغات البرمجة مثل (**Java**) أو (**Perl**) أو (**C**) فقد تفضل تعلم لغة **C#** ؟!

طبقـات البرمجيات
Software Layers

هي عبارة عن كود سبق أن تمت صياغته من قبل ، فهو – (كود جاهز) – أنشئ بمعرفة مطورين قاموا بتوظيفه ، بحيث يشمل مجموعة من العمليات الجاهزة ، التي يمكنك استخدامها أثناء كتابتك للكود .

فلو أنك - على سبيل المثال - أردت القيام بتصميم تطبيق لإحدى قواعد بيانات ، وهذا التطبيق يعمل بأسلوب معين على نظام (ويندوز اكس بي) ويعمل بطريقة أخري على نظام (ويندوز 2000) ويعمل بطريقة ثالثة على (ويندوز 98) ، إذن فسوف تحتاج لجعل البرنامج يتعرف على نوع النظام الذي يعمل عليه ، قبل قيامه بأية عمليات خاصة بقواعد البيانات ، ولك أن تتخيل حجم الكود الذي لابد أن تكتبه في هذه الحالة .

و لكن تخيل أيضا أن المطورين تمكنوا من تطوير أداة جاهزة يمكن إدراجها داخل تطبيقنا تتيح لك التعرف على النظام ، لتخبر التطبيق به ، بحيث يعمل التطبيق وفقا للقيمة ، التي تعود إلى هذه الأداة الجاهزة .

فهذه الطرق الجاهزة تسمح لك بتنفيذ آلاف المهام مثل :

- العمل مع الوقت والتاريخ .
- كتابة بيانات الملف واستعراضها وتعديلها .
- الاتصال بقواعد البيانات المختلفة ومعالجتها .
- الاتصال بالشبكة العنكبوتية (بالإنترنت) .
- عرض النوافذ على الشاشة .

وهكذا..

إنها مكتبة كاملة ، لم تدع شيئا تجعلنا نضطر لكتابة كود خاص به ، فيكفى أن نتعلم الـ (MSDN) لكي نعرف مدى اتساع هذه المكتبة ، التي تعرض كثيرا من المهام ، التي لم تعد تحتاج إلى كتابة كود خاص بها .

قاعدة البيانـات

هي مجموعة من الملفات على جهاز الحاسب، نقوم من خلالها بتخزين البيانات بطريقة منظمة؛ بحيث يمكننا استدعاء هذه البيانات بطريقة سهلة ، كما أنها تحفظها بطريقة آمنة، تتفاوت درجة الأمان فيها تبعا لنوع قاعدة البيانات المستخدمة .

وهذه البيانات تتم معالجة القواعد الخاصة بها من خلال أحد البرامج . فيمكنك على - سبيل المثال - إنشاء قاعدة بيانات (Access) ومعالجتها وفقا لبرنامج يعتمد نظام (visual Basic) ، أو برنامج (**C**#) أو - حتى - برنامج مثل (Visual Basic 2005 أو Delphi) .

أهمية قاعدة البيانـات

ربما تكون قد سبق لك في عملك القيام بعمليات التسجيل الورقية (الأرشيفية) ، وربما أيضا الملفات النصية على جهاز الحاسب ، أو أي طريقة من الطرق التي يمكنك من خلالها تسجيل بياناتك ، أو بيانات عملائك وربما تكون مستريحا لهذه الطرق القديمة ، لكن لاشك أن هناك فرق بين تخزين تلك البيانات (أو الاحتفاظ بها) وفقا للطرق القديمة ، وبين تخزينها وفقا لنظم الحاسب الآلي الحديثة . والفرق بين الطريقتين يشبه الفرق بين ركوب السيارة وركوب الطائرة .

فقد تكون مستريحا لركوب السيارة تماماً ، لكنك حين تجرب الطائرة لاشك أنك سوف تفضل استخدامها في قطع المسافات ، خاصة تلك المسافات الطويلة . وكذلك النظم الحديثة لتخزين البيانات ، فهي وسائل تعينك على العمل بشكل دقيق ، وفى أسرع وقت .

ففي الطرق القديمة يصعب عليك البحث داخل أرشيفك الورقي أو ملفاتك النصية ، كما يصعب عليك تعديل البيانات التي أخطأت في تسجيلها ، وكذلك يصعب استرجاع البيانات التي مر عليها وقت طويل . كما أن عنصر الأمان يكون غير متوفر لتلك الملفات ، إذ يمكن لأي شخص الإطلاع عليها ، أو سرقتها بسهولة .

ثم .. كم من الوقت تحتاجه لحساب ربحك الشخصي ، وكيف يمكنك معرفة أكثر السلع مبيعا إذا كانت مؤسستك كبيرة ، أو كنت من الذين لا يباشرون العمل بشكل مباشر، بل عن طريق توكيل العمل إلى آخرين ؟

هناك أسئلة كثيرة .. وهناك مزايا كثيرة ، لا يمكن أن تتوفر إلا من خلال إنشاء قواعد للبيانات . فمن خلال هذه القواعد يمكنك حساب ربحك السنوي في ثانية واحدة ، ويمكنك – كذلك – معرفة ما إذا كان النشاط أفضل في هذا العام أم الذي قبله ، كما يمكنك عمل علاقات تربط بين الصادر والوارد لمعرفة ما بيع ، ومازال موجودا ، هذا بخلاف عنصر الأمان الذي تتمتع به ؛ بحيث لا يمكن فقدان أي من هذه البيانات ، أو اطلاع الآخرين عليها .

إن بإمكانك - من خلال قواعد البيانات - أن تعرف وقت بيع السلعة – أو شرائها – بالشهر واليوم والساعة والدقيقة والثانية ، كما يمكنك أخذ نسخة احتياطية من تلك البيانات ، وأهم من هذا كله يمكنك الإطلاع على البيانات الخاصة بعملك وأنت في أي مكان في العالم ، دون الحاجة لأن تكون موجوداً في مقر عملك ، أو دفع ثمن باهظ للإطلاع على هذه البيانات .

قاعدة البيانـات العلائقية

هذه القاعدة هي التي تساعد على الحفاظ علي سلامة البيانات ، وذلك بالتأكيد على دقة نوعها ، سواء أكانت نصية أم رقمية . وهى تعمل على استبعاد البيانات العاطلة التي تؤدي إلى أخطاء . كما تعمل على استبعاد الأخطاء التي قد تؤدي بدورها إلى محو البيانات .وهى توفر إمكانية معالجة بطريقة جيدة وفعالة ، و سهولة استرجاعها .

لقد بدأ موضوع قاعدة البيانات العلائقية علي يد المبرمج (كاد) ، في العام 1979 وتطورت في العام 1989 تطوراً ملحوظاً . حاول (كاد) إيجاد حلول للمشاكل التي كانت تواجه العمل آنذاك ، حيث كانت الأجهزة ضخمة و الجداول تسجل على أشرطة لا على القرص الصلب كما هو الحال الآن ، فكانت عمليات التسجيل تكلف مبالغ ضخمة .

الآن.. أصبح الأمر غاية في السهولة ، كما أصبح أكثر دقة وأمنا ، فتستطيع الآن أن تختار نوع البيانات الخاص بكل حقل بطريقة سهلة ، كما تحدد نوع البيانات المسجلة ، والأخطاء التي يجب تلافيها .

جداول قاعدة البيانـات
Database Tables

قد تحتاج قاعدة البيانات لجدول واحد ، كما قد تحتاج إلى أكثر من جدول . فبإمكانك إنشاء قاعدة بيانات لكل جدول لكل جدول ، كما باستطاعتك إنشاء قاعدة بيانات تحتوي على عدة جداول .

فربما نحتاج إلى جدول للعملاء ، وآخر للمشتريات ، وثالث للمبيعات ، كما قد تحتاج إلى عدة أعمدة داخل جدول العملاء ، مثل اسم العميل ، ورقم هاتفه ، واسم شركته ، وعنوانه .. وتحتاج إلى عدة أعمدة – كذلك – داخل جدول المشتريات مثل ، اسم العنصر ، ووقت الشراء ، واسم الشركة الموردة . وكذلك الحال مع جدول المبيعات .

والجدول عبارة عن مجموعة من الأعمدة (Columns) والصفوف (Rows) ، التي تسمي سجلات (Records).. والصفوف (أو السجلات) عبارة عن عدة حقول ، عددها يساوي عدد الأعمدة الموجودة داخل الجدول ، والصف حاصل جمع حقول الأعمدة .

والشكل التالي يوضح جدولا كاملا :

CustomerID	FirstName	LastName
1	Bob	Tabor
2	Steve	Jaworski
3	Brian	Faley
4	Mark	Eaton
5	Andrew	Flowers
6	Scott	Logan

الأعمدة Columns

هي الحقول التي تتكرر في كل صف ، بحيث يتم ملؤها بالبيانات . وكل عمود يخص نوعا من البيانات ، بمعني أنه يوجد عمود خاص باسم العميل وهو (نصي) ، بينما العمود الخاص برقم الهاتف (رقمي) ، أما عمود وقت البيع فهو (تاريخ) . كما أنه ينبغي التنبه إلى أن كل عمود له اسم خاص به نستدعيه من خلاله ، وذلك حتى يمكننا تسجيل البيانات داخله والحصول عليها فيما بعد .

وفيما يلي جدول يوضح عموداً داخل أحد الجداول :

CustomerID	FirstName	LastName
1	Bob	Tabor
2	Steve	Jaworski
3	Brian	Faley
4	Mark	Eaton
5	Andrew	Flowers
6	Scott	Logan

أما الجدول التالي فيوضح صفا أو (سجلا) داخل أحد الجداول :

CustomerID	FirstName	LastName
1	Bob	Tabor
2	Steve	Jaworski
3	Brian	Faley
4	Mark	Eaton
5	Andrew	Flowers
6	Scott	Logan

بينما يوضح الجدول التالي بياناً داخل أحد الأعمدة :

CustomerID	FirstName	LastName
1	Bob	Tabor
2	Steve	Jaworski
3	Brian	Faley
4	Mark	Eaton
5	Andrew	Flowers
6	Scott	Logan

برامج المعالجة

تقوم هذه البرامج بالاتصال بقاعدة البيانات والوصول إليها ، وهى التي تسمح بإدخال بيانات جديدة ، أو الاستعلام عن القديمة ، كما تسمح بحذف هذه البيانات إذا تطلب الأمر ذلك .

البيـــــانات

تعبر البيانات عن (نصوص) أو (أرقام) أو(صور) أو - حتى – (أصوات) ، حيث يتم تخزين هذه العناصر جميعا داخل الحاسب وكذلك معالجتها .
و البيانات في حد ذاتها ربما تكون بلا معنى ؛ لذلك ، لابد من تفسيرها ، أو معالجتها لتصبح معلومات مفهومة ، يمكن إدراكها ، فالبيانات تعد معلومات ذات مغزي معين .

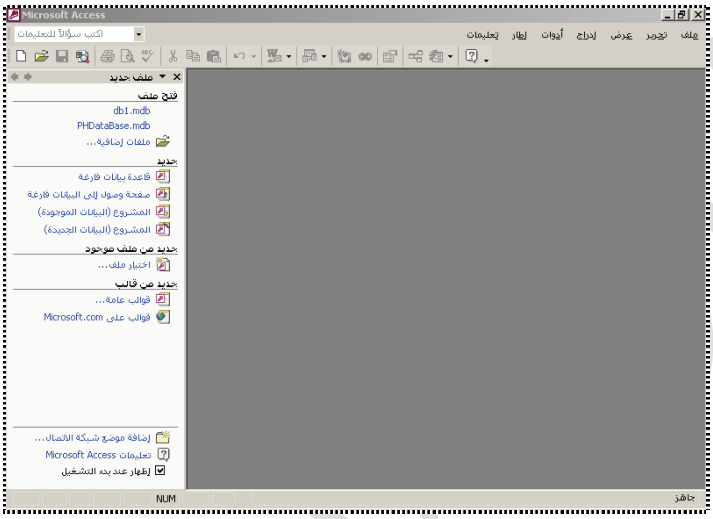
وهناك فرق بين مفهوم البيانات ومفهوم المعلومات . ولتوضيح ذلك نقول : أن كلمات مثل
(خدمة العملاء) أو (1500) أو (10) تشير إلى بيانات ، ولكن إذا أمكن تفسيرها على اعتبار أنها بيانات خاصة بشركة معينة فقد تشير تلك البيانات (على سبيل المثال) إلى أن قسم خدمة العملاء به (10) موظفين ، ومتوسط الدخل السنوي لكل واحد منهم
(15000) دولار ، عندئذ تصبح تلك البيانات معلومات واضحة . ولاشك أن من البديهي – أيضا - إمكانية اعتبار هذه البيانات ذات مغزى مختلف ، إذا تم تفسيرها بشكل آخر .

ولابد من الإشارة إلى أن البيانات تعد بمثابة المكونات الأساسية التي يقوم عليها أي عمل ، وحين تتم معالجتها ويتم دمجها معاً باستخدام تقنيات متعددة تظهر من خلالها نتائج مختلفة .
والآن..
قم بتركيب (**Office Xp Arabic**) .. وقد اخترنا ، مثل هذه النسخة لأنها متوفرة ، يمكنك الحصول عليها بسهولة .

