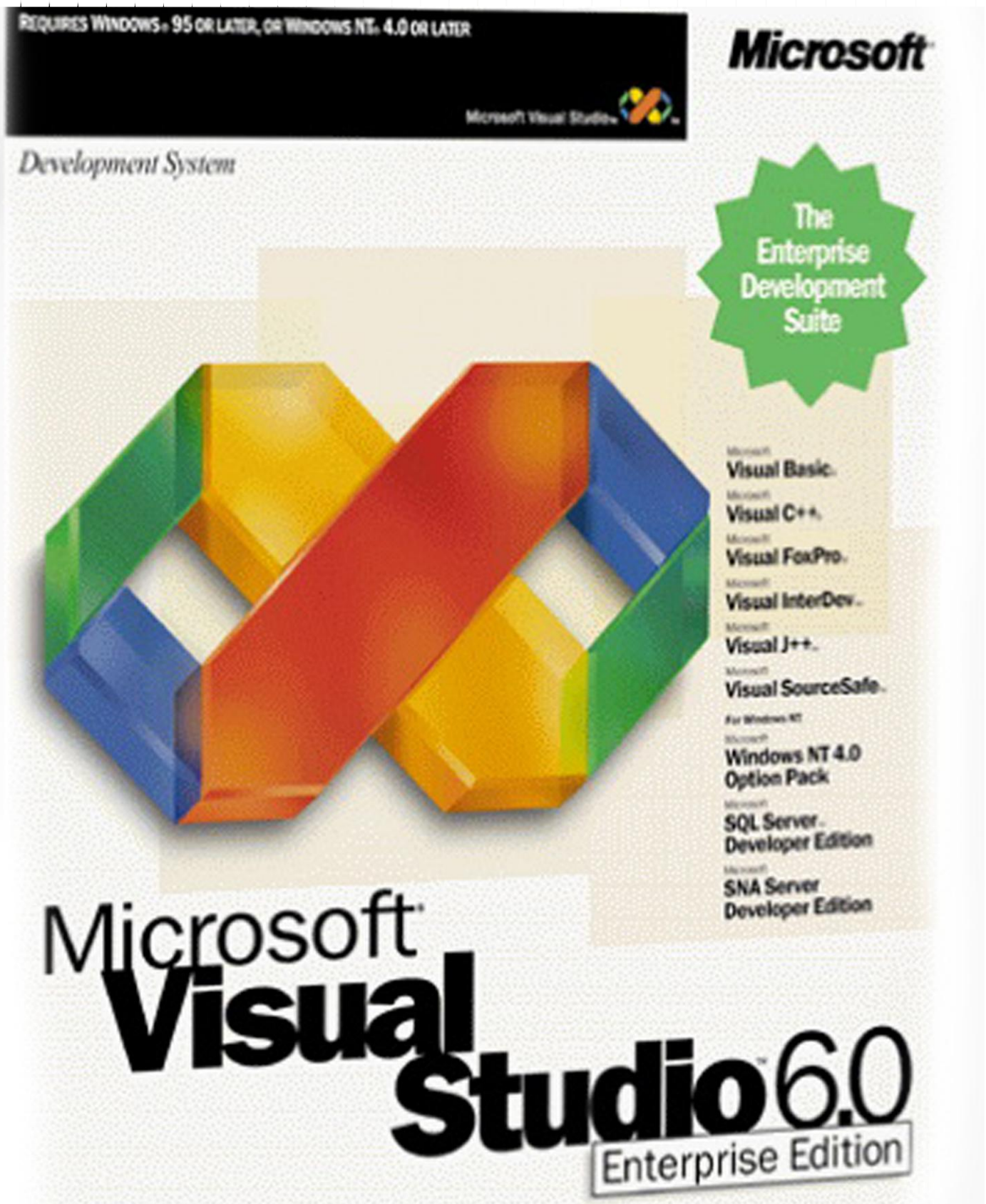


تعلم البرنامج مع الصور التوضيحية

بابسط الطرق

اعداد المهندس
محمد محمود

هندسة الالكترونيك
و السيطرة

الكلية التقنية
كركوك

مقدمة في فيجوال بيسك

لقد قامت شركة مايكروسوفت بتصميم فيجوال بيسك بناءً على لغة البرمجة بيسك . هذه اللغة (بيسك) مصممة خصيصاً للمبتدئين في البرمجة نظراً لسهولتها على عكس اللغات التي كانت متوفرة حينذاك مثل الكوبول و فورتران و لغة التجميع Assembly

رغم أن لغة بيسك مصممة للمبتدئين إلا أن برامجها تحتاج لبعض الدراسة لفهمها و هي تعتمد على النصوص بشكل تام لذا فإن البرامج التي تكتب بلغة بيسك تنتج ناتجاً نصياً أيضاً و لا تظهر في نوافذ كما هو حال برامج الويندوز الحالية .

لغة فيجول بيسك 6 غنية عن التعريف، هي لغة برمجة وبيئة تطوير ومعالج، تعتبر من اللغات الشيئية الكائنتية المنحنى

تعتبر لغة فيجول بيسك 6 اخر اصدارات البيسك المرئي استكمالا للفيجول بيسك 1.... وصولاً للفيجول بيسك 5

وبعدها بدءت لغة اصدارات الفيجول ستوديو دوت نت في سنة 2003 وصولاً الى الاصدار الاخير سنة 2010

على الرغم من قدم لغة vb 6 سنة 1998 الا انها اللغة الاكثر شعبية في العالم لسهولته الشديدة مقارنة من السي بلس

وغيرها من اللغات ،، الجميل في الفيجول بيسك شاشتها مرئية وبالامكان التعديل على البرنامج بسهولة على عكس الاسمبلي

شاشتة تشبه نظام الدوس

1 - أصبح بإمكان المبرمج أن يصمم برامج ذات واجهات جميلة حيث توفر لغة فيجوال بيسك للمبرمج (Objects) وأدوات تستخدم فى نظام التشغيل نفسه كصناديق النصوص والصور إلخ ...

2 - دعم شركة ميكروسوفت لهذة اللغة حيث أضافتها ضمن مجموعة برامج أوفيس من أجل تطوير قواعد البيانات التى ينشأهل برنامج على سبيل المثال MS-Access .

3 - تم تطوير عدد كبير من (OCX ACTIVEX) من خلال العديد من الشركات والمحترفين وطرحت فى الأسواق لتستخدم فيما بعد فى دعم برامج فيجوال بيسك.

4 - دعم جديد من ميكروسوفت للغة فيجوال بيسك فى الإصدار الخامس حيث أضافت فى الإصدار الخامس من فيجوال بيسيك إمكانية بناء عناصر التحكم (OCX ACTIVEX) وبذلك أصبحت لغة فيجوال بيسيك تضاهى بقية اللغات القوية فى إمكانياتها .

5 - أضافت ميكروسوفت إمكانية إستدعاء الكثير من الوظائف الموجودة فى نظام التشغيل وإمكانية الإستفادة من بعض البرامج التى تعمل فى بيئة ويندوز وتسخيرها لصالح البرنامج المنشأ بلغة الفيجوال بيسيك .

6 - مع كل هذا التطور بقيت لغة فيجوال بيسك حتى الإصدار الخامس لها ضعيفة فى معالجة قواعد البيانات وفى إنشاء تطبيقات تدعم الإنترنت مما دفع شركات أخرى كشركة الى طرح عدة إصدارات من لغتها البرمجية (Borland Delphi)دلفى والتى تتمتع بمحرك قواعد بيانات قوى ودخلت ميكروسوفت منافسة مع شركة بورلاند جعلت هذة المنافسة شركة

ميكروسوفت تطور فيجوال بيسك لتفوز فى هذة المنافسة وأصدرت ميكروسوفت الإصدار السادس الذى يتميز عن سابقه بدعمه القوى لقواعد البيانات ولا زالت المنافسة مستمرة فظهرت إصدارات أخرى من دلفى وإصدار جديد من الفيجوال بيسيك . Visual Basic.net

وهذة المنافسة فى صالح المبرمجين الآن فيجب أن يستفيدو منها حيث تطرح هذة الشركات أحدث ما توصلو إليه من علوم

البرمجة وأحياناً بلا مقابل لتدعم كفائتهافى تلك المنافسة.

شرح واجهة الفيجوال بيسك

هذه الواجهه سوف تظهر عند بدء تشغيل الفيجوال بيسك وفي الأسفل شرح لها.
يمكنك إزالتها من بدء التشغيل عن طريق تفعيل علامة الإختيار **Dont't Show this dialog in the future**

علامات التبيويب في أعلى الواجهة:

New : وتعني إختيار مشروع جديد
Existing : فتح ملفات مخزنة لديك .
Recent : قائمة بآخر الملفات التي قمت بفتحها حسب التاريخ

New : يضم مايلي:-

Standard EXE: وهو يقوم بإنشاء برنامج تنفيذي بعد طلبك لذلك من (Make EXE) في قائمة (File) .

Activex EXE: وهو أيضاً يقوم بإنشاء ملفاً له الامتداد (DLL) وهو ملف ذو برامج فرعيه مساعدة وهو لا يعمل بنفسه بل معexe.

Activex DLL: وهو ينشئ ملفات لها امتداد EXE تعمل مع برامج إخرى لكي يعمل في شكل (OLE) .
Activex control: وهو يساعد على إنشاء ملف من نوع (ocx) و هي مهمة وتساعد على إحتواء واجهة مستخدم أو برامج فرعية.
Activex Document DLL: وهو يساعد على إنشاء ملف ذو امتداد (DLL) يقوم بتشغيل برامج على موقع في الانترنت.
Activex Document exe: يقوم بإنشاء ملفات تظهر على الانترنت .

Add In: وهو خاص بالفيجوال بيسك حيث يمكنك إضافة واجهة مع ما ذكر سابقاً خاصة بك .

VB Application Wizard:وهو معالج تلقائي في هذه اللغة يساعدك على إنشاء نوافذ عدة دون كتابة سطر واحد من النوع **exe**

ISS Aplication : نقصد بـــ (**ISS**) إختصار لكلمة (**Internet Information Server**) و يقوم بالتحضير لتصميم (**web class**) وهو ما يسميه المستخدم صفحات(**HTML**) .

Data project : تضع مشروعاً لقواعد بيانات وتهيئة في بيئة متكاملة من أدوات الربط وصنع التقارير دون الحاجة لفتح برامج أخرى.

Vb wizard manager : و تبدو مشابهة لمعالجات (**Microsoft**) يمكنك إستخدامه .

DHTML Application : تسهل عليك صنع صفحات (**HTML**) ديناميكية للإنترنت دون تعلم لغة **Java** .

Vb Enterprise Edition controls : وهي تساعد على وضع جميع الأدوات المستخدمة بكثرة لقواعد البيانات و الاتصال ومن ثم تطبيقها وهي للمحترفين.

يحتوي الفيجوال بيسيك علي 13 قائمة ، و هي كالتالي :

File : تحتوي علي اوامر اساسية، مثل فتح و حفظ مشروح، و فتح مشروع جديد، و تحويل البرنامج الي امتداد**Exe**.

Edit : تحتوي علي اوامر التحرير العادية بالاضافة الي اوامر اخري.

View: تعمل محتويات هذه القائمة علي اظهار بعض الاشياء مثل صفحة كود و فورم و صندوق الادوات...الخ .

Project: تحتوي هذه القائمة علي اوامر خاصة بمحتويات المشروع مثل اضافة **Module** او **ClassModule** او غيرها.

Format: بهذه القائمة يمكنك تنسيق برنامجك، مثلا وضع زر في منتصف الشاشة تماما او محاذاته الي اليمين او اليسار او غيره، و بها ايضا الامر **LockControls** الذي يمكنك من عدم السماح بتحريك او تغير حجم كائن، و يمكنك ازالته بالضغط عليه مرة اخري.

Debug : تحتوي هذه القائمة علي معظم اوامر التشغيل، و منها اختيار طريقة تنفيذ البرنامج، مثل تنفيذ سطر واحد منه **StepInto** ، او تنفيذ كله **StepOver** ، او الامر السابق **StepOut** ، او التنفيذ حتي السطر الذي يوجد عليه مؤشر الكتابة **RunToCursor** ، و **BreakPoints** التي تظهر باللون الاحمر عند;كتابه الكود، فعند اختيارها يتم توقف مؤقت للبرنامج اثناء تشغيله.

Run : من خلال هذه القائمة يمكنك تشغيل او ايقاف البرنامج، و اعادة تشغيله.

Query : تحتاج هذه القائمة الي جمل استعلام SQL.

Diagram : تحتاج هذه القائمة الي SQL او ORACLE.

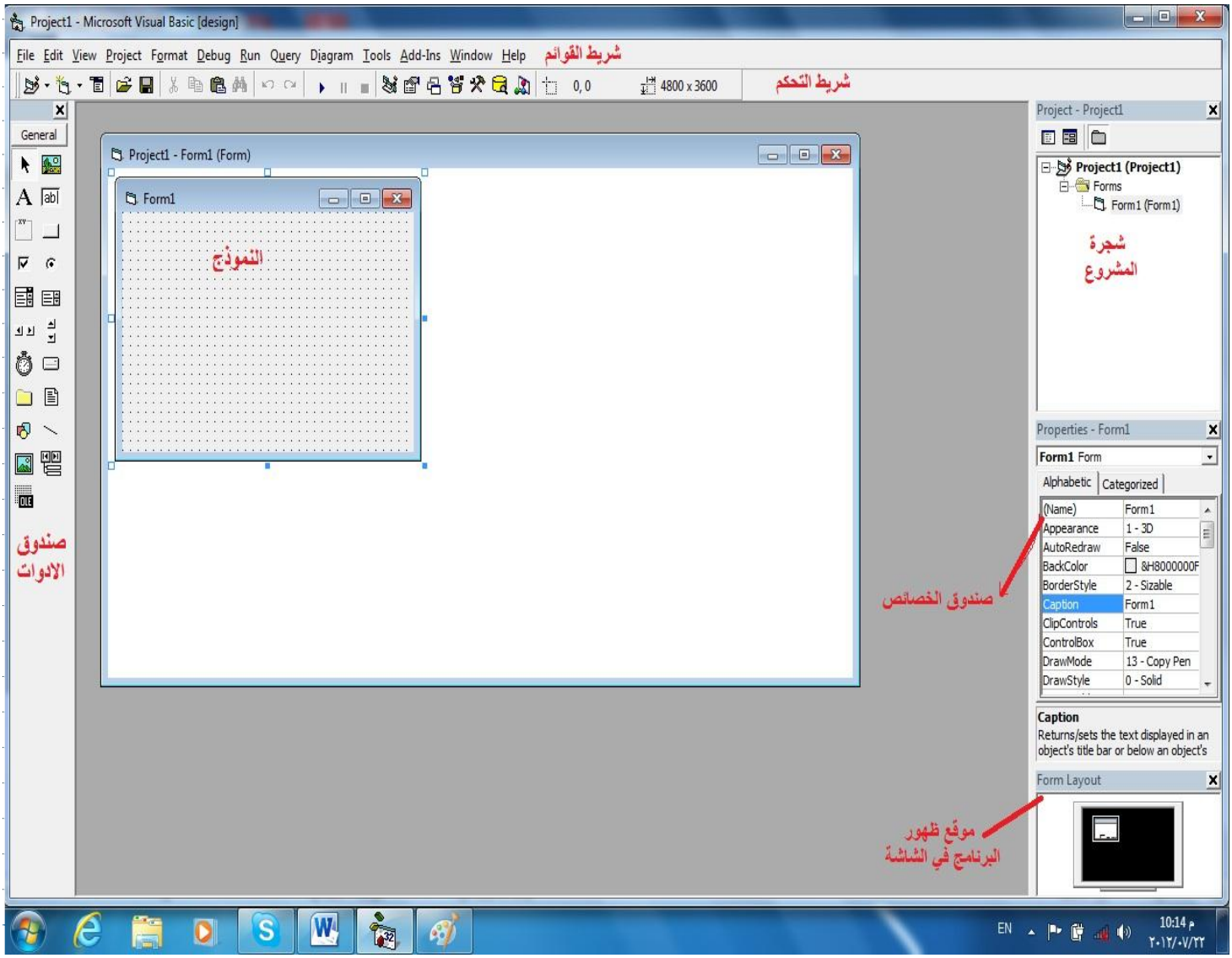
Tools : تحتوي هذه القائمة علي اوامر كثيرة مثل AddProcedur الذي يمكنك من كتابة اجراء جديد و لكن يجب ان تكون في صفحة الكود عند اختيار هذا الامر، اما ProcedureAtributes فهو يمكنك تخصيص بعض الاشياء للاجراء الذي انشأته مثل تغيير ID و غيرها.

Add-Insert : تحتوي هذه القائمة علي برامج مستقلة توفر خدمات للبرنامج .

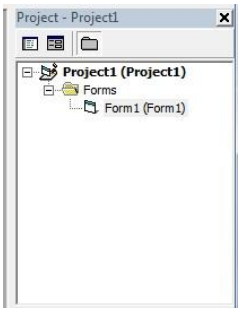
Window : تنظم هذه القائمة صفحة الكود و الفورم ليكونا ظاهرين مع بعضهما.

Help : تحتوي هذه القائمة علي تعليمات خاصة ولا يمكنك فتحها الا اذا تم تحميل مكتبة MSDN و بها ايضا الصفحة الخاصة بنبذة عن للفجوال.

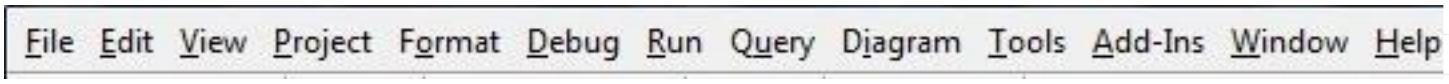
واجهة البرنامج



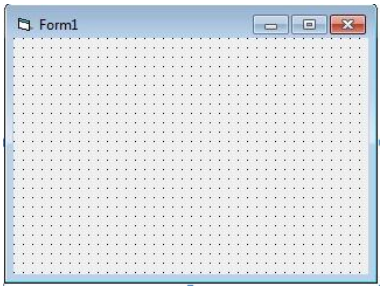
1 - شجرة المشروع (TREE PROJECT) : ويقوم بعرض جميع المشاريع و النماذج للمشروع الحالي.



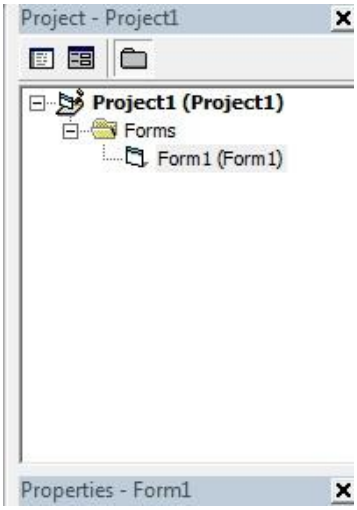
2 - شريط القوائم (Menu bar): وبه قائمة الملف و التحرير و المشروع و التنسيق و متابع الاخطاء و التشغيل و كذلك الأدوات و الاضافة و النافذة وقائمة المساعدة.



3 - نافذة البرنامج (Form Window) : و هي النافذة النشطة و التي تكون عاملاً عليها مثل(Form1).



4 - نافذة الخصائص (Properties Window): وتظهر بها خصائص النموذج المفتوح حالياً وبها تنسيق لون الخلف والامام و الحجم و الموقع و الرؤية و التمكين وغيرها حسب كل أداة.



5 - نافذة الأدوات (TOOLS Window) : وهي تكون على يسار الواجهة وهي قد تكون نص أو عنوان أو صورة أو مؤقت وغيرها كثير جداً ما عليك إلا الضغط على الأداة و من ثم رسمها على نافذة البرنامج .

6 - شريط الأدوات (Toolbar) : وتعرض بها الأوامر التي تستخدم في الفيجوال بيسك بكثرة مثل القص و اللصق و التشغيل و غير ذلك

.

كونات لغة البرمجة فيجوال بيسك

1 - الوظائف. (Functions)
2 - الطرق أو الوسائل(Methods)
3 - الأحداث (Events)
4 - مفاتيح المهام(Keywords by Task)
5 - الخصائص(Properties)
6 - التعبيرات(Statements)
7 - الكائنات(Objects)

الدوال الجاهزة(Bulit-Function):

في ما يلي جدول بالدوال الخاصة بلغة فيجوال بيسك حيث تكون بشكل برامج فرعية جاهزة

Function	Value Returend	Example
Abs	Absolute Value	Abs (-3) =3
Asc	ASCII code of chaeacter	Asc(“A”) = 65
Chr	Character of ASCII code	Che(65) =A
Cos	Cosine of angle	Cos(3.14159)
Date	Current Date as string	Date = 19/11/2007
Format	Date of Number converted to a text string	
Left	Selected left end of a text string	Left (“abcd”,2,3)= bc
Len	Number of character in a string	Len(“Ahmed”)=5
Log	Logarithm of value	Log(1)=0
Mid	Selected portion of a text string	
Now	Current time and date	Now =19/12/2007 10:25:28م
Right	Select right end of a text string	Right(“abcde”,3,2)= bc
Round	Approximate to nearest integer	Round (3.45) =3
Rnd	Random number in range (0-1)	Rnd = 0.7055475
Sin	Sine of an angle	Sin(0)=0
Sqr	Square root of a number	Sqr(9) =3
Str	Number converted to text string	Str (99) =99

Tan	Tan of an angle	Tan(0)=0
Time	Current time and date	Time = 10:25:43م
Val	Numeric value of a gavin string	Val(99) =99

بنية الفيجوال بيسك

تتكون لغة الفيجوال بيسك من مجموعة من التعليمات مكتوبة في أسطر وتعليمة الفيجوال بيسك تتكون من :

1- الكلمات المحجوزة للفيجوال للبيسك والخاصة بتعليمة معينة : مثلاً (End و If و For)

2- المحارف والمتحولات والثوابت :

أولا_ المحارف

تستخدم لغة البيسك ثلاثة أنواع من المحارف هي:

A – الأحرف الأنكليزية الصغيرة والكبيرة

B – الأرقام من 0 الى 9

C – محارف خاصة متعددة الأنواع :

المعامل	اسمه	وظيفته
^	الأس	رفع عدد الي اس معين
*	الضرب	وهي لضرب عددين
/	القسمه	وهي لقسمه عددين
MOD	باقي القسمه الصحيح	وهي باقي قسمه عددين
+	الجمع	وهي لجمع عددين او اكثر
-	الطرح	وهي لطرح عددين

المعامل	اسمه	معناه
<	اصغر من	البيان الاول اصغر من البيان الثاني
>	اكبر من	البيان الاول اكبر من البيان الثاني
=	يساوي	البيان الاول يساوي البيان الثاني
<>	لا يساوي	البيان الاول لا يساوي البيان الثاني
<=	اصغر من او يساوي	البيان الاول اصغر من او يساوي البيان الثاني
>=	اكبر من او يساوي	البيان الاول اكبر من او يساوي البيان الثاني

المعامل	وظيفته
&	دمج اي سلسلتين (حرفيتين او حرفيه وعدديه) والناتج يكون سلسله حرفيه جديدة
+	دمج سلسلتين حرفيتين والناتج يكون سلسله حرفية جديدة

ان التعابير الحسابية (التعابير الرياضية) في البرنامج تكتب كتعابير برمجية كما موضح في الامثلة ادناه:

تعابير برمجية	تعابير رياضية
3*(x+4*y)	3(x+4y)
X^2+4/2	$X^2+4\div 2$
16^(1/4)+3^3+10-(5*4)/3^2-2^3	$\sqrt[4]{16}+3^3+10-5\times 4\div 3^2-2^3$
(5*y)/(x^2-4)+x-1	$\frac{5y}{x^2-4}+x-1$

علامات الترقيم : وهي الفاصلة والفاصلة المنقوطة والنقطة والنقطتان فوق بعضهما وإشارة الإسفهام وإشارة التعجب والفراغ والأقواس وعلامات التنصيص
ورموز أخرى مثل $ وإشارة النسبة % وإشارة العطف & ورمز الإعداد # والشخطة _ و @ والخط المائل

العمليات المنطقية

عملية NOT : اذا كانت هذه العملية false فتكون نتيجة الشرط true .
عملية AND : يجب ان تكون الشرطان صحيحان لتكون النتيجة true .
عملية OR : تكون true اذا كانت احد الشرطين صحيح .
عملية XOR : تكون النتيجة true اذا كان واحد فقط من التعبيرين الشرطين صحيحا وتكون النتيجة خطأ اذا كان كلاهما صح او خطأ .
عملية EQV : تكون النتيجة true اذا كان كلا التعبيرين الشرطين صحيحا او خطا .

ثانياً – الثوابت Constants

الثابت قيمة تستخدم في البرنامج دون أن يطرأ عليه أي تغيير خلال تنفيذه وهي على نوعين :
1 – ثابت عددي وهذه القيمة تقسم إلى نوعين :
A – ثابت عددي صحيح INTEGER
B – ثابت عددي حقيقي REAL وله شكلان :

a - شكل عشري : مثل 1.15
b – شكل أسي : مثل 3 ^3

2 – ثابت حرفي STRING :

وهو تابع لمحارف مكتوبة بين علامتي التنصيص " " لا يستخدم الثابت المحرفي في العمليات الحسابية وإنما يعامله الكمبيوتر كنص
مثلاً لو أدخلنا التعليمة Print 2*3 فإنه سوف يظهر ناتج ضرب العددين أي 5
أما لوأدخلنا التعليمة Print "2*3" فإنه سوف يظهر الرقمان كما هما (2*3)

* كلمة مفتاحية لاعلان عن الثوابت Const
* القيم الثابتة المخصصة للثابت الاول Value-1
* اسم الثابت الاول Constname1

3 – المتغيرات VARIABLES

موقع محدد في الذاكرة يمكن ان يحتوي على بيانات من اي نوع فالمتغير لايعطي قيمة ثابتة بل يمكن تغييرها بعكس الثابت

الشروط اللازمة لكتابة اسماء المتغيرات في البرنامج :

1- عدم استخدام الكلمات المحجوزة في البرنامج

(if dim while do)

2 - عدم البدء برقم في اسم المتغير مثل لذلك :-

```
dim 2m as string
```

3 - يمنع استخدام الرموز في اسماء المتغيرات مثل

```
dim $ as string
```

4 - يمنع استخدام النقطة او الفاصلة في اسم المتغير مثل :-

```
dim a . as string
dim a , as string
```

تنقسم المتغيرات إلى ثلاث اقسام رئيسية وهى كالتالى :-

1 - متغيرات عددية .
2 - متغيرات نصية .
3 - متغيرات منطقية.
4 - متغير عام ويحمل كل هذه القيم .

انواع المتغيرات

الطريقة الاولى لكتابة نوع المتغير هو بوضع علامة كما موضح في في الجدول:-

نوع النتغير	الرمز
Integer	%
Long	&
Single	!
Double	#
Currency	@
String	$

```
Dim Input1 $
Static Av %
Private Total !
```

المتغيرات والمساحات التخزينية التي تأخذها من الذاكرة توضحها الجدول التالي :-

أنواع البيانات التي يمكن أن تخزن أرقام عشرية :

نوع البيان	الحجم الذى يشغله فى الذاكرة بالبايت
single	4 Byte
double	8 Byte
decimal	16 Byte

أنواع البيانات التي يمكن أن تخزن أرقام صحيحة :

نوع البيان	الحجم الذى يشغله فى الذاكرة بالبايت
Byte	1 Byte
Short	2 Byte
Integer	4 Byte
Long	8 Byte
decimal	16 Byte

أنواع البيانات الحرفية:

نوع البيان	الحجم الذى يشغله فى الذاكرة بالبايت
char	2 Byte
string	10+(2*عدد الحروف)

أنواع البيانات الأخرى:

نوع البيان	الحجم الذى يشغله فى الذاكرة بالبايت
date	8 Byte
boolean	2 Byte

الطريقة الثانية كما مبين من الجدول

النوع	الوصف	طريقة التعريف	مثال على المتغير و قيمته
Integer	عدد صحيح اعتيادي يتراوح من 32768-الى 32767	Dim x as integer	X= 15
Long	عدد صحيح طويل اكبر من 32767او اصغر من -32768	Dim b as long	B = 335674855
single	عدد عشري ذو دقة اعتيادية	Dim y as single	Y= 10.50
Double	عدد عشري ذو دقة مضاعفة	Dim AA as double	AA= 10.1234567
String	متغير حرفي (للحروف فقط)	Dim name as string	Name1 = "ali"
Date	تاريخ	Dim DT as Date	DT = #8-1-1982#
Boolean	متغير منطقي	Dim Y as boolean	Y = true
Variant	كل انواع المعطيات المتغيرة	Dim V as variant	V = 82 V = "college"
Currency	قيم مبالغ الدولارات والإشارة $	Dim C as curtency	O = 55.4
Byte	نوع البيانات الرقمية الموجبة الصحيحة فقط	Dim DG as byte	byte my Byte = 255

كيفية الاعلان عن المتغير

Public : كلمة مفتاحية تستخدم لتعريف أي متغير بحيث يمكننا ان نستخدمه ونستخدم قيمته الاخيرة المخزنة في الذاكرة في أي جزء من المشروع .