وتعال الآن لنقوم بإنشاء قاعدة بيانات بسيطة وفقا للخطوات التالية :

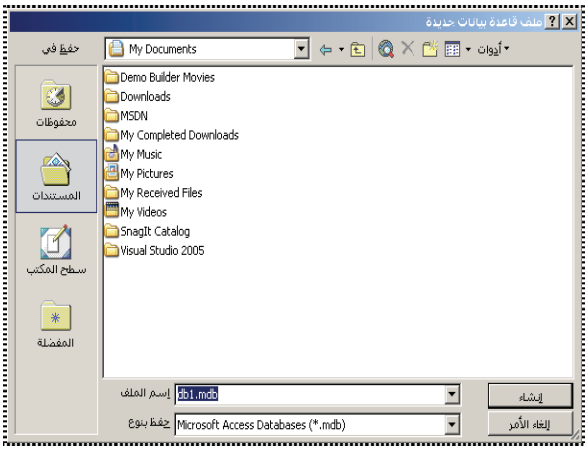
قم بفتح (Microsoft Access)

ستظهر لك النافذة التالية :

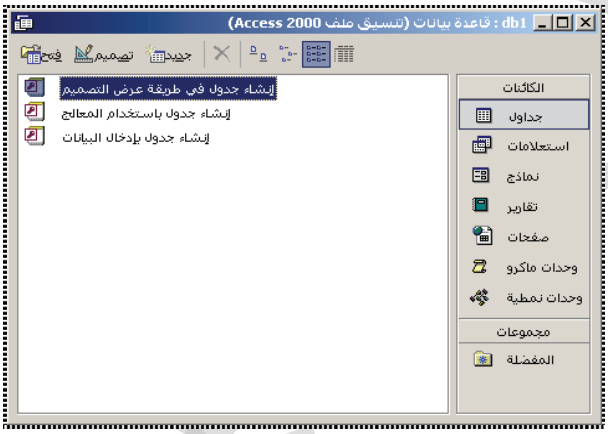


ثم اضغط على (قاعدة بيانات فارغة) :

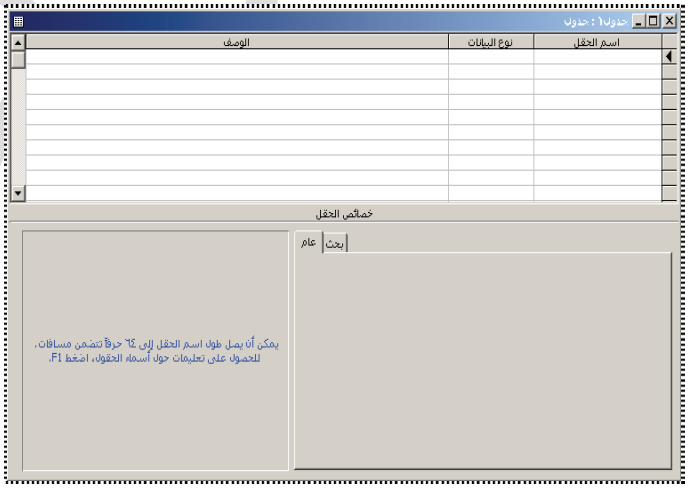
اختر الآن مكان حفظ قاعدة البيانات :



وسوف يعطيك البرنامج الاسم الإفتراضي (DBi) ، فقم بتغييره إلى (Ph Database) ؛ ثم اضغط على (إنشاء جدول في طريقة عرض التصميم) ، لبدء عملية التصميم :



ستظهر لك النافذة التالية التي سوف نقوم من خلالها بتصميم الأعمدة التي هي متوفرة داخل الجدول :



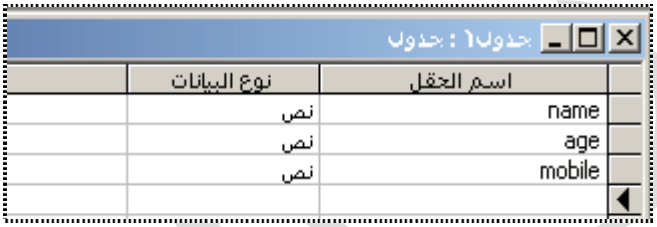
لاحظ :

إن الجدول الذي سنقوم بتصميمه بسيط جداً فهو يتكون من ثلاثة أعمدة (Name) و (Age) و (Mobile) ، ولن يكون هناك (Primary Key) أو أي شيء آخر:

قم – الآن – بكتابة كلمة name في خانة (اسم الحقل) ، ثم توجه إلى الخانة التي أسفل تلك الخانة و أكتب كلمة (age) ، ثم في الخانة التي بأسفلها كلمة (mobile) ، وسوف تلاحظ أنك كلما تركت خانة وتوجهت إلى الأخرى أن نوع البيانات يكتب تلقائيا (نص) وهو المطلوب . ويمكنك الرجوع إلى مفهوم نوع البيانات (Data Type) لفهم ذلك أكثر.

لاحظ :

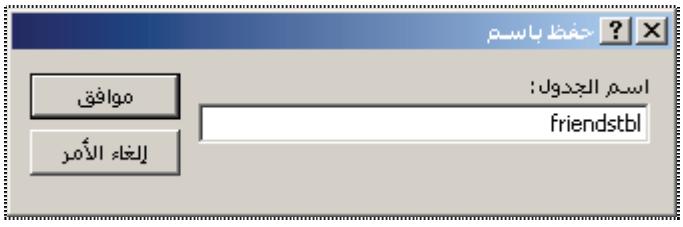
أن العمود age عبارة عن رقم ، و لكننا فضلنا اعتباره نصاً ؛ لكي تعرف أنه من الممكن كتابة أنواع بيانات مختلفة ، طالما أنها لا تتعارض مع نوع بيانات الحقل نفسه .



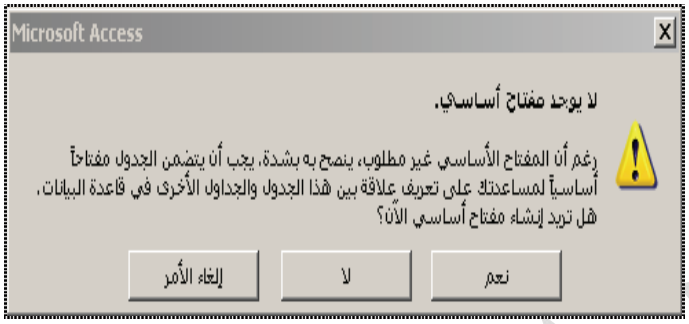
قم الآن بالضغط على زر الحفظ ؛ لحفظ التغييرات التي قمنا بها :

وسوف تظهر لك النافذة التالية لحفظ اسم الجدول :

قم بكتابة (friendstbl) : ليكون ذلك اسم الجدول ، ثم اضغط علي (موافق)

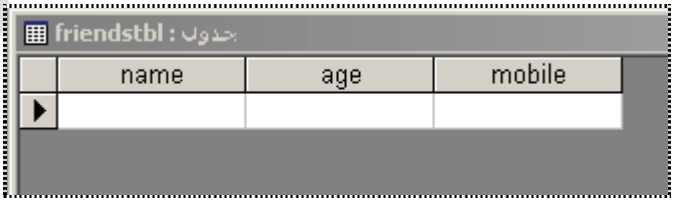


ستظهر لك الرسالة التالية ، التي تفيد عدم وجود مفتاح أساسي (Primary Key) . تجاهل هذه الرسالة وذلك بالضغط على (NO) :



ستجد إنه تمت إضافة الجدول الذي أنشأته إلى النافذة ، فقم بفتحه :

أمامك الآن الحقول الخاصة بالأعمدة فارغة ، فقم بكتابة ما تشاء من بيانات :

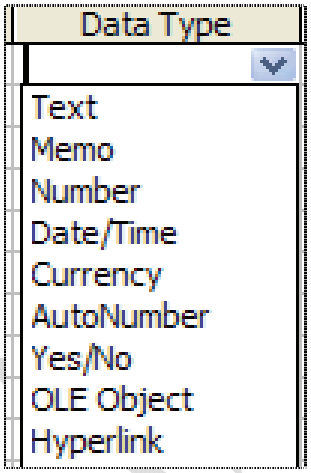


وقد قمنا بكتابة سجلين ، كما ترى في الشكل التالي :

friendstbl : جدول

name	age	mobile
Osama	21	0122849516
Muhammd	22	0100000000

وبطبيعة الحال فإن هذا أمر خاطئ ، إذ يجب علينا اختيار نوع البيانات (DataType) المناسب للحقل الذي نعمل عليه . وفى الشكل التالي مجموعة أنواع البيانات المتاحة :



إذ انه ينبغي علينا اختيار نوع بيانات (نص TEXT) للحقل النصي ، بينما نختار للحقل الرقمي نوع بيانات (رقم Number) ، أما الحقل الذي سيسجل فيه التاريخ فيجب أن نختار له نوع بيانات (تاريخ Date) .

الآن سوف نقوم بتصميم قاعدة البيانات ، التي سوف نعمل عليها في الـ Visual Basic ، وستكون قاعدة بيانات خاصة بالموظفين تحتوي على الحقول التالية :

EMPNO
ENAME
JOB
MGR
HIREDATE
SAL
COMM
DEPTNO

وسوف يكون نوع بيانات الحقول على النحو التالي :

EMPNO	NUMBER

ENAME	TEXT
JOB	TEXT
MGR	NUMBER
HIREDATE	DATE
SAL	NUMBER
COMM	NUMBER
DEPTNO	NUMBER

ملحوظة :

تذكر أنه يجب القيام بضبط الخاصية (NULL) ، بحيث لا يسمح بترك حقول فارغة ، خاصة الحقول الهامة .

الآن قم بحفظ الجدول تحت اسم (emp) ، وتذكر إدخال مجموعة من السجلات حتي تختبر معالجتها من خلال التطبيق ، الذي سنقوم بإنشائه بعد قليل .

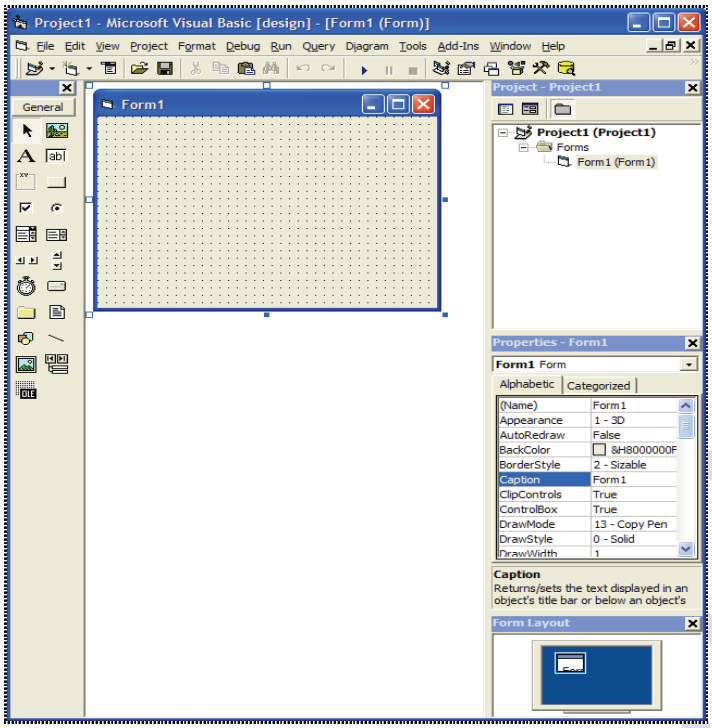
لنبدأ – الآن – بتصميم تطبيق (Visual Basic) ، يقوم بمعالجة قاعدة البيانات التي قمنا بتصميمها .

التطبيق بلغة برمجة :

قم بفتح مشروع جديد في الفيجوال بيسيك من النوع (Standard EXE) كما هو مبين في الشكل التالي :

وفيما يلي سوف نتناول شرح أهم أقسام واجهة العمل بشكل مختصر وسريع ، لكي نزيد إيضاح الأمر لمن لم يسبق له أن رأى واجهة العمل في الفيجوال بيسيك .

عندما قمنا بإنشاء المشروع الجديد ظهرت لنا واجهة العمل الافتراضية للفيجوال بيسيك كما في الشكل التالي :



فلنتناول الآن أقسام واجهة العمل هذه :

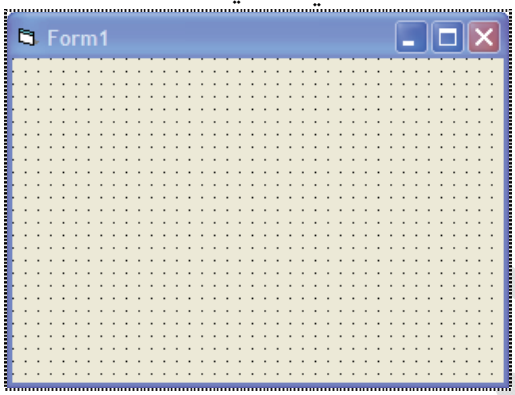
صندوق الأدوات Toll Box :

هذا الصندوق هو الذي يتيح لنا إدراج الأداة التي نريدها ، علي النموذج الذي نقوم بتصميمه . فهو يحتوى - على سبيل المثال - على أداة العنوان (Label) وأداة النص (Text)وأداة الزر(Command) وأداة البيانات (Data Control) وأدوات بيانات أخرى متقدمة مثل (ADODC) و أداة عرض البيانات الشبكية(Flex Grid) :

وبطبيعة الحال فإن صندوق الأدوات يحتوى أدوات أكثر مما ذكرنا (سابقاً)، لكننا نذكر ما سوف يفيدنا في مشروعنا التالي.

تصميم النموذج Form :

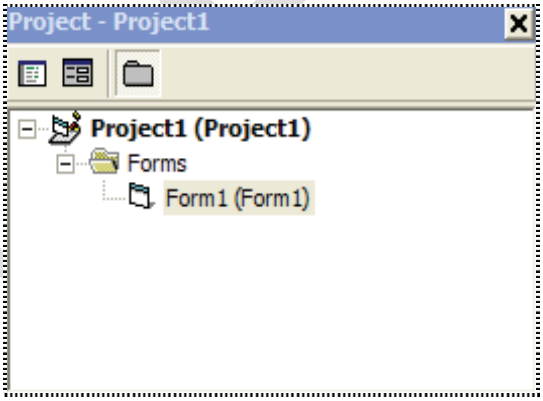
بالإمكان رسم وتصميم الأدوات على النموذج التالي ، الذي يتيح لنا تنفيذ الوظائف المطلوبة ، وفقا لحاجتنا :



فهذا النموذج يمكننا من إدراج أية أداة موجودة في صندوق الأدوات ، لتظهر على شكل النموذج السابق ، ولكن نستخدمها في الوظيفة التي نحددها .

متصفح المشروع Project Explorer :

يحتوي هذا المتصفح على النماذج المصاحبة للمشروع ، وعلى الوحدات والفصائل والملفات الأخرى :



نافذة الخصائص Properties Window

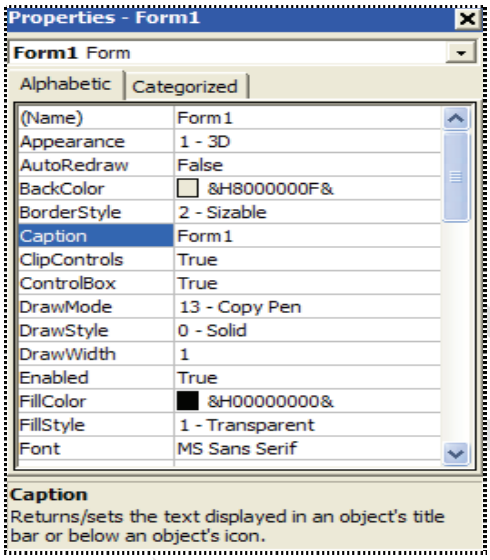
لكي نوضح معنى خصائص الأدوات نضرب المثال التالي :

لو أنك أردت إنشاء أداة نص (Text) ، فإن عليك أن تضغط زر الفارة الأيمن مرتين (Double Click) حتى يمكنك إضافة الأداة على النموذج .

ولكن..

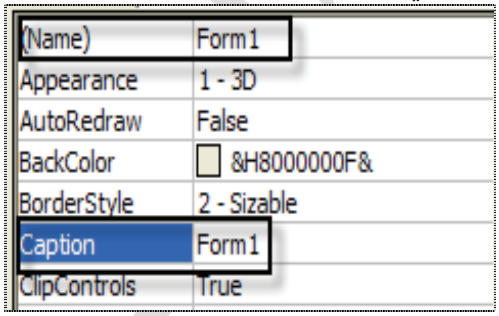
هب أنك تريد تغيير لون النص المكتوب داخل الأداة (أو لون الأداة ذاتها) لكي يتوافق مع الشكل العام للنموذج ، فكيف يمكنك فعل ذلك ؟

بطبيعة الحال يمكنك فعل ذلك من خلال نافذة الخصائص (Properties Window)

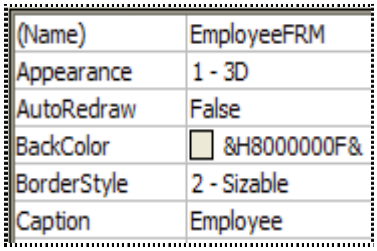


إذن فتعال – الآن – لنقوم معا بتغيير اسم النموذج وعنوانه .

قم بالضغط علي اسم النموذج وعنوانه في مكان التصميم ، ثم توجه إلى نافذة الخصائص :

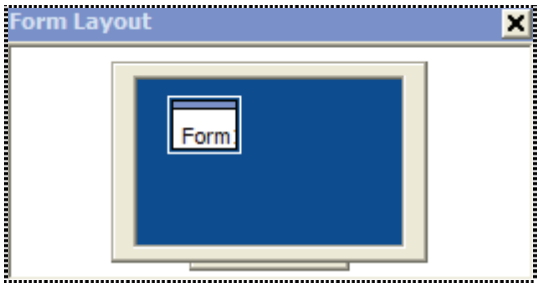


سوف تلاحظ وجود خاصيتي (Name) و (Caption) ، قم بضبط الأولى (EmployeeFRM) والثانية (Employee) ، ونعني بكلمة ضبط كتابة العبارة من خلال لوحة المفاتيح :



موقع النموذج Form Layout

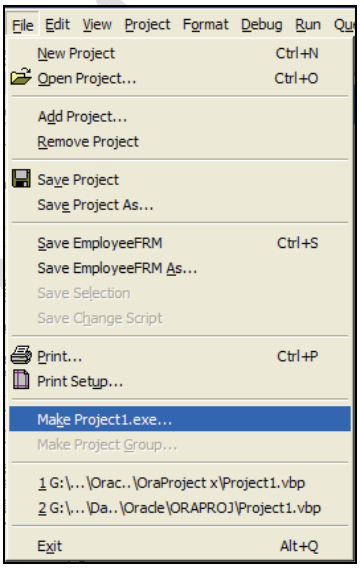
هذه النافذة تتيح لنا التحكم في مكان النموذج على الشاشة ، بحيث يمكن أن تظهره في أقصى اليمين أو اليسار، وفي الأعلى أو الأسفل، أو في منتصف الشاشة ، إذا أردنا .



كيفية تجميع المشروع :

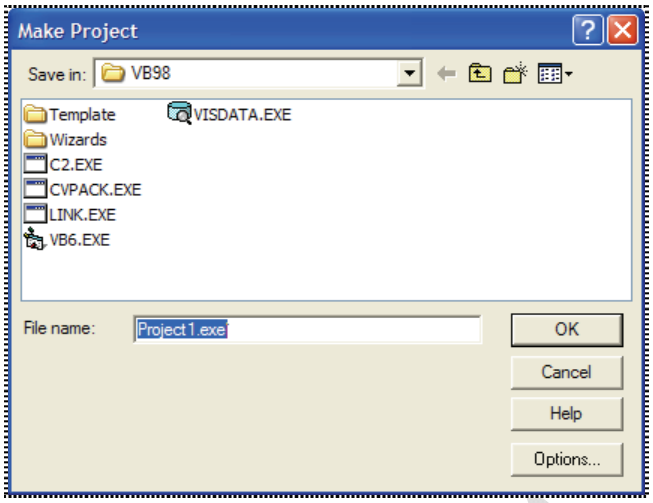
بعد أن تنتهي من تصميم المشروع يمكنك تحويله إلى ملف تنفيذي (EXE) وذلك وفقا للخطوات التالية :

توجه إلى القائمة (File) ثم إلى (Make project1.exe) :

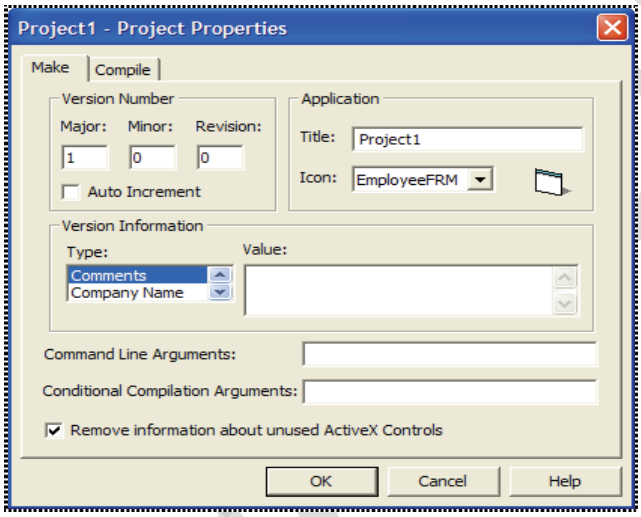


سوف يطلب منك البرنامج اختيار مكان التجميع، اضغط(ok)، ليصبح لك الآن ملفك التنفيذي.. ولكن سنعرض الأمر بشكل آخر .

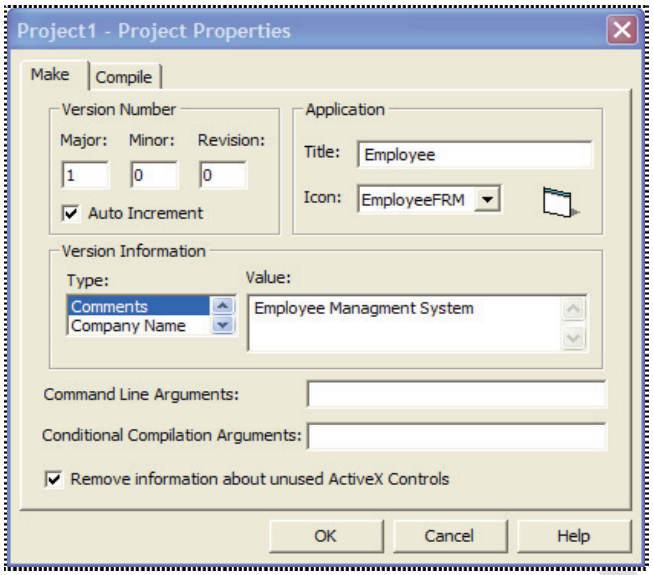
فعند توجهك إلى القائمة (File) ثم إلى (Make project1.Exe) سوف تظهر لك النافذة التالية :



اضغط علي الزر(Options) ، لتظهر لك النافذة التالية :



الآن.. قم بكتابة (Employe) في الخانة (Title)، ثم اكتب ما تريده من خلال التطبيق في الخانة (Comments) :



ملحوظة :

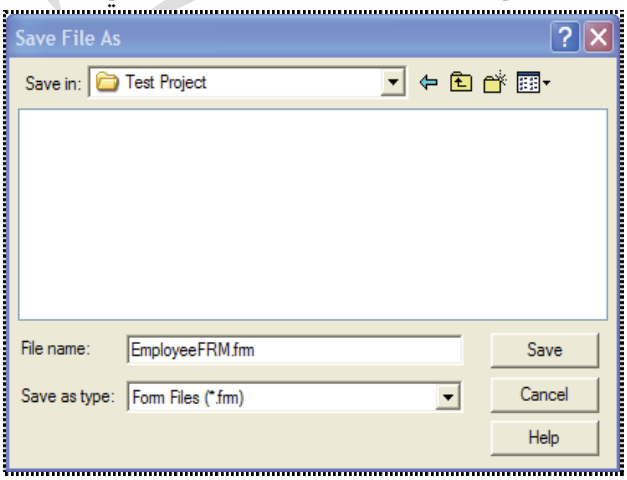
الخطوات التي سبق ذكرها تتعلق بضبط خصائص المشروع وبياناته العامة .

حفظ المشروع :

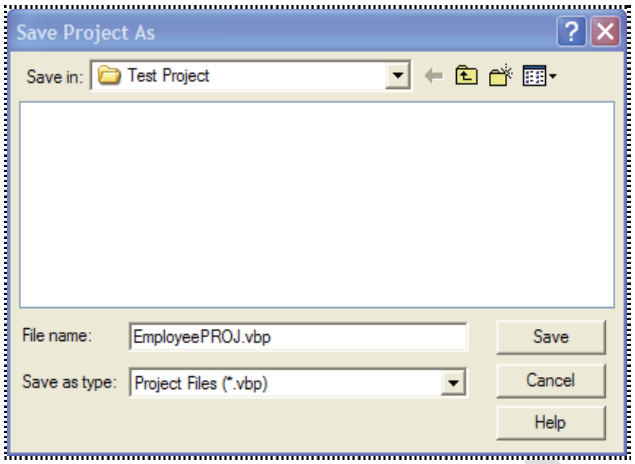
لكي يعمل مشروعك بشكل صحيح و جيد ، يجب عليك بين الحين والآخر أن تقوم بحفظ آخر التعديلات والإضافات التي تقوم بها .

ويتم ذلك بالضغط علي الأيقونة (Save) في شريط الأدوات :

وحين تقوم بعملية الحفظ للمرة الأولى سوف تظهر لك النافذة التالية ، لكي تحدد مكان حفظ النموذج :



بعدها سوف تظهر لك النافذة التالية ، لتحديد مكان حفظ الملف الخاص بالمشروع :



مشروع موظفي الشركة
Employee Management System

بعد أن تناولنا كيفية معالجة البيانات فيما سبق ، سوف نتناول الآن كيفية معالجة تلك البيانات ، من خلال تقنيتي(SQL) و (ADO) ، عبر (Visual Basic).

هدف المشروع :

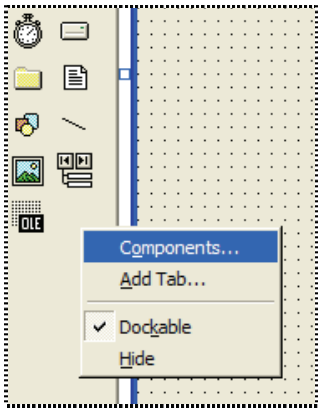
يهدف المشروع إلى تصميم نموذج يقوم بمعالجة البيانات الخاصة بجدول الموظفين(EMP) وكيفية القيام بعمليات التعديل والإضافة والحذف والبحث .

بدء العمل :

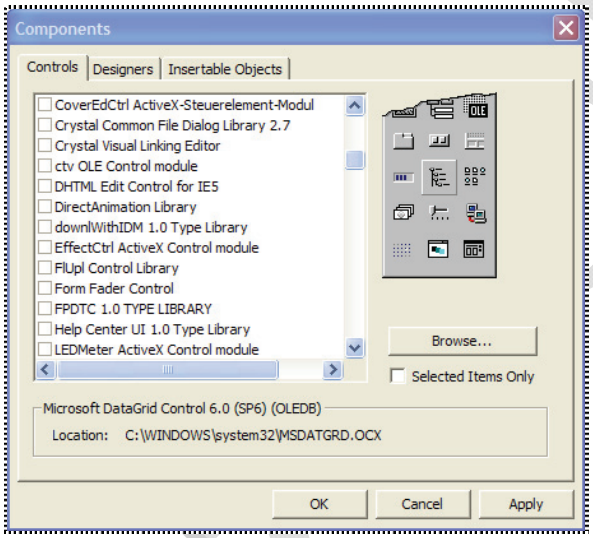
قبل البدء في تصميم هذا النموذج علينا أن نضيف مجموعة أخرى من الأدوات والمكتبات التي من شأنها أن تساعدنا في عملية تصميم المشروع ، الذي نحن بصدده .

إضافة الأدوات :

لكي تقوم بإضافة الأدوات التي ذكرناها قم بالضغط على الزر الأيمن للفارة في أي مكان فارغ في صندوق الأدوات ، وستظهر لك النافذة التالية ، فاختر منها (Components) .



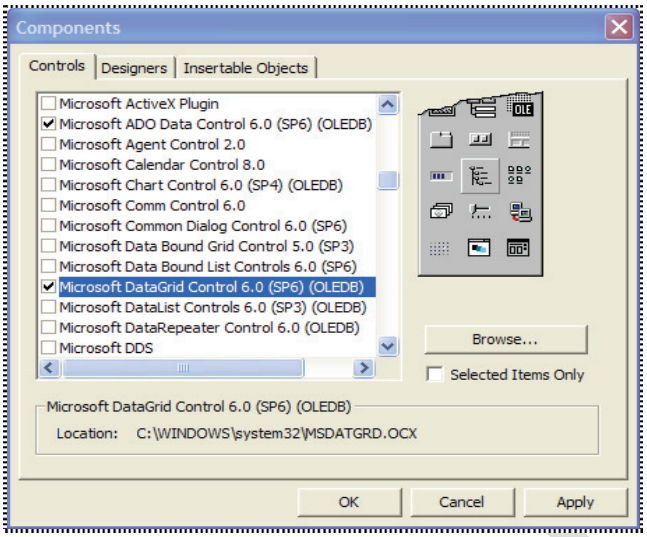
وستظهر لك النافذة التالية :



وهذه النافذة تتيح لنا إضافة مجموعة من الأدوات (التي لا تظهر تلقائياً على صندوق الأدوات) ، وذلك لكي نستخدمها في تطبيقنا الذي نحن بصدده :

الآن أنزل باللفافة (scroll) ، حتى تصل إلى الأداتين التاليتين :

وبطبيعة الحال.. عليك أن تؤشر عليهما معاً



فإذا فعلت ذلك سوف تجد أنك قد قمت بالفعل بإضافة أداتين إلى صندوق الأدوات :

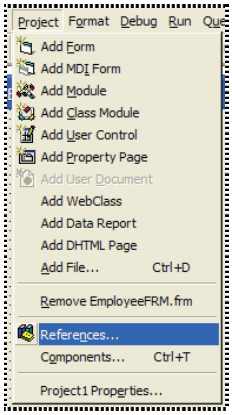
ولن نتعامل الآن مع هاتين الأداتين ، فهذا له وقت آخر ، لكننا سوف نعمل الآن على إضافة مجموعة جديدة من المكتبات ، أو كما يسمونها المراجع (references).