Dim : كلمة مفتاحية تستخدم لتعريف أي متغير بحيث يمكننا ان نستخدمه بصورة محلية **Local** .

Static : كلمة مفتاحية تستخدم للاعلان عن الاجراء ككل لتصبح جميع المتغيرات من النوع الساكن **Static** .

Private : كلمة مفتاحية تستخدم للاعلان عن المتغير وتعني ان المتغير ذا صيغة خاصة با لاجراء آو النموذج الذي تم خلاله الاعلان عنه ولا يمكن استخدامه في مكان اخر غير هذا الاجراء آو النموذج .

طرق تعريف المتغيرات في البرنامج باستخدام الكلمة المحجوزة Dim

1 - الطريقة الافتراضية لتعريف المتغيرات :

```
Dim name as string
```

2 - طريقة تعريف متغير منفصل مع نسب القيمة له في نفس السطر :

```
Dim name as string = "ali "
```

3 - طريقة تعريف اكثر من متغير في سطر واحد ويكونوا لنفس النوع (اي كلاهما من نوع نصي او كلاهما من نوع رقمي)

مثال لذلك :

```
Dim f _ name as string , 1_ name as string
```

في هذا المتغير عرفنا متغيريين من نوع نص اسم المتغير الاول F_name والثاني L_name

4 - تعريف متغيرات مختلفة في نفس السطر (اي مختلفة في النوع اي متغير نصي ومتغير رقمي ومتغير منطقي) مثل:

```
Dim name as string , age as integer  , elec as boolean
```

عرفنا 3 متغيرات من انواع مختلفة منها النصي ومنها الرقمي ومنها المنطقي..

لاحظ هنا اننا كتبنا المتغيرات جميعها بكلمة واحد فقط للتعريف وهي Dim

5 - تعريف متغيرات من غير نوع في سطر واحد ونسب القيم لها مثل:

```
Dim name as string = " hassan " , age as integer = 12 , elec as boolean = true
```

6 - تعريف متغير من دون ذكر الكلمة Dim مثل :-

```
Name = " ali "
```

7- باستخدام الكلمة المحجوزة static :

```
Static  name as string
```

سيبقى البرنامج محتفظاً بها داخل البرنامج من لحظة تشغيله إلى لحظة ايقافه ، المعلومات المستخدمة لاضافة المتغيرات باستخدام الامر Dim لاتختلف عن طرق اضافة المتغيرات في الاستاتيك static

مع التنبيه ان الكلمة الاولى تكون اخف على الذاكرة من الكلمة الثانية لان الاولى تحتفظ بالقيمة داخل اجراء منفصل بها والثانية تحتفظ بها طيلة وقت البرنامج.

الثوابت

تنقسم الثوابت في لغة الفيجوال بسيك إلى ثلاثة اقسام رئيسية وهي(نصية او رقمية ومنطقية)
ويمكننا تعريف ثابت في البرنامج من خلال الكلمة المحجوزة :

```
Cons  name  as  type = value
```

ولاحظ هنا عند كتابة الثابت لا يمكنك نسب قيمة له إلا في نفس السطر الذي قمت بكتابة الثابت فيه .

الطرق التى من خلالها تستطيع تعريف ثوابت لبرنامج

1 - الطريقة التقليدية تعريف ثابت ونسب قيمة له مثل :

```
Cons name as string = "ali "
```

2 - كتابة الثابت بدون تحديد نوع له مثل :-

```
Cons name = "ali "
```

3 - كتابة اكثر من ثابت مع تحديد القيمة لها في سطر واحد مثل :-

```
Cons name as type = value

Cons name as string = "ali " , age as integer = 22
```

4 - كتابة اكثر من ثابت بودن تحديد نوع الثابت مثل :

```
Cons name = "ali " , age = 22
```

فوائد الثوابت في البرنامج

1 - تسهيل البرمجة عندما تريد التعويض بقيمة ثابتة لعدد غير معين من المتغيرات.
2 - انها تتعامل مع المصفوفات بكل يسر وسهولة ويمكنها الوصول إلى مصفوفة معينة .

الاخطاء المتوقعة حدوثها عند تعاملك مع الثوابت

1 - تغيير قيمة الثابت في البرنامج او اثناء التشغيل مثال لذلك :-

```
Cons age  as integer = 20
Age = 25
```

2 - تعريف الثابت في سطر ونسب القيمة له في سطر اخر كما قلنا سابقاً .

```
Cons age  as integer
age = 15
```

ملاحظة هامة جداً جداً :- عند تعاملنا مع الثوابت يجب علينا ادخال القيمة للثابت في نفس جملة تعريف الثابت اي في نفس السطر.

طرق الادخال والإخراج في لغة الفيجوال بيسك

1- الادخال INPUT :

نستخدم نوعين من طرق ادخال البيانات

A- Textbox
B- Inputbox

2- الاخراج output:

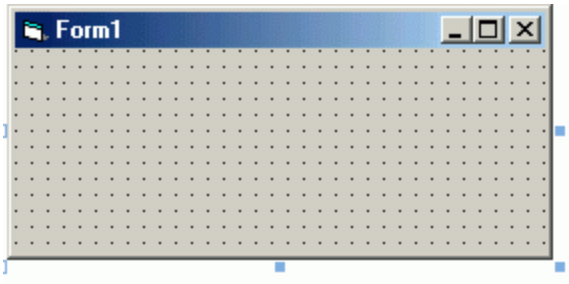
يوجد عدة طرق لاخراج البيانات

A- Textbox
B- Label
C- message box
D- print

شكل الفورم والتعامل معه

النموذج (form1) و هو عبارة عن نافذة أو هو عبارة عن نموذج فارغ و هو الذي نكوّن و نضع عليه الرسوم و الصور و الأزرار و عناصر الإدخال و الإخراج و غير ذلك و الذي سيكون واجهة برنامجك الذي ستصنعه.

وهو حقيقة يشبه لوحة رسم لأنك تستطيع الرسم عليه بسهولة كما يمكنك تغيير حجمه و لونه و أطرافه ، كما تستطيع تغيير عنوانه ايضا .



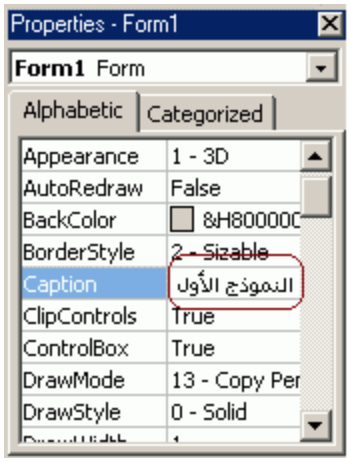
شكل الفورم

ملاحظة : الفورم السابق هو نتيجة إختيارنا في البداية لـــ standar.exe .

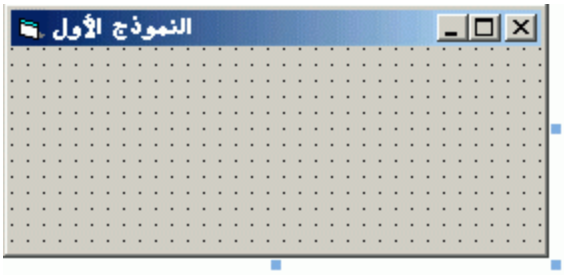
هذا يعني أن هذا الفورم هو الفورم الإفتراضي.

* كيف نغير عنوان الفورم من كلمة form1 إلى العنوان الذي نريده ؟

انقر فوق الفورم نقرة واحدة (لتحدده أو تختاره) ، ثم اذهب لمربع الخصائص أو نافذة الخصائص properties (window) موقعه على يمين الشاشة إذا كان غير ظاهر على واجهة البرنامج إذهب إلى شريط الأدوات و انقر على الزر عندها تظهر نافذة الخصائص اختر caption و اكتب : " النموذج الأول " .



نافذة الخصائص

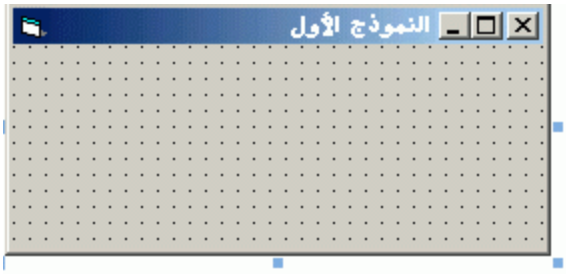


تغير عنوان الفورم انظر فوق

ملاحظة : نافذة الخصائص تخصص جميع العناصر الموجودة على الفورم كالأزرار و عناصر الادخال والإخراج إلخ.....

كيفية وضع العنوان على اليمين

في نافذة الخصائص السابقة ابحث عن **rigth to left** يظهر سهم منسدل للأسفل انقر عليه وستلاحظ أن لهذه الخاصية قيمتان **fals** و **true** حيث **fals** هي الإفتراضية غيرها أنت إلى **true**.



لقد تغير عنوان الفورم من اليسار إلى اليمين

ماذا تلاحظ الآن تغيرت و أصبحت من اليمين إلى اليسار أليس كذلك..

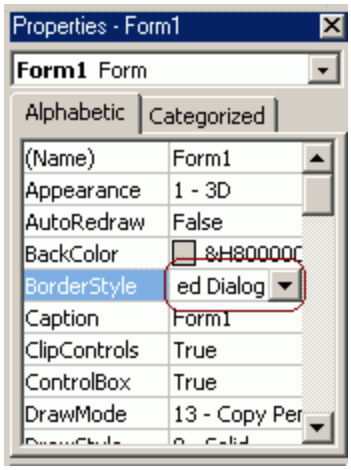
كيف تغير حجم الفورم ؟

تستطيع تغيير حجم الفورم بطريقة سهلة ... أنت ترى في الاعلى على الفورم السابق مربعات صغيرة زرقاء اللون أو في أركان الفورم ضع مؤشر الفارة فوق أحد تلك المربعات سيتغير شكل المؤشر إلى مؤشر ذو رأسين تستطيع من خلال ذلك تغيير حجم الفورم بالسحب و الإفلات.

و هناك طريقة أخرى لتغيير حجم الفورم ...إذهب إلى نافذة الخصائص انقر فوق السهم المتجه نحو الأسفل في عمود الإنزلاق الموجود على يمين نافذة الخصائص و ابحث عن خاصية العرض Width و خاصية الطول Heigth و غير إلى المقاس الذي تريده بالتحديد .... االعملية سهلة للغاية.

كيف تغير إطار الفورم برز إغلاق فقط أو بدون أزرار من فوق؟

في مربع الخصائص سوف تجد الخاصية Border Style أنقر فوق السهم للأسفل لعرض خيارات وتغيير شكل الفورم. أنظر إلى الصورة التالية :



هذا الجدول يبين شرح خصائص الفورم:

اسم الخاصية	معناها
Name	اسم الفورم
Appearance	الظهور
BackColor	لون الخلفية
BorderStyle	نمط الإطار
CauseValidation	التصحيح
Caption	العنوان
ControlBox	إظهار وإخفاء الأزرار العلوية
Enabled	الفاعلية
Font	نوع الخط
ForColor	لون الخط
Left	البعد عن الحافة اليسرى
Height	الإرتفاع
Icon	أيقونة الفورم
MaxButton	تفعيل وإلغاء زر التكبير
MDIChild	يعمل داخل فورم النافذة الأم

MinButton	تفعيل وإلغاء زر التصغير
RightToLeft	الإتجاه من اليمين لليسار
TabIndex	ترتيب Tab
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تشمل مؤشر الماوس
MoveAble	قابل للحركة
Picture	صورة الخلفية
StartUpPosition	موضع الفورم على الشاشة
TabStop	إيقاف الTab
ToolTipText	ماسيظهر عند وضع الماوس فوقها
Tag	Tag
Caption	النص
Top	البعد عن حافة النافذة العليا
Visible	مرئي أو مخفي
Width	العرض
WindowsStat	التشغيل حجم الفورم عن

شرح صندوق الأدوات

شكل الأداة	إسمها	شرحها
	Picture Box	صندوق عرض الصور
	Text Box	أداة النصوص والكتابة
	Label	أداة عرض النصوص
	Frame	إطارات داخل البرنامج
	Check Box	زر إختيار
	Combo Box	قائمة منسدلة
	Command Button	زر أمر
	Data Control	ربط قاعدة بيانات
	Drive Box	أداة عرض السواقات
	Dir Box	أداة عرض المجلدات
	File Box	أداة عرض الملفات
	Image	أداة الصور
	List Box	أدة القائمة
	Line	خط
	Ole Object	عنصر تحكمOle

	Option Box	زر خيارات
	Vertical Scroll Bar	شريط التمرير العمودي
	Horizontal Scroll Bar	أشرطة التمرير الأفقي
	Shape	أداة الأشكال
	Timer	أداة التوقيت

شرح نافذة الخصائص

في هذا الجدول خصائص عناصر التحكم القياسية

الخاصية	شرحها
name	تحديد اسم للاداة
backcolor	لون الخلفية
borderstyle	نوع وسمك اطار النافذة
caption	عنوان الاداة
text	عنوان الاداة
controlbox	مايظهر في اعلى كل نافذة في اقصى اليمن للتحكم بالنافذة
enabled	لتمكين الاداة
font	الخط
forecolor	اللون الامامي
height	الارتفاع
icon	الرمز
left	موقع الاداة من اليسار
mdichild	تستخدم في البرامج متعددة النوافذ
mousepointer	مؤشر الفارة
picture	الصورة
righttoleft	مفيدة جدا في البرامج العربية لتحويل القوائم من اليسار الى اليمين
top	موقع الركن العلوي من الاداة
visible	اخفاء الاداة
width	العرض
windowstate	حالة النافذة من حيث الحجم
backstyle	جعل الاداة شفافة
borderstyle	حذف الاطار المحيط بالاداة
databasename	اسم قاعدة البيانات
recordset	seek اذا استخدمنا البحث في قاعدة البيانات بالامر table يجب ان توضع في

	حالة
recordsource	تحديد الجدول من قاعدة البيانات
datafield	ربط اداة النص بحقل معين من جدول معين
datasource	ربط اداة النص بجدول معين من قاعدة البيانات
value	القيمة
max	الحد الاعلى
min	الحد الادنى
smallchange	التغيير الصغير
largechange	التغيير الكبير
interval	الفترة الزمنية المراد تنفيذ حدث المؤقت بعد مرورها
tooltiptext	التعليق الذي يظهر فوق الاداة عند ابقاء الفارة علية لمدة قصيرة
tabstop	توقيف التنقل بزر tab على زر معين
tabindex	ترتيب التنقل بين الادوات

يوجد طريقتين لكتابة الشفرة في الفيجوال بيسك :

1 - الخاصية:- القاعدة العامة لاي شفرة بلكود في البرنامج

مثلا (TEXT , LIBEL)

```
Object . property  = value
```

مثال :-

```
libel1  . text = "welcome"
```

2 - الطريقة :- القاعدة العامة لكتابة الشفرة

مثلا (LIST , COMBO)

```
Object . method  value
```

مثال : -

```
List1 . Additem  "Iraq"
```

الالوان في الفيجوال بيسك

يوجد 4 طرق للتعامل مع الالوان

1 - عن طريق الدوال الجاهزة منها دوال VB و هي .

VbRed أحمر

VbYellow أصفر

VbGreen أخضر

VbBlue أزرق

VbBlack أسود

VbWhite أبيض

VbGray رمادي

vbMagenta زهري مائل للبنفسجي

مثال :-

```
Form1.BackColor = vbGreen
Form1.BackColor = vbBlue
```

2 - و هناك دوال أخرى تسمى بدوال Qb و هي اختصار ل Quick Basic

و تسند إليها أرقام من 0 إلى 16 كالتالي

QbColor(1)

مثال :-

```
Form1.BackColor = QBColor (0)      اسود
Form1.ForeColor = QBColor (15)     ابيض
```

وهذا جدول يبين رقم كل لون في البرنامج

الرمز	اللون	الرمز	اللون
0	Black أسود	8	Grey رمادي
1	Blue أزرق	9	Light Blue أزرق فاتح
2	Green أخضر	10	Light Green أخضر فاتح
3	Cyan أزرق سماوي	11	Light Cyan سماوي فاتح
4	Red أحمر	12	Orange برتقالي
5	Pink زهري	13	Light Pink زهري فاتح
6	Brown بني	14	Yellow أصفر
7	White أبيض	15	Light White أبيض ساطع

3 - الطريقة الثالثة من خلال نظام (RGB) أي red green blue

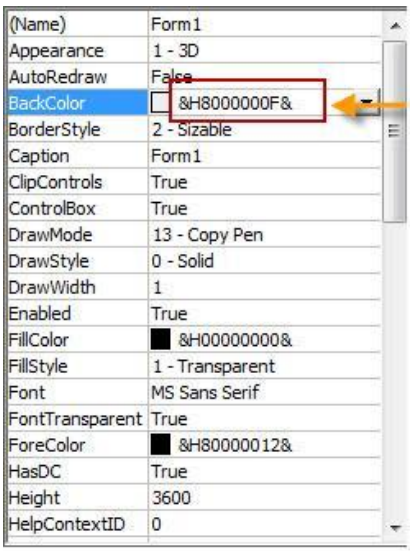
حيث تحدد كمية اللون الأحمر و الأخضر و الأزرق و هذه بصراحة للمحترفين بالألوان.

مثال :-

```
Form1.BackColor = RGB (255, 255 , 255)
```

حيث أعلى قيمة تكون 255

4 - تحديد اسم اللون من خلال خصائص الاداة .



خاصية موقع الفورم

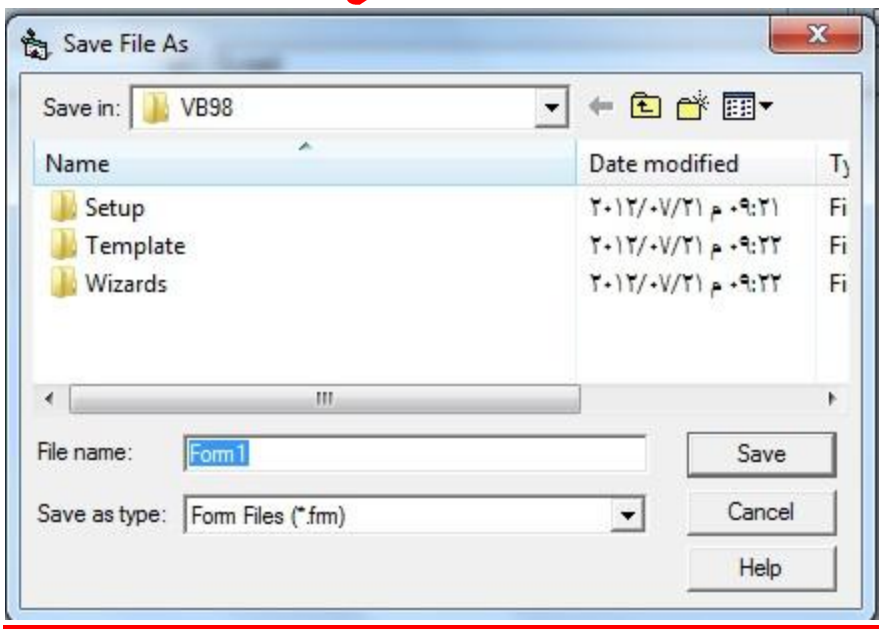
لتغيير موقع الفورم نتبع الاتي :

```
Form1.left=100

Form1.top=200

Form1.height=300

Form1.width=400
```

ويوجد طريقة سهلة نتجنب بها بطئ التنفيذ وكتابة اربعة اسطر من الاوامر وذلك بكتابة الامر move

```
Form1.Move 100, 200, 300, 400
```

حفظ المشروع



بعد أن عملت برنامجاً على الفيجوال بيسك فإنك سوف تحفظه وطريقة الحفظ هي :

1- عند إغلاق الفيجوال بيسك فسوف يتم سؤالك إذا كنت تريد الحفظ . تختار Yes فيظهر لك صندوق حوار ويكون قد أعطى لمشروعك إسم Project1

لنفرض أنك لم تغير الإسم . أنقر فوق الزر Save فيتم حفظ المشروع .

2- ويمكنك الحفظ من قائمة File ثم تختار Save وتحفظ المشروع

بعد ان حفظنا المشروع (البرنامج) باسم project1 اذهب قائمة File ثم اختر منها make project1.exe

طبعا عند حفظ البرنامج البرنامج فجوال بيسيك تلقائيا يضيف العبارة make project1.exe

ليسهل عليك العمل و هذا اكيد يختلف فعندك مثلا لا يظهر project1.exe make لكن يظهر اسم البرنامج الذي حفظته

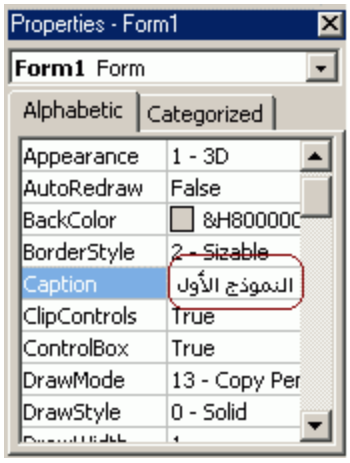
الان اذهب للمسار الذي كنت حفظت فيه البرنامج تراه قد عمل ايقونة لبرنامجك اوتوماتيكيا و عند النقر على الايقونة يفتح برنامجك الرائع .

التعامل مع النماذج

اضافة عدة فورمات للبرنامج

لقد قمنا سابقاً بشرح واجهة الفورم والأن سوف نتوسع بالتعامل معه مثلاً لو كان لديك عدة فورمات في برنامجك كيف ستظهر وتغلق النماذج .

1 – بعد فتح الفيجوال بيسك وتجهيز الفورم الأول سوف يكون إسمه بالأساس Form1 غير خاصية Caption أي العنوان إلى "النموذج الأول" .



أضف إلى الفورم زرين أمر وغير خاصية Caption لكل منهما

ثم أضف فورم ثاني Form2 وغير Caption إلى "النموذج الثاني" بواسطة add projedt >>> file
أضف إلى الفورم الثاني زر أمر "عودة للنموذج الأول"

في الفورم الأول "النموذج الأول" أنقر فوق الزر "إظهار النموذج الثاني " نقرتان لفتح المكان المخصص لكتابة الشيفرة أكتب الشيفرة التالية :

```
Unload Me
Form2.Show
```

السطر الأول يعني إغلاق الفورم الحالي أي "النموذج الأول"
السطر الثاني يعني إظهار الفورم الثاني الذي إسمه Form2

وفي الفورم الثاني أنقر فوق الزر "عودة للنموذج الأول " نقرتان وأكتب نفس الكود السابق ولاكن عليك تبديل Form2 إلى Form1
إذاً نستنتج أن شفرة إغلاق فورم لوحده هي Me Unload أو Form1.Hide إستبدل Form1 بإسم الفورم الذي تريد إخفائه

الأن عد إلى الفورم الأول وأنقر نقرتان فوق الزر " إغلاق البرنامج " وأكتب التعليمة التالية End

وهي تقوم بإغلاق كافة البرنامج أي كافة النماذج الموجودة
إذا تعليمة إغلاق نموذج واحد هي Me Unload
وتعليمة إغلاق كافة البرنامج هي End

هذا بالنسبة للإظهار والإغلاق . هناك تصغير الفورم عن طريق النقر فوق زر
أضف إلى أحد الفورمات السابقة زر أمر قم بستميته "تصغير"
أكتب التعليم التالية:

```
Form1.WindowState = 1
```

لاحظ أن WindowsState هي خاصية من خصائص الفورم ولها ثلاث خيارات

0 – Normal الوضع العادي
1 – Minimized أصغر حجم
2 – Maximized أكبر حجم

في التعليمة السابقة لو وضعت رقم 2 فإن النموذج سوف يكبر إلى ملىء الشاشة

كيفية كتابة الشفرة :

إذا أردت تنفيذ أي تغيير في خصائص أي أداة أثناء تشغيل العرض فيلزمك أن تكتب للبرنامج شفرة بذلك أي تعليمية برمجية أي (Code) و جميع الخصائص يتم تغيير خصائصها بهذه الطريقة :

اسم الأداة . اسم الخاصية = القيمة

مثال:-

إذا أردت تغيير الكتابة على الزر إلى (تم الضغط على الزر) مثلا

```
CommandButton1.Caption = " تم الضغط على الزر"
```

اسم الأداة — أساسية — خاصية التسمية — أساسية — قيمة نصية

و بالمثل : إذا أردت تغير حجم الخط إلى حجم (20) لمربع النص تكتب هذه الشفرة :

خاصية سمك (حجم) الخط

```
TextBox1. Font.Size = 20
```

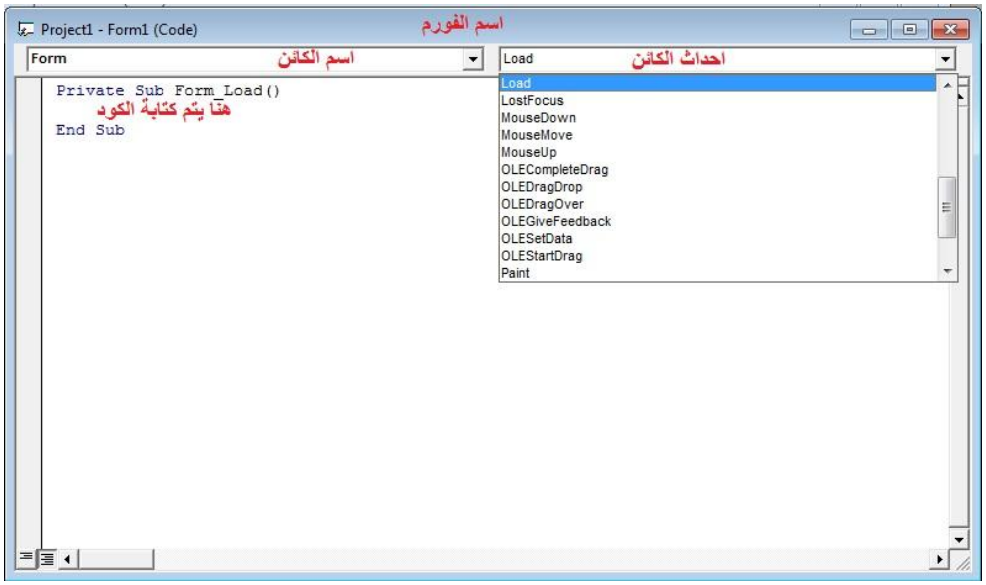
اسم الأداة — أساسية — خاصية الخط — أساسية — قيمة حجم الخط

مثلا تريد أن تخفي أداة تستعمل الخاصية Visible كالتالي :

```
TextBox1.Visible = False
```

اسم الأداة — أساسية — خاصية إخفاء و إظهار — أساسية — قيمة false تعني إخفاء

لكن ستحتاج إلى حدث تضع فيه هذا الكود مثل :
إذا أردت أن إذا ضغطت على الزر (CommandButton1) يقوم بإخفاء مربع النص (TextBox1) فأنت يلزمك حدث الضغط على الزر فقم بالضغط المزدوج على الزر (CommandButton1) ستظهر لك هذه الصورة هي عبارة عن محرر الشفرة في هذا العرض.