إضافة المراجع References

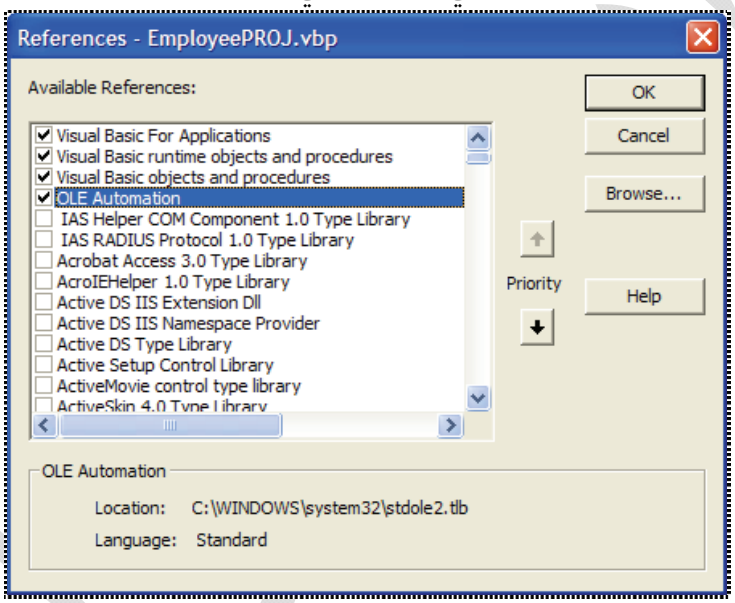
هذه المراجع (المكتبات) تساعدنا على كتابة كود أقل حجماً ، كما تساعدنا على الاستفادة من الأكواد الفرعية المختبئة .

فعلى سبيل المثال تساعدنا المكتبة Microsoft ActiveX Data Objects 2.8 على التعامل مع قاعدة البيانات بشكل بسيط وبسرعة عالية ، وذلك دون الحاجة إلى كتابه الكثير من الأكواد ، أو الاعتماد على الأدوات

الجاهزة . ولكي تصل إلى نافذة إضافة المكتبات اضغط على (references) الموجودة في القائمة (Project) ، كما في الشكل التالي :



وستظهر لنا النافذة (References) . كما في الشكل التالي :



انزل إلى الأسفل لتجد المكتبة التالية :

ActiveX Data Objects 2.8

ملحوظة :

هناك أكثر من إصدار للمكتبة الأولي ، لذا عليك التأكد من اختيار المكتبة (2.8) على وجه التحديد .

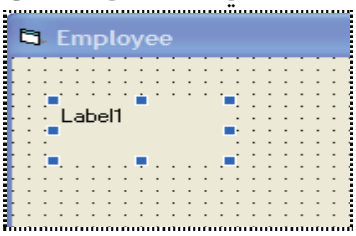
تصميم النموذج

الآن سوف نبدأ في الجزء المفضل دائما وهو الأدوات على النموذج ،

ولنبدأ أولاً في وضع أدوات العناوين .

أداة العنوان :

يكفي الضغط عليها مرتين في صندوق الأدوات لكي تظهر على النموذج ، كما ترى في الشكل التالي :



الآن..توجه إلى نافذة الخصائص ؛ ثم قم بكتابة (Employee Number) في خاصية (Caption) :

ثم قم بإضافة 7 أدوات أخرى إلى النموذج ، وقم بضبط خصائصها لتناسب الشكل التالي :

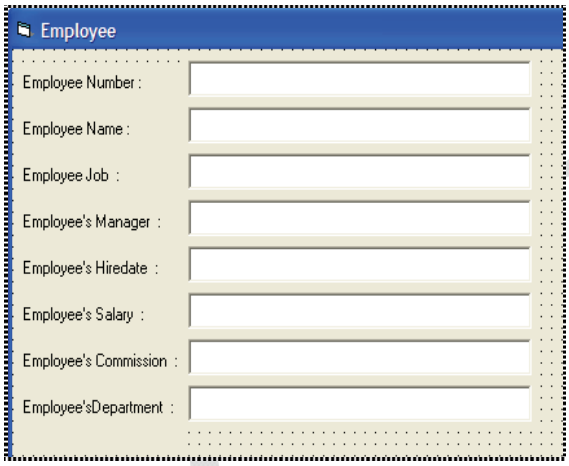
توجه إلى صندوق الأدوات ثم قم بإدراج 8 أدوات نص Text :

قم بضبط خاصية (Name) في أدوات النص على النحو التالي :

EMPNOTXT
ENAMETXT
JOBTXT
MGRTXT
HIREDATETXT
SALTXT
COMMTXT
DEPTNOTXT

والآن.. عليك القيام بمحو ما هو مكتوب في خاصية (Text) ، بحيث يبقى محتوى أداة النص فارغا ، ثم قم بوضع أداة النص أمام أداة العنوان المناسبة لها . فعلى سبيل المثال ، عليك أن تضع أداة النص (EMPNOTXT) أمام أداة عنوان(Employee Number)..

وهكذا ، المهم أن يصبح النموذج كما في الشكل التالي :



من جديد توجه إلى صندوق الأدوات ، ثم قم بإضافة أربع أدوات (زر) إلى النموذج :

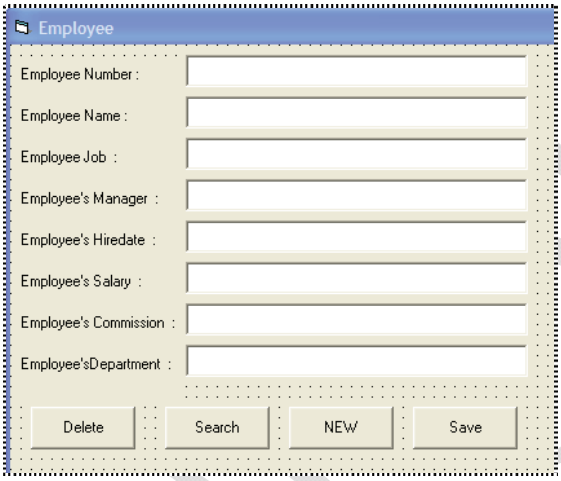
ثم قم بضبط خاصية (Name)، وفقا للترتيب التالي :

SAVEBCMD
NEWCMD
FINDCMD
DELCMD

ثم خاصية (Caption)، وفقا للترتيب التالي :

Save
NEW
Search
Delete

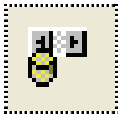
والآن قم بوضعها جميعا في الموقع الذي تظهر به في الشكل التالي :



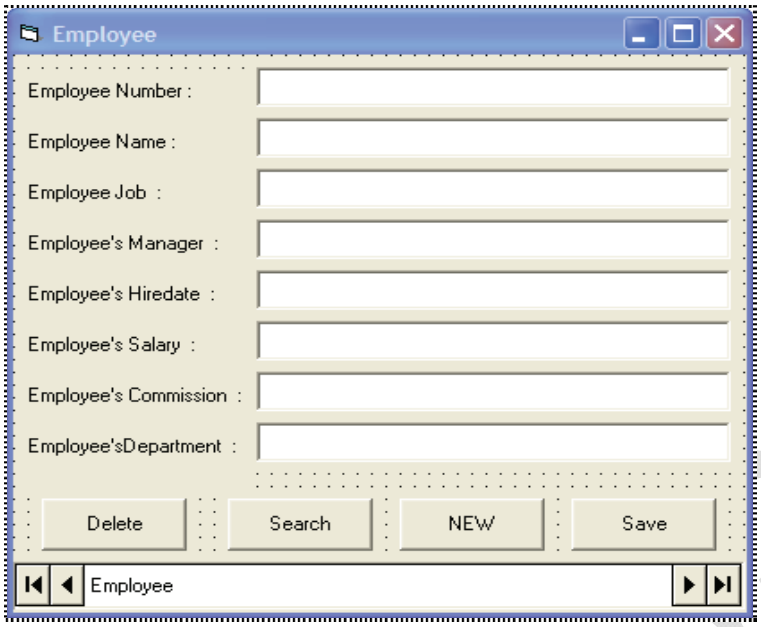
إضافة أداة البيانات :

نعود الآن إلى الأداتين ، اللتين سبق أن أضفناهما إلى صندوق الأدوات .

اضغط (مرتين) على الأداة (ADODC) :



ثم قم بضبط خاصية (Name) بتحويلها إلى (EmployeeADODC) ، ثم خاصية (Caption) إلى (Employee)، وضعهما في الموقع الذي تظهران به في الشكل التالي :



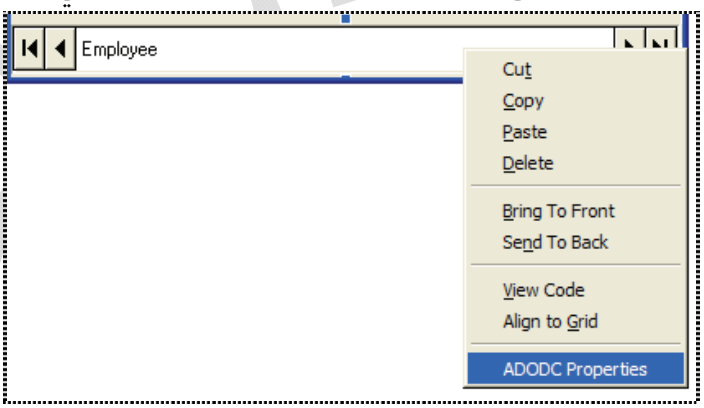
ملحوظة :

هذه الأداة سوف تسمح بالتوجه إلى السجل التالي والسجل السابق ، كما ستسمح لنا بالعودة إلى أول السجلات والذهاب إلى آخرها .

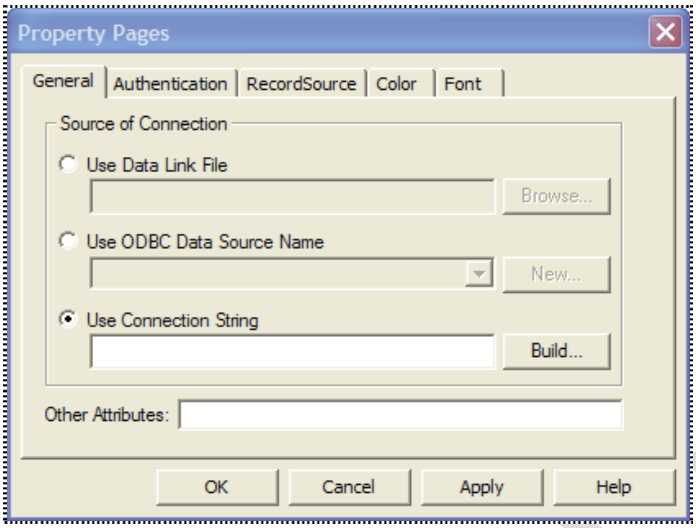
الاتصال مع قاعدة البيانات - ACCESS

سوف نتناول الآن أهم الخطوات ، وهى ربط التطبيق الذي نعمل عليه بقاعدة البيانات ، ومن ثم ربطها بجدول الموظفين .

وللقيام بذلك اضغط بزر الفارة الأيمن على الأداة (EmployeeADODC) علي النموذج:



وسوف تظهر لك النافذة التالية ، التي تختص بضبط خصائص الاتصال بقواعد البيانات المختلفة :

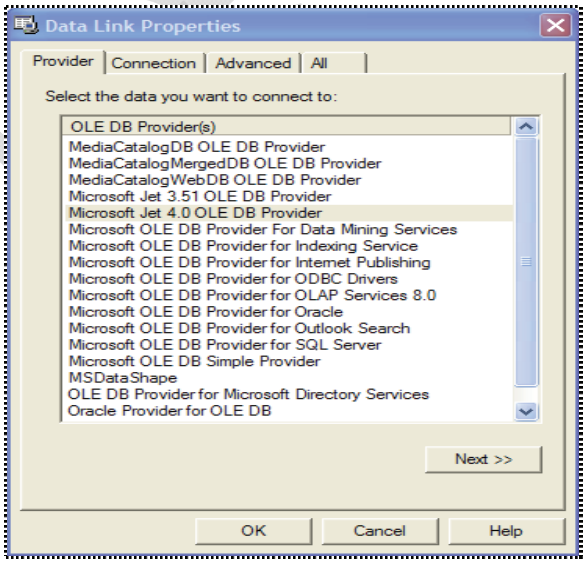


ملحوظة :

الخطوات السابق ذكرها يمكن تطبيقها على كافة أنواع قواعد البيانات ، سواءً كانت (MYSQL) أو (ACCESS Database File) أو (SL Server Database)، حيث أن التقنية المستخدمة واحدة هي (ADO) .

ولنعد الآن إلى ما كنا بسبيله ..

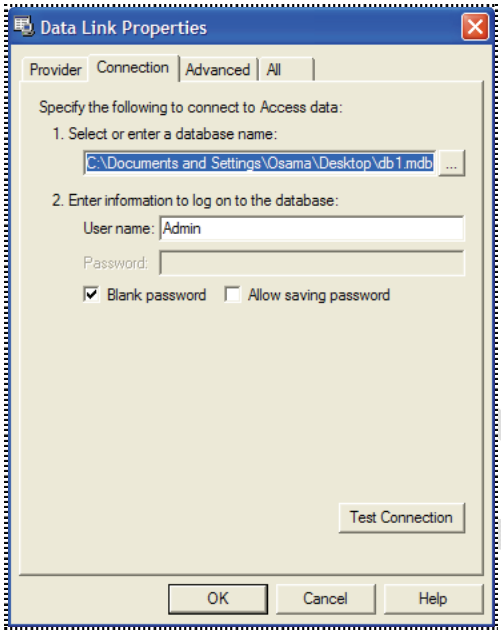
الضغط على الزر(Build) الذي تراه في الشكل السابق، وستظهر لك نافذة ربط قاعدة البيانات، كما في الشكل التالي:



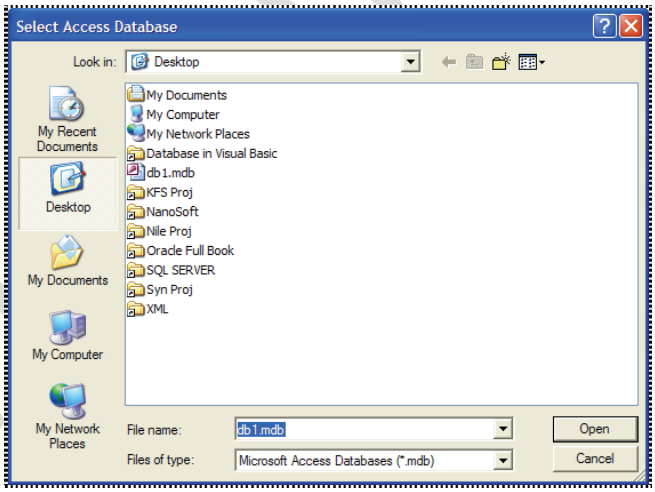
اختر (Jet 4.0 OLE DB Provider) ، ثم اضغط (Next) .

وسنقوم الآن بوضع اعتبارات الدخول والاتصال .

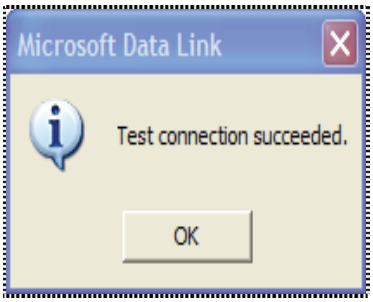
اضغط على الزر المجاور للخانة التي أسفل العبارة (Select or enter a database name) كخطوة أولى ، كما في الشكل التالي :



وستظهر لك النافذة التالية :

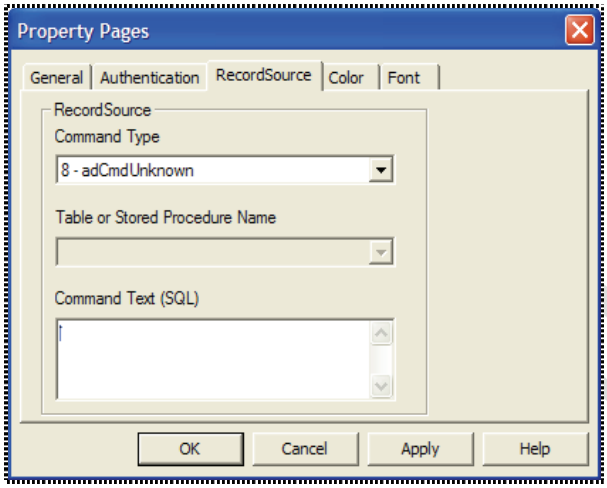


ابحث الآن عن قاعدة البيانات التي سبق أن أنشأناها وذلك من خلال الضغط على الزر Open . ولنجرب عملية الاتصال اضغط على الزر(Test Connection) :

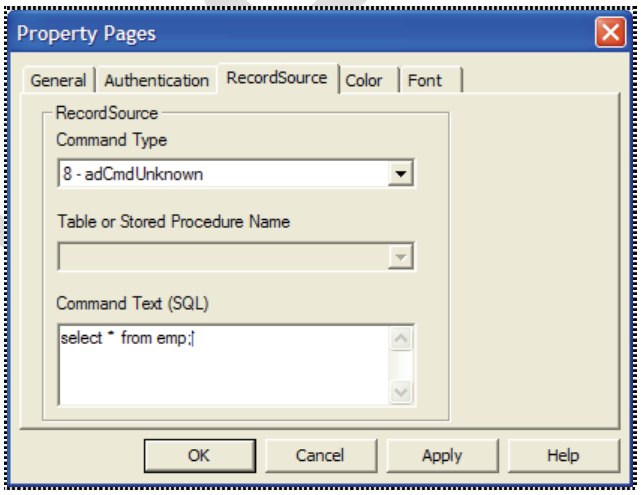


ونأمل أن تظهر لك الرسالة السابقة !

توجه إلى اللسان(Record Source) ، واختر الجدول الذي ستعمل عليه ، وهو بالطبع جدول الموظفين :

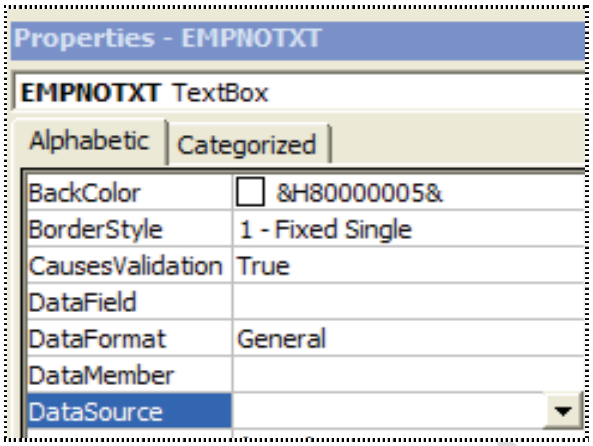


قم بكتابة جملة استعلام في الخانة (Command) ، ولتكن جملة بسيطة مثل : select * (from emp ، لأننا سوف نقوم بتغييرها بعد قليل :

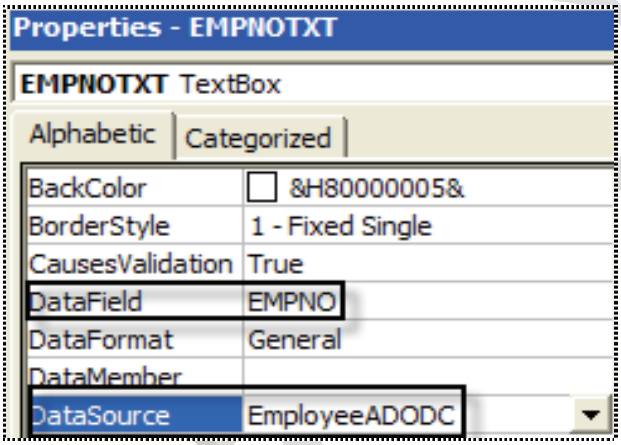


الآن اضغط (Ok) ، وبهذا يكون العمل قد انتهى .

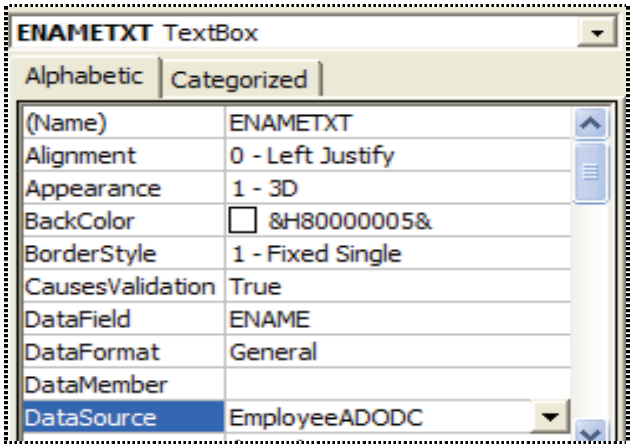
والآن..
لنتوجه إلى أدوات النص التي سنركز جهدنا عليها..
أمسك بأداة النص الأولي (EMPNOTXT) على النموذج ، ثم توجه إلى نافذة الخصائص. وقف على الخاصية (Data Source) :



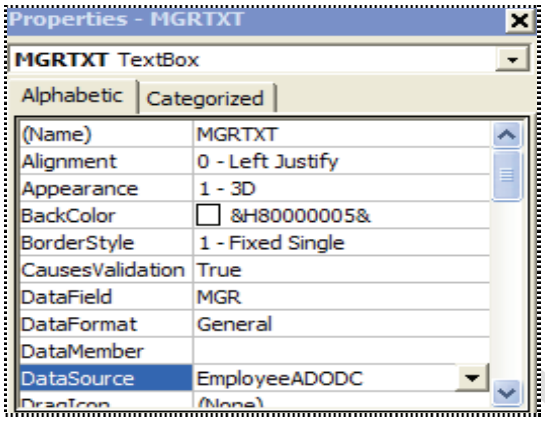
قم بضبط الخاصية (Data Source) إلى الأداة (Employee ADODC) ، ثم اكتب الحقل (EMPNO) في الخاصية (DataField) :



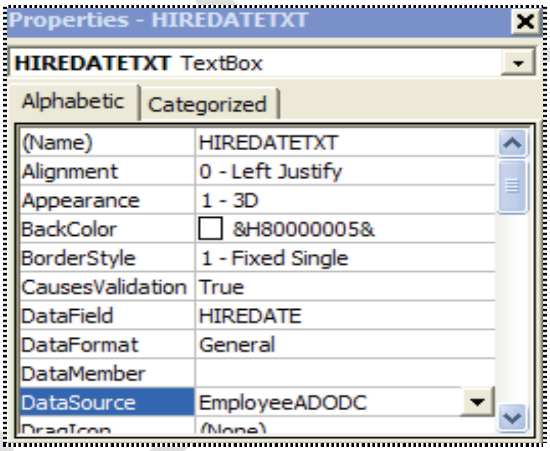
ثم..في الأداة (ENAMETXT) قم بضبط الخاصية الأولى (Data Source) إلى الأداة (Employee ADODC) ، ثم اكتب الحقل (ENAME) في الخاصية DataField :



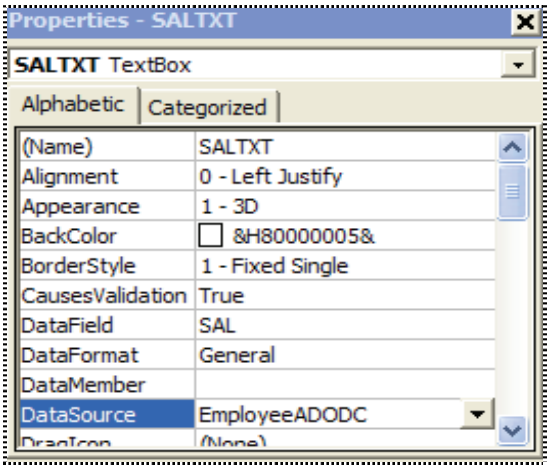
ثم..في الأداة (MGRTXT) قم بضبط الخاصية الأولى (DataSource) إلى الأداة (Employee ADODC) ، ثم اكتب الحقل (MGR) في الخاصية (DataField) :



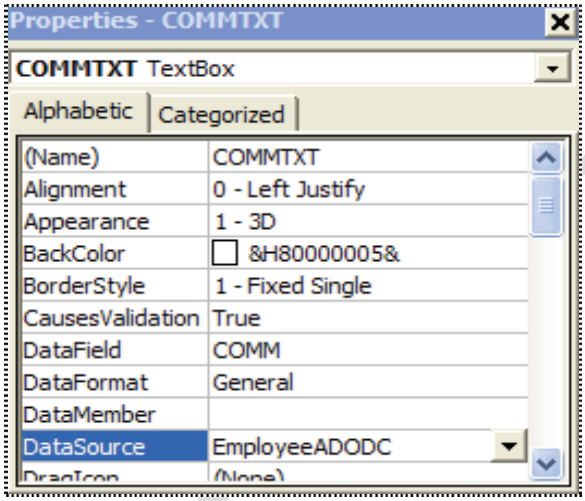
ثم.. في الأداة (HIREDATETXT) قم بضبط الخاصية الأولى (Data Source) إلى الأداة (ADODC) ، ثم اكتب الحقل (HIREDATE) في الخاصية (DataField) .



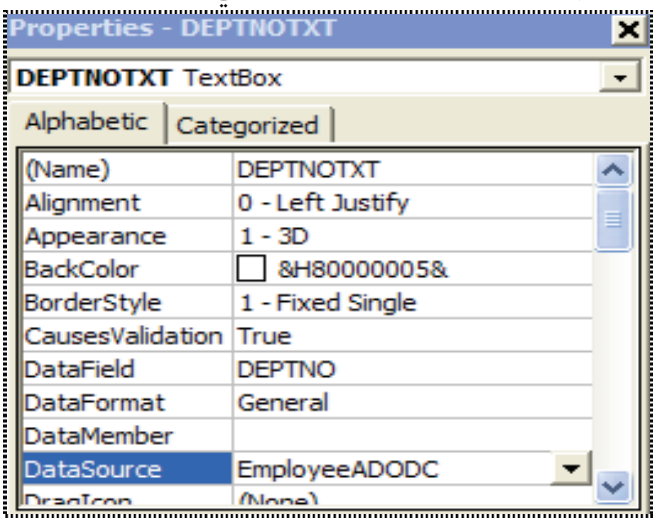
ثم.. في الأداة (SALTXT) قم بضبط الخاصية الأولى (Data Source) إلى الأداة (EmployeeADODC) ، ثم اكتب الحقل (SAL) في الخاصية (DataField) :



ثم..في الأداة (COMMTXT) قم بضبط الخاصية الأولى (DataSource) إلى الأداة (EmployeeADODC) ، ثم اكتب الحقل (COMM) في الخاصية DataField



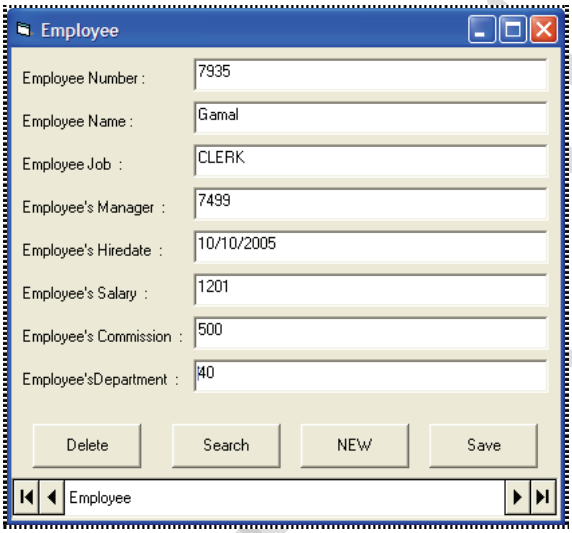
ثم..في الأداة (DEPTNOTXT) قم بضبط الخاصية الأولى (DataSource) إلى الأداة (EmployeeADODC) ، ثم اكتب الحقل (DEPTNO) في الخاصية (DataField)



تشغيل التطبيق RUN

بعد أن أنهينا هذا الجزء البسيط من البرنامج ، لاشك أننا نرغب في مشاهدته وهو يعمل وأن نتأكد من صواب ما قمنا به . وذلك يتحقق كما يلي :

اضغط على الزر(Run)من شريط الأدوات ، أو الزر **F5** من لوحة المفاتيح ، وسيظهر لك التطبيق لتجد أنه قد تم تحميل البيانات الخاصة بجدول الموظفين بشكل ناجح :



قم بتجربة أزرار أداة (**EmployeeADODC**) ، بحيث يمكنك التنقل بين السجل السابق و التالي وبين بداية السجلات و نهايتها :

وهكذا ، نكون قد إنتهينا من جزء هام ، ورغم أنه غير كاف إلا أنه أثبت لنا سهولة الاتصال بقواعد بيانات (**Access**) من خلال لغة (فيجوال بيسيك 6) .

ملحوظة :

يمكنك الاستفادة من كتاب (كل شيء عن قواعد البيانات) في (فيجوال بيسيك 8) ، فهو يتناول (بشكل تفصيلي) كيفية التعامل مع كافة أنواع قواعد البيانات (أوراكل) ، من خلال (الفيجوال بيسيك 2005) .

***

بعض المهام المتقدمة

سوف ندخل الآن إلى أبعاد أعمق في مشروعنا ، وسوف نقوم بتصميم باقي المهام المتبقية ، فنقوم بتحديد إجراءات الضغط على زر الحفظ ، وزر الإضافة ، وكذلك على زر البحث وزر الحذف .

زر إضافة سجل جديد :

توجه إلى صفحة الكود ، وذلك من خلال الضغط على أي أداة في النموذج ، ثم توجه إلى أعلى الكود واكتب الكود التالي :

```
:Code
Dim NEWR As Boolean
```

وبذلك تكون قد قمت بإنشاء متغير باسم (**NEWR**) ، ومن النوع (**Boolean**) ، أي إنه سيقوم بتخزين قيمة من اثنتين : إما (**True**) أو (**False**) . (وإذا كنت من المبتدئين فالشكل التالي يوضح مكان الكود في نافذة الكود :

الآن..