قم بكتابة هذه الشفرة بداخله :

```
Private Sub CommandButton1_Click()
TextBox1.Visible  = False
End Sub
```

و تستطيع أن تلغي عمل شفرة ما بدون مسحها بوضع قبلها العلامة (') مثل :

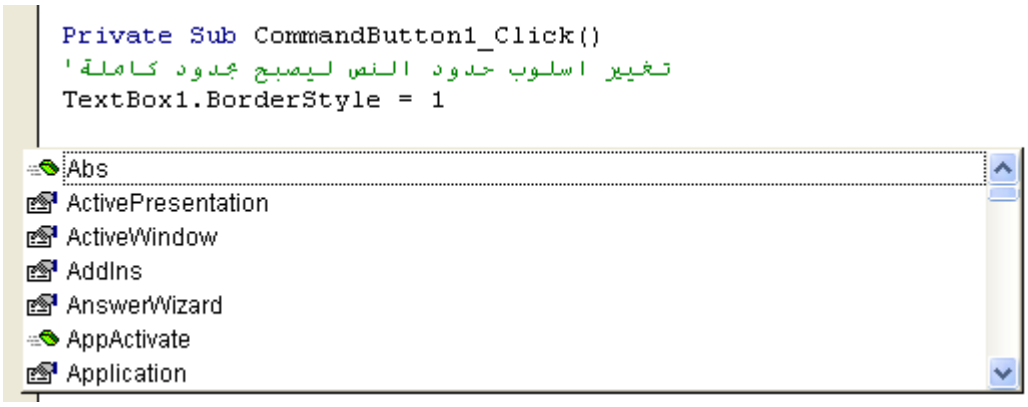
```
Private Sub CommandButton1_Click()
'TextBox1.Visible  = False
TextBox1.BorderStyle  = 0
End Sub
```

هذه الشفرة لن تنفذ بسبب العلامة (') في بدايتها

و حتى تستطيع كتابة تعليق على الشفرة بنفس الطريقة .

```
Private Sub CommandButton1_Click()
'تغيير أسلوب حدود النص ليصبح بحدود كاملة
TextBox1.BorderStyle  = 1
End Sub
```

قد تواجه صعوبة في حفظ الشفرات أنصحك بعدم حفظها بل اجعل البرنامج هو من يساعدك فقم بما يلي:
إذا أردت أن يساعدك البرنامج فقط قم بالضغط على زر **Ctrl** من لوحة المفاتيح مع المسافة سوف يظهر لك جميع الشفرات المتاحة مثل التالي :
اذهب لمحرر الشفرات و المكان الذي تريد وضع فيه الشفرة ثم اضغط على المسافة و **Ctrl** .
ستظهر لك مثل ما بالصورة التالية :



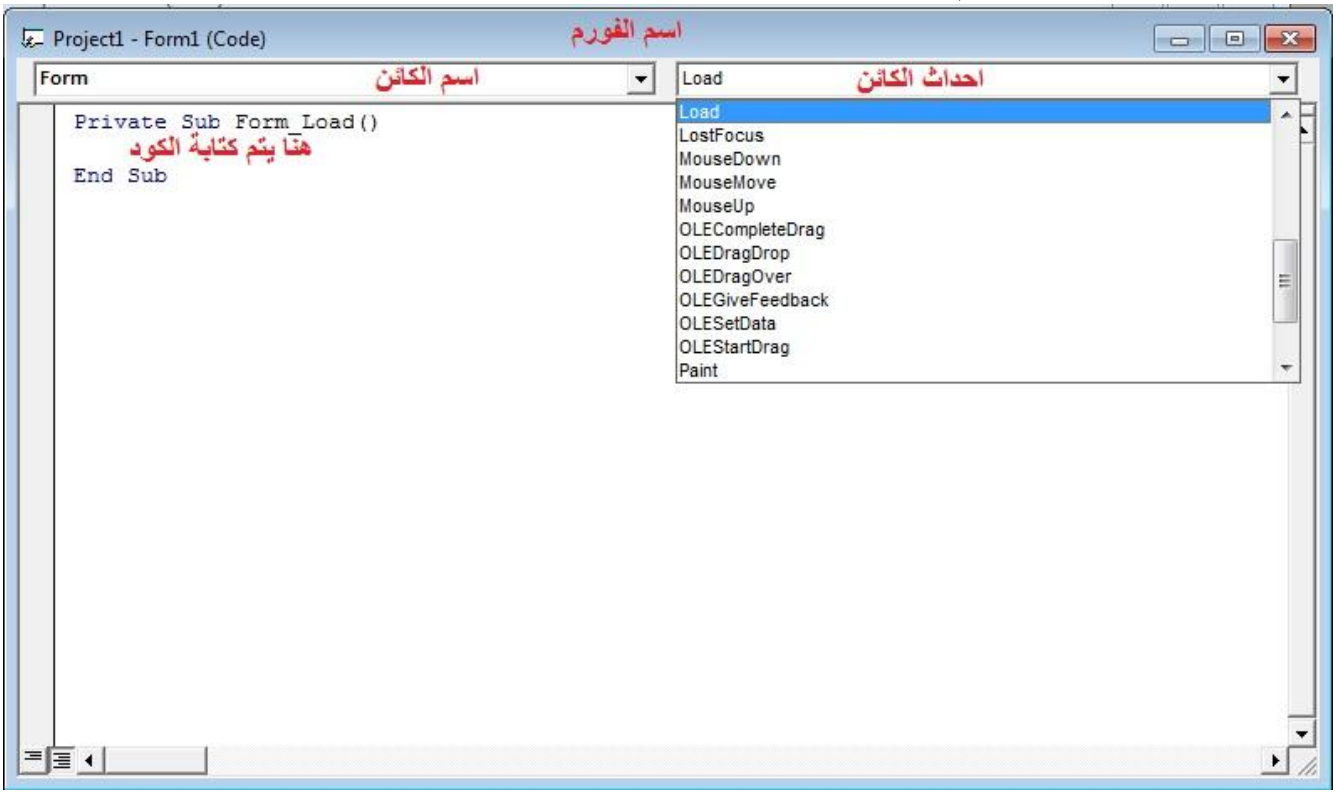
اكتب الحروف التي تريدها أو اسم الكائن و سيساعدك البرنامج في اختيار اسم الأداة و حتى الخصائص .

نصيحة : إذا أخطأت في حرف واحد أو حتى رقم لن يتم تشغيل الشفرة فتأكد دائما من الحروف و استعمل مساعد الشفرة دوما .

الأحداث Events:

إذا ارت ان تنقر بفأرة (الماوس) على شيء فهذا يسمى حدث و الفجوال بيسك 6 لا يقدم هذا الحدث فقط بل جملة من الأحداث المختلفة لكل الأدوات . و سنأخذ المهم منها :

قم بالضغط المزدوج على الزر (CommandButton1) ستظهر لك هذه الصورة :
من هنا يتم تغيير الأحداث للأدوات.



اضغط على السهم الذي يغير الأحداث ستظهر لك الأحداث المتاحة لهذا الكائن .
و تستطيع نسخ و لصق و الحذف أي شفرة (code) بدون خوف مما يجعل إمكانية التحرير للشفرة واسعة .

و تختلف أحداث الكائنات على حسب الاحتياجات ، اختر الأحداث المناسبة لتنفيذ عملك و سنأخذ معلومات عن الأحداث .

مجموعة من الأحداث :

:Click
و هو حدث الضغط بالفأرة مرة واحدة .

:Dblclick
و هو حدث الضغط المزدوج بالفأرة.

:DragDrop
و هو حدث السحب بالفأرة ثم الإفلات .

:MouseMove
و هو حدث يظهر عندما تحرك الفأرة .

:MouseDown
و هو حدث يظهر عندما تستمر بضغط الفأرة على الكائن.

:MouseUp
و هو الحدث الذي يظهر عندما تسحب يدك من الضغط من الفأرة على الكائن.

:KeyDown
و هو حدث يظهر بعدما تضغط على لوحة المفاتيح.

:KeyUp
و هو حدث يظهر بعدما تسحب يدك من الضغط على لوحة المفاتيح.

:KeyPress
و هو حدث يظهر عندما تضغط على أزرار معينة على لوحة المفاتيح

ملاحظة : * للسهولة تستطيع نسخ و لصق و حذف أي شفرة من مكان و ضعها في مكان آخر.
* الشفرة يتم ترجمتها بالتسلسل سطرا سطراً أي ما كتبته أولا يتم ترجمته ثم الذي يليه .

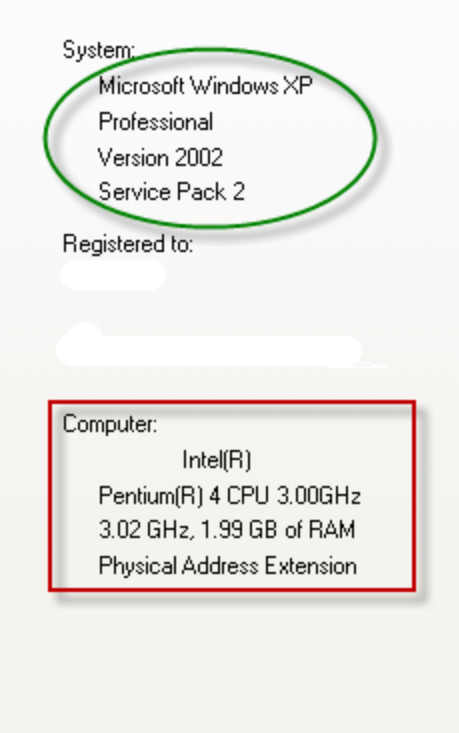
التعامل مع Label و Text

أولا التعامل مع الأداة Label

مثل عرض مواصفات الكومبيوتر من خلال

My Computer >>>> properties

فتظهر النافذة التالية :



تستخدم الأداة Label لإظهار النصوص على النموذج دون أن تتمكن من الكتابة فيها .

لتغير النص الذي سيظهر بداخلها طبعاً عن طريق Caption

وهذا الجدول يبين الخصائص الأساسية Label

اسم الخاصية	معناها
Name	اسم الأداة
Aignment	محاذاة
Appearance	الظهور
AutoSize	حجم تلقائي حسب النص
BackColor	لون الخلفية
BorderStyle	نمط الإطار
Caption	النص

Enabled	الفاعلية
Font	نوع الخط
ForColor	الخط لون
Left	البعد عن الحافة اليسرى
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تشمل مؤشر الماوس
RightToLeft	الإتجاه من اليمين لليسار
TabIndex	ترتيب Tab
TabStop	إيقاف الTab
ToolTipText	ماسيظهر عند وضع الماوس فوقها
Tag	Tag
Caption	النص
Top	البعد عن حافة النافذة العليا
Visible	مخفي أو مرئي
Width	العرض

كتابة كود عدة خصائص ل text بواسطة with

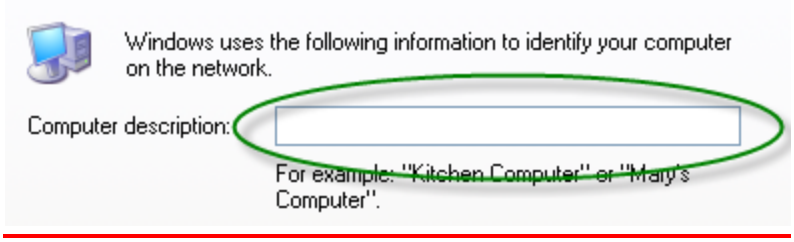
```
With text1
.text1="فيجوال بيسك ٦"
.font.bold=true
.font.name="rod"
.font.size=40
.backcolor=vbblak
..forecolor=vbwhite
```

التعامل مع TextBox

مثل عرض وتغيير اسم الكومبيوتر

My Computer >>>> properties

فتظهر النافذة التالية :



أداة TextBox تستخدم في إدخال النصوص إلي البرنامج كما يمكن إستخدامها في إخراج النصوص.
وهي لاتختلف كثيراً عن Label سوى أنها يمكن الكتابة فيها أما Label فلا يمكن الكتابة فيها .

وهذا جدول الخصائص الأساسية لTextBox

اسم الخاصية	معناها
Name	اسم الأداة
Aignment	محاذاة
Appearance	الظهور
AutoSize	حجم تلقائي حسب النص
BackColor	لون الخلفية
BorderStyle	نمط الإطار
CauseValidation	التصحيح
Text	النص
Locked	مقفول لا يمكن الكتابة بداخلها
MaxLength	الحد الأقصى
Enabled	الفاعلية
Font	نوع الخط
ForColor	لون الخط
Left	البعد عن الحافة اليسرى
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تشمل مؤشر الماوس
PasswordChar	حرف كلمة المرور
MultiLine	هل تستخدم لعدة أسطر أم لا
ScrollBars	أشرطة التمرير
RightToLeft	الإتجاه من اليمين لليسار
TabIndex	ترتيب Tab
TabStop	إيقاف الTab
ToolTipText	ماسيظهر عند وضع الماوس فوقها
Tag	Tag
Caption	النص
Top	البعد عن حافة النافذة العليا
Visible	مرئي أو مخفي
Width	العرض

إن التعامل مع هاتين الأداتين ليس صعباً ، عليك فهم صندوق الخصائص فتعرف كيف تتعامل مع كافة الأدوات

إذا أردت أن تربط بين جملتين استعمل الرمز & و ممكن أن تستعمل بدلها (+) . و هي تتم هكذا :

```
Text1.Text = "حسن" & " زينب"
```

و النتيجة ستكون (حسن محمد) مثل نتيجة الجملة التالية :

```
Text1.Text = "حسن محمد"
```

آو حتى تربط بن جملة و رقم .

التعامل مع Picture Box و Image Box

صندوق الصور PictureBox

NAME	أسم الأداه
Aignment	محاذاة
Appearance	نوع الإطار
AutoSize	حجم تلقائي حسب الصورة
ToolTipText	ماسيظهر إذا وضع الماوس فوقها
Picture	الصورة
BackColor	الون الخلفيه
Visible	مخفي أم مرئي

يستخدم هذا الصندوق لعرض الصور بحجمها الطبيعي وله إستخدامات أخرى كثيرة فهو يستخدم كإطار لأدوات تحكم أخرى
وهو أيضاً يستخدم لعرض الفيديو ويستخدم للرسم والإنشاء ويمكن تحويله إلى أداة عرض نصوص وزر امر ، سوف نتحدث عن ذلك لاحقاً.

لوضع صورة : من مربع الخصائص الخاص ب Picture سوف تجد الخاصية Picture أنقر فوق الزر لتحديد صورة .

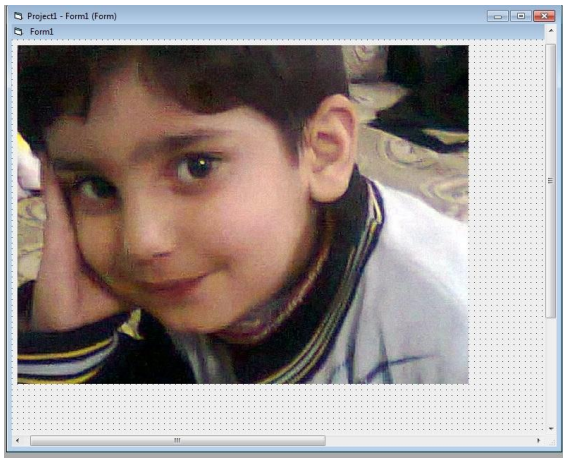
أو عن طريق الكود التالي في حدث Load للفورم :

```
Picture1 . Picture = LoadPicture (هنا ضع عنوان الصورة)
```

يمتاز صندوق الصور Picture عن صندوق عرض الصور Image بمايلي:

- يعرض الصورة بحجمها الطبيعي ولو صغرت إطار الصندوق فلن تصغر الصورة. بعكس Image
- يمكن تغيير لون الخلفية ونمط الإطار أما Image لا
- يستخدم في أغراض كثيرة أما Image لعرض الصور فقط
-

هذا شكل الأداة على الفورم



هذه الخصائص الأساسية ل Picture Box قارن بينها وبين خصائص Text:

اسم الخاصية	معناها
Name	اسم الأداة
Aignment	محاذاة
Appearance	نوع الإطار
AutoSize	حجم تلقائي حسب الصورة
BackColor	لون الخلفية
BorderStyle	نمط الإطار
Caption	النص
Enabled	الفاعلية
Font	نوع الخط
ForColor	لون الخط
Height	الإرتفاع
Index	ترتيب مفتاح Tab
Left	البعد عن الحافة اليسرى
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تشمل مؤشر الماوس
Picture	الصورة
RightToLeft	الإتجاه من اليمين لليسار
TabIndex	ترتيب Tab
TabStop	إيقاف الTab
ToolTipText	ماسيظهر عند وضع الماوس فوقها
Tag	Tag
Caption	النص
Top	البعد عن حافة النافذة العليا
Visible	مخفي او مرئي
Width	العرض

صندوق الصورة Image Box:

ذكرنا سابقاً أنه يستخدم فقط لعرض الصور وتكبيرها وتصغيرها حسب ماتريد .

لوضع صورة : من مربع الخصائص الخاص ب Image سوف تجد الخاصية Picture أنقر فوق الزر لتحديد صورة .

أو عن طريق الكود التالي في حدث Load للفورم :

```
Image1 . Picture = LoadPicture (هنا ضع عنوان الصورة)
```

لتغير حجم الصورة هناك خاصية تحت خاصية Picture إسمها Stretch عليك تبديلها إلى True

هذا شكل ال Image على الفورم لاحظ حجم الصورة الطبيعي في أعلى الصفحة ولاحظ هنا.

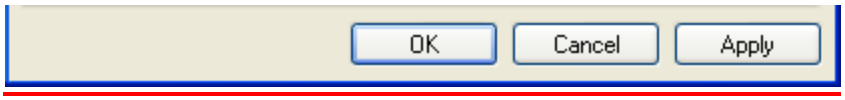
وهذا جدول الخصائص الأساسية ل Image

اسم الخاصية	معناها
Name	اسم الأداة
Aignment	محاذاة
BorderStyle	نوع الإطار
Enabled	الفاعلية
Font	نوع الخط
ForColor	لون الخط
Height	الإرتفاع
Index	ترتيب مفتاح Tab
Left	البعد عن الحافة اليسرى
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تشمل مؤشر الماوس
Picture	الصورة
Stretch	السماح بتغيير حجم الصورة
TabIndex	ترتيب Tab

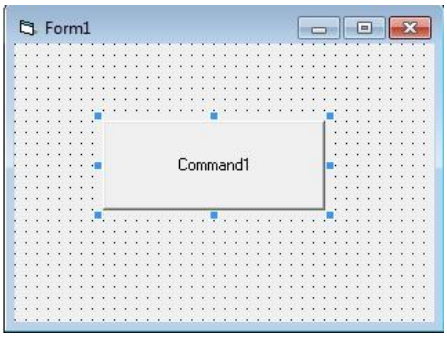
TabStop	إيقاف الTab
ToolTipText	ماسيظهر عند وضع الماوس فوقها
Top	البعد عن حافة النافذة العليا
Visible	مخفي او مرئي
Width	العرض

التعامل مع زر الأمر Command Button

وشكلها في الكومبيوتر كما مبين :



زر الأمر هو من الأدوات المهمة في الفيجوال بيسك وإذا عملت أي برنامج فسوف تحتاج بالتأكيد إلى وضع زر أمر في برنامجك ، فهو الأداة الأساسية للأحداث



لكتابة أي حدث في الزر أنقر فوقه نقرتين ثم أكتب الشيفرة التي تريد

لتغيير النص الذي سيظهر في الزر طبعاً تعرفها في خاصية Caption أكتب التسمية.

ويمكنك أيضاً إضافة صورة إلى الزر ويمكنك تغير اللون . ولاكن عليك أن تبدل خاصية الستايل Style إلى Graphical

ويمكن أن يستخدم الزر ليعمل أكثر من حدث في نفس الوقت في Click و MouseDown و MouseUp

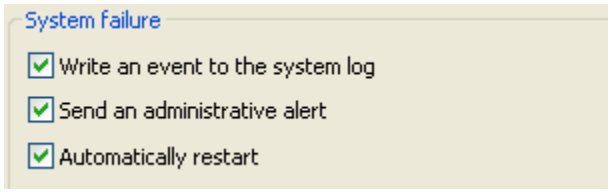
هذه الخواص الأساسية للزر

اسم الخاصية	معناها
Name	اسم الأداة
Appearance	الظهور
BackColor	لون الخلفية
Caption	النص

Default	زر إفتراضي عند ضغط مفتاح الإدخال
DisablePicture	الصورة عندما يكون غير متاح
DownPicture	الصورة عند الضغط لأسفل
Enabled	الفاعلية
Font	نوع الخط
ForColor	الخط لون
MaskColore	قناع اللون
Style	النمط
Left	البعد عن الحافة اليسرى
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تشمل مؤشر الماوس
RightToLeft	الإتجاه من اليمين لليسار
TabIndex	ترتيب Tab
TabStop	إيقاف ال Tab
ToolTipText	الماوس فوقها ماسيظهر عند وضع
Tag	Tag
Picture	الصورة في الوضع العادي
Top	البعد عن حافة النافذة العليا
Visible	مخفي او مرئي
Width	العرض

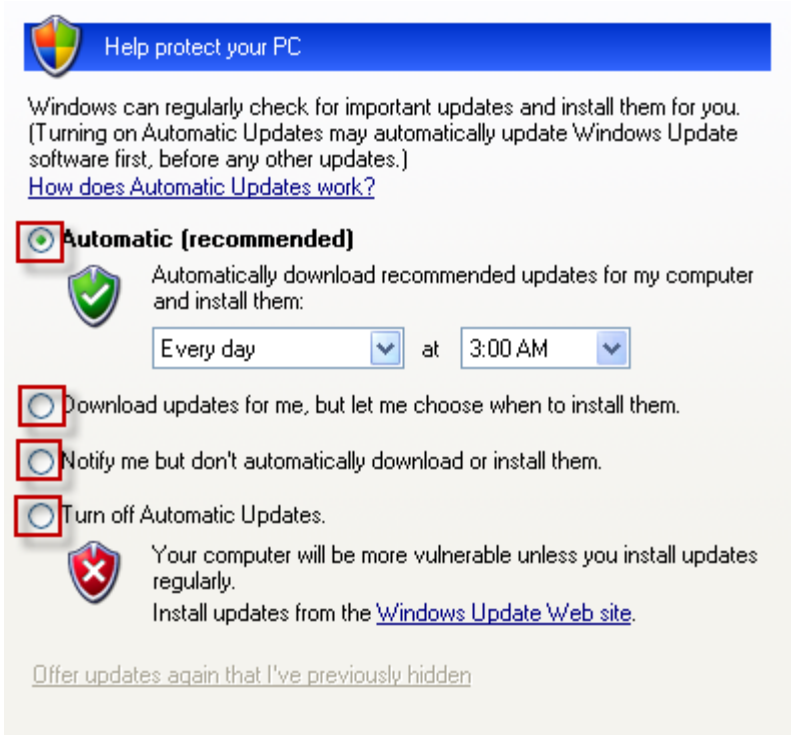
التعامل مع Check Box و Option

Check Box - 1



اذا فرضنا انه يوجد علي الفورم 3 منها يمكننا اختيار واحد فقط او اثنين او الثلاثه معا ويمكننا ايضا ان لا نختار منهم شيئا

Option - 2



ولنفرض انه يوجد منها 3 واخترنا الاول ثم نأتي لنختار الثاني سيلغي اختيار الاول ويقوم بأختيار الثاني اي ان الاختيار من واحده منها يلغي عمل الاخر

اهم خصائصه: (Caption , Font, Enabled, BackColor, Forecolor, value and visible)

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

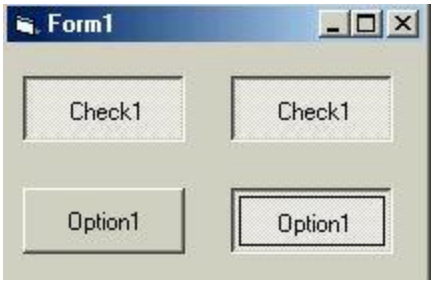
وكود الاداة Option كلتالي ويتم كتابة الامر بعد كلمة Then .

```
If Option.Value = True Then
```

او لاظهار صورة مثلا نكتب

```
Image1.Visible = False
Image2.Visible = False
Image3.Visible = True
```

وعليك أن تعرف أن هنا نمطان لكل أداة أنظر الشكل في السفل



تم تغيير النمط عن طريق خاصية Style إلى Graphical

ولاحظ كيف تمكنا من تحديد خانتي الإختيار بينما لم نحدد سو زر خيار واحد.

لتغيير القيمة الإفتراضية لكل أداة من خاصية Value في مربع الخصائص

وللاداتين نفس الخصائص وهذا جدول للخصائص الأساسية:

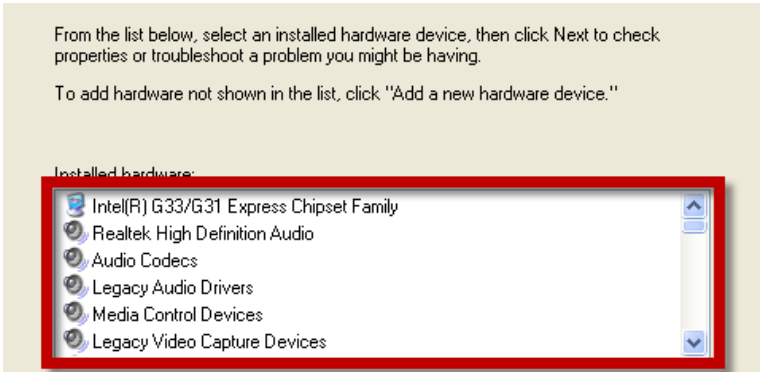
اسم الخاصية	معناها
Name	اسم الأداة
Aignment	محاذاة
Appearance	الظهور
BackColor	لون الخلفية
Caption	النص
DisabledPicture	الصورة عند عدم الفاعلية
DownPicture	الصورة عند النقر للأسفل
Enabled	الفاعلية
Font	نوع الخط
ForColor	الخط لون
Left	البعد عن الحافة اليسرى
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تشمل مؤشر الماوس

RightToLeft	الإتجاه من اليمين لليسار
TabIndex	ترتيب Tab
TabStop	إيقاف الTab
ToolTipText	الماوس ما سيظهر عند وضع فوقها
Tag	Tag
Picture	الصورة
Style	النمط
Value	القيمة
Top	البعد عن حافة النافذة العليا
Visible	مخفي او مرئي
Width	العرض

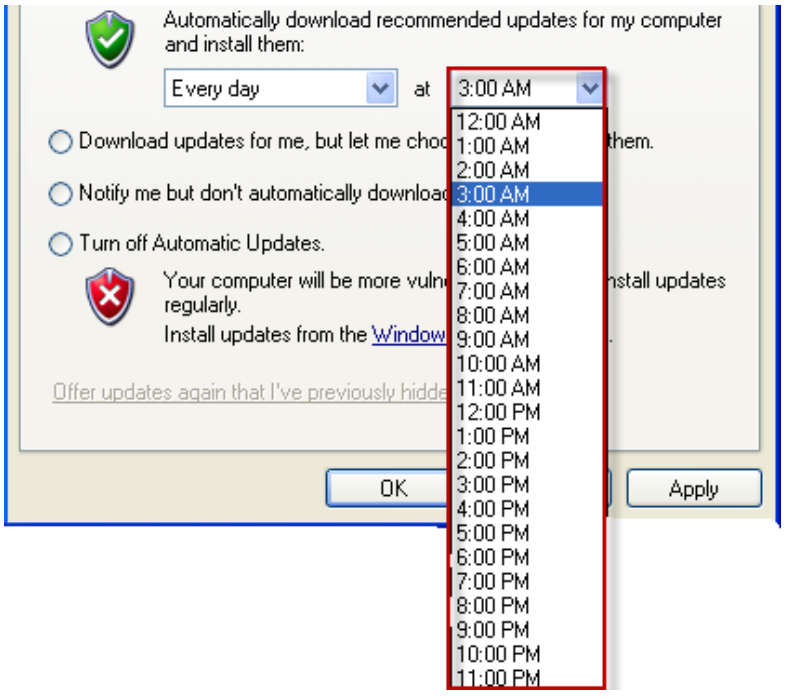
التعامل مع List Box و Combo Box

الكومبو والليست يستخدمان لضم قائمة للإختيار منها لاحظ الشكل.

List Box - 1



Combo Box - 2



ملاحظة : يستخدم في الاداتين القاعدة العامة الثانية لكتابة الكود (Object . method value)

لايختلف ال combo عن ال list كثيراً سوى أن ال combo لايظهر القائمة إلى عند النقر عليه ويستخدم الكومبو أيضاً للكتابة بداخله.

طريقة الإضافة إلى الأداتين هي نفس الطريقة:

1- عن طريق الخاصية List يمكنك كتابة العناصر

2- إذا كنت تريد إضافة عناصر عند تحميل البرنامج عليك كتابة هذا الكود في حدث Load للفورم:

```
List1.AddItem "النص هنا"
.
.
.
Combo1.AddItem "النص هنا"
```

3- أما إذا كنت قد وضعت أداة الكتابة TextBox فأكتب الكود التالي:

```
List1.AddItem Text1.Text
Combo1.AddItem Text1.Text
```

إذا لإضافة نص عليك وضعه بين علامتي الإقتباس " "

ولإضافة النص من صندوق كتابة ضع إسم ال text دون علامات الإقتباس

ويمكنك تعديل اليست ListBox للسماح بتحديدات متعددة. أنظر الشكل في الأسفل

وذلك عن طريق تعديل الخصائص Style و MultiSelect

ويمكن ربط ال combo و ال list بقاعدة بيانات جاهزة ويتم عرض السجلات بداخلهما .

هذا جدول الخصائص الأساسية ل Box List

اسم الخاصية	معناها
Name	اسم الأداة
Appearance	الظهور
columns	عرض التحديد
Back Color	لون الخلفية

Enabled	الفاعلية
Font	نوع الخط
ForColor	الخط لون
Left	البعد عن الحافة اليسرى
MultiSelect	السماح بتحديدات متعددة
Style	يحوي على خانات إختيار
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تشمل مؤشر الماوس
RightToLeft	الإتجاه من اليمين لليسار
TabIndex	ترتيب Tab
TabStop	إيقاف الTab
ToolTipText	الماوس فوقها ماسيظهر عند وضع
Tag	Tag
Caption	النص
Top	البعد عن حافة النافذة العليا
Visible	مخفي او مرئي
Width	العرض

وخصائص الكومبو ComboBox هي نفس الخصائص السابقة سوى بعض التغييرات:

Text وتعني النص الذي سيظهر في التحديد الأول للكومبو

Locked ممنوع من الكتابة بداخله

وتختلف خاصية الستايل Style أيضا

التعامل مع أشرطة التمرير

أشرطة التمرير تستخدم لتغيير القيمة زيادة أو نقصان سوف نأخذ مثالاً لهذه الأداة أفضل من الشرح .

لهذه الأداة نوعان :

شريط تمرير أفقي : HScroll

شريط تمرير عمودي : VScroll

ملاحظة:شريط التمرير الأفقي والعمودي لهما نفس الوظيفة والخصائص والأحداث ولكن الفرق في أتجاه الشريط فقط لا غير

بعد تجهيز الفورم أضف إليه ثلاث أشرطة تمرير إما أفقي أو عمودي . لنفرض أنك وضعت أشرطة أفقية HScroll .