توجه إلى نافذة تصميم الكود مرة أخرى (أو بإمكانك اختيار الأداة من القائمة الموجودة بأعلى نافذة الكود) ، ثم قم بالضغط مرتين على أداة الزر (NEWCMD) ، ثم اكتب الكود التالي داخل الإجراء الخاص بهذه الأداة :

```
:Code
NEWR = True

EMPNOTXT.Text = ""

ENAMETXT.Text = ""

JOBTXT.Text = ""

MGRTXT.Text = ""

HIREDATETXT.Text = ""

SALTXT.Text = ""

COMMTXT.Text = ""

DEPTNOTXT.Text = ""
```

زر حفظ التغييرات :

قم بالضغط مرتين على الزر (SAVECMD) لتتوجه إلى الإجراء الفرعي الخاص بالضغط على هذه الأداة ، ثم اكتب الكود التالي :

:Code
```
On Error GoTo er:

If NEWR = True Then

rs.AddNew

End If

rs!EMPNO = EMPNOTXT.Text

rs!ENAME = ENAMETXT.Text

rs!JOB = JOBTXT.Text

rs!MGR = MGRTXT.Text

rs!HIREDATE = HIREDATETXT.Text

rs!SAL = SALTXT.Text

rs!COMM = COMMTXT.Text

rs!DEPTNO = DEPTNOTXT.Text

rs.Update

MsgBox "saved Successfully"

NEWR = False

er:
MsgBox "Mission Failed !!!!"
```

زر البحث عن سجل :

قم بالضغط مرتين على الزر (FINDCME) ، لتتوجه إلى الإجراء الفرعي الخاص بالضغط على هذه الأداة ، ثم اكتب الكود التالي :

:Code
```
Dim x As String

x = InputBox("what u lookin for ?")

rs.MoveFirst

rs.Find "EMPNO='" + x + "'"
```

```
EMPNOTXT.Text = rs!EMPNO
ENAMETXT.Text = rs!ENAME
JOBTXT.Text = rs!JOB
MGRTXT.Text = rs!MGR
HIREDATETXT.Text = rs!HIREDATE
SALTXT.Text = rs!SAL
COMMTXT.Text = rs!COMM
DEPTNOTXT.Text = rs!DEPTNO
```

زر حذف سجل :

قم بالضغط مرتين على الزر (**DELETECMD**) لتتوجه إلى الإجراء الفرعي الخاص بالضغط على هذه الأداة ، ثم اكتب الكود التالي :

Code:

```
Dim x
x = EMPNOTXT.Text
rs.Find "EMPNO='" + x + "'"
rs.Delete
MsgBox "Deleted Successfully"
```

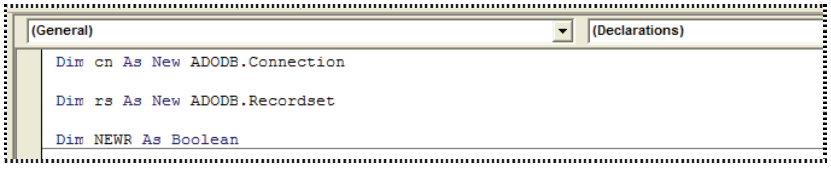
وهكذا.. نكون قد إنتهينا من العمل .. ولاشك أنه توجد بعض الأخطاء والمشاكل التي ستواجهنا ، وسوف نتناول ذلك في الصفحات القادمة ، و لكن قبل ذلك سوف نشرح بإيجاز كيفية الاتصال بقاعدة بيانات (أكسيس) ، من خلال الكود فحسب ، ومن خلال والمكتبات ، التي سبق أن تناولنا الحديث عنها .

في جزء من التصريحات العامة قم بإضافة الكود التالي :

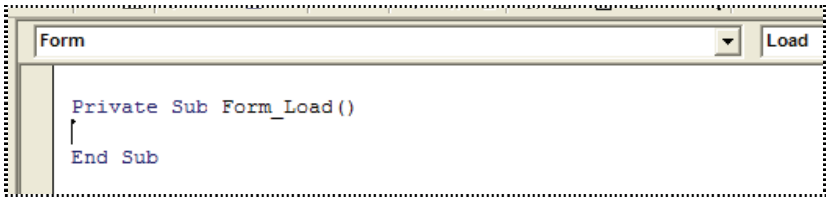
Code:

```
Global DB As Database
Global RS As Recordset
```

أي أن الكود - بأكمله - سيكون في التصريحات العامة على هذا النحو :



اضغط – الآن - في أي منطقة فارغة في نافذة تصميم النموذج مرتين لكي تتوجه إلى الإجراء (Form_Load) :



ثم قم بكتابة الكود التالي :

Code:

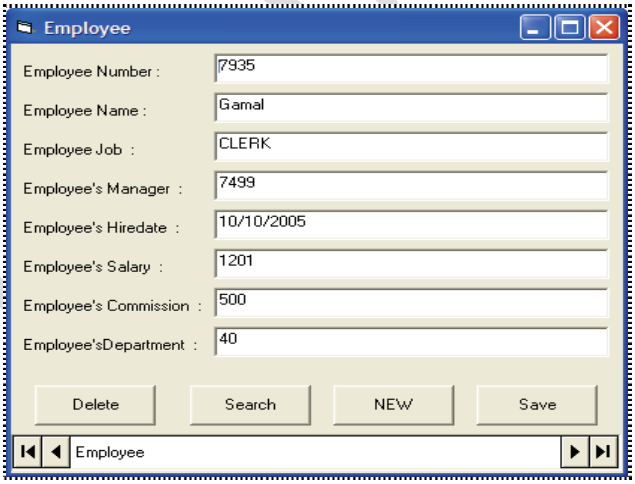
```
Set DB = OpenDatabase(App.Path & "/DataBase/PHDataBase.mdb", False, False, "MS Access;PWD=561920why")

Set RS = DB.OpenRecordset("emp", 2)
```

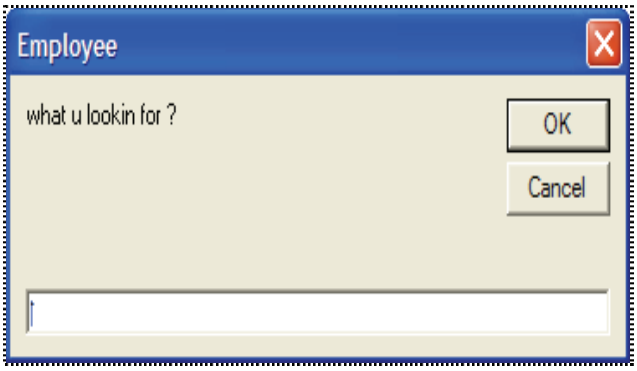
مشكلة Null :

بدل أن نقوم بشرح هذه المشكلة ، سنجعلك تواجهها بنفسك .

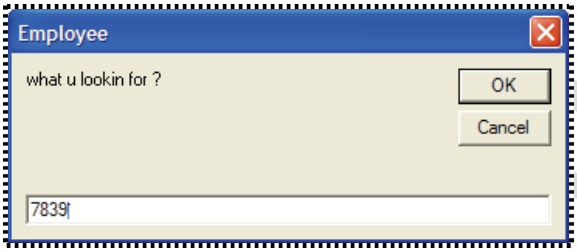
قم الآن بتشغيل التطبيق ، وسوف تجد أن هذه المشكلة ستواجهك إذا كنت تستخدم قواعد بيانات من نوع آخر مثل (**Oracle**) :



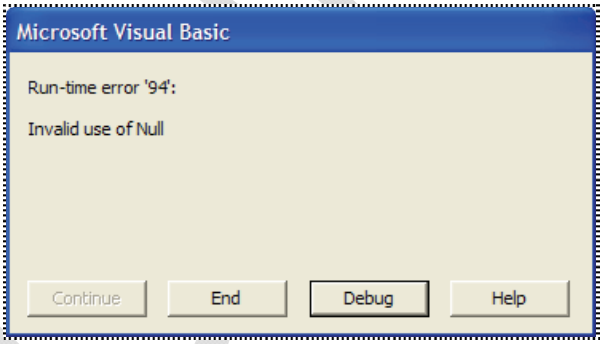
ثم قم بالضغط على زر البحث :



الآن.. اكتب الرقم 7839 ، ثم اضغط (**Ok**) :



وها هي رسالة الخطأ !



لقد كان السبب في حدوث ذلك الخطأ ، أن الحقل (**MGR**) يحمل قيمة (**NULL**) لم نستخدمها استخداما صحيحا :

```
JOBTXT.Text = rs!JOB

MGRTXT.Text = rs!MGR

HIREDATETXT.Text = rs!HIREDATE
```

وهذه المشكلة من الممكن حلها بأكثر من طريقة ، لكننا نفضل حلها من خلال جملة الاستعلام (**NULL**) ، التي تجعلنا نعوض عن القيمة (**NULL**) بأي أرقام أخرى :

أي إننا نريد تحويل جملة الاستعلام التالية :

```
Select * From Emp ;
```

إلى الجملة التالية :

```
select EMPNO,ENAME,JOB,nvl(MGR,0) as
MGR,HIREDATE,SAL,nvl(COMM,0) as COMM,DEPTNO
from emp;
```

وهناك موضعان يجب علينا القيام بتغيير الجملة فيهما :

المكان الأول :

هو خانة (Command) ، في نافذة خصائص الأداة (EmployeeADODC) .

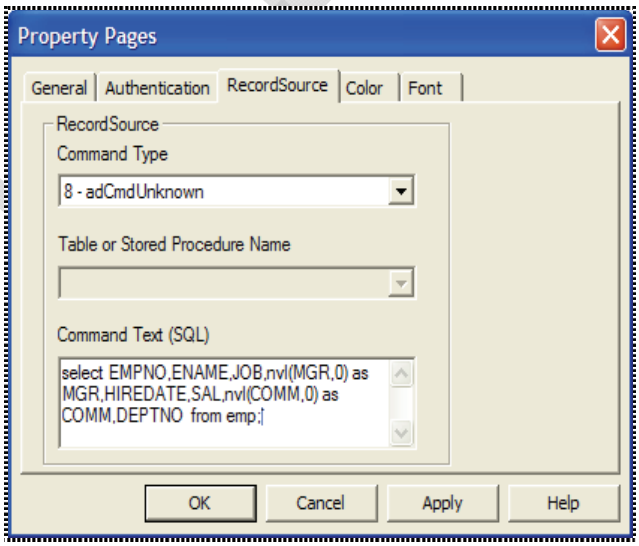
والمكان الثاني :

هو في الإجراء (Form_Load) عند فتح مصدر السجلات (RS) .

ويتم ذلك على النحو التالي :

قم بالضغط على الأداة (Employee ADODC) بالزر الأيمن للفارة . وسوف تظهر لك نافذة خصائص الأداة .

توجه إلى اللسان (Record Source) واستبدل جملة حديثة بالجملة القديمة .



أما فيما يتعلق بالمكان الثاني Form_Load عند فتح مصدر السجلات كما في الشكل التالي :

```
Private Sub Form_Load()

cn.Open "provider=oraoledb.oracle.1;password=nsync;user id=scott;"

rs.Open "select * from emp;", cn, adOpenDynamic, adLockOptimistic

End Sub
```

فهنا قم أيضا باستبدال جملة حديثة بالجملة القديمة :

```
Private Sub Form_Load()

cn.Open "provider=oraoledb.oracle.1;password=nsync;user id=scott;"

rs.Open "select EMPNO,ENAME,JOB,nvl(MGR,0) as MGR,HIREDATE,SAL,nvl(COMM,0) as COMM,DEPTNO f

End Sub
```

وهكذا.. نكون قد إنتهينا من حل هذه المشكلة بطريقة عملية ، نكون قد أنهينا تصميم تطبيق خاص بإدارة شئون الموظفين بشكل جيد .

ولكن..

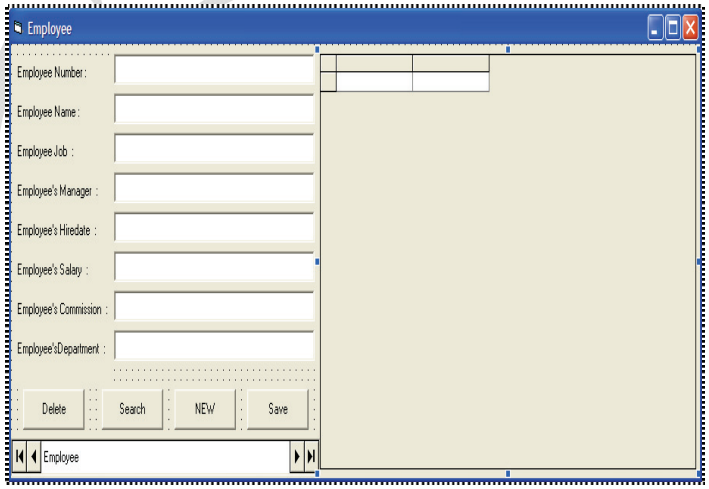
تبقي وظيفة واحدة لم نتناولها ، هي إضافة أداة شبكة بيانات .

وهذا ما سوف نتناوله في الصفحات التالية .

شبكة البيانات Data Grid :

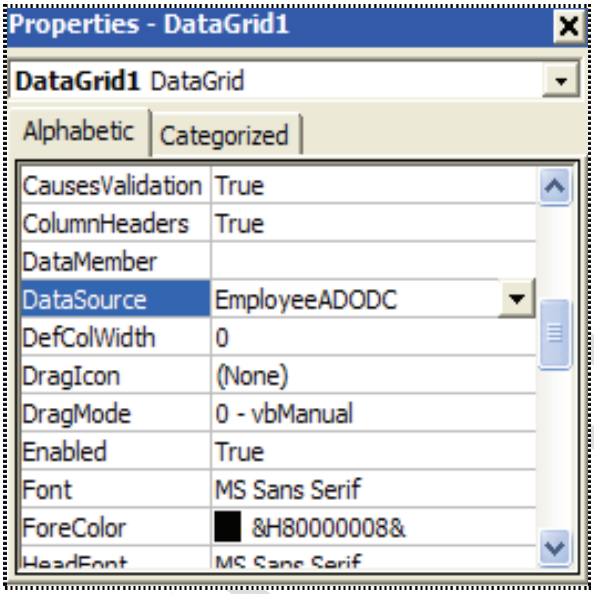
سوف نبدأ – الآن – في استخدام الأداة الثانية التي أدرجناها من صندوق (Components) ، وهي الأداة (DataGrid) :

قم بالضغط مرتين على هذه الأداة ؛ لكي تضيفها إلى النموذج ، ثم حدد موقعها ، كما هو موضح في الشكل التالي (بعد أن تقوم بتكبير حجم النموذج نفسه ليلاءم حجم أداة شبكة البيانات) :

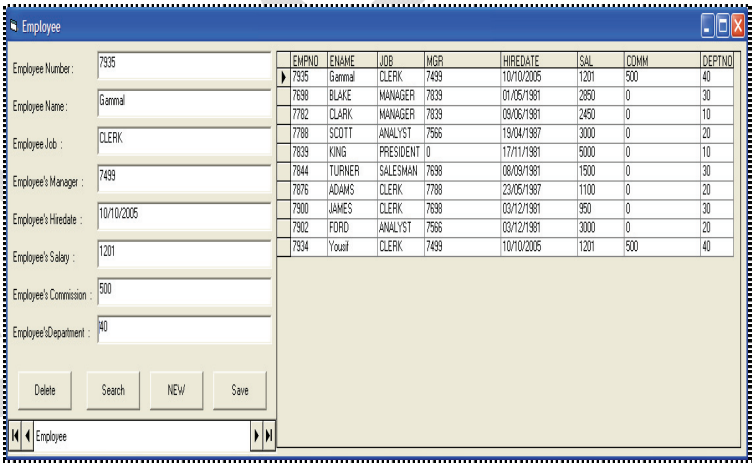


عليك بعد ذلك القيام بربط الأداة (DataGrid) بالأداة (EmployeeADODC)

إذن.. توجه إلى خصائص الأداة (DataGrid) ، ثم قم بضبط الخاصية (DataSource) إلى (EmployeeADODC) ، كما تري في الشكل التالي :

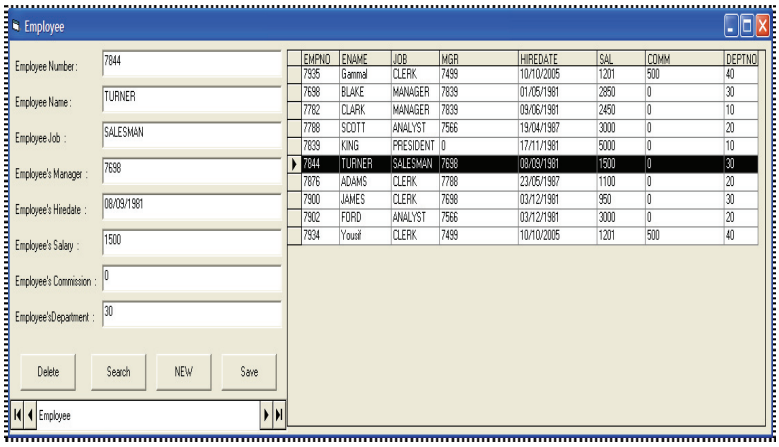


الآن.. قم بتشغيل التطبيق لترى نتيجة عملك ، وستجد أن بيانات الموظفين كافة قد ظهرت في أداة شبكة البيانات :



ملحوظة :

بما أن البيانات والسجلات مرتبطة بعضها بالبعض ، فعند استخدامك الأداة (EmployeeADODC) ، للتحول من سجل إلى آخر ستجد أن أدوات النص وأداة شبكة البيانات تتحرك إلى السجل نفسه ، كما أنه عند الضغط على سجل أي موظف في شبكة البيانات ستجد أنها تم عرضها في أدوات النص :



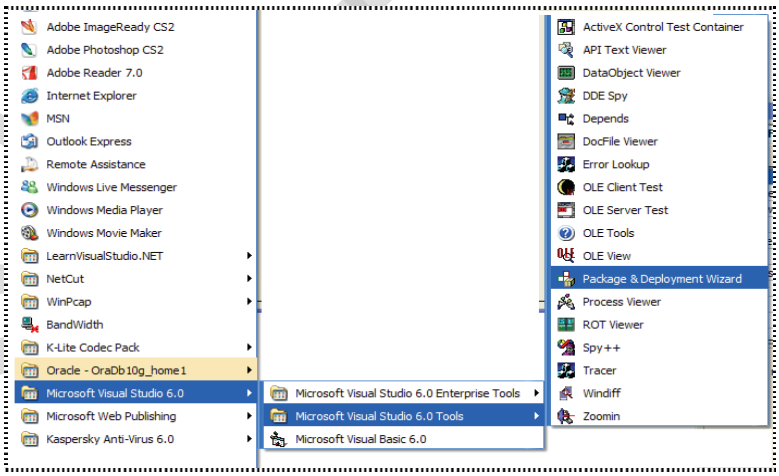
والآن سنتناول كيفية تصميم (Package) خاص بالمشروع الذي نفذناه ، لكي يعمل على أي جهاز آخر .

بناء حزمة للتطبيق Program Package

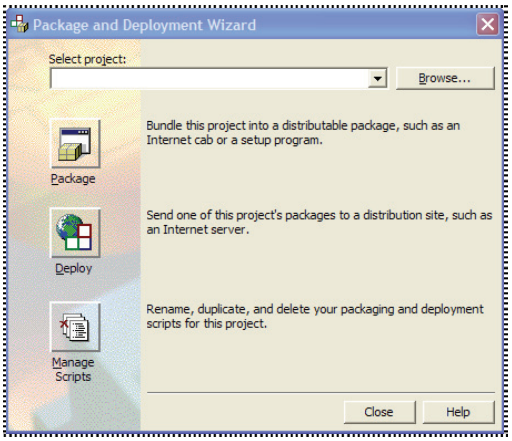
لعل هذا الموضوع خارج عن نطاق الكتاب ، لكننا نرى أنه من الضروري معرفة كيفية إنشاء حزمة لبرنامج سبق أن قمت بتصميمه من خلال الفيجوال بيسيك ، إذ ليس من المنطقي أن تقوم بتصميم تطبيق تعجز عن تشغيله على نظام آخر .

لذا تابع معنا كيفية القيام بهذا .

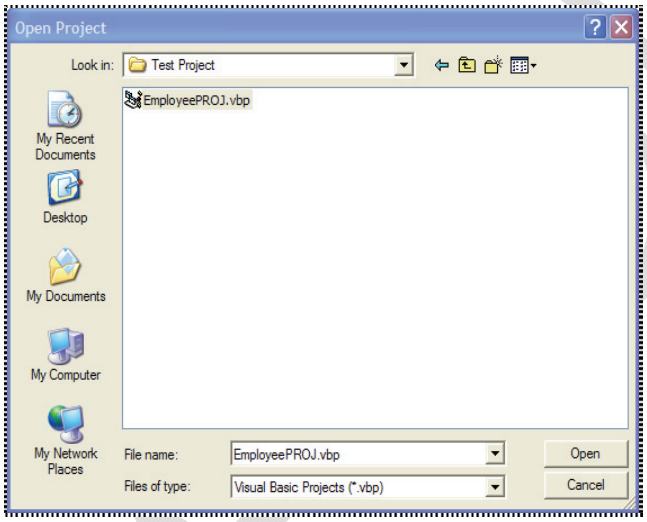
وجه إلى أدوات الفيجوال ستوديو ، ثم قم بتشغيل **(DevelopmentWizard & Package**



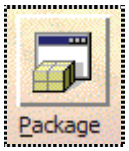
وسوف يظهر لك برنامج مصمم حزم التطبيقات، فقم بالضغط علي **(Brows)** :



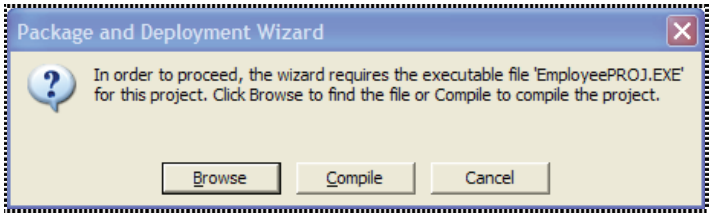
ثم قم بتحديد مكان ملف المشروع الذي سبق لك تصميمه :



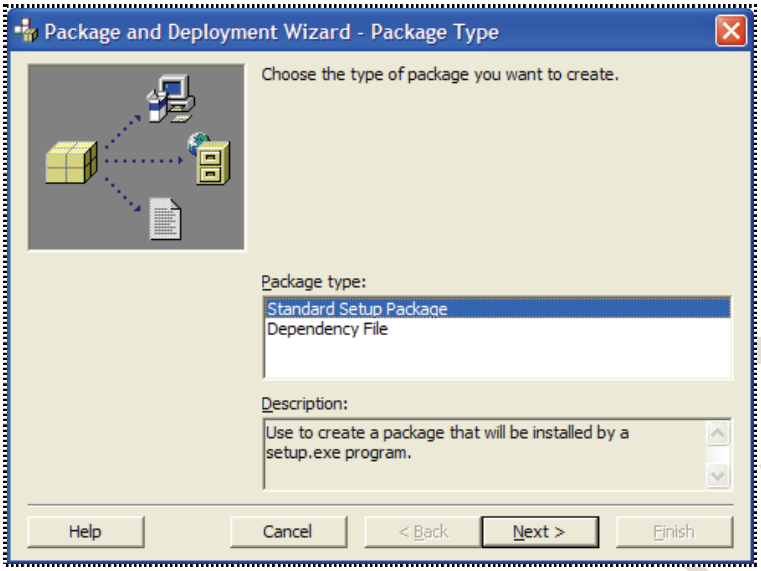
الآن.. اضغط على (Package) .



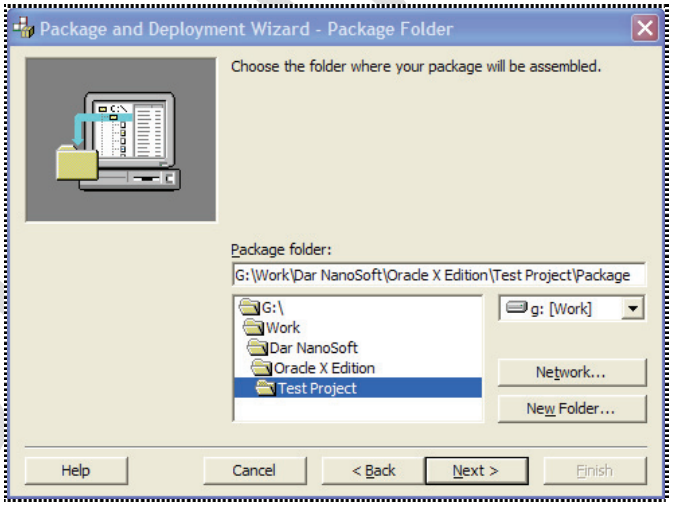
وسوف تظهر لك النافذ التالية :



قم بالضغط على (Compile) ، لكي يتم تجميع البرنامج من جديد ، حتى يتم إدراجه إلى حزمة البرنامج ، وانتظر قليلا حتى تظهر لك النافذة التالية :



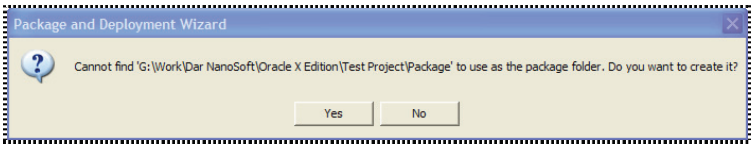
اختر من هذه النافذة (Standard Setup Package) ، ثم اضغط (Next) ، وسيتم تحويلك للخطوة التالية ، التي يتم من خلالها تحديد مكان وضع الحزمة البرمجية :



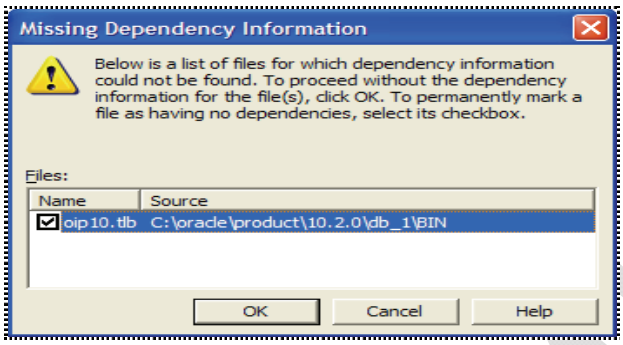
قم باختيار مكان بناء الحزمة التي سوف يتم وضعها تلقائياً في نفس مجلد المشروع

وقد تظهر لك الرسالة التالية ، التي تشير إلى عدم وجود المجلد الذي تريد إدراج الحزمة فيه (في حالة اختيارك المكان الافتراضي) .

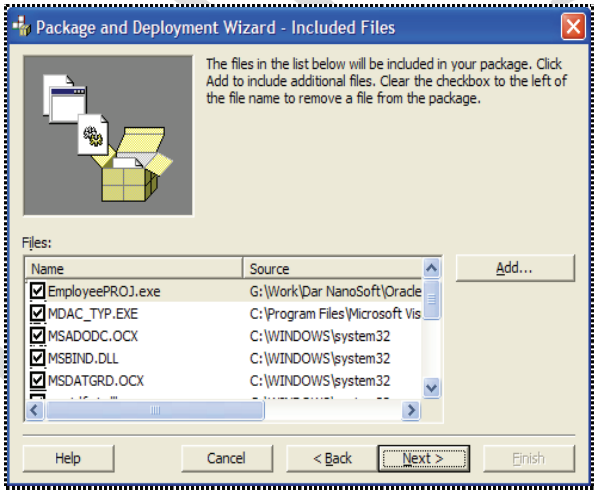
إذن .. قم بالضغط علي (Yes) ، ليقوم البرنامج بإنشاء ذلك المكان :



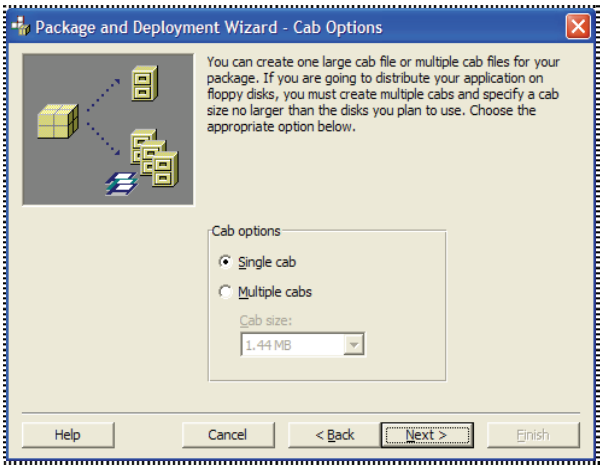
وأثناء قيامك ببناء الحزمة ، ستظهر لك الرسالة التالية :