أسمائها على التتالي: HScroll3 HScroll2 HScroll1

ثم أضف إلى الفورم صندوق الصورة Picture1

ولاتنسى الأدوات Label والتي تعني النصوص التي بجانب كل شريط تمرير

أنقر فوق الفورم نقرتين وأضف الكود التالي في حدث Load :

```
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
```

شرح الأسطر السابقة : وتعني تحديد القيمة الصغرى والكبرى لأشرطة التمرير وتحديد سرعة التمرير .

أضف الكود التالي إلى شريط التمرير الأول:

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

الكود التالي للشريط الثاني:

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

الكود التالي للشريط الثالث :

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

الحدث بعد تشغيل البرنامج جرب تبديل أشرطة التمرير ولاحظ

يبين الخصائص الأساسية لأشرطة التمرير هذا الجدول

اسم الخاصية	معناها
Name	اسم الأداة
Enabled	الفاعلية
LargeChange	قيمة التبديل الكبرى
Max	الحد الأقصى
Min	الحد الأدنى
SmallChange	قيمة التبديل الصغرى
Value	القيمة الإفتراضية
Left	البعد عن الحافة اليسرى
MousePointer	مؤشر الماوس
MouseIcon	الأيقونة التي تشمل مؤشر الماوس
RightToLeft	الإتجاه من اليمين لليسار
TabIndex	ترتيب Tab
TabStop	إيقاف الTab
ToolTipText	الماوس فوقها ما سيظهر عند وضع
Tag	Tag
Top	البعد عن حافة النافذة العليا
Visible	مخفي او مرئي
Width	العرض

التعامل مع أدوات عرض الملفات

أدوات عرض الملفات هي ثلاث :

صندوق عرض محركات الأقراص Driver List Box

استعمالها : الأداة تقوم بإظهار شتى الأقراص المركبة على الحاسوب .

صندوق عرض المجلدات Dir List Box

استعمالها : الأداة تقوم بإظهار المجلدات الموجودة في قرص ما أو في مجلد آخر .

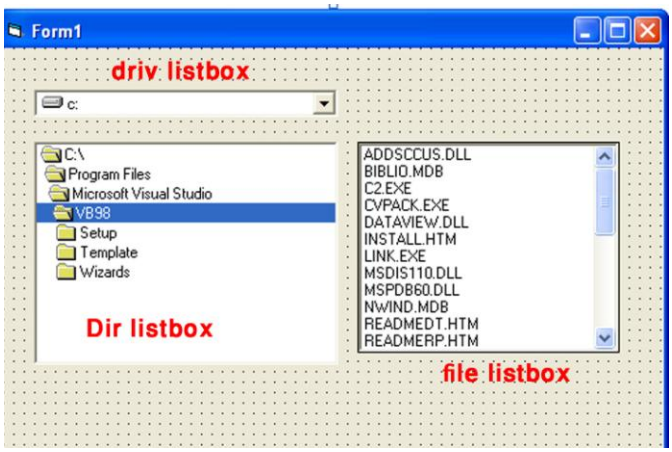
صندوق عرض الملفات File List Box

استعمالها : الأداة تقوم بإظهار الملفات الموجودة في مجلد ما أو قرص ما .

وهذه الأدوات يجب أن تكون مترابطة ببعضها .

لنعمل نموذج بسيط لتعرف كيف تربطها ببعضها:

بعد تجهيزك للفورم أضف إليه الأدوات السابقة . وأضف أدوات النصوص Label كالشكل التالي:



يمكنك كتابة المسار مباشرة للخاصية

```
Dir1.Path = "C:\Windows"
File1.Path = "C:\Winnt"
```

أنقر نقرتين فوق صندوق محركات الأقراص Drive1 وأكتب:

```
On Error Resume Next
Dir1.path = Drive1.drive
```

السطر الأول يعني عند الخطأ لاتتابع مثلاً لو حدد محرك القرص اليزري ولم يكن هناك قرص فسوف يعطيك رسالة خطأ. أما هذا السطر يمنع الخطأ ومتابعة العمل أيضاً

في السطر الثاني عرض المجلدات داخل محرك الأقراص الذي تم تحديده

أنقر فوق صندوق المجلدات Dir1 وأكتب :

```
File1 . path = Dir1 . path
Label 4 = Dir1
```

السطر الأول: عرض الملفات التي داخل هذا المجلد (في صندوق الملفات File1)
السطر الثاني: عرض المسار الكامل في أداة النصوص Label4

أنقر نقرتين فوق صندوق الملفات File1 وأكتب الشيفرة التالية :

```
Label4 = Dir1 + "\" + File1
```

ويعني عرض المسار الكامل للملف في Label4
جرب البرنامج ولاحظ الحدث
هذه الأدوات مهمة في أي برنامج فهي تستخدم لفتح ملف مثلاً.

ملاحظة: يمكنك تحديد نوع الملفات التي تريده أن يظهر في صندوق عرض الملفات صور Bmp ملفات صوت Wav وذلك عن طريق خاصية Pattern

```
file1 . pattern
```

مثال :-

لعرض كافة التنسيقات أكتب مايلي:

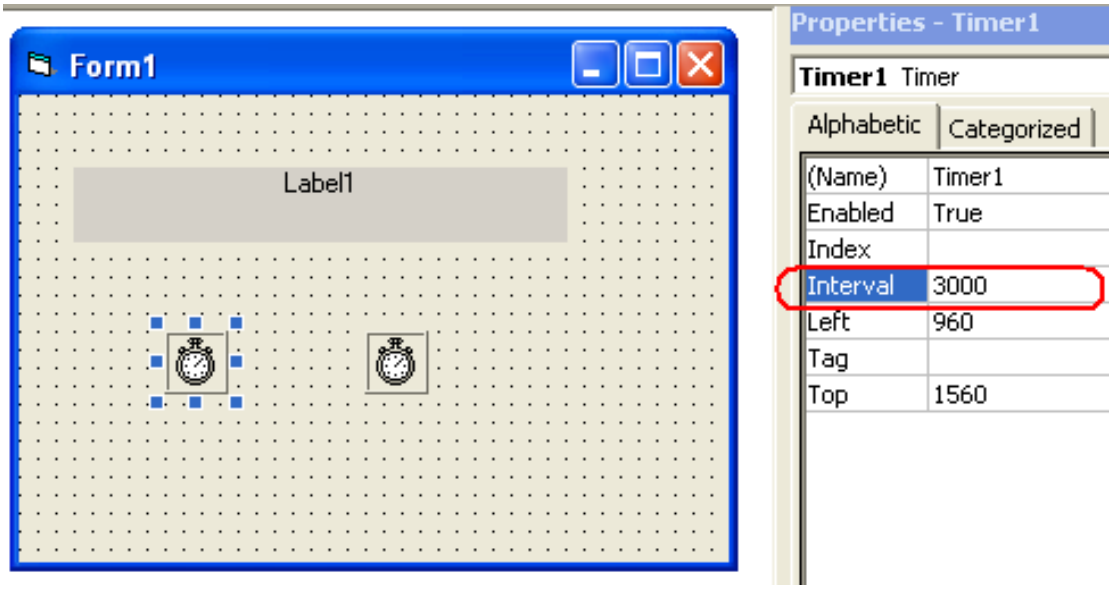
أكتب *.exe لعرض البرامج فقط
فقط عليك أن تعرف ملحق التنسيق وكتابته بدلاً من exe لعرضه في الصندوق
عرض نوعين أو أكثر دون عرض كافة الأنوع كمايلي: لنفرض تريد عرض البرامج exe والصور bmp تكتب كما يلي:

```
*.exe;*.bmp
```

التعامل مع أداة التوقيت Timer

وهي اداة تستخدم لعمل حدث معين بعد مرور فترة معينة او لتكرار الحدث طول فترة زمنية معينة . واهم خاصية من خواص تلك الاداة هي خاصية interval وهي التي تحدد الفترة الزمنية لتنفيذ الحدث او الكود وتحدد الفترة الزمنية بالملي ثانية ms فمثلا لو ارت ان يعمل كل ثانية عليك كتابة 1000 للخاصية interval القيمة الكبرى للتوقيت هي (65535) فانها تساوي تقريبا دقيقة واحدة و 5 ثواني ونصف الثانية .

ملاحظة : يجب وضع قيمة ل Interval حسب المشروع والا لن يحدث اي تغيير للمشروع .



لتعطيل التايمر هناك خاصية Enabled تتيح تعطيله وتشغيله

مثال : ساعة رقمية

إصنع فورم وأضف إليه Timer و Text
غير خاصية Interval إلى 500 أو أي قيمة تحت 1000
أنقر فوق التايمر نقرتين وأكتب السطر التالي

```
Text1=Time
```

شغل المثال سوف يظهر لك الوقت متغيراً في أاة النصوص Text

مثال2: - لتغيير اسم libel بعد مدة زمنية

```
Label1.Caption = "vb"
```

مثال3: - لتغيير لون libel بعد مدة زمنية

```
Label1. BackColor = vbBlue
```

ملاحظة :- لا تحاول الاكثار من الاكواد الطويلة بداخل الحدث timer للاداة timer فذالك يسبب استهلاك كبير للمعالج مما يؤدي الى ابطاء تنفيذ برامج و البرامج الاخرى في نظام التشغيل .

دوال الوقت و التاريخ

يجب ان توضع بين علامتين# #

```
Dim MyDate As Date
MyDate = #1/20/2001#
Print MyDate

MyDate = #2:30:00 AM#
Print MyDate

MyDate = #1/20/2001 2:30:00 AM#
Print MyDate
```

الهجري و الميلادي

VBA.Calendar = vbCalHijri هجري

VBA.Calendar = vbCalGreg ميلادي

الدالة now تشمل الوقت و التاريخ معاً

لاظهار التاريخ في Label1 — Label1.Caption = Date

لاظهار الوقت في Text1 — Text1.Text = Time

لاظهار الوقت و التاريخ في Label1 — Label2.Caption = Now

الخاصية	العمل
Now	استرجاع التاريخ والوقت الآن
Date	إدراج تاريخ
Day	أدارج رقم اليوم خلال الشهر 1- 31
Day of year	أدارج رقم اليوم خلال السنة 1- 366
Hour	أدارج الساعة 0-23
Minute	أدراج الدقيقة 1-59
Second	أدارج الثانية0-59
Month	أدراج اسم الشهر 1- 12
الطريقة Method	العمل
Add days	إنشاء تاريخ جديد متقدم أو متأخر بعدد أيام مثل x. adddays(1)
Add hour	إنشاء تاريخ جديد متقدم (أو متأخر)عدد من الساعات x.addhoure(5)
Add minutes	إنشاء تاريخ جديد متقدم أو متأخر عدد من الدقائق x.addminutes (30)

مثال: لعرض تاريخ اليوم

```
Dim x as Date
    X = now
Label1.text=x.date
```

مثال: لعرض الساعة

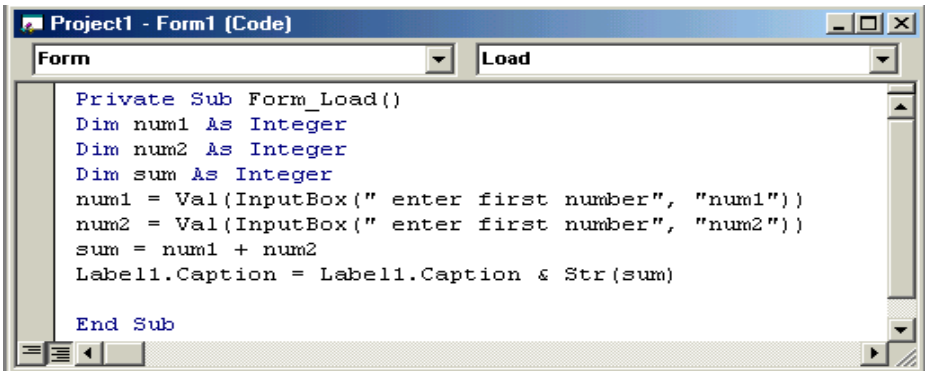
```
Dim x as Date
    X = now
Label1.text=x.Hour
```

مثال: لعرض ساعة رقمية تعرض الوقت و التاريخ الحالي

```
Label1.text=date.Now
```

صندوق الادخال Input box

يستخدم لعرض نافذة مستقلة وكتابة المعلومات فية .



عند كتابة سطر الاوامر التالي :

```
Num1 = val(inputbox("enter first number",num2"))
```

تظهر النافذة السابقة لادخال الرقم الاول او الثاني ....الخ

المصفوفات والمتغيرات

المصفوفة هي قيم مخزنة في البرنامج يتم الإضافة إليها وعرضها.

الاعلان عن متغير :-

هناك ثلاثة طرق للاعلان عن هذا النوع تعتمد على النطاق scope الذي سوف تغطيه المصفوفة :

1 - للاعلان عن مصفوفة عامة puplic array تستخدم جملة puplic في قسم الاعلان section declaration للوحدة module .

2- للاعلان عن module – level array تستخدم جملة private في قسم الاعلان الخاص procedure

3- للاعلان عن local array تستخدم جملة dim في ال procedure

1- في الحدث

```
Dim a, b As Integer
```

a و b متحولات عددية

2- في قسم التصريحات General:

```
'تعريف متحول عددي
Dim no1 As Integer

تعريف مصفوفة عددية سعتها عناصرها 10
Dim list(10) As Integer

'تعريف مصفوفة مؤلفة من 10 عناصرعرض و10 إرتفاع أي 100 عنصر
Dim matrix(10, 10) As Integer

'تعريف متغير نصي
Dim name1 As String

'تعريف متغير نصي بطول 50 حرف
Dim name1 As String * 50

'تعريف مصفوفة نصية تتكون من 10 عناصر
Dim list1(10) As String

'تعريف مصفوفة نصية تتألف من 10 عناصر بطول 50 حرف
Dim list1(10) As String * 50

'تعريف متغير للقيم نعم او لا
Dim a As Boolean
```

التصريح في الوحدة النمطية Module:

'تعريف متغير عددي لكافة نوافذ البرنامج
```
Global no1 As Integer
```

'تعريف عدة متغيرات في مجموعة واحدة
```
Type user
user_no As String * 70
user_name As String * 30
user_mark As Integer
End Type
```

الأن نأتي إلى كيفية إستخدام المتغيرات

طريقة الإضافة إلى المتغير هي

مثلاً لو صرحنا عن متحول عددي a نكتب كما يلي:

```
A = 90
```

أوعن طريق تعليمة الإسناد Let وتكتب كما يلي

```
Let a =90
```

أو يمكن إسناد قيمة عملية رياضية بين عددين مثل

```
Let a =5*6
```

لكي تظهر قيمة المتحول على الفورم أكتب التعليمة Print ثم إسم المتحول

```
Print a
```

ويمكنك عرض قيمة المتحول في صندوق نص Textbox

```
Text1 = a
```

يجب ان تشمل المصفوفة على:

A – اسم المصفوفة: الاسم الذي ستستعمله للمصفوفة في البرنامج .

B – نوع البيانات : يحدد نوع البيانات التي ستخزنها المصفوفة.

C – عدد الأبعاد : لمعظم المصفوفات بعد واحد (قائمة قيم) أو بعدين (جدول قيم) .

D – عدد العناصر : يحدد عدد عناصر المصفوفة وبشكل افتراضي تكون أول خانة في المصفوفة برقم فهرس ، (صفر) كما هو الحال مع مصفوفات التحكم .

تسمى المصفوفات التي تحتوي على عدد عناصر محدد بمصفوفات ثابتة الحجم Fixed Size أما المصفوفات التي تحتوي على عدد عناصر متغير فهي مصفوفات يمكن توسيعها خلال تشغيل البرنامج ويطلق عليها اسم Dynamic مصفوفات ديناميكية تصريح مصفوفة ثابتة .

التركيب النحوي الأساسي لتصريح مصفوفة ثابتة الحجم عامة هو Public Arry Name(Dim)

تغير جملة Option Base رقم أول عنصر في المصفوفة بجعله أي رقم بدلا من الرقم صفر الافتراضي فإذا أردت زيادة الوضوح في البرنامج بجعل رقم أول عنصر في المصفوفة هو العنصر رقم 1 بدلا من رقم 0 يمكنك وضع جملة Option Base التالية في وحدة نمطية قياسية Option Base 1 تربط هذه الجملة العنصر الأول (الأساس Base) لكل المصفوفات في البرنامج بالرقم 1

مثال استخدام المصفوفات

بعد تجهيزك للفورم أضف إليه زرين أمر Button Command
غير خاصية Caption لكل من الأزرار كما يلي:

Command2 = إدخال
Command2 = إظهار النتيجة

ولا تنسى تغيير خاصية الإتجاه للفورم Right To Left

وأضف وحدة نمطية Module
أكتب السطرين التاليين في الوحة النمطية

```
Option Base 1
Public Degs(9) As Variant
```

أضف الكود التالي إلى زر تخزين القيم

```
cls: prompt$ = "إدخال قيمة"
For i% = 1 To 3
Title$ = "القيمة" & i%
Degs(i%) = InputBox(prompt$, Title$)
Next i%
```

تمحو جملة Cls في أعلى الأجراء الحدثي إخراج الجمل Print السابقة على النموذج ليصبح بإمكانك إدخال اكثر من مجموعة قيم

ثم أضف الكود التالي إلى الزر الثاني

```
cls: Print "القيم": Print "----------"
For i% = 1 To 3
Print "المادة" & i% & "قيمة" & Degs(i%): Print "----------"

total! = total! + Degs(i%): Next i%
Print "مجموع القيم" & total!
```

multi – dimensional arrays المصفوفة متعددة الابعاد

يقصد بكلمة بُعد هو عدد الدلائل (indexes) المستخدمة للرجوع الى عنصر في المصفوفة . تستخدم المصفوفات متعددة الابعاد لتخزين بيانات مرتبطة ببعضها البعض .

مثال :-

تخزين كل نقطة pexel موجودة على شاشة الحاسوب نحتاج لتخزين احداثياتها X & Y وهذا ممكن باستخدام ذات بعدين .

يمكن معرفة ابعاد مصفوفة بالنظر الى تعريفها ، حيث يتم الاعلان عن مصفوفة متعددة الابعاد بذكر قيمة لابعادها بعد اسمها مباشرة وداخل الاقواس ، يتم فصل قيمة كل بعد بالفاصلة ، سواء كانت بعدين او ثلاثة ابعاد كما ستوضح الامثلة التالية .

```
Static a (3,4) as double
Static a (1 to 10 , 1 to 10) as integer
Dim b (3, 1 to 10 , 1 to 10)
```

ملاحظة :-

- عدد عناصر المصفوفة متعددة الابعاد يمكن معرفته بضرب قيم الابعاد ببعضها .
- عند اضافة ابعاد المصفوفة فان مساحة التخزين المطلوبة سوف تزيد زيادة كبيرة ولذلك ينبغي الاحتراس و تفادي استخدام النوع variant قدر الامكان لما يتطلبه من مساحة تخزينية كبيرة .

استخدام جمل الدوران مع المصفوفات :

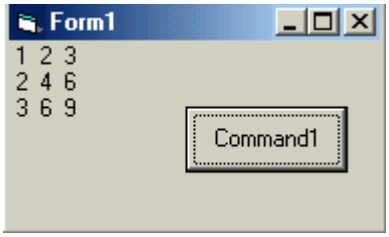
استخدام ال loops لمعالجة المصفوفات :

يمكن بكفاءة وسهولة معالجة المصفوفات باستخدام ال loops وخصوصا المصفوفات متعددة الابعاد حيث يتم معالجتها باستخدام loops متداخلة .

مثال :-

```
Dim I integer , j as integer
Static  A (1 To 10, 1 To 10) As Integer
For I = 1 To 10
For j = 1 To 10
A(i, j) = i * j
Next j
Next I
```

مثال :- اكتب الكود اللازم لشكل الفورم




```
Private Sub Command1_Click()
Dim a(1 To 3, 1 To 3) As Integer
Dim i As Integer, j As Integer
For i = 1 To 3
For j = 1 To 3
a(i, j) = i * j
Print a(i, j); "";
Next j
Print
Next I
End Sub
```

مثال :- ماهو حاصل الضرب للبرنامج التالي

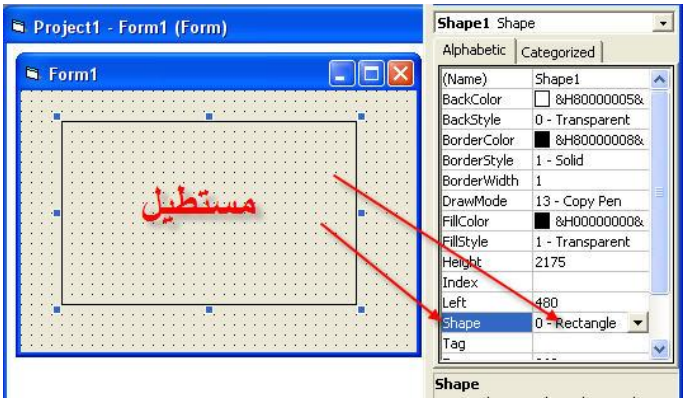
```
Option Explicit
Dim a(5) As Integer
Dim I As Integer
Private Sub Command1_Click()
For I = 0 To 5
 a(I) = I * 2
 Print a(I)
Next
End Sub
```

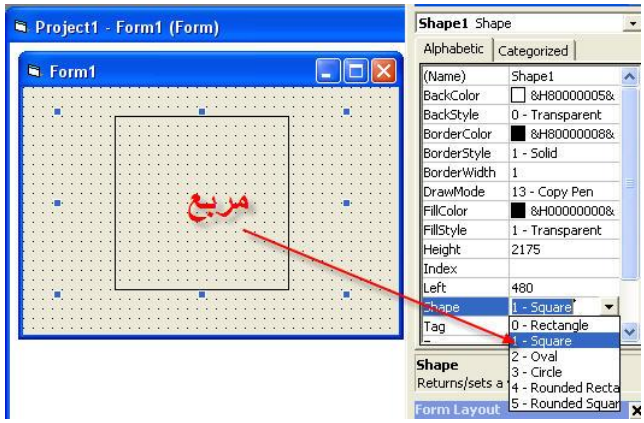
ويمكن قراءة المصفوفة باستخدام (input box)

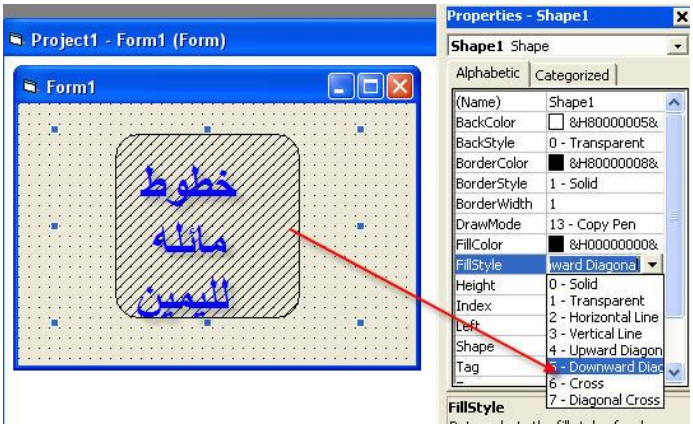
```
Dim a(5) As Integer
Dim I As Integer
Private Sub Command1_Click()
For I = 0 To 5
 a(I) = InputBox("enter value")
Print a(I)
Next
End Sub
```

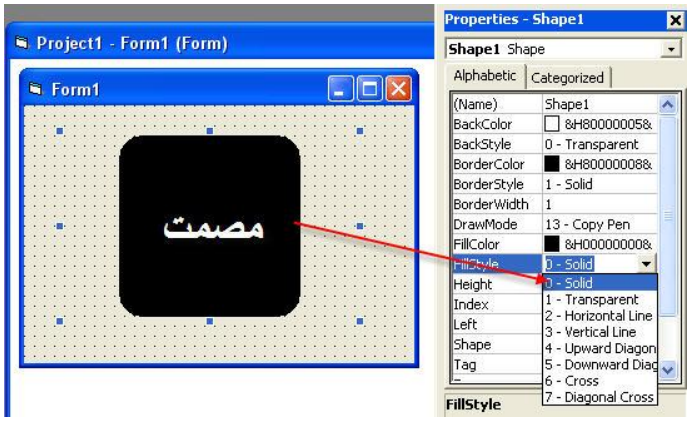
أداة الأشكال(Shape)

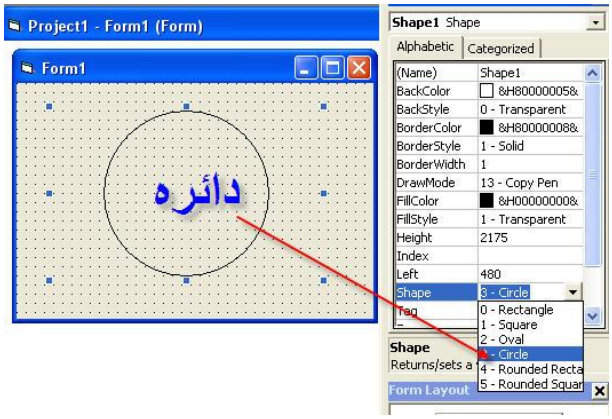
تستخدم لرسم شكل هندسي على الفورم للتجميل فقط وتأخذ أشكال هندسية عديدة منها:

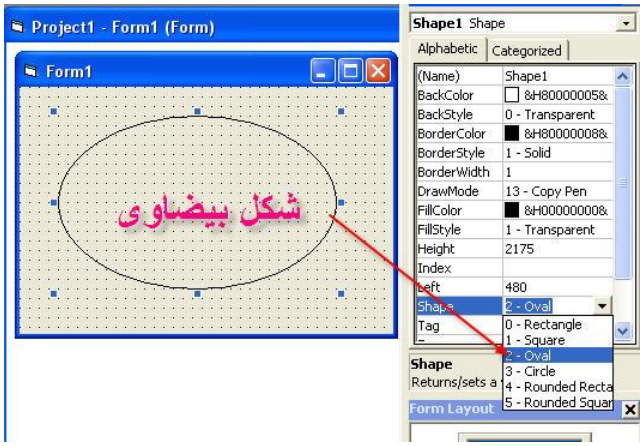


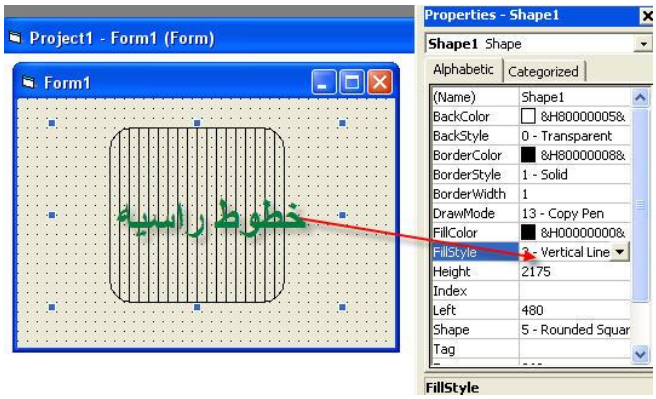


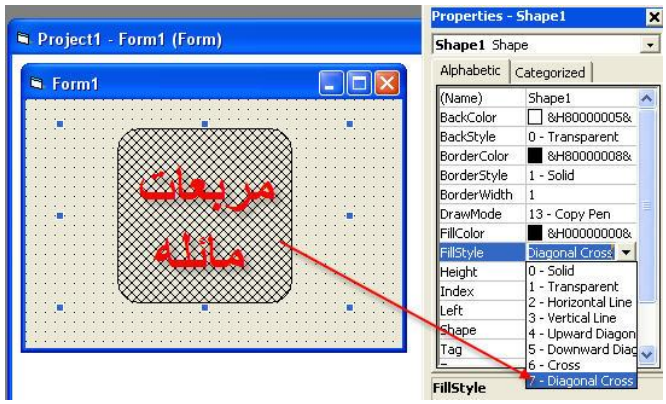












أداة رسم الخط (Line) تستخدم لرسم خط على الفورم وهى أيضا للتجميل فقط .

الطباعة باستخدام Print Object

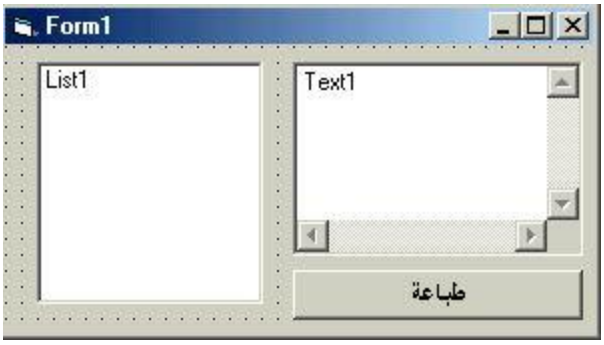
إن الطباعة من الأساسيات في برامجك فأي برنامج يحتاج إلى الطباعة
في هذا الدرس سوف نقدم مثالاً بسيطاً لطباعة نص دون إستخدام توابع API

أولاً يجب أن يكون لديك طابعة إفتراضية مثبتة ولو لم يكن لديك طابعة أضف طابعة وهمية للتجريب . عن طريق لوحة التحكم <- الطابعات <- إضافة طابعة

ثانياً عليك تجهيز الفورم بمايلي:

1 – أداة النصوص Text1
2 – الصندوق List1
3 – زر أمر Command1

هذه الأدوات التي عليك إضافتها



الأن نبدأ بالعمل :

1- أنقر نقرتين فوق الفورم وأكتب الشيفرة التالية في الحدث Load للفورم

```
Dim a As Integer
For a = 0 To Printers.Count -1
List1.AddItem Printers(a).DeviceName
Next
End Sub
```

شرح الكود :

هذا الكود يقوم بوضع قائمة بأسماء الطابعات المثبتة لديك في الصندوق List1 لتتمكن من إختيار الطابعة التي تريد الطباعة عليها.

2- أنقر نقرتين فوق List1 وأكتب الكود التالي :

```
Set Printer = Printers(List1.ListIndex)
```

هذا الكود يقوم بتحديد الطابعة من المجموعة المثبتة لديك

3- الأن نبدأ بعملية الطباعة :

أكتب النص الذي تريد طباعته في Text1
أضف الكود التالي إلى زر Command1:

```
Printer.FontSize = 40
Printer.Print Text1.text
Printer.EndDoc
```

في السطر الأول تحديد حجم الخط
في السطر الثاني إرسال النص إلى الطابعة
في السطر الثالث بدء الطباعة الفعلية

هذه عملية الطباعة البسيطة بإستخدام Print Object باستخدام توابع API .

لتعامل مع الحافظة

سوف تعرف كيف تنسخ النص وتلصق ماهو منسوخ من أي برنامج أخر وكذلك الصور.
فلو إلتقطت صورة عن طريق مفتاح Print Screen SysRa فإنك تستطيع لصق الصورة في برنامجك وحفظها .
طريقة العمل ليست صعبة . بعد تجهيز الفورم أضف العناصر التالية:

أداة النصوص Text1
5 أزرار Command غير Caption كما في الصورة
أداتين عرض الصور Picture1 و Picture2

ناتي إلى كتابة الشيفرة الخاصة بالعمل:

في الزر مسح الحافظة أضف السطر التالي

```
Clipboard.Clear
```

في الزر نسخ النص أضف السطر التالي:

```
Clipboard.SetText Text1.Text
```

في الزر لصق النص أضف السطر التالي:

```
Text1.Text = Clipboard.GetText
```

هذه الأسطر السابقة تقوم بنسخ كافة النصوص ولصقه

أما إذا أردت نسخ جزء محدد فإستبدل Text1.text بالكود Text1.SelText

نأتي الأن إلى نسخ ولصق الصور:

نفس الطريقة السابقة ولكن إستبدل Text1.text بإسم أداة الصور Picture1 أو Picture1

في الزر نسخ الصورة أضف السطر التالي:

```
Clipboard.SetData Picture1.Picture
```

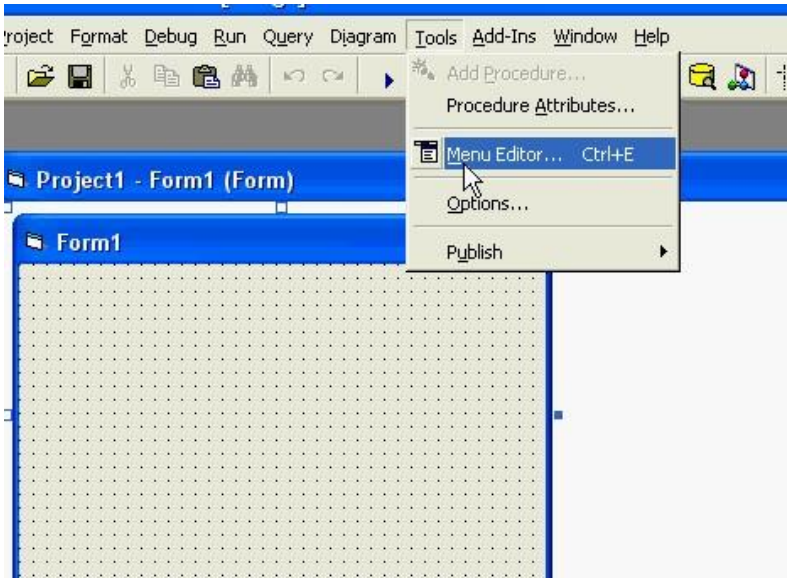
في الزر لصق الصورة أضف الكود التالي :

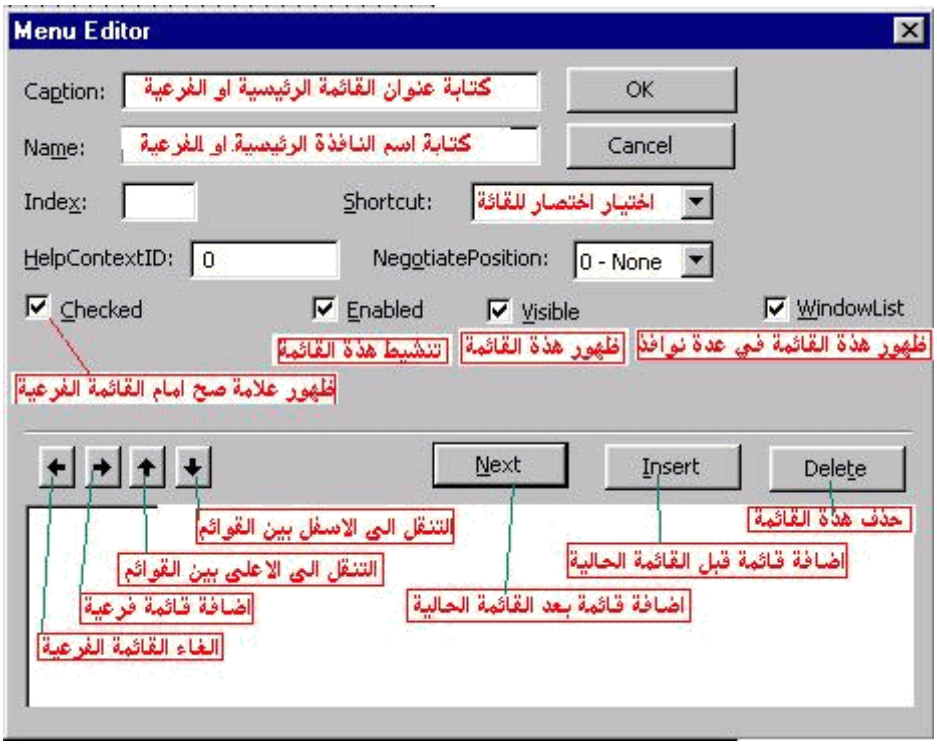
```
Picture2.Picture = Clipboard.GetData
```

محرر القوائم

إن أبسط برنامج تريد صنعه سوف يحتاج بالتأكيد إلى قوائم منسدلة مثلاً : ملف . تحرير . تعليمات .....
وهذا الدرس يساعدك على كيفية إنشاء القوائم:

لفتح محرر القوائم أنقر فوق الزر في شريط الأدوات أو من قائمة Tools إختر Menu Edior فيظهر لك محرر القوائم





الاساسيات المهمة:

1- caption منها بتكتب اسم القائمة الذي تريدها مثلا file اومن الممكن ان تكتب بالعربي ملف مثلا .

2- name وفيها تكتب اي رمز حرفي بالانجليزي المهم كل قائمة لها اسم خاص بها او حرف .

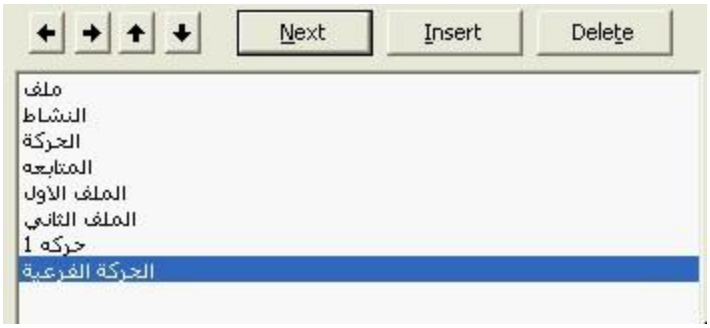
3- short cut ومنها بتختار اسم اختصاري من الكي بورد للقائمة بشرط ان لا تكون قائمة رئيسية مثل / file view/ ولكن المنسدل فقط

4- next ومنها تستطيع انشاء قائمة جديدة منها .

5- delete ومنها تقوم بحذف القائمة الغير مطلوبه .

6- وهي الاهم : عندك اربع اسهم في اتجاهات مختلفة اول سهمان اتجاههم فوق وتحت

و فائدتهم ترتيب القائمة حسب الاسبقية يعني مثلا انت تعمل قوائم كالتالي (ملف - النشاط - الحركة - المتابعه - الملف الاول - الملف الثاني – حركة1 - الحركه الفرعية)
وبعد كتابتهم نريد ترتيبهم عندما نضغط على القائمة(ملف)يظهر منها (الملف الاول) ومن الملف الاول (الملف الثاني) ونريد عمل قائمة بجانب (ملف) تكون القائم(النشاط) ونكتب بجانب القائمة (النشاط) قائمة اخرى وهي (الحركه) ومن القائمه الحركة(حركة ١ - الحركة الفرعية) . فأنك ستقوم بترتيب القوائم حسب ظهور القائمة ثم الذي يليها يعني مثلا الصورة الاتية تم عمل القوائم



ونريد ظهورها مثل ما نكتب يبقى الترتيب سيكون كالاتي : الذي نريده الاول وبعدها يتفرع منه الثاني كالتالي ولاحظ الفرق بين الترتيب الاتي والصورة السابقة:

1- الملف
2- الملف الاول
3- الملف الثاني
4- النشاط
5- الحركة
6- حركة1
7- الحركة الفرعيه

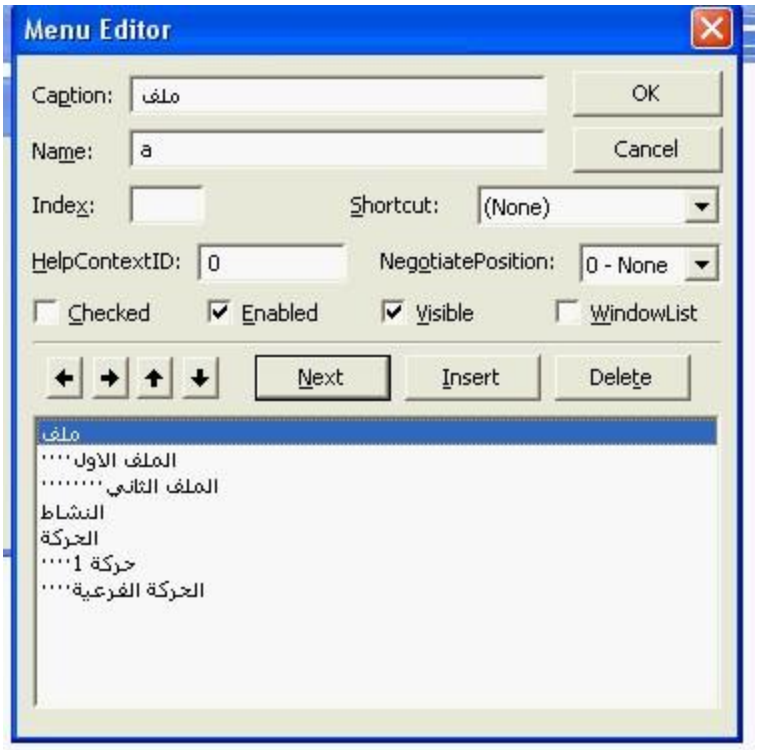
وهذه الترتيب بأستخدام السهمان فوق وتحت مع مرعات الوقوف بالماوس على القائمة المراد تحريكها
وجرب هذا ثم اضغط ok وبعدها شغل البرنامج ستلاحظ بان كل القوائم بجنب بعض ولاكن بقي السهمان الاخريين

للسهمان الاخران وهما مهمين جدا لقدعملنا في المثال السابق الاتي :

1- الملف
2- الملف الاول
3- الملف الثاني
4- النشاط

5- الحركة
6- حركة1
7- الحركة الفرعيه

نريد عندما نضغط على ملف يتفرع الملف الاول ثم على الملف الاول ونضغط عليها بالماوس وبعدها ننتقل الى السهم اليمين وضغط عليه مره واحده من القائمة ملف وننتقل الى الملف الثاني ونضغط على السهم اليمين مرتين يعني متفرع من الملف الفرعي الخارج من ملف وهكذا وسيكون شكلهم كالاتي:



من قائمة الحركة يظهر تحتهم حركة 1 وتحتها الحركة الفرعية
وجرب تشغيل البرنامج وعندما نريد عمل اسم مختصر للقائمة الملف الاول من shortcut تختار اسم لها ولكن لا تستطيع عمل اسم مختصر للقائمة الرئيسية مثل ملف .

وجرب تشغيل البرنامج سيكون شكلها كالتالي:

ربما سوف تتسائل كيف تظهر القائمة عند النقر بالزر الأيمن للماوس ! هذا بسيط :

فقط في مكان الحدث عند النقر بالزر الأيمن فوق الفورم مثلاً عل فرض أن الفورم إسمه Form1

إختر الحدث Mouse Down لل Form1 وأكتب السطر التالي

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

الرسائل Msgbox

الشكل العام

MsgBox "Message", Symbol, "Title"

Message: عبارة عن سلسلة حرفية (string)تمثل الرسالة التي نريدها تظهر في صندوق الرسائل

Title: عبارة عن سلسلة حرفية (string)تمثل عنوان الصندوق الذي يظهر في شريط العنوان للصندوق

Symbol: عبارة عن رمز يمثل قيمة صحيحة (integer)آو ثابتا حرفيا (constant) وتمثل القيمة الصحيحة آو الثابت الحرفي

هذا الجدول يعطيك أمثلة عن الرسائل

الشكل العام	التنفيذ
Msgbox "رسالتك هنا"	
MsgBox "حدث خطأ", vbCritical, "خطأ"	

MsgBox "رسالتك هنا", vbExclamation, "العنوان"

MsgBox "رسالتك هنا", vbInformation, "العنوان"

MsgBox "رسالتك هنا", vbMsgBoxHelpButton, "العنوان", "C:\WINDOWS\HELP\calc.CHM", 0

MsgBox "رسالتك هنا ", vbOKCancel, "العنوان"

MsgBox "رسالتك هنا ", vbYesNo + vbQuestion, "العنوان"

MsgBox "رسالتك هنا ", vbSystemModal, "العنوان"

MsgBox "رسالتك هنا ", vbAbortRetryIgnore, "العنوان"

الجدول التالي يبين الأيقونات وقيمها والثوابت الحرفية التي تمثلها

الاستخدام	شكل الايقونة	القيمة	الثابت
عند حدوث خطأ		16	vbcritical
عند طلب معلومات اضافية		32	vbquestion
عند حدوث خطأ والطلب من المستخدم التدخل		48	vbexclamation
لاخبار معلومة معينة		64	vbinformation

الجدول التالي يبين أزرار الأوامر وقيمها والثوابت الحرفية التي تمثلها

أنواع الـــCommands	القيمة	الثابت
Ok	0	Vbokonly
Ok, Cancel	1	Vbokcancel
Abort, Retry, Ignore	2	vbAbortRetryIgnor
Yes, No, Cancel	3	vbYesNoCancel
Yes, No	4	vbYesNo
Retry, Cancel	5	vbRetryCancel

ملاحظة:- تستطيع جمع ارقم عدة ادوات لعرضها في الرسالة مثال (65) هو عبارة عن جمع اداتي (Ok and Cancel) و (vbInformation) .

ملاحظة : كيفية وضع عدة أسطر تحت بعضها في الرسالة:

```
Dim a As String
a = "الأول السطر" & vbCrLf
a = a & "السطر الثاني" & vbCrLf
a = a & "السطر الثالث"
MsgBox a, vbMsgBoxRight + vbInformation, "العنوان"
'عدد السطور التي تريد بنفس الطريقة ويمكنك أن تضيف
      ' "&vbCrlf" السطر الأخير بدون
```

لاحظ المكتوب بالاحمر وهو نوع الرسالة وهو يعني إتجاه الرسالة من اليمين لليسار + تلميحات وهذا هام في الرسالة المكتوبة باللغة العربية

أي إنه يمكنك إضافة إشارة + بعد النوع الأول لإختيار نوع آخر مثلاً سؤال ("نعم ولا"+ "خطأ") هذه طريقة كتابتها:

```
MsgBox "الرسالة", vbCritical + vbYesNo, "عنوان"
```

وهذه طريقة أخرى

```
Dim a As String
a = "السطر الأول" & vbCrLf & "السطر الثاني" & vbCrLf & "السطر الثالث"
MsgBox a, vbMsgBoxRight + vbInformation , "العنوان"
```

صنايق الحوار Common Dialog

سوف نتحدث عن كيفية استخدام صناديق الحوار CommonDialog ربما لم تسمع بهذه الأداة سابقاً ولاكن سوف تعرفها:
عندما تفتح أي برنامج فإنك سوف تحفظ عملك في النهاية ! ماذا يظهر لك عندما تختار "حفظ" ؟
وإذ فتحت أي برنامج وتريد فتح ملف فسوف يظهر لك أيضا صندوق الحوار الخاص بالفتح وهو نفس صندوق الحفظ!
هذا الصندوق يسمى Common Dialog و هذه الأداة تستخدم في معظم البرامج المبرمجة بواسطة الفيجوال بيسك.

هناك طريقتان لإستخدام هذه الأداة :
1 - بإستخدام أداة جاهزة: Microsoft Common Dialog Control
2 - بإستخدام إجراءات API : وهذه عملية صعبة لا نتطرق اليها حاليا .
لذلك سيكون درسنا بإستخدام الأداة : Microsoft Common Dialog Control . طبعاً تعرف كيف تضعها على الفورم

كليك يمين فوق صندوق الأدوات إختر Components ثم تجد الأداة Microsoft Common Dialog Control 6،0 حددها وأنقر Apply

سوف تظهر الأيقونة على صندوق الأدوات . ضعها على الفورم . سوف يكون إسم الأداة CommonDialog1
ضع على الفورم الأداة Picture1 وزر Command1
ضع الكود التالي في الزر:

```
CommonDialog1.DialogTitle = "فتح ملف"
CommonDialog1.Filter = "Gif صور|*.gif|JPG صور|*.jpg|Bmp صور|*.bmp|الملفات كافة|*.*"
CommonDialog1.ShowOpen
If CommonDialog1.FileName = "" Then Exit Sub
Picture1.Picture = LoadPicture(CommonDialog1.FileName)
```

شرح الكود:

السطر الأول : تحديد العنوان الذي سوف يظهر في أعلى الصندوق
السطر الثاني : تحديد الملفات التي سوف يعرضها الصندوق
السطر الثالث : إظهار الصندوق
السطر الرابع : إذا لم يتم تحديد ملف ألغي الحدث

السطر الخامس: عرض الصورة التي قمت بتحديدها في الأداة Plcture1

كان هذا مثال بسيط لتمسك رأس الخيط . مازل هناك العديد من العمليات بواسطة CommonDialog. منها: تحديد الخط والتلوين و خصائص الطباعة و الحفظ .

الأن لنأخذ مثال عن إستعمال ألوان CommonDialog:

نفس الأدوات السابقة أضف زر Command2 وأكتب الكود التالي

```
CommonDialog1.ShowColor
If CommonDialog1.Color = 0 Then Exit Sub
Picture1.BackColor = CommonDialog1.Color
```

شرح الكود :

السطر الأول : إظهار ألوان الكمون دايلوك
السطر الثاني : إذا لم يتم تحديد لون ألغي الحدث

السطر الثالث : تلوين مربع الصورة باللون الذي تم تحديده CommonDialog:

سوف نتحدث الأن عن عملية فتح وحفظ مستند نص:

جهز الفورم بالأدوات التالية:
زرين أمر : فتح و حفظ
أداة عرض النص Label1
أداة الكتابة Text1 ولاتنسى تبديل الخصائص MultiLine و ScrollBars
وأهم شيىء الأداة CommonDialog

لاحظ الشكل التالي:

كتابة الكود:

أنقر دبل كليك على زر الفتح وأكتب الكود التالي:

```
Dim file As String
CommonDialog1.Filter = "النصوص ملفات|*.txt"
CommonDialog1.ShowOpen
Open CommonDialog1.FileName For Input As #1
Input #1, file
Close
Text1.Text = file
```

السطر الأول : التصريح عن متحول نصي
السطر الثاني : تحديد الملفات التي سيعرضها الكمون دايلوك
السطر الثالث : إظهار اللكمون دايلوك
السطر الرابع : والخامس والسادس فتح المستند كمتحول

السطر السابع : عرض مسار الملف في Label1
السطر الأخير : عرض النص في أداة الكتابة Text

نأتي إلى الحفظ :

ضع الكود التالي في زر الحفظ:

```
Dim a As String
a = Text1
CommonDialog1.Filter = "Text File|*.txt"
CommonDialog1.FileName = a
CommonDialog1.ShowSave
```

السطر الأول : التصريح عن متحول نصي (a)
السطر الثاني : إسناد النص المكتوب إلى المتحول (a)
السطر الثالث: تحديد قيمة الكمون دايلوك وهو المتحول(a)

السطر الرابع : عرض مربع الحفظ .

وبذلك نكون قد أعطينا فكرة عن إستعمال الكمون دايلوك . طبعاً هناك العديد من الأحداث لهذه الأداة سوف تمر معنا في الدروس القادمة.

Module

ماهو المديول Module

الموديول هى شاشه لكتابه الأكواد العامه والتصريحات

لماذا نستخدم المديول Module و ما الفرق بين تعريف متغير في المديول Module او في الفورم .

نستخدم الموديول عندما نحتاج لتعريف متغير او دالة او اجراء في جميع المشروع و لا يكون مقتصر على فورم واحد

عندما اريد تعريف المتغير X في الفورم اتعامل معه داخل الفورم فقط و عندما اعرفه في الموديول استطيع استخدامه في كل المشروع
باستخدام Public قبل المتغير يكون متاح للاستخدام في كل Forms بخلاف Private
وكذلك الحال بالنسبة للدوال Functions و الاجراءات Procedures

و الصيغة العامة للتعريف:

المتغيرات:

```
[Private | Public] VariableName as [Type]
```

الاجراءات:

```
[Private | Public] Sub Sub_Name ([Parameters])
الكود
End Sub
```

الدوال:

```
[Private | Public] Sub Sub_Name ([Parameters])
الكود
End Sub
```

ولتعرف كيف يتم ادراج Module جاهز في مشروعك تابع الصورة

Project1 (Project1)
Forms
Form1 (Form1)
6
Modules
انقر في أي مكان هنا بالزر الايمن
View Code
View Object
Project1 Properties...
Add
1
Save Form1
Save Form1 As...
Print...
Dockable
Hide
Form
MDI Form
Module
Class Module
User Control
Property Page
User Document
WebClass
Data Report
DHTML Page
Data Dynamics ActiveReports 2.0
More ActiveX Designers...
Add File...
2
False
&H8000000F&
2 - Sizable
Form1
True
True
13 - Copy Pen
0 - Solid
DrawWidth
1

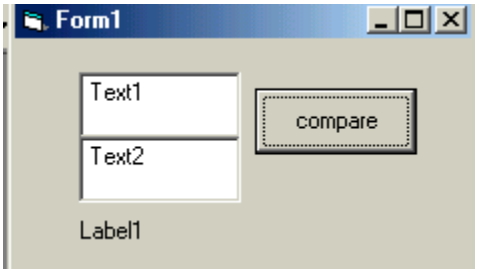
الدوال

1 - دالة المقارنة string compare

الشكل العام لهذه الدالة هي : Str Comp(String 1 , String 2):

مثال :- قارن بين " زيد " و " حسن "

```
Private Sub Command1_Click()
Dim res As Integer
res = StrComp(Text1.Text, Text2.Text)
If res = -1 Then
Label1.Caption = " first  < second "
ElseIf res = 1 Then
Label1.Caption = "first  > second"
Else
Label1.Caption = "first  = second "
End If
```

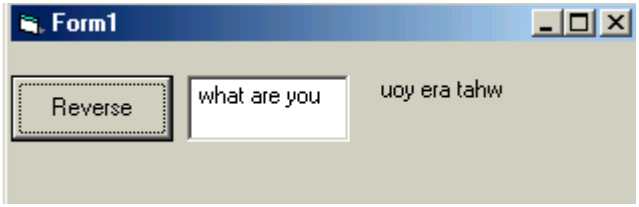


2 -الدالة العكسية Reverse:

الشكل العام لهذه الدالة هي : Str Reverse (String)

مثال:- جد عكس التصريح " what are you "

```
Private Sub Command1_Click()
Label1.Caption = StrReverse(Text1.Text)
End Sub
```



3- دالة الاستبدال Replace:

الشكل العام لهذه الدالة هي: Replace (String ,S1,S2)

مثال :- استبدل كلمة ،، hussen،، مع ،، ali،،

```
Private Sub Command1_Click()
Label1.Caption = Replace(Text1.Text, "hussen", "Ali")

End Sub
```

4- الدالة الاحرف الكبيرة والاحرف الصغيرة Ucase, Lcase :

الشكل العام لهذه الدالة: Ucase $(Str.Text) Lcase $(Str.Text)

مثال :-

```
Private Sub Command1_Click()
Text2.Text = UCase$(Text1.Text)
Text3.Text = LCase$(Text1.Text)
End Sub
```

5- دالة التحويل من نظام لاخر Convert:

الشكل العام لهذه الدالة هي : النظام السداسي عشر Hex$(text1.text)
النظام الثماني Oct$(text2.text)

مثال :-

```
Private Sub Command1_Click()
Text2.Text = Hex$(Text1.Text)
Text3.Text = Oct$(Text1.Text)
End Sub
```

6- دالة Val :-

تقوم بتحويل النصوص لرقم مثل :
لو كنت تريد مثلا أن تكتب في نص و ثم تدع البرنامج يضرب الرقم في 10
هذا هو المطلوب :

```
Label1.Caption = Val(Text1.Text) * 10
```

و تحتاج لإضافة أداتي Text ، Label و زر Command يتم تنفيذ بداخله الشفرة و نختار حدث مناسب ألا و هو الضغط على الزر Click .

7- الدالة Rnd :

- و هي دالة للإيجاد رقم عشوائي عشري و لكي لا يتكرر الرقم استخدم قبلها الدالة (Randomize).

```
Label1.Caption = Rnd
```

8- الدالة Int :

- تقوم بجعل الرقم العشري عدد صحيح مثال :

```
Label1.Caption = Int(1.95667899)
```

ستكون نتيجة التنفيذ هو الرقم 1 .

9- الدالة Str:- تقوم بتحويل الرقم إلى نص في الكود .

10- الدالة Round:- تقوم بتقريب العدد للمنزلة المطلوبة

```
Label1= Round(2.56789, 1)
```

و هذا معناه إننا نقرب الرقم 2.56842 إلى أقرب منزلة عشرية لانا و ضعنا 1 بعد الرقم .
و سيعطينا البرنامج الجواب و هو 2.6
و بدون الرقم نقرب لأقرب عدد طبيعي

مثال :-

```
Label1= Round(2.56789)
```

11- دالة len:- و تستخدم لمعرفة عدد أحرف جملة معينة أو حتى ارقام مثل التالي :

```
Label2.Caption = Len("حسن")
```

و ستكون النتيجة هي الرقم 3 لان عدد الأحرف 3 أرقام

12- دالة Trim :- و تفيد في انك إذا كانت لديك جملة نصية أو رقمية و احتمال كتب المستخدم في بدايتها مسافة أو حتى في نهايتها فبإمكانك محو المسافة من الجملة بهذه الدالة مثل :

```
Label1.Caption = Trim(Text1.Text)
```

13- دالة LTrim :- تمحو المسافات من بداية النص مثل التالي :

```
Label1.Caption = LTrim(Text1.Text)
```

14- دالة RTrim :- تمحو المسافات من نهاية النص مثل التالي :

```
Label1.Caption = RTrim(Text1.Text)
```

15- دالة Left :- تقوم بعرض عدد معين من الأرقام أو الأحرف حسب الطلب مثل المثال التالي تقوم بقص رقم واحد من بداية النص أو الرقم :

```
Label1.Caption = Left(Text1.Text, 1)
```

16- دالة Right :- تقوم بعرض عدد معين من الأرقام و الأحرف حسب الطلب مثل المثال التالي تقوم بقص رقم واحد من نهاية النص أو الرقم :

```
Label1.Caption = Right(Text1.Text, 1)
```

17- دالة Mod :- تقوم بعرض باقي القسمة مثل باقي قسمة 4 على 3 هو 1 فناتج العبارة سيكون 1 :

```
Label1.Caption = 4 Mod 3
```

18- دالة التحويل من النظام العشري الى النظام الثنائى :-

```
Private Sub Command1_Click()
MsgBox Dec2Bin(Text1.Text)
End Sub
Function Dec2Bin(ByVal n As Long) As String
Do Until n = 0
If (n Mod 2) Then Dec2Bin = "1" & Dec2Bin Else Dec2Bin = "0" & Dec2Bin
n = n \ 2
Loop
End Function
```

التعامل مع الأخطاء

إذا كنت تريد عمل أي برنامج بسيط فلابد أن يحصل في بعض الأحداث الأخطاء وهذاالخطأ سوف يتسبب في إغلاق البرنامج.