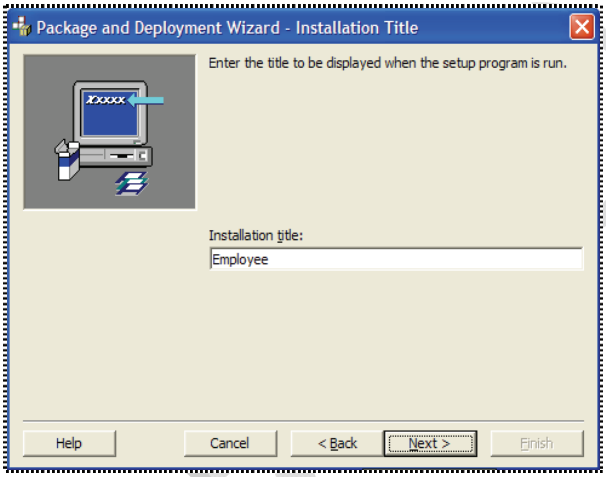
وتوضح تلك النافذة الملفات ، التي قد يحتاجها التطبيق ، مثل ملفات (OCX) و (DLL) . لذا عليك اختيار أي ملف يظهر أمامك ثم اضغط (OK) ، ثم اضغط (Next) :



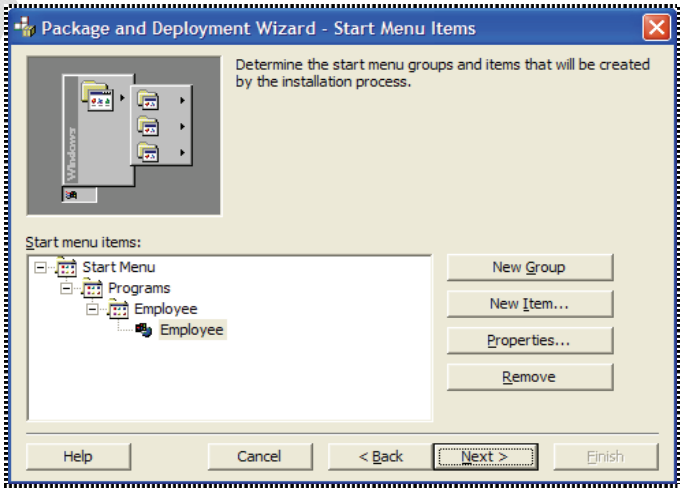
قم الآن باختيار نوع الحزمة ، سواء أكانت من النوع الفردي أو المتعدد . والأفضل أن نختار (Single CAB) ، ثم اضغط (Next) :



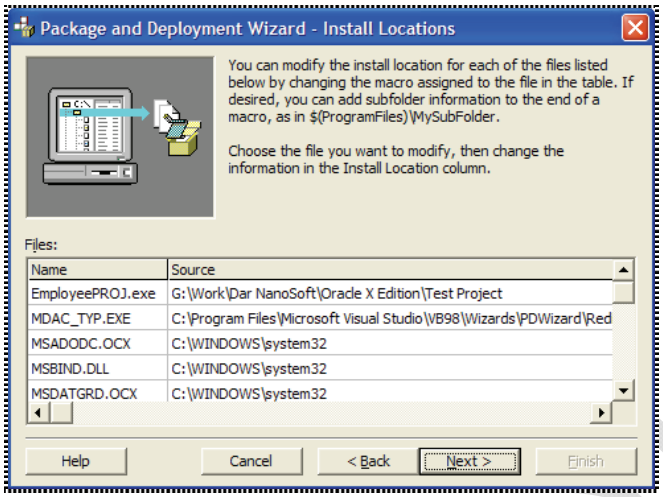
الآن قم بكتابة عنوان البرنامج الخاص بالتركيب في خانة (Installation Title) ، كما في الشكل التالي :



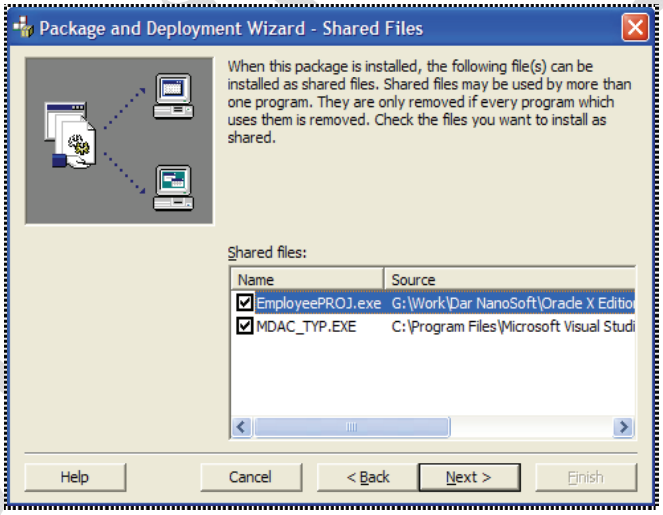
عليك الآن تحديد المكان الذي سيتم تركيب البرنامج فيه . إذن.. قم بتحديد المكان ، أو اتركه كما هو في مكانه الافتراضي ، ثم اضغط (Next) :



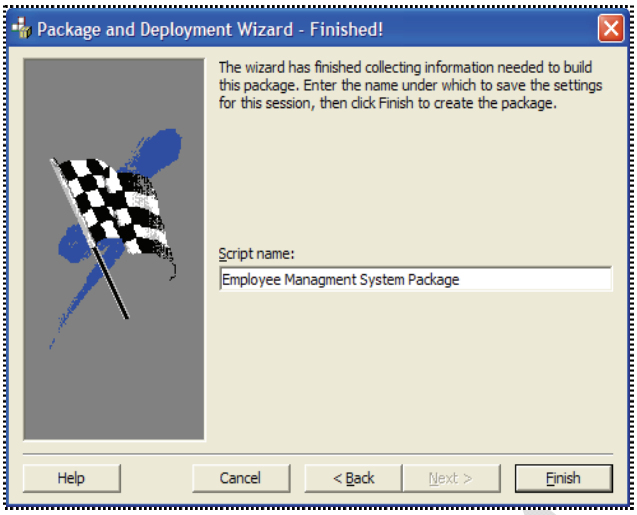
وسوف يؤكد البرنامج مجموعة الملفات التي تم إدراجها من حزمة البرنامج ، اضغط على الزر (Next) للتأكيد :



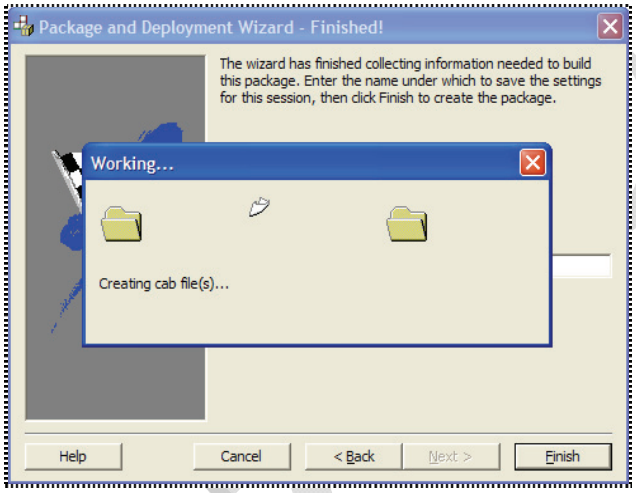
والآن اضغط على (Next) في الخطوة (Shared Files) ، دون القيام بأي تغيير :



ثم قم بكتابة العنوان الذي سيظهر أثناء عملية التركيب ، والآن اضغط (Finish) كما تري في الشكل التالي :

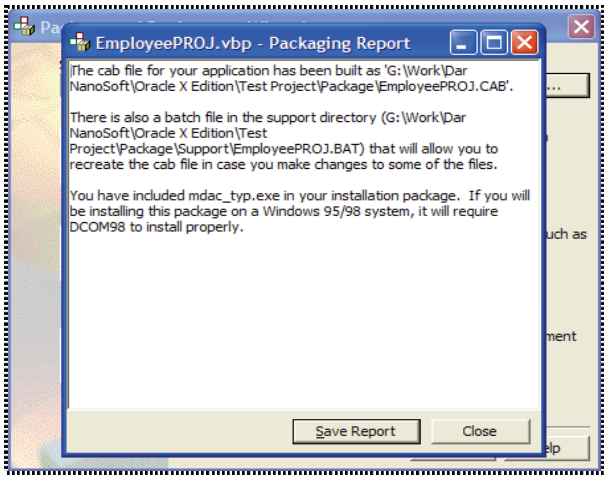


الآن يتم إنشاء حزمة البيانات للبرنامج :

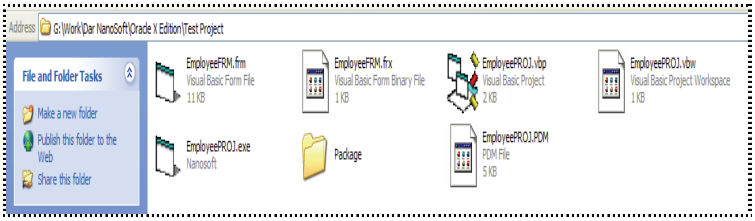


وسوف تظهر الرسالة التالية ، التي تفيد الإنتهاء من العملية بنجاح .

اضغط الآن على **Close** :

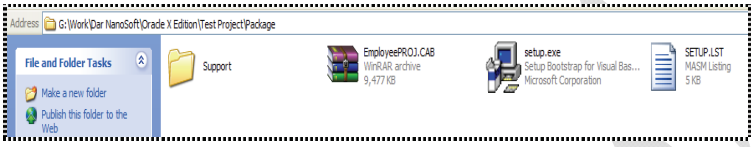


وسوف تجد مجلداً باسم (Package) داخل مجلد المشروع الذي قمت بإنشائه .



الآن..

يمكنك تركيب البرنامج على أي نظام آخر مادام يتصل بقاعدة بيانات (أوراكل) :



هنا.. نكون قد إنتهينا من هذا الجزء الخاص بأساسيات هذا المجال ؟!

وسوف نتوجه الآن إلى الجزء الثاني من الكتاب والخاص ببيان كيفية تصميم برنامج النظام الإلكتروني الخاص بإدارة موظفي الشركة وشركات الأدوية .

المحتويات

مقـــدمـــــة 6

الفصل الأول

فكر التطوير

تعريف المطور:- 8

ما هي رؤية المحلل : 9

تحليل النظم System analysis :- 9

مفهوم أخر لتحليل النظم Systems Analysis :- 10

محلل النظام :- 10

الخبرات التي من الضروري أن تتوفر في محلل النظام :- 11

تحليل النظم 12

مراحل تحليل نظم المعلومات المحاسبية 12

الفصل الثاني

دورة تطوير حياة النظم

دورة تطوير حياة النظم Systems Development Life Cycle : 18

1.تحديد المشكلات Identifying Problems : 18

2.دراسة الجدوى Feasibility Study : 19

3.تحليل النظم Systems Analysis : 20

4.مرحلة تصميم النظم Systems Design : 21

5.مرحلة التطبيق Implementation : 21

♦**استراتيجيه التحول الفوري Immediate Replacement** : 22

♦**التشغيل المتوازي Parallel Operation** : 22

♦الإحلال التدريجي **Phase Replacement** : 22

♦مرحلة التقييم **Evaluation** :..... 23

الفصل الثالث

بناء نظام معلومات

ثالثاً : التخطيط الاستراتيجي لنظم المعلومات :- 28

رابعاً : سياسات وخطط نظم المعلومات :- 31

سابعاً : تحليل النظام :- 34

ثامناً: التصميم العام للنظام وتقويمه :- 38

تاسعاً : التصميم التفصيلي للنظام :- 39

عاشراً : توثيق مستند المواصفات :- 41

الرابع عشر : المتابعة والتقويم :- 45

السادس عشر: المهن المتعلقة بالحاسوب ونظم المعلومات :- 48

الفصل الرابع

التحليل الإحصائي للنظم

مقدمة إلى النظام الإحصائي spss 52

مقدمة 52

أولاً: طرق اختيار العينة من مجتمع 52

عملية إدخال البيانات في SPSS : 53

المرحلة الأولى : كتابة اسم المتغير 58

المرحلة الثانية: تعيين نوع المتغير 58

المرحلة الثالثة : تعيين الأوصاف للمتغير 61

المرحلة الرابعة: تحديد القيم المفقودة :- 62

المرحلة الخامسة : تحديد شكل العمود :- 63

المرحلة السادسة : تحديد مقياس المتغير : 63

الفصل الخامس

التطبيق العملي

مقدمـــة 68
برمجة الحاسب 68
طبقـات البرمجيات 71
قاعدة البيانـات 71
أهمية قاعدة البيانـات 72
قاعدة البيانـات العلائقية 72
جداول قاعدة البيانـات 73
الأعمدة Columns 74
برامج المعالجة 75
التطبيق بلغة برمجة : 81
صندوق الأدوات Toll Box : 82
تصميم النموذج Form : 83
متصفح المشروع Project Explorer : 83
كيفية تجميع المشروع : 85
حفظ المشروع : 87
مشروع موظفي الشركة 88
إضافة المراجع References 90
تصميم النموذج 91
إضافة أداة البيانات : 94
الاتصال مع قاعدة البيانات - ACCESS 95
تشغيل التطبيق RUN 102

بعض المهام المتقدمة 103
مشكلة Null : 106
شبكة البيانات Data Grid : 109
بناء حزمة للتطبيق Program Package 111

لمن له الحق :

المراجع التي رجع إليها المؤلف أثناء كتابته لهذا الكتاب هي كالتالي :

System Analysis Principles DR.Alim
Programming Cycle Unknown Author
Intro to System Analysis Dr.Ghazy
تطوير وبناء نظم المعلومات للكاتب سعودي العتيبي

اسامه محمد فتحي

يمكنك تحميل الأكواد والبرامج المذكورة في هذا الكتاب من خلال موقعنا على الإنترنت

www.daralbraa.com

الموقع الرسمي لدار البراء

WWW.DarAlbraa.com

يقدم الموقع العديد من الخدمات المجانية منها:

📖 خدمة الدعم الفني

إذا واجهتك أي مشكلة فنية فتوجه مباشرة إلى قسم الدعم الفني بموقعنا وأرسل لنا بطبيعة المشكلة وانتظر الرد على بريدك الالكتروني

📖 إصدارات الدار

يمكنك من خلال الموقع التعرف على كافة إصداراتنا من الكتب والبرامج كما يمكنك الاطلاع على بيانات تفصيلية عن كل كتاب ، وكذلك تحميل الفهرس الخاص بالكتاب ، أما لتحميل مرفقات الكتاب من برامج وتمارين وملفات الدعم فعليك البحث عن الكتاب وعرض التفاصيل الخاصة به وستجد روابط يمكنك بواسطتها تحميل الملفات المرفقة

من اصدارات دار البراء

1. تحليل النظم
2. اكسل 2010
3. بوروبوينت 2010
4. وورد 2010
5. Microsoft Accounting
6. كل شئ عن الانترنت
7. الادارة الاستراتيجية
8. تعلم ثلاث لغات في كتاب واحد
9. فوتوشوب CS5
10. أوفيس 2010
11. أوراكل g11
12. فلاش CS5
13. التويفل
14. Special Effects

دار البراء