فدرسنا اليوم هو للتصريح عن الخطأ وتجاهله :

التعليمة القياسية لتجاهل الأخطاء هي :

```
On Error Resume Next
```

تكتب قبل الحدث وتقوم بإلغاء الحدث عند وقوع خطأ

وهناك طريقة أخرى وهي عمل حدث كرسالة مثلاً عند وقوع الخطأ

أولا عليك التصريح عن الخطأ كمايلي

```
On Error Goto R
```

(R) هي اسم الخطأ يمكنك تغييره لما تريد

بعد ذلك أكتب الحدث .... مثلاً Form2.Show اذا لم يكن الفورم 2 موجوداً سوف يتم اغلاق البرنامج

فعليك أن تكتب التعليمة بالشكل التالي

```
On Error GoTo R
Form2.Show
Exit Sub
R:
```

```
MsgBox "في البرنامج حدث خطأ", vbCritical, "غير موجود"
```

في السطر الأول التصريح عن الخطأ

في السطر الثاني الحدث الأساسي

في السطرالرابع والخامس يتم تحديد فعل المتحول

يمكنك تحديد أي حدث للمتحول بدلاً من الرسالة أو يمكنك أن تكتب السطر التالي :

```
MsgBox Err.Description
```

أي أن الرسالة ستكون من أصل الخطأ

إليك مثالاً مفيداً للتصريح عن الخطأ

اصنع Form أضف اليه أداة Drive1 "محركات الأقراص" وأداة Dir1 "صندوق المجلدات "

أنقر فوق Drive1 نقرتين وأضف الأسطر التالية:

```
On Error GoTo error1
Dir1.Path = Drive1.Drive
Exit Sub
error1:
MsgBox "الرجاء إدخال قرص في محرك الأقراص هذا", vbCritical, "خطأ"
```

السطر الأول التصريح عن الخطأ

السطر الثاني الحدث الأساسي "عرض المجلدات داخل محرك الأقراص المحدد"

الأسطر الأخيرة عمل المتحول وهو عرض رسالة خطأ

ويمكنك الإستغناء عن الرسالة السابقة بالرسالة الأساسية للخطأ بالتعليمة التالية:

```
MsgBox Err.Description
```

هذا مجرد شرح بسيط عن الأخطاء لكي يكون برنامجك نظيفاً خالياً من أي خطأ Error

تشغيل الملفات

هناك عدة طرق لتشغيل الملفات في الويندوز من داخل الفيجوال بيسك.
وقد وضعنا في هذا الدرس أهم الطرق لتشغيل الملفات التنفيذية والملفات الأخرى. لعنا نقدم الفائدة لكل مبرمج في الفيجوال بيسك.

نبدأ بتشغيل الملفات التنفيذية (البرامج EXE)

1 – بإستخدام الدالة Shell :

هذه الدالة تستخدمها الفيجوال بيسك لتشغيل الملفات التنفيذية .

صيغتها العامة كما يلي.

```
Shell "File Name Path",
WindowsStyle
```

File Name Path : مسار الملف التنفيذي المراد تشغيله.

WindowsStyle : طريقة عرض نافذة البرامج المراد تشغيله . وهذا الجدول يبين الثوابت لهذا الوسيط:

الثابت	قيمته	شرحه
vbHide	0	تكون النافذة مخفية
vbNormalFocus	1	تكون النافذة في حجمها الطبيعي مع إعطاء التركيز لها
vbMinimizedFocus	2	تكون النافذة في وضع التصغير على شريط المهام وإعطائها التركيز
vbMaximizedFocuse	3	تكون النافذة في وضع التكبير مع إعطائها التركيز
vbNormalNoFocus	4	تكون النافذة في وضع التصغير دون إعطائها التركيز
vbMinimizedNoFocus	5	تكون النافذة في وضع التصغير دون إعطائها التركيز

مثال : –

```
Shell "notepad.exe",
vbNormalFocus
الوضع العادي تشغيل المفكرة في
```

لاحظ انه لم نكتب المسار الكامل للمفكرة لأنها داخل مجلد النظام .

العبارة الشرطية IF

1 – العبارة: IF - Then

تستخدم عبارة الشرط If في شروط أي أنه إذا تحقق الشرط عند ذلك إعمل الحدث وإلا أي إذا لم يتحقق ألغي الحدث أو إعمل حدث أخر.

التعليمة البسيطة لعبارة If هي كالتالي:

IF	Condition	Then	Statement
إذا	الشرط	عندئذ	تعليمة

مثال :- اكتب برنامج لادخال درجة طالب ثم طباعة كلمة pass اذا كان الطالب ناجح

في هذه التعليمة إذا تحقق الشرط تنفذ كل اتعليمات التي بعد كلمة Then وإذا لم ينفذ الشرط تنفذ التعليمات الواردة على السطر التالي لعبارة الشرط.

```
Dim x as integer
Private sub command1_click()
X= cint(text1.text)
If x>= 50 then text2.text= "pass"
End sub
```

2 – العبارة IF – Then - Else

وهو عندما يتحقق الشرط نفذ تعليمة ما وإذا لم يتحقق فهناك تعليمة أخرى سوف تنفذ في حال عدم تحقق الشرط.

الشكل العام للتعليمة:

IF	Condition	Then	Statement 1	Else	Statement 2
إذا	الشرط	عندئذ	تعليمة	وإلا	تعليمة عدم تحقق الشرط

مثال:- اكتب برنامج لادخال درجة طالب ثم طباعة كلمة ناجح اذا كان درجته اكبر او يساوي 50 و طباعة كلمة راسب خلاف ذلك .

```
Dim x As Integer
Private Sub command1_click()
x = CInt(Text1.Text)
If x >= 50 Then
Text2.Text = "pass"
Else
```

```
Text2.Text = "fail"
End If
End Sub
```

اختبار الجمل المنطقية :

- (=) ممكن اختبار القيم عن طريق معامل (=) إذا أردت تحقق شرطين متماثلين مثل :

```
If a = 0 Then b = 0
```

- (<>) و معناها لا تساوي مثل :

```
If a <> 0 Then b = 0
```

- (>=) و معناها أكبر أو تساوي مثل :

```
If a >= 0 Then b = 0
```

- (<=) و معناها أصغر أو تساوي مثل :

```
If a <= 0 Then b = 0
```

- (or) (و معناها أو) إذا أردت تختبر تحقق أحد الشرطين أو أكثر أربط الجمل الجملة (or) مثل :

```
If a = 0 Or c = 1 Then b = 0
```

- (and) إذا وجب تحقق عدة شروط في آن واحد استخدم أداة الربط و (and) مثل :

```
If a = 0 And c = 1 Then b = 0
```

- (Xor) (معناها أو إذا) إذا وجب تحقق أحد الشرطين في آن واحد و ليس كلاهما استخدم أداة الربط (Xor) مثل :

```
If a = 0 Xor c = 1 Then b = 0
```

- (Eqv) إذا وجب تحقق الشرطين معا في حال الصحة معا و في حال الخطأ معا استخدم أداة الربط (Eqv) مثل :

```
If a = 0 Eqv c = 0 Then b = 0
```

- (Not) متى ما إذا لم يحقق الشرط فالعبارة تكون صحيحة استخدم أداة الربط (Not) مثل :

```
If Not a = 0 Then b = 0
```

مثال :-

كود:

```
Private Sub Command1_Click()
Cls
Dim x As Integer
For x = 0 To 20 Step 1
```

```
Print x
Next x
End Sub
```

شرح الكود

سوف يقوم الحاسبة بتنفيذ الاكواد بلتسلسل

حجزنا متغير نوعه Integer .

في السطر الثاني عملنا حلقت تكرار كم 0 الى 20 اي عداد يبدء من 0 و ينتهي ب 20

السطر الثالث يقوم بطباعة القيمة x

السطر الرابع يقوم بلقفز الى القيمة for ويقوم بلعد مرة اخرى الى ان يصل الى 20 ثم يطبعه وينهي البرنامج .

العبارة Select Case

جملة Select تشبه تقريباً جملة If إلا أنها تستعمل حالات متعددة وفي كل حالة تكتب تعليمة مختلفة.

الشكل العام للعبارة:

ونستخدمها عندما نريد ان ننفذ مجموعة من الجمل والتي تعتمد على قيمة متغير واحد وتنفذ بالشكل

```
Select Case (العنصر)
Case (الحالة)
التعليمة
End select
```

يمكنك إضافة الكثير من الحالات ولاكن قبل End Select

سوف نعمل مثال بسيط لهذه الجملة.

أضف فورم وأضف الأدوات التالية:

Text1

Label1

Command1

أنقر فوق الزر نقرتين وأكتب الشيفرة التالية:

```
On Error GoTo b

Dim a As Integer
a = Text1
Select Case a
Case 0 To 10
Label1 = "سيىء"
Case 10 To 40
Label1 = "ضعيف"
Case 40 To 60
Label1 = "وسط"
Case 60 To 80
Label1 = "جيد"
Case 80 To 100
Label1 = "ممتاز"
End Select

Exit Sub
b:
MsgBox "الرجاء إدخال قيمة عددية ", vbCritical, "الكتابة خطأ في"
```

شرح الشيفرة السابقة:

السطر الأول : التصريح عن خطأ أسميناه (b)
السطر الثاني: التصريح عن متحول عددي صحيح (a)
السطر الثالث: إسناد النص في Text1 إلى المتحول (a)
السطر الرابع : إلى End Select بداية التعليمة Select . أي حدد حالات المتحول a في كل حالة عرض تعليمة مختلفة
السطور التي بعد End Select هي عمل الخطأ أي عرض رسالة عند الخطأ . حاول كتابة نص عادي فسوف تظهر لك الرسالة.
لا حظ أنه أنهينا العبارة ب End Select مثل End If
ويمكنك إستعمال العبارة الشرطية IF ولاكن عليك كتابة الجملة كاملة في كل حالة من الحالات السابقة.

حلقة التكرار Do

تستعمل جملة التكرار Do لعمل حدث ما بعدد معين .
يجب أن تصرح عن متحول قبلها لكي تعمل الجملة:

```
Dim a As Integer
Do
a=a+1
التعليمة
Loob Until (a Value)
```

ربما شرح الجملة غير مفهوم قليلاً . في السطر الأول صرحنا عن متحول عددي

السطر الثاني بداية جملة التكرار
السطر الثالث زيادة قيمة المتحول a في كل حلقة تكرار
السطر الرابع تكتب التعليمة التي تريدها
السطر الخامس هو عدد مرات التكرار وهو سوف يكون قيمة المتحول a مثلاً : 6>a أو 5=a
ذلك سوف يتم عمل التعليمة التي كتبتها ٥ مرات ولهذا السبب قلنا عليك التصريح عن متحول لأنه دون ذلك سوف تكرر التعليمة إلى مالانهاية.

سوف نعمل مثال للتعرف أكثر على الجملة:

مثال :- اكتب برنامج لطباعة كلمة hello خمسة مرات مع تسلسل كل كلمة مستخدماً do while

```
Dim i as integer
Private Sub Command1_Click ()
i = 1
Do while i <= 5
Print "hello"; i
i = i + 1
Loop
  End Sub
```

حلقة التكرار For

حلقة التكرار For مشابهة تقريباً للحلقة السابقة Do
تستعمل لتكرار عمل معين بعدد محدد.
الشكل العام للتعليمة:

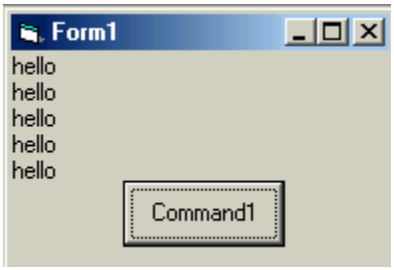
```
For a= القيمة الأولى To القيمة الثانية
الحدث (التعليمة)
Next
```

ملاحظة : - (a) مجرد وسيط يمكنك تغيره

مثال سريع للجملة البسيطة :

اكتب برنامج لطباعة كلمة hello خمسة مرات .

```
Dim i as integer
Private Sub Command1_Click ()
For i = 1 To 5
Print "hello"
Next i
  End SuB
```



جمل التكرار المتداخلة nested for

يمكن ان تكون الجمل التابعة لجمل التكرار الاولى هي جمل تكرار اخرى، في هذه الحالة يجب انهاء الحلقات الداخلية اولاً ثم الحلقات الخارجية بالتتابع كما يلي.

```
For i = 1 To n
 For j = 1 To m
  Statements
Next j

Next i
```

مثال :- اكتب برنامج لطباعة جدول الضرب

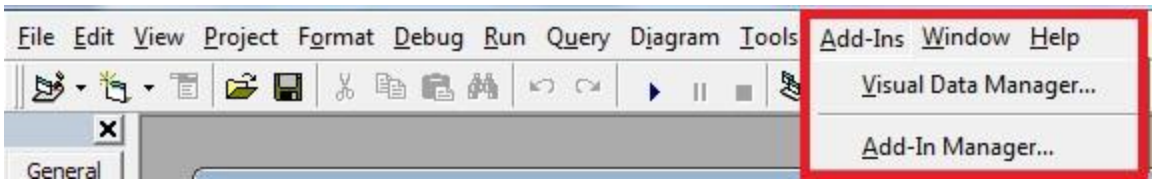
```
Dim I, j As Integer
Private Sub command1_click()
For I = 1 To 10
For j = 1 To 10
p = I * j
Print I; "*"; j; "="; p,
Next j
Print
Next I

End Sub
```

انشاء قواعد البيانات بـــ Visual Data Manager

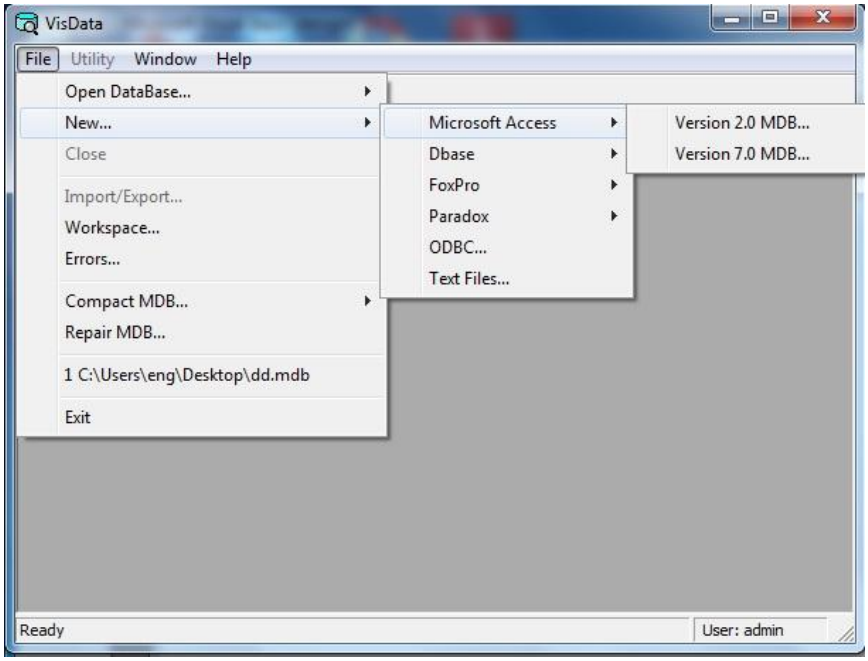
كما هو ممكن انشاء قاعدة البيانات داخل برنامج الـــ Access نفسه فإنه من الممكن ايضاً انشاء أي نوع من انواع قواعد البيانات داخل بيئة

خاصة داخل VB تسمى Visual Data Manager، و ذلك كما يلي :

1 - الخطوة الأولى فتح الفيجوال بيسك كالمعتاد و من ثم الذهاب إلى قائمة Add-ins → visual data manager...

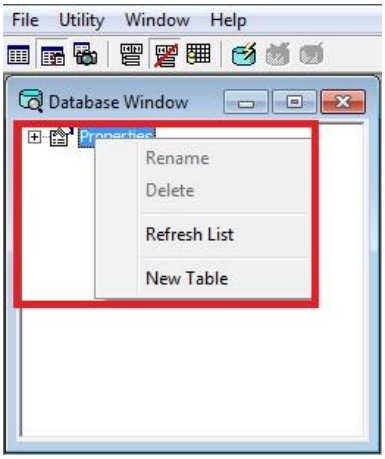


2 - بعد ذلك ستفتح لك نافذة البرنامج اذهب إلى File → new → Microsoft access → version 7.0 ..mdb

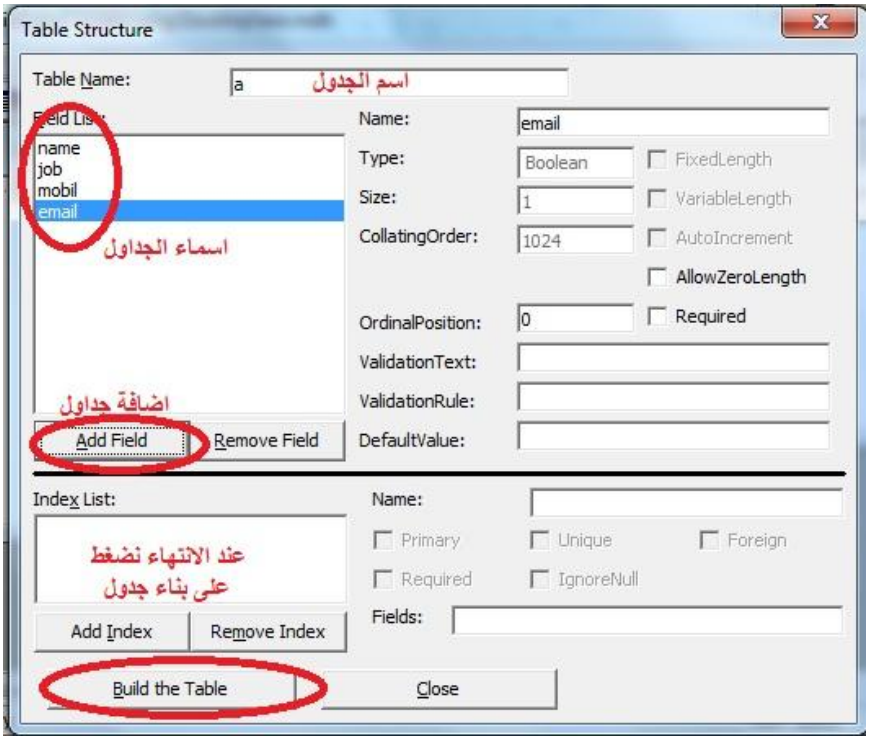


سيظهر لك مربع حفظ لتحدد الموقع الذي تريد أن تحفظ فيه قاعدة بياناتك بعدها اختر حفظ .

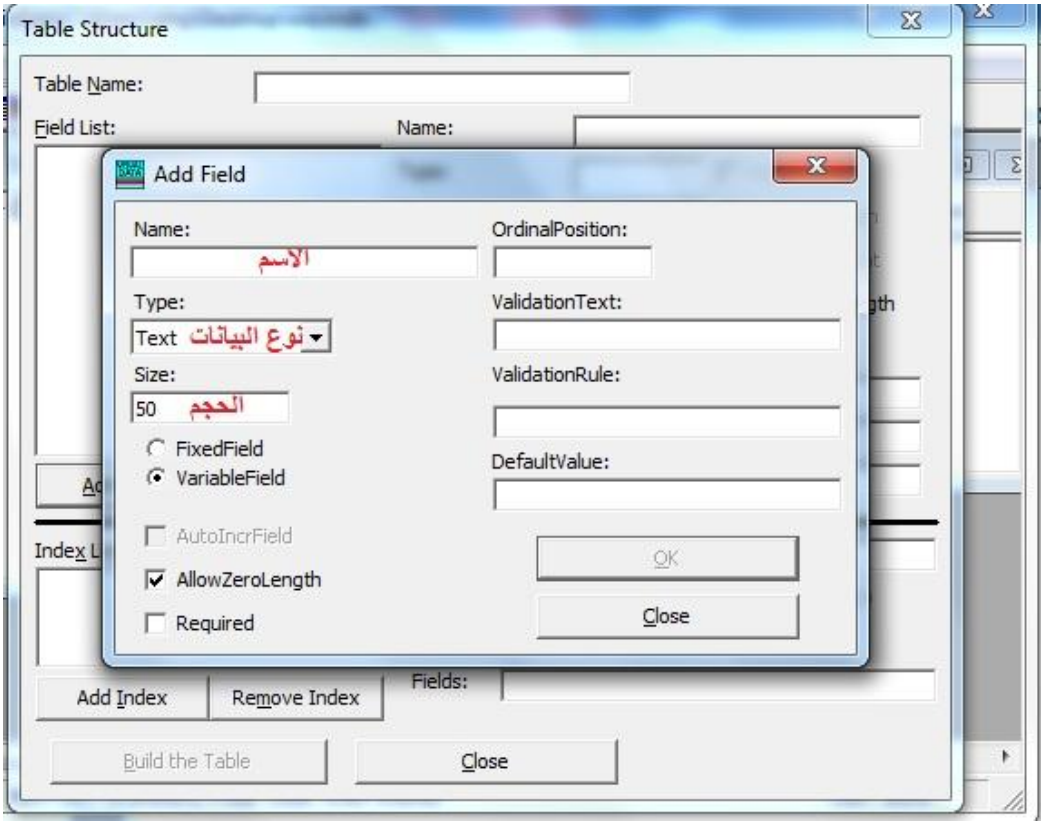
3 – بعد ذلك ستظهر لك قاعدة البيانات ، حدد الخصائص Properties بالزر الأيمن ثم أختر New Table من اجل انشاء جدول .



4 - بعد ذلك ستفتح لك نافذة تكتب فيها أسم الجدول و ثم أضف حقول الجدول الحقول كما هو مبين .



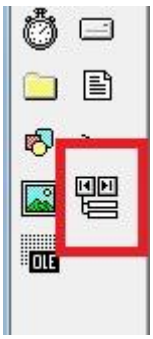
5 – يتفتح لك نافذة إضافة الحقول ، اكتب أسم الحقل في المكان المخصص و حدد نوع بيانات الحقل أي هل هي رقمية أو حرفية ... الخ ، ثم كرر العملية حتى تنهي جميع الحقول التي تريدها .



تقنيات الوصول الى قواعد البيانات

- التقنية (DAO) *Data Access Objects* : توفر هذه التقنية مجموعة من الكائنات (Objects) للتخاطب مع قاعدة البيانات، وإرسال الأوامر لها. وهي مصممة بشكل رئيس للتعامل مع قواعد بيانات أكسس (الى الاصدارة 97) ، و يمكنك استخدامها في التعامل مع قواعد بيانات أخرى من خلال المحرك Jet الذي توفره.

- اداة التحكم في البيانات Data Control : و هي تقنية سهلة للتعامل مع قاعدة البيانات و هي احدى ادوات التحكم الموجودة في صندوق الأدوات القياسي ، و هي تمكننا من الاتصال بمجموعة من السجلات Recordset في قاعدة بيانات Jet للتعامل معها و كذلك ربطها بعدة ادوات في فيجوال بيسك مثل اداة Textbox..

- 

- تقنية Remote Data Objects (RDO) : و طوّرت هذه التقنية بشكل رئيس لتوفر طبقة مُبَسِطة لتقنية ODBC لمبرمجي لغة فيجوال بيسك.

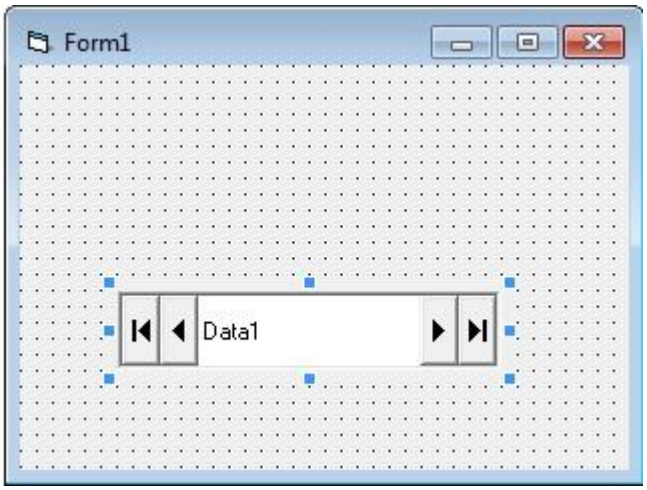
- تقنية ActiveX Data Objects (ADO) : واستكمالاً لمسيرة التقدم والتطوير طرحت شركة مايكروسوفت بعدها نظام ADO، والتي قامت بتسهيل عملية التعامل مع قواعد البيانات بشكل كبير، وأصبحت الشكل النهائي للوصول لقاعدة البيانات ليس في لغة VB و حسب ، بل و في العديد من لغات البرمجة .

الربط بالأداة Data Control

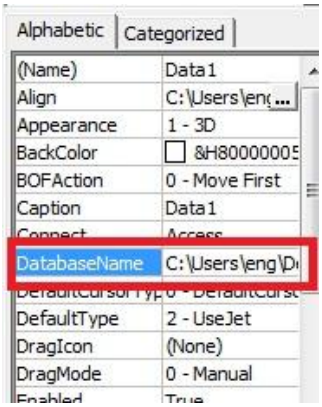
- قبل عملية الربط يجب علينا اولاً تصميم قاعدة البيانات ، و يجب ان تكون قاعدة البيانات من نوع Access 97 على اقصى تقدير .

- خطوات الربط ستكون على النحو الآتي :

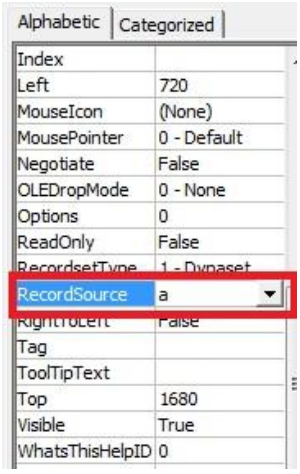
1- يتم انزال الاداة Data Control من صندوق الادوات الى الواجهة Form ، و سيعطى لها الاسم Data1 .



2- من خلال خصائص الاداة Data Control يتم تحديد قاعدة البيانات المراد الاتصال بها عن طريق الخاصية DatabaseName .

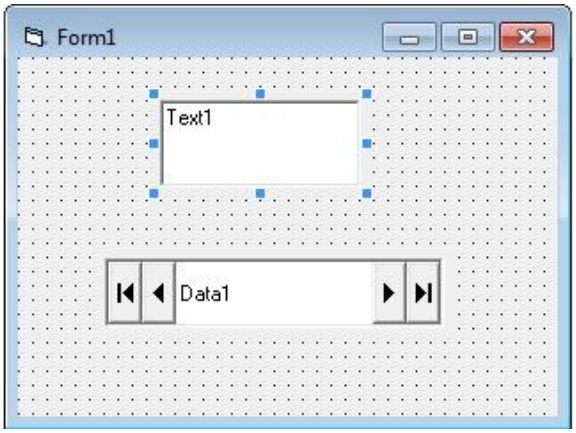


3- بعد ذلك يتم تحديد الجدول المراد الاتصال به عن طريق الخاصية RecordSourse و التي معناها السجلات التي اود استرجاعها من قاعدة البيانات ..

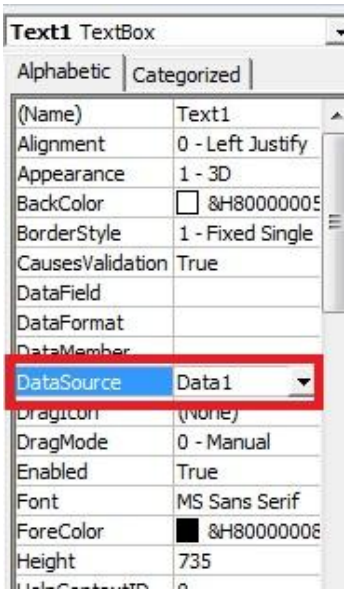


ربط مربعات النص Textboxes بحقول قاعدة البيانات

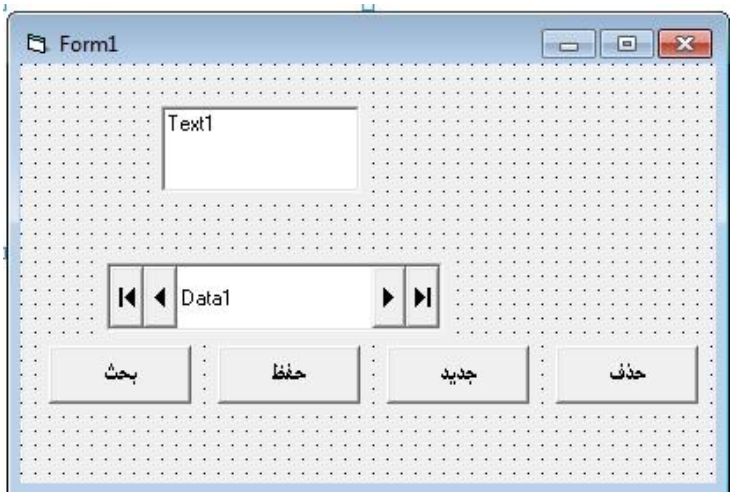
يتم ربط قاعدة البيانات بمربعات النص Textbox عن طريق الاداة Data Control من اجل عرض حقول البيانات فيها ، و يتم ذلك بإتباع الخطوات التالية :



1 - اختر الخاصية DataSource من خصائص مربع النص المراد عرض حقل من البيانات من خلاله و ضع فيه اسم الـ Data Control التي تصلنا بقاعدة البيانات (و التي سيكون اسمها في هذا السياق Data1) .



نقوم باضافة الازرار الثلاثة ونكتب الكود الخاص لكل زر (بحث ، حفظ ، جديد) كما في الشكل .



1- انشاء سجل جديد في قاعدة البيانات

```
Data1.recordset.Addnew
```

2- حفظ البيانات في قاعدة البيانات.

```
Data1.recordset.update
```

3- حذف سجل من قاعدة البانات

```
Data1.recordset.delete
Data1.refresh
```

4- البحث عن سجل

- اول سجل يطابق المطلوب من البحث

```
Data1.Recordset.FindFirst"name='mohmmed'"
```

- اخر سجل يطابق المطلوب من البحث

Data1.Recordset.FindLast

- السجل التالي الذي يطابق المطلوب من البحث

Data1.Recordset.FindNext

- السجل السابق الذي يطابق المطلوب من البحث

Data1.Recordset.FindPrevious

- الذهاب الي السجل الاول

data1.Recordset.MoveFirst

- الذهاب الي السجل الاخير

data1.Recordset.MoveLast

- الذهاب الي السجل التالي

data1.Recordset.MoveNext

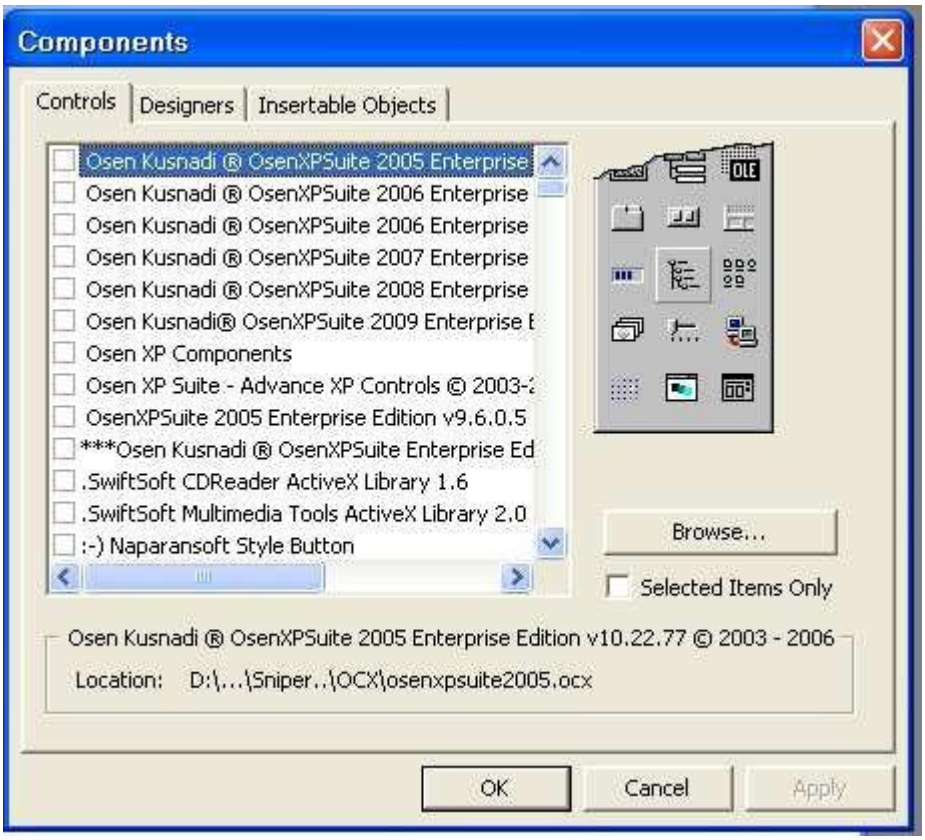
طريقة إضافة وتبديل ال skin

يستخدم اداة ال skin لاعطاء مشروعك رونقا و جمالا بدل الشكل القديم للبرنامج

1 - طريقة اضافة ال skin :

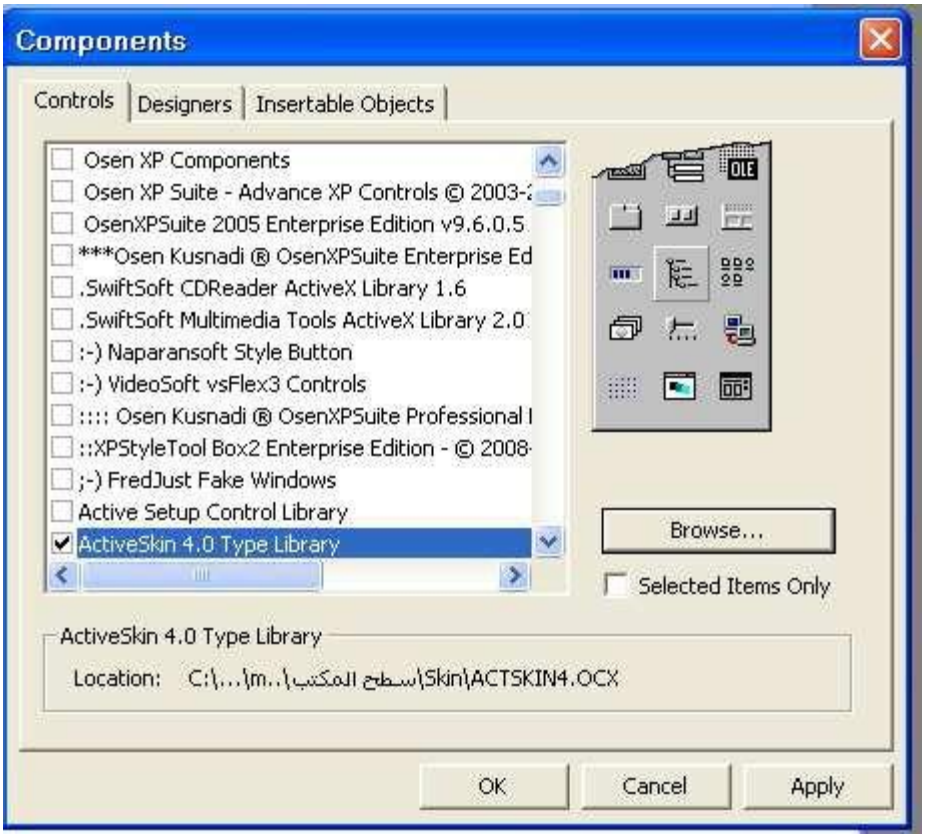
اضغط click بزر الفارة الايمن على ال tool box كما في الصورة التالية وقم باختيار components

ستظهر شاشة كما في الصورة

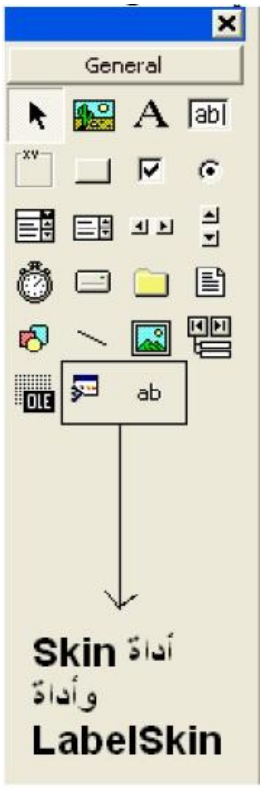


اضغط على Browse لاضافة اداة ال skin كما في الصورة التالية :

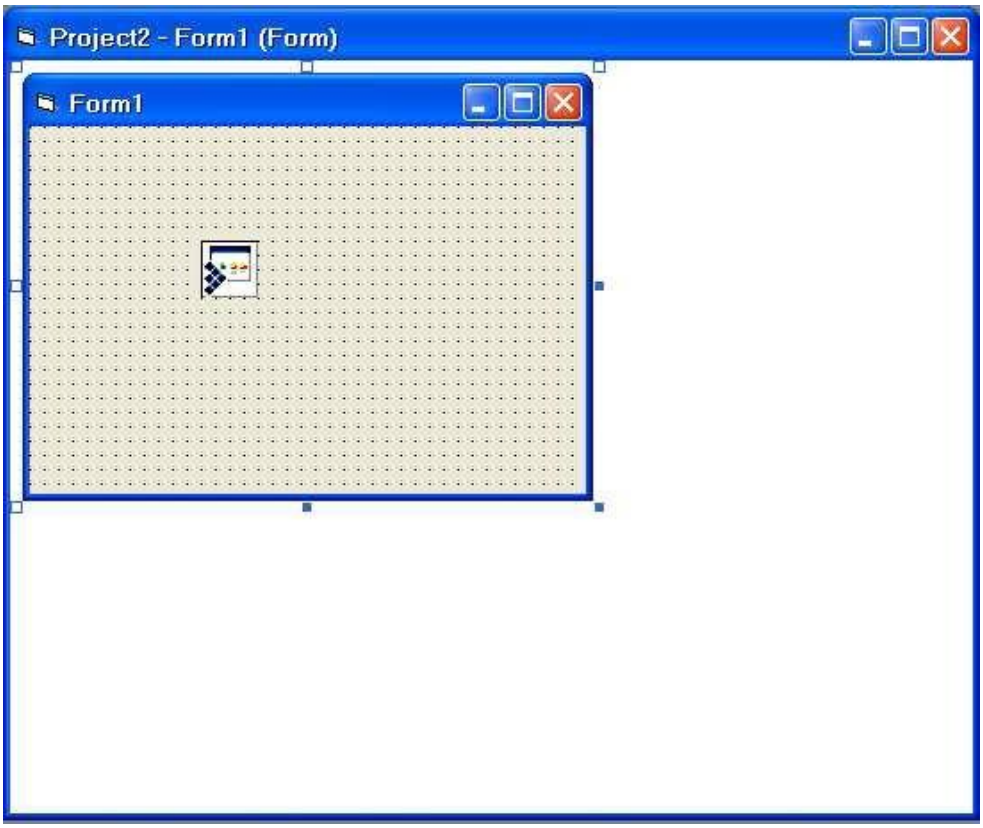
ملاحظة :- اداة ACTSKIN4 . OCX هو اداة يتم تحميله من الانترنيت الى الحاسبة وادراجه الى البرنامج من خلال الخطوات التالية .



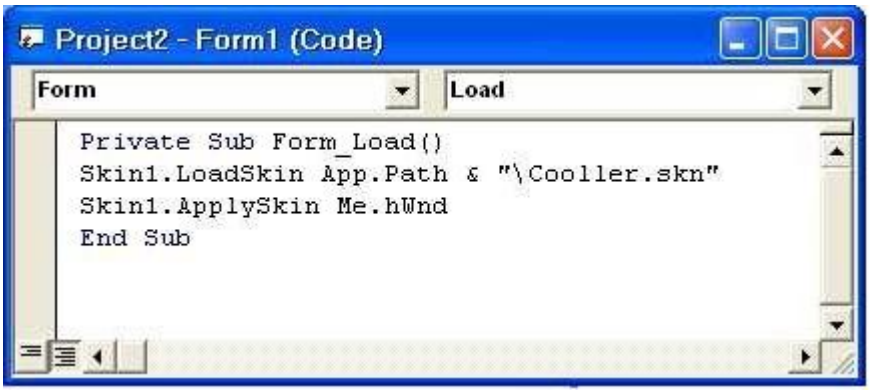
ستظهر اداة ال SKIN واداة ال label skin في صندوق ال toolbox كما في الصورة التالية :



قم باضافة الاداة الى مشروعك كما في الصورة التالية



افتح نافذة كتابة الكود كما في الصورة التالية و اكتب الكود كما موضح



ملاحظة مهمة : - Cooller.skn هو اسم ال skin الذي تريد اضافته الى المشروع .

شرح الكود السابق

```
Skin1.LoadSkin App.Path & "\Cooller.skn"
```

هذا الكود لتحديد مسار ال skin

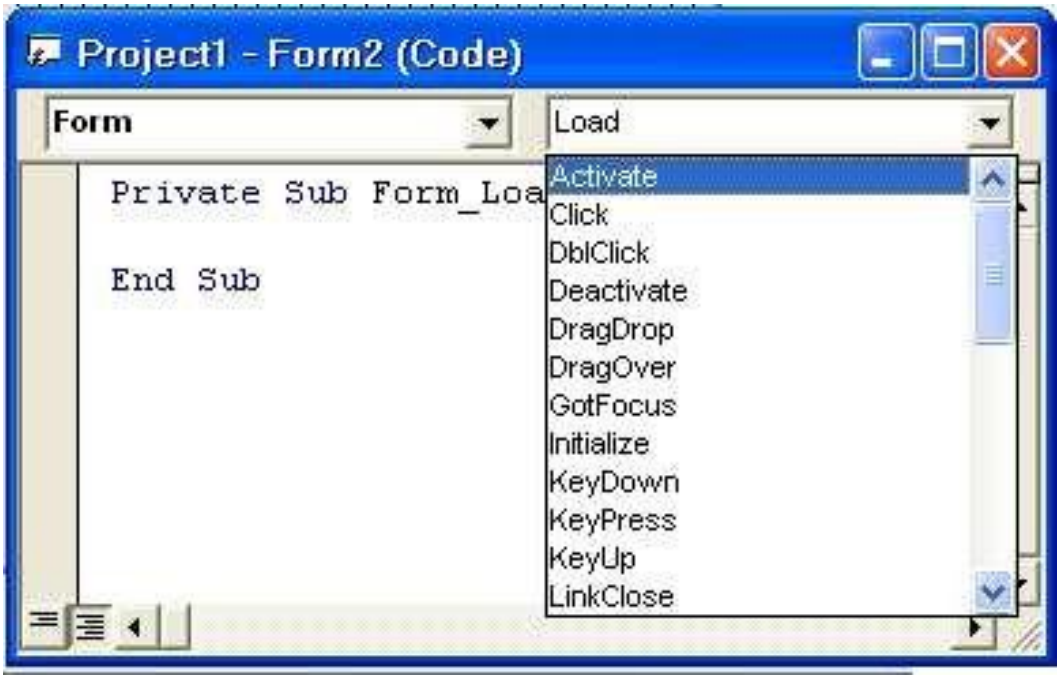
```
Skin1.ApplySkin Me.hWnd
```

هذا الكود لتشغيل اداة ال skin

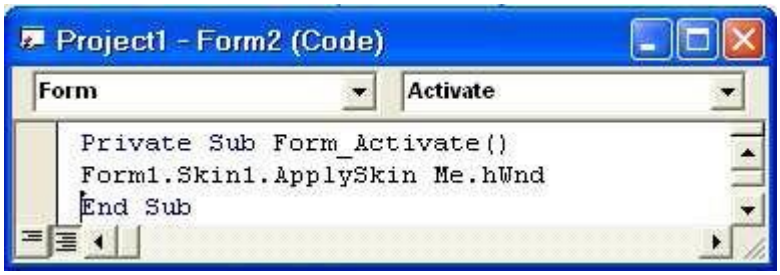
قم بتشغيب البرنامج سيظهر لك الفورم كما في الصورة التالية .

طريقة تطبيق ال skin على جميع نوافذ المشروع من خلال اداة ال skin الموجودة في الفورة الاول .

نقوم بفتح فورم جديد كما في الصورة التالية ونفتح نافذة الكود للفورم و نختار حدث Activate كما في موضح

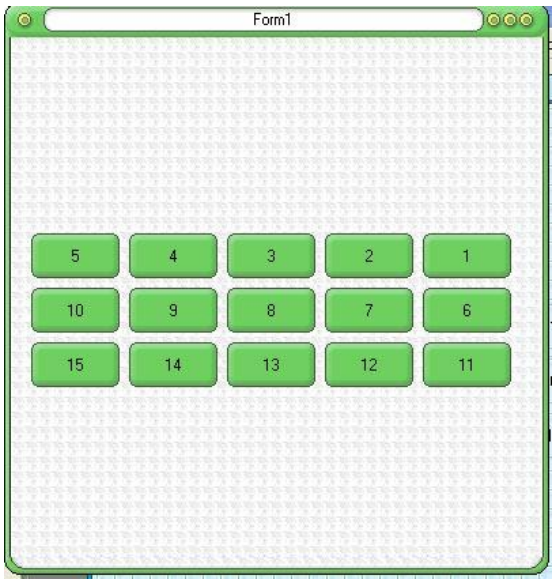


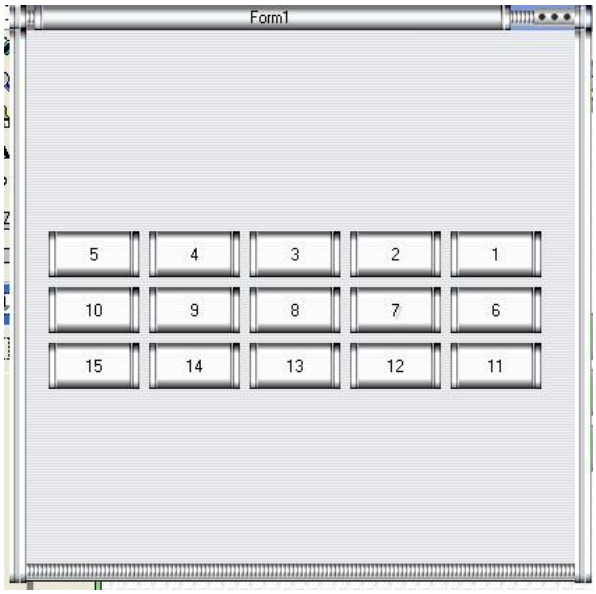
نقوم بكتابة الكود اللازم لتفعيل ال skin على الفورم الثاني كما موضح في الصورة



ملاحظة مهمة جدا :- اداة ال Actskin4 . ocx وجميع الثيمات التي يتم وضعها في المشروع يجب ان تكون في حافظة واحدة (Folder) لكي يعمل بشكل صحيح .

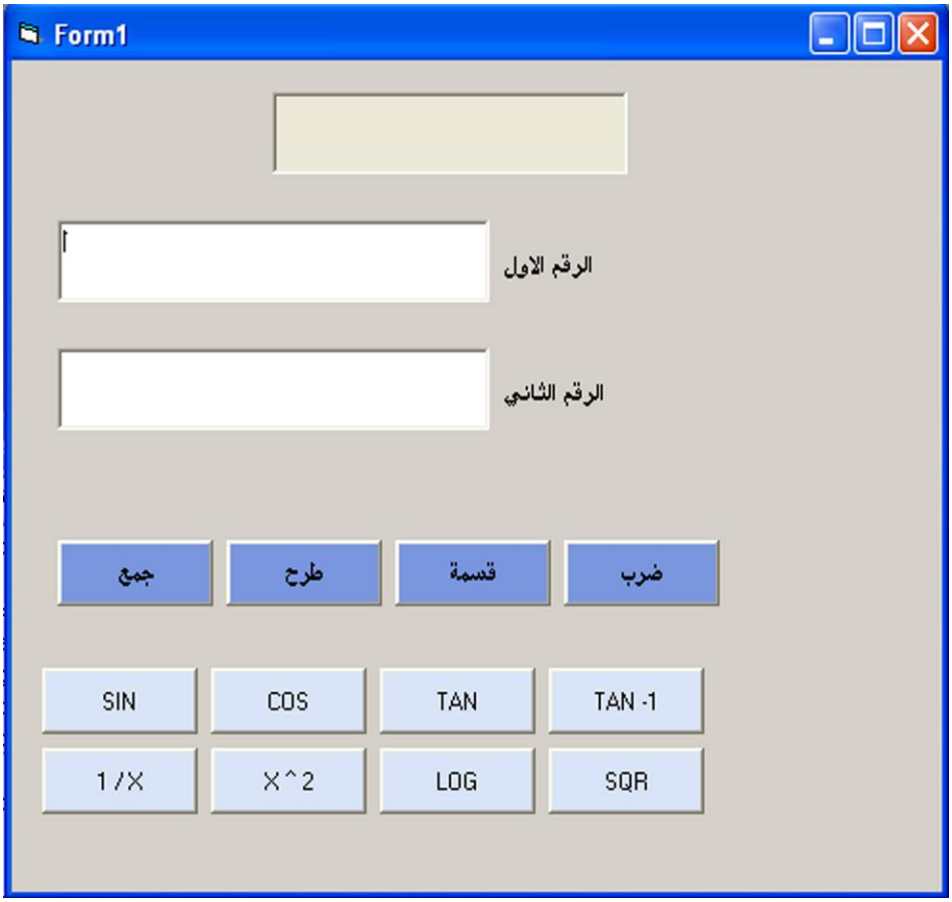
ويوجد اشكال جميلة من الثيمات ليعطي مشروعك منظرا جميلا كما مبين





عمل آلة حاسبة بسيطة جدا

الشكل العام



يجب تعريف الثابت في قسم التصاريح العامة بالشكل التالي :

```
Const pi = 22 / 7
```

اضف 3 حقول text
اضف 3 label
اضف 12 زر امر وسمهم كما موضح في الصورة اعلاه

الكودات

زر الجمع (+)

```
Label1 . Caption = Val(Text1.Text) + (Text2.Text)
```

زر الطرح (-)

```
Label1. Caption = Val(Text1.Text) - (Text2.Text)
```

زر القسمة (/)

```
Label1. Caption = Val(Text1.Text) / (Text2.Text)
```

زر الضرب (*)

```
Label1. Caption = Val(Text1.Text) * (Text2.Text)
```

زر ال (sin)

```
Label1. Caption = sin (val(Text1.Text) * pi / 180)
```

زر ال (cos)

```
Label1. Caption = cos(val (Text1.Text) * pi / 180)
```

زر ال (tan)

```
 Label1. Caption = tan(val (Text1.Text) * pi / 180)
```

زر ال (tan -1)

```
Label1. Caption = 1/ tan(val (Text1.Text) * pi / 180)
```

زر ال (log)

```
If Val(Text1.Text) >= 0 Then
Label1.Caption = Log(Val(Text1.Text)
Else
MsgBox " "
End If
```

زر ال (1/x)

```
Label1. Caption = 1/val (Text1.Text)
```

زر ال (X ^2)

```
Label1 . Caption = val (Text1.Text) ^ 2
```

زر ال(sqr)

```
If Val(Text1.Text) >= 0 Then
Libel1 . Caption = Sqr(Val(Text1.Text))
Else
MsgBox " "
End If
```

ويوجد انواع وطرق عديدة لعمل للحاسبة منها الاعتيادية و العلمية للمزيد من المعلومات يرجى التحميل من الانترنيت لزيادة المعلومات وكيفية كتابة الكودات .

رسم الاشكال الهندسية في الفيجوال بيسك

ر سم مستقيم	Line (x1,y1)-(x2,y2), color ,bf
رسم الدائرة	Circle(X,Y),Radius,Color
رسم المثلث	Line (x1,y1)- (x2,y2) ,color
رسم المربع	Line (x1,y1)-(x2,y2),color
رسم نقطة	pset(X,Y) ,Color
حركة الفورم	Move X1,Y1
مسح ما مرسوم ومطبوع على الفورم	(Cls) او (Me . Cls) او (form1 . Cls)
اظهار الفورم	Form1 . show
اخفاء الفورم	Form1 . hide

حساب محيط ومساحة دائرة
علماً أن قانون المساحة هو

Q = R^2* PI

حيث Q نقصد بها المساحة، و R^2 تعني نصف القطرمرفوع للقوة 2 أما Pi فهو ثابت يساوي 3.14

أما قانون المحيط فهو

A= 2*R * PI

مثال :

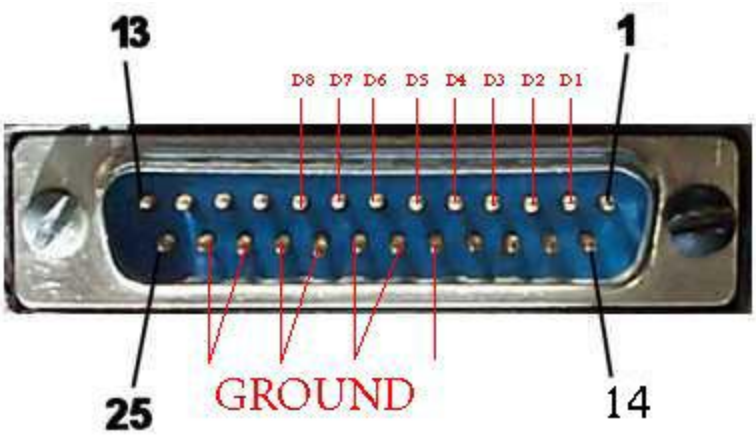
```
Dim R As Integer
Dim Area As Double, Perimeter As Double
Const Pi As Double = 3.14
(R = Val(Text1.Text
Area = R ^ 2 * Pi
Perimeter = 2 * R * Pi
Area & " المساحة" MsgBox
Perimeter & " المحيط" MsgBox
```

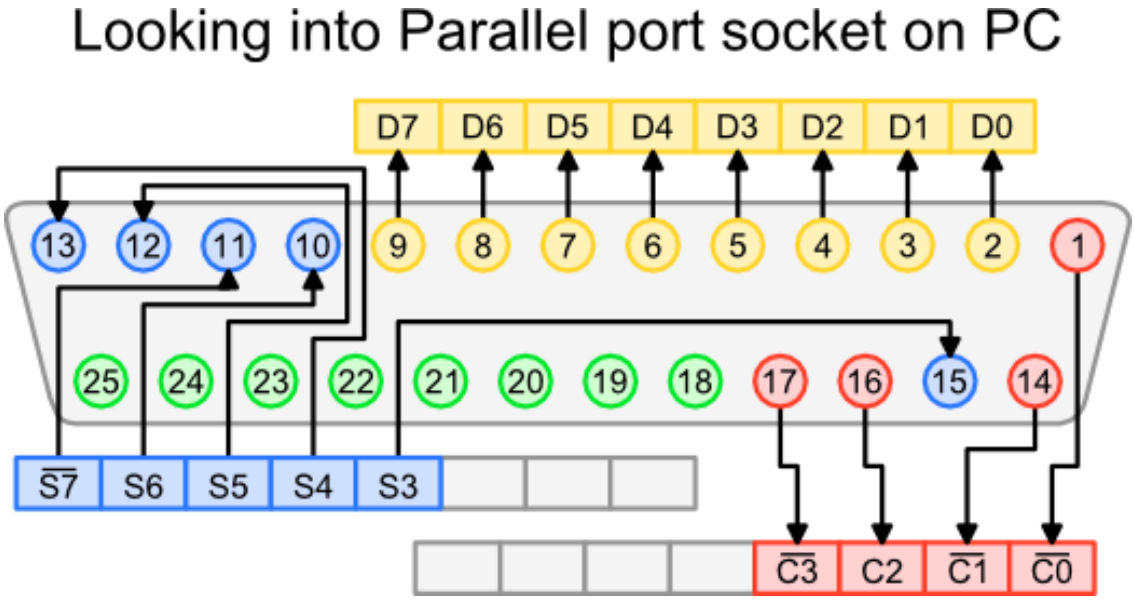
الفيجوال بيسك و دوره في التحكم

منفذ التوازي ذو 25 طرف Parallel port

وباختصار فإن منفذ الطابعة مكون من خمسة وعشرين Pin أو Hole حسب شكل المنفذ (بمعنى غاطس أو بارز)

تقوم بارسال 0 في حالة Close و 1 في حالة - Open حيث أن 1 هي قيمة تقترب من volte 5 في أغلب الأحيان





الرقم 1 + 14، 16، 17 تسمى بالـ Control Pins

الأرقام من 2 إلى 9 تسمى بالـ Data Pins

الأرقام من 10 إلى 13 + 15 تسمى بال. Statues Pins
الأرقام من 18 إلى 25 تسمى بالـ. Ground Pins

Control Pins :
عدد 8 من المنافذ ولكن 4 منها غير ظاهرة تستخدم للادخال والاخراج في نفس الوقت.

Data Pins :
هذه هي ال Pins هي الأشهر والأكثر استخداماً ، وتستخدم دائماً لعرض البيانات بعد توصيلها بجهاز مناسب ، وعددها 8 ، وفي الطابعات كانت تستخدم هذه المنافذ لنقل البيانات إلى الطباعة تمهيداً لعملية طباعتها.

Statues Pins :
تستخدم هذه المنافذ لادخال البيانات ، وعددها الفعلي هو 8 وليس خمسة ولكن هناك ثلاث منها غير ظاهرة ، وفي الطابعات كانت تستخدم لادخال بيانات من الطابعة إلى الكومبيوتر مثل رسائل اشعار انتهاء ورق الطابعة.

Ground Pins:
كأي دائرة كهربائية ، لا بد من وجود قطب موجب وقطب سالب لتعمل الدائرة ، ال Ground هو ما يشكل القطب السالب ، وفي العادة لا يتم استخدام سوى واحد فقط من ال 8 pins لجميع ال Pins الخاصة بالادخال والإخراج.

ارسال البيانات عبر منفذ الطابعة.

لارسال البيانات عبر منفذ الطابعة لا بد من تحديد رقم المنفذ اولاً وعادة ما يكون H387 ومن ثم ارسال قيمة ما بين 0 إلى 255 لتحديد ال Pins التي ستكون عليها القيمة 1 بالشكل التالي مثلاً:
لجعل جميع Pins مفتوحة = 11111111 بالعد الثنائي binary وهو ما يساوي 255
لجعل Pins الأول والثاني والرابع والسادس مفتوحة = 00101011 بالعد الثنائي binarty وهو ما يساوي 1+2+16+64 = 83 .

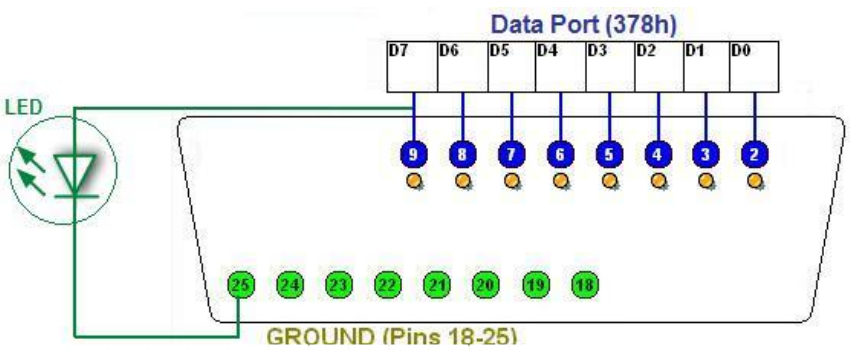
تجربة

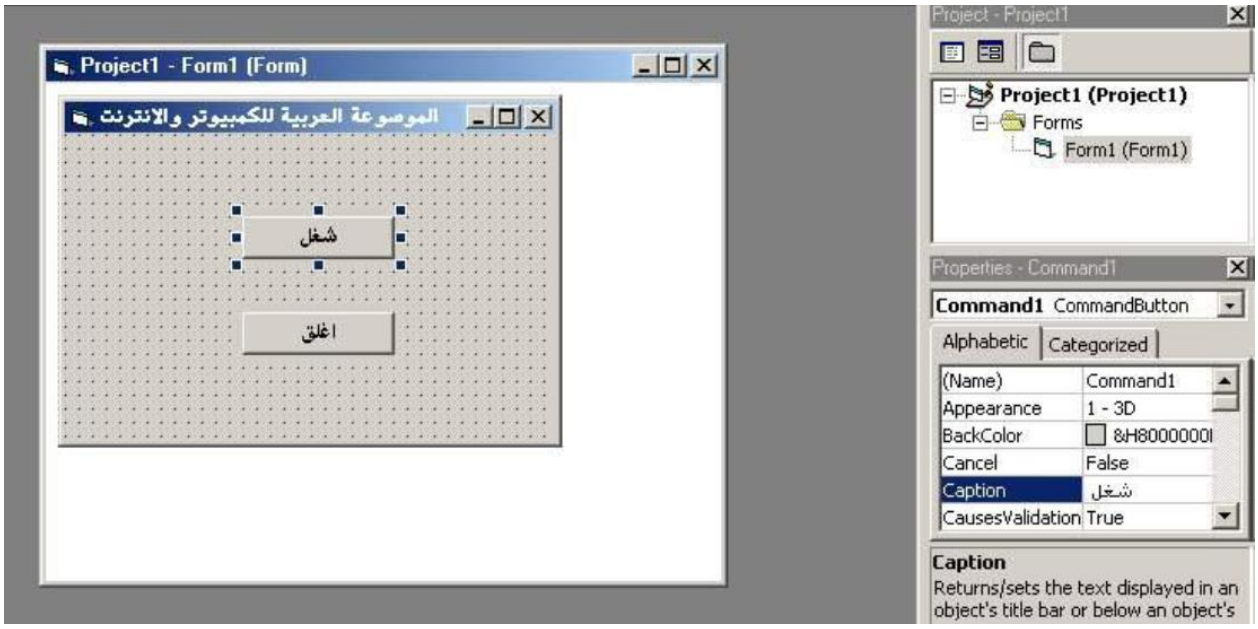
قم بشراء واحد من ال LED's من أي محل للالكترونيات ، وباختصار فال Leds هي عبارة عن انوار كهربائية بسيطة تنير عندما يمر فيها فرق جهد (5) فولت .

خطوتنا الثانية هي معرفة أن لل LED توصيلة للسالب وأخرى للموجب ، قم بتوصيل السالب بواحد من ال Grounds في منفذ الطابعة لديك وقم بتوصيل الموجب في أحد ال Data Pins وليكن الأول لديك ، ومن ثم قم بعمل برنامج سريع يقوم بفتح واغلاق هذا ال. Pin .

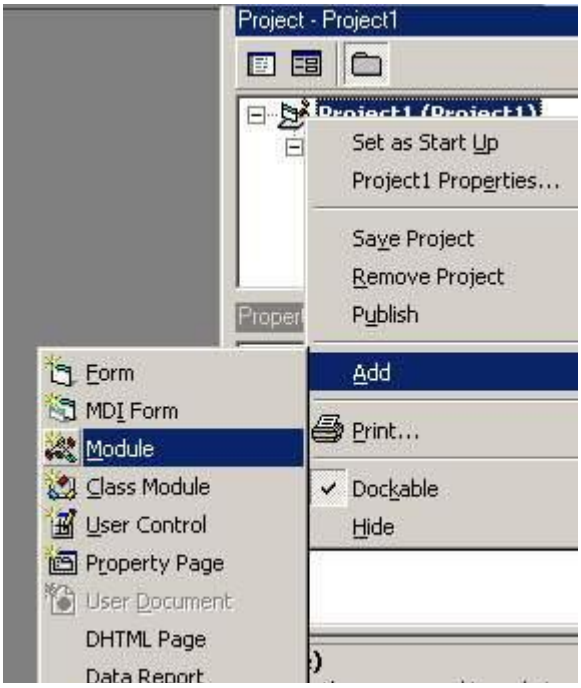
قم بفتح مشروع جديد

اضف الى الفورم الذي تراه امامك زرين امر (command buttons) وغير ال caption كما هو موضح في الصورة:

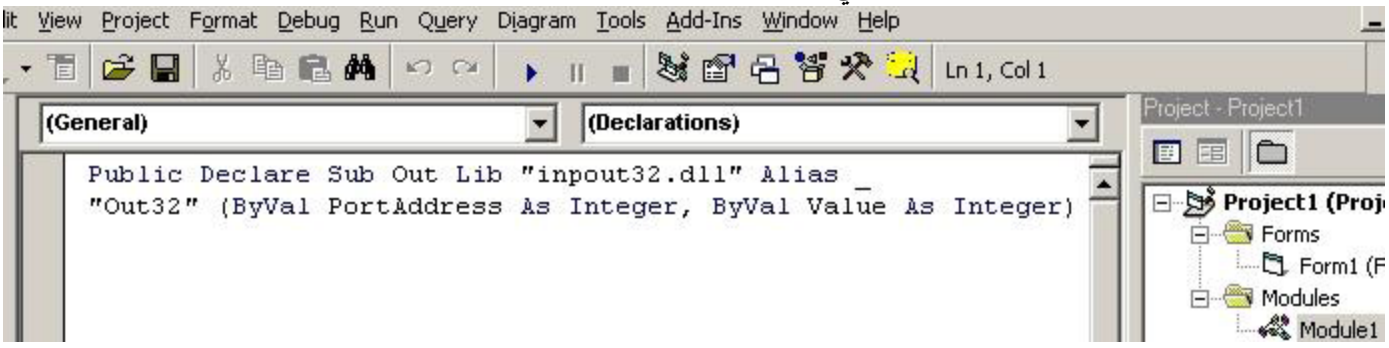




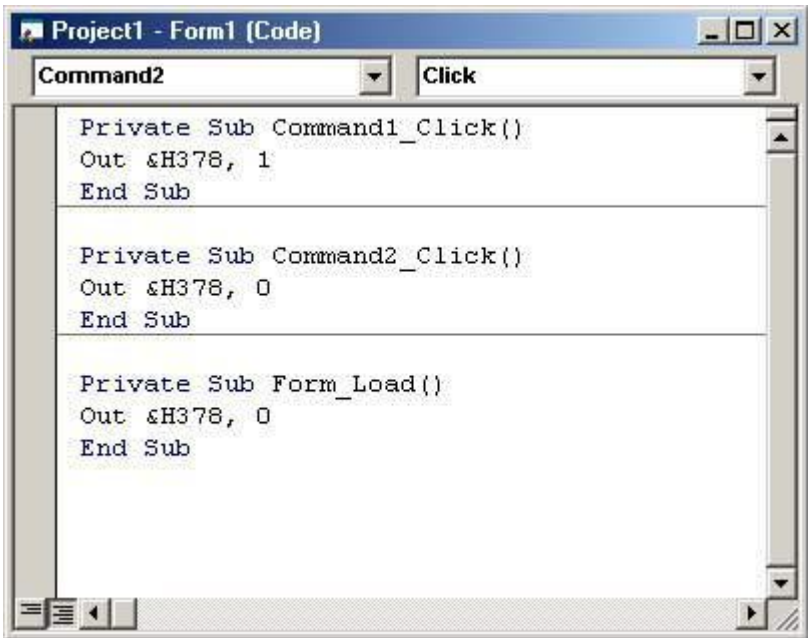
الان يجب اضافة **module** و هو بالضغط بالىمين على كلمة **project** في القائمة الموجودة على الىمين كما هو موضح في الصورة:



الان نكتب التالي في داخل ال **module**
يجب ان يكون الكود مشابه تماماً لما ترونه في الصورة التالية :



يفضل نسخ الجملة في الاعلى بل اطلب نسخة حتى لا تظهر مشاكل تعود لعدم كتابته بالطريقة الصحيحة .
عموما الكود الخاص بالتحكم سيكون كالتالي كما هو موضح في الصورة:




```
Public Declare Function Inp Lib "inpout32.dll"
Alias "Inp32" (ByVal PortAddress As Integer) As Integer
Public Declare Sub Out Lib "inpout32.dll"
Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As Integer)
```

ومن ثم في زر الأمر(فتح):

```
Out "&H378", 1
```

ومن ثم في زر الأمر (اغلاق) :

```
Out "&H378", 0
```

حيث يتم فتح واغلاق كامل ال Pins في كل خطوة.

ملاحظة: البورت او الحاسبة باكملة يتعامل مع الارقام الثنائية فقط اي (1,0) وان H378 هو اسم المنفذ في الحاسبة

كيف نخرج تيار كهربائي من بن اخر اليك الحل
اذا اردنا تشغيل البن الثاني نكتب رقم 1
اذا اردنا تشغيل البن الثالث نكتب رقم 2
اذا اردنا تشغيل البن الرابع نكتب رقم 4
اذا اردنا تشغيل البن الخامس نكتب رقم 8
اذا اردنا تشغيل البن السادس نكتب رقم 16
اذا اردنا تشغيل البن السابع نكتب رقم 32
اذا اردنا تشغيل البن الثامن نكتب رقم 64
اذا اردنا تشغيل البن التاسع نكتب رقم 128
اذا اردنا تشغيل البن الاول + الثاني نكتب رقم 3
اذا اردنا تشغيل البن الاول + الثالث نكتب رقم 5

اذا اردنا تشغيل البن الاول + الثاني + السادس نكتب رقم 33
النتيجة انه اذا اردنا تحديد بن واحد ناخذ رقم البن ونكتب بدلاً منه رقم (1) والبنات التي قبله بارقام الاصفار

مثال ::::

1 1 1 1 1 1 1 1 ارقام البنات من 1 الى 8
0 0 0 0 1

النتيجة تشغيل البن رقم (5) ونكتب رقمه في الكود (16)
طبعاً لا ننسى اننا حولنا الرقم الذي حصلنا عليه وهو (10000) من العد الثنائي الى العد العشري
اذا اردنا تحديد اكثر من بن واحد هذا مثال عليه

مثال ::::

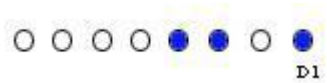
1 1 1 1 1 1 1 1 ارقام البنات من 1 الى 8
0 0 1 0 1

هذا المثال لتخريج تيار كهربائي من البن الثالث والخامس ونكتب قيمته بالكود (20)

BIT	D0	D1	D2	D3	D4	D5	D6	D7
VALUE	1	2	4	8	16	32	64	128

ماعلاقة الارقام الثنائية بالفولت و المنفذ

الصورة التالية مثلا، تبين ان الابر D1, D3 and D4 في وضع عمل ، او بمعنى اخر قيمتهما 5 فولت و البقية قيمتهم صفر فولت.

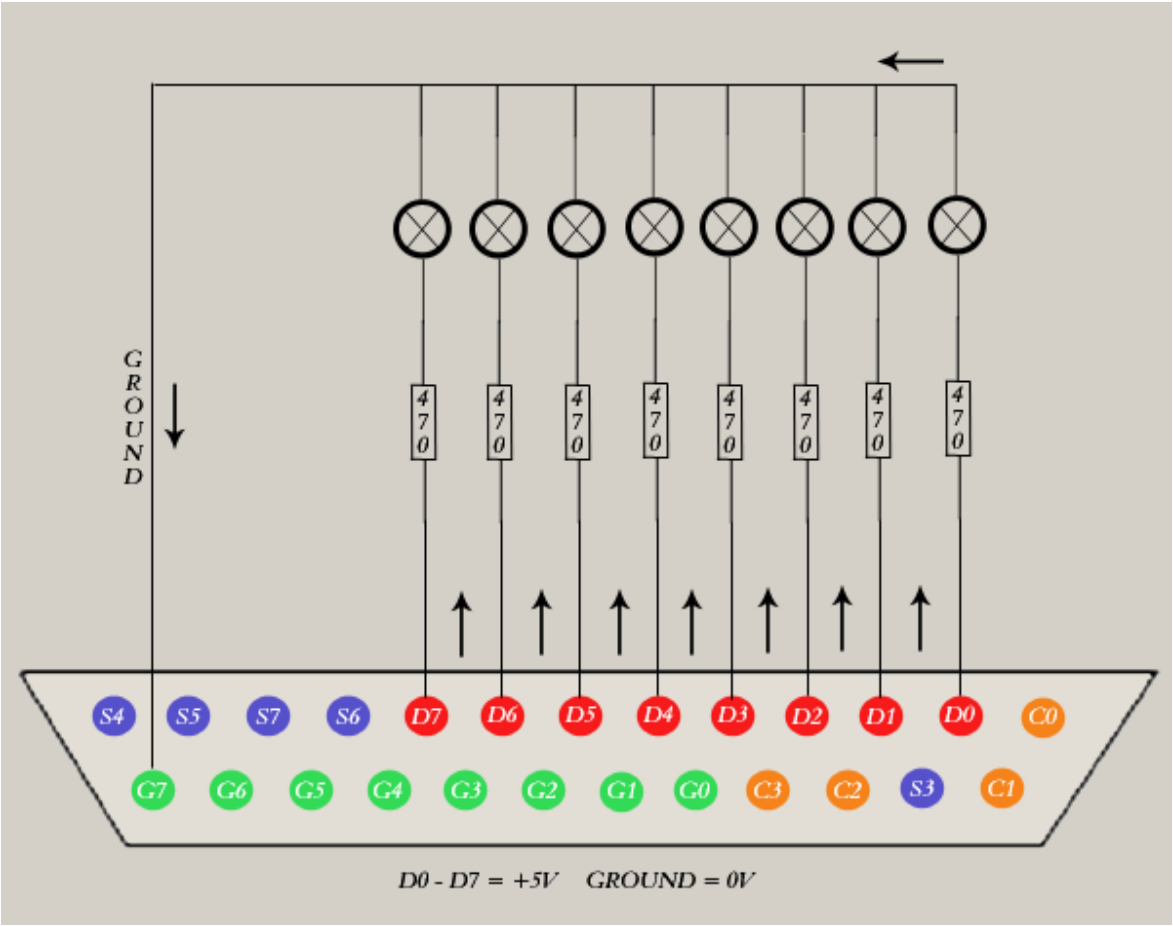


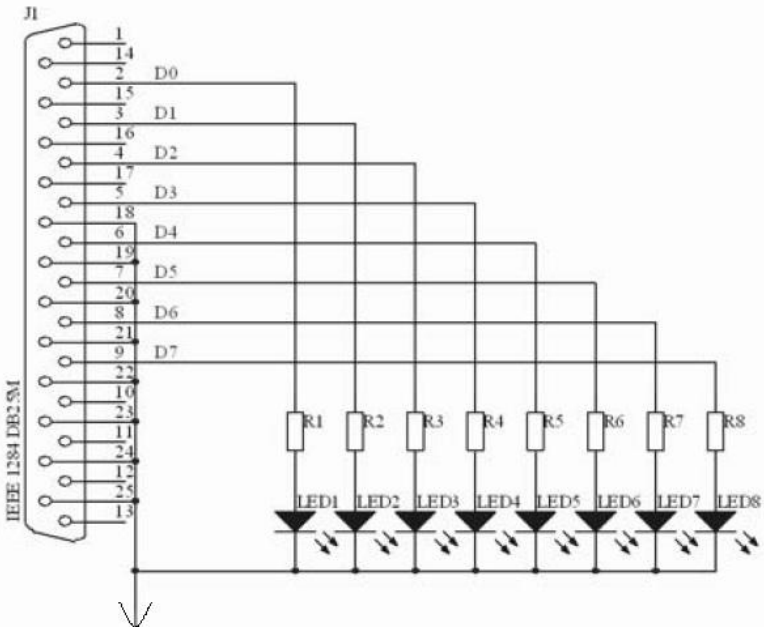
الان، يمكننا تحويل او تفسير الصورة الى الرقم التالي 00001101 الاصفار الاربعة نهملها لان لا قيمة لها ، فيصبح الرقم 1101 و بتحويله الي الرقم العشري فينتج لدينا الرقم 13 .

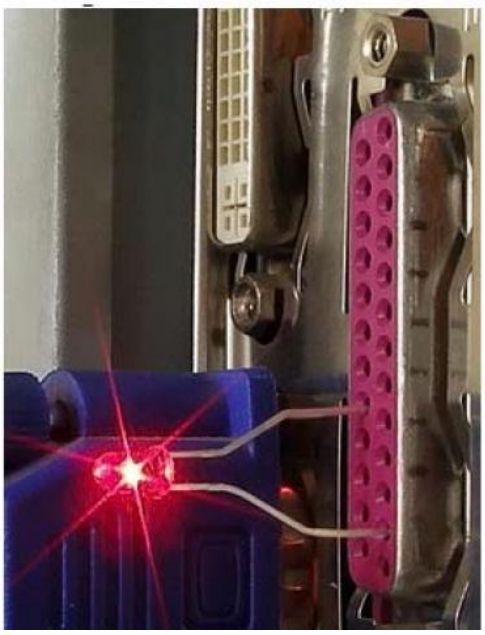
تحذير:

الرجاء التعامل بحذر مع القطع الكهربائية و التأكد الف مرة من انها ليست موصلة بالكهرباء حين العمل عليها و التأكد الف مرة ايضا بانك تعلم ماذا تفعل.
منفذ الطابعة قد يتلف او قد يؤدي الى اضرار عند سوء الاستخدام ، يرجى توصيل المنفذ بكيبل و تكون التجارب على الكيبل لا على المنفذ الملتصق باللوحة الام، مع اخذ الحيطة والحذر و شروط السلامة في كل خطوة تعملها .الكهرباء قد تكون صديقة لنا في كثيرمن الاحيان ،، لكنها قاتلة في كثير من الاحيان ايضا.

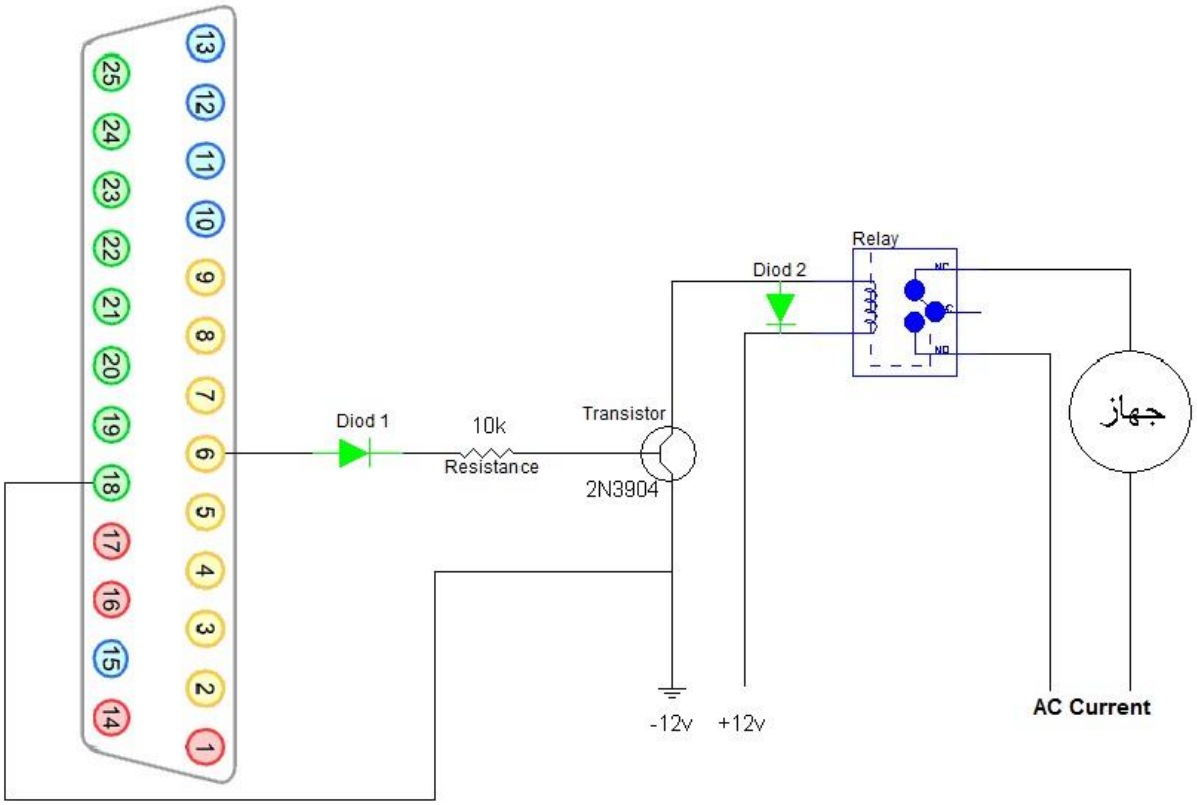
تنبيه : في حالة كونك تعمل على Windows XP فسوف تحتاج لاستخدام هذه المكتبة في الغالب بدلاً من الافتراضية inpout32.dll وهي موجودة في الانترنيت و ننسخها الى C:\Windows\System32.



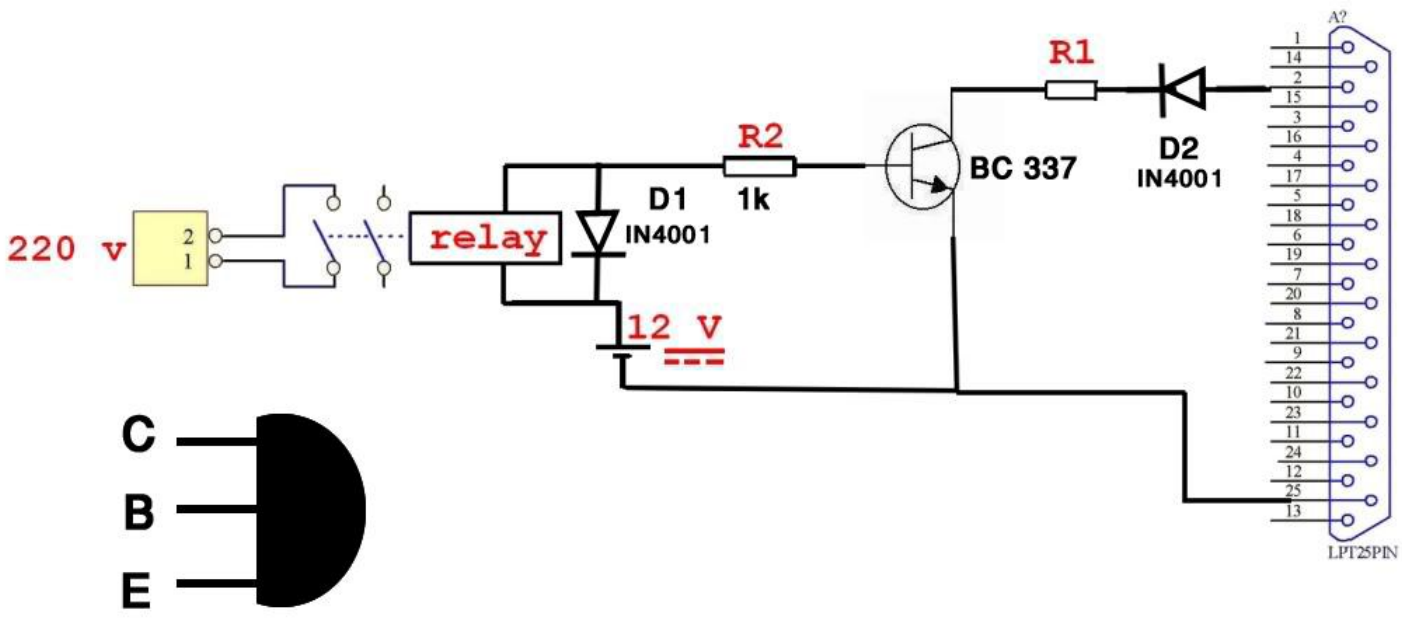




هذه طريقة اعتيادية للتحكم في مصباح صغير و الطريقة الاخرى هي بواسطة الريليه والتحكم بواسطته على اجهزة كهربائية مختلفة بفولتية عالية ولاكن يجب التاكد جيدا قبل تشغيل الدارة تاكيدا جيدا لنتفادى الاخطاء وعطب الوحة الام mather bord



واذا لم يتم العثور على الترانسيستور فهذا دارة ثانية



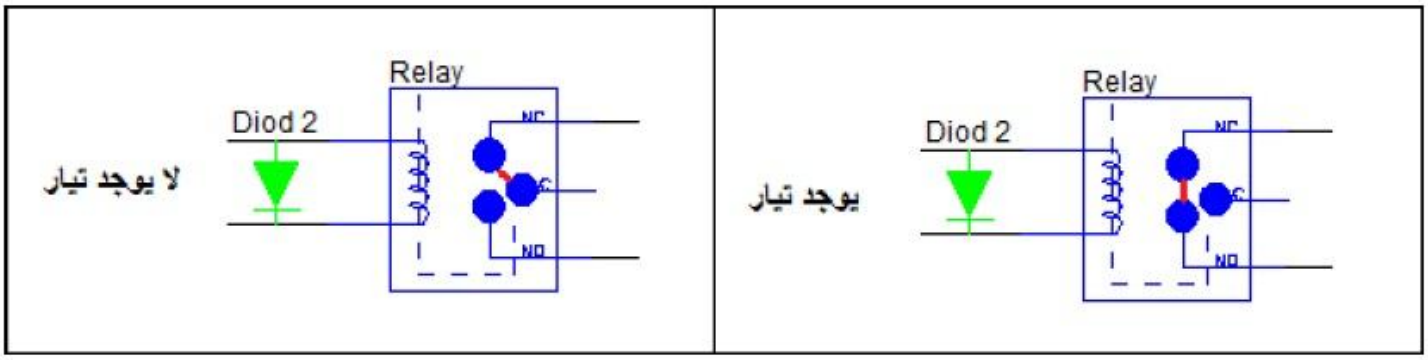
Diod1 يستخدم لحماية LPT من التيارات العكسية

R حماية الترانسيستور

TR عند استقبال 5V من الكومبيوتر يقوم بتمرير المصدر الثاني للكهرباء وهو هنا 12V وذلك لتشغيل ملف ال relay ونقوم باستخدام المصدر الثاني للكهرباء بناءا على قيمة الملف في ال relay فمن الممكن ان يكون 6V, 9V, 12V , 24V

Diod 2 لحماية الترانسستر من التيارات العكسية ، كما انه يحمي الجهاز عند ضعف الامبير في مصدر الطاقة الثانية 12V في حال كان بطارية

RELAY عند استقبال التيار الكهربائي (12V) يقوم بجذب الذراع الداخلي به فيتصل طرفيه ببعض ليسمح بمرور التيار ، مثل الصورة التالية
(انظر للخط الاحمر وهو التوصيل)



وطريقة اخرى للتحكم هي باستخدام IC الدوائر المتكاملة ولاكن لن نتطرق اليها .

Mmmmm97@yahoo.com

engineerjhz ele (facebook